

---

## **CONTRATO RELATIVO A LA PRESTACIÓN DE LOS SERVICIOS DE MANTENIMIENTO Y EVOLUCIÓN DE APLICACIONES DE LA UNIVERSIDAD OBERTA DE CATALUNYA**

---

### **PLIEGO DE PRESCRIPCIONES TÉCNICAS**

---

**Expediente HSE00017/2025**

# ÍNDICE

<b>1. Objecte del Contracte</b>	<b>5</b>
1.1. Model de gestió operativa de les TIC	9
<b>2. Objectius del Contracte</b>	<b>16</b>
<b>3. Descripció dels serveis</b>	<b>21</b>
3.1. Visió global del servei	21
3.1.1 Serveis recurrents	21
Manteniment d'aplicacions: Correctius, adaptatius i preventius	22
Suport tècnic i operatiu sobre les aplicacions	23
Guàrdies i actuacions fora d'horari laboral / Suport funcional estès / Atenció especial en activitats clau	24
Millora continua	24
Gestió operativa	25
Manteniment de les Plataformes Funcionals	26
3.1.2 Serveis addicionals sota demanda	27
Desenvolupament sota demanda: Petit evolutiu	27
Desenvolupament sota demanda: Projecte Evolutiu	28
Condicions per la valoració i execució de projectes evolutius	30
3.1.3 Suport en eines basades en Intel·ligència Artificial (IA)	31
3.2. Volumetria i dimensionament dels Serveis Recurrents	33
3.3. Volumetria i dimensionament dels Serveis Sota Demanda	41
<b>4. Condicions d'execució del servei</b>	<b>43</b>
4.1 Model de relació	43
4.1.1 Estructura de responsabilitats	43
4.1.2 Òrgans de Govern	46
4.2 Mecanismes de control i reporting	49
4.2.1 Acords de Nivell de Servei (ANS) del servei recurrent	52
4.2.2 Sistema de bonificacions del Acords de Nivell de Servei	56
4.2.3 Pla d'indicadors	61
4.2.3.1 Indicadors de Gestió de Serveis Recurrents	61
4.3 Metodologia, estàndards i lliurables	65
Annex - Normativa de Seguretat	66
4.4 Entorns de desenvolupament	67
4.5 Desenvolupaments "Cloud-Ready"	67

4.6 UX - experiència d'usuari i accessibilitat	68
4.6.1 Investigació amb usuaris	68
4.6.2 Conceptualització UX	69
4.6.3 Disseny gràfics (UI)	70
4.6.4 Front end	71
4.6.5 Accessibilitat	72
4.6.6 Automatització i eficiència en les proves QA d'UX i accessibilitat amb IA	73
4.6.7 Manteniment, evolució i governança del sistema de disseny i del UI Kit institucional	74
4.7 Qualitat	74
4.8 Eines	76
4.9 Auditòries	77
4.10 Perfil i funcions	78
4.11 Garantia dels desenvolupaments	88
4.12 Gestió de l'obsolescència tecnològica	89
4.13 Govern de les dades	89
4.14 Adequació a les variacions del servei	90
4.15 Altres condicions del servei	91
4.15.1 Horari del servei recurrent	91
4.15.2 Ubicació del servei	91
4.15.3 Calendari de treball	91
4.15.4 Desplaçaments	91
4.15.5 Comunicacions	92
4.15.6 Idioma	92
<b>5. Fases de la prestació del servei recurrent</b>	<b>93</b>
5.1 Fase de Transició del Servei	93
5.1.1 Model de Transició del Servei	93
5.2 Fase regular del servei recurrent	98
5.3 Fase de devolució del servei	98
5.3.1 Model de devolució	98
<b>6. Contingut i estructura de l'oferta</b>	<b>99</b>
<b>Annex - Sistemes del Lot A</b>	<b>101</b>
<b>Annex - Sistemes del Lot B</b>	<b>111</b>
<b>Annex - Sistemes del Lot C</b>	<b>128</b>
<b>Annex - Sistemes del Lot D</b>	<b>135</b>
<b>Annex - Sistemes del Lot E</b>	<b>136</b>
<b>Annex - Principis de Tecnologia</b>	<b>139</b>
Principis estratègics	139

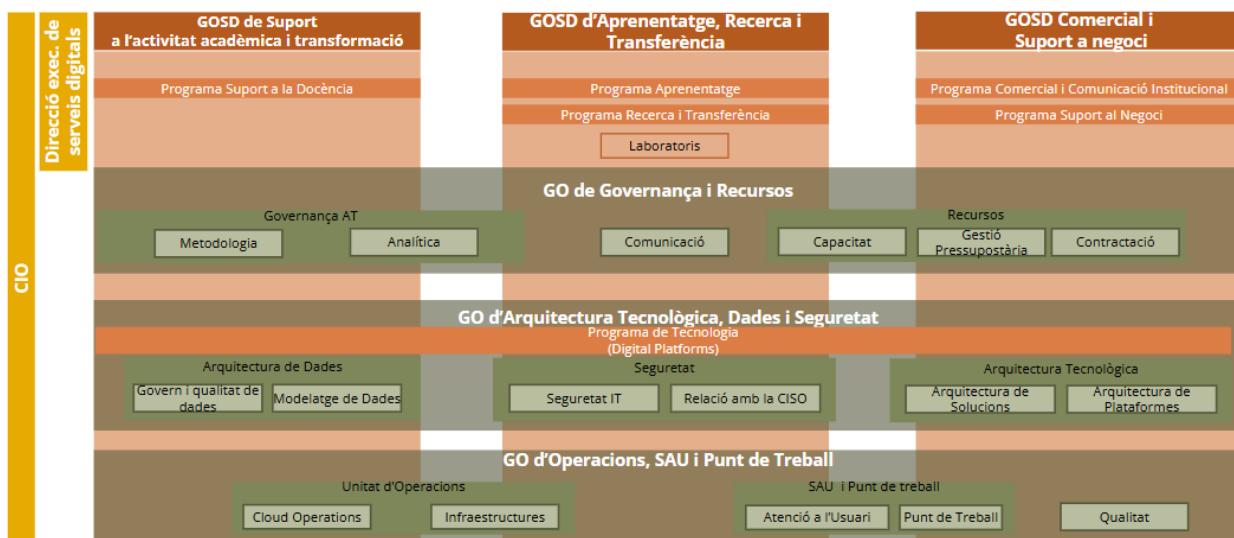
Principis sobre el disseny d'aplicacions	142
Principis sobre la tecnologia	143
Principis sobre el cost i manteniment de les solucions	144
<b>Annex - Normativa de lliurables</b>	<b>145</b>
Introducció	145
Propòsit	145
Audiència	145
Normatiu	146
<b>Annex - Normativa de Qualitat</b>	<b>168</b>
Introducció	168
Abast	168
Propòsit	168
Audiència	168
Indicadors Normatius	168
Incompliment de la Normativa	175
Control d'aplicació de la Normativa	175
Eines i documents	175
<b>Annex - Normativa sobre UX i Accessibilitat</b>	<b>176</b>
UX	176
Accessibilitat	184
<b>Annex - Normativa DevOps</b>	<b>190</b>
<b>Annex - Normativa de Seguretat</b>	<b>198</b>
Plataformes SaaS	198
Desenvolupaments i configuracions	199
Protecció de dades personals	205
Protecció WAF (Web Application Firewall)	205
Desenvolupament segur del software	205
<b>Annex - Full de Ruta de Programari</b>	<b>210</b>

# 1. Objeto del Contrato

La Universitat Oberta de Catalunya (en adelante, “la UOC”) es una universidad innovadora, arraigada en Cataluña y abierta en el mundo, que forma las personas a lo largo de la vida contribuyendo a su progreso y al de la sociedad, a la vez que lleva a cabo investigación sobre la sociedad del conocimiento. Su modelo educativo se basa en la personalización y el acompañamiento del estudiante mediante el e-learning; en este sentido, se trata de la primera universidad a la red en el estado español.

La UOC quiere ser una universidad que, conectada en red con el resto de universidades del mundo, impulsa la construcción de un espacio global de conocimiento y la investigación frontera en sociedad del conocimiento. Innova en el modelo educativo propio que se centra en el estudiante, ofreciendo una formación de calidad y personalizable, para fomentar su competitividad y contribuir al progreso de la sociedad.

Dentro de la universidad, el Área de Tecnología vela por la gobernabilidad de los Sistemas de Información que se agrupan en Servicios Digitales y en Programas, todos gestionados bajo un Grupo Operativo tal y como se muestra a continuación:



- **CIO:** Es el responsable de la gestión y la dirección de la estrategia de tecnología de la información, que incluye:
  - Desarrollo de la estrategia tecnológica: El CIO tiene que alinear la tecnología con los objetivos comerciales de la empresa.
  - Gestión de sistemas de información: Supervisar la implementación, el mantenimiento y la actualización de sistemas y aplicaciones informáticas.
  - Seguridad de la información: Garantizar la seguridad e integridad de los datos de la empresa.

- Gestión de proyectos tecnológicos: Supervisar proyectos tecnológicos desde la concepción hasta la implementación.
- Innovación tecnológica: Evaluar nuevas tecnologías y su aplicabilidad a la empresa.
- Gestión de equipos: Liderar y gestionar equipos de profesionales de TI.
- Colaboración interdepartamental: Trabajar con otros líderes empresariales para asegurar que la estrategia tecnológica esté alineada con los objetivos generales de la organización.
- Dirección Ejecutiva de Servicios Digitales:
  - Coherencia y consistencia: Es el responsable de garantizar la coherencia y la consistencia del modelo de servicio it para toda la UOC, las economías de escala, la gestión del riesgo táctico y operativo y la coordinación de los GOSDs preservando su autonomía y la agilidad.
  - Impulso de la transformación ágil: También es el responsable de impulsar, implantar y mejorar de forma continua el modelo de gobernanza y agilidad sobre los programas, maximizando el retorno de valor a la organización.
  - Proyectos de transformación cross-programa: Asegurar la ensambladura de los grandes proyectos de transformación que tienen un carácter transversal (cross-programa).
  - Gobierno transversal de los recursos y operativa: Gobierna de forma transversal los recursos y la operativa necesaria para garantizar la ejecución del plan TIC (Pla de Proyectos Anual) y la prestación de servicios IT al negocio de forma consistente, eficiente y eficaz.
- Grupos Operativos de Servicios Digitales:
  - Responsabilidad operativa extremo a extremo: Visión única extremo a extremo de la cadena de demanda, planificación, ejecución y operación, velando por la calidad y sostenibilidad de su portafolio de servicios.
  - Identificación de oportunidades y riesgos: Identificación de oportunidades y mitigación de riesgos que permitan la elaboración de un roadmap que maximice el retorno de valor a la organización.
  - Máxima interlocución tecnológica en los comités de programa: Máxima representación tecnológica dentro de los comités de programa y garante del alineamiento entre negocio y tecnología.
  - Gestión integral de los recursos y la capacidad: Gestiona con visión integral los recursos económicos y humanos para garantizar la operativa asignados a su ámbito, y garantiza la coordinación con el resto de GO transversales.
- Grupo Operativo de Gobernanza y Recursos:
  - Impulsa el liderazgo operativo en la transformación ágil del área, y realiza las funciones transversales de gobierno a través del seguimiento y reporting global, del impulso en el uso y aplicación de las herramientas y metodologías, y la gestión del marco de aprovisionamiento común de servicios y recursos.
- Grupo Operativo de Arquitectura Tecnológica, Datos y Seguridad:

- Despliega la función transversal de diseño y entrega de la estrategia tecnológica del área integrada con la visión de seguridad. Aporta valor directo al negocio a través de su participación en la actividad de los GO de Servicios Digitales, y mediante la implantación y evolución de las plataformas digitales que conforman el horizonte IT de la UOC de las cuales es responsable extremo a extremo.
- Despliega la función transversal de diseño y entrega de la estrategia tecnológica alrededor del gobierno, la gestión y el uso de los datos, con el objetivo dotar de los servicios tecnológicos personalizados y eficientes necesarios porque la gestión y uso de los datos se convierten en un activo estratégico para la UOC y así conseguir una universidad orientada a los datos.
- Grupo Operativo de Operaciones, SAU y Punto de Trabajo:
  - Asegura la calidad, disponibilidad y sostenibilidad de los servicios de infraestructura y cloud, mediante la gestión, la supervisión y la mejora continua de los mismos y de acuerdo con las necesidades de negocio.
  - Responsable del servicio de atención al usuario de Tecnología, y de la transformación y aprovisionamiento del punto de trabajo extremo a extremo.

El objeto de la presente licitación es la contratación de los servicios de mantenimiento y evolución de los sistemas de información de la UOC que se especifican en este pliego de prescripciones técnicas, y se incluye tanto servicios recurrentes (mantenimiento correctivo, apoyo y operación), como servicios de desarrollo evolutivo sobre los sistemas.

Los servicios, a pesar de que están relacionados, constituyen cada uno una unidad funcional independiente, de forma que se ha previsto la realización individual de cada servicio mediante los siguientes lotes:

LOTE A: Mantenimiento y Evolución de

- Sistemas del Programa de Aprendizaje

basados en las siguientes Tecnologías y Plataformas Funcionales:

- Canvas (Instructure)
- LTI (Learning Tools Interoperability) con Canvas
- Aplicaciones desarrolladas sobre Outsystems (LowCode)
- Tecnología Web (HTML, CSS, Java y Java Script, Ruby, PHP, Phyton) tradicional y Cloud Native

LOTE B: Mantenimiento y Evolución de:

- Sistemas del Programa de Apoyo a la Docencia
- Sistemas del Programa de Apoyo al Negocio

basados en las siguientes Tecnologías:

- PL/SQL y Oráculo Forms
- Tecnología Web (HTML, CSS, Java y Java Script, Ruby, PHP, Phyton y Go) tradicional y Cloud Native
- Legacy BI: Azure DataFactory + PBI, Oráculo Discoverer, Tableau, PL/SQL y Talend

LOTE C: Mantenimiento y Evolución de:

- Sistemas del Programa Comercial y Comunicación
- Sistemas del Programa de Apoyo a la Docencia

basados en las siguientes Tecnologías y Plataformas Funcionales:

- DXP (Plataforma de Experiencia Digital de Adobe)
- Wordpress
- Liferay
- OpenCMS
- OneTrust (Gestión de Cookies)
- Tecnología Web (HTML, CSS, Java y Java Script, Ruby, PHP, Phyton y Go) tradicional y Cloud Native

LOTE D: Mantenimiento y Evolución de:

- Sistemas del Programa Comercial y Comunicación

basados en las siguientes Tecnologías y Plataformas Funcionales:

- Salesforce CRM sobre EDA (Education Fecha Architecture)
  - Salesforce Marketing Cloud
  - Salesforce Sales Cloud
  - Salesforce Service Cloud

LOTE E: Mantenimiento y Evolución de:

- Sistemas del Programa de Apoyo al Negocio
- Sistemas del Programa de Tecnología y Seguridad

basados en las siguientes Tecnologías y Plataformas Funcionales:

- JIRA Software Fecha Center (Atlassian)
- JIRA Service Management Fecha Center (Atlassian)
- Confluence Fecha Center (Atlassian)
- SonarQube

- XRAY (pluggin de JIRA)

A los Anejos “Anejo - Sistemas del Lote” se describen los sistemas y las funciones más relevantes de cada lote.

La relación de sistemas y aplicaciones asociadas a cada lote se revisarán con una periodicidad anual.

La relación de sistemas de cada lote varía en el tiempo, por la propia gestión de la obsolescencia, y la ejecución de proyectos de transformación que proporcionan nuevos sistemas que se tendrán que incorporar a los mantenimientos. Las modificaciones de los sistemas implicados no podrá suponer una variación de la línea baso del recurrente superior al 20 % (como se indica en el apartado 3.2 de volumetría y dimensionado).

No obstante lo anterior, el listado concreto de tecnologías podrá variar si durante la ejecución del contrato hay cambios en este sentido.

## 1.1. Modelo de gestión operativa de las TIC

El objeto del contrato se enmarca en un proceso interno de transformación del modelo de gestión operativa de las TIC a la UOC.

Este nuevo modelo operativo, basado en programas estables y programas temporales de transformación, se enfoca a:

- La reducción del tiempo que transcurre entre la detección de la necesidad y el despliegue de la solución. Minimizando así, los riesgos de pérdida de oportunidad o de incumplimiento legal.
- Facilitar las modificaciones a lo largo del ciclo de vida de proyecto, buscando el alineamiento entre la necesidad planteada y la solución aportada.
- Establecer los mecanismos de demanda continua por la priorización de necesidades con Gestión y la Academia.

### Características del modelo - Aspectos Fundamentales

- El registro de iniciativas es continuo y se estructura por Programas e iniciativas Estratégicas. Los Programas se corresponden a procesos de la cadena de valor, y las Iniciativas Estratégicas a proyectos de gran alcance y transformadores por la UOC.
- Se crean Comités de Programa (con participantes de Gestión, Academia y Tecnología) para priorizar las necesidades existentes en cada Programa, seguir la ejecución y responsabilizarse de los resultados.
- Los Equipos de Programa tienen 3 roles/ funciones clave, Business Owner, Business Partner (referente de las áreas y Estudios) y Program Owner, con responsabilidad compartida en el logro

de objetivos. Además, cada programa tendrá una figura de esponsor que garantiza el alineamiento estratégico.

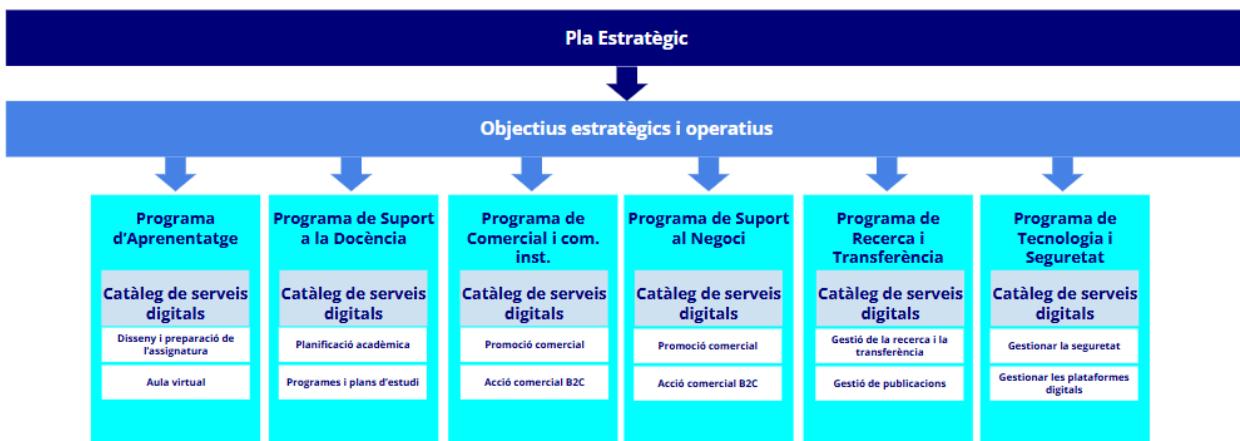
- Business Owner:
  - Responsable de definir la visión y los objetivos de un producto o servicio.
  - Supervisa el desarrollo y la evolución del producto para asegurar su valor para el negocio.
  - Prioriza las funcionalidades o iniciativas para maximizar el retorno de la inversión.
  - Trabaja de cerca con equipos de desarrollo, operaciones y estrategia.
- Business Partner
  - Actúa como enlace entre las unidades de negocio y las áreas operativas (como IT, finanzas, recursos humanos).
  - Ayuda a traducir las necesidades del negocio en soluciones estratégicas.
  - Participa en la toma de decisiones basada en análisis y alineación con los objetivos generales de la empresa.
  - Tiene una visión más transversal del negocio y facilita la comunicación entre equipos.
- Program Owner
  - Responsable de la gestión y ejecución de un programa (una iniciativa compleja con múltiples proyectos).
  - Supervisa la coordinación entre los diferentes equipos y proyectos para garantizar cohesión.
  - Asegura que el programa cumpla objetivos estratégicos y operativos.
  - Controla los riesgos, la calidad y el calendario de ejecución.
- El Comité de Procesos, Tecnología y Datos establece de acuerdo con criterios estratégicos la dotación presupuestaria, y valida los objetivos e indicadores que se asignan al desarrollo de cada Programa.
- El Consejo de Dirección asegura el alineamiento entre la estrategia de la UOC y la asignación de recursos a las Iniciativas Estratégicas.
- Los Comités de Programa pueden gestionar el presupuesto restando (no asignado a iniciativas blindadas) a otras iniciativas que estén alineadas con los objetivos establecidos.
- La metodología de desarrollo es preferentemente ágil, con ciclos cortos de desarrollo y construcción incremental de los productos.
- Después de cada ciclo de desarrollo se evalúan los resultados y las prioridades existentes, y el Equipo de Programa define las prioridades a abordar en el siguiente ciclo.

## Estructura de Programas e iniciativas Estratégicas



## Estructura de Servicios Digitales

Se crean los Servicios Digitales para definir los objetivos conjuntamente con el negocio, fomentando la corresponsabilidad negocio/tecnología en base a un trabajo en equipo y una exigencia continua.



## Servicios Digitales

- Es el eje de unión entre negocio y tecnología, a través del cual se pivotará la relación y comunicación, con una visión extremo-extremo.
- Es un lenguaje común por negocio y está ligado a los procesos y áreas de negocio de la uoc.
- un servicio digital tiene un responsable de negocio y un responsable dentro del área de tecnología (grupos operativos de servicios digitales GOSD).
- Es un medio para la creación conjunta de valor al facilitar los resultados que las áreas de la uoc quieren lograr.
- Un servicio digital es responsabilidad de un Programa, puede dar respuesta a diferentes procesos de negocio y puede asociarse a diferentes soluciones.
- Los servicios digitales pueden tener diferentes niveles de criticidad y se pueden identificar diferentes actividades críticas de la UOC.

#### Responsable de Negocio

- Es el responsable de la concreción y de la priorización del conjunto de iniciativas de transformación del servicio digital al que está vinculado para lograr los objetivos del programa.
- En el modelo franquiciado de gestión de las TIC, son los responsables de la gestión y coordinación del último tramo de entrega de las soluciones TIC, conjuntamente con los partners tecnológicos.

#### Responsable Servicios Digitales

- Es el responsable de la definición, la gestión, la sostenibilidad, los resultados y la calidad en la prestación de los servicios.
- Conecta todos los elementos que componen los servicios y vincula con las actividades y procesos de la organización, con las partes interesadas.

#### Servicios Digitales por Programa

##### Aprendizaje

- Diseño y preparación de la asignatura
- Seguimiento de la asignatura (learning analytics)
- Aula virtual
- Comunicación en el aula
- Colaboración en el aula
- Vídeo en el aula
- Herramientas específicas de aprendizaje
- Prácticas virtuales
- Evaluación en el aula
- Experimentación para la docencia

##### Comercial y Comunicación

- Acción institucional

- Gobernanza ecosistema digital
- B2B - Gestión de contenidos Web
- Servicio lingüístico
- Gestión de actividad y protocolo
- Gestión de la Innovación
- Trans - Gestión derechos digitales (COMO SD5)
- B2C - Explotación y análisis de datos
- Gestión CRM B2C (COMO SD4 )
- [OBsoleto] B2C - Gestión de acontecimientos
- B2B - Colaboraciones Corporativas
- B2B - Marketing y Matrícula Corporativa
- [OBsoleto] B2B\_Gestión de acontecimientos
- [OBsoleto] B2B - Gestión y explotación datos
- [OBsoleto] B2B- Atención al cliente
- Gestión Cookies (COMO SD6)
- [OBsoleto] Alumni (COMO SD7)
- B2C - Gestión de Contenidos Portal Institucional (excepto OF)
- B2C - Gestión de Contenidos portal Ofrecida Formativa

#### Apoyo a la Docencia

- Gestión matrícula
- Biblioteca
- Producción Recursos de aprendizaje
- Gestión Recursos de aprendizaje
- Gestión acción docente
- Convalidación y reconocimientos
- Operaciones académicas
- Programas y planes de estudio
- Oferta académica
- Acceso académico
- Expediente académico
- Certificados, Acreditaciones y Titulación académica
- Evaluación digital
- Identidad y autoría a las PAFs
- Tutoría
- Prácticas y movilidad
- Portafoli académico
- Provisión de datos oficiales
- Carrera profesional
- Apoyo al estudiante

- Calidad de las titulaciones (MVSMA)
- Contenidos académicos e interacción con el usuario
- Apoyo a SGIQ

#### Apoyo al Negocio

- Compliance
- Gestión económica y financiera
- Gestión de cobros, pagos y facturación a clientes
- Administración Electrónica
- Archivo, Gestión Documental y Registro
- Servicios transversales
- Administración Equipo propio
- Gestión de PDC
- Desarrollo del talento
- Prevención y Salud Integral
- Apoyo a la selección y formación Equipo Propio
- Selección equipo propio
- Relaciones laborales y compensación
- Gestión de los datos del Área de Personas
- Infraestructura de equipaciones
- Gestión de datos de Gerencia
- Jurídico Contractual
- Jurídico Universitario
- Protección de Datos
- Sistemas de Apoyo de Gestión

#### Investigación y Transferencia

- Gestión de la Investigación
- Servicios científico-técnicos
- Ciencia Abierta y Transferencia
- Experimentación para la Investigación y la Innovación

#### Tecnología y Seguridad (Servicios Digitales Internos de Tecnología)

- Gobierno y competencia en UX
- Capacidad
- Calidad operativa
- Gestión del punto de trabajo
- Service Desk
- Apoyo a programas y proyectos
- Gestión de incidentes críticos y problemas

- Gobierno y gestión del Cloud
- Monitoreig y observabilitat
- Administración de infraestructura
- Aprovisionamiento y despliegue
- Servicios y procedimientos transversales
- Compliance del Área de Tecnología
- Gestión presupuestaria
- Gestión de la contratación
- Newsletter
- Comunicación Interna
- Comunicación Institucional
- Modelo operativo y procedimientos internos
- Centro de Competencia de metodología
- Planificación de ejecución del Portafoli
- Gobierno de los servicios de mantenimiento transversales
- Herramientas de seguimiento de ejecución de los AM's
- Centro de competencia de servicios de mantenimiento y desarrollo
- Reporting directivo
- Dashboards de gestión del Área
- Gobierno de los datos de gestión del Área de Tecnología
- Datos
- IAM
- Workspace
- Automatización
- Interoperabilidad
- LowCode / NoCode
- Interoperabilidad. Apoyo en el diseño de soluciones a programas y proyectos
- Gobierno de la arquitectura de interoperabilidad
- Datos. Apoyo en el diseño de soluciones a programas y proyectos
- Gobierno de la arquitectura de datos
- Identity & Access Management
- Vulnerability Management
- Apoyo de seguridad en el diseño de las soluciones
- Gobierno y gestión de la seguridad IT
- Apoyo en la gestión de incidentes críticos y problemas
- Diseño de soluciones tecnológicas
- Estrategia y gobierno de la tecnología y los datos
- Herramientas de apoyo a la gestión

## 2. Objetivos del Contrato

En el desarrollo del objeto del presente contrato se pretende lograr, entre otros, los siguientes objetivos:

- Asegurar el funcionamiento ordinario de los sistemas especificados al apartado 1 del presente documento, apoyando experto y garantizando el mantenimiento de aplicaciones del tipo correctivo, preventivo, adaptativo y evolutivo
- Aumentar el nivel de calidad y de madurez del servicio de apoyo intermediando
  - Centralización y profesionalización de los servicios orientados a apoyar al usuario
  - Incorporación de nuevas actividades en el seguimiento de las mismas
  - Automatización de los procedimientos operativos
- Aumentar el nivel de calidad y de madurez del servicio de mantenimiento y evolución de las aplicaciones intermediando:
  - Medida de la calidad a través de Acuerdos de Nivel de Servicio específicos por el servicio.
  - Disponer de un servicio global de mantenimiento y evolución de las aplicaciones que permita llevar a cabo las necesidades detectadas por la gestión de la demanda.
  - Incremento de la interlocución directa entre los jefes de evolutivo del adjudicatario y los usuarios clave en los desarrollos de evolutivos.
  - Un compromiso de mejora continua en los resultados, los procesos y los procedimientos.
  - Apoyo técnico experto por los diferentes sistemas.
- Mejorar la eficiencia del servicio. Disminución del mantenimiento correctivo (incluyendo la depuración de datos) e incremento de las tareas de valor añadido.
- Mejorar la documentación funcional y técnica para poder mejorar el porcentaje de resolución de peticiones desde el 1.º nivel de atención al usuario para un potencial uso eficaz de soluciones basadas en IA.
- Servicio flexible, alineado con la evolución tecnológica y organizativa de la UOC.
- Garantizar la estabilidad y permanencia del conocimiento.

- Garantía en la transición y devolución del servicio.

### **Objetivos de Continuidad del Servicio**

- Disponibilidad: Garantizar que las aplicaciones estén disponibles y operativas 24/7, con un tiempo de inactividad mínimo.
- Recuperación de Desastres: Implementar planes de recuperación de desastres para asegurar la continuidad del servicio en caso de fallos graves.
- Monitorització: Utilizar sistemas de monitorització para detectar y resolver problemas de manera proactiva antes de que afecten los usuarios.

### **Objetivos de Calidad**

- Cumplimiento de requisitos funcionales y no funcionales
  - Que el software haga lo que tiene que hacer (funcionalidad) y lo haga bueno (rendimiento, seguridad, etc.)
- Fiabilidad
  - Que funcione de manera estable y sin errores, incluso en condiciones adversas.
- Eficiencia
  - Que utilice los recursos (memoria, CPU, red) de manera óptima.
- Usabilidad
  - Que sea fácil de utilizar, intuitivo y agradable para el usuario final.
- Mantenibilitat
  - Que se pueda modificar, corregir o ampliar con facilidad.
  - Portabilidad
  - Que pueda funcionar en diferentes entornos o plataformas con mínimos ajustes.
- Seguridad
  - Implementar medidas de seguridad para proteger los datos de los usuarios y evitar vulnerabilidades.
- Escalabilidad
  - Que pueda crecer en funcionalidades o volumen de usuarios sin perder calidad.
- Cumplimiento normativo
  - Que siga los estándares y regulaciones aplicables en la UOC (ISO, GDPR, etc.)
- Satisfacció del Usuario:
  - Recoger feedback de los usuarios para mejorar continuamente las aplicaciones y asegurar un alto nivel de satisfacció.

### **Objetivos de Eficiencia**

- Optimización de Recursos: Utilizar recursos de manera eficiente para reducir costes operativos y mejorar el rendimiento de las aplicaciones.

- Automatización: Implementar procesos automatizados para reducir la carga de trabajo manual y aumentar la eficiencia.
- Escalabilidad: Asegurar que las aplicaciones puedan escalar para gestionar un aumento en el número de usuarios o en la carga de trabajo sin comprometer el rendimiento.

### Objetivos de Gestión

- Planificación Estratégica: Desarrollar planes estratégicos para la mejora continua de las aplicaciones, alineados con los objetivos de la universidad.
- Gestión de Proyectos: Utilizar metodologías de gestión de proyectos para asegurar que las tareas se cumplan dentro de los plazos y presupuestos establecidos.
- Transparencia: Mantener una comunicación clara y transparente con todas las partes interesadas sobre el estado de los servicios y las mejoras implementadas.

### Objetivos de Formación y Apoyo

- Formación Continua: Proporcionar formación continua a los usuarios y al personal técnico sobre las nuevas funcionalidades y mejoras.
- Apoyo Técnico: Ofrecer un apoyo técnico eficiente y accesible para resolver cualquier incidencia o duda de los usuarios.

### Objetivos de Ciberseguridad

- Confidencialidad:
  - Proteger los datos sensibles (personales, financieras, sanitarias...) contra accesos no autorizados.
  - Implementar mecanismos de cifrado en tráfico y en reposo.
- Integridad:
  - Garantizar que los datos no sean alterados de manera maliciosa o accidental.
  - Utilizar firmas digitales, controles de versiones y validaciones de entrada.
- Disponibilidad:
  - Asegurar que la aplicación esté operativa y accesible cuando se necesita.
  - Prevenir ataques de denegación de servicio (DDoS) y disponer de planes de contingencia.
- Autenticación y autorización:
  - Verificar la identidad de los usuarios y controlar sus permisos.
  - Implementar protocolos seguros como OAuth2, OpenID Connect, MFA (autenticación multifactor).
- Trazabilidad y registro:
  - Registrar las acciones de los usuarios y del sistema para detectar comportamientos anómalos.

- Facilitar la auditoría y la respuesta a incidentes.
- Resiliencia ante vulnerabilidades:
  - Minimizar los riesgos mediante el desarrollo seguro (DevSecOps).
  - Hacer pruebas de penetración, escaneos de vulnerabilidades y actualizaciones regulares.
- Cumplimiento normativo
  - Adaptarse a regulaciones como el RGPD, la LOPDGDD, RÍA o la ISO/IEC 27001.
  - Garantizar que el tratamiento de datos se hace con base legal y transparencia.
- Educación y concienciación del desarrollador
  - Formar los equipos en buenas prácticas de seguridad.
  - Fomentar una cultura de “seguridad desde el diseño”.

### Objetivos de UX (Experiencia de Usuario)

En relación a las aplicaciones y servicios digitales:

- Ofrecer y garantizar que todos los usuarios de la comunidad UOC (estudiantes, futuros estudiantes, docentes, equipo de gestión, alumni, etc), independientemente de sus competencias, contextos y capacidades funcionales, tengan una excelente experiencia de usuario en su relación e interacción con las aplicaciones y SD que ofrece la UOC.
- Conocer las necesidades, objetivos, modelo mental y contextos de los usuarios para ofrecer aplicaciones y SD pertinentes, usables, con sentido y significativos a través del uso de las metodologías adecuadas.
- Iterar las soluciones con los usuarios finales
- Cumplir con los estándares y buenas prácticas en materia de UX.
- Basarse en datos cualitativos y cuantitativos para tomar decisiones que afecten a la UX.
- Facilidad de uso: Que el usuario pueda entender y utilizar las aplicaciones y SD sin confusión.
- Eficiencia: Que el usuario pueda completar tareas con el mínimo esfuerzo y tiempo.
- Eficacia: ofrecer soluciones que demanden poca carga cognitiva al usuario
- Satisfacción: Que la experiencia sea agradable y gratificante.
- Consistencia: Que la interfaz mantenga patrones visuales y funcionales coherentes.
- Identidad de marca: Que las interfaces siguen los estilos gráficos de la Marca UOC y que el usuario sepa en todo momento que se encuentra en espacios digitales UOC.
- Retroalimentación clara: Que el sistema informe al usuario sobre lo que está pasando (errores, confirmaciones, etc.).
- Adaptabilidad: Que el software se adapte a diferentes dispositivos, contextos, navegadores, idiomas y necesidades.

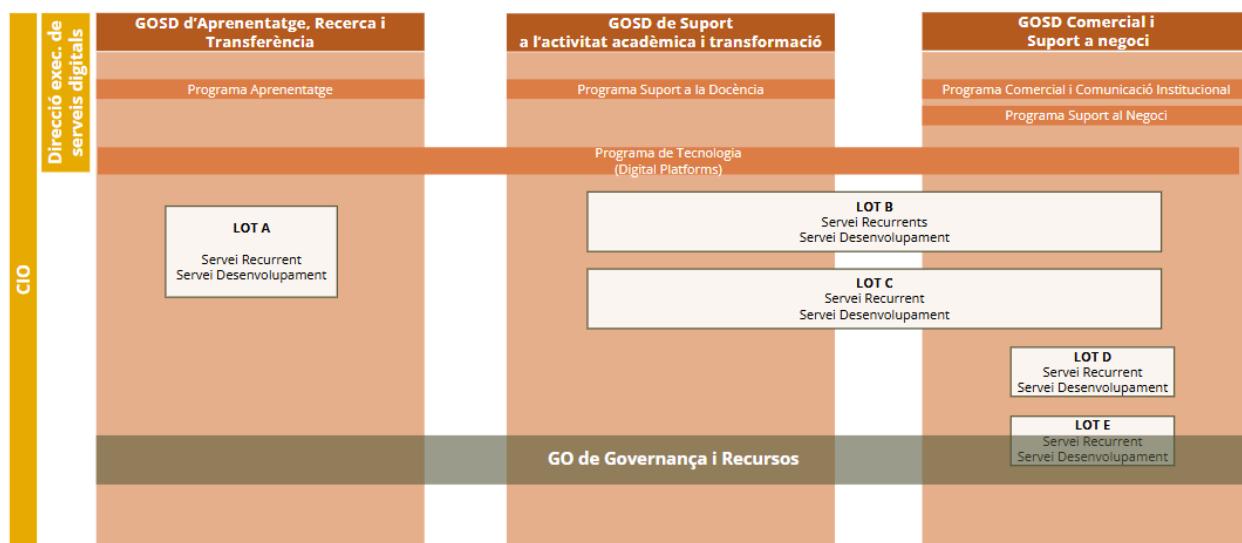
### Objetivos de Accesibilidad

- Cumplimiento de la normativa vigente y de como mínimo la doble A de las WCAG.
- Ofrecer y garantizar que todos los elementos de interacción y relación (aplicaciones, web, contenidos, pdf, etc) son accesibles y también usables para todos los usuarios de la comunidad UOC.
- Conocer las necesidades, objetivos, modelo mental y contextos de los usuarios con diversidad funcional para ofrecer aplicaciones y SD pertinentes, usables, con sentido y significativos a través del uso de las metodologías adecuadas.
- Inclusión universal: Que nadie quede excluido por limitaciones físicas, sensoriales o cognitivas.
- Compatibilidad con tecnologías asistivas: Como lectores de pantalla, teclados alternativos, etc.
- Ofrecer y garantizar posibilidades de interacción y relación no dependientes de ratón, visión, colores y sonidos.

## 3. Descripción de los servicios

### 3.1. Visión global del servicio

En primer lugar, y a modo ilustrativo, se presenta una visión global de los servicios relacionados con los servicios objeto del presente pliego y a modo de relación entre con los Grupos Operativos del Área de Tecnología:



Los Responsables de los Servicios Digitales de la UOC son los responsables de la definición, la gestión, la evolución y la sostenibilidad, los resultados y la calidad en la prestación de los servicios. Conecta todos los elementos que componen los servicios y los vincula con las actividades y procesos de la organización con las partes interesadas.

Los servicios a prestar se aplicarán sobre los sistemas de cada lote y se ejecutarán siguiendo las condiciones de ejecución descritas en el apartado [4. Condiciones de ejecución del servicio](#), y son los siguientes:

#### 3.1.1 Servicios recurrentes

Cada adjudicatario de los servicios de mantenimiento se responsabiliza de la operatividad de las aplicaciones, asegurando y garantizando que la aplicación funciona y presta los servicios por los cuales fueron diseñadas e implementadas.

Actividades asociadas a la corrección o mejora del software de las aplicaciones (correctivo, preventivo, adaptativo), apoyo técnico en el uso eficiente y la correcta configuración de las aplicaciones de apoyo a los procesos de negocio, la mejora continua así como tareas orientadas a la gestión del servicio global

Las incidencias, consultas y peticiones que se tratan en los servicios de operación llegan centralizadas por la ventanilla única (SAU), que actúa como primer nivel de atención. No obstante, los servicios también atenderán peticiones directas de miembros del Área de tecnología así como peticiones de otros servicios de atención.

El servicio SAU cubre el proceso completo de atención sobre los servicios y aplicaciones que presta el Área de Tecnología de la UOC al usuario trabajador de la Universidad, hasta la resolución de su petición de apoyo, incluyendo todas las etapas comprendidas en su gestión: registro, valoración, seguimiento (con respuestas parciales al usuario que entra la petición siempre que sea posible), escalado, coordinación, resolución y cierre. Quedan incluidos en su alcance las peticiones de apoyo que puedan hacer otros servicios de atención como el dirigido al estudiante. Quedan fuera de su alcance cambios de funcionalidad, proceso, necesidades de proyecto y la gestión de la demanda en general.

- Nivel 1: Ventana única por el servicio de apoyo del Área de Tecnología de la UOC. Responde a consultas, incidencias, problemas y peticiones sobre los procesos y aplicaciones de tecnología. Es el que ahora se conoce como SAU.
- Nivel 2: Responde a tickets que requieren un conocimiento o acceso a herramientas especializadas, estos llegan y vuelven a través del nivel 1. El servicio lo prestan varios AM y servicios especializados en ámbitos y aplicaciones como por ejemplo el Apoyo en su punto de Trabajo.
- Nivel 3: Casos no procedimentados que no puede resolver el nivel 2. Hace cosas como la gestión del cambio y resuelve problemas, entendidos como causa raíz de incidencias que se repiten. El servicio lo prestan varios AM y servicios. Los aspectos más complejos técnicamente, se escalan a proveedores propietarios de la solución, como fabricantes, Google, Instructure, etc

Cada adjudicatario será responsable de llevar a cabo las siguientes actividades:

### **Mantenimiento de aplicaciones: Correctivos, adaptativos y preventivos**

Actividades asociadas a la corrección y mejora del software de las aplicaciones, segundos tres tipos:

- Correctivo

El objetivo principal es localizar y eliminar los posibles defectos del software. Se entiende por defecto cualquier característica del sistema que tiene la potencialidad de producir una quiebra, es decir, un comportamiento del sistema diferente al que está establecido en las especificaciones.

Esta actividad se inicia con la recepción de las incidencias de los usuarios finales o con la detección de defectos por el propio equipo de trabajo de mantenimiento.

- Adaptativo

El objetivo principal es la modificación del software debido a cambios en el entorno en el que se ejecuta, tanto si estos cambios son debidos al hardware, software o requerimientos legales.

Esta actividad se inicia con la petición de integraciones de desarrollos o soluciones de terceros, o con las propuestas realizadas por el propio equipo de trabajo de mantenimiento y acordadas con la UOC.

- Preventivo

El objetivo principal es la mejora en el diseño de los sistemas sin variar significativamente sus funcionalidades, incluyendo las tareas proactivas de mejora técnica y optimización del software existente para minimizar los riesgos de quiebra del software y los errores de usuario en los sistemas.

Hay que contemplar también en este apartado las pruebas necesarias para garantizar el correcto funcionamiento de los sistemas en caso de implantación o actualización de Service Packs y parches del software.

### **Apoyo técnico y operativo sobre las aplicaciones**

Tareas orientadas al apoyo técnico a los usuarios en el uso eficiente y la correcta configuración de las aplicaciones de apoyo a los procesos de negocio.

- Consultas

Actividad que incluye la atención y respuesta a dudas, consultas o sugerencias sobre las aplicaciones. Se trata, por lo tanto, de una actividad de carácter meramente informativo sobre las funcionalidades y características técnicas de los sistemas.

- Apoyo a la Configuración

Actividad de apoyo en la configuración de las herramientas y la resolución de incidencias causadas por la incorrecta definición de datos por parte del usuario, o por la incorrecta configuración de una herramienta.

- Análisis de impacto y elaboración de ofertas de los servicios bajo demanda

Actividad que incluye el análisis y la aproximación en la solución del desarrollo necesario para dar cobertura a una determinada necesidad planteada.

- Explotación de Datos

Tareas de extracción y presentación de información almacenada en bases de datos o ficheros, mediante las herramientas oportunas o la creación de scripts y procedimientos ad hoc.

- Creación o modificación de informes

Tareas de creación de nuevos informes o modificación de los existentes para cubrir nuevas necesidades funcionales.

- Parametrización

Tareas de modificación de la parametrización de aplicaciones para atender nuevas necesidades funcionales.

- Corrección de datos erróneos

Tareas de resolución de incidencias causadas por incoherencia o inconsistencia de datos, ya sean como consecuencia de la entrada de información por parte de los usuarios finales, o como resultado de un proceso automático. La resolución pasa siempre por la adecuación de los datos mediante la cirugía oportuna en la base de datos.

### **Guardias y actuaciones fuera de horario laboral / Apoyo funcional extendido / Atención especial en actividades clave**

La guardia consiste en el apoyo para cubrir la resolución de incidencias de aplicaciones que apoyan a procesos críticos de la UOC, fuera del horario laboral. Este apoyo puede ser requerido de forma presencial a las oficinas de la UOC, aunque habitualmente no es necesario.

Se incluye también el apoyo en la instalación de actualizaciones de software, como por ejemplo service packs, pegados de seguridad, etc. que se tengan que realizar fuera del horario laboral.

Se prevén un máximo de 12 peticiones de guardia al año por cada lote, que se pedirán con dos semanas de antelación como mínimo y que tendrán una duración de una semana cada una y con una cobertura de 24x7.

### **Mejora continua**

De manera proactiva, el servicio propondrá las iniciativas que inciden en la mejora de las funcionalidades y/o la eficiencia y la calidad del servicio. Se incluyen pues:

Tareas orientadas a la gestión global del servicio, el aumento de la calidad, la mejora continua, la atención permanente a las necesidades y expectativas del servicio y el cumplimiento de los estándares, procedimientos y normativas establecidas.

Auditorías del servicio y actividades asociadas al análisis y mejora de la eficiencia del servicio prestado.

Actividades asociadas al desarrollo e implantación de acciones por el aumento de la calidad y la eficiencia de los servicios prestados con el objetivo de garantizar su sostenibilidad.

### **Gestión operativa**

- Coordinación y seguimiento de peticiones

Tareas orientadas a la gestión del registro, recepción y formalización de peticiones, así como la coordinación y seguimiento de su resolución.

- Recepción, diagnóstico y formalización de las peticiones

Se incluye la reproducción del caso, la recopilación de todos los datos necesarios para que el gestor pueda resolver la petición, la derivación al técnico especialista responsable de su resolución, ya sea de este mismo servicio u otros servicios tecnológicos.

- Seguimiento de la resolución

Seguimiento de las incidencias importantes o críticas, cambiando prioridades y garantizando su correcta resolución.

- Coordinación de la resolución compartida de peticiones

Coordinación con otros grupos para dar respuesta a una incidencia, cuando su resolución no está en el ámbito del servicio.

- Gestión del conocimiento

Tareas orientadas a incorporar, publicar y hacer difusión de los documentos que la UOC determine así como toda aquella documentación que el adjudicatario considere necesaria por la gestión integral de los servicios del cual es responsable.

Así mismo, se llevarán a cabo tareas orientadas a mantener y evolucionar la base de datos de conocimiento para la resolución de problemas e incidentes.

- Incorporación de nuevos desarrollos y baja de aplicaciones

Actividades orientadas a la incorporación de nuevos desarrollos al servicio referentes a nuevas aplicaciones o modificación de las existentes, ya sea en el marco del propio servicio objeto de este contrato o bien fuera de este servicio y desarrollados por el mismo proveedor u otros, previa validación del desempeño de las normativas de desarrollo de la UOC y siguiendo el proceso de incorporación establecido.

Así mismo tendrá que garantizar la estabilidad y el gobierno del ecosistema tecnológico objeto del contrato en uno en torno a desarrollos evolutivos y proyectos multe proveedor.

También se consideran en este apartado las tareas orientadas a la baja de aplicaciones obsoletas (actuaciones y gestiones sobre el repositorio de código, la CMDB, el catálogo de aplicaciones en mantenimiento, desinstalación del código, documentación de la baja, etc.)

### **Mantenimiento de las Plataformas Funcionales**

Para las tareas de mantenimiento relacionadas con las Plataformas Funcionales como Canvas, DXP, Salesforce y Atlassian, habrá que realizar las siguientes actividades:

- Gestión y monitorización de la plataforma
  - Supervisión del rendimiento y disponibilidad del sistema.
  - Identificación y resolución de posibles errores o caídas del servicio.
  - Optimización de la infraestructura para garantizar una experiencia fluida.
- Actualizaciones y mantenimiento
  - Aplicación de parches de seguridad y actualizaciones de software.
  - Validación de compatibilidad con integraciones existentes después de cada actualización.
  - Implementación de nuevas funcionalidades y configuración de opciones avanzadas.
- Gestión de integraciones
  - Configuración y mantenimiento de conexiones con otros sistemas (ERP, CRM, LMS, etc.).
  - Resolución de problemas de sincronización de datos entre aplicaciones.

- Optimización de flujos de trabajo para mejorar la eficiencia operativa.
- Seguridad y cumplimiento normativo
  - Implementación de protocolos de seguridad para proteger datos sensibles.
  - Gestión de accesos y permisos para garantizar que solo los usuarios autorizados puedan acceder a determinadas funcionalidades.
  - Cumplimiento de normativas como GDPR para proteger la privacidad de los usuarios.
- Apoyo y formación a los usuarios
  - Resolución de dudas e incidencias técnicas.
  - Creación de documentación y guías de uso para facilitar la gestión de la plataforma.
  - Formación a los equipos para maximizar el uso de las herramientas disponibles.
- Evaluación de novedades y evolución funcional:
  - Revisar las actualizaciones periódicas del fabricante o proveedor.
  - Analizar las nuevas funcionalidades, cambios de interfaz o mejoras de rendimiento.
  - Valorar la aplicabilidad de estas novedades sobre el parque de servicios digitales existentes.
  - Identificar oportunidades de integración o sustitución de servicios actuales.
- Servicio de acompañamiento síncrono del entorno de aprendizaje:
  - Atención síncrona de manera virtual para resolver dudas sobre el funcionamiento de la plataforma Canvas.
  - Frecuencia de estas sesiones síncronas adaptable según las actividades críticas de la universidad, disponibilizado más horas síncronas en periodos de alta actividad y menos en periodos valle.
  - Contabilización de las atenciones realizadas, con fines analíticos y de mejora continua.

### 3.1.2 Servicios adicionales bajo demanda

#### Desarrollo bajo demanda: Pequeño evolutivo

Adaptaciones de la aplicación actual a nuevos requerimientos con el objetivo de dar respuesta a las nuevas peticiones de los sistemas existentes y modificaciones y/o mejoras orientadas a actualizar/ampliar la funcionalidad actual de los sistemas de información.

Estas nuevas funcionalidades y mejoras de aplicaciones existentes son identificadas por los usuarios que conjuntamente con la figura del Responsable del Servicio por parte de la UOC (que es quien velatorio para el correcto funcionamiento de los Servicios Digitales con una visión extremo a extremo) trabajan estas necesidades y es el Responsable del Servicio de la UOC quién identifica el pequeño evolutivo a desarrollar. Estos desarrollos tienen un esfuerzo aproximado inferior a 100 horas.

El Servicio de Pequeño Evolutivo incluye las actividades propias del desarrollo de aplicaciones para atender las necesidades identificadas por el Responsable del Servicio de la UOC conjuntamente con los usuarios finales.

El procedimiento utilizado por la gestión de estas necesidades será similar al siguiente:

- El Responsable del Servicio de la UOC registra el pequeño evolutivo a la herramienta corporativa por el registro de peticiones (JIRA) con los datos relevantes:
  - Descripción de la necesidad
  - Vinculación con el adjudicatario responsable del mantenimiento
  - Planificación del evolutivo al sprint deseado
- El adjudicatario hace una valoración a alto nivel de la necesidad.
- La UOC acepta o rechaza la valoración.
- El Responsable del Servicio de la UOC revisa conjuntamente con el adjudicatario la capacidad para incluir el pequeño evolutivo en el sprint deseado.
- Control y monitorización del estado del pequeño evolutivo. La gestión y la ejecución del pequeño evolutivo se realiza a través del plugin Agile del JIRA.

### **Desarrollo bajo demanda: Proyecto Evolutivo**

Los desarrollos con un esfuerzo superior a las 100h, se portarán a término en a modalidad de proyecto evolutivo, llevándose a cabo las siguientes tareas orientadas a su gestión:

- Planificación, Control y Seguimiento del proyecto

Se incluyen las actividades de control y seguimiento centradas en la preparación del proyecto, la monitorización del estado, la gestión de riesgos, la resolución de problemas y la información del estado al responsable del servicio de la uoc y a la comisión de seguimiento y control del proyecto. Estas actividades tienen que permitir corregir desviaciones y detectar problemas incluso antes de que estos se presenten.

- Análisis y Diseño

Se incluyen las actividades de análisis de requerimientos funcionales y técnicos, coordinadamente con el especialista/los especialistas de aplicación uoc, identificando las necesidades a cubrir por las herramientas a desarrollar. Se incluyen también las actividades de diseño técnico y funcional de la solución aportada, en coordinación con los usuarios y con el equipo de arquitectura de la uoc. El diseño de la solución incluirá la documentación necesaria para su implementación según la normativa vigente de la UOC.

- Desarrollo

Se incluyen las actividades de desarrollo sobre la arquitectura correspondiente. Estas actividades de desarrollo incluirán la codificación en el lenguaje de la aplicación, las pruebas unitarias de cada uno de los desarrollos, y la documentación técnica de los mismos. También se incluyen las actividades de integración con otros sistemas existentes, cuando se proceda.

- Implantación

Se incluyen las actividades de adaptación y configuración de las diferentes funcionalidades de la solución adoptada así como la introducción de los datos básicos para la validación funcional de los mismos. Aun así, se incluyen actividades de migración de datos y de migración de aplicaciones y/o funcionalidades cuando corresponda.

- Pruebas

Se incluyen en estas actividades el diseño, desarrollo, apoyo, gestión y seguimiento de las pruebas tanto de integración, unitarias, de volumen y de adaptación de la aplicación o desarrollo a posar en producción. También se incluyen todas las pruebas necesarias en el caso de migraciones a nuevas versiones de cualquier producto base.

Estas pruebas estarán encaminadas a asegurar el correcto funcionamiento de la aplicación en su totalidad, tanto de las diferentes configuraciones parciales como de los desarrollos, integración con otras herramientas y adaptación a los sistemas UOC. Siempre siguiendo el modelo de calidad vigente a la UOC.

- Apoyo Puesto-implantación

Se incluyen las actividades de apoyo puesto-implantación técnico y/o funcional necesarias para asegurar la estabilidad de los desarrollos y/o funcionalidades posadas en producción.

Las actividades de apoyo pueden ser requeridas de manera tanto presencial como remota.

- Gestión del Cambio

Se incluyen las actividades derivadas de la gestión del cambio habitual en cualquier implementación o desarrollo de una herramienta informática o sistema de información.

Se incluyen también dentro de este apartado el diseño, desarrollo, planificación y formación tanto a los usuarios responsables de la aplicación y el establecimiento de medidas y políticas de comunicación que minimicen el impacto de la implementación.

- **Traspaso a mantenimiento**

Se incluyen las actividades destinadas a traspasar el control del desarrollo al equipo estable de mantenimiento de las soluciones uoc. Este traspaso se realiza durante la garantía del desarrollo y finaliza al agotarse el periodo de garantía.

### **Condiciones por la valoración y ejecución de proyectos evolutivos**

- Proyectos con alcance cerrado (Un proyecto de alcance cerrado es aquel en que los objetivos, los requisitos y las limitaciones están claramente definidos desde el principio, sin posibilidad de modificaciones significativas durante el desarrollo. En este tipo de proyectos, el presupuesto, los plazos y los recursos se mantienen fijos, y cualquier modificación requerirá una gestión del cambio).

El adjudicatario:

- Conjuntamente con el Responsable de Servicio de la UOC, hace la definición detallada del alcance (descripción de requisitos/historias de usuario)
- Conjuntamente con el especialista de aplicación y arquitectura UOC determina la solución técnica más apropiada.
- Habrá que determinar la composición del equipo de proyecto, incluyendo la figura del jefe de proyecto.
- Se presentará una estimación en horas por cada perfil incluido en el equipo de trabajo propuesto.
- Se podrá justificar la estimación de esfuerzos presentada con posibles eficiencias obtenidas por el uso de herramientas de apoyo como la IA generativa.
- Presenta la propuesta por la ejecución del proyecto, incluyendo el cumplimiento Normativo de la UOC, en términos de Entregables, DevOps, Calidad, Seguridad, UX y Accesibilidad, así como el detalle de las actividades a realizar y las eficiencias obtenidas.
- La UOC acepta la propuesta
- Realiza el proyecto según las condiciones UOC y la propuesta presentada
- Si la UOC no acepta la propuesta, habrá que rehacerla por parte del adjudicatario.

- Proyectos ágiles de alcance abierto y esfuerzo cerrado (el proyecto se desarrolla de manera iterativa, permitiendo ajustamientos en los requisitos y las características en función de las necesidades emergentes, pero sin superar los límites de tiempos, equipo o presupuesto predefinidos)

El adjudicatario:

- Conjuntamente con el Responsable del Servicio Digital de la UOC, hace la definición detallada de los requisitos/historias de usuario iniciales
  - Conjuntamente con el especialista de aplicación y arquitectura UOC determina la solución técnica más apropiada
  - Determina la composición del equipo de proyecto
  - Acuerda con el Responsable del Servicio Digital de la UOC el esfuerzo mínimo y máximo que tiene que suponer cada sprint dentro del proyecto
  - Realiza el proyecto con tantos sprints como se puedan realizar dentro del esfuerzo previsto por el proyecto
  - Cada sprint finalizado contabilizará en el proyecto con el valor de esfuerzo pactado, independientemente del esfuerzo real que haya supuesto conseguirlo
  - El alcance de cada sprint se acordará entre el responsable del servicio digital de la uoc y el proveedor, siempre dentro de los márgenes de esfuerzo mínimo y máximo, y con tiempo suficiente para acordar el esfuerzo que supone y no iniciar ningún sprint que no tenga un valor acordado.
- 
- Proyectos ágiles de alcance abierto y esfuerzo abierto (tanto los objetivos como los recursos pueden evolucionar durante el desarrollo. En este tipo de proyecto, no hay una limitación estricta de tiempo, equipo o presupuesto, y se permite expandir las capacidades en función de las necesidades emergentes.)

El adjudicatario:

- Conjuntamente con el Responsable del Servicio Digital de la UOC, hace la definición detallada de los requisitos/historias de usuario iniciales
- Conjuntamente con el especialista de aplicación y arquitectura UOC determina la solución técnica más apropiada
- Determina la composición del equipo de proyecto
- Acuerda con el Responsable del Servicio Digital de la UOC el esfuerzo mínimo y máximo que tiene que suponer cada sprint dentro del proyecto

- Realiza el proyecto. Al final de cada sprint, el equipo UOC determinará si hay que continuar con el siguiente sprint. Cuando se decida no continuar, se valorará el proyecto en la suma del valor de los sprints que ha ejecutado el adjudicatario
- Cada sprint finalizado contabilizará en el proyecto con el valor de esfuerzo pactado, independientemente del esfuerzo real que haya supuesto conseguirlo
- El alcance de cada sprint se acordará entre el responsable del servicio digital de la uoc y el proveedor, siempre dentro de los márgenes de esfuerzo mínimo y máximo, y con tiempo suficiente para acordar el esfuerzo que supone y no iniciar ningún sprint que no tenga un valor acordado.

### **3.1.3 Apoyo en herramientas basadas en Inteligencia Artificial (IA)**

Será recomendable usar soluciones basadas en IA para generar eficiencias dentro del servicio y así poder realizar todas las tareas descritas en el mismo.

Una de las ventajas más grandes de incorporar la IA a los servicios de mantenimiento de aplicaciones es la automatización de tareas repetitivas y manuales, como por ejemplo pruebas de software, generación de código básico o incluso la corrección automática de errores. Al automatizar las tareas rutinarias, los equipos pueden enfocarse en aspectos más estratégicos y de más valor, cosa que se traduce en tiempos de entrega más rápidos y una mejor calidad del producto final. La lista siguiente es orientativa y no pretende ser completa ni restrictiva:

- Mantenimiento Predictivo: Utilizar IA para analizar datos históricos y actuales de las aplicaciones para predecir posibles errores antes de que pasen. Esto permite hacer mantenimientos preventivos y evitar tiempos de inactividad no planificados.
- Automatización de Tareas Repetitivas: Implementar bots y algoritmos de IA para automatizar tareas rutinarias como por ejemplo actualizaciones de software, pruebas de regresión y monitorización de rendimiento. Esto libera a los desarrolladores para que se concentren en tareas más complejas y creativas.
- Análisis de código y detección de errores: Utilizar herramientas de IA para revisar el código y detectar errores o vulnerabilidades de seguridad. Estas herramientas pueden proporcionar recomendaciones para mejorar la calidad del código y reducir el tiempo de depuración.
- Personalización de la Experiencia de Usuario: La IA puede analizar el comportamiento de los usuarios para ofrecer experiencias personalizadas. Esto incluye recomendaciones de contenido, ajustes de interfaz y otras personalizaciones que pueden mejorar significativamente la satisfacción del usuario.
- Desarrollo Basado en Datos: La IA puede analizar grandes cantidades de datos para identificar tendencias y patrones que pueden informar el desarrollo de aplicaciones. Esto permite a los

desarrolladores tomar decisiones más informadas sobre las funcionalidades y las mejoras que hay que implementar.

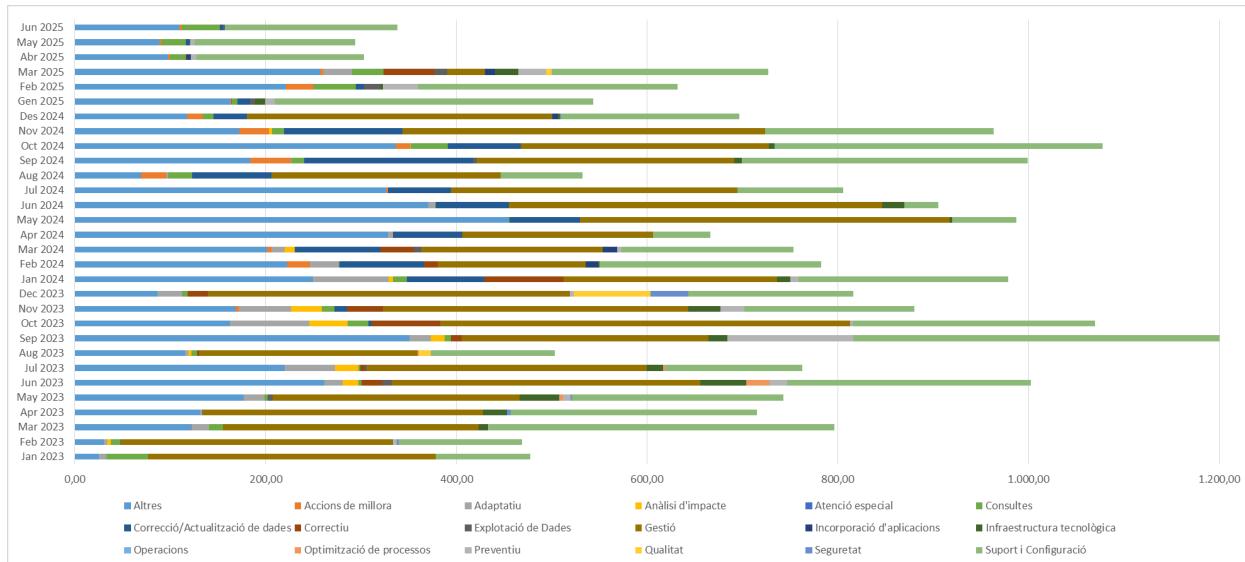
- Asistencia en la Resolución de Problemas: Los asistentes de IA pueden ayudar a los desarrolladores a resolver problemas técnicos proporcionando sugerencias y soluciones basadas en una amplia base de datos de conocimiento. Esto puede reducir el tiempo de resolución de problemas y mejorar la eficiencia general del desarrollo.
- Desarrollo de nuevos Proyectos: Los asistentes de IA pueden ayudar los desarrolladores en las diversas fases de la elaboración de Proyectos:
  - Fase de Análisis de requerimientos y Diseño: mejora por el análisis de requerimientos, por la estimación de esfuerzos y por la predicción de riesgos.
  - Fase de Desarrollo e implementación: mejora por la generación y revisión de código y por la re-factorización de código.
  - Fase de Pruebas: mejora por la generación de casos de prueba, por la automatización de pruebas y por el análisis de resultados de prueba.
  - Pase a Producción: mejora por la automatización del despliegue.
  - Mantenimiento: mejora por la monitorización del rendimiento, por el análisis de logs y por la predicción de incidentes.
  - Gestión del conocimiento: mejora para la generación de documentación y entregables del proyecto

Las empresas licitadoras, tal como se indica al PCP, dentro de la propuesta técnica, tendrán que realizar una propuesta de solución basada con el apoyo de IA para la generación de mejoras y eficiencias, que cumplan con los requerimientos de Protección de Datos y Confidencialidad de Datos así como de aquello establecido en el Reglamento de inteligencia artificial respecto de los modelos y/o sistemas de inteligencia artificial propuestos, de forma que los datos de la UOC no se usen para bases de datos globales, sino que se usen solo para el entorno propio de la UOC. En cualquier caso, las empresas licitadoras tendrán que presentar la solución que quieren implementar con IA generativa, y será la UOC quién tendrá la facultad de aprobarla o rechazarla antes de que se pueda hacer uso.

## **3.2. Volumetría y dimensionado de los Servicios Recurrentes**

Este apartado tiene como finalidad principal ofrecer una descripción detallada de los volúmenes básicos asociados al servicio recurrente que constituye el objeto de la presente licitación. Los datos que se presentan tienen un carácter meramente orientativo y se han elaborado a partir del análisis de la experiencia acumulada durante los últimos ejercicios en la prestación de servicios de naturaleza recurrente. Esta información se fundamenta en la línea base correspondiente a los sistemas incluidos en cada uno de los Lotes, y pretende proporcionar una visión general que permita a los licitadores dimensionar adecuadamente los recursos y medios necesarios para la ejecución del contrato, sin que esto implique ningún compromiso firme en cuanto a los volúmenes futuros:

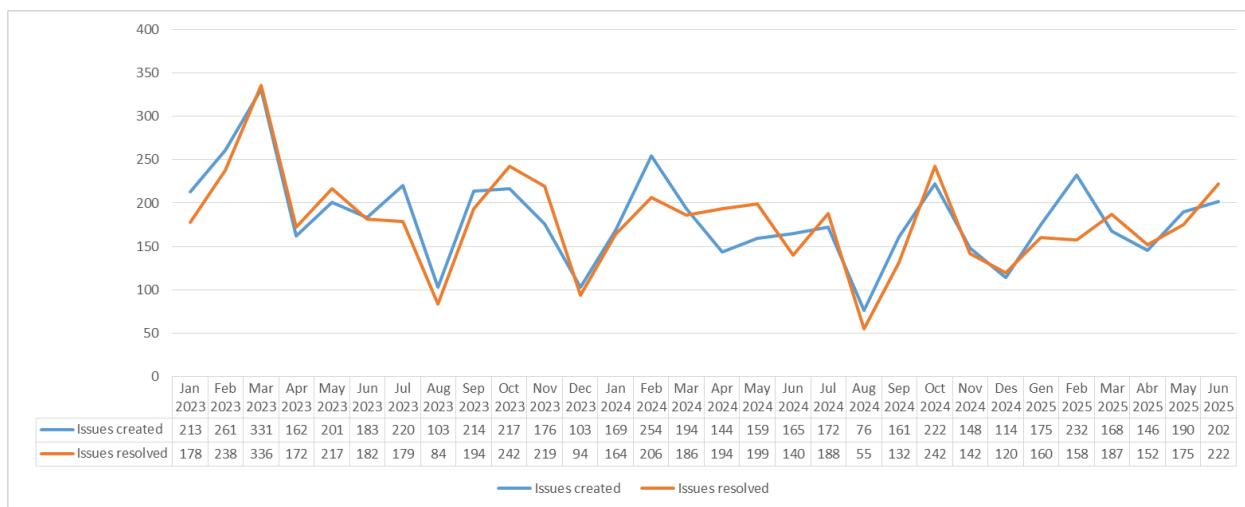
## LOTE A: Mantenimiento y Evolución de Sistemas de Aprendizaje

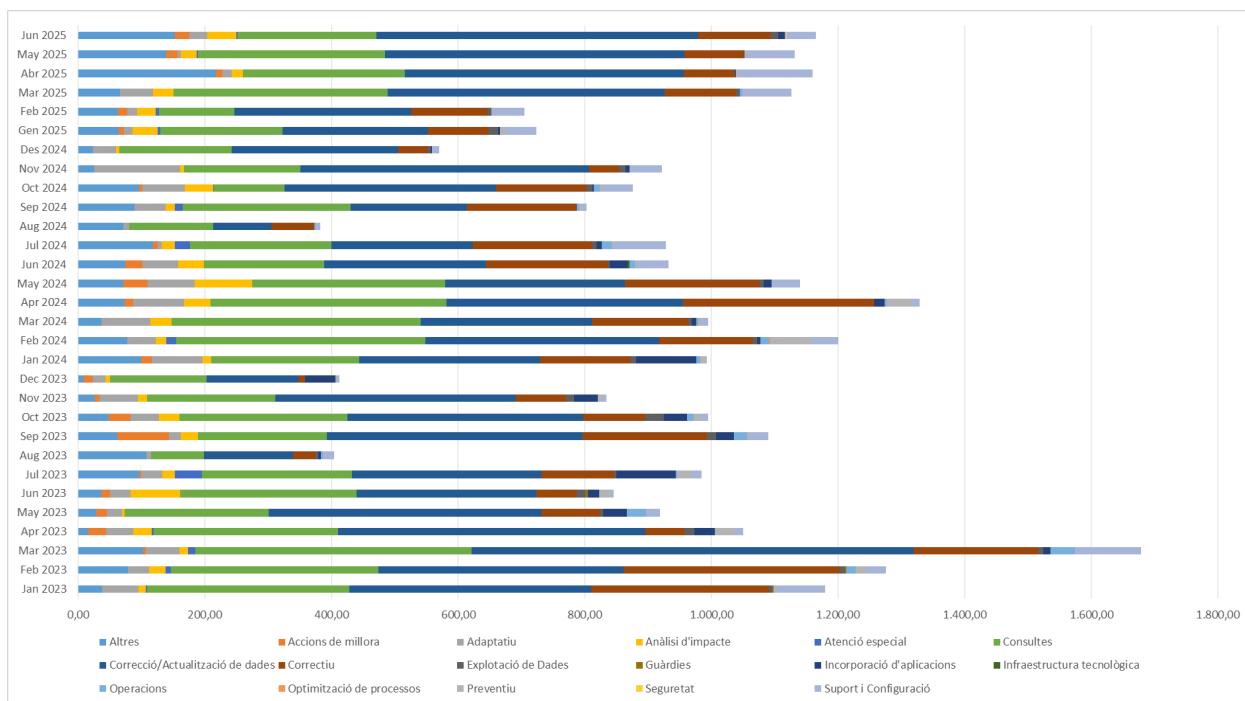


- Se prevé una dedicación anual estimada de 13.380 horas para la realización de las tareas del Servicio Recurrente del LOTE A, distribuidas entre los diferentes perfiles profesionales personas/año.

Servicio Recurrente LOTE A	Horas (h/año)
Responsable del servicio	1.050

Ninguno de proyecto	430
Arquitecto	1.000
Consultor senior / analista senior	4.400
Maquetador	500
Analista-desarrollador	6.000
<b>TOTAL</b>	<b>13.380h</b>

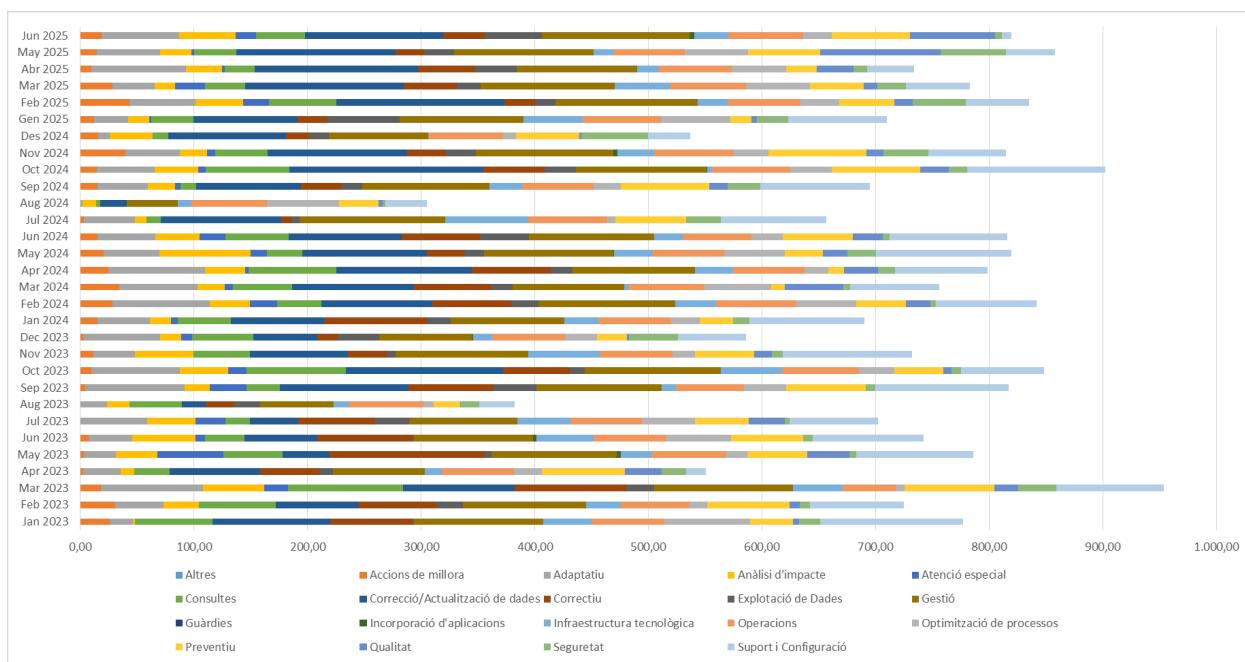
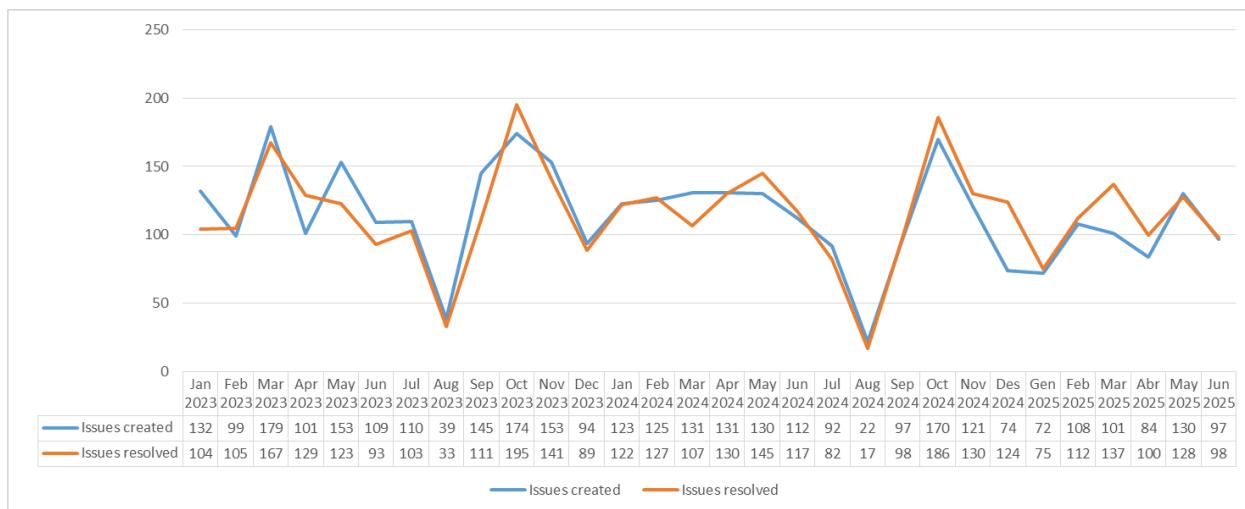
**LOTE B: Mantenimiento y Evolución de sistemas:**



- Se prevé una dedicación anual estimada de 16.371 horas para la realización de las tareas del Servicio Recurrente del LOTE B, distribuidas entre los diferentes perfiles profesionales personas/año.

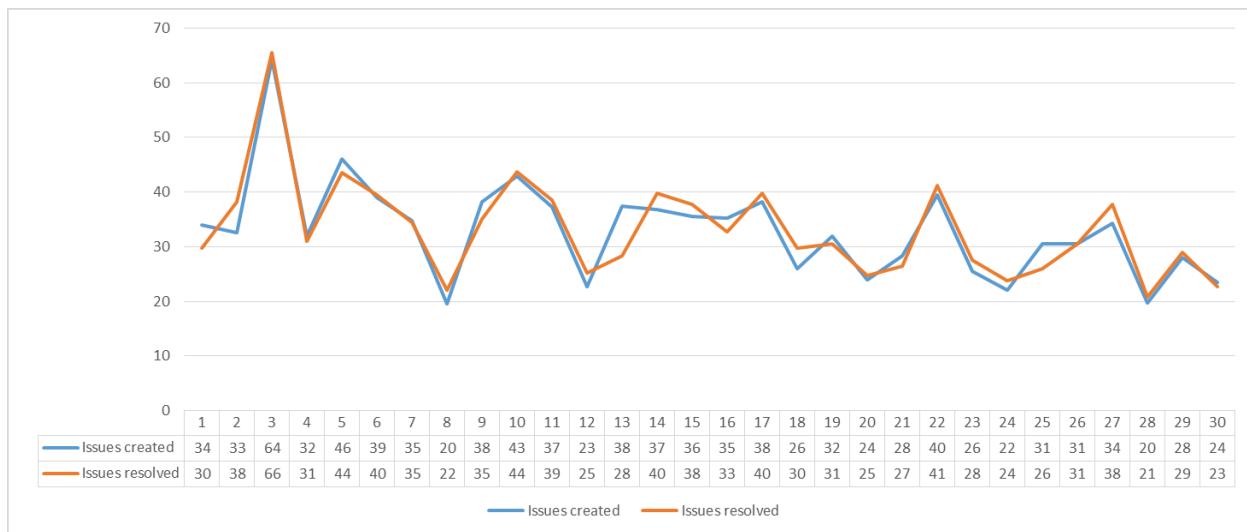
Servicio Recurrente LOTE B	Horas (h/año)
Responsable del servicio	1.200
Ninguno de proyecto	500
Arquitecto	1.200
Consultor senior / analista senior	5.470
Maquetador	1.000
Analista-desarrollador	7.001
<b>TOTAL</b>	<b>16.371h</b>

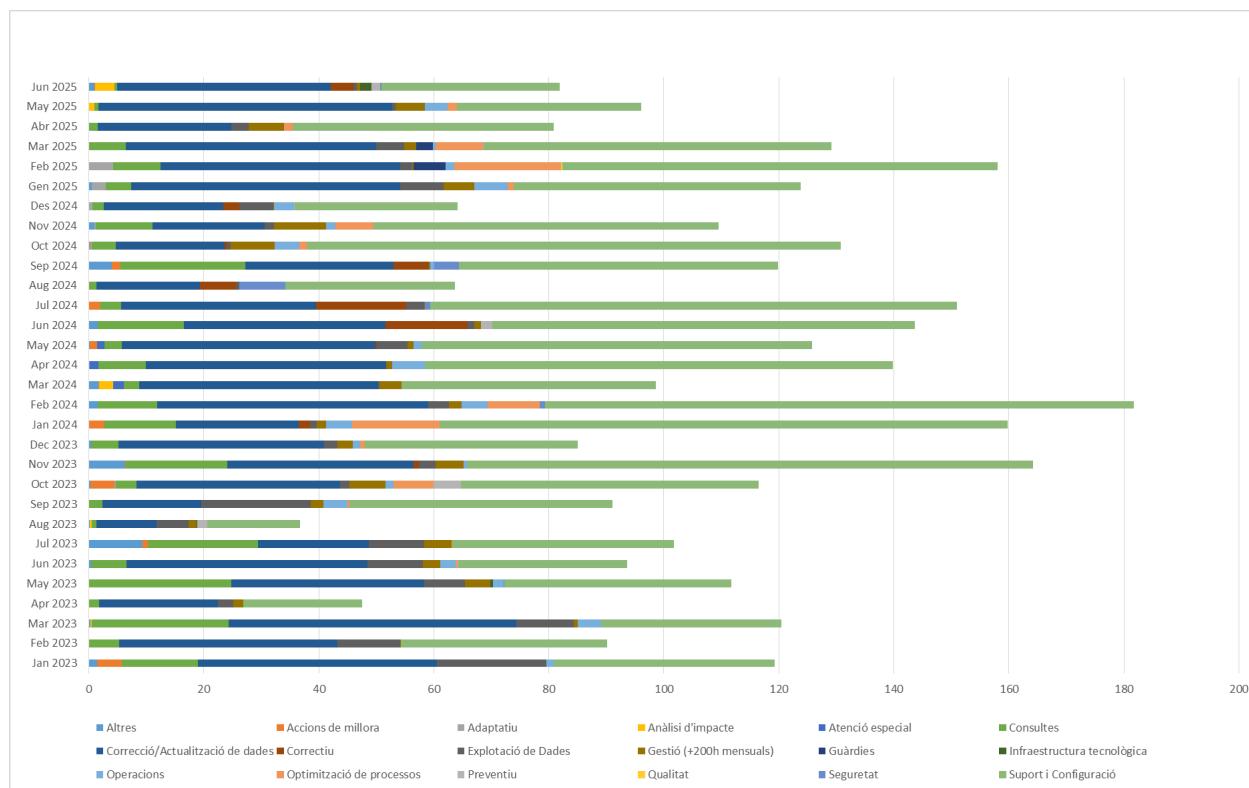
## LOTE C: Mantenimiento y Evolución de sistemas:



- Se prevé una dedicación anual estimada de 10.516 horas para la realización de las tareas del Servicio Recurrente del LOTE C, distribuidas entre los diferentes perfiles profesionales personas/año.

Servicio Recurrente LOTE C	Horas (h/año)
Responsable del servicio	800
Ninguno de proyecto	200
Arquitecto	1.216
Consultor senior / analista senior	2.900
Maquetador	400
Analista-desarrollador	5.000
<b>TOTAL</b>	<b>10.516h</b>

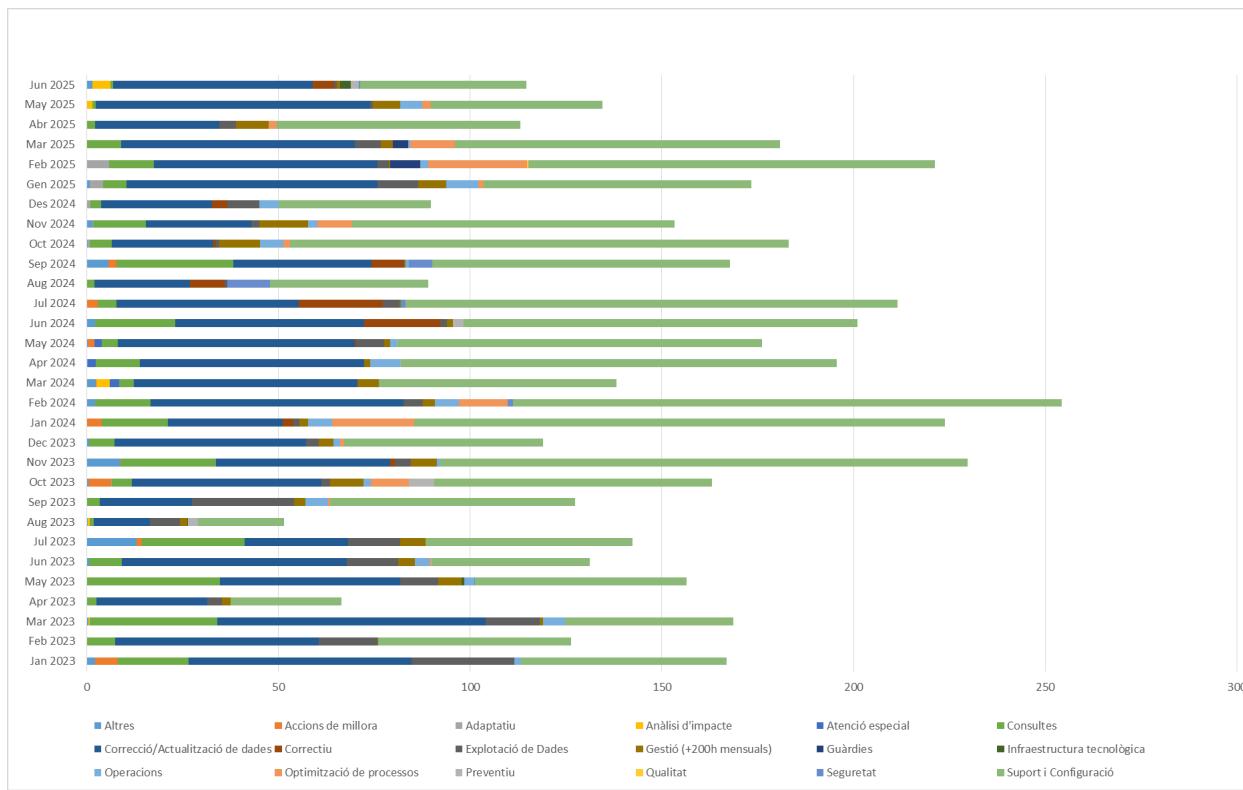
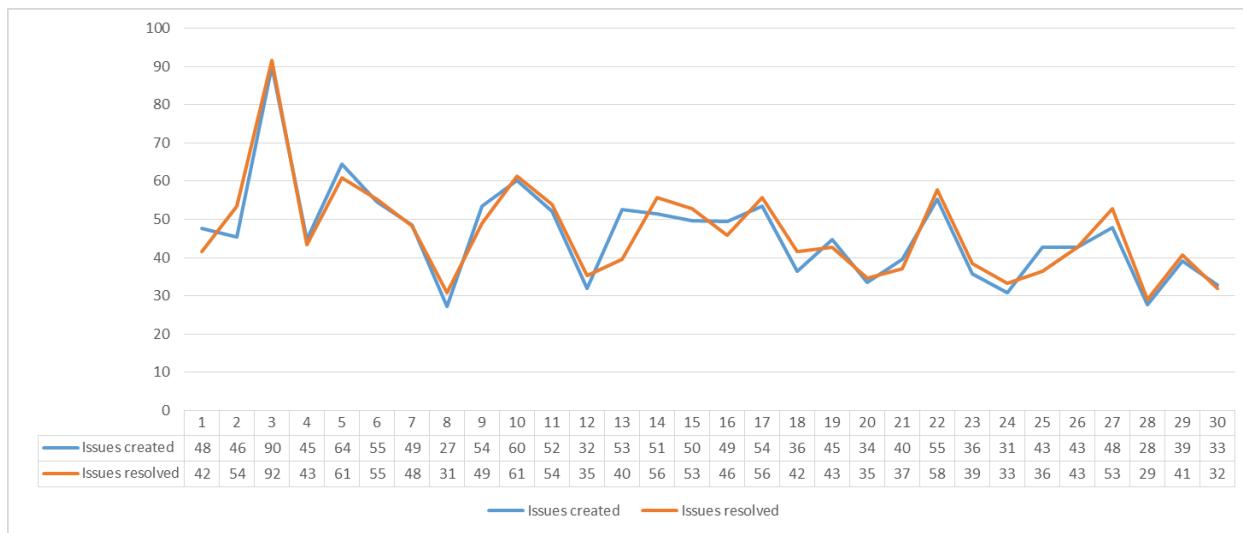
**LOTE D: Mantenimiento y Evolución de sistemas:**



- Se prevé una dedicación anual estimada de 2.780 horas para la realización de las tareas del Servicio Recurrente del LOTE D, distribuidas entre los diferentes perfiles profesionales personas/año.

Servicio Recurrente LOTE D	Horas (h/año)
Responsable del servicio	100
Ninguno de proyecto	50
Arquitecto	200
Consultor senior / analista senior	730
Maquetador	100
Analista-desarrollador	1.600
<b>TOTAL</b>	<b>2.780h</b>

## LOTE E: Mantenimiento y Evolución de sistemas:



- Se prevé una dedicación anual estimada de 3.837 horas para la realización de las tareas del Servicio Recurrente del LOTE E, distribuidas entre los diferentes perfiles profesionales personas/año.

Servicio Recurrente LOTE E	Horas (h/año)
Responsable del servicio	200
Ninguno de proyecto	80
Arquitecto	300
Consultor senior / analista senior	1.007
Maquetador	150
Analista-desarrollador	2.100
<b>TOTAL</b>	<b>3.837h</b>

Esta información se tiene que considerar como una estimación orientativa que permita valorar inicialmente la dotación de equipo necesario para llevar a cabo las actividades asociadas a los servicios recurrentes de cada lote, teniendo en cuenta las fluctuaciones de intensidad, tanto los picos como los valles, de la actividad prevista para cada servicio.

El servicio se valorará en función del grado de cumplimiento de los Acuerdo de Nivel de Servicio (MÁS BIEN).

### **3.3. Volumetría y dimensionado de los Servicios Bajo Demanda**

La UOC se encuentra inmersa en un proceso de transformación tecnológica continuo, que implica la necesidad de llevar a cabo proyectos estratégicos de gran envergadura con una continuidad en el tiempo. Esta necesidad se ha visto incrementada exponencialmente año tras año, fruto de la evolución del Plan de Proyectos Anual.

Por este motivo, se ha elaborado la siguiente estimación de dedicación, teniendo en cuenta la evolución de los últimos años en la inversión tecnológica y el volumen de proyectos. Este análisis histórico nos permite proyectar una tendencia ascendente en las necesidades de mantenimiento y evolución de aplicaciones.

Por otro lado, aparte de las necesidades actuales ya identificadas y el crecimiento orgánico previsto, se ha realizado una estimación al alza previendo posibles incrementos presupuestarios futuros. Esta

previsión se basa en la experiencia de los últimos años, donde se han producido incrementos significativos no siempre anticipados, y en la voluntad de la UOC de mantenerse en vanguardia tecnológica y adaptarse rápidamente a las nuevas demandas educativas y de gestión.

Por eso, hemos elaborado la siguiente previsión de dedicación para los LOTES A, B y C, que incorpora un margen para proyectos no previstos y nuevas tecnologías que puedan surgir:

Servicio Bajo Demanda	Horas (h/año)
Responsable del servicio	1.200
Ninguno de proyecto	9.000
Arquitecto	6.010
Consultor senior / analista senior	18.100
Maquetador	3.500
Analista-desarrollador	33.430
<b>TOTAL</b>	<b>71.240</b>

y la siguiente estimación de dedicación para los LOTES D y E:

Servicio Bajo Demanda	Horas (h/año)
Responsable del servicio	890
Ninguno de proyecto	6.720
Arquitecto	5.040
Consultor senior / analista senior	13.150
Maquetador	2.610
Analista-desarrollador	25.000
<b>TOTAL</b>	<b>53.410</b>

## 4. Condiciones de ejecución del servicio

Las condiciones de ejecución del servicio aplican a los cuatro lotes de forma individual a cada uno de ellos.

### 4.1 Modelo de relación

El modelo de relación define las funciones y responsabilidades de la empresa adjudicataria y la UOC en un marco de actuación común, para asegurar el cumplimiento de las obligaciones de cada una de las partes. Es un marco de relación que permite acordar el contenido y nivel de la prestación de los servicios, así como el seguimiento de la prestación real en los aspectos contractuales, estratégicos, tácticos y operativos.

#### 4.1.1 Estructura de responsabilidades

La estructura mínima de responsables dentro del contrato que tendrá que proveer el adjudicatario será la siguiente:

- **Responsable de Cuenta**

El adjudicatario designará un único responsable de cuenta que será el referente por todos los contratos que tenga con la uoc, será el último responsable de los de la prestación de los servicios. Esta figura se mantendrá durante toda la vida del contrato en la gestión comercial, durante la ejecución del servicio y hasta la devolución de este, y será el interlocutor y responsable en caso de que se produzcan cambios en el alcance o coste de los servicios que impliquen una modificación contractual.

- **Responsable de Servicio**

El adjudicatario designará un único responsable de servicios por el contrato que asumirá las siguientes funciones:

- Garantizar la atención a los responsables del Servicio de la UOC en todos los ámbitos del servicio, atendiendo a las necesidades de negocio de la UOC
- Mantener la visibilidad global de la prestación del servicio, en todas sus dimensiones (servicio recurrente, servicios adicionales y Servicios de desarrollo bajo demanda).
- Garantizar la calidad del servicio en todos sus ámbitos
- Garantizar el desempeño de los Niveles de Servicio
- Garantizar la correcta organización del equipo de trabajo por el desempeño con el servicio acordado

- Garantizar la capacidad operativa, el dimensionado, del equipo en el decurso del servicio
- Seguimiento y control de los recursos asignados a los servicios
- Control de costes, estimación de esfuerzos y su seguimiento
- Análisis de las desviaciones del servicio (en alcance y coste)
- Elaboración de los informes de servicio y justificación del cumplimiento de los ANTES
- Asegurará la buena colaboración entre los diferentes adjudicatarios con quienes se tiene que relacionado para mejorar el servicio de negocio final
- Proponer e incorporar, si son aceptadas, mejoras en la gestión global del servicio
- Formar parte del comité de dirección.

- **Responsable de Proyectos:** Es el responsable de asegurar la visión global del seguimiento de los proyectos evolutivos que se ejecuten en el marco de lote, esto como de transmitir y coordinar la aplicación de la metodología de gestión de proyectos establecida por la UOC. Las principales funciones respecto a los proyectos son las siguientes:

- Velar por el logro de los hitos.
- Velar por el logro de los objetivos.
- Supervisión de los riesgos críticos asegurando que se están aplicando las medidas de mitigación adecuadas y que se están tratando y reportando adecuadamente.
- Coordinar las relaciones entre proyectos y actuaciones (sean proyectos o actuaciones propias o realizadas por terceros).
- Mantener una planificación global.
- Velar por la aplicación de los requisitos metodológicos, de calidad, de seguridad o de UX en coordinación con el resto de responsables.
- Proponer e incorporar, si son aceptadas, mejoras en la gestión global de los proyectos
- Formar parte del comité de dirección.

- **Responsable de Calidad:**

El adjudicatario asignará un **responsable de calidad del servicio**: es el responsable de coordinar y asegurar el cumplimiento de la normativa de desarrollo del área de tecnología de la uoc vigente en cada momento. Formará parte del comité de calidad. Las principales funciones son:

- Asegura una estrategia de calidad por los proyectos, servicios y aplicaciones
- Aseguramiento de la calidad en los desarrollos
- La verificación de la ejecución del control de la calidad
- Identificar y proponer actuaciones de mejora en el ámbito de la calidad y la accesibilidad, e implementarlas si éstas son aceptadas.

- **Responsable de Arquitectura:** es el responsable de coordinar, armonizar e implementar los principios de tecnología de la uoc, la hoja de ruta de software y el catálogo de servicios TI y al diseño de

la arquitectura de los sistemas de información y servicios a construir o mantener por el adjudicatario. Sus principales responsabilidades son velar y establecer los mecanismos de control por el cumplimiento de los principios y estándares de arquitectura de la UOC así como diseñar conjuntamente con el equipo de arquitectura UOC las soluciones. Formará parte del comité de arquitectura.

El adjudicatario tendrá que asignar un referente de arquitectura del lote (o los que considere) que será la persona o personas a quién:

- se enviará información relacionada con novedades
- participará en la gestión de incidentes o problemas (y que derivará al arquitecto del proyecto relacionado, si lo considera)
- Se convocará a las reuniones periódicas donde se reunirán los diferentes arquitectos de los Lotes para compartir novedades y propuestas.

● **Responsable de Seguridad:** es el responsable de garantizar liderar e impulsar el cumplimiento del marco normativo de seguridad de la UOC. Actuará como enlace entre el adjudicatario y los diferentes agentes implicados cuando se traten temas de seguridad. Formará parte del comité de seguridad.

● **Responsable de UX y (User eXperience) y de Accesibilidad:** es el responsable de garantizar, liderar e impulsar el cumplimiento de las directrices de la uoc en relación a ux y la accesibilidad. A continuación se detallan sus responsabilidades:

- Responsable transversal en el ámbito UX y de accesibilidad en las iniciativas y mantenimientos ejecutados por el partner
- Velar por la calidad y los resultados de las actividades y entregables UX y de accesibilidad como responsable último del servicio siguiendo la Normativa sobre UX y accesibilidad (Anexo) y de cumplir con los criterios de aceptación de cada entrega.
- Conocer la normativa y procedimientos UX y de accesibilidad, difundirla entre los roles del servicio /partner participantes en las iniciativas y mantenimientos y asegurar su cumplimiento
- Asegurar la disponibilidad y calidad de los perfiles necesarios para la ejecución de las actividades UX y de accesibilidad.
- Planificar las actividades UX y de accesibilidad de manera coordinada e integrada con el conjunto de las actividades del Lote en general
- Actuará como enlace entre el adjudicatario y los diferentes agentes implicados cuando se traten temas de UX.
- Conocer los Servicios Digitales impactados por las iniciativas ejecutadas por el Servicio/Partner
- Conocer las plataformas y/o aplicaciones tecnológicas que sustentan los Servicios Digitales.
- Identificar y proponer actividades, metodologías, entregables y actuaciones de mejora en materia de UX y accesibilidad por el ámbito y Servicio Digitales que le correspondan.

- Formar parte del comité de UX. Participará en estos comités para realizar el seguimiento, evaluación y report de la actividad UX y de accesibilidad en las iniciativas
- **Responsable de Innovación:** es el responsable de la mejora continua y la eficiencia del servicio:
  - Optimización de procesos: Identificar oportunidades para automatizar tareas repetitivas y mejorar la gestión de los errores e incidencias.
  - Investigación de nuevas tecnologías: Estar atento a las últimas tendencias en desarrollo y mantenimiento de software para integrar mejoras.
  - Gestión proactiva del rendimiento: Implementar herramientas de análisis para anticipar problemas y optimizar los recursos.
  - Colaboración con otros departamentos: Trabajar con equipos de desarrollo y operaciones para aplicar las mejores prácticas en la gestión del software.
  - Fomento de la cultura de innovación: Inspirar al equipo para pensar en nuevas formas de mejorar el servicio, promoviendo uno en torno a creatividad.

Por su parte, la UOC designará un responsable del contrato que tendrá las funciones derivadas de la dirección, comprobación, información y vigilancia de la correcta realización de los trabajos y otros que le sean asignadas. Aun así, la UOC se reserva el derecho a realizar todos aquellos controles e inspecciones que crea oportunos con el fin de garantizar el cumplimiento del contrato.

El responsable del contrato de la UOC estará en estrecha comunicación con el Responsable de Cuenta designado por el adjudicatario que, como interlocutor único, proporcionará toda la información necesaria para poder conocer con el detalle suficiente la evolución del contrato.

#### **4.1.2 Órganos de Gobierno**

Por el control y seguimiento de la ejecución del contrato se crearán los órganos de gobierno siguientes:

- **Comité de Dirección**

Asumirá las funciones de supervisión de la ejecución del contrato así como la toma de decisiones que afecten al objetivo y alcance del contrato.

Las principales funciones del Comité de Dirección son:

- Asegurar la adecuación del equipo de acuerdo con las necesidades de cada fase del servicio
- Seguimiento ejecutivo y de los riesgos de los proyectos. Cambios de planificación

- Gestión de conflictos, validaciones de las gestiones de cambio
- Seguimiento contractual del servicio (inicio, aprobación de trabajos, finalización, cambios de alcance)

El Comité de Dirección estará formado como mínimo por el Responsable de Cuenta y el Responsable de Servicios y los representantes que la UOC determine, y se reunirá con una periodicidad trimestral aunque se podrá convocar con carácter extraordinario siempre que se considere necesario.

- **Comité de Seguimiento Operativo**

Se encargará del control de la ejecución del servicio con el nivel de detalle justo y necesario para resolver las incidencias y conflictos que aparezcan en el decurso del contrato.

Las principales funciones del Comité de Seguimiento son:

- Seguimiento de la evolución del servicio
- Aprobación de re-planificaciones y cambios
- Proponer al Comité de Dirección los cambios estratégicos, económicos, de alcance y de gestión que considere oportunos por la buena marcha del servicio y si se tercia, las modificaciones contractuales que correspondan
- Gestión de riesgos
- Resolución de conflictos
- Seguimiento de la operación diaria del servicio, y verificación de la correcta gestión de peticiones y cambios
- Seguimiento y control de los evolutivos
- Desarrollo y mantenimiento de los procedimientos operativos necesarios para el correcto funcionamiento del servicio
- Análisis de peticiones y situaciones de cambio en los servicios
- Escalado de posibles mejoras detectadas en el servicio

El Comité de Seguimiento estará formado como mínimo por el Responsable de Servicios y los representantes que la UOC determine. En función de los evolutivos que estén en ejecución o planificados, se requerirá la asistencia de los Responsables de Evolutivo o Jefes de Proyecto. Este Comité se reunirá con una periodicidad mensual aunque se podrá convocar con carácter extraordinario siempre que se considere necesario.

- **Comités con los responsables de cada lote.**

Tienen la función de dar consistencia de servicio único desde el punto de vista UOC a los diferentes lotes en materias específicas, coordinar actuaciones inter-lote, hacer propuestas de valor, difundir principios,

metodologías, buenas prácticas, etc. así como garantizar la gestión del cambio de los respectivos servicios en cada una de las materias.

Estos comités se reunirán de forma ordinaria bimensualmente, pudiéndose convocar de forma extraordinaria a instancia de la UOC o de cualquiera de sus miembros. También pueden existir convocatorias específicas vinculadas en el calendario de actuación de la UOC (inicio de semestre, inicio de campaña, etc.).

- **Comité de ejecución:**

- Estará formado por las personas Responsables de servicios y de proyectos de los diferentes lotes y por las personas que a tal efecto designe la UOC.
- El objetivo del comité de ejecución es hacer seguimiento con los responsables de los servicios de mantenimiento del at para una mejor coordinación:
  - Revisión conjunta de backlog: Encontrar sinergias o dependencias entre tickets de AMs diferentes.
  - Revisión cambios normativos: Puesta en común de los últimos cambios normativos para estar enterados de las novedades.
  - KPI comunes: Tiempo de resolución, número de incidencias reabiertas, satisfacción de usuario...

- **Comité de calidad:**

- Estará formado por las personas responsables de calidad de los diferentes lotes, la persona responsable de calidad de la UOC y por otras personas que a tal efecto designe la UOC.
- El objetivo del comité de calidad es hacer seguimiento con los responsables de calidad de los servicios de mantenimiento del at para una mejor coordinación:
  - Asegurar la eficacia y eficiencia del mantenimiento (correctivo, evolutivo y adaptativo)
  - Monitorizar la calidad del servicio ofrecido por las diferentes áreas (Revisión Quality Gates)
  - Fomentar la prevención de incidencias y no solo su resolución
  - Promover la mejora continua y el alineamiento con los objetivos de negocio

- **Comité de arquitectura:**

- Estará formado por las personas responsables de arquitectura de los diferentes lotes, por la persona responsable de arquitectura de la UOC y por otras personas que a tal efecto designe la UOC.
- El objetivo del comité de arquitectura es hacer seguimiento con los responsables de arquitectura de los servicios de mantenimiento del at para una mejor coordinación:
  - Gobernanza: Definir y velar por una arquitectura técnica común y alineada entre servicios.

- Integración e interoperabilidad: APIs, sistemas interconexados, protocolos compartidos
- Evolución tecnológica: roadmap de actualización de versiones, obsolescencia, tech radar
- Revisar los impactos arquitectónicos de las propuestas de mejora o evolución funcional
- Fomentar el uso de estándares, buenas prácticas y componentes reutilizables.
- **Comité de seguridad:**
  - Estará formado por las personas responsables de seguridad de los diferentes lotes, la persona responsable de seguridad de la UOC y por otras personas que a tal efecto designe la UOC.
  - El objetivo del comité de seguridad es hacer seguimiento con los responsables de seguridad de los servicios de mantenimiento del AT para una mejor coordinación:
    - Análisis de vulnerabilidades: resultados de escaneos (SAST, DAST), dependencias inseguras, etc.
    - Incidentes de seguridad: gestión, respuestas, lessons learned
    - Políticas de acceso y permisos: revisión de privilegios, gestión de usuarios
    - Seguridad en el código fuente: revisión de prácticas, uso de secretos, gestión de credenciales
    - Xifratge y comunicaciones seguras: protocolos, certificados, HTTPS, TLS...
    - Planes de contingencia y recuperación: simulacros, documentación, roles activados
- **Comité de UX y Accesibilidad:**
  - Estará formado por las personas responsables de UX de los diferentes lotes, la persona responsable de UX de la UOC y por otras personas que a tal efecto designe la UOC.
  - El objetivo del comité UX es hacer seguimiento con los responsables UX de los servicios de mantenimiento del AT para una mejor coordinación:
    - Control y evaluación del estado de la aplicación de las directrices y normativa UX y de accesibilidad establecidas por la AT en las iniciativas y mantenimientos ejecutadas por estos a través de las herramientas de reporting definidas por la UOC.
    - Seguimiento del conjunto de actividades en materia de UX realizada por los Partners
    - Identificar y compartir dificultades, mejoras y casos de éxito en la aplicación de las directrices UX en las iniciativas
    - Identificar y compartir novedades en las directrices y normativa UX
- **Comité de Innovación:**

- Estará formado por las personas responsables de Innovación de los diferentes lotes, la persona responsable de Innovación de la UOC y por otras personas que a tal efecto designe la UOC.
- El objetivo del comité de innovación es hacer seguimiento con los responsables de innovación de los servicios de mantenimiento del at para una mejor coordinación:
  - Automatización de tareas repetitivas: RPA (Robotic Process Automation), scripts inteligentes, bots de apoyo.
  - Aplicación de IA o ML: predicción de incidencias, optimización de recursos, análisis de logs.
  - Nuevas metodologías de mantenimiento: DevOps, Site Reliability Engineering (SRE), prácticas Lean.
  - Tecnologías emergentes: Serverless, contenedores, edge computing...
  - Mejoras en UX de los equipos técnicos: interfaces, entornos de pruebas, documentación más visual.
  - Reutilización de componentes: desarrollo con arquitectura basada en componentes compatibles.

## 4.2 Mecanismos de control y reporting

Los mecanismos principales de control y reporting de los servicios prestados serán los informes y las memorias, que serán elaborados por el adjudicatario y presentados en los órganos de gobierno correspondientes con la periodicidad establecida. Los diferentes tipos de informes y memorias, con sus particularidades, serán como mínimo los siguientes:

- **Seguimiento Operativo**

El Responsable de Servicios de la empresa adjudicataria elaborará un documento y lo presentará al Comité de Seguimiento con una periodicidad mensual. Este documento tendrá que ser librado con una antelación mínima de 2 días laborables antes de la fecha convocada por el comité. Su contenido tendrá que incluir, como mínimo, las siguientes secciones:

- Análisis de indicadores de actividad
- Análisis de indicadores de compromiso (MÁS BIEN) del servicio recurrente
- Planificación y seguimiento de las acciones de mejora continua del servicio
- Planificación y seguimiento de los pequeños evolutivos
- Planificación y seguimiento de los desarrollos bajo demanda (proyectos evolutivos)

- Análisis de los Indicadores de Gestión de Proyectos
- Incidencias y actividades más relevantes
- Gestión de riesgos identificados
- Gestión de cambios

- **Seguimiento Dirección**

El Responsable de Servicios de la empresa adjudicataria elaborará un documento y lo presentará al Comité de Dirección con una periodicidad trimestral. Su contenido será un resumen con la visión general del servicio prestado y tendrá que incluir, como mínimo, las siguientes secciones:

- Aspectos relevantes del periodo actual y del siguiente
- Evolución de indicadores de actividad, análisis cuantitativo y cualitativo
- Evolución de indicadores de compromiso (ANS) análisis cuantitativo y cualitativo
- Evolución de indicadores de proyectos, análisis cuantitativo y cualitativo
- Análisis de situación del servicio
- Pla de acciones correctoras y de mejora
- Propuestas de cambios estratégicos, económicos, de alcance o de gestión

- **Memoria Anual**

El Responsable de Servicios de la empresa adjudicataria elaborará el documento memoria del servicio y lo presentará al Comité Ejecutivo con una periodicidad anual.

- Evolución de indicadores de actividad, análisis cuantitativo y cualitativo
- Evolución de indicadores de compromiso (ANS), análisis cuantitativo y cualitativo
- Evolución de indicadores de proyectos, análisis cuantitativo y cualitativo
- Análisis de situación del servicio

Adicionalmente, el adjudicatario elaborará la documentación e informes específicos que la UOC determine que son necesarios por el seguimiento y evaluación por cada uno de los servicios prestados, como por ejemplo los relativos a la [Normativa de entregables](#) en los desarrollos bajo demanda o informes adicionales sobre los servicios recurrentes o servicios extremo-extremo.

Los mecanismos de control y reporting se adaptarán a las necesidades de cada momento, y cumplirán con los siguientes principios:

- La UOC realizará las enmiendas que considere adecuadas hasta disponer de los mecanismos de control consensuados por todas las partes.
- El formato y contenido de los documentos tendrán que ser validados por la UOC.
- Los documentos permitirán la navegación por los mismos y facilitarán la comparación y cruce entre los diferentes elementos de servicio mesurados.
- Los indicadores utilizados, su método de cálculo y el formato de presentación serán los definidos al Plan de Indicadores descrito en este documento, a pesar de que el licitador podrá complementarlos con otros adicionales que considere adecuados.

El licitador concretará en detalle a su propuesta técnica, según el PCP, los mecanismos de control y reporting que usará, cumpliendo con los mínimos anteriormente definidos, así como los recursos que destinará para realizar el seguimiento de la calidad de los servicios.

#### **4.2.1 Acuerdos de Nivel de Servicio (ANS) del servicio recurrente**

Los MÁS BIEN servirán para definir el compromiso de servicio acordado entre la UOC y el adjudicatario del contrato y se tendrán que aplicar los mecanismos de gestión necesarios para controlar su grado de cumplimiento.

Cada MÁS BIEN está determinado por los siguientes parámetros:

- Indicadores de medida
- Valores aceptables por los indicadores
- Objetivo de cumplimiento del ANS

- Frecuencia de las medidas

### **Acuerdos de Nivel de Servicio mínimos**

Los indicadores son los siguientes:

- Indicadores de actividad

Mesuran volúmenes y dedicación y son indicativos, sin estar sujetos a ningún ANS.

- Indicadores de Compromís

Miden el grado de cumplimiento del servicio en términos de eficiencia y efectividad del servicio y están sujetos a acuerdos de nivel de servicio:

- Tiempo de evaluación y compromiso de peticiones
  - Intervalo comprendido entre la apertura de un ticket y el momento en que es atendido y tratado. Para su registro correcto, hace falta transicionar el estado del ticket hacia uno de los estados válidos indicados en la tabla del modelo de cálculo de los MÁS BIEN, e indicar, si procede, una previsión sobre su resolución.
- Tiempo de resolución de peticiones
  - Tiempo transcurrido entre la apertura de un ticket y su cierre. Para considerarlo resuelto, hay que situar el ticket en los estados Closed o Resolved, de acuerdo con la tabla del modelo de cálculo de los MÁS BIEN, y describir las acciones realizadas para solucionarlo.
- Tiempo máximo en circuito de las peticiones
  - Refleja el tiempo límite que puede permanecer abierto un ticket en caso de incumplimiento del ANS de resolución. Se calcula igual que el indicador de tiempo de resolución, es decir, desde el momento de la apertura hasta el cierre, y requiere que el ticket se ubique en los estados Closed o Resolved, documentando las acciones ejecutadas para resolverlo.

Los tickets con ANS son los siguientes:

- Incidencias e incidencias de Seguridad
- Peticiones
- Consultas

Los valores y objetivos de cumplimiento mínimos de los indicadores de compromiso por las acciones de operación son:

### Tiempo de evaluación y compromiso

el ANS de tiempo de evaluación y compromiso es el tiempo máximo en el cual los tickets se tienen que atender dentro de unos umbrales establecidos, en función de su prioridad.

Prioridad	Objetivo ANS	Objetivo Servicio
Blocker, Critical	3h	95%
Alta	6h	90%
Minor, Trivial	25h	85%
Normal	16h	88%

### Tiempo de resolución

Prioridad	Objetivo ANS	Objetivo Servicio
Blocker , Critical	6h	95%
Alta	15h	90%
Normal	36h	88%

Minor, Trivial	60h	85%
----------------	-----	-----

El ANS de tiempo de resolución indica el tiempo máximo en el cual los tickets se tienen que resolver dentro de unos umbrales establecidos, en función de su prioridad.

#### **Tiempo máximo en circuito**

Prioridad	Objetivo ANS	Objetivo Servicio
Blocker , Critical	12h	98%
Alta	90h	98%
Normal	90h	95%
Minor, Trivial	90h	95%

El ANS Tiempo máximo en circuito, indica el tiempo máximo que puede estar activo un ticket en caso de incumplimiento del ANS Tiempo de resolución, para que se resuelva al menos dentro de unos nuevos umbrales según la prioridad.

Las prioridades se asignan a partir de la evaluación de la afectación de la incidencia y el comportamiento de la aplicación, y puede ser modificada a posteriori por el gestor de la petición o por algún responsable UOC, si se considera oportuno.

Se medirán los tiempos a partir del calendario laboral del servicio, su horario y en visión servicio (descontando los tiempos en los cuales la petición está pendiente de información, o en espera de actuación de un tercero).

#### Modelo del cálculo de los ANS según estados de Jira

ANTES	Start Status	End Status	Pause Status
Tiempo de evaluación y compromiso	[Open]	[Closed] [En Valoración] [In progress] [Pendiente Info] [Pendiente OT] [Pendiente PRE] [Pendiente PRO] [Resolved]	[Donde Hold] [Pendiente Info]
Tiempo de resolución	[In progress] [Open] [Pendiente PRE] [Pendiente PRO] [Reopened]	[Closed] [Resolved]	[Donde Hold] [Pendiente Info] [Pendiente OT] [Pendiente PRE] [Pendiente PRO]

Tiempo máximo en circuito	[In progress]  [Open]  [Pendiente PRE]  [Pendiente PRO]  [Reopened]	[Closed]  [Resolved]	[Donde Hold]  [Pendiente Info]  [Pendiente OT]  [Pendiente PRE]  [Pendiente PRO]
---------------------------	---	----------------------------	--

#### **ANS especial para la eficiencia en el mantenimiento recurrente anual**

Dentro de la descripción del Acuerdo de Nivel de Servicio (ANS), se incluirá un ANS especial y complementario enfocado a la generación de eficiencia en la dedicación de horas de mantenimiento recurrente anual. La eficiencia se calculará según la siguiente fórmula:

- Eficiencia = (Horas ahorradas año / Horas iniciales año) ×100.

Cada inicio de año de contrato se establecerá un marco y universo de referencia, que servirá para la comparación anual de la dedicación requerida.

Dónde horas ahorradas será el total de horas de dedicación iniciales menos las horas de dedicación actuales.

#### **Propuesta de Acuerdos de Nivel de Servicio**

El licitador a su propuesta técnica, tal como se indica al Pliego de Cláusulas Particulares, tendrá que proponer los acuerdos de niveles de servicio que pueda ofrecer. Como mínimo tendrán que constar los indicadores de compromiso con los niveles de servicio indicados en el apartado anterior.

Durante la fase de servicio regular se podrán revisar y corregir los valores de los ANS, siempre y cuando la UOC y el proveedor lo consideren necesario y haya consenso entre ambas partes.

#### **4.2.2 Sistema de bonificaciones del Acuerdos de Nivel de Servicio**

Los Acuerdos de Nivel de Servicio definidos para cada servicio serán de obligado cumplimiento a lo largo del contrato, exceptuando la fase de transición del servicio.

Para cada servicio, el adjudicatario tiene que cumplir plenamente los Acuerdos de Nivel de Servicio definidos una vez finalizada la fase de prestación en transición.

Si al finalizar el año de contrato, se cumplen todos los ANS, se podrá acceder al siguiente sistema de bonificaciones de ANS anual:

ANS	Tiempo de evaluación y compromiso	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Blocker/Critical	96%	0,1%
	97%	0,2%
	98%	0,3%
	99%	0,4%
	100%	0,5%

ANS	Tiempo de evaluación y compromiso	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Alta	92%	0,1%
	94%	0,2%
	96%	0,3%
	98%	0,4%
	100%	0,5%

ANS	Tiempo de evaluación y compromiso	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Normal	90%	0,1%
	92%	0,2%

	94%	0,3%
	97%	0,4%
	100%	0,5%

ANS	Tiempo de evaluación y compromiso	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Minor/trivial	88%	0,1%
	91%	0,2%
	94%	0,3%
	97%	0,4%
	100%	0,5%

ANS	Tiempo de resolución	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Blocker/Critical	96%	0,1%
	97%	0,2%
	98%	0,3%
	99%	0,4%
	100%	0,5%

ANS	Tiempo de resolución	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Alta	92%	0,1%
	94%	0,2%

	96%	0,3%
	98%	0,4%
	100%	0,5%

ANS	Tiempo de resolución	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Normal	90%	0,1%
	92%	0,2%
	94%	0,3%
	97%	0,4%
	100%	0,5%

ANS	Tiempo de resolución	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Minor/trivial	88%	0,1%
	91%	0,2%
	94%	0,3%
	97%	0,4%
	100%	0,5%

ANS	Tiempo máximo en circuito	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Blocker/Critical	99%	0,2%
	100%	0,5%

ANS	Tiempo máximo en circuito	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Alta	99%	0,2%
	100%	0,5%

ANS	Tiempo máximo en circuito	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Normal	96%	0,1%
	97%	0,2%
	98%	0,3%
	99%	0,4%
	100%	0,5%

ANS	Tiempo máximo en circuito	
Prioridad	Valor obtenido	Bonificación Servicio Recurrente Anual
Minor/trivial	96%	0,1%
	97%	0,2%
	98%	0,3%
	99%	0,4%
	100%	0,5%

Para poder optar a las bonificaciones hay que cumplir los siguientes requerimientos:

- No incumplir ningún ANS en su valor objetivo

- Que el Indicador de tickets reabiertos sea inferior al 5 %

Se establece un periodo de alegaciones y revisiones de más bien, de un mes, por lo tanto, durante el mes de abril, por ejemplo, se podrán realizar peticiones de revisión de tickets por parte de la uoc que no hayan cumplido la más bien y que sean imputables a una causa ajena al alcance del servicio de mantenimiento prestado por parte del licitador.

### **Bonificación por eficiencia**

Este ANS reconocerá el esfuerzo de optimización mediante una bonificación progresiva no acumulativa basada en el porcentaje de eficiencia conseguido en las horas destinadas a operaciones de mantenimiento:

- Entre un 5 % y un 10 % de eficiencia: bonificación del 1 % sobre el coste del servicio recurrente anual.
- Entre un 10 % y un 15 % de eficiencia: bonificación del 2 % sobre el coste del servicio recurrente anual.
- Entre un 15 % y un 20 % de eficiencia: bonificación del 3 % sobre el coste del servicio recurrente anual.
- Más de un 20 % de eficiencia: bonificación del 4 % sobre el coste del servicio recurrente anual.

La UOC aprovechará la eficiencia obtenida mediante las acciones siguientes:

- Reducción de servicio: El 50 % de las horas ahorradas se destinará a posibles ampliaciones o a la prestación de servicios que requieran presupuesto adicional (como el Pequeño Evolutivo, guardias, etc.).
- Inversión en mejora del servicio: El 50 % restante se dedicará a tareas que contribuyan a la mejora del servicio. El licitador podrá utilizar estos recursos para generar nuevas eficiencias si se tercia.

Cada inicio de año de contrato se establecerá un marco y universo de referencia, que servirá para la comparación anual de la dedicación requerida.

La fórmula del cálculo de eficiencia será:

- Eficiencia = (Horas ahorradas año / Horas iniciales año) ×100.

Dónde horas ahorradas será el total de horas de dedicación iniciales menos las horas de dedicación actuales.

Para poder optar a las bonificaciones hay que cumplir los siguientes requerimientos:

- Que el Indicador de tickets reabiertos sea inferior al 5 %
- Que la encuesta de satisfacción del servicio que se realiza anualmente sea favorable.

Los automatismos y procesos desarrollados por el licitador para la generación de eficiencias serán propiedad de la UOC y tendrán que ser librados debidamente documentados.

#### **4.2.3 Plan de indicadores**

La UOC dispone de un Plan de indicadores para medir la actividad de los Servicios Continuados y los Proyectos de Desarrollo en el ámbito de gestión de Proyectos y Aplicaciones. Este plan es de aplicación para los servicios objeto de la presente contratación y, por lo tanto, servirá de base para definir el conjunto de indicadores que se utilizarán y determinar su cálculo y formato de presentación.

El Plan de Indicadores está en continua evolución y, por lo tanto, la UOC podrá añadir indicadores nuevos o modificar los existentes con el objetivo de mejorar el gobierno de los servicios y proyectos.

Adicionalmente, el adjudicatario podrá complementar este Plan incorporando otros indicadores que sean de interés.

El Plan de Indicadores dispone de una herramienta para el cálculo y la generación automática de los gráficos y mesas, en los formatos normalizados establecidos, partiendo de los datos que se pueden extraer de JIRA y otras fuentes de información complementarias.

##### **4.2.3.1 Indicadores de Gestión de Servicios Recurrentes**

Se incluyen en este bloque los indicadores de actividad e indicadores de compromiso de las actividades asociadas a los servicios recurrentes:

- Indicadores de actividad: mesuran volúmenes y son indicativos, sin estar sujetos a ningún ANS.
- Indicadores de compromiso: medirán el grado de cumplimiento de los servicios en términos de eficiencia y efectividad y están sujetos a los ANS.
- Propuesta de XLAs (Experience Level Agreements):
  - Los licitadores podrán proponer indicadores para medir la experiencia del usuario (evaluando su satisfacción, la percepción y el impacto del servicio al usuario final).

CÓDIGO	NOMBRE INDICADOR

<b>ISV01</b>	<b>Operaciones – Evolución de Tickets</b>
ISV01-1	Tickets Entrados vs Resueltos
ISV01-2	Tickets Resueltos por Subtipos Operación
ISV01-3	Tickets Resueltos por Aplicación
ISV01-4	Tickets Resueltos por Prioridad
ISV01-5	Tickets con Prioridad <=Normal vs Prioridad <=Alta
<b>ISV02</b>	<b>Operaciones – Evolución de la Dedicación</b>
ISV02-1	Dedicación en Tipo de Presupuesto Operación por Subtipo Operación
ISV02-2	Dedicación en Tipo de Presupuesto Operación por Aplicación
ISV02-3	Dedicación en Tipo de Presupuesto Evolutivos por Ámbito
<b>ISV03</b>	<b>Operaciones – Evolución del Rati de Tickets Reabiertos</b>
ISV03-1	Tickets reabiertos vs tickets entrados

<b>ISV04</b>	<b>Operaciones No Planificadas – Cumplimiento ANTES</b>
ISV04-1	Cumplimiento MÁS BIEN Tiempo de Evaluación y Compromís
ISV04-2	Cumplimiento MÁS BIEN Tiempo de Resolución
ISV04-3	Cumplimiento MÁS BIEN Tiempo Máximo en Circuito
<b>ISV05</b>	<b>SAU y Satisfacción de usuario</b>
ISV05-1	Grado de satisfacción del usuario en la resolución de los tickets
ISV05-2	Porcentaje de resolución de tickets resueltos desde el 1.º nivel del SAU

#### **4.2.3.2 Indicadores de Gestión de Proyectos**

Se incluyen en este bloque los indicadores de las actividades asociadas a los servicios de desarrollo

<b>CÓDIGO</b>	<b>NOMBRE INDICADOR</b>
IPR01	Proyectos - Proyectos por Estado

IPR02	Proyectos Finalizados en Fecha
IPR03	Desviación de las Planificaciones
IPR04	Proyectos Finalizados en Coste
IPR05	Desviación de los Costes
IPR06	Factores de Desviación de Planificaciones y Costes
IPR07	<b>Proyectos Cancelados</b>
IPR08	Proyectos en Cola
IPR09	Esfuerzo de Resolución de Incidencias de Proyectos en Garantía
IPR10	Gestión de Riesgos
IPR11	Despliegues fallados vs Despliegues realizados correctamente
IPR12	# Correctivos por Proyecto desplegado a Producción en el último trimestre

IPR13	<p>Desviación en las previsiones de entrega de valoraciones de Proyectos en función de la fecha pactada. La fecha pactada irá en relación con el grado de complejidad de la solución:</p> <ul style="list-style-type: none"><li>• Solución básica (No requiere de Diseño de Arquitectura)</li><li>• Solución avanzada (Incluye un Diseño de Arquitectura)</li><li>• Solución estratégica (Incluye un Diseño de Arquitectura complejo con elementos de integración con otros sistemas)</li></ul>
IPR14	<p>Cumplimiento Normativo:</p> <ul style="list-style-type: none"><li>• Revisión de Calidad, Seguridad, Arquitectura, Operaciones, Entregables, Gobierno y Accesibilidad/UX de los Proyectos librados</li></ul>
IPR15	# Proyectos traspasados correctamente a Mantenimiento

## 4.3 Metodología, estándares y entregables

La organización del trabajo y ejecución del servicio se tendrá que adecuar a las metodologías, estándares y entregables establecidos por la uoc, vigentes en el momento de la ejecución del servicio.

El licitador podrá proponer cambios en la metodología, estándares y entregables con el objetivo de hacer mejoras en el servicio y que permita garantizar su efectiva implantación y mejora continua durante la vigencia del contrato. La metodología propuesta tendrá que permitir hacer un control basado en indicadores objetivamente medibles.

Se puede consultar información de referencia en los correspondientes anejas:

### Anejo - Normativa de entregables

El objetivo de la normativa de entregables es dar a los diferentes participantes involucrados en el desarrollo, integración, evolución o mantenimiento de software de la uoc, la normativa sobre los entregables mínimos que hay que entregar en un proyecto.

Se describen también aspectos generales normativos, plazos, nomenclatura y qué son los repositorios donde guardar cada tipo de entregable.

Para cada uno de los entregables mínimos normativos, se describen las fases y los hitos asociados.

### Anejo - Normativa de Calidad

Su objetivo es dar a conocer la normativa que aplica a los proyectos en términos de calidad. Esta normativa se basa en la (ISO/IEC 25010) System and software quality modelos, que describe el modelo de calidad para el producto software y para la calidad en el uso. En concreto, nos centramos en la calidad para el producto software y sus características y subcaracterísticas de calidad sobre las cuales evaluar el software.

Su misión es asegurar que, en términos generales, los proyectos siempre mejoran la calidad de los sistemas de información de la UOC.

En aquellos desarrollos que se requiera, se incluirán requerimientos adicionales por la aplicación del modelo de calidad de producto.

### Anejo - Normativa sobre UX y Accesibilidad

El propósito de la **Normativa UX** es asegurar en lo posible que los usuarios de los productos y servicios digitales de la UOC tengan una buena experiencia de usuario, es decir, que encuentren los productos y servicios de la UOC fáciles de utilizar, útiles, agradables en su uso y con sentido.

Para llegar este punto, esta normativa establece un marco de trabajo común entre la UOC y el proveedor para qué este conozca:

- Los principios y criterios UX que los productos digitales tienen que tener y que la UOC considera aceptables
- Las dinámicas de trabajo definidas por la UOC en la construcción y aceptación de los entregables UX

En la medida que esta normativa se cumple, se asegura el control de los riesgos que pueden provocar desviaciones en el objetivo de ofrecer una buena experiencia de usuario a los usuarios de la comunidad UOC.

El objetivo de la **Normativa de accesibilidad** es dar a los diferentes participantes involucrados en el desarrollo, integración o mantenimiento de software de la UOC, unas líneas maestras en las cuales basar la accesibilidad de todas las interfaces web y aplicaciones móviles. Todos alineados con las buenas prácticas reconocidas internacionalmente pero adaptadas a las necesidades y especificidades del Área de Tecnología de la UOC.

En aquellos desarrollos que se requiera, se incluirán requerimientos específicos por la aplicación de las guías y recomendaciones de UX en relación a la Maquetación, Wireframing, Diseño Gráfico y Guía de estilo.

### Normativa de Arquitectura

La normativa de arquitectura UOC se el marco de trabajo para la construcción de sistemas de información en la UOC. Esto es, qué principios los rigen ([Anejo - Principios de Tecnología](#)) y con qué tecnologías, diferenciando tecnologías de construcción ([Anejo - Hoja de ruta de Software](#)) y tecnologías de despliegue y servicios auxiliares para la construcción.

El Documento de Descripción de Arquitectura es el entregable donde se trabaja con el integrador la implementación concreta de principios y tecnología que aplican a la construcción de un sistema de información concreto.

### Normativa de Operaciones

La normativa de operaciones formaliza el paso a producción de una aplicación o infraestructura implantada por un proyecto, recogiendo la información necesaria porque AiS - Operaciones pueda

mantener el proyecto en producción garantizando la disponibilidad, contingencia, coherencia y seguridad de los datos y pueda resolver los posibles incidentes que puedan surgir durante la vida en servicio de la aplicación o infraestructura.

#### Anejo - Normativa de Seguridad

Normativa destinada a clarificar los derechos y obligaciones relacionadas con la seguridad de las relaciones comerciales de todas las partes involucradas en el desarrollo

### **4.4 Entornos de desarrollo**

Como norma general, los entornos de desarrollo los tendrá que proporcionar el adjudicatario, ya sea para sistemas de información que se desplegarán a CPD UOC como los que se desplieguen en Cloud UOC y se hará cargo (en caso de que hayan) de los costes asociados a su uso (acceso, licenciamiento, integración, etc..).

En el caso del Cloud UOC, se automatiza tanto el código de la aplicación en construcción (CI/CD, DevOps) como la parte de la infraestructura subyacente sobre la que se despliega este código y otros servicios que utilice (Infraestructura como Código, Terraform, GitOps).

Para montar entornos equivalentes a los que alojarán finalmente los sistemas de información construidos, la UOC facilitará plantillas de "infraestructura como código" que automatizarán la creación de la red a la nube y aquellos elementos de infraestructura base necesarios porque corran las aplicaciones.

Para poder conectar sistemas de la cuenta cloud del adjudicatario con entornos de Test del CPD UOC, se tendrán que habilitar mecanismos diversos, entre ellos un NATO Gateway (que permitirá identificar la IP del adjudicatario a los servicios de la UOC) así como VPN cuando sea necesario porque no existan servicios expuestos en el exterior por parte de la UOC.

### **4.5 Desarrollos “Cloud-Ready”**

Por norma general los nuevos sistemas que se construyan tendrán que ir al Cloud de la UOC y ser desplegados vía CI/CD. Esto implica que se tengan que construir siguiendo una serie de premisas:

- tienen que haberse desarrollado siguiendo los Principios y la Hoja de ruta de Tecnología UOC
- tienen que estar adecuadamente “componentizados”. Esto quiere decir que se tienen que estructurar en diferentes componentes los cuales tendrán su ciclo de vida independiente (Ej. infra, frontend, backend, funciones, ...)

- se construyen una única vez y se promocionan entre entornos. Esto implica que cualquier configuración tiene que realizarse mediante variables de entorno, no “harcoded” al código de la aplicación
- no escriben logs a fichero, tienen que hacerlo a la salida estándar. el entorno cloud se encargará de recolectarlos por su visualización
- tienen que poder ser monitorizados. Por tanto, hace falta que expongan endpoints de healthcheck que permitan testar su correcto funcionamiento
- no tienen que verse afectados en caso de escalado dinámico
- tienen que evitar almacenar el estado de la sesión a la aplicación
- tienen que evitar dependencias con el sistema operativo y la infraestructura subyacente mejorando su portabilidad
- tienen que utilizar protocolos resilientes (http, ssl, standard database, queuing)
- su instalación tiene que estar automatizada, no puede requerir tareas manuales
- en el caso de aplicaciones web, exponen su funcionalidad vía HTTP sin necesidad otras runtimes (aplicaciones autocontenidoas)
- diseños adaptados a la arquitectura de AWS, con soluciones Cloud Native orientadas a entornos Serverless, como por ejemplo AWS Lambda o Fargate.

En la fase de diseño técnico de la aplicación, y como complemento del Documento de Arquitectura, el integrador tendrá que llenar una Receta Cloud donde se certificarán estos puntos.

## 4.6 UX - experiencia de usuario y accesibilidad

### 4.6.1 Investigación con usuarios

Esta sección recoge aquellas actividades y metodologías vinculadas a

- La identificación y conocimiento de requerimientos y necesidades por parte de los usuarios UOC y responsables de negocio y técnicos de la UOC
- La evaluación de las propuestas y soluciones.

Principales actividades a realizar:

- Recogida de requerimientos con usuarios de la comunidad UOC (estudiantes, profesores responsables de actividad, técnicos, etc) a través de la ejecución (presencial u online) de metodologías como entrevistas, workshops, design thinking, cocreación, foco group, cuestionarios y metodologías similares, que permitan la identificación y conocimiento de los requerimientos de estos usuarios necesarios por la conceptualización de interfaces gráficas, definiciones

funcionales, definición de servicios, definición de la arquitectura de información, etc., obteniendo como consecuencia entregas y tangibles propios del ámbito de la UX y del diseño de servicio, como Personas, Escenarios, Arquitectura de información, Journeys, Service Blueprints, Mapa de empatía, etc. Contempla la realización de:

- Redacción de guión/preguntas y preparación material propio de la metodología utilizada
- Definición muestra
- Gratificación a usuarios
- Captación de usuarios
- Agendación de las sesiones con usuarios
- Ejecución de la metodología
- Análisis de resultados
- Informe final
- Iteraciones para obtener la validación del equipo UOC
- Realización de test de usabilidad/guerrilla: evaluación, presencial u online, de la usabilidad de wireframes, prototipos, diseños, desarrollos en pre-producción, en producción, etc, con usuarios de la comunidad UOC, en cualquier apoyo (papel, mockup, maqueta, web, etc.), así como en cualquier dispositivo (dispositivos móviles, ordenador, etc.), contemplando la realización:
  - Redacción de guión/preguntas y preparación material propio de la metodología utilizada
  - Definición muestra
  - Gratificación a usuarios
  - Preparación del prototipo por la test
  - Captación de usuarios
  - Agendación de las sesiones con usuarios
  - Ejecución de la test
  - Análisis de resultados
  - Informe final
  - Iteraciones para obtener la validación del equipo UOC
- Realización de card sortings, tree testing y metodologías similares, de forma presencial u online, que permitan identificar y obtener el conocimiento suficiente por la definición y creación de la arquitectura de la información de los productos tecnológicos de la UCO (web, aplicaciones, portales, etc). Contempla la realización de:
  - Redacción de guión/preguntas y preparación material propio de la metodología utilizada
  - Definición muestra
  - Gratificación a usuarios
  - Preparación del prototipo por la test
  - Captación de usuarios
  - Agendación de las sesiones con usuarios
  - Ejecución de la test
  - Análisis de resultados

- Informe final
- Iteraciones para obtener la validación del equipo UOC

Dentro de esta sección se incluye también la realización de heurísticos, recorridos cognitivos y metodologías similares.

#### 4.6.2 Conceptualización UX

Principales actividades a realizar:

- Realización de wireframes/conceptuales (prototipos de baja y alta definición donde se muestre la estructura visual y las interacciones de una interfaz de usuario) en Figma (o herramientas similares que decida la UOC) en base a los requerimientos de usuario y de la UOC, analíticas, etc, para interfaces por escritorio, responsive y app nativas, siguiendo las buenas prácticas y principios en usabilidad, diseño de interacción, arquitectura de información, diseño gráfico, responsive y accesibilidad.
- Iteraciones del wireframes para obtener la validación del equipo UOC
- Redacción de funcionalidades de interacción, y actualización del mismo a medida que las fases de proyecto avancen (documento donde se explica los itinerarios e interacciones que hace el usuario por una herramienta, explicando las diferentes opciones y funcionalidades que existen).
- Definición de casos de uso.
- Identificación de problemas de UX y propuestas de mejora.
- Apoyo y acompañamiento durante toda las fases del proyecto (diseño, maquetación, desarrollo) para garantizar y validar que los criterios y principios de UX definido y pactados se cumplen y/o para identificar de nueces.
- Ejecución de pruebas QA en materia de UX y accesibilidad durante las fases de maquetación, desarrollo y puesta en producción.
- Propuesta de copy.
- Generación de la documentación necesaria para facilitar el traspaso de la entrega a la UOC y de su posible mantenimiento y reaprovechamiento.

#### 4.6.3 Diseños gráficos (UI)

Las actividades de diseño gráfico comprenden tareas de mantenimiento y evolución del diseño gráfico de las interfaces de las herramientas tecnológicas de la UOC. Estas actividades no comprenden en ningún caso la modificación de diseños gráficos de sistemas o herramientas tecnológicas que sean de titularidad de terceros (propietarios).

Entendemos por mantenimiento las modificaciones en el diseño gráfico de las herramientas tecnológicas como resultados de identificación de errores o mejoras detectados en estos.

Entendemos por evolución las modificaciones del diseño gráfico actual o generación/creación de nuevo diseño como resultado de nuevas necesidades identificadas por la UOC.

Por regla general todos los diseños gráficos se realizarán siguiendo las guías de estilo de la UOC, UI Kits de la UOC y la guía de estilo frente-end de la UOC. En el supuesto de que las guías correspondientes no tengan soluciones adecuadas a las necesidades del proyecto, se tendrá que generar un diseño/componente gráfico nuevo.

Se encargará de las siguientes actividades:

- Realización del diseño en Figma (o herramientas similares que decida la UOC) que el proyecto requiera, para cualquier dispositivo (ordenador escritorio, dispositivos móviles, u otros), siguiendo las buenas prácticas y principios en usabilidad, diseño de interacción, arquitectura de información, diseño gráfico, responsive y accesibilidad.
- Iteraciones del diseño para obtener la validación del equipo UOC.
- Actualización y mantenimiento de las guías y UI kits de la UOC
- Apoyo puntual a las posibles consultas, dudas o necesidades sobre el diseño gráfico durante las diferentes fases del desarrollo del proyecto para garantizar y validar que criterios y principios de UI definidos y pactados se cumplen y/o para identificar de necesidades.
- Ejecución de pruebas QA en materia de UI durante las fases de maquetación, desarrollo y puesta en producción.
- Generación de la documentación necesaria para facilitar el traspaso de la entrega a la UOC y de su posible mantenimiento y reutilización.

#### 4.6.4 Front end

Las actividades de Frente end comprenden las tareas de mantenimiento y evolución de frente end de las herramientas tecnológicas actuales así como del desarrollo del nuevo frente end vinculado a nuevos proyectos que se ejecutarán según demanda de la UOC.

La alineación, adecuación y actualización de las tecnologías se tienen que aplicar como medida para frenar la obsolescencia de las soluciones de frente end. Unificar tecnologías y soluciones de mercado sólidas tienen que ser objeto de revisión en los nuevos proyectos.

Entendemos como desarrollo de frente end:

- El desarrollo de Single Page Applications (SPA) con tecnologías actuales del mercado.
- El desarrollo y mantenimiento de maquetación HTML5 según buenas prácticas.

- El desarrollo y optimización de CSS incluyendo pre-procesadores modernos de mercado (Sass, PostCSS, etc.).
- La optimización de javascript y el uso de librerías y frameworks actuales y mantenidos (ex: react, angular), así como herramientas de build modernas como webpack, vite o similares.

Todo esto enmarcado dentro de la normativa y las directrices de buenas prácticas que constan indicadas a la Normativa sobre UX y Accesibilidad que, a manera de resumen, incluye:

- Uso de los frameworks y entornos de desarrollo aprobados por la uoc (gef, outsystems, dxp, etc.).
- Posibilidad de personalizar el frente-end para poder aplicar línea gráfica UOC
- Correcto uso de la semántica y buenas prácticas en HTML5 que garanticen la accesibilidad web (nivel Doble A – WCAG 2.1).
- Orientación hacia las mejoras de rendimiento (Web Performance Optimization, incluyendo métricas de Core Web Vitales).
- Compatibilidad multi-navegador (cross-browser) con navegadores modernos segun especificación de la UOC.
- Maquetaciones con soluciones adaptativas / responsives.
- Optimización SEO en el caso de aplicaciones públicas e indexables en motores de busca

Las interfaces desarrolladas serán el resultado de una generación dinámica de contenido a partir de un catálogo de servicios proporcionados por la UOC. Esta tiene que ser capaz de renderizarse y comunicar con los servidores de la UOC y ser totalmente funcional en base a una definición de comportamientos (workflow) que vendrá definido por las especificaciones del proyecto en el cual se enmarque.

El desarrollo de frente end se hará siguiendo las directrices UI / UX identificadas en fases previas del proyecto y / o por la normativa de UX / UOC y las representaciones gráficas resultante del proceso de wireframing. Tiene que ser totalmente fiel a su definición gráfica.

Por lo tanto, el adjudicatario se encargará de:

- Realización de la maquetación y el frente-end indicado anteriormente siguiendo las buenas prácticas y principios en usabilidad, diseño de interacción, arquitectura de información, diseño gráfico, responsive y accesibilidad.
- La entrega de una maqueta funcional y dinámica preparada para ser reutilizada por el desarrollo de la herramienta o sitio web, preferentemente en formato html/css/js modular o en componentes reutilizables (ex: storybook, web componentes, etc.).
- Apoyo puntual a las posibles consultas, dudas o necesidades del equipo técnico que desarrollará la herramienta maquetada, una vez la maqueta se haya librado, para que el desarrollo se alinee con la maqueta.

#### 4.6.5 Accesibilidad

En este ámbito se incluye el conjunto de actividades transversales en materia de UX/UI durante las diferentes fases de una iniciativa que garanticen el cumplimiento de la doble A de las WCAG de accesibilidad, que incluye:

- La presa de requerimientos y la validación de prototipos conceptuales y productos tecnológicos de la UOC con usuarios UOC con diversidad funcional
- El análisis heurístico de la accesibilidad en las interfaces digitales
- Conceptualización UX/UI teniendo en cuenta la accesibilidad desde el inicio: definición de patrones UX/UI, de elementos de interacción y de diseño siguiendo los principios de diseño inclusivos (tamaño apropiado, orden de lectura de los elementos de la interfaz gráfica, interacciones no dependientes teclado, navegación con teclado, colores con contraste, etc)
- La documentación de buenas prácticas y criterios de accesibilidad de los entregables UX/UI para equipos de desarrollo, como por ejemplo las anotaciones de:
  - Las “Regiones” principales de la llanura
  - El etiquetado semántico (H1, H2 etc, listas, navegación, etc)
  - La orden de lectura
  - Sido de los elementos
- Ejecución de pruebas QA del ámbito de la accesibilidad de los desarrollos para asegurar su cumplimiento.

#### 4.6.6 Automatización y eficiencia en las pruebas QA de UX y accesibilidad con IA

Con el objetivo de garantizar la máxima calidad, eficiencia y sostenibilidad en el mantenimiento, evolución y creación de los productos y herramientas digitales de la universidad, se pide a las empresas licitadoras que incluyan en su propuesta una estrategia detallada para la automatización de las pruebas de calidad (QA) durante las diferentes fases de desarrollo (conceptualización, diseño, maqueta/frente-end, desarrollo en test/pre) mediante el uso de inteligencia artificial (IA) y/o herramientas de validación/testing u otros ámbitos con IA, orientada a los Requerimientos UX y de accesibilidad informados en el Anexo - Normativa sobre UX y accesibilidad:

- Usabilidad y diseño de interacción
- Layout, estructura, navigation y arquitectura de información
- Coherencia visual y línea gráfica institucional.
- Frente-end

- Adaptabilidad responsive y multidispositivo.
- Compatibilidad entre navegadores (cross-browsing).
- Rendimiento y eficiencia de carga.
- Optimización para motores de busca (SEO).
- Multilingüismo.
- Accesibilidad digital
- Satisfacción y comportamiento del usuario.

Se valorará positivamente la propuesta de soluciones que:

- Reduzcan significativamente los tiempos de validación manual.
- Reduzcan significativamente el tiempo que dedica la UOC a hacer validaciones de entregables y requerimientos
- Aumenten la cobertura de pruebas sin incrementar los costes operativos.
- Mejoren la trazabilidad y la toma de decisiones basada en datos.
- Faciliten la colaboración entre equipos técnicos y de diseño.

#### **4.6.7 Mantenimiento, evolución y gobernanza del sistema de diseño y del UI Kit institucional**

El mantenimiento de los diferentes UI Kits de la UOC es necesario para adaptarlo a la evolución tecnológica (nuevos frameworks, navegadores y dispositivos) con el objetivo de:

- Garantizar que los componentes cumplen la normativa de accesibilidad
- Incorporar cambios en la línea gráfica institucional,
- Evitar la duplicación de esfuerzos e inconsistencias entre proyectos y errores de diseño (especialmente cuando han varios proveedores)
- Responder a las necesidades de los diferentes proyectos digitales,
- Establecer criterios claros de versión, documentación y control de calidad.

Las tareas asociadas a este mantenimiento tendrían que incluir

- La revisión y actualización periódica de los componentes y estilos,
- La documentación clara y estructurada dentro de Figma,
- La definición de buenas prácticas de uso,

- La coordinación con equipos de desarrollo para asegurar la implementación correcta de los componentes

Este mantenimiento no es solo una tarea de diseño gráfico, sino una pieza estratégica para garantizar la calidad, la eficiencia y la escalabilidad del ecosistema digital de la universidad.

## 4.7 Calidad

A continuación se detallan las principales actividades que hay que realizar en cada etapa del proyecto en el ámbito de Calidad:

### Planificación de las pruebas

- Definir la estrategia de pruebas (alcance, cronograma, recursos, entorno, tipo de test, informes, raci pruebas).
- Configurar el Panel de Calidad del proyecto
- Configurar el plugin de Sonarqube para Jira
- Foto inicial de la calidad del código de las aplicaciones del proyecto

### Análisis y Diseño de las Pruebas

- Trabajar con historias de usuario y evaluar su "testejabilitat"
- Diseñar la configuración de la en torno a prueba.
- Identificar las diferentes tipologías de pruebas
- Gestión de los datos de prueba
- Registrar los casos de pruebas (macetas) en la herramienta de gestión de pruebas
- Configurar el plan de pruebas (testplan)
- Seguir las bolas prácticas en el diseño de casos de prueba definidas

### Inicio de las pruebas

- Configurar el entorno a prueba y asegurarse que funciona.
- Implementar datos de prueba y casos de prueba.
- Definir los criterios de entrada, salida, suspensión y reanudación de pruebas
- Verificar que el entorno puede ser probado.
- Ejecución de pruebas de humo

### Ejecución de las pruebas

- Ejecutar los protocolos de prueba (suites, casos, tareas) y analizar sus resultados.
- Registrar las evidencias en la herramienta de gestión de pruebas.
- Informar de los defectos y seguimiento hasta el cierre
- Evaluar los resultados de las pruebas

### Evaluación de las Pruebas

- Verificar la finalización de cada trabajo de prueba, teniendo en cuenta la corrección de defectos.
- Evaluar si cada nivel de pruebas consigue sus criterios de salida.
- Generar un informe de Resultado de las pruebas para las partes interesadas.
- Registrar el informe en el espacio del proyecto de Confluence.

### Seguimiento de las pruebas

- Analizar la correspondencia del estado actual de las actividades y tareas de pruebas con el planificado y los objetivos estratégicos.
- Comparar el progreso real de la prueba versus el progreso planificado y tomar medidas correctivas si es necesario.
- Gestión de riesgos de las Pruebas
- Revisar y asegurar la cobertura de pruebas
- Gestión y seguimiento de defectos

### Cierre de las Pruebas

- Asegurar que todas las actividades previstas a la Estrategia de pruebas han sido completadas y así lo refleja el TestPlan correspondiente.
- Discutir y documentar las lecciones aprendidas del proceso de pruebas.
- Asegurar que los buenos procesos se repitan y los malos se eliminan en procesos de pruebas futuros.
- Archivar o cerrar todo el software de prueba (Milestones a TestRail, Planes de prueba a TestRail, scripts, en torno, casos, informes, etc.) después del proceso de pruebas y asegurarse que toda la información está recopilada.
- Pasar la Checklist de Calidad antes de cerrar el proyecto
- Recordar que el proyecto no puede haber empeorado la calidad de la aplicación. (Comparación foto inicial calidad y foto final)

## 4.8 Herramientas

La UOC determinará y/o proporcionará las herramientas que soportan los procesos para gestionar y gobernar los servicios.

se tendrán que cumplir los siguientes condicionantes:

- El adjudicatario tendrá que usar las herramientas propuestas por la uoc en las condiciones que éste establezca.
- El adjudicatario se hará cargo (en caso de que haya) de los costes asociados al uso de estas herramientas (acceso, licenciamiento, integración, etc..).
- El adjudicatario podrá proponer modificaciones a las herramientas para obtener una mejor eficiencia y calidad en el servicio, siempre que se asegure la continuidad de los acuerdos de nivel del servicio. cualquier petición de cambio tendrá que estar documentada previamente porque la uoc pueda analizar y autorizar la conveniencia de su implantación.
- El adjudicatario podrá hacer uso de herramientas adicionales, previa autorización de la uoc. esto no lo exime del cumplimiento y del uso de las herramientas que haya determinado la uoc. El uso de estas herramientas adicionales no puede deteriorar el servicio o suponer un sobrecoste para la UOC y no puede poner en riesgo la continuidad del servicio después de la finalización de la relación contractual. la propiedad de la información será de la uoc y el adjudicatario tendrá que declarar el uso que hace de esta información.
- La UOC podrá evolucionar las herramientas escogidas en cualquier momento de la duración del contrato.
- La UOC se reserva el derecho de incorporar nuevas herramientas. En cualquier caso, se dará un preaviso a los proveedores de un mínimo de 2 meses antes de su implantación.

Actualmente, para el registro y tratamiento de las peticiones e incidencias de los servicios objeto de este contrato se está usando la herramienta JIRA de Atlassian.

El entorno de desarrollo UOC se compone de un conjunto de herramientas que se tendrán que utilizar en cualquier tarea de desarrollo y mantenimiento, siguiendo unos procedimientos previamente marcados por el paso de las aplicaciones entre entornos, por las pruebas y por la aceptación.

Las herramientas implicadas al proceso de desarrollo son:

- Herramienta de seguimiento de errores, incidencias y gestión operativa de proyectos: JIRA + Plugin eazyBI + Otros Plugin
- Herramienta de gestión de pequeños evolutivos: JIRA
- Herramienta de gestión de proyectos evolutivos: JIRA + Confluence
- Repositorio de código, donde tiene que residir todo el código desarrollado: GitLab

- Herramienta de test funcional: Plugin XRAY de JIRA.
- En torno a integración continua por los desarrollos, que se encarga de tirar las compilaciones e informes de calidad de los desarrollos de forma automática: Jenkins
- Monitorització de la calidad del código: SonarQube
- Herramienta de documentación integrada con la de seguimiento: Confluence y Drive
- VPN, que permite a usuarios externos acceso en la red de desarrollo y de pruebas
- eTreball, en torno a Teletrabajo que permite trabajar de manera remota a la red de la UOC, necesaria para acceder a los recursos de la red interna y circuitos de distribución de aplicaciones

## 4.9 Auditorías

La UOC podrá realizar auditorías para verificar el cumplimiento de los compromisos contractuales y la fiabilidad de la información facilitada.

El adjudicatario proporcionará su total cooperación a la realización de estas auditorías. Esto incluirá la entrega de documentación y el acceso físico en las instalaciones donde se estén realizando los servicios objeto del contrato, al personal de la UOC o a los terceros subcontratados.

No habrá que dar aviso previo para realizar tareas de auditoría donde no se requiera colaboración activa del personal del adjudicatario. En los casos en que la UOC pida una colaboración activa del personal del adjudicatario, se dará aviso con una semana de antelación.

## 4.10 Perfiles y funciones

El equipo de trabajo del adjudicatario estará integrado, además de los perfiles descritos en el apartado *4.1.1 Estructura de responsabilidades*, como mínimo, por los perfiles siguientes. Cada perfil incluirá las funciones y responsabilidades indicadas en cada caso:

### Responsable del Servicio

El adjudicatario designará un único responsable de todos los servicios por el contrato que asumirá las siguientes funciones:

- Garantizar la atención a los responsables del Servicio de la UOC en todos los ámbitos del servicio, atendiendo a las necesidades de negocio de la UOC.
- Mantener la visibilidad global de la prestación del servicio, en todas sus dimensiones (servicio recurrente, servicios adicionales y Servicios de desarrollo bajo demanda).
- Garantizar la calidad del servicio en todos sus ámbitos.

- Garantizar el desempeño de los Niveles de Servicio.
- Garantizar la correcta organización del equipo de trabajo por el desempeño con el servicio acordado.
- Garantizar la capacidad operativa, el dimensionado, del equipo en el decurso del servicio.
- Seguimiento y control de los recursos asignados a los servicios.
- Control de costes, estimación de esfuerzos y su seguimiento.
- Análisis de las desviaciones del servicio (en alcance y coste).
- Elaboración de los informes de servicio y justificación del cumplimiento de los ANTES.
- Asegurar la buena colaboración entre los diferentes adjudicatarios con quienes se tiene que relacionar para mejorar el servicio de negocio final.
- Proponer e incorporar, si son aceptadas, mejoras en la gestión global del servicio.
- Formar parte del comité de dirección.

### **Ninguno de proyecto**

Definición de los estándares y metodologías de acuerdo con la normativa UOC vigente

- Planificación de las actividades del proyecto
- Gestión y seguimiento diario del proyecto
- Gestión de los recursos asignados al proyecto
- Coordinación con los proveedores otros sistemas de la UOC que tengan dependencias con el proyecto
- Gestión del alcance, el calendario, el presupuesto, la calidad, las comunicaciones y los riesgos del proyecto
- Gestión de cambios

En la prestación de servicios de desarrollo de proyectos evolutivos ágiles, sus funciones como Scrum Master son:

- Promover los valores, los principios y las prácticas scrum
- Gestiona los impedimentos y conflictos del equipo
- Ayudar al Product Owner con la comunicación con el equipo de trabajo, gestionar el backlog...
- Ayudar al Equipo de trabajo a mantener el foco, a ser autosuficiente, a aprovechar los artefactos de Scrum.

### **Arquitecto**

- Definir el diseño conceptual de la solución y la arquitectura de componentes tecnológicos necesarios para cubrir las necesidades del proyecto específico.

- Participar en la presa de requerimientos, identificar las necesidades y definir las propuestas de solución del proyecto específico.
- Definir las diferentes opciones posibles para dar cobertura a los objetivos que se pretenden conseguir dentro del proyecto específico, determinando la mejor solución posible de todas las disponibles, siempre con la visión de diseñar un sistema el más parametrizable, flexible y escalable posible.
- Proporcionar las especificaciones funcionales de los servicios de integración con otros sistemas con los que tenga relación la nueva solución.

**Consultor Sénior / Analista sénior**

- Realizar la presa de requerimientos, identificar las necesidades y definir las propuestas de solución.
- Realizar la presa de requerimientos de los desarrollos a medida que sean necesarios y la documentación funcional de diseño y de pruebas.
- Realizar la construcción de los procesos de negocio en base a las especificaciones proporcionadas por el analista de procesos. Construirá los sistemas de medida, notificación y control.
- Dar las especificaciones funcionales de los servicios de integración con otros sistemas con los que tenga relación el nuevo sistema.
- Especifica los planes de pruebas.
- Coordinar y participar en los desarrollos que sean necesarios según las especificaciones funcionales y de diseño.
- Realiza las pruebas funcionales y técnicas del sistema.
- Participa en las actividades derivadas de los planes de formación, comunicación y apoyo puesto-implantación.
- Realizar su trabajo conforme a los objetivos marcados por la dirección de proyecto en el desarrollo y construcción del sistema.

**Analista Desarrollador****Desarrollo frente-end:**

- Desarrollo de aplicaciones webs (single page applications) y llanuras web en HTML5 , CSS3 y programación (javascript, etc) para diferentes dispositivos (Responsive Web Design, Mobile First y Crossbrowser), siguiendo estándares de accesibilidad a un nivel de doble A.
- Uso de estándares web y accesibilidad (w3c, wcag, aria) y validadores de accesibilidad del w3c y otros

- Orientación y dedicación a la correcta aplicación de los diseños propuestos con atención a los detalles visuales y la correcta utilización del framework propio de la UOC
- Orientación al uso de vanilla JS ante bibliotecas como jQuery y a buenas prácticas en desarrollo Javascript
- Uso de frameworks: Angular, React, Vue, Bootstrap, Material Design, etc.
- Maquetación de layouts para CMS como OpenCMS, Liferay, Wordpress, etc
- USO DE NODO.JS PARA LA CONSTRUCCIÓN Y OPTIMIZACIÓN DE LOS ACTIVOS A LIBRAR.
- Control de versiones con Git
- Uso de task runners
- Experiencia en desarrollo con SASS, LESS, y uso de linting que garanticen las buenas prácticas en optimización.
- Experiencia en el uso de CDN.
- Experiencia en la optimización de los activos de frontend y mejora de la performance web siguiendo las mejores prácticas del mercado. Entre otros: lazy loading, non blocking resources, optimización de imágenes, compresión y bundling.
- Optimización SEO en el caso de aplicaciones públicas e indexables en motores de busca

**Desarrollo backend:**

- Programación en lenguajes incluidos dentro de la Hoja de ruta de Tecnología UOC: Java, Nodo.js, Python, Golang
- USO DE FRAMEWORKS DE DESARROLLO COMO POR EJEMPLO SPRING FRAMEWORK (JAVA) Y EXPRESS (NODO.JS)
- Experiencia en el uso de herramientas de gestión de proyectos como Maven (Java) o Npm (Nodo.js)
- Conocimiento de herramientas RAD (Rapid Application Development) como JHipster por la generación de aplicaciones Java y Angular
- Experiencia en bases de datos tanto relacionales (MySQL, PostgreSQL) como no relacionales (MongoDB, DynamoDB).
- Conocimiento de sistemas de caché como Redis o Memcached
- Experiencia en mecanismos de desacoplamiento de sistemas, como por ejemplo colas, sistemas de notificación, publicación y suscripción
- Experiencia en el uso de servicios Cloud por storage (Ej. S3), indexación (Ej. ElasticSearch), ...
- Experiencia en integración de sistemas, principalmente RISTRA APIO y SOAP
- Control de versiones con repositorios Git y uso de herramientas integración y despliegue continuo (Jenkins, SonarQube, Artifactory). Maquetación de layouts para CMS como OpenCMS, Liferay, Wordpress, etc
- Oráculo Forms

- PL/SQL de Oracle y PostgreSQL

## Calidad

### Responsable Calidad del Servicio

Rol calificado en Calidad que supervisa y apoya en temas de calidad a los diferentes proyectos y aplicaciones del servicio.

- Sesiones periódicas de Seguimiento de Calidad con la persona Responsable de Calidad UOC
- Control de la realización, Realización o apoyo en temas de Calidad a los proyectos y mantenimientos del servicio como:
  - Definición de estrategias prueba de proyectos
  - Resolución de issues referentes a la Calidad de Código
  - Cumpliment de la normativa de Nomenclatura
  - Definición historias de usuario siguiendo las buenas prácticas, recomendaciones y estándares.
  - USO HERRAMIENTAS SIGUIENDO LAS GUÍAS Y NORMATIVAS
  - Cumplimiento Normativo
  - Definición de casos de prueba (o supervisión)
  - Control actualización Fichas Proyecto y Fichas Aplicación
  - Aseguramiento correcto de la documentación
- Apoyo en el despliegue de las aplicaciones (CICD)
- Resolver dudas internas y hacer sesiones de difusión en el ámbito de Calidad
- Proponer acciones de mejora tanto en los procedimientos como las aplicaciones
- Apoyo en la definición y mantenimiento de pruebas automatizadas
- Velar por la resolución de No conformidades y Riesgos del servicio en el ámbito de Calidad
- Control del correcto versionado de las aplicaciones

### Responsable Calidad del Proyecto

- Coordinación con la persona Responsable del Calidad de Servicio y los Testers del proyecto.
- Aseguramiento de la correctesa de las historias de usuario
  - USO DE LAS PLANTILLAS CORRECTAS
  - Completitud de los apartados básicos
  - apoyo en el seguimiento de buenas prácticas, recomendaciones y estándares.
- Definición de la estrategia de pruebas del proyecto

- Planificación de las pruebas
- Definición del alcance de las pruebas
- Gestión de los Riesgos de las pruebas del proyecto
- Definición de los Criterios de entrada, de salida, de suspensión y de reanudación de pruebas
- Supervisión de los datos y entornos de pruebas
- Definición de las pruebas del proyecto
  - Validación de historias de usuario aptos para ser desarrolladas y testadas
  - Supervisión de la correctitud de la definición y clasificación de las pruebas manuales y automatizadas
  - Supervisión en la elección de los casos de prueba a automatizar y del seguimiento de buenas prácticas establecidas
  - Supervisión de la tarea del Rol de Tester e ingeniero de automatización del proyecto
  - Supervisión de los datos y entornos de pruebas
  - Validación y reporting de la cobertura en historias de usuario por pruebas
- Apoyo y organización de las UAT
  - Calendarització
  - Selección de las UAT
  - Apoyo y formación a los usuarios en el uso de la herramienta xray
  - Apoyo en la ejecución y reporting de UAT
- Velar por el cumplimiento normativo previo a cada promoción entre entornos
- Velar por el cumplimiento de la normativa de Nomenclatura
  - Actualización Fichas Proyecto y Fichas Aplicación
- Velar por el cumplimiento y actualización de la estrategia de pruebas
- Velar por el desempeño y completitud de la documentación del proyecto según la calculadora de entregables o normativa vigente.
- Apoyo en el despliegue de las aplicaciones (CICD) y resolución de incidencias reportadas
- Velar por la resolución de No conformidades y Riesgos del servicio en el ámbito de Calidad
- Control del correcto versionado de las aplicaciones
- Control de versiones y entregas:
  - Coordinar los calendarios de versiones asegurando calidad y estabilidad en las entregas.
  - Validar que las funcionalidades sean conformes a las especificaciones.
  - Publicar un release notes cuando sea necesario con los cambios de versión de la aplicación
- Colaboración interequipos:
  - Trabajar conjuntamente con desarrolladores, arquitectos y product owners para asegurar la calidad desde la fase de concepción.

## Tester a Proyectos

**El Rol de tester a los proyectos tiene que ser una persona diferente a los desarrolladores de aquella aplicación.**

- Análisis de requisitos:
  - Revisar la documentación funcional y técnica para entender qué se tiene que validar.
  - Identificar casos límites, escenarios complejos y puntos críticos del sistema.
- Diseño de casos de prueba:
  - Crear escenarios de test detallados, tanto funcionales como no funcionales.
  - Documentar los pasos, entradas, salidas esperadas y criterios de aceptación.
- Ejecución de pruebas:
  - Realizar pruebas manuales o automáticas en entornos de desarrollo, test y preproducción.
  - Validar que las funcionalidades funcionen correctamente y que no se produzcan regresiones.
- Gestión de errores:
  - Reportar defectos encontrados de manera clara y estructurada (con capturas, logs, etc.).
  - Hacer seguimiento hasta su resolución y revalidación.
- Validación de rendimiento y seguridad:
  - Hacer pruebas de carga, escalabilidad y resiliencia.
  - Verificar vulnerabilidades y comportamientos ante escenarios extremos.
- Colaboración con equipos técnicos:
  - Trabajar con desarrolladores, analistas y product owners para resolver incidencias y validar soluciones.
  - Participar en las ceremonias Scrum si trabaja en un entorno Agile.
- Documentación e informes de calidad:
  - Elaborar informes de resultados de test, cobertura e índices de defectos.
  - Proponer mejoras continuas al proceso de test.
- Coordinación total y reporting al Responsable de Calidad del Proyecto

## Ingeniero de Automatización de Pruebas de Calidad

- Análisis de casos de prueba automatizables:
  - Evaluar qué escenarios tienen sentido automatizar (macetas repetitivas, regresiones, funcionalidades críticas, end to end).
  - Priorizar en función del impacto y la complejidad.
- Diseño y desarrollo de scripts de test:

- Escribir scripts intermediando frameworks como Selenium, Cypress, Playwright, Appium, entre otros.
- Utilizar lenguajes como Javascript, Python, Java o C# según la tecnología del producto.
- Integración en entornos de integración continua (CI/CD):
  - Utilizar las pipelines del CI/CD porque las pruebas se ejecuten automáticamente con cada despliegue (GitHub Actions, Jenkins, Azure DevOps...).
  - Asegurar que los automatismos se integran correctamente a la pipeline propuesta por la UOC
- Análisis de resultados y seguimiento de errores:
  - Analizar los logs de ejecución para identificar las causas de errores.
  - Reportar defectos detectados en el proceso automatizado.
- Colaboración con los equipos de desarrollo y QA manual:
  - Sincronizarse para asegurar que cubrimos todas las funcionalidades importantes.
  - Apoyar a los testers manuales en la migración hacia la automatización.
- Documentación técnica:
  - Generar manuales de uso y mantenimiento de las macetas automatizadas.
  - Crear dashboards de cobertura y rendimiento de las pruebas.

## UX y Accesibilidad

### Responsable UX y de accesibilidad del servicio

- Responsable transversal en el ámbito UX y de accesibilidad en las iniciativas y mantenimientos ejecutados por el partner
- Velar por la calidad y los resultados de las actividades y entregables UX y de accesibilidad como responsable último del servicio siguiendo la Normativa sobre UX y accesibilidad (Anexo) y de cumplir con los criterios de aceptación de cada entrega UX.
- Conocer la normativa y procedimientos UX y de accesibilidad, difundirla entre los roles del servicio /partner participantes en las iniciativas y mantenimientos y asegurar su cumplimiento
- Asegurar la disponibilidad y calidad de los perfiles necesarios para la ejecución de las actividades UX y de accesibilidad.
- Planificar las actividades UX y de accesibilidad de manera coordinada e integrada con el conjunto de las actividades del Lote en general
- Identificar y proponer elementos, metodologías y actuaciones de mejora en el ámbito de la UX
- Actuará como enlace entre el adjudicatario y los diferentes agentes implicados cuando se traten temas de UX.
- Conocer los Servicios Digitales impactados por las iniciativas ejecutadas por el Servicio/Partner
- Conocer las plataformas y/o aplicaciones tecnológicas que sustentan los Servicios Digitales.

- Identificar y proponer actividades, metodologías, entregables y actuaciones de mejora en materia de UX y accesibilidad por el ámbito y Servicio Digitales que le correspondan.
- Formar parte del comité de UX. Participará en estos comités para realizar el seguimiento, evaluación y report de la actividad UX y de accesibilidad en las iniciativas

**Responsable UX en una iniciativa/proyecto**

- Decidir y describir el enfoque UX en la propuesta de solución de la iniciativa: requerimientos, metodología, planificación, definición de entregables, tipología de roles UX/UI, etc
- Completar/informar del enfoque, objetivos, actividades y entregables en materia UX en los documentos propios de una iniciativa (Kickoff, plan de pruebas, UAT, cierre, etc)
- Coordinar las actividades que ejecuta el rol UX/UI del Partner
- Asegurar la calidad de los entregables UX y de accesibilidad y de la calidad UX y de accesibilidad de los desarrollos en la iniciativa
- Conocer la normativa y procedimientos UX y de accesibilidad y asegurar su cumplimiento en la iniciativa.
- Conocer las plataformas/aplicaciones tecnológicas objeto de la iniciativa
- Conocer los SD impactados por las iniciativas.

**Perfil UX / Service Designer / User Researcher**

- Ejecución de las actividades vinculadas a la
  - Recogida de requerimientos (investigación con usuarios)
  - Conceptualización de las interfaces y de su calidad
  - Evaluación usabilidad
- Interpretación y conceptualización de los resultados obtenidos en la investigación con usuarios
- Elaboración de los entregables pertinentes (wireframes, funcional de interacción, arquitectura de información, y resto de informes/tangibles resultantes de las metodologías propias del ámbito UX del Service Design) en el formato pedido.
- Conocer y aplicar la normativa UX y de accesibilidad
- Asegurar que la conceptualización de las interfaces cumple con los requerimientos de negocio, funcional y técnico (interoperabilidad, arquitectura datos, etc)
- Pruebas QA de UX/UI y accesibilidad
- Uso de figma y herramientas similares a la conceptualización/wireframing
- Uso de herramientas de analítica ux
- Propuesta de copy
- Realización de heurísticos

- Actualización, mantenimiento y documentación guías de estilo y UI Kits de la UOC

**Perfil UI (diseño gráfico)**

- Diseño de interfaces gráficas y de interacción, en diferentes interfaces (responsive web design, apps, etc)
- Diseño de aspectos estáticos (organización visual del contenido) y dinámicos e interactivos (flujos de navegación, etc.)
- Uso de figma y herramientas similares que sean las principales herramientas de diseño gráfico del mercado.
- Conocimiento en UX y en los principios básicos de usabilidad, diseño de interacción, arquitectura información, accesibilidad, etc.
- Conocer y aplicar la normativa UX y de accesibilidad
- Asegurar que la conceptualización de las interfaces cumple con los requerimientos de negocio, funcional y técnico (interoperabilidad, arquitectura datos, etc)
- Pruebas QA de UX/UI y accesibilidad
- Actualización, mantenimiento y documentación guías de estilo y UI Kits de la UOC

**Experto/a en accesibilidad digital**

El experto/a en accesibilidad es la persona responsable de garantizar que los productos y servicios digitales desarrollados cumplan con los estándares de accesibilidad establecidos por la normativa vigente (doble a de accesibilidad según las wcag 2.1/2.2) y otras directrices aplicables en el ámbito de la administración pública y la educación superior.

Este perfil puede actuar de manera transversal en varios proyectos o como referente específico dentro de una iniciativa concreta, y trabajar estrechamente con los equipos de desarrollo, diseño, UX y calidad.

Sus funciones principales serían

- Evaluar la accesibilidad de productos digitales (webs, apps, documentos, interfaces) mediante herramientas automáticas y pruebas manuales.
- Realizar auditorías de accesibilidad y generar informes con resultados, recomendaciones y prioridades de mejora.
- Supervisar el cumplimiento normativo segundos la UNE-EN 301549 y las WCAG 2.1/2.2, y velar por su aplicación en todas las fases del ciclo de vida del producto.
- Colaborar con los equipos de diseño y desarrollo para asegurar que los componentes, interfaces e interacciones sean accesibles desde su diseño inicial (accessibility by design).
- Participar en las pruebas QA de UX/UI y accesibilidad, incluyendo pruebas con lectores de pantalla, navegación por teclado y otras tecnologías de asistencia.

- Contribuir a la documentación del sistema de diseño y del UI Kit, asegurando que los componentes cumplan los requisitos de accesibilidad.
- Hacer seguimiento de las incidencias de accesibilidad detectadas y validar la resolución.
- Proponer mejoras continuas en metodologías, procesos y herramientas para garantizar una accesibilidad efectiva y sostenible.

**Experto SEO**

La empresa adjudicataria tendrá que disponer de uno/a experto/a en seo que pueda ser asignado/da a los proyectos digitales que se desarrollen en el marco de las tecnologías de la universidad. este perfil tendrá que garantizar que cualquier proyecto (web, aplicación, microsite, etc.) cumpla con los estándares actuales de optimización para motores de búsqueda, contribuyendo así a la visibilidad, accesibilidad y eficiencia de los activos digitales de la institución.

Funciones específicas:

- Velar por el cumplimiento de los estándares SEO en cada proyecto asegurando que la arquitectura de la información, el código y los contenidos cumplan con las buenas prácticas SEO establecidas por la universidad.
- Trabajar conjuntamente con desarrolladores, diseñadores y jefes de proyecto para integrar criterios SEO desde las fases iniciales del proyecto.
- Participar en la revisión de maquetas, estructuras de contenido, etiquetado HTML, metadatos, tiempos de carga y adaptación responsive.
- Hacer recomendaciones para mejorar el posicionamiento orgánico, la indexación y la accesibilidad de los proyectos digitales.
- Elaborar informes de auditoría SEO iniciales y finales para evaluar el grado de cumplimiento de los requisitos y detectar posibles áreas de mejora.
- Velar por la correcta configuración de Google Search Consuelo, Google Analytics y otras herramientas de seguimiento del rendimiento SEO.
- Contribuir a la definición y actualización de guías internas de SEO para garantizar la coherencia y calidad de los proyectos digitales institucionales.

Tal como se pide a la propuesta técnica en el pliego de prescripciones técnicas el adjudicatario tendrá que presentar una estructura organizativa correctamente dimensionada en cada uno de los perfiles para asegurar en todo momento el éxito en la prestación de los servicios.

Dada la criticidad de este servicio, debido a que es el sistema de información clave de la UOC a la hora de gestionar transversalmente el pago y cobro en todos los procesos de la organización y que aplica tanto al servicio recurrente como las necesidades evolutivas, los requerimientos de adscripción de perfiles al servicio pedidos a la cláusula R del PCP hará que no todos los recursos tengan que estar necesariamente activos durante todo el periodo de contrato pero que resultarán necesarios para garantizar la continuidad del servicio con los SLAs establecidos.

Por el mismo motivo de criticidad, el adjudicatario tendrá que cumplir con las siguientes condiciones respecto a la posible rotación de los perfiles adscritos al servicio:

- Aviso con un mínimo de antelación de la salida de un perfil del equipo de trabajo: 1 semana,
- Tiempo máximo por la sustitución de una baja de un perfil del equipo de trabajo: 2 semanas, previo presentación y aprobación de su sustituto por los Responsables de la UOC.

## Seguridad

### Responsable de Seguridad

- Gestión de vulnerabilidades: Identificar, evaluar y corregir vulnerabilidades en el código, infraestructura o componentes externos.
- Monitorización continua: Supervisar comportamientos sospechosos, accesos no autorizados y anomalías en el rendimiento.
- Actualización segura: Validar que las actualizaciones, parches o modificaciones cumplan con las políticas de seguridad.
- Control de accesos: Definir y mantener los permisos de acceso según los roles y necesidades, aplicando el principio del mínimo privilegio.
- Auditorías y cumplimiento: Hacer seguimiento de logs, preparar informes de seguridad y garantizar que se cumplen las normativas (como ISO 27001 o GDPR).
- Gestión de incidentes: Establecer protocolos de actuación ante posibles ataques o escapes de información.

## 4.11 Garantía de los desarrollos

Los proyectos derivados de este contrato, disfrutarán de una garantía en el desarrollo correspondiente al 50 % de la duración del proyecto (o de 3 meses en proyectos de duración inferior a los 6 meses) a partir de la finalización del proyecto y la firma del acta de recepción final del trabajo.

El adjudicatario estará obligado durante este periodo a resolver las anomalías detectadas imputables al adjudicatario, si así lo solicita la uoc.

Esta garantía incluirá la corrección de errores y comportamientos incorrectos, que se ponen de manifiesto en el funcionamiento de las aplicaciones (incluidos los artefactos provenientes de cualquier fase del proyecto y el impacto que este provoque en fases posteriores) . Los productos librados como consecuencia de la corrección de errores se harán en conformidad con el presente pliego, y por tanto disfrutarán de un nuevo periodo de garantía

## 4.12 Gestión de la obsolescencia tecnológica

Con el fin de evitar la obsolescencia tecnológica de las aplicaciones del lote, el adjudicatario realizará de forma proactiva, y con una periodicidad mínima semestral, un análisis de riesgos de obsolescencia tecnológica de los diferentes componentes (lenguajes de programación, middlewares, bases de datos, ...), proponiendo y valorando las actuaciones de actualización tecnológica sobre las aplicaciones y sus correspondientes plataformas baso para alinearse con la versión soportada de la Hoja de ruta de Software de la UOC (que se actualiza dos veces el año).

La UOC gestionará con cada ámbito la decisión de ejecución de estas actuaciones, que serán tratadas como servicios bajo demanda (excepto aquellas consideradas dentro del servicio recurrente adaptativo).

Las peticiones habrá que tratarlas con un equipo dedicado, a fin de no interferir en el servicio RUN estable ni los proyectos Bajo Demanda existentes.

## 4.13 Gobierno de los datos

El servicio de mantenimiento de aplicaciones contribuye a la gobernanza de los datos mediante actividades que garantizan la calidad, la seguridad y la trazabilidad de los datos gestionados por las aplicaciones. Estas incluyen:

- Gestión de la calidad de los datos: Validación, corrección y monitorización de los datos para asegurar la precisión, la consistencia y la completitud.
- Control de acceso y seguridad: Implementación y mantenimiento de políticas de acceso a los datos, de acuerdo con las normativas de privacidad y seguridad.
- Documentación de metadatos: Actualización de diccionarios de datos, definiciones de campos y estructuras de bases de datos.
- Apoyo a la trazabilidad: Registro de los cambios en los datos y procesos para facilitar auditorías y análisis históricos.
- Cumplimiento normativo: Asegurar que las aplicaciones cumplan con regulaciones como el RGPD, LOPDGDD, RÍA, ISO 27001, etc.
- Gestión de incidencias de datos: Resolución de errores o inconsistencias detectadas en los datos por parte de los usuarios o sistemas.

- Gestión de alertas: el adjudicatario del lote ante la modificación o alteración del catálogo de datos y/o procesos, identificará y alertará a la uoc de estas modificaciones sobre los elementos de arquitectura de empresa de la uoc.

## 4.14 Adecuación a las variaciones del servicio

Dado al dinamismo de la UOC, durante la vigencia del contrato pueden producirse cambios en la arquitectura que soporta los sistemas objeto del contrato o en sus aplicaciones, cambios en el entorno tecnológico, cambios en las actividades a llevar a cabo, cambios en las metodologías aplicables, cambios organizativos y, en general, cambios que pueden incidir de alguna manera en la prestación de los servicios pedidos.

Cualquier cambio será debidamente documentado y notificado por la UOC a la empresa adjudicataria y, siempre que no suponga un riesgo en el cumplimiento de la calidad del servicio establecido contractualmente entre la UOC y la empresa adjudicataria, se hará sin coste adicional. En caso contrario, se evaluarán de forma consensuada las necesidades adicionales de servicio y, de mutuo acuerdo, se podrán hacer las modificaciones en los compromisos y niveles de servicio acordados para adaptarlos a las nuevas necesidades, siempre con la conformidad previa y explícita por parte de la UOC.

Las cargas puntuales de trabajo tendrán que ser asumidas por el servicio sin afectar los MÁS BIEN ni el precio del contrato. El adjudicatario tendrá que ser capaz de afrontar incrementos puntuales de un mínimo del 20 % en las métricas sin ninguna afectación en la calidad del servicio ni en el precio.

EL ADJUDICATARIO TENDRÁ QUE ADAPTARSE A LAS VARIACIONES (REDUCCIONES O AMPLIACIONES) DE LAS CARGAS DE TRABAJO, O CAMBIOS, QUE AFECTEN A LA DEMANDA DEL SERVICIO. Hará falta que se proponga, dentro de la propuesta técnica, tal como indica el PCP, un modelo flexible de prestación del servicio que contemple las posibles variaciones de la demanda, al producirse de forma continuada cambios en la volumetría del servicio más allá del ± 15 % de la prevista a la propuesta inicial.

EL ADJUDICATARIO INFORMARÁ PERIÓDICAMENTE EL CONSUMO POR CADA SERVICIO/ACTIVIDAD Y LAS DESVIACIONES RESPECTO LA PREVISIÓN INICIAL. En las reuniones de seguimiento se gestionarán las posibles desviaciones tomando las decisiones que correspondan, como pueden ser, por ejemplo, decidir el traspaso de horas entre los ámbitos del servicio o actividades.

Todo el anterior sin perjuicio de que en materia de modificaciones contractuales regirá aquello previsto al Pliego de Cláusulas Particulares.

## 4.15 Otras condiciones del servicio

#### **4.15.1 Horario del servicio recurrente**

Este horario se refiere al periodo que el adjudicatario tiene la obligación de atender al usuario, ya sea directamente o indirectamente a través de un nivel de atención superior.

- Horario: lunes a viernes de 08:30 a 18:30h.

Más allá de este horario, se pactarán los horarios concretos de las guardias y actuaciones fuera de horario habitual.

#### **4.15.2 Ubicación del servicio**

Los servicios se prestarán en general desde las instalaciones del adjudicatario, a excepción de las actividades que la UOC estime en cada momento que para dar un mejor servicio conviene que se lleven a cabo a las instalaciones de la UOC.

En caso de que la UOC considere necesaria la prestación puntualmente o temporalmente desde sus instalaciones el adjudicatario proporcionará por cada puesto de trabajo de los recursos materiales y técnicos necesarios para esta eventualidad.

#### **4.15.3 Calendario de trabajo**

El calendario laboral de la UOC no contempla períodos de cierre por vacaciones y, por lo tanto, el servicio contratado tendrá que estar disponible durante todo el año incluyendo vacaciones de verano, Semana Santa y Navidad.

Sin embargo, la dedicación en períodos de vacaciones, teniendo en cuenta el histórico, es muy inferior al resto de períodos.

#### **4.15.4 Desplazamientos**

Los costes de cualquier desplazamiento a las dependencias de la UOC para la resolución de peticiones, incidencias, asistencias técnicas, reuniones de seguimiento o cualquier otra tarea contemplada dentro del servicio serán a cargo del adjudicatario y con medios de transporte facilitados por él mismo.

#### **4.15.5 Comunicaciones**

Los costes de las comunicaciones (voz y datos) e intercomunicaciones entre los centros de trabajo del adjudicatario y de la UOC, ocasionados por la prestación del servicio, serán a cargo del adjudicatario.

#### **4.15.6 Idioma**

Obligatoriamente, el adjudicatario desarrollará los sistemas e interfaces de usuario externas en los idiomas requeridos en cada aplicación que como mínimo serán el catalán y castellano. En cuanto a la documentación de gestión y documentación técnica requerida y librada durante la ejecución del contrato, el idioma será el catalán o el castellano.

## 5. Fases de la prestación del servicio recurrente

En la prestación de los servicios recurrentes se contemplan las siguientes fases que cubrirán todo el periodo de duración del contrato.

- Fase de Transición del Servicio
- Fase Regular del Servicio
- Fase de Devolución del Servicio

### 5.1 Fase de Transición del Servicio

La propuesta del licitador respecto al plan de transición del servicio, se especificará teniendo en cuenta la terminología siguiente:

**adjudicatario saliente:** es el proveedor que en la actualidad se hace cargo del servicio o servicios objeto de este pliego.

**adjudicatario entrante:** es el adjudicatario futuro y, por lo tanto, será el responsable de la provisión del servicio o servicios objeto de la licitación.

**fase de captura de conocimiento:** es la fase previa dentro de la fase de transición, durante la cual, y antes de que se inicie la ejecución del contrato, el adjudicatario entrante realiza con el apoyo del adjudicatario saliente la captura del conocimiento y la transferencia tecnológica necesaria que le permitirá lograr la provisión definitiva del servicio al final de la fase de transición. esta fase no será facturable.

**fase asunción del servicio:** es el periodo de tiempo, dentro de la fase de transición, en la que el adjudicatario entrante empieza a asumir el servicio con el apoyo y acompañamiento del adjudicatario saliente y que finaliza en el momento que el adjudicatario entrante se hace cargo completamente del servicio asumiendo los niveles de servicio requeridos.

#### 5.1.1 Modelo de Transición del Servicio

El modelo de transición, que contempla la etapa de captura de conocimiento y la fase de asunción del servicio, tiene una limitación temporal máxima de 2 meses a sufrir de la fecha de formalización del contrato. Finalizará en el momento en que el adjudicatario entrante asume completamente su prestación del servicio recurrente y que se materializará con la firma de un documento de aceptación del servicio.

A pesar de que el licitador tendrá que detallar en su oferta técnica el calendario, la organización, la metodología, las actividades, las acciones a emprender y los planes de contingencia para que la etapa de transición no afecte al usuario, a continuación se relacionan las condiciones sobre el modelo de transición.

Los roles y las responsabilidades de cada parte en cada una de las fases son:

Etapa	Responsabilidad de el adjudicatario saliente	Responsabilidad de el adjudicatario entrante
Antes de la firma del nuevo contrato	Facturación de los servicios  Tiene la responsabilidad de la prestación del servicio y el cumplimiento de los MÁS BIEN actuales	
Fase de Transición: Captura de conocimiento	Facturación de los servicios.  Tiene la responsabilidad de la prestación del servicio y del cumplimiento de los MÁS BIEN actuales.  Facilita la atención, la colaboración y la información necesaria para realizar una correcta transferencia de conocimiento y tecnológica que permitirá al nuevo adjudicatario hacerse cargo del servicio.	Tiene que proponer la temporalidad de la fase de captura del conocimiento y de la asunción del servicio.  Tiene que proponer los procesos seguidos para garantizar esta transferencia.
Fase de Transición: Assumpció del servicio por el adjudicatario entrante	Facturación de los servicios.  Sigue teniendo la responsabilidad de la prestación del servicio y el cumplimiento de los ANTES.	Tendrá que seguir las etapas y la temporalidad de la fase de transición ofrecida a la propuesta.  Hará seguimiento de la prestación del servicio en modo shadow.

	Apoyará al adjudicatario entrante para que éste asuma el servicio sin interrupción.	
Finalizada la fase de transición: Fase de ejecución		Facturación mensual del servicio recurrente (según año de servicio) y los servicios adicionales que se encarguen.  Tiene la responsabilidad de la prestación del servicio y el cumplimiento de los MÁS BIEN y facturará mensualmente el servicio recurrente.

Hay que tener en cuenta que puede haber aplicaciones que actualmente no estén en el marco de ningún servicio continuado y por tanto, la fase de captura de conocimiento tenga que realizarse mediante tareas de auto-aprendizaje.

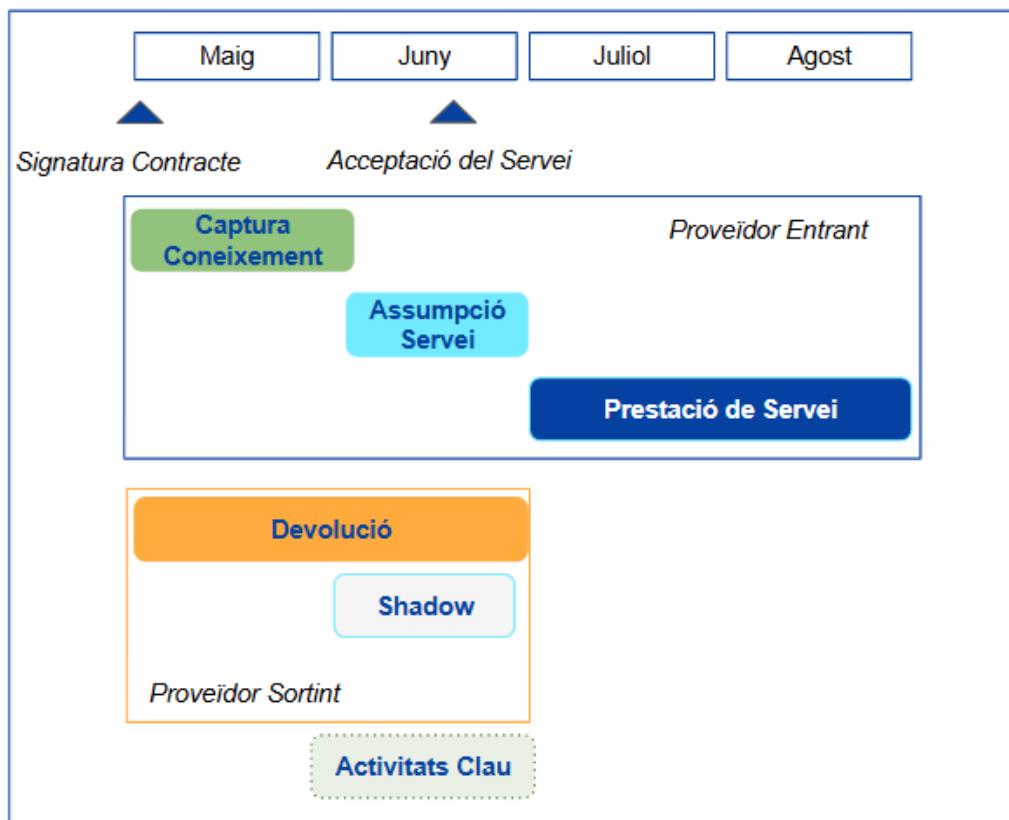
El adjudicatario entrante, el adjudicatario saliente y la UOC acordarán la finalización de la fase de Transición del Servicio mediante la firma de un documento de aceptación del servicio:

Último año de contrato:



### Consideraciones a la fase de Transición: Actividades Clave

El calendario de la fase de transición de los servicios es la que se muestra en el siguiente gráfico:



No obstante, hay que tener en cuenta que el calendario de actividades clave de la UOC es el siguiente:

Gener	<b>Evaluación, matrícula</b>
Febrero	<b>Evaluación, matrícula, inicio de semestre</b>
Marzo	<b>Inicio de semestre (PG)</b>
Abril	Ninguna actividad clave
Mayo	Ninguna actividad clave
Junio	<b>Matrícula, evaluación</b>
Julio	<b>Matrícula, evaluación</b>
Agost	Ninguna actividad clave
Septiembre	<b>Matrícula, Inicio de semestre</b>
Octubre	Inicio de semestre
Noviembre	Ninguna actividad clave
Diciembre	Ninguna actividad clave

y que esta actividad, durante los meses de enero y febrero, genera altos volúmenes de peticiones sobre el servicio recurrentes con prioridades Crítica y Alta, así como la activación de servicios de atención especial y guardias. Tanto la resolución de estas peticiones como la prestación de los servicios adicionales con pericia, son actividades altamente críticas y necesarias por la continuidad de los servicios que dan respuesta a la evaluación, la matrícula y el inicio de semestre.

A continuación se denominan las principales actividades del periodo:

- Adecuaciones previas, test previos y pruebas de estrés sobre a los sistemas implicados en la evaluación y cierre de actas
- Atención especial (priorización, monitorització y seguimiento) de los sistemas implicados al proceso de evaluación y cierre de actas
- Guardias de exámenes
- Seguimiento de la campaña de Matrícula
- Seguimiento de los procesos previos al Inicio de Semestre
  - traspaso de estudiantes en las aulas
  - revisión de los impactos de los proyectos desarrollados entre periodos (releases)
  - plan docente y copia en el aula
  - recursos de aprendizaje
- Atención especial (priorización, monitorización y seguimiento) de los sistemas implicados en el inicio de semestre.

Todos los servicios actuales de mantenimiento de sistemas están altamente implicados en las actividades de estas actividades clave:

- Servicios de mantenimiento de desarrollos a medida: por las actividades de Evaluación, Cierre de actas, Inicio de semestre y Matrícula
- Servicios de mantenimiento de tecnologías por el aprendizaje y la comunidad: por las actividades de Evaluación, e inicio de semestre
- Servicio de mantenimiento de portales: por las actividades de Inicio de semestre
- Servicios de mantenimiento del CRM: por las actividades de matrícula

Es por este hecho, que se estima oportuno disponer de un periodo de transición superior que garantice la prestación del servicio por parte del proveedor saliente durante la mayor parte del periodo crítico de la uoc y el inicio de la prestación del servicio del proveedor entrante al

finalizar el inicio de semestre de titulaciones homologadas. Durante el inicio de semestre de posgrado (Marzo) el proveedor entrante prestará el servicio con el apoyo del proveedor saliente.

## 5.2 Fase regular del servicio recurrente

La fase regular del servicio empezará justo al finalizar la fase de transición y durará hasta la finalización del contrato de servicio. En esta fase el adjudicatario será responsable de la correcta ejecución del servicio en los términos establecidos en este pliego.

## 5.3 Fase de devolución del servicio

En caso de cese o finalización del contrato, el proveedor estará obligado a volver el control de los servicios objeto del contrato, teniendo que realizar en paralelo los trabajos de devolución con los de prestación del servicio, sin coste adicional por la UOC. En caso de que se tenga que ejecutar la devolución, esta se llevará a cabo durante los dos últimos meses del contrato.

### 5.3.1 Modelo de devolución

El modelo de devolución tendrá que cumplir, como mínimo, los siguientes requerimientos:

- La fase de devolución tiene una duración temporal estimada de 2 meses para completar la correcta transferencia de conocimientos y la transferencia tecnológica por tipo de servicio.
- Se aplicarán las mismas condiciones del punto 5.1.1 Transición del servicio de este pliego, aplicando al adjudicatario de este contrato las condiciones de adjudicatario saliente.
- Durante la fase de devolución el adjudicatario saliente se compromete a dedicar como mínimo el apoyo de un Técnico a dedicación completa para facilitar al adjudicatario entrando la fase de ejecución de la transición.
- En caso de que la devolución del servicio no sea satisfactoria a la finalización del segundo mes, es decir, que no se cumplan los requerimientos de este pliego más los comprometidos a la oferta del adjudicatario, podrá ser requerido el apoyo del adjudicatario saliente durante los siguientes dos meses.

El licitador tendrá que detallar en su oferta el calendario, la organización, la metodología, las actividades, los recursos a los que se compromete para ejecutar la devolución, las acciones a emprender y los planes de contingencia para que la etapa de devolución no afecte al usuario.

## 6. Contenido y estructura de la oferta

### 0 Resumen ejecutivo

Resumen de los contenidos más significativos de la propuesta de servicios de mantenimiento y evolución de aplicaciones, destacando los aspectos más relevantes. El documento tendrá que contener los compromisos que propone adquirir el licitador de forma clara y concisa.

Este apartado se tendrá que resumir en un máximo de 5 páginas

### 1 Servicio Recurrente

#### 1.1 Descripción técnica del servicio

Descripción técnica del servicio en base a las prescripciones técnicas descritas al pliego.

Descripción detallada del modelo de prestación del servicio recurrente, sus órganos de gobierno, instrumentos de gestión e indicadores, así como los procesos de mejora continua, gestión de la obsolescencia, e incorporación de aplicaciones al mantenimiento

Este apartado se tendrá que resumir en un máximo de 10 páginas

#### 1.2 Equipo de trabajo

Se presentará el equipo de trabajo por la prestación del servicio recurrente, con una descripción detallada de los perfiles, sus tareas, y dedicación ofertada. También presentará evidencias de la experiencia, y conocimientos de los perfiles.

Este apartado se tendrá que resumir en un máximo de 5 páginas

EL ANEXO CON LAS EVIDENCIAS DE CADA PERFIL NO ESTÁ LIMITADO EN EXTENSIÓN

### 2 Servicios de desarrollo bajo demanda

#### 2.1 Descripción técnica por tipo de servicio

El licitador presentará la descripción técnica del servicio en base a las prescripciones técnicas descritas en el pliego. Hará una descripción detallada de los diferentes modelos de prestación de servicio de desarrollo bajo demanda , sus órganos de gobierno e instrumentos de gestión.

Este apartado se tendrá que resumir en un máximo de 10 páginas

## **2.2 Procedimientos de calidad**

El licitador presentará los procedimientos de calidad por los servicios de desarrollo bajo demanda en base a las prescripciones técnicas descritas en el pliego.

Este apartado se tendrá que resumir en un máximo de 10 páginas

## **2.3 Equipo de trabajo**

Se presentarán los perfiles que podrán ser requeridos por la UOC en los servicios de desarrollo a medida, en aquellos desarrollos que por su criticidad o volumen lo considere oportuno. Se presentará una descripción detallada de los perfiles y las evidencias de la experiencia y conocimientos.

Este apartado se tendrá que resumir en un máximo de 5 páginas

EL ANEXO CON LAS EVIDENCIAS DE CADA PERFIL NO ESTÁ LIMITADO EN EXTENSIÓN

# **3 Modelo de Servicio**

## **3.1 Descripción del modelo de servicio global**

El licitador presentará la descripción del modelo de servicio global en base a las prescripciones técnicas descritas en el pliego. Hará una descripción detallada del modelo de relación global, y la estructura de responsabilidades.

Este apartado se tendrá que resumir en un máximo de 10 páginas

## **3.2 Fases del Servicio**

El licitador presentará la descripción de las diferentes fases del servicio recurrente que cubrirán todo el periodo de duración del contrato en base a las prescripciones técnicas descritas en el pliego. Hará una descripción detallada de su temporalidad, las actividades, el equipo y dedicaciones previstas.

Este apartado se tendrá que resumir en un máximo de 10 páginas

#### **4 Arquitectura**

El licitador presentará la descripción de los mecanismos para garantizar la aplicación de los principios y normativa de arquitectura descritos al pliego

Este apartado se tendrá que resumir en un máximo de 5 páginas

#### **5. Plan de transformación**

Para valorar el nivel de comprensión del licitador de los retos que desde el punto de vista de los sistemas de información supone el modelo de gestión operativa de las TIC basado en programas, servicios digitales y su mejora continua, el licitador presentará un plan de transformación de los servicios de mantenimiento que maximice el logro de estos retos. En este sentido, tendrá que proponer potenciales escenarios de servicio en referencia a su configuración, al modelo de prestación y a la evolución de los sistemas de información.

Este apartado se tendrá que resumir en un máximo de 10 páginas

#### **6 Innovación y Valor añadido**

##### **6.1 Formación en tendencias tecnológicas**

El licitador podrá aportar una propuesta complementaria de servicios asociada a un plan de formación a los responsables del área de tecnología de la UOC, en temas técnicos relacionados con tendencias tecnológicas.

Este apartado se tendrá que resumir en un máximo de 2 páginas

##### **6.2 Pruebas de concepto**

El licitador podrá aportar una propuesta complementaria de servicios asociada al desarrollo de pruebas de concepto que permitan visualizar y explorar nuevas tecnologías.

Este apartado se tendrá que resumir en un máximo de 2 páginas

## Anexo - Sistemas del Lote A

A continuación se incorpora una relación de las principales aplicaciones objeto del lote A de este pliego que dan apoyo a los sistemas, procesos y actividades de Aprendizaje y

- Sistemas de Aprendizaje
- Sistemas de Investigación y Transferencia

Key	Summary	Descripción Corta	Tecnología	Servicio Digital
ADMINCV-2	ADMINCV - Administrador del Campus Virtual	Herramienta de administración del Campus Virtual. Permite la gestión de usuarios, dominios (aulas, asignaturas, etc), recursos (conversaciones, ficheros, etc), menús, procesos Batch y constantes por empresa (asignaciones por defecto, tipo de usuario, etc).	Java	TECNO-599 T5 Identity & Access Management
ANTICOP-12	ANTICOP - AntiCòpia (PACPlagi)	Queremos disponer de una herramienta que nos permita saber si la PEC que ha librado es un trabajo original o si ha copiado otros compañeros o de información de Internet.	C, Java - J2EE web, Java - J2EE webservices	APREN-236 Evaluación en el aula
ASPA-1	ASPA	ASPA herramienta para simular prácticas virtuales.	PHP	APREN-235 Prácticas virtuales
ATENEA-1	Atenea Academy	En torno a aprenentage LMS Canvas por los trabajadores UOC.	Html / Css, Javascript	NEGOCIO-44 0 Desarrollo del talento
AULACA-1	AULACA - Aula Centrada en la Actividad	Versión de aula centrada en la actividad.	Java	APREN-230 Aula virtual

AUTH-1	AUTH - Autenticació n de Campus (cgibin,cgilog )	Autentificador del usuario al Campus Virtual	Java	
BLOGA-30	BLOGA - Blogs Aula	Servicio de blogs en las aulas, grupos, salas, tutorías, etc. de campus.	PHP - LAMP	APREN-232 Colaboración en el aula
BUSTIACA-1	BUSTIACA - Buzón Compartido del Aula	Adaptación del WebMail para hacer uso en buzones compartidos del aula (típicamente foros, debates y tableros). El proyecto consiste al añadir al WebMail la funcionalidad necesaria por su uso en una mensajería compartida (permisos por roles, threading de mensajes, etc.).	Html / Css, Java	APREN-231 Comunicació n en el aula
CANVAS-1	CANVAS - Plataforma LMS Canvas		CANVAS Html / Css, Javascript	APREN-230 Aula virtual
CANVASSINC-1	Sincronizació n actividades y secciones	Herramienta para la sincronización de actividades y secciones entre el blueprint y los cursos asociados.	LowCode - Outsystems	APREN-231 Comunicació n en el aula
CERCAULA-1	CERCAULA - Buscador de aulas	Aplicación que permite buscar las aulas de una APPID concreta, y la muestra siempre y cuando estés asignado en el aula.	Otros, Html5, Javascript	APREN-230 Aula virtual
CGRE-1	CGRE - CampusGate way RISTRA webapp	Aplicación web que expone servicios OKI como recursos RISTRA.	Otros, Java	APREN-230 Aula virtual
CLASS-184	CLASS - Classroom (Contiene		Html / Css, Java	APREN-230 Aula virtual

Plan Docente)				
CLUST-45	CLUST - Cluster	Clúster de mensajería del Campus Virtual.	Java	APREN-231 Comunicació n en el aula
CODELAB-1	CODELAB	LA HERRAMIENTA CODELAB OFRECE, A LOS ESTUDIANTES DE ASIGNATURAS DE INTRODUCCIÓN EN LA PROGRAMACIÓN, UNA AULA DE LABORATORIO QUE SIMULE LA INTERACCIÓN QUE SE LLEVA A CABO EN UN LABORATORIO DE PROGRAMACIÓN PRESENCIAL	LowCode - Outsystems	APREN-234: Herramienta s específicas de aprendizaje
DESKA-1	DESKA - Servicio de descarga de materiales-so ftware	Aplicación que hace de puente entre un servidor de descarga de software y el campus.	Java - J2EE webservices	DOCENCIA-3 13 Gestión Recursos de aprendizaje
EASSIGN-1	EASSIGN - Espacio Asignatura	Espacio para el diseño y gobierno del contenido académico de las asignaturas, que tiene que permitir gestionar la información al margen del aula con docencia y publicar el contenido cuando corresponda y ente los canales de salida correspondientes.		APREN-228 Diseño y preparación de la asignatura
EDCOL-1	EDCOL - Edición colaborativa de materiales en HTML5, Chat en los materiales y anotaciones	Edición colaborativa de los materiales en HTML5 del aula.	Html5, Javascript, Nodo.js	APREN-228 Diseño y preparación de la asignatura

EDNA-1	EDNA - Edición Docente del Nido al aula	Edición Docente del Nido al aula (EDNA).	Java	APREN-228 Diseño y preparación de la asignatura
EPLADOCENT-1	EPLADOCENT - Nueva herramienta de edición del Plan docente	Nueva herramienta de edición del plan docente	LowCode - Outsystems	APREN-228 Diseño y preparación de la asignatura
ESTADLOGIN-1	Estadísticas de conexiones		PL/SQL	
EVM-1	EVM - Enterprise Vídeo Manager		Nodo.js	APREN-233 Vídeo en el aula
FUNDAE-40	FUNDAE	Aplicación en OutSystems para gestionar las matrículas bonificadas por la Fundación Fundae.	LowCode - Outsystems	COMCOM-2 55 B2B Colaboracion es Corporativas (COMO SD2)
GDD-123	GDD - Generación de documentos digitales		LowCode - Outsystems	DOCENCIA-3 21 Certificados, Acreditacion es y Titulación académica
GENERAMAT-1	GENERAMAT		Otros	

		-		DOCENCIA-3
		Herramienta de generación de materiales multiformat		12 Producción Recursos de aprendizaje
GOOGLE-4	GOOGLE - Integración de Google Apps en el Campus	Integración de las herramientas de Google Apps al campus de la UOC.	Java	
ICOMP-7	ICOMP - iComponents	Desarrollo de componentes accesibles para otras aplicaciones.	Java	APREN-230 Aula virtual
IRMAP-1	IRMAP - Mapa de Relaciones Institutionale s	Mapa interactivo de las relaciones de la UOC con otras universidades.	Html / Css, Ruby	NEGOCIO-44 9 Jurídico Universitario
LANGAPI-1	LANGAPI - APIO lingüística	Servicios web que hacen de interfaz de programación de aplicaciones para las herramientas de tratamiento lingüístico.	Java - J2EE web	APREN-231 Comunicació n en el aula
LTI-82	LTI - Learning Tools Interoperabil ity	LTI nos permite integrar dos aplicaciones simplemente compartiendo un consumer_key y un secreto. Este proyecto mantiene las librerías del campus y de las herramientas (en PHP y Java) que permiten su integración.	Java - J2EE web, PHP	APREN-230 Aula virtual
LTIMULTILINGUE -1	LTIMULTILIN GUE - LTI Multilingüe	Herramienta LTI que permite la configuración de más de un idioma a las aulas Canvas, como las funcionalidades de traducción y creación de contenido multilingüe.	CANVAS Html / Css, Javascript	APREN-231 Comunicació n en el aula

LTIPROXY-1	LTIPROXY		Java - J2EE web	APREN-230 Aula virtual
MOODLEPAF-1	MOODLEPAF - Moodle para las PAF	Moodle específico para las Pruebas de Evaluación Final (PAF).	LowCode - Outsystems	DOCENCIA-3 16 Operaciones académicas
MOODOS-1	MOODOS - Plataforma Moodle 2.X	Instalación, integración y utilización de la plataforma Moodle 2.4 al campus.	PHP - LAMP	APREN-236 Evaluación en el aula
MSTEAMS-1	MSTeams	Herramienta SaaS por el trabajo en equipo y la colaboración dentro del aula.	N/A	APREN-235 Prácticas virtuales
MYMAT-1	MYMAT - Mis materiales	Acceso indefinido a la versión más actualizada de los materiales didácticos de las asignaturas matriculadas.	Java	DOCENCIA-3 13 Gestión Recursos de aprendizaje
MYWAY-1	Myway	Servicio de generación de multiformat para cada uno de los materiales disponibles al repositorio <a href="http://setmenat.uoc.es/svn-materia">http://setmenat.uoc.es/svn-materia</a> ls/xml/	Java, XPL, XSLT	DOCENCIA-3 12 Producción Recursos de aprendizaje
NOTAMIN-1	Nota Mínima	Herramienta para definir la nota mínima en la evaluación final de asignaturas.	LowCode - Outsystems	APREN-236 Evaluación en el aula
OKI-60	OKI - Servicios OKI	Conjunto de piezas de software que implementan el estàndar OKI del MIT.	Java, Java - J2EE web, PHP	APREN-230 Aula virtual
OPENAPI-1	OPENAPI - APIO pública de Campus	APIO pública de Campus por los TFG de los Estudios de Informática.	Java	APREN-230 Aula virtual

OUTFORUMS-1	OUTFORUMS	Aplicación sobre la plataforma lowcode que sirve para poder gestionar las herramientas de comunicación de las aulas (sobre aulaca).	LowCode - Outsystems	APREN-231 Comunicació n en el aula
PDHUB-1	PDHUB - Pla docente agregador	Aggregación del contenido de los diferentes servicios para publicar en el Plan docente de GATO a Canvas.	LowCode - Outsystems	APREN-228 Diseño y preparación de la asignatura
PDLTI-1	PDLTI - Publicación del Plan docente vía LTI	Nueva aplicación para la publicación del Plan docente de GATO dentro de Canvas.	Java	APREN-228 Diseño y preparación de la asignatura
PERFIL-1	PERFIL - Mi perfil 2.0		Html / Css, Ruby	TRANSFORM ACIO-33 Contenidos académicos e interacción con el usuario
PHPWIRIS-1	PHPWIRIS - Editor Formulas Wiris	Aplicación PHP que permite incluir símbolos matemáticos en el buzón antiguo del Campus.	PHP	
PLACE-1	PLACE - PlacementTest	Prueba de nivel de idioma de un alumno.	PHP - LAMP	APREN-234 Herramientas específicas de aprendizaje
PLIN-486	PLIN - Plana Inicio	Página de inicio de campus compuesta por widgets.	Ruby	

PUET-1	PUET - Punto Entrada Único Tutores Entrada Único Tutores	Punto Entrada Único Tutores (PUET).	LowCode - Outsystems	TRANSFORM ACIO-34 Tutoría
PUBMAT-1	PUBMAT - Publicación materiales	Sistema que permite publicar los materiales de la UOC que edita Biblioteca-Producción.	Java	DOCENCIA-3 12 Producción Recursos de aprendizaje
RAC-128	RAC - Registro de evaluación continua	Registro de Evaluación Continuada. Gestión de las entregas y la evaluación continua.	Java	APREN-236 Evaluación en el aula
RALTI-1	RALTI - Publicador de accesos a los recursos de aprendizaje a Canvas vía LTI	Publicador de accesos a los recursos de aprendizaje a Canvas vía LTI.	Java	APREN-228 Diseño y preparación de la asignatura
RLAB-1	RLAB - Reserva de laboratorios de los EIMT	Aplicación para la reserva de laboratorios virtuales de los EIMT	Java, LowCode - Outsystems	APREN-235 Prácticas virtuales
SCORM-2	SCORM	Servidores de SCORM.	Java	APREN-228 Diseño y preparación de la asignatura
SERVTUT-1	SERVTUT - Servicio de Tutorías	Nueva aplicación del Servicio de Tutorías que sustituye las actuales aulas de tutorías de inicio EVA de Aulaca.	Lowcode - Outsystems	DOCENCIA-4 04 Tutoría
TANDM-3	Tandem	Herramienta de gestión y servicio de contenidos para ejercicios orales	PHP - LAMP	APREN-234 Herramienta

síncronos para estudiantes de lenguas.				
TEAMSLTI-1	TEAMSLTI - MS-Teams LTI	Integración tecnológica del MS-Teams y el en torno a aula Canvas	Java, LowCode - Outsystems	APREN-235 Prácticas virtuales
TEMATRES-2	Tematres	Aplicación web para la gestión de lenguajes documentales y otros.	PHP	APREN-234 Herramientas específicas de aprendizaje
VFLOW-1	Vídeo Flow	Edición e incorporación automática de la imagen gráfica y el logotipo corporativos al material audiovisual producido en la UOC.	Python	COMCOM-2 42 Gobernanza ecosistema digital
WUINTUOC-1	WEBM - Webmail	Espacio de interacción y mensajería instantánea para posar en el nuevo espacio de la IntraUOC.	Java	
DIGICAT-1	DIGICAT - Carné digital	Nueva aplicación que genera los carnés de la UOC en formato digital para hacer más accesible y sostenible el uso del carné de la UOC	LowCode - Outsystems	DOCENCIA-3 16 Operaciones académicas
TIMERHUB-1	TIMERHUB - Componente para relacionar PIOLIN con FUNDAEREP	Componente en Outsystems que permite relacionar las de matrículas de Piolin con estudiantes bonificados por Fundae con el repositorio de usuarios FUNDAEREP.	LowCode - Outsystems	COMCOM-2 55 B2B Colaboraciones Corporativas (COMO SD2)
FRAEMPRESA-1	FRAEMPRESA - Solicitud de	FRAEMPRESA - Solicitud de Factura a nombre de empresa	LowCode - Outsystems	NEGOCIO-43 4 Gestión de

	Factura a nombre de empresa	SaveCancel		cobros, pagos y facturación a clientes
GESTAULA-1	GESTAULA - Gestión de aulas de docencia y tutoría	Creación de un sistema para gestionar las aulas de docencia y tutoría del Campus, por solventar el riesgo actual de gestión, debido al uso de herramientas no gobernadas.	LowCode - Outsystems	DOCENCIA-3 14 Gestión acción docente
CSM-1	CSM- Campus Service Management	Aplicación destinada a cubrir las necesidades de gestión de campus.	LowCode - Outsystems	DOCENCIA-3 14 Gestión acción docente

La relación de sistemas de cada Lote es viva, por la propia gestión de la obsolescencia, y la ejecución de proyectos de transformación que proporcionan nuevos sistemas que se tendrán que incorporar a los mantenimientos.

También están incluidos en el ámbito de este pliego interfaces con sistemas mantenidos por otros servicios no objetos de este pliego.

Se contempla en este lote el apoyo en la administración de las Google Apps

Se detallan a continuación los diferentes elementos del entorno tecnológico actual de la FUOC, donde lo Campus Virtual es actualmente un entorno en constante evolución, el cual acumula arquitectura en proceso de migración, y otras en constante evolución.

- Sistemas Operativos: Windows, Gnu / Linux; MAC OSX; Unix.
- Bases de datos: Oráculo 11g; MySql; MongoDB; Cassandra
- Servidores/Servidores de Aplicaciones: Apache; Tomcat, JBOSS
- Lenguajes: Java; PL/SQL; PHP; Ruby; Html; Xml; Javascript; C++; JSP (MVC con Struts); Python; Objetivo-C

- Herramientas de desarrollo: Oráculo SQL Developer; Eclipso; Git; Subversion; Raíles; Nodo.js; Struts; Spring; Symfony; Phonegap; Django
- Plataformas: CANVAS

Para garantizar la calidad del servicio, la UOC requerirá al adjudicatario que el grupo técnico que trabaje en el servicio objeto de este pliego esté especializado y disponga de un alto nivel de calificación en las diferentes tecnologías implementadas a la UOC y, así mismo, conozca el detalle de la arquitectura y del entorno específico de la UOC.

Durante la ejecución del servicio, se pueden producir evoluciones o decisiones de arquitectura tecnológica de la FUOC que impliquen cambios al entorno tecnológico indicado y que el adjudicatario tendrá que asumir sin coste adicional teniendo en cuenta un plan de gestión del cambio consensuado.

## Anexo - Sistemas del Lote B

A continuación se incorpora una relación de las principales aplicaciones objeto del lote B de este pliego que apoyan a los sistemas, procesos y actividades relacionadas con:

- Sistemas de Gestión Académica y Docente
- Sistemas Gestión de Personas
- Sistemas Económicos y Financieros
- Sistemas Transversales

Key	Summary	Descripción Corta	Tecnología	Servicio Digital
ADN-1	ADN - Acces a los Datos Necesarios	Acceso a los Datos Necesarios	Form Server, PL/SQL	DOCENCIA-320 Expediente académico
ADSALESF-1	ADSALESF - Adaptación sistemas previos al Salesforce		Java - J2EE web	COMCOM-253 Gestión CRM B2C (COMO SD4 )
AEP-1	AEP -	Herramienta para gestionar el	Form	

	Evaluación de Estudios Previos (BART)	trámite de AEP (Evaluación de Estudios Previos).	Server, Java - J2EE web, Java - J2EE webservicios, PL/SQL, XML, XSLT	DOCENCIA-31 5 Convalidación y reconocimientos
AIUNEIX-1	AIUNEIX - Acuerdos de Interface UNE		Otros, Java, PL/SQL	TRANSFORM ACIO-32 Apoyo a SGIQ
AYUDA-1	AYUDA - Planificación gestión y enlace	Herramienta que permite el cálculo de las unidades docentes académicas que realiza cada profesor.	Java - J2EE web	DOCENCIA-31 4 Gestión acción docente
ASSIEARXIU-1	Asistente (eArchivo)	Aplicación de registro de documentos a eArxiu.	Java - J2EE web	NEGOCIO-43 6 Archivo, Gestión Documental y Registro
BOF-1	BOF - Banco de Oferta Formativa	Banco de Oferta Formativa.	Java - J2EE webservicios	TRANSFORM ACIO-30 Portafolio académico
CAAM-1	CAAM - Web de confirmación del acceso antes de la matrícula	Web de confirmació del acceso antes de la matrícula	Web	DOCENCIA-31 9: Acceso académico

CARNÉ-1	CARNÉ - Carnés de la Comunidad UOC	Gestión de Carnés.	PL/SQL	DOCENCIA-31 6 Operaciones académicas
CAU-1	CAU - Centro de Atención de Usuarios	Centro de Atención de Usuarios.	Form Server, Java - J2EE web	TRANSFORM ACIO-36 Apoyo al estudiante
CDG-1	CDG - Control de Gestión	Control de Gestión. Procesos de tipos ETL sobre los ingresos de las matrículas de estudiantes de la UOC para el área de negocio Discoverer del mismo nombre. Los datos económicos de COFROS-1 se amplían en estos procesos con datos de la docencia obtenidas de GATO-1 y PIOLIN-1 a través de la relación con las matrículas.	Form Server, PL/SQL	NEGOCIO-43 3 Gestión económica y financiera
CEXPEDWEB-1	Consulta Expediente	Consulta del expediente de Titulaciones Homologadas (GATO).	Java - J2EE web	DOCENCIA-32 0 Expediente académico
CLUB-1	Club UOC - Plataforma de contratación de servicios	Plataforma de pago que se utilizará para pagar la suscripción del proyecto Alumni Premium.	PL/SQL	COMCOM-24 1 Acción institucional
COCO-1	COCO - Contenidos y contratos (Autores)	Herramienta por la gestión de contenidos o materiales de las asignaturas que se imparten.	Java - J2EE web, Java - J2EE webservicios	DOCENCIA-31 3 Gestión Recursos de aprendizaje
COFROS-1	COFROS -	Gestión de los diferentes pagos de	Form	

	Gestión de Cobros	matrículas y otros tributos de los estudiantes.	Server, Java - J2EE web, PL/SQL	NEGOCIO-43 4 Gestión de cobros, pagos y facturación a clientes
COMESTADIS-1	Formularios de preguntas (COMUN ESTADIOS)		Form Server, PL/SQL	TRANSFORMACIO-36 Apoyo al estudiante
CONVENIOS-1	CONVENIOS - Convenios con Terceros (Plantillari)	Convenios con Terceros.	PL/SQL	NEGOCIO-44 8 Jurídico Contractual
CORO-1	Correspondencia	Gestión de Correspondencia	Legacy	
COLA-7	COLA - Gestor de colas de correo	Aplicación por el envío de emails y gestión de la cola	Java	TECNO-605 T5 Interoperabilidad
DAGA-1	DAGA	Despliegue de Aulas y Gestión de Asignaciones.	Cloud Native	DOCENCIA-31 7 Programas y planes de estudio
DEFENSATF-1	DEFENSATF - Defensa síncrona de los trabajos finales (TFs)	Solución para la defensa síncrona de los trabajos finales del programas de la UOC.	Angular, Java, Java - J2EE webservices, Python	DOCENCIA-33 1 Evaluación digital
DIETAS-1	DIETAS - Gestión de dietas por Colaboradores		Form Server, PL/SQL	NEGOCIO-43 9 Gestión de PDC

DIMAX-1	DIMAX - Biblioteca (Gestión de la colección digital)	Estantería virtual de recursos de las asignaturas.	Form Server, Java - J2EE web	DOCENCIA-31 3 Gestión Recursos de aprendizaje
DONACIONES-1	DONACIONES - Gestión de donaciones	Aplicación de gestión de campañas de donación (también conocida como CARPANTA).	Form Server, PL/SQL	
DUPLICADOS-1 9	DUPLICADOS - Gestión de duplicados	Aplicación para la gestión de duplicados.	Angular, Java, Java - J2EE web, Java - J2EE webservic es	TRANSFORM ACIO-32 Apoyo a SGIQ
EADMIN-8	eAdmin suite (IDM, eSign, eArchivo, Asistente)	Suite de la empresa TECSIDEL por la gestión de documentación y relación con la administración electrónica.	Java	NEGOCIO-43 6 Archivo, Gestión Documental y Registro
EDOCU-1	eDocument	eDocument es una aplicación que preten ser un registro de documentación que aporta el estudiante en diferentes trámites de la UOC. El primero de estos trámites sería la documentación a aportar en la solo•licitud de becas. La pieza se divide en un módulo de administración donde se paramatritzen los diferentes tipos de documentos a tratar , sus metadatos y los trámites que los relacionan y donde se gestionan las suele•licitudes (aceptación o denegación) registradas por el otro módulo de la pieza de eDocument	Java	DOCENCIA-32 1 Certificados, Acreditacione s y Titulación académica

que es el registro de las solicitudes de los estudiantes y que queda fuera del alcance de este proyecto y se desarrolla en el proyecto de becas.

EGOV-1	EGOV - Plataforma de interoperabilidad	Java - J2EE web, Java - J2EE webservices	NEGOCIO-43 5 Administració n Electrónica
ENCUESTAS-1	Encuestas (en php)	Encuestas en PHP.	PHP TRANSFORM ACIO-33 Contenidos académicos e interacción con el usuario
ENQVELLA-1	ENQVELLA - Encuestas vieja (en Forms y PLSQL)	Form Server, PL/SQL	TRANSFORM ACIO-33 Contenidos académicos e interacción con el usuario
ENVIASMS-1	ENVIASMS - Envío de SMS	Aplicación interna que gestiona las notificaciones que se realizan vía SMS.	Java - J2EE web TRANSFORM ACIO-33 Contenidos académicos e interacción con el usuario
ESIGNUM-1	E- Signum	Componente de arquitectura que	Java - J2EE

		unifica en un servicio operaciones comunes que sino tendrían que implementarse múltiples veces en diferentes aplicaciones.	web, Java - J2EE webservicios	NEGOCIO-43 5 Administración Electrónica
ESTRATEGIES-1	Estrategias de Evaluación	Definición y marcaje de las estrategias de evaluación para asegurar la acreditación de la identidad y la autoría de la evaluación continua y la evaluación final.	Herramientas a Medida	DOCENCIA-33 1: Evaluación digital
FITXAEST-1	FITXAEST - Ficha Estudiando	Redefinición de la ficha del estudiante en un formato más visual y con la incorporación de una gestión de alertas y permisos.	Java - J2EE web	TRANSFORMACIO-36 Apoyo al estudiante
HICE-1	HICE - Enlace gastadas	Aplicación pasarela de registros de cobros y pagos de COFROS a FUSION.	Form Server, Java, Java - J2EE web	NEGOCIO-43 4 Gestión de cobros, pagos y facturación a clientes
FSISWEB-3	F6 WEB		Form Server, PL/SQL, XML	Acción institucional
GAME-1	GAME - Gestión de los materiales y envíos		Form Server, Html / Css, PL/SQL, XML	DOCENCIA-31 1 Biblioteca
GATO-1	GATO - Gestión Académica Telemática	Gestión Académica Telemática.	C, Form Server, Java, Java - J2EE	DOCENCIA-31 4 Gestión acción

Sistemas de Gestión de la Investigación y el Aprendizaje				
Sistemas de Gestión de la Investigación y el Aprendizaje				
GEPAF-2				webservic es, Javascript, PL/SQL, XML, XSLT
GEPAF-2	GEPAF - Gestión y asignación a sedes de examen		Java - J2EE web, PL/SQL	DOCENCIA-31 6 Operaciones académicas
GESDOCWEB-1	Gestordocum- web		Java - J2EE web	DOCENCIA-31 9 Acceso académico
GIC-2	GIC - Gestión integral de los colaboradores		Java - J2EE web	NEGOCIO-43 9 Gestión de PDC
GIRUOC-1	GIRO-UOC	Aplicación web Portal investigadores	Java	RECIN-102 Gestión de la Investigación
GPRA-1	GPRA - Gestión de la producción de recursos de aprendizaje	Herramienta para la gestión de la producción de recursos de aprenenatge (GRPA).	Java	DOCENCIA-31 3 Gestión Recursos de aprendizaje
GRUGAPPSII-1	GRUGAPPSII - Sincronizador datos de personas Google Apps	Pieza batch que mantiene los datos profesionales de los usuarios de google apps, su ubicación en el árbol organizacional y su tipología	Java	
GUD-1	GUD - Gestor de Uploads y Downloads	El sistema Gestor de Uploads y Downloads (GUD) es el encargado de gestionar la subida (upload) y	Java - J2EE web, Java - J2EE	TECNO-605 T5 Interoperabili

		descarga (download) de ficheros a las máquinas de la UOC.	webservices	dad
GSM-1	GSM (Disfrutó Service Management)	Sistema que permite disponibilizar los datos de manera eficiente y transparente para los consumidores, sin importar en qué sistema o base de datos se encuentren almacenadas.	Herramientas a Medida	DOCENCIA-31 7 Programas y planes de estudio
GWEGOV-1	GatewayEgov - Registro de evidencia / Integración trámites con eAdministración	GatewayEgov aplicación con dos funcionalidades principales: por un lado gestiona un registro de evidencia de solicitudes de documentos UOC por parte de los estudiantes y por otro lado, además del registro de evidencia, integra solicitudes de documentos con eAdministración servicios AOC vía comunicación con la aplicación eGov.	Java - J2EE web, Java - J2EE webservices	NEGOCIO-43 5 Administración Electrónica
INCIDENCIAS-1	INCIDENCIAS	Aplicación que utiliza Ibermática por el control de sus tareas, consumo de horas y otras tareas suyas.	Form Server	
MAPALLOCS-1	MAPALLOCS - Mapa de puestos de trabajo	Aplicación para gestionar los puestos de trabajo	Legacy	
MASHUPS-1	MASHUPS - Mapas de Google	Aplicación que utiliza los servidores de Google Maps para crear mapas que se suban utilizar con el OpenCMS.	PHP	
MDPOET-1	Música de Poetas - MDPOET	Espacio virtual creado por Lletra, para recopilar música y literatura.	Html / Css, Html5, Java - J2EE web, Javascript,	COMCOM-24 3 B2B - Gestión de contenidos

			PL/SQL	Web
MULTIDIOMA-1	MULTIDIOMA -	Aplicación Forms para mantener datos multiidioma de diferentes aplicaciones de gestión.	Form Server	DOCENCIA-31 7 Programas y planes de estudio
	Mantenimiento o datos multiidioma			
NOMENCLAS-1	NOMENCLAS - Nomenclatura UOC	Aplicación de definición de nomenclaturas lingüísticas de la UOC.	Angular, Java	COMCOM-24 4 Servicio lingüístico
NOTIFICADOR-1	NOTIFICA - Notificador	Sistema para gestionar notificaciones masivas o individuales mediante plantillas.	Form Server, Java - J2EE webservices, PL/SQL, XML	TRANSFORM ACIO-37 Carrera profesional
NOUFSIS-1	NOUFSIS	Reescritura de GOZO formateo Web	Java - J2EE web, Visual Basic	Acción institucional
PACO-1	PACO - Plataforma de pagos y Contratación a colaboradores	Gestión de pagos y contratación a colaboradores.	Form Server, Html / Css, Java - J2EE webservices, Javascript, XML, XSLT	NEGOCIO-44 4 Relaciones laborales y compensación
PCRM-1	PCRM	Configuración de las campañas de Solicitud de Información y Acceso.	Html / Css, Javascript, PL/SQL, XML, XSLT	DOCENCIA-31 0 Gestión matrícula
PEP-1	PEP -	Asignación de encargos a personas	Form	

	Encàrrecs a colaboradores	que colaboran con la UOC para cada asignatura y aula ("consultores" o a nivel de programa ("tutor").	Server, Java, Java - J2EE web, Java - J2EE webservic es, PL/SQL	DOCENCIA-31 4 Gestión acción docente
PIOLIN-1	PIOLIN - Gestión de Formación No homologada	Matriculación y gestión académica de las iniciativas de formación continuada y oferta corta de la Universidad.	Form Server, Html / Css, Javascript, PL/SQL, XML, XSLT	DOCENCIA-31 0 Gestión matrícula
POP-1	POP - Gestión de objetivos personales	Gestión de objetivos personales	Form Server, Java, PL/SQL	
PORTASIGN-1	PORTASIGN - Puerta Firma	Trámite de solicitud de Certificados Académicos: firma electrónica y envío certificado del documento solicitado utilizando servicios de la plataforma eSignum.	Java, Java - J2EE web, Java - J2EE webservic es	NEGOCIO-43 6 Archivo, Gestión Documental y Registro
PUBENQ-1	PUBENQ - Web de publicación de encuestas del aula y asignatura	Web de publicación de encuestas del aula y asignatura.	Html5, Java	TRANSFORM ACIO-33 Contenidos académicos e interacción con el usuario
RAEP-1	RAEP (Reconocimie nto Académico Experiencia	Gestión del trámite de Reconocimiento Académico de la Experiencia Profesional.	Java - J2EE webservic es	DOCENCIA-31 5 Convalidación y

		Profesional)		reconocimien tos
ROBINSONS-1	ROBINSONS - Repository Robinsons	Gestión de Robinsons	Java	COMCOM-24 1 Acción institucional
SEC-1	SEC - Sistema de Enlace con Campus	SEC - Sistema de Enlace con Campus.	Form Server, PL/SQL	DOCENCIA-31 4 Gestión acción docente
SIGNADOR-1	SIGNADOR - Firma masiva de documentos	Firma masiva de documentos.	Form Server, Java - J2EE web, Java - J2EE webservic es	NEGOCIO-43 5 Administració n Electrónica
SISCUA-8	SISCUA - Sistema de COLA de conexión a Salesforce	Sistema de COLA de conexión con Salesforce.	Html5, Python	COMCOM-25 3 Gestión CRM B2C (COMO SD4 )
SIT-33	SIT - Sistema Títulos	Sistema generalista de configuración y consecución de títulos de todas las iniciativas docentes contempladas en UOC.	Java, PL/SQL	DOCENCIA-32 1 Certificados, Acreditacione s y Titulación académica
TANOS-1	TANOS	Aplicación por el Cierre de Actas y Notas	Cloud Native	DOCENCIA-31 4 Gestión acción docente

TARIFA-1	Tarificador Transversal	Aplicación que mujer respondida a todas las necesidades de tarificación de la UOC en cualquier de sus procesos de gestión académica ( Matrícula, Certificados, AEP, Traslado expediente, Título, RAEP, Mayores de 40/45, Pruebas nivel lenguas, Prueba acceso más de 25, Leglització títulos, Admisión mayores de 40, Lectura de Tesis, Admisión ( documentación) MU Psico. Ene. Sanit)	Javascript	DOCENCIA-31 0 Gestión matrícula
TERCIOS-1	TERCIOS - Gestión de terceros		TECNO-593 T5 Estrategia y gobierno de la tecnología y los datos	
TITULADOS-1	TITULADOS - Herramienta por la gestión de titulados de la UOC. Se utiliza desde la web de Alumni.		NEGOCIO-44 5 Gestión de los datos del Área de Personas	
TRAMACC-1	TRAMACC - Tramitador de acceso (control de documentación aportada)	Herramienta que contiene el proceso de tramitación de la documentación de acceso a los estudios de la UOC aportada por los estudiantes. Usa la configuración del trámite de acceso definida a la herramienta CONTRAM e incorpora toda la gestión de las notificaciones de las diferentes casuísticas.	Java - J2EE web	DOCENCIA-31 9 Acceso académico
TRAMITSCFG-1	TRAMITSCFG - Configuración de Trámites	Configuración de Trámites Académicos (documentación, notificaciones, reglas ...).	Java	DOCENCIA-31 9 Acceso

	Académicos	Actualmente están configurados los trámites de acceso, RAEP y validación de datos de acceso.		académico
TREN-1	TREN - Transacciones de Registros Entre Nodos	Gestión del control de accesos de los usuarios a las aplicaciones de la UOC	C, Form Server, PL/SQL	TECNO-608 T5 IAM
TREX-1	T-REX	Desarrollo web por la gestión de los recursos externos por parte del PRA.	Html / Css, Java - J2EE web, Javascript, PL/SQL, XML	DOCENCIA-31 3 Gestión Recursos de aprendizaje
ULISES-1	ULISES - Envío de correos - Servicio AWS SUS	ULISES - Uno(ganso)liS(imple)E(mail)S(ervivce )	Javascript, Nodo.js	TECNO-604 T5 LowCode / NoCode
VALIDACIOTP-1	VALIDACIOTP - Gestión de la identidad en titulaciones propias	Motor de comprobación y envío de recordatorios para los alumnos de TPP que no han validado su identidad.	Angular, Java	DOCENCIA-31 9 Acceso académico
VALIDMAT-1	Validación de datos de acceso y matrícula	Validación datos de acceso y validación datos de nacionalidad y residencia.	Java - J2EE web	DOCENCIA-31 9 Acceso académico
TSM-1	TSM - Tutorias Service Management	La aplicación TSM es la encargada de realizar la gestión de las aulas de Tutorías.	LowCode - Outsystems	TRANSFORMACIO-34 Tutoría
RECLAMTELCOB-1	Gestión de reclamaciones telefónicas de cobros	Aplicación por la Gestión de reclamaciones telefónicas de cobros.	LowCode - Outsystems	NEGOCIO-43 4 Gestión de cobros, pagos

y facturación  
a clientes

BISIS-1	BISIS (Business Intelligence SEIS)	Business Intelligence - SEIS (Student Information System)	Azure DataFactor y + PBI	
DADO-1	Datos académicos UOC (fecha warehouse)	Datawarehouse de datos académicos. Se trata de todos los procesos de extracción y tratamiento (ETL) de la información. Se relaciona con Discoverer.	Oráculo Discoverer	TRANSFORM ACIO-32 Apoyo a SGIQ
QDC-1	QDC - Cuadro de mando Dirección	Proyecto para estabilizar las sincronizaciones de datos del QdC de Tableau online con nuestras BBDDs.	Tableau	NEGOCIO-44 7 Gestión de datos de Gerencia
DISCO-1	DISCO - Discoverer		Oráculo Discoverer	
DMMDP-1	DMMDP - Datamart MDP (Marge directo por programa)	Datamart MDP incluye la información sobre la contabilidad analítica de los programas UOC.	PL/SQL	NEGOCIO-44 7 Gestión de datos de Gerencia
DMUNEIX-1	DMUNEIX - Datamart Une	Datamart Une....	Talend	TRANSFORM ACIO-32 Apoyo a SGIQ

La relación de sistemas de cada Lote es viva, por la propia gestión de la obsolescencia, y la ejecución de proyectos de transformación que proporcionan nuevos sistemas que se tendrán que incorporar a los mantenimientos.

También están incluidos en el ámbito de este pliego interfaces con sistemas mantenidos por otros

servicios no objetos de este pliego.

A continuación se incorpora una relación de los principales sistemas a los que apoyan las aplicaciones correspondiente al Lote B:

- Diseño de programas (Acción Formativa)
  - o Gestión académica donde se definen los estudios/programas/planes y asignaturas, sus atributos docentes, su estructura y la relación entre ellos.
  - o Gestión de autores. Incluye gestión de encargos a autores, enlace con contratos de autores, plan de publicación y pago
- Gestión del acceso y matrícula
  - o Pre-matrícula, tutorización de matricula y Matricula. Incluye el sistema de tasas.
  - o Captación y enlace con las diferentes piezas de acceso o matrícula
- Actividad Docente
  - o Encargos a personal docente colaborador por parte de los profesores responsables de una asignatura.
  - o Selección, contratación y pago de colaboradores docentes
  - o Gestión del trámite de evaluación de estudios previos (convalidaciones)
  - o Gestión de las órdenes de remisión de materiales derivadas de la venta
  - o Traspaso de información de estudiantes/aulas y cursos al Sistema Campus para hacer la docencia
  - o Workflow por la gestión de trámites académicos
  - o Registro y gestión de la evaluación continua
  - o Gestión de solicitudes de prácticas a empresas.
- Fidelización (Alumni)
  - o Directorio de antiguos alumnos

- Servicio al estudiante
  - o Recogida y gestión de consultas, quejas e incidencias
  - o Ficha del estudiante / colaborador
- Gestión económica y financiera
  - o Gestión de los cobros
  - o Workflow de validación de facturas
  - o Control de gestión
- Sistemas y aplicaciones transversales
  - o Gestión de datos personales
  - o Notificación y envío de mails personalizados
  - o Acceso a aplicaciones y directorio de personas
  - o Generación de encuestas y recopilación de la información
  - o Gestión centralizada de uploads y downloads

## Tecnologías

Se detallan a continuación los diferentes elementos del entorno tecnológico actual de la UOC:

- Sistemas Operativos: Windows; Gnu / Linux; MAC OSX; Unix.
- Bases de datos : Oráculo 9i; Oráculo 10g; MySql; Ms-Access, MondoDB, Posgress
- Servidores de Aplicaciones: OC4J, Tomcat, JBOSS
- Lenguajes: Java; PL/SQL; Oráculo Forms; PHP; Html; Xml; Javascript; Pro\*C; C; Visual C++; Ms-Basic Access; JSP (MVC con Struts);
- Enes de desarrollo: PL/SQL Developer; Oráculo Developer suite; Oráculo Discoverer; Eclips
- Oracle Discoverer Herramienta de explotación corporativa de la UOC
- PI/SQL QueryReporter Herramienta utilizada para generar informes
- Qlikview Herramienta utilizada por el equipo de marketing y que podrían necesitar de un espaldarazo desde Tecnología.
- Tableau

- Desktop Herramienta por la construcción de los dashboards y storetellings. se utiliza actualmente por perfiles tecnológicos.
- Online Herramienta donde están publicados los dashboards, cuadros de mando, storytellings... Y donde acceden los usuarios, la dirección de la UOC.
- R herramienta que se utiliza para analizar la información. Todavía no se ha extendido su uso, pero se prevé su necesidad dada su capacidad de análisis.
- Google Analytics

Para garantizar la calidad del servicio, la FUOC requerirá al adjudicatario que el grupo técnico que trabaje en el servicio objeto de este pliego esté especializado y disponga de un alto nivel de calificación en las diferentes tecnologías implementadas a la FUOC y, así mismo, conozca el detalle de la arquitectura y del entorno específico de la FUOC.

Durante la ejecución del servicio, se pueden producir evoluciones o decisiones de arquitectura tecnológica de la FUOC que impliquen cambios al entorno tecnológico indicado y que el adjudicatario tendrá que asumir sin coste adicional teniendo en cuenta un plan de gestión del cambio consensuado.

## Anexo - Sistemas del Lote C

A continuación se incorpora una relación de las principales aplicaciones objeto del lote C de este pliego que dan apoyo a los sistemas, procesos y actividades relacionadas con:

- Portales

Key	Summary	Descripción Corta	Tecnología	Servicio Digital
AIGENSAT-1	AIGENSAT	Agente de IA Generativa para los Servicios de Atención	Cloud Native	DOCENCIA-40 9 Apoyo al estudiante
BDBIBLIOM-1	Base de datos Bibliométrica		Java - J2EE web	RECIN-104 Ciencia Abierta y Transferencia

BIBWEB-1	Web de la Biblioteca de la UOC	web para el Servicio de Biblioteca (http://biblioteca.uoc.edu), destinado a toda la comunicado UOC, en el que se integran los diversos servicios, recursos y herramientas que ofrece la Biblioteca.	PHP-LAMP	
BLOGS-1	BLOGS - Plataforma de blogs institucional es WP	Creación del Proyecto de mantenimiento de la Plataforma de blogs WP, destinado a recibir las peticiones de alta, baja, incidental y evolutivo que se pidan sobre la nueva plataforma de blogs worpress desplegada al azure.	PHP	COMCOM-24 1 Acción institucional
CLOUDREDIRECT OR-1	CLOUDREDIRECTOR		Nodo.js	COMCOM-24 3 B2B - Gestión de contenidos Web
CMSBIBLIO-1	CMSBIBLIO - Web de la biblioteca sobre OpenCMS		Java - J2EE web, React OpenCMS	DOCENCIA-31 1 Biblioteca
CMSDIVULGA-1	CMSDIVULGA - Opencms Revistas Divulgación		Java - J2EE web OpenCMS	RECIN-104 Ciencia Abierta y Transferencia
CMSESTUDI-1	CMSESTUDI - OpenCMS Estudiando	Gestor de contenidos basado en el OpenCMS, orientado a la publicación de información hacia los estudiantes dentro del Campus.	Java - J2EE web OpenCMS	TRANSFORMACIO-33 Contenidos académicos e interacción

Sistemas de Gestión de Contenidos y Portales				
				con el usuario
CMSGRUPRO-1	Grupos y Proyectos - OPENCMS	OpenCms de Grupos y Proyectos	Java - J2EE web OpenCMS	RECIN-104 Ciencia Abierta y Transferencia
CMSPORTAL-1	CMSPORTAL - Gestor de Contenidos de Portal OPENCMS	Gestión de contenidos del portal y de numerosos subportales de la UOC.	Java - J2EE web OpenCMS	COMCOM-24 2 Gobernanza ecosistema digital
CMSRECERCA-1	OpenCMS Investigación	OpenCms del Área de Investigación	Java - J2EE web OpenCMS	
DXP-1	DXP - Plato. Experiencia Digital - Adobe Experience Cloud (AEC)	Espacio que recoge todos los elementos vinculados a la DXP.	Java	COMCOM-28 9 Gestión de Contenidos Portal Institucional, excepto Oferta Formativa
EGRAD-1	EGRAD - Herramienta de Graduación	Herramienta de graduación.	Java - J2EE web	COMCOM-24 5 Gestión de actividad y protocolo
FOLIO-1	FOLIO	Folio es una herramienta de portafolio orientado al estudiante y la publicación y contenido multimedia.	PHP	APREN-234 Herramientas específicas de aprendizaje
FPUOCX-1	FPUOCX - Portal de	Portal de formación profesional de Fundación Jesuites	Azure	

formación  
profesional  
de  
Fundación  
Jesuites

GEF-1	GEF - Guía Estilo Frontend	Se trata de unificar el diseño, el comportamiento y las herramientas de desarrollo web para toda la comunidad de desarrolladores FUOC.	Html / Css	COMCOM-28 9 Gestión de Contenidos Portal Institucional, excepto Oferta Formativa
GPS-1	GPS	Aplicación por el servicio GPS	Herramientas a medida	DOCENCIA-40 8 Carrera profesional
INTRAOSRT-1	Intranet OSRT - Liferay	Intranet de la OSRT de Investigación	HTML - CSS	
INTRAUOC-1	IntraUOC - LIFERAY	Liferay de la intranet de IntraUOC.	Java - J2EE web Liferay	COMCOM-24 1 Acción institucional
NPLIN-1	NPLIN - Nueva Plana de Inicio	Desarrollo de nueva llanura de inicio del campus.	Angular, Java	TRANSFORM ACIO-33 Contenidos académicos e interacción con el usuario
ONETRUST-1	ONETRUST - OneTrust	Aplicación para la gestión de la RGPD y las Cookies.	OneTrust (Gestión	NEGOCIO-45

SaaS (RGPD)		de cookies)	O Protección de Datos
PLATACON-1	PLATACON - Plataforma de contenido	Plataforma de contenidos que gestionará la creación y generación de las webs de Ficha PDI y WEB R&I.	Angular, Html / Css, Java, Nodo.js, Python, Terraform
PVV4ALLCONFIG -1	PVV4ALLCON FIG Configuració n Pruebas Virtuales	Aplicación por la configuración de parámetros de las aplicaciones relacionadas con las pruebas virtuales.	Python, Terraform
PVV4ALLPAF-1	PVV4ALLPAF - Pruebas de Evaluación Final	Aplicación por el acceso a las pruebas finales, acceso al chat, apertura de VNEXT y descarga de enunciados y subida de respuestas.	Angular, Go, Terraform
PVV4ALLPDF-1	PVV4ALLPDF - Conversor de PDF	Solución de conversión a PDF de los examens de la UOC.	Python, Terraform
PVV4ALLPLAGI-1	PVV4ALLPLA GI - Revisión Plagio	Solución de revisión del plagio de los exámenes presentados por los estudiantes.	Python, Terraform
PVV4ALLWR-1	PVV4ALLWR - Waiting Room por las pruebas virtuales		Angular
PVV4ALLXAT-1	PVV4ALLXAT - Chat de comunicació n por las	Chat avisos pruebas virtuales.	Angular, Nodo.js, Terraform

pruebas virtuales					
RESUAPREN-1	RESUAPREN	Aplicación RESUAPREN para gestionar los resultados de aprendizaje de la oferta UOC (programas y asignaturas) y para definir estructuras como microcredenciales con el objetivo de relacionarlas con los datos del mercado laboral	Cloud NativeHtm l / Css	Ficha Servicio Digital - DOCENCIA-40 8 Carrera profesional	
SATPORTAL-1	SATPORTAL - Servicio Atención Portal		Java - J2EE web	TRANSFORM ACIO-36 Apoyo al estudiante	
SNPUSH-1	Sistema de Notificaciones Push	Gestor de notificaciones push basado en Amazon SNS	Java - J2EE webservic es		
SYMPOFRONT-1	Plantillas html Symposium	Plantillas html configuradas a Symposium	Html / Css	COMCOM-24 5 Gestión de actividad y protocolo	
VNEXT-1	VNEXT- Aplicación de reconocimiento de identidad del estudiante	Solución tecnológica que permite asegurar la identidad del estudiante.	Angular, Html / Css, Python, Terraform	DOCENCIA-33 1 Evaluación digital	
WCASTELLS-1	Web de Manuel Castells		Html	RECIN-104 Ciencia Abierta y Transferencia	

WESTATIC-1	WESTATIC - Webs Estáticas	Modificaciones en sites nuevos de portal como la web de los 30 años.	Terraform	COMCOM-28 9 Gestión de Contenidos Portal Institucional, excepto Oferta Formativa
WIDGETS-499	WIDGETS - Widgets de la llanura de inicio	Widgets de información relevante a la llanura de inicio de campus.	Ruby	APREN-230 Aula virtual
WIDGETSERV-1	WIDGETSERV - Widget Server	Widgets de GIRO	Java	
WPAZURE-1	WPAZURE - WordPress Azure	Proyecto para gestionar los Wordpress que se tienen que alojar a Azure.	Wordpress	DOCENCIA-31 2 Producción Recursos de aprendizaje
WPSTATIC-1	WPSTATIC - Plataforma Wordpress con generación de sites estáticos	Plataforma de recursos de aprendizaje en Wordpress que genera sites estáticos	Wordpress	DOCENCIA-31 2 Producción Recursos de aprendizaje
WSGATEWAY-1	WSGATEWAY - Pasarela de Consumo WS	Pasarela web que permite a las aplicaciones de Campus / Portal acceder en los web services del entorno a Gestión.	Java	TRANSFORMACIO-33 Contenidos académicos e interacción

con el  
usuario

La relación de sistemas de cada Lote es viva, por la propia gestión de la obsolescencia, y la ejecución de proyectos de transformación que proporcionan nuevos sistemas que se tendrán que incorporar a los mantenimientos.

También están incluidos en el ámbito de este pliego interfaces con sistemas mantenidos por otros servicios no objetos de este pliego.

Por sus particularidades, a continuación se describen las tareas que se espera que realice el proveedor en relación al mantenimiento y evolución del GEF:

- Mantenimiento del framework de frente end de la UOC (GEF) (<http://guia-frontend.uoc.edu>), compuesto por patrones HTML, CSS y Javascript reutilizables.

En función de la naturaleza de la actuación:

Mayor versión

- Cambios que afecten al layout de GEF con afectación a la estructura del framework y que tendrán que ser valorados por el equipo de GEF keeper (responsable/s UOC de la GEF) por el desarrollo de una nueva versión del framework que quedará en el backlog de posibles cambios y mejoras futuras.
- Cambios que impliquen refactorización de procesos y tecnología de mejoras de compilación, task runners, cambios en la arquitectura de archivos, etc
- Esta versión también puede incluir cambios MINOR y PATCH versiones.

Minor versión

- Cambios que afectan a componentes básicos y de interfaz de GEF con afectación a la estructura HTML, y creación de nuevas reglas de CSS e implementación de JS
- Variaciones de componentes existentes y creación de nuevas.
- Inclusión de nueva fuente iconográfica y/o tipográfica, elementos de formulario y colores
- Nuevas funcionalidades de JS o refactorización de las existentes
- Esta versión también puede incluir cambios PATCH versión.

Patch versión

- Cambios que afectan a componentes básico y de interfaz de GEF que impliquen modificaciones, bug fijas o correcciones de CSS, sin afectación a la estructura HTML
- Bug fijas de Javascript
- Ampliación de nuevos iconos y sustitución de assets existentes que no impliquen cambios significativos respecto al CSS, HTML o JS.
- Peticiones con solución técnica que tengan un impacto transparente por un proceso de QA

## Anejo - Sistemas del Lote D

A continuación se incorpora una relación de las principales aplicaciones objeto del lote B de este pliego que dónen apoyo a los sistemas, procesos y actividades relacionadas con:

- Salesforce CRM sobre EDA (Education Fecha Architecture)
  - Salesforce Marketing Cloud
  - Salesforce Sales Cloud
  - Salesforce Service Cloud

<i>Key</i>	<i>Summary</i>	<i>Descripción Corta</i>	<i>Tecnología</i>	<i>Servicio Digital</i>
CRMSIS-1	CRMSIS - Salesforce SEIS-CRM	CRM para la gestión de las personas, oportunidades y catálogo comercial de la UOC (nace para el proyecto SEIS). Este será el sistema que permanecerá en el tiempo y se desactivará el que actualmente mujer servicio a QBIS.	Salesforce	COMCOM-25 3 Gestión CRM B2C (COMO SD4 )
MCLOUD-1	MCLOUD - Marketing Cloud		Salesforce	COMCOM-25 3 Gestión CRM B2C (COMO SD4 )

La relación de sistemas de cada Lote es viva, por la propia gestión de la obsolescencia, y la ejecución de proyectos de transformación que proporcionan nuevos sistemas que se tendrán que incorporar a los mantenimientos.

También están incluidos en el ámbito de este pliego interfaces con sistemas mantenidos por otros servicios no objetos de este pliego.

Para garantizar la calidad del servicio, la FUOC requerirá al adjudicatario que el grupo técnico que trabaje en el servicio objeto de este pliego esté especializado y disponga de un alto nivel de calificación en las diferentes tecnologías implementadas a la FUOC y, así mismo, conozca el detalle de la arquitectura y del entorno específico de la FUOC.

Durante la ejecución del servicio, se pueden producir evoluciones o decisiones de arquitectura tecnológica de la FUOC que impliquen cambios al entorno tecnológico indicado y que el adjudicatario tendrá que asumir sin coste adicional teniendo en cuenta un plan de gestión del cambio consensuado.

## Anexo - Sistemas del Lote E

A continuación se incorpora una relación de las principales aplicaciones objeto del lote B de este pliego que apoyan a los sistemas, procesos y actividades relacionadas con:

- JIRA Software Fecha Center (Atlassian)
- JIRA Service Management Fecha Center (Atlassian)
- Confluence Fecha Center (Atlassian)
- SonarQube
- XRAY (pluggin de JIRA)

Key	Summary	Descripción Corta	Tecnología	Servicio Digital
CONFLU-36	Confluence	El Confluence es un gestor documental (wiki + blog) donde se deposita la documentación de los	Atlassian	TECNO-581 T4 Herramienta

		proyectos, aplicaciones, Normativa y guías, entre otros.		s de apoyo a la gestión
CONFLUSD-1	CONFLUSD - Confluence Service Desk	Se trata de la instancia de Confluence que se usa para Jira Service Desk, que está en la red en abierto.	Atlassian	NEGOCIO-46 2 Sistemas de Apoyo de Gestión
JENKINSCI-1	JENKINS CI Integración Continua	Jenkins es un servidor Open Source para la Integración Continua.	Herramientas de Mercado	TECNO-626 T7 Aprovisionamiento y despliegue
JIRA-884	JIRA	El Jira es una aplicación de ticketing de Atlassian que en la UOC usamos para gestionar todo los despliegues, mantenimientos de aplicaciones y gestión de proyectos, entre otros utilidades.	Atlassian	TECNO-581 T4 Herramientas de apoyo a la gestión
JIRASD-1	JIRASD - Jira Service Desk	Se trata de la instancia de Jira que se usa para los CAU's, que está en la red en abierto.	Atlassian	NEGOCIO-46 2 Sistemas de Apoyo de Gestión
SONAR-1	SonarQube	SonarQube (conocido anteriormente como Sonar) es una plataforma para evaluar código fuente. ES SOFTWARE LIBRE Y USA VARIAS HERRAMIENTAS DE ANÁLISIS ESTÁTICO DE CÓDIGO FUENTE COMO CHECKSTYLE, PMD O FINDBUGS PARA OBTENER MÉTRICAS QUE PUEDEN AYUDAR A MEJORAR LA CALIDAD DEL CÓDIGO DE UNA APLICACIÓN.	SonarSource	TECNO-634 T7 Calidad operativa

JIRA-XRAY

XRAY

Xray es una herramienta de gestión de pruebas para Jira, utilizada en el desarrollo de software para garantizar la calidad y la eficiencia de los proyectos. Se integra directamente en Jira, permitiendo a los equipos de desarrollo y QA crear, planificar y ejecutar pruebas dentro de sus proyectos

Atlassian

TECNO-634  
T7 Calidad  
operativa

# Anexo - Principios de Tecnología

A continuación se muestra la normativa de Calidad vigente en el momento de redacción del pliego. Esta Normativa se revisa 2 golpes al año a la Release Normativa. Hay que tener en cuenta que se tiene que cumplir la normativa vigente en cada momento.

## Principios estratégicos

Los nuevos sistemas y aplicaciones se regirán por los principios de Business first, cloud first y cloud Native , mobile first (en el sentido de multidispositiu), Fecha-driven (el dato como activo más importante de la organización) y finalmente el principio de aportación de Valor.

### Principio de Business First

La tecnología desplegada en la UOC tiene que responder a las necesidades y los objetivos de negocio de la institución en cada momento. Por lo tanto, tiene que ser flexible y adaptable, para ajustarse de manera gradual a los cambios del negocio de la institución o en todo caso, ser fácilmente reemplazable por nueva tecnología que sí que se ajusten.

Por lo tanto, hay que tener en cuenta que:

- Los sistemas de información están en constante transformación.
- Cuando se despliega una nueva solución hay que tener en cuenta tanto los procesos de actualización que un sistema sufrirá a lo largo del tiempo como los procesos de sustitución del producto una vez finalizado su tiempo de vida.
- Así pues, muchos productos tienen una obsolescencia programada, que coincide con la finalización de su contrato o con la estrategia de mejora tecnológica de la institución (veáis ROI First y propuesta de valor).
- A veces, un producto no desaparece del ecosistema, pero una parte de su funcionalidad es asumida por otro sistema (strangler application).
- La transformación digital en un ecosistema complejo, solo es posible bajo un buen gobierno. La mejora de los procesos de gobierno de la tecnología y de los datos es un factor clave que hace falta no descuidar nunca.

### Principio de Cloud first y cloud Native

- Despliegue de aplicaciones y sistemas a la nube de manera preferente y pensando en las propiedades y los aventajos de la nube.
- La computación a la nube convierte el hardware en software. EL USO DE COMPONENTES SOFT

EN VEZ DE HARD, HACE QUE LAS PROPIEDADES DEL SOFTWARE SEAN APLICABLES A LAS INFRAESTRUCTURAS TECNOLÓGICAS DE FORMA QUE ES POSIBLE ADMINISTRARLAS Y AUTOMATIZARLAS.

- Cloud native quiere decir contratar o desarrollar una aplicación o un sistema pensante desde el inicio en estas propiedades y ventajas del cloud.
- Los proveedores de computación a la nube permiten gestionar el hardware como servicio (ved SaaS), de forma que muchas de estas propiedades del software aplicadas a las infraestructuras están disponibles como servicio (usar cloud hoy es usar un servicio SaaS).

La UOC prioriza la elección de soluciones SaaS (Software as a Service):

- El SaaS es un modelo de distribución del software donde la lógica (código en ejecución) y los datos las gestiona un (o más de un) proveedor que ofrece la funcionalidad del software a través de la nube.
- El SaaS normalmente se consume bajo un modelo de 1 (el software) a N (los clientes), en cualquier momento y con una política de pago por uso o de suscripción basada en métricas de consumo.
- El SaaS permite modelos de despliegue de nuevas funcionalidades de manera automática y continuada (versionless), de forma que el servicio contratado no se ciñe a unas funcionalidades fijas sino variables a lo largo del tiempo. Por este motivo, la contratación del SaaS está siempre ligada a la capacidad del proveedor para hacer avanzar y evolucionar el producto contratado (incluida la solvencia del proveedor para ofrecer el servicio durante todo el contrato y mitigar así el riesgo de lock-in).

Para poder maximizar el uso de SaaS, ninguno pensar la solución aplicando el Principio de Market fit:

- Adecuación de un producto tecnológico a las necesidades del mercado.
- Cuando se desarrolla un producto a medida, es habitual pedir los requisitos y las necesidades a los futuros usuarios de este. Este proceso, a menudo no es aplicable cuando se contrata un producto de mercado.
- Para seleccionar correctamente un producto de mercado, es imprescindible analizar primero nuestros procesos para adecuarlos a la manera de operar estándar o más utilizada en la mayoría de organizaciones de nuestro sector.
- Las metodologías Lean ofrecen un buen marco de referencia sobre como afrontar esta adecuación a las necesidades del mercado.

Teniendo en cuenta todos estos factores, la receta a utilizar es:

1. Si existe un SaaS, utilízalo
2. Si no existe un SaaS, probablemente en el futuro, ya sea porque has podido aplicar Market fit o porque el mercado ha evolucionado, seguramente existirá. Mientras tanto, puedes cubrir este GAP funcional desarrollando una aplicación en LowCode o Cloud Native.

- Las herramientas de desarrollo rápido o Low Code, permiten crear aplicaciones y sistemas tecnológicos de manera visual y rápida, reduciendo la complejidad asociada al desarrollo de software.
  - Cuando un producto de mercado no cubre todas las prestaciones esperadas, se recomienda desarrollar este gap de manera rápida y dentro de la fase de implantación del producto mediante herramientas Low Code.
  - Las herramientas Low Code, también permiten la experimentación y el prototipatge para la realización de pilotos o pruebas de concepto.
3. Si no existe, eres único: construye
- siempre, reduce al mínimo las líneas de código, puesto que el código de hoy será el legacy de mañana.
4. Este desarrollo (Low Code o Cloud Native) y las integraciones que con otras soluciones que requiera, se harán siempre sobre servicios administrados (esto es, operación incluida en el servicio) de forma que te focalices en la lógica de negocio, en el código. Esto aplica tanto:
- a los building blocks de despliegue: PaaS, CaaS, FaaS, SaaS por encima de IaaS/VM
  - a los servicios existentes en los clouds de referencia: servicios de notificaciones, colas, CDN, storage, ..., por encima de soluciones desarrolladas a medida

### **Principio Fecha-Driven**

- La Interoperabilidad es la capacidad de dos o más sistemas, de compartir datos con el objetivo de facilitar el intercambio de información entre ellos.
- Los sistemas de información de una organización son cada vez más complejas y se comportan cada vez más como un ecosistema.
- En un ecosistema digital, los datos son aquello que se intercambian los sistemas y, por lo tanto, son uno de los principales drivers por la toma de decisiones a varios niveles (Fecha-driven). De hecho, conocer y gobernar estos datos, es esencial y necesario tanto para los procesos de analítica de datos como para la interoperabilidad entre aplicaciones y sistemas.
- Por cada nuevo sistema o aplicación a introducir a un ecosistema, es imprescindible valorar tanto los datos que este nuevo sistema provee y consume como los mecanismos tecnológicos que aporta para hacerlo (Integrabilidad).

### **Principio de aportación de Valor**

- Cuando se contrata, desarrolla o integra un producto, hay que calcular el retorno de la inversión (ROI).
- El cálculo del retorno de la inversión permite una planificación empresarial y financiera de los costes y beneficios tecnológicos.
- Cuando este producto no tiene unas funcionalidades fijas sino variables (SaaS) hay que gestionar

a lo largo del contrato su ROI, que dependerá de las capacidades de la institución para adoptar lo antes posible estas nuevas funcionalidades disponibles.

- Una propuesta de valor es una estrategia empresarial que maximiza la demanda a través de configurar de manera óptima la oferta. Así pues, en la gestión del portafolio (oferta), hay que considerar tanto la homogeneización (maximización del ROI) como el riesgo de dispersión tecnológica (que satisface la demanda pero sin optimizar la oferta).
- En definitiva estamos diciendo que hay que ponderar siempre tecnología vs negocio: la mejor tecnología no siempre es la mejor solución para la organización → se tienen que balancear aspectos tecnológicos, de gobierno, funcionales, organizativos y económicos.

## Principios sobre el diseño de aplicaciones

1. **Segregación de funciones/responsabilidades:** las aplicaciones tienen que estar estructuralmente divididas en bloques independientes por funcionalidades, procesos de negocio o servicios, para **evitar los monolitos** → patrón de microservicios:
  - a. Este principio es de **aplicación a todas las capas**: la división lógica de las funcionalidades también se tendría que corresponder a una división “física” en el despliegue → un servicio, una base de datos
2. La **presentación** (cliente/frontend) y lo **backend**/negocio estarán **desacoplados**.
3. **Orientación a servicios.** Las aplicaciones pueden ser consumidas externamente o bien tienen que integrarse con 3.os Los backend tienen que exponer su funcionalidad de negocio vía servicios para facilitarlo. Estos servicios serán, además, fácilmente integrables al **API Gateway**, para ser securitzats y gobernados de manera centralizada.
4. **Almacena a la memoria cae** todo aquello que sea posible. Para hacerlo, utiliza la tecnología que mejor se adapte tanto a cliente (html5 cache, localstorage , etc.) como servidor (Redis, Varnish, Memcache, caché personalizada, etc.)
5. **Desacopla** de la interfaz de usuario aquellos **procesos que consuman muchos recursos** (cpu, ram) o que **puedan ser ejecutados de manera asíncrona**: utiliza una **cola** que informe vía **events** a un nuevo **proceso** que se ejecutará **bajo demanda** y con una concurrencia X (preferiblemente para este caso de uso utilizaremos tecnologías serverless)
6. Ten presente siempre la **compatibilidad hacia atrás de tus servicios**: si expones una API RISTRA y actualizas tu servicio, que sea compatible con versiones anteriores para evitar actualizaciones innecesarias a tus consumidores y de este modo poder evolucionar el servicio libremente.

7. **Diseña** la aplicación o servicio teniendo presente los conceptos **de elasticidad** (en ninguna parte de para soportar los picos de carga), **de alta disponibilidad, de alta concurrencia y cero downtime**.
8. Piensa en la **portabilidad de las aplicaciones**, esto es, que se puedan mover con facilidad de un cloud privado a un cloud público, por ejemplo.
9. A la hora de diseñar tu sistema hay que incorporar **aspectos cualitativos al ciclo de vida**:
  - a. **Pruebas** para verificar la calidad del código, el apoyo de carga o requisitos no funcionales del sistema.
  - b. **Documentación detallada** del proyecto (descripción de arquitectura, documento funcional, manual de despliegue, manual de explotación, ...).
  - c. **Control de versiones**. El sistema en su conjunto tiene que ser tratado como un producto con sus versiones mayores, menores, etc
  - d. **Despliegue automatizado**, ejecución de pruebas automáticas que verifiquen la instalación e integración continua.
10. Si tienes que emplear datos de otro sistema, mejor consúmelas vía servicios. Los datos de aquel sistema tienen que ser “**el única fuente de la verdad**”
11. Delega las decisiones de autenticación, utilizando o bien el protocolo **SAML** o bien **OAuth** (si se trata de una SPA, aplicaciones móviles o bien de Apios). Las decisiones de autorización se tendrían que tomar en el ámbito de la aplicación, pero preferiblemente con los datos proporcionados por el sistema de autenticación de la UOC (perfiles, tipos de usuario...).

## Principios sobre la tecnología

1. **Utiliza la tecnología que mejor encaje al caso de uso**, si hace falta combinando tecnologías, puesto que en un mismo sistema pueden convivir diversas: por ejemplo, al dividir las aplicaciones en servicios, no todos tienen porque estar construidos en los mismos lenguajes o utilizar las mismas bases de datos.
2. Los **servicios** (backend) expondrán su negocio preferentemente **mediante RISTRA y en formato JSON**.
3. En el caso de aplicaciones web, **la presentación estará construida con tecnologías estáticas** (html5/javascript/css) y consumirá los servicios que le proporcione el backend.

4. Ante soluciones estándares utilizaremos preferentemente las tecnologías del **Hoja de ruta Tecnológico**. Este hecho no excluye que para nuevas soluciones se puedan utilizar otras tecnologías, que eventualmente pasarán a formar parte.
5. **Planifica y gestiona la obsolescencia tecnológica** (sistemas operativos, middlewares, frameworks de desarrollo, productos, ...): es una **inversión que reducirá riesgos**.
6. Cualquiera **nueve sistema creado, renovado o refactoritzat**, se tendrá que hacer con las versiones más modernas de middlewares, lenguajes, frameworks, módulos disponibles, y en caso de duda, utilizar versiones LTS, siempre alineado con la Hoja de ruta.

## Principios sobre el coste y mantenimiento de las soluciones

1. **Hay que tener presentes los costes de infraestructura y el modelo de licenciamiento** requeridos para poner en marcha una solución puesto que representan un coste recurrente.
2. Piensa en los **costes** y en la suya **optimización**:
  - a. **Monitoriza** tus servicios para identificar necesidades de ampliación o reducción de recursos y poder ajustar los costes en consecuencia
  - b. Arquitectura/diseña las **cargas de trabajo** con los costes en mente
3. A la hora de elegir entre utilizar uno **producto** (opensource o comercial) o hacer uno **desarrollo** a medida hay que hacer una evaluación del **coste vs. beneficio** de la opción elegida respecto a las otras.
4. Piensa en el **impacto de actualización** que pueda tener un cambio de sistema operativo, middleware o producto allá donde corre la aplicación: cuánto **menos acoplamiento** con el sistema de base y **utilizando estándares**, más sencilla será la actualización o la ampliación de funcionalidades de la aplicación.

# Anexo - Normativa de entregables

A continuación se muestra la normativa de Calidad vigente en el momento de redacción del pliego. Esta Normativa se revisa 2 golpes al año a la Release Normativa. Hay que tener en cuenta que se tiene que cumplir la normativa vigente en cada momento.

## Introducción

Definimos un entregable como un objeto resultado de la ejecución de un proyecto. Los entregables los podemos categorizar como entregables de gestión del proyecto que nos permiten hacer el seguimiento y control de proyecto, o entregables que acontecen resultados de la ejecución del proyecto en sí mismos, la obtención de los cuales, es el motivo por el cual ejecutamos el proyecto.

Por otro lado, se definen una serie de artefactos del proyecto que, a pesar de no ser entregables, requerimos que se mantengan actualizados a lo largo del proyecto para garantizar la correcta ejecución, seguimiento y control del proyecto.

## Alcance

La Normativa se aplica a todo el ciclo de vida de un proyecto.

## Propósito

EL OBJETIVO DE LA NORMATIVA DE ENTREGABLES ES DAR A LOS DIFERENTES PARTICIPANTES INVOLUCRADOS EN EL DESARROLLO, INTEGRACIÓN, EVOLUCIÓN O MANTENIMIENTO DE SOFTWARE DE LA UOC, LA NORMATIVA SOBRE LOS ENTREGABLES MÍNIMOS QUE HAY QUE ENTREGAR EN UN PROYECTO.

Se describen también aspectos generales normativos, plazos, nomenclatura y qué son los repositorios donde desar cada tipo de entregable.

Para cada uno de los entregables mínimos normativos, se describen las fases y los hitos asociados.

## Audiencia

Este Normativo está dirigido a todos aquellos proveedores del Área de Tecnología de la UOC que estén ofreciendo sus servicios actualmente, o lo quieran hacer en un futuro, así como para los miembros del equipo propio del Área de Tecnología UOC.

## Normativo

La Normativa de entregables la componen los siguientes ítems:

### Aspectos generales sobre entrega de documentación

- Todos los entregables tienen que ser proporcionados en el formato que defina la UOC y sobre las plataformas indicadas en cada momento
- Hay que usar las herramientas que la UOC pone a disposición para la gestión de los proyectos. A la descripción de cada entregable y de cada actividad se indica cuál es el repositorio que hay que usar
- Todos los entregables tienen que tener un formato que permita su edición, incluyendo los diagramas incrustados en los documentos. Si se libra un documento pdf por algún motivo particular, habrá que librar una copia en formato editable
- Los entregables no tienen que contener logotipos de ningún proveedor, solo los logos corporativos vigentes de la UOC
- La UOC es la propietaria de todos los entregables elaborados por los diferentes proveedores
- Tanto a la documentación de proyectos como la de aplicaciones, hay que mantener el registro de entregables actualizado así como el resto de artefactos descritos más adelante
- Hay que producir y mantener actualizada la documentación mínima que facilite la correcta comprensión del proyecto o el mantenimiento del servicio
- Los proveedores de proyectos mantendrán actualizada la documentación asociada a un proyecto hasta la finalización de la garantía del mismo. Del mismo modo, el servicio de mantenimiento de una aplicación o sistema, tendrá que mantener actualizados los entregables correspondientes hasta la devolución del servicio
- Los proveedores tendrán acceso a las aplicaciones corporativas siempre con su correo UOC

### Entregables y artefactos mínimos obligatorios de cada fase del proyecto

- Además de los entregables destinados a la gestión del proyecto, hay que mantener la documentación de las aplicaciones o componentes implicados en el proyecto
- En fase de contratación del proyecto, se pueden requerir entregables adicionales siguiendo recomendaciones de DCU, Calidad, Seguridad o Arquitectura, que habrá que añadir al registro de entregables del proyecto
- Habrá que tener aceptados todos los entregables de una fase antes de poder darla por cerrada
- Cuando un entregable de aplicación va acompañado de una "n" indica que hay que hacer tantos entregables de aquel tipo como número de aplicaciones haya implicadas al proyecto
- Los artefactos de seguimiento y control sirven para poder consultar en cualquier momento el estado del proyecto. El proveedor tiene que mantener actualizados obligatoriamente los artefactos el máximo posible

## Entregables Existentes

A continuación se muestra la relación completa de entregables existentes asociados a cada fase y el hito de proyecto a la cual está asociados.

### 1. Fase Maduración

Hitos: El hito inicial de esta fase es: "Inicio de Proyecto", el hito resultante de la aceptación de los entregables y actividades que corresponden a esta fase es: "KickOff del proyecto"

Entregable	Detalle
Ficha Iniciativa	Una Iniciativa es una agrupación de uno o más Proyectos. Una Iniciativa pertenece en un Programa, que a su vez es una agrupación de una o varias Iniciativas.
Pedido	Los pedidos son básicos para poder empezar un proyecto y sirven para vehicular los pagos a los proveedores.
Contrato	El contrato es el marco legal bajo el cual se puede llevar a cabo un proyecto. Nos servirá como referencia para consultar las condiciones pactadas con el proveedor. Contratación de servicios o adquisición de licencias que requieren tramitación o concurso: contratos menores, negociados, abiertos, armonizados y derivadas en concursos de homologación. Modificación de contratos ya existentes
Ficha de Programa	Desde un punto de vista operativo, un Programa es una agrupación de varias Iniciativas (1,N).
Gestión de Cambio	Una Gestión de Cambio es un tipo de issue que se utiliza cuando el presupuesto inicial se ve afectado por un cambio de alcance en el proyecto o la aparición de un nuevo requerimiento.
Objetivo	La issue de tipo "Objetivo" refleja el motivo por el cual emprendemos una acción (Programa o Iniciativa), y define que pretendemos conseguir con esta acción. Nos dice donde estamos y donde queremos llegar con la implantación.
Documento de Necesidades	El documento de Necesidades define las necesidades del proyecto, los aspectos metodológicos que tiene que satisfacer el proveedor y las condiciones de contratación y prestación del servicio. Para validar la Propuesta de solución, y para elaborar el pliego de necesidades, existe como recurso la <b>checklist de verificación de la propuesta de solución</b> (la plantilla se encuentra en el apartado de plantillas y repositorio del

	entregable")
Propuesta de Solución	La propuesta de solución es el documento que presenta el proveedor como respuesta a la petición de proyecto que los libramos con el Pliego de Necesidades.

## 2. Fase Inicio

Hitos: El hito inicial de esta fase es: "Inicio de Proyecto", el hito resultante de la aceptación de los entregables y actividades que corresponden a esta fase es: "KickOff del proyecto"

Entregable	Detalle
Presentación kickoff	Utilizar la plantilla que facilita la UOC. En ningún caso, se puede utilizar logo, imágenes, etc... del proveedor. Por más detalles normativos, ver Normativa de Entregables El contenido mínimo corresponde a: <ul style="list-style-type: none"><li>o Objetivos</li><li>o Alcance</li><li>o Planificación</li><li>o <b>Organización</b> (modificada matriz RACI)</li><li>o Órganos de seguimiento</li><li>o Riesgos</li><li>o Entregables + Calculadora de entregables (<b>Novedad</b>)</li><li>o <b>Presupuesto y costes</b> (Presupuesto inicial, posibles rangos de costes, ...)</li><li>o Herramientas</li></ul> La Planificación indicada tiene que estar pactada por todos los implicados del proyecto (Tanto sean de carácter externo como cualquier unidad operativa de carácter interno)
Historias de usuario	Las historias de usuario son una descripción breve de una funcionalidad del producto tal como la percibe el usuario. Una definición de alto nivel de un requisito, que contiene solo la información suficiente porque los desarrolladores puedan producir una estimación razonable del esfuerzo para implementarlo. Tiene que aportar un valor claro al usuario Los requerimientos de un proyecto hay que transformarlos en historias de usuario y épicas. Estos requisitos se elaboran en base a necesidades recogidas al Anteproyecto y al pliego de necesidades.

Planificación	Realizar una planificación estimada en semanas, considerando el Calendario de Distribuciones y las posibles de dependencias del proyecto con otros proyectos en curso.
Documento de Arquitectura (DA) (contiene DAC)	<p>El Documento de Descripción de Arquitectura (DA) es una declaración de intenciones hacia la construcción de un sistema de información, que se convierten en acuerdos una vez aprobada.</p> <p>En este entregable se tiene que describir:</p> <ul style="list-style-type: none"><li>o La relación del nuevo sistema con otros</li><li>o Cómo se construirá el nuevo sistema (con qué tecnologías y porque)</li><li>o Cómo se desplegará el nuevo sistema</li><li>o Qué entidades de datos utilizará</li><li>o Justificación de decisiones que hacen que ciertos aspectos se realicen de una manera concreta</li></ul> <p>El DA lo tiene que elaborar el integrador y se puede iterar hasta que se llega al acuerdo antes de empezar a implementar. La aprobación del DA es responsabilidad del equipo de arquitectura.</p>

### 3. Fase Diseño e implementación

Hitos: En esta fase existen varias hitos, cada una de ellas es obligatoria para que se pueda iniciar el Despliegue a PRO. El hito final de esta fase es "Despliegue a PRO"

Entregable	Detalle
Diseño Funcional	El diseño funcional contiene una descripción detallada de las necesidades del sistema, estableciendo qué hará y como se espera que lo haga. Los casos de uso, los requisitos funcionales y requisitos no funcionales completan conjuntamente la descripción del sistema.
Diseño Técnico	El diseño técnico describe el diseño del sistema, generado a partir de la documentación del análisis y diseño funcional, considerando el conjunto de posibilitats y restricciones específicas de la plataforma.

Código Aplicación - Código Font	<p>El código fuente de una aplicación son todos los ficheros necesarios que se requieren para que una aplicación funcione correctamente. Se divide en Componentes Técnicos.</p> <p>Se considera código fuente:</p> <ul style="list-style-type: none"><li>o Código que generará el ejecutable de la aplicación y todos los ficheros asociados necesarios (ficheros de configuración, ...)</li><li>o Código de cada uno de los componentes técnicos</li><li>o Código de pruebas unitarias y pruebas de integración, código de pruebas automatizadas</li><li>o Scripts de base de datos</li><li>o Infraestructura como código</li><li>o Los artefactos que genera una aplicación</li></ul>
Receta despliegue a la nube	<p>La receta Cloud es un documento en forma de formulario que té dos partes.</p> <p>La primera hace referencia al despliegue de los servicios de infraestructura y en ella se recogen detalles específicos para poder hacer el aprovisionamiento de estos servicios.</p> <p>La segunda hace referencia a la información necesaria para poder incorporar la aplicación en el sistema de CI/CD de la UOC.</p>

Estrategia de Pruebas	<p>El documento de estrategia de pruebas tiene como propósito describir el enfoque con el cual se abordarán las pruebas de un proyecto. Describe los objetivos perseguidos con las pruebas a realizar durante el proyecto, el alcance de estas pruebas, qué elementos serán probados y qué elementos no, las necesidades que se tienen que cubrir para realizar unas pruebas con éxito, qué equipos participarán en las pruebas, qué estrategia de diseño y ejecución de pruebas se seguirá en el proyecto, así como una planificación de todas estas actividades.</p> <p>El contenido mínimo que tiene que tener es:</p> <ul style="list-style-type: none"><li>o Planteamiento general</li><li>o Referencias a la documentación funcional</li><li>o Planificación de las pruebas</li><li>o Incluido en el alcance de las pruebas</li><li>o No incluido en el alcance de las pruebas</li><li>o Riesgos según características de calidad</li><li>o Riesgos según Funcionalidades / Historias de usuario</li><li>o Riesgos del proceso de pruebas</li><li>o Estrategia de ejecución de pruebas</li><li>o Criterios de entrada, de salida, de suspensión y de reanudación de pruebas</li><li>o En torno</li><li>o Infraestructura y Datos</li><li>o Informe de resultados</li><li>o Catálogo de pruebas</li></ul> <p>Los criterios de aceptación del documento de estrategia de pruebas son:</p> <ul style="list-style-type: none"><li>o La estrategia de pruebas está consensuada con todos los miembros del equipo: (proveedor, equipo proyecto AT y usuarios)</li><li>o La estrategia de pruebas es realista y realizable</li><li>o La estrategia de pruebas define las pruebas que realizarán cada uno de los equipos en en cuanto a funcionalidades, técnicas y temporalidad</li></ul>
Manual de Operaciones	<p>Este documento recoge la información necesaria para que AiS Operaciones mantenga el servicio ala nivel requerido.</p> <p>El Contenido Principal es información relativa a los elementos de configuración del servicio, monitorización, backup, operativas de respuesta a incidentes, continuitat, ....</p> <p>El responsable de Operaciones asignado al proyecto aceptará el Manual de Operaciones.</p>

Terraform	<p>Este documento especifica la <b>normativa y como desarrollar código HCL dentro de la UOC porque sea aceptado</b>.</p> <p>Terraform es una herramienta de código abierto de “Infraestructura como código (IaC)”, creada por HashiCorp, que permite crear, modificar y versionar infraestructura de manera eficiente.</p> <p>ES UNA HERRAMIENTA DE CODIFICACIÓN DECLARATIVA (ANSIBLE, EN CAMBIO, COMBINA LA CONFIGURACIÓN DECLARATIVA CON LA PROCEDIMENTAL) QUE UTILIZA UN LENGUAJE DE ALTO NIVEL LLAMADO HCL (HASHICORP CONFIGURATION LANGUAGE) PARA DESCRIBIR EL ESTADO FINAL DE UNA INFRAESTRUCTURA CON EL OBJETIVO DE EJECUTAR UNA APLICACIÓN.</p>
Wireframe	<p>El objetivo de este entregable es definir y decidir sobre:</p> <ul style="list-style-type: none"><li>o La estructura básica de la interfaz (layout)</li><li>o El diseño de interacción en escritorio y responsive</li><li>o Arquitectura de la información, sistemas de navegación y flujo de pantallas</li><li>o Elementos de interacción para usuarios de lector de pantalla</li><li>o Como las funcionalidades quedan representadas en la interfaz y reflejan los objetivos de negocio contemplados en los abates del proyecto</li></ul>
Diseño gráfico	<p>El objetivo de este entregable es definir y decidir sobre la aplicación de los estilos gráficos UOC en la interfaz.</p> <p>Si no se ha elaborado un wireframe previo, también permite definir:</p> <ul style="list-style-type: none"><li>o La estructura básica de la interfaz (layout)</li><li>o El diseño de interacción en escritorio</li><li>o Arquitectura de la información, sistemas de navegación y flujo de pantallas</li><li>o Elementos de interacción para usuarios de lector de pantalla</li><li>o Como las funcionalidades quedan representadas en la interfaz</li></ul>
Funcional de interacción (UX)	<p>El objetivo de este entregable es describir las interacciones y funcionalidades de la interfaz con detalle suficiente para que el equipo de desarrollo pueda:</p> <ul style="list-style-type: none"><li>o Valorar el esfuerzo económico</li><li>o Desarrollar la solución</li></ul>
Anotaciones de accesibilidad sobre wireframe y diseño	<p>El objetivo de este entregables es definir y decidir sobre ciertos elementos de la interfaz a nivel semántico y de equitatge que condiciona la UX de usuarios con lector de pantalla y el cumplimiento de la doble A.</p>

Anotaciones de los colores empleados en el diseño	Las anotaciones de los colores empleados en el diseño contiene una lista de la combinación de colores que hemos empleado y su relación de contraste a la hora de crear los diseños gráficos de la herramienta.
Frente-end	Este entregable permite consolidar: <ul style="list-style-type: none"><li>o Elementos de interacción y navegación</li><li>o Diseño gráfico</li><li>o Cumplimiento de la accesibilidad</li><li>o Estándares y buenas prácticas html</li></ul>
Producto en PRE (UX)	Tener un producto/aplicación/funcionalidad plenamente funcional para consolidar: <ul style="list-style-type: none"><li>o Elementos de interacción y navegación</li><li>o Diseño gráfico</li><li>o Cumplimiento de la accesibilidad</li><li>o Estándares y buenas prácticas html</li><li>o Contenido real o simulado que replique las casuísticas de visualización</li></ul>
Service Blueprint	El objetivo de este entregable es mapejar los Servicio digitales para diseñarlos y/o mejorarlos, identificando: <ul style="list-style-type: none"><li>o Las acciones del usuario y sus touchpoints</li><li>o Los flujos de tareas información</li><li>o Las tareas y personas del Negocio</li><li>o Herramientas y procesos internos</li></ul>

Estrategia de Pruebas	<p>El documento de estrategia de pruebas tiene como propósito describir el enfoque con el cual se abordarán las pruebas de un proyecto. Describe los objetivos perseguidos con las pruebas a realizar durante el proyecto, el alcance de estas pruebas, qué elementos serán probados y qué elementos no, las necesidades que se tienen que cubrir para realizar unas pruebas con éxito, qué equipos participarán en las pruebas, qué estrategia de diseño y ejecución de pruebas se seguirá en el proyecto, así como una planificación de todas estas actividades.</p> <p>El contenido mínimo que tiene que tener es:</p> <ul style="list-style-type: none"><li>o Planteamiento general</li><li>o Referencias a la documentación funcional</li><li>o Planificación de las pruebas</li><li>o Incluido en el alcance de las pruebas</li><li>o No incluido en el alcance de las pruebas</li><li>o Riesgos según características de calidad</li><li>o Riesgos según Funcionalidades / Historias de usuario</li><li>o Riesgos del proceso de pruebas</li><li>o Estrategia de ejecución de pruebas</li><li>o Criterios de entrada, de salida, de suspensión y de reanudación de pruebas</li><li>o En torno</li><li>o Infraestructura y Datos</li><li>o Informe de resultados</li><li>o Catálogo de pruebas</li></ul> <p>Los criterios de aceptación del documento de estrategia de pruebas son:</p> <ul style="list-style-type: none"><li>o La estrategia de pruebas está consensuada con todos los miembros del equipo: (proveedor, equipo proyecto AT y usuarios)</li><li>o La estrategia de pruebas es realista y realizable</li><li>o La estrategia de pruebas define las pruebas que realizarán cada uno de los equipos en en cuanto a funcionalidades, técnicas y temporalidad</li></ul>
-----------------------	--

Pla de Pruebas (Test Plan)	<p>Los casos de prueba se desarrollan para definir los elementos que son necesarios validar a fin de asegurar que la aplicación funciona correctamente, soporta la carga esperada y está construida con un alto nivel de calidad.</p> <p>La definición de los casos y de la test plan hace falta que responda al definido a la Estrategia de Pruebas</p> <p>Un caso de prueba tiene que contener básicamente:</p> <ul style="list-style-type: none"><li>○<ul style="list-style-type: none"><li>§ Precondiciones</li><li>§ Datos de prueba</li><li>§ Paso a paso</li><li>§ Resultado Esperado</li></ul></li></ul> <p>Hay que definir tantas macetas planas como se defina a la estrategia de pruebas y hace falta que sean coherentes y completos.</p> <p>Al plan de pruebas se tienen que incluir las pruebas de carga (ver Pruebas de Rendimiento)</p> <p>El Plan de pruebas de accesibilidad a confluence es obligatorio</p>
Informe de Resultados de Pruebas	El Informe de resultado de las pruebas (por cualquier tipo de plan de pruebas) acontece el conjunto de resultados surgidos de la ejecución del plan de pruebas. Tiene que existir la trazabilidad entre el que se define a la estrategia de pruebas, los hitos del proyecto, los milestones a Testrail y los informes de resultados.
Informe resultados pruebas accesibilidad	El informe de pruebas de accesibilidad contiene una descripción detallada de las pruebas que se han llevado a cabo en la maqueta y a PRE para garantizar el logro del nivel de accesibilidad indicado en los requerimientos. Hay que referenciar las herramientas empleadas (como lectores de pantalla, herramientas automáticas, etc). Completa el informe la mesa donde se indica si pasa o no cada uno de los criterios .

#### 4. Fase Implementación

Hitos: ES OBLIGATORIO TENER LOS ENTREGABLES DE ESTA FASE ACEPTADOS PARA LOGRAR LA ACOTA "INICIO DESPLIEGUE A PROD". Una vez lograda, los siguientes hitos que se tienen que indicar sueño: "Despliegue a PROD Finalizado" y "GO LIVE". En algunos casos, estos dos hitos pueden tener el mismo End Dato, pero se tienen que diferenciar tal como se indica.

Entregable	Detalle
------------	---------

Manual de Operaciones	<p>Este documento recoge la información necesaria para que AiS Operaciones mantenga el servicio al nivel requerido. El contenido principal es información relativa a los elementos de configuración del servicio, monitorización, backup, operativas de respuesta a incidentes, continuidad, etc.</p> <p>El responsable de Operaciones asignado al proyecto aceptará el Manual de Operaciones.</p>
Documento de Acuerdos de Ciberseguridad (DAC)	<p>Este documento establece un acuerdo de compromiso vinculante entre la UOC y el desarrollador por el qué se acuerda maximizar la seguridad del software desarrollado y de las aplicaciones adquiridas, de acuerdo con una serie de términos que se tendrán que pactar antes de empezar cualquier actividad del desarrollo. Al más alto nivel la UOC y el desarrollador acuerdan que:</p> <ul style="list-style-type: none"><li>● Las decisiones de seguridad comprometidas en este acuerdo están basadas en la comprensión del riesgo inherente a la aplicación desarrollada y los datos tratados.</li><li>● Que las actividades dedicadas a incrementar la seguridad de la aplicación estarán balanceadas y fuertemente distribuidas a lo largo del ciclo de vida del desarrollo.</li><li>● Las actividades y la documentación generada hará falta que se integren en el ciclo de vida del desarrollo del software y en ningún caso estarán separadas del resto del proyecto.</li><li>● Toda la información relativa a la seguridad de la aplicación será revelada, informada y compartida por parte del proveedor y la UOC. Si se diera el caso, el proveedor especificará de forma argumentada qué aspectos de la seguridad de la aplicación y del desarrollo piensa no cumplir.</li></ul>
Manual de Usuario	El manual o guía de usuario recoge la especificación por el uso del software.
Pla de formación	El Plan de formación define las acciones de formación para los usuarios y actores relacionados.

## 5. Fase Cierre

Hitos: El hito resultante de esta fase es "Cierre". Se tiene que asegurar que todos los hitos indicados tipos "Hitos claves de NEGOCIO" hayan sido logradas.

Entregable	Detalle

Informe de Cierre	<p>El informe de cierre detalla todos los aspectos del cierre del proyecto como administrativos y de relación con los proveedores, logro de objetivos, etc.</p> <p>El informe de cierre incluye una evaluación global del desarrollo del proyecto, con el objetivo de reflejar la calidad y grado de satisfacción de los productos obtenidos, evolución del proyecto en tiempo, alcance, alineamiento con los objetivos iniciales, etc. Además, se recogerán las experiencias positivas y negativas por que sirvan de ayuda en situaciones futuras.</p> <p>Criterios de aceptación</p> <ul style="list-style-type: none"><li>o La aplicación está a disposición del usuario.</li><li>o Se han librado todos los entregables mínimos Normativos siguiendo la Normativa de Entregables</li><li>o Se ha hecho planificación bisiesta a mantenimiento</li><li>o Se ha hecho una reunión de cierre del proyecto con los actores interesados.</li></ul>
Traspaso a Mantenimiento	<p>El objetivo de este procedimiento, es asegurar que un desarrollo, una vez superada el periodo de garantía, esté soportado por un servicio de mantenimiento que disponga de todo el conocimiento necesario para poder lograr los MÁS BIEN definidos. Este procedimiento es normativo, independientemente de que el proveedor desarrollador del proyecto sea el mismo o no que el proveedor que asume el mantenimiento del producto resultante.</p> <p>Este proceso se inicia junto con la fase de Garantía del proyecto. Al finalizar el periodo de garantía, se tiene que asegurar que el traspaso ya tiene la aceptación por parte del Responsable del Servicio de la UOC.</p> <p>Para asegurar un proceso bisiesto satisfactorio y sin impactos en el servicio, es indispensable:</p> <ul style="list-style-type: none"><li>o Identificación de los interlocutores claves del proceso</li><li>o Identificación de las necesidades de mantenimiento</li><li>o Entrega de la documentación requerida tanto a nivel funcional como técnico</li><li>o Sesiones de formación / resolución de dudas del proveedor destino</li><li>o Aceptación por parte del responsable del servicio de la UOC</li></ul>
Informe declaración accesibilidad	<p>El objetivo de la declaración de accesibilidad es obtener la declaración del proveedor respecto al cumplimiento de la doble A. La cual contiene una declaración del nivel de accesibilidad que se ha logrado al final del proyecto. Esta declaración de accesibilidad se tendría que mostrar en el enlace "accesibilidad" que hay que añadir en el pie de toda aplicación web. Si se trata de una aplicación móvil si tiene que poder acceder desde la app.</p>

## 6. Artefactos Seguimiento y Control

Artefactos Básicos de Proyecto	Detalle
Estos artefactos son los básicos para el seguimiento y control de un proyecto. Aplican a cualquier tipología de proyecto	
Acta de Reunión	<p>El acto de la reunión es un artefacto de seguimiento que contiene un resumen de todos los temas tratados en las diferentes tipos de reuniones y seguimientos que se realizan en el curso del proyecto. Hay que crear una acta de cada reunión que se haga durante la ejecución del proyecto. Todas estas actas de reuniones se agrupan en el registro de reuniones que se encuentra en el espacio Confluence del proyecto de forma que nos permite conocer en todo momento el historial de las decisiones que se han tomado en las diferentes reuniones. Para generar la lista de actas de reunión se puede usar la plantilla CLAVE_Registro de Reuniones</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none"><li>o Las actas se darán por aprobadas automáticamente si en el plazo de 3 días laborables si no se recibe ninguna enmienda.</li><li>o Las enmiendas, en caso de recibirse, serán introducidas en el acta y esta será nuevamente distribuida, considerándose como definitiva. Cualquier apreciación sobre las enmiendas se tendrá que tratar en la sesión siguiente</li></ul>
Registro de Reuniones	El registro de reuniones es un artefacto (página de confluence) que contiene un índice de todas las actas realizadas durante el proyecto. Hay que configurarlo al inicio del proyecto. Se actualiza dinámicamente a medida que se crean actas de reunión en el espacio del proyecto tal y como indica Acta de Reunión. Hay que revisar que siempre contenga todas las actas de reunión del proyecto.
Informe de Dirección	Informe por el Comité de Dirección. Contenidos mínimos: Informaciones relevantes y balance de la ejecución, Planificación, Hitos y dependencias, Riesgos
Informe de Seguimiento	Informe de Segment del Proyecto. Contenidos mínimos: Tareas en curso, Tareas pendientes, Planificación, Hitos y dependencias, Hechos relevantes, Riesgos.
Ficha de Proyecto	<p>La ficha de proyecto es el artefacto de seguimiento que contiene información del estado del proyecto, los participantes, la planificación, y los costes del proyecto</p> <p>Hay que mantenerla actualizada con una periodicidad máxima de una semana.</p>

Seguimiento de hitos	<p>Todos los hitos consensuados en el kick off tienen que estar registradas a JIRA. Se indispensable que el jefe de proyecto revise el estado y la planificación de los hitos una vez a la semana durante el proyecto para que en cualquier momento se pueda coneixer el avance del proyecto revisando la planificación de los hitos.</p> <p>Los hitos las registramos a JIRA con la issuetype llamado "Fita". Todos los proyectos tienen unos hitos básicos comunes. Hay un mecanismo de creación masiva que las genera, ayudando de este modo al jefe de proyecto con la creación de hitos. Hará falta que revisáis los hitos creados y las adaptáis a vuestro proyecto.</p>
Seguimiento de riesgos	<p>Todos los riesgos consensuados en el kick off, y cualquier riesgo que se detecte durante la ejecución del proyecto, tienen que estar registrados a JIRA. Los riesgos tienen que estar muy informados indicando:</p> <ul style="list-style-type: none"><li>o Tipo Riesgo</li><li>o Responsable</li><li>o Plà de mitigación</li><li>o Fecha posible mitigación</li></ul> <p>ES INDISPENSABLE QUE EL JEFE DE PROYECTO ACTUALICE ESTA INFORMACIÓN UNA VEZ POR SEMANA. En caso de ser necesario, desde la issue Type Riesgo del JIRA, se puede escalar a la CDA marcando el campo "flagged" = impedimento</p>
Registro de entregables	<p>El registro de entregables es un artefacto que contiene un índice del proyecto que sirve para hacer el seguimiento de la entrega y la aceptación de los entregables de cada fase del proyecto.</p> <p>Al inicio del proyecto o de cada fase hay que definir el conjunto de entregables que se tienen que generar durante la ejecución del proyecto. Todos estos entregables se agrupan en el registro de entregables que esta dentro de la hombre del proyecto de forma que nos permite coneixer en todo momento el estado de los entregables de cada fase. Para generar la lista de entregables se puede usar la plantilla CLAVE_Registro de entregables</p> <p>Criterios de aceptación</p> <ul style="list-style-type: none"><li>o Aparece una entrada para cada entregable específico del proyecto (entregables que se definieron al kickoff)</li><li>o Todos los entregables específicos del proyecto se encuentran en estado Aprobado o N/A</li></ul>
Hombre de los Proyectos	<p>La hombre del proyecto se la puerta de entrada a la documentación del proyecto a confluence.</p> <p>Al dar de alta el proyecto se crea transparentemente por el equipo esta portada. Cuando el espacio del proyecto está creado a Confluence, el equipo de proyecto tiene que configurar el espacio del proyecto: adaptar las macros y páginas de todo el espacio,al contexto del proyecto.</p> <p>Hay que mantener actualizadas y en funcionamiento, todas las macros y</p>

	páginas de Confluence así como las cabeceras de registro de aceptación de cada una de las páginas que pertenecen en el espacio
<b>Artefactos de proyectos de implantación o evolución de aplicaciones</b>	<b>Detalle</b>
Estos artefactos hay que añadirlos cuando el proyecto implica la implantación o evolución de una o más aplicaciones	
Ficha de Aplicación y Componente Técnico	La ficha de aplicación contiene toda la información relativa a una aplicación y es un issuetype de Jira. Normalmente, solo existe una por aplicación. Las aplicaciones se dividen técnicamente en componentes técnicos. La ficha de aplicación y la de sus componentes técnicos tiene que estar siempre actualizada. <b>ES OBLIGACIÓN DEL SERVICIO DE MANTENIMIENTO MANTENERLA ACTUALIZADA PERO CUANDO UN PROYECTO LA IMPLANTA O LO EVOLUCIONA, ES RESPONSABILIDAD DEL PROYECTO MANTENERLA ACTUALIZADA</b>
Panel de Calidad	El panel de Calidad es un Board que permite hacer el seguimiento de la calidad del proyecto en tiempo real. Hay que configurarlo al inicio del proyecto y velar porque siempre muestre como mínimo: <ul style="list-style-type: none"><li>o Incidencias / Bugs pendientes de resolver</li><li>o Deployes</li><li>o Calidad del código (SonarQube)<ul style="list-style-type: none"><li>§ Calidad de código inicial</li><li>§ Calidad del código actual</li></ul></li><li>o Checklist de entregables</li></ul>
<b>Artefactos Agrupadores de documentos de aplicaciones</b>	<b>Detalle</b>
Estos artefactos hay que configurarlos en el espacio del Proyecto para poder mostrar la documentación que se genera en los espacios de las aplicaciones en el marco del proyecto. Su propio nombre indica los documentos de las aplicaciones que contiene.	
Documentos de Arquitectura (agrupador)	Contiene referencias a todos los documentos de arquitectura de las aplicaciones implicadas en un proyecto.
Documento de acuerdos de ciberseguridad	Contiene referencias a todos los documentos de acuerdos de ciberseguridad de las aplicaciones implicadas en un proyecto.

(Agrupador)	
Manuales de Operaciones (Agrupador)	Contiene referencias a todos los manuales de operaciones de las aplicaciones implicadas en un proyecto.
Diseños Técnicos (Agrupador)	Contiene referencias a todos los diseños técnicos de las aplicaciones implicadas en un proyecto.
Diseños Funcionales (Agrupador)	Contiene referencias a todos los diseños funcionales de las aplicaciones implicadas en un proyecto.
Manuales de usuario (Agrupador)	Contiene referencias a todos los manuales de usuario de las aplicaciones implicadas en un proyecto.
Traspasos a Mantenimiento (Agrupador)	Contiene referencias a todos los documentos bisiestos a mantenimiento de las aplicaciones implicadas en un proyecto.
Recetas de despliegue a la nube (Agrupador)	Contiene referencias a todas las recetas de despliegue a la nube de las aplicaciones implicadas en un proyecto.

## Documentación y entregables de los pequeños Evolutivos

- Los evolutivos, del mismo modo que evolucionan la aplicación, también tienen que evolucionar su Suite de Pruebas, creando o eliminando tantos Casos de Prueba y Planes de Pruebas como sea necesario y siempre vinculados a un Milestone que identifique el evolutivo (con la Clave)

El espacio Confluence de la aplicación también hay que mantenerlo actualizado ya sea para hacer las modificaciones pertinentes o si no hay contenido, habrá que añadirlo solo de las partes afectadas por el evolutivo. Estos son los documentos que normalmente hay que revisar del espacio de la aplicación pertinente:

- Diseño Técnico
- Diseño Funcional
- Test Suite

- Manual de Usuario
- Documento de Arquitectura
- Manual de Operaciones
- Hoja de ruta: Hay que actualizar los cambios que se hacen a la aplicación en una línea temporal.

## Documentación y entregables durante el mantenimiento de aplicaciones

- Durante el periodo de mantenimiento de las aplicaciones, hay que mantener actualizada la documentación de las aplicaciones a Confluence y también su ficha de aplicación. Los principales documentos que hay que mantener actualizados son:
  - Diseño Técnico
  - Diseño Funcional
  - Test Suite
  - Manual de Usuario
  - Documento de Arquitectura
  - Manual de Operaciones
  - Hoja de ruta: Hay que actualizar los cambios que se hacen a la aplicación en una línea temporal.
  - Errores Documentados
- Toda la documentación en lo referente a una aplicación tiene que mantenerse dentro del espacio Confluence de la aplicación. (No en el espacio del servicio de mantenimiento o el espacio dedicado a otros equipos o proyectos). En todo caso, puede referenciarse desde otros espacios.

## Revisión y aceptación de los entregables

La revisión y aceptación de los entregables se hace mediante el registro de entregables de los proyectos que permanecerá a Confluence.

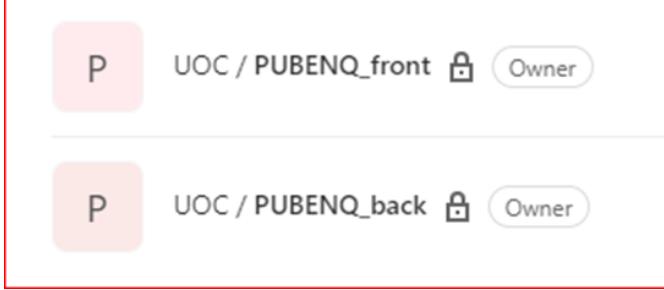
- Se considera la fecha de entrega cuando el proveedor hace un comentario a pie de página del entregable de Confluence informando al responsable UOC de su entrega y actualizará la cabecera del documento.
- Por defecto se dispondrán de un máximo de 10 días laborables por la aceptación de los documentos. Por las actas de reunión este plazo es de 3 días laborables. Pasados estos plazos, el proveedor puede dar el documento por aceptado automáticamente y modificar la cabecera del entregable a Confluence.
- En caso de enmiendas, se añadirá un comentario a pie de página del documento y se puede complementar con comentarios inline en el propio documento.
- Las no conformidades se tendrán que resolver por defecto en una máximo de 2 días laborables. En este caso, las revisiones posteriores por parte de la UOC dispondrán, por defecto 5 días laborables.

## Nomenclatura

- ES OBLIGATORIO MANTENER LA NOMENCLATURA DE LAS APLICACIONES DE FORMA COHERENTE A TODAS LAS HERRAMIENTAS PARA PODER GARANTIZAR LA TRAZABILIDAD ENTRE ELLAS.
- Cuando hablamos de "CLAVE" (KEY), estamos hablando de la CLAVE que viene derivada de la aplicación a JIRA.

## Nomenclatura a las herramientas de desarrollo

Herramientas de desarrollo	Aplicación "DEMO"	Componentes Técnicos " DEMO_frente, DEMO_back..."
JIRA	<p>Cada aplicación tiene un proyecto JIRA asociado. Ver [Tipo de proyectos ji</p> <p>La clave (CLAVE) de este proyecto de JIRA es el identificador principal de la aplicación (que se propaga al resto de herramientas)</p> <p>La información básica de la aplicación se encuentra a la Ficha de Aplicación</p>	<p>Muchas aplicaciones se dividen en componentes, a JIRA se definen con un issuetype "componente" que cuelga de la Ficha de Aplicación</p> <p>Al campo "componente técnico" de las issues de tipos "deploy" se muestran todos los componentes de la aplicación.</p>

<b>Confluence</b>	<p>Hay un espacio de documentación a Confluence para cada aplicación. La relación con los proyectos a JIRA es 1proyecto jira=1espaí confluence.</p>	<p>La documentación de los componentes queda dentro del espacio de Confluence de la aplicación. Se organiza en páginas y subpàginas</p>
<b>Gitlab</b>	<p>Todos los repositorios de aplicaciones cuelgan bajo el grupo principal /UOC. Los repositorios de los componentes técnicos tienen que estar anidats sobre una carpeta con la KEY de la aplicación.</p> <p><b>Aplicaciones o componentes técnicos nuevos</b> Dentro de la carpeta de la aplicación se crean los diferentes repositorios, uno para cada componente técnico que requiera repositorio de código. El nombre del Componente Técnico sigue la Normativa de Nomenclatura</p> <p><b>Ejemplo de organización correcta de aplicación y componentes técnicos a gitlab</b></p> <p><b>Nueces Componentes técnicos de aplicaciones legacy</b> En aplicaciones legacy, a veces no hay anidació de componentes técnicos bajo la carpeta de la aplicación. En el momento de crear un nuevo componente técnicos, si no está la carpeta, habrá que pedir su creación y organizar correctamente la jerarquía como mínimo del nuevo componente técnico. <b>ES RECOMENDABLE DEPENDIENTE DE LA DEDICACIÓN, APROVECHAR Y RECOLOCAR TAMBIÉN LOS OTROS COMPONENTES TÉCNICOS BAJO LA NUEVA CARPETA DE LA APLICACIÓN.</b></p> <p><b>Ejemplo de organización incorrecta de aplicación y componentes técnicos a gitlab.</b> Los componentes técnicos cuelgan directamente de /UOC en ninguna parte de /UOC/PUBENQ, que no existe.</p> 	
<b>Outsystems</b>	<p><b>Por cada componente (Application dentro de OutSystems)</b> se tendrá que especificar el siguiente:</p> <p><i>Nombre: &lt;PROGRAMA&gt; &lt;APLICACIO&gt; &lt;COMPONENT&gt; (Ej. GOCE GDD Backoffice, GOCE GDD Core, ...)</i>  <i>Tipo: Reactivo WebApp, Tablet App, Phone App, External Web Portal, Traditional Web, Service</i>  <i>Capa (calculado): End User, Core, Foundation</i>  <b>Y por cada módulo dentro de una Application:</b>  <i>Nombre: &lt;APLICACIO&gt;_&lt;COMPONENT&gt;_&lt;SUFFIX&gt;</i>  <i>Tipo: Reactivo WebApp, Tablet App, Phone App, External Web Portal, Traditional Web, Service, Library, Extension</i>  <i>Capa (calculado): End User, Core, Foundation</i></p>	

<b>Jenkins</b>	<p>Se crea una carpeta por aplicación denominándola CLAVE FOLDER</p> <p>Dentro de la carpeta de la aplicación se crean las pipas o jobs necesarios para cada componente técnico. El nombre del componente técnico es el prefijo de la pipe o job de jenkins. El nombre del componente técnico coincide con el summary de la issue componente a JIRA y con los valores que tiene el campo "Componente técnico" definido para aquella aplicación a Jira.</p> <p>Los nombres tienen que coincidir para qué CI/CD funcione correctamente</p>
<b>SonarQube</b>	<p>La aplicación (CLAVE) se define a nivel de proyecto</p> <p><b>Normativo 2020/01:</b> el parámetro "key" de Sonarqube, tiene que coincidir con el nombre del componente técnico. Con el procedimiento de CICD, las ramas no productivas se identifican con un sufijo automáticamente.</p> <p>ejemplo no correcto: key (Sonarqube): edu.uoc.demo:demo_back</p> <p>ejemplo correcto: key (Sonarqube): DEMO_back</p> <p>Ejemplo de rama: DEMO_back_staging</p>
<b>TestRail</b>	<p>Existe un repositorio general de casos de prueba que se encuentra al proyecto TestRail denominado "_APLICACIONES"</p> <p>Por cada aplicación (coincidente con CLAVE JIRA) se creará <b>una test suite</b> dentro del proyecto "_APLICACIONES"</p> <p>Se define un Milestone para cada proyecto, su nombre es la CLAVE del proyecto.</p> <p>La definición de los casos de prueba normalmente se hace en términos funcionales o por historias de usuario, no se separa por componente técnico pero en caso de hacerse, se haría usando el concepto de TestRail de "sections" dentro de la test suite.</p>

<b>CMDB</b>	CI (Aplicaciones) equivaliendo a la CLAVE de jira.	CI (Aplicaciones) añadiendo la relación de "consists of" con la aplicación y el entorno como sufijo (excepto a PRO)  El ejemplo quedaría así:  DEMO_frente_pre  DEMO_frente_maceta  DEMO_frente  DEMO_infra
<b>ApiGateway</b>		
<b>AWS</b>	Política de nomenclatura del servicios cloud públicos	Política de nomenclatura del servicios cloud públicos
<b>Azure</b>	Política de nomenclatura del servicios cloud públicos	Política de nomenclatura del servicios cloud públicos

## Función Ninguno de Proyecto del licitador/proveedor

- Asume los objetivos globales del Proyecto
- Formula planes para la realización del Proyecto
- Hace seguimiento de la actividad y monitora los resultados extremo a extremo.
- Asigna y gestiona actividades
- Soluciona los problemas
- Gestiona un equipo
- Busca los datos y genera información sobre el proyecto
- Emplea adecuadamente los interesados (Stakeholders).
- Gestiona los riesgos operativos del proyecto con visión global
- Lidera el comité de seguimiento
- Prepara y colidera el Comité de Dirección
- Fase de inception:
  - Define las actividades necesarias y hace una estimación de costes.

- Participa en la maduración y concreción del diseño
- Prepara la ejecución
- Gestiona el alcance
- Se responsabiliza de la actualización de la documentación del proyecto (jira, confluence, drive, KickOff, Seguimientos, Cierre, etc.).
- Seguimiento del Proyecto:
  - Seguimiento Operativo con el estado actual del proyecto (Evolución, equipo, tareas, dependencias, riesgos, planificación, etc.) (Semanal).
  - Escalado de riesgos que ponen en peligro el logro de objetivos del proyecto
  - Identificación de Gestión de cambios que suponga una ampliación del presupuesto del Proyecto

## Control de aplicación del Normativo

El registro de entregables es el indicador de salud sobre el estado de los entregables y también el indicador de cumplimiento de la Normativa.

Hace falta que todos los entregables de una fase del proyecto acontezcan aceptados antes de poder pasar a la siguiente fase del proyecto.

Se realizarán auditorías Internas del control de aplicación del Normativo sin aviso previo.

El incumplimiento de esta Normativa generará No conformidades en los proyectos o los mantenimientos de las aplicaciones.

# Anexo - Normativa de Calidad

A continuación se muestra la normativa de Calidad vigente en el momento de redacción del pliego. Esta Normativa se revisa 2 golpes al año a la Release Normativa. Hay que tener en cuenta que se tiene que cumplir la normativa vigente en cada momento.

## Introducción

La Normativa de Calidad se basa en el Modelo de Calidad de la UOC y extrae los indicadores básicos y Normativos que se tienen que cumplir y las condiciones en las cuales se tiene que hacer.

## Alcance

La Normativa es aplicable a todo el ciclo de vida de un proyecto o aplicación y durante todo el periodo que dura el contrato con un proveedor.

## Propósito

EL OBJETIVO DE ESTE DOCUMENTO ES DAR A CONOCER A LOS COLABORADORES DEL ÁREA DE TECNOLOGÍA LA NORMATIVA QUE APLICA A LOS PROYECTOS EN TÉRMINOS DE CALIDAD. Esta normativa se basa en la (ISO/IEC 25010) System and software quality modelos y en el ISTQB.

La misión es asegurar que, en términos generales, **los proyectos SIEMPRE mejoran la calidad de los sistemas de información de la UOC** basado en los estándares oficiales aplicándolos sobre el contexto de la UOC..

## Audiencia

Este Normativo está dirigido a todos aquellos proveedores del Área de Tecnología de la UOC que estén ofreciendo sus servicios actualmente o lo quieran hacer en un futuro y evidentemente, para los miembros del equipo propio del Área de Tecnología UOC.

## Indicadores Normativos

La Normativa de calidad posa foco en el cumplimiento de los siguientes Indicadores primarios de

obligado cumplimiento.

Indicadores de Calidad	Características relacionadas	Categoría	Métricas	Normativo
Cumplimiento de la doble A de accesibilidad	Usabilidad Accesibilidad	Accesibilidad	A = 0 defectos A y B  B = 0 defectos A y 10 % o menos defectos B  C = 0 defectos A y 20 % o menos defectos B  D = 0 defectos A y más de 20 % defectos B  E = 1 o más defectos A	Siempre que no se especifique el contrario al proyecto:  Severidad de los defectos categorizada en A, B, C pueden promocionar de entornos  Severidad de los defectos categorizada en D, E no pueden promocionar entre entornos
Quality Gate	Fiabilidad, Mantenibilitat, Seguridad	Ánálisis estático de código fuente	Según <a href="#">Quality Gate</a> aplicada al repositorio de Código Font	Tiene que pasar OK (en verde) para poder promocionar entre entornos

Severidad de los Defectos	Fiabilidad, Idoneidad Funcional	Testing Funcional	A = <b>0</b> defectos B = <b>0</b> defectos bloquejants, críticos, altos, normales o menores C = <b>0</b> defectos bloquejants, críticos, altos o normales D = <b>0</b> defectos bloquejants, críticos, altos E = <b>1</b> defecto bloqueando o crítico	<b>Promoción en torno PRE</b> Severidad de los defectos categorizada en A, B, C pueden promocionar Severidad de los defectos categorizada en D, E NO pueden promocionar  <b>Promoción entorno PRO</b> Severidad de los defectos categorizada en A, B pueden promocionar Severidad de los defectos categorizada en C, D, E NO pueden promocionar
---------------------------	---------------------------------	-------------------	---	---

Los indicadores principales se componen de los siguientes indicadores secundarios, también de obligado cumplimiento. Con el cumplimiento de los primarios, se cumplen directamente los secundarios.

Indicadores de Calidad	Características relacionadas	Categoría	Métricas	Normativo
------------------------	------------------------------	-----------	----------	-----------

Rati de Seguridad	Seguridad	Análisis estático de código fuente	A = 0 Vulnerabilidades B = como mínimo 1 Vulnerabilidad Minor C = como mínimo 1 Vulnerabilidad Mayor D = como mínimo 1 Vulnerabilidad Critical E = como mínimo 1 Vulnerabilidad Blocker	Siempre que no se especifique el contrario al proyecto, si el resultado es:  A pueden promocionar de entornos  B, C, D, E no pueden promocionar entre entornos
-------------------	-----------	------------------------------------	---	--

Vulnerabilidades	Seguridad	Análisis estático de código fuente	<p><b>1. BLOCKER</b> Bug with a high probability to impact the behavior of the application in production: memory leak, unclosed JDBC connection, .... The code MUST be fixed immediately.</p> <p><b>2. CRITICAL</b> Either a bug with a low probability to impact the behavior of the application in production or an issue which represents a security flaw: empty catch block, SQL injection, ... The code MUST be reviewed immediately.</p> <p><b>3. MAYOR</b> Quality flaw which can highly impact the developer productivity: uncovered piece of code, duplicated blocks, unused parameters, ...</p> <p><b>4. MINOR</b> Quality flaw which can slightly impact the developer productivity: lines should not be too long, "switch" statements should have at least 3 cases, ...</p> <p><b>5. INFO</b> Neither a bug nor a quality flaw, just a finding.</p>	<p>Siempre que no se especifique el contrario al proyecto, si el resultado es:</p> <p>Vulnerabilidades = 0; pueden promocionar de entornos</p> <p>Vulnerabilidades &gt; 0; NO pueden promocionar de entornos</p>
------------------	-----------	------------------------------------	---	--

Cobertura de Código	Fiabilidad	Análisis estático de código fuente	<p>(literal de SonarQube)</p> <p>It is a mix of Line coverage and Condition coverage.</p> <p><math display="block">\text{Coverage} = (\text{CT} + \text{CF} + \text{LC}) / (2 * \text{B} + \text{LO})</math></p> <p>where</p> <ul style="list-style-type: none"><li>• CT = conditions that have been evaluated to 'true' at least once</li><li>• CF = conditions that have been evaluated to 'false' at least once</li><li>• LC = covered lines = lines_to_cover - uncovered_lines</li><li>• B = total number of conditions</li><li>• EL = total number of executable lines (lines_to_cover)</li></ul> <p>*Cobertura de línea y condition coverage no se evalúan de forma independiente.</p>	<p><b>Aplicaciones / Componentes técnicos nuevos:</b></p> <p>Cobertura de Código &gt;= 80 %</p> <p><b>Aplicaciones Legacy:</b></p> <p>Cobertura de Código Nuevo &gt;= 80 %</p> <p>Cobertura &gt; Cobertura Inicial al proyecto</p>
---------------------	------------	------------------------------------	--	--

Code Smells	Mantenibilitat	Análisis estático de código fuente		Incluido al Indicador Quality Gate
Número de Bugs al Código	Fiabilidad	Análisis estático de código fuente		Según Quality Gate aplicada al repositorio de Código Font
Deuda Técnica	Mantenibilitat	Análisis estático de código fuente		

Número de Defectos	Fiabilidad	Testing Funcional	<p><b>Blocker</b> = Defectos que bloquean completamente una funcionalidad de la aplicación y no se puede acceder de ninguna forma</p> <p><b>Critical</b> = Defectos que bloquean parcialmente la funcionalidad de la aplicación causante graves problemas para usar la funcionalidad</p> <p><b>Mayor</b> = Defectos graves que no bloquean la funcionalidad</p> <p><b>Normal</b> = Defectos de criticidad media</p> <p><b>Minor</b> = Defectos de criticidad baja</p> <p><b>Trivial</b> = Defectos estéticos u ortográficos</p>	Ver Severidad de los Defectos
--------------------	------------	-------------------	---	-------------------------------

## Incumplimiento de la Normativa

El incumplimiento de esta Normativa generará No conformidad en los proyectos, las aplicaciones, los servicios o los programas.

Se considerará una falta grave añadir opacidad en parámetros técnicos para lograr de forma engañosa los criterios de aceptación de los entregables o los indicadores. Por ejemplo es una falta grave definir

casos de prueba triviales o ficticios para cubrir aspectos de cobertura de las pruebas, o concluir que las pruebas de rendimiento han sido exitosas cuando realmente los indicadores obtenidos son malos.

## Control de aplicación de la Normativa

Principalmente será el jefe de proyecto ( o alguien en quien delegue) quién velará por la aplicación correcta del estándar mediante diferentes mecanismos de detección implantados al sistema. En todo caso, cualquier miembro del Área de Tecnología tiene la potestad de controlar la aplicación de este Normativo.

## Herramientas y documentos

Las herramientas requeridas para el cumplimiento de este normativo y provistas por la UOC son:

- JIRA
- Confluence
- Sonarqube
- XRAY
- Jenkins

# Anexo - Normativa sobre UX y Accesibilidad

A continuación se muestra la normativa de Calidad vigente en el momento de redacción del pliego. Esta Normativa se revisa 2 golpes al año a la Release Normativa. Hay que tener en cuenta que se tiene que cumplir la normativa vigente en cada momento.

## UX

### Introducción

La normativa de UX establece los requerimientos, entregas y marco de trabajo a tener en cuenta en los proyectos de AT para asegurar que los clientes y usuarios de los productos y servicios digitales desarrollados y ofrecidos por la UOC tengan una experiencia de usuario satisfactoria, incluyendo también, como experiencia satisfactoria, la accesibilidad.

Esta normativa define, por lo tanto,

- Requerimientos en materia de UX, frente-end y accesibilidad
- Tipología y características de las entregas UX de Accesibilidad pedidos
- Marco de trabajo de las actividades UX en los proyectos

Este requerimientos, entregas y dinámicas de trabajo tienen que ser tenidos en cuenta por parte de los diferentes partners, y tienen que asegurar su cumplimiento:

- En las propuestas de solución
- Durante la ejecución de los proyecto
- La definición y ejecución del plan de pruebas

El grado y alcance del cumplimiento se ajustarán en función de la naturaleza y objetivos de los proyectos, en consenso con el partners.

### Alcance

Esta normativa se aplica en todos aquellos proyectos, tareas y actuaciones sobre productos y servicios digitales donde sus interfaces y/o funcionalidades se vean modificadas y/o se tengan que crear desde cero, y por tanto, tengan un impacto directo o indirecto en la UX de los usuarios.

ES DECIR, SE APLICA TANTO EN PROYECTOS DONDE EL OBJETIVO SEA CONCEPTUALIZAR UN PRODUCTO DIGITAL DESDE CERO, ASÍ COMO PROYECTOS/TAREAS CONSISTENTES AL MODIFICAR, EVOLUCIONAR Y/O AÑADIR UNA FUNCIONALIDAD, LA IMPLEMENTACIÓN DE LA CUAL PROVOQUE QUE LAS INTERFACES DE USUARIO SE TENGAN QUE MODIFICAR PARA PODER DAR SOLUCIÓN A ESTA FUNCIONALIDAD OBJETO DEL PROYECTO/TAREA.

La normativa se aplica tanto en productos a medida (nueces o legacy), como en productos de mercado, y en productos para usuarios de gestión como usuarios cliente (estudiantes, potenciales estudiantes, docentes, etc).

En función de las características del producto y del proyecto/tarea, se podrán hacer excepciones a la normativa. Por ejemplo, habrá más flexibilidad en la aplicación de la normativa en productos digitales UOC antiguos y destinados a usuarios gestión o en productos de mercado, y es será más exigente en productos nuevos destinados a estudiantes.

## Propósito

El propósito de la normativa es, en último término, asegurar en lo posible que los usuarios de los productos y servicios digitales de la UOC tengan una buena experiencia de usuario, es decir, que encuentren los productos y servicios de la UOC fáciles de utilizar, útiles, accesibles, agradables en su uso y con sentido.

En la medida que esta normativa se cumple, se asegura el control de los riesgos que pueden provocar desviaciones en el objetivo de ofrecer una buena experiencia de usuario a los usuarios de la comunidad UOC.

## Audiencia

Esta normativa está dirigido a todos aquellos partners del Área de Tecnología de la UOC que estén ofreciendo sus servicios actualmente o lo quieran hacer en un futuro y evidentemente, para los miembros del equipo propio del Área de Tecnología UOC.

## Requerimientos de UX

A continuación se presenta el conjunto de requerimientos en materia de UX, Frente-end y de Accesibilidad que los productos y servicios digitales de la UOC tienen que cumplir para garantizar que sus usuarios tengan una experiencia de usuario satisfactoria, homogénea y de calidad.

Estos requerimientos tienen que ser tenidos en cuenta por parte de los diferentes partners en los proyectos/mantenimientos que se los encargue, y tienen que asegurar que se planifiquen y ejecuten el conjunto de actividades y entregas necesarias por su cumplimiento durante el ciclo de vida del proyecto/mantenimiento ( fase de maduración y propuestas de solución, durante la ejecución de los proyectos, en la definición y ejecución del plan de pruebas).

REQUERIMIENTOS	DESCRIPCIÓN	EVIDENCIAS
<b>Diseño Centrado en el Usuario (ISO 9241-210:2019)</b>	<p>Relevante por proyectos de transformación o de alto impacto en la UX y en el diseño de servicio</p> <ul style="list-style-type: none"><li>• El diseño está basado en una comprensión explícita de usuarios, tareas y entornos.</li><li>• Los usuarios están involucrados durante el diseño y el desarrollo.</li><li>• El diseño está dirigido y refinado por evaluaciones centradas en usuarios.</li><li>• El proceso es iterativo.</li><li>• El diseño está dirigido a toda la experiencia del usuario.</li><li>• El equipo de diseño incluye habilidades y perspectivas multidisciplinarias.</li></ul> <p>Aplicable también en productos legacy y productos de mercado que contemple ciertas posibilidades de personalización y/o tengan impacto en el diseño y/o la experiencia de uso de un servicio digital.</p>	<p>Se contempla fase de user research para obtener conocimiento sobre las necesidades del usuario final (interno o externo).</p> <p>SE HA REALIZADO UNA INICIATIVA PREVIA DONDE SE HA OBTENIDO CONOCIMIENTO SOBRE LAS NECESIDADES DEL USUARIO FINAL (INTERNO O EXTERNO).</p> <p>Se contemplan actividades dirigidas a la evaluación de la UX.</p> <p>Se contemplan iteraciones previstas con usuarios finales</p> <p>Se contemplan iteraciones previstas con UX Lead de la UOC y Negocio UOC.</p>

<b>Perfiles UX/Service</b> Designer/User Researcher/ UI	El equipo de diseño incluye habilidades y perspectivas multidisciplinarias.	Se contempla en la iniciativa el detalle de los perfiles del proveedor incluidos en la ejecución del proyecto (researcher, UI, UX, etc).
---	---	--

<b>Usabilidad y diseño de interacción</b>	<p>USO DE ESTÁNDARES, BUENAS PRÁCTICAS, Y PRINCIPIOS BÁSICOS DE USABILIDAD E INTERACCIÓN</p> <p>Diseño de interacción homogéneo con el resto de espacios digitales de la UOC y la guía de estilo/UI kits de la UOCs.</p> <p>En aplicaciones legacy y de mercado con pocas capacidades de personalización, se prioriza el diseño de interacción propuesto por la propia aplicación para preservar la homogeneidad de la UX dentro de la aplicación.</p>	<p>Existe una test usabilidad/test guerrilla, en desktop y responsive, con usuarios finales con un informe de resultados.</p> <p>Existe un heuristic, en desktop y responsive, con un informe de resultados.</p> <p>Las decisiones sobre diseño de interacción y usabilidad están justificadas en el user research, benchmarking y/o buenas prácticas.</p> <p>Existe un entregable de wireframe y/o diseño gráfico y funcional de interacción</p> <p>El perfil UX/UI del partner revisa y valida las diferentes iteraciones del producto/entregable y el resultado final</p> <p>El Lead UX de AT revisa y valida las diferentes iteraciones del producto/entregable y el resultado final.</p> <p>El responsable de negocio valida el resultado final del producto/entregable.</p>
<b>Layout, estructura, navigation, AI</b>	<p>El producto/herramienta digital tiene que poder ofrecer un layout de interfaz (estructura y ubicación de elementos en la interfaz) igual o similar a los layouts ya existentes en el conjunto de espacios web de la UOC.</p> <p>El producto/herramienta digital tiene que poder ofrecer una navegación igual a las ya existentes en el conjunto de espacios web de la UOC</p> <p>En aplicaciones legacy y de mercado con pocas capacidades de personalización, se prioriza el layout y navegación propuesto por la propia aplicación, para preservar la homogeneidad de la UX dentro de la aplicación</p>	<p>Existe un entregable de wireframe y/o diseño gráfico, y funcional de interacción</p> <p>Existe un entregable d'árbol de contenidos, arquitectura de información, o similar.</p> <p>El perfil UX/UI del proveedor revisa y valida las diferentes iteraciones del producto/entregable y el resultado final</p> <p>El Lead UX de AT ha revisa y valida las diferentes iteraciones del producto/entregable y el resultado final.</p> <p>El responsable de negocio valida el resultado final del producto/entregable.</p>

<b>Estilo gráfico</b>	<p>La interfaz gráfica sigue los estilos gráficos de la UOC indicados en la Guía de estilo/UI kits de la UOC.</p> <p>En aplicaciones legacy y de mercado con pocas capacidades de personalización,, se replica el estilo gráfico de la propia aplicación para preservar la homogeneidad de la UX dentro de la aplicación.</p>	<p>Existe un entregable de diseño gráfico.</p> <p>Existe un entregable de frente-end con el estilo gráfico aplicado.</p> <p>El perfil UX/UI del proveedor revisa y valida las diferentes iteraciones del producto/entregable y el resultado final</p> <p>El Lead UX de AT ha revisa y valida las diferentes iteraciones del producto/entregable y el resultado final.</p> <p>El responsable de negocio valida el resultado final del producto/entregable</p>
<b>Frente-end</b>	<p>LA HERRAMIENTA PERMITE ESCOGER UN FRAMEWORK PARA DESARROLLAR UN TEMA CERCA CON EL OBJETIVO DE PODER APLICAR LA LÍNEA GRÁFICA DE LA UOC. En caso de no ser posible realizar ciertas personalizaciones, se tiene que cumplir con las mejores prácticas del mercado.</p> <p>Se hace carga asíncrona y los archivos JS están minimizados y optimizados.</p> <p>HTML válido (demostrable a <a href="https://validator.w3.org/">https://validator.w3.org/</a>) o HTML5 semántico, basado en estándares internacionales de W3C (<a href="https://www.w3.org/standards/">https://www.w3.org/standards/</a>).</p> <p>Garantiza cumplimiento doble A de accesibilidad.</p> <p>En aplicaciones legacy, el frente-end resultante es continuista con el de la aplicación.</p>	Existe un entregable de frente-end
<b>Responsive</b>	<p>Si la herramienta es una webapp, tendrá que ser responsive para los 4 mínimos puntos de corte (breakpoints) descritos a la Normativa para los diferentes Viewports; XS, SM, MD, LG. (Viewport XL, opcional).</p> <p>En desktop, 12 columnas a 1200 píxeles.</p> <p>Se persigue encontrar diseños responsive, no únicamente adaptativos.</p> <p>Las aplicaciones legacy o de mercado son las que determinarán las capacidades de responsiveness.</p>	<p>Existe un entregable de wireframe y/o diseño gráfico donde se pueda visualizar la interfaz responsive.</p> <p>Existe un entregable de frente-end donde se visualiza la interfaz responsive.</p> <p>El perfil UX/UI del proveedor revisa y valida las diferentes iteraciones del producto/entregable y el resultado final</p> <p>El Lead UX de AT ha revisa y valida las diferentes iteraciones del producto/entregable y el resultado final.</p> <p>El responsable de negocio valida el resultado final del producto/entregable</p>

<b>Crossbrowsing</b>	<p>LA HERRAMIENTA TENDRÁ QUE TENER UNA CORRECTA VISUALIZACIÓN Y COMPATIBILIDAD CON LOS PRINCIPALES NAVEGADORES DE MERCADO.</p> <ul style="list-style-type: none"><li>● Desktop (dos últimas versiones de los navegadores):<ul style="list-style-type: none"><li>○ Windows: Chrome, Firefox y Edge (ie11, navegable. Se conserva y se asegura el acceso al contenido)</li><li>○ iOS: Chrome, Firefox o Safari</li><li>○ Linux: Firefox</li></ul></li><li>● Tablets / Mobile:<ul style="list-style-type: none"><li>○ Android - Android Browser, Chrome</li><li>○ iOS - Safari</li></ul></li></ul> <p>Las aplicaciones legacy o de mercado son las que determinarán las capacidades de crossbrowsing</p>	Existe un entregable donde se demuestra la correcta visualización en los diferentes navegadores del producto entregado.
<b>Performance y Optimización</b>	<p>Buen rendimiento, mesurado con PageSpeed Insights:</p> <ul style="list-style-type: none"><li>● Por encima de 89 %.</li><li>● Entre el 90 y el 100 %</li><li>● 100 % en desarrollos nuevos.</li></ul> <p>Optimización: Iconos en formato fuente o svg, CSS optimizadas y minimizadas.</p> <p>Las aplicaciones legacy o de mercado son las que determinarán las capacidades de performance.</p>	Existe un entregable donde se demuestra la correcta performance del producto entregado.
<b>Multiidioma</b>	<p>El producto/herramienta digital tendrá que ser multidioma, y tendrán que estar disponibles en los idiomas: catalán, castellano e inglés. Así pues, tendrá que ser compatible con caracteres latinos como puede ser el caso de los acentos, la "ñ" y la "ç" (UTF-8).</p> <p>Tiene que incorporar los textos y literales revisados por el Servicio de Lengua de la UOC.</p>	NA

<b>Accesibilidad</b>	<p>Cumplimiento de las pautas de accesibilidad para las páginas webs, consigiendo como mínimo el nivel AA de las directrices de accesibilidad para contenidos web 2.0 del W3C / WAI.</p> <p>Cumplimiento de las normas:</p> <ul style="list-style-type: none"><li>• Decreto 209/2023 del Código de Accesibilidad de Cataluña, que es de aplicación directa: Decreto 209/2023 - Código de accesibilidad de Cataluña</li><li>• Directive (EU) 2019/882 of the European Parliament and of the Council of 17 April 2019 donde the accessibility requirements for products and services</li><li>• Cumplimiento de la Norma UNE-EN 301549 V3.2.1:2022, "Requisitos de accesibilidad de productos y servicios TIC aplicables a la contratación pública a Europa "-" Accessibility requirements suitable for public procurement of ICT products and services in Europe".</li></ul> <p>Se contempla solo la accesibilidad que corresponda al alcance de la actuación, no al conjunto de la aplicación donde se prevé hacer la actuación.</p> <p>Las aplicaciones legacy o de mercado son las que determinarán las capacidades de cumplir con accesibilidad.</p>	<p>Existe el entregable anotaciones de accesibilidad</p> <p>Existe un informe de conformidad</p> <p>Existe una test-run de accesibilidad e informe de pruebas correspondiente.</p>
<b>SEO</b>	<p>Integración de buenas prácticas de SEO dentro del diseño y desarrollo UX/UI, con el objetivo de garantizar una óptima indexación y posicionamiento del sitio web a los motores de busca:</p> <ul style="list-style-type: none"><li>• Arquitectura de la información optimizada por SEO</li><li>• Tiempo de carga optimizado</li><li>• Diseño responsive</li><li>• Accesibilidad</li><li>• Metadatos y etiquetado</li><li>• Estructura de URL amigable</li><li>• Integración con herramientas de análisis y SEO</li></ul>	<p>Informe de auditoría SEO inicial y final.</p> <p>Mapa del lugar (Sitemap) y arquitectura de la información.</p> <p>Informe de tiempo de carga y rendimiento</p> <p>Informe de accesibilidad</p> <p>Documento de metadatos y etiquetado SEO</p>

<b>Satisfacción Usuario</b>	Mantenimiento y mejora de los índice de satisfacción en los servicios y productos de la UOC (en el supuesto de que hayan): CSAT, CES, NPS, SUS, etc.  Integración de herramientas de encuesta en los productos/herramientas digitales.	NA
-----------------------------	--	----

## Accesibilidad

### Introducción

Teniendo en cuenta que la UOC es una universidad privada que recibe fondos públicos, nos es de aplicación el RD 193/2023 de 21 de marzo, por el cual se regulan las condiciones básicas de accesibilidad y no discriminación de las personas con discapacidad para el acceso y utilización de los bienes y servicios a disposición del público". Este Real Decreto 193/2023, especifica que todos los sitios web y aplicaciones móviles el contenido de las cuales se refiera a bienes y servicios a disposición del público tienen que cumplir con la norma de accesibilidad UNE-EN 301549

Aparte, como universidad hay que tener en cuenta el que establece la "Ley Orgánica 2/2023, de 22 de marzo, del Sistema Universitario", que establece que en materia de accesibilidad, las universidades tienen que garantizar a personas con discapacidad un acceso universal en los edificios y sus entornos físicos y virtuales, así como al proceso de enseñanza-aprendizaje y evaluación.

La UOC está desde sus orígenes comprometida con la inclusión social de las personas con discapacidad, y por tanto procura facilitar la accesibilidad de la universidad para todo el mundo quien quiera formar parte de los diferentes colectivos de nuestra comunidad (estudiantes, docentes colaboradores, profesorado y personal de gestión).

Por todo esto, todos los productos y servicios digitales de la UOC tienen que cumplir las pautas WCAG

2.1 publicadas por la WAI al nivel AA (doble A), esto quiere decir qué se tendrán que cumplir los puntos de verificación de prioridad 1 y 2 de las pautas.

## Alcance

Esta Normativa establece cuáles son los requisitos que se tienen que seguir en el desarrollo de las interfaces web, en cuanto a criterios de accesibilidad durante todo el ciclo de vida de un proyecto o aplicación, marcando como estándar las pautas de accesibilidad recogidas a la WCAG 2.1 (Web Content Accessibility Guidelines) especificados en las

- Decreto 209/2023 del Código de Accesibilidad de Cataluña, que es de aplicación directa:  
Decreto 209/2023 - Código de accesibilidad de Cataluña
- Directive (EU) 2019/882 of the European Parliament and of the Council of 17 April 2019 donde the accessibility requirements for products and services
- Cumplimiento de la Norma UNE-EN 301549 V3.2.1:2022, "Requisitos de accesibilidad de productos y servicios TIC aplicables a la contratación pública a Europa "-" Accessibility requirements suitable for public procurement of ICT products and services in Europe".

Próxima actualización: En octubre de este 2023, las WCAG 2.2., ya pasan a ser recomendación. Se está ya trabajando en el "working draft" de las WCAG 3.0.

## Propósito

El objetivo de la normativa de accesibilidad es dar a los diferentes participantes involucrados en el desarrollo, integración o mantenimiento de software de la uoc, unas líneas maestras en las cuales basar la accesibilidad de todas las interfficies web y aplicaciones móviles. Todos alineados con las buenas prácticas reconocidas internacionalmente pero adaptadas a las necesidades y especificidades del Área de Tecnología de la UOC.

## Audiencia

Esta normativa está dirigida a todos aquellos proveedores del Área de Tecnología de la UOC que estén ofreciendo sus servicios actualmente o lo quieran hacer en un futuro y evidentemente, para los miembros del equipo propio del Área de Tecnología UOC.

La Normativa se ha escrito para personas familiarizadas con esta disciplina. Para adquirir conocimientos complementarios con esta disciplina se ha anexado documentación complementaria que añade bibliografía para ampliar conocimientos.

## Modelo de gestión

El modelo de gestión de la accesibilidad se basa en

- Gobierno:
  - El eje estructural es disponer de una política de accesibilidad que será de obligado cumplimiento para todos nuestros proveedores.
  - La comunidad UX será la responsable de versionar esta política. La periodicidad de esta renovación se decidirá en la comunidad UX.
  - La comunidad UX será la responsable de comunicar la política a los proveedores y asegurar el entendimiento de la necesidad de cumplimiento a través de los comités de gobierno del UX que ya existen.
  - Definir mecanismos que permitan evaluar el nivel de cumplimiento de la política de accesibilidad.
- Ejecución:
  - Serán los proveedores los responsables de ejecutar la política de accesibilidad y presentar las correspondientes evidencias.
  - Los proveedores tienen que asegurar la accesibilidad de sus entregables y ejecuciones.
  - Por lo tanto los proveedores tienen que tener el conocimiento necesario para aplicar correctamente la política de accesibilidad, garantizar su cumplimiento y dar soluciones a problemáticas aparecidas.
- Verificación del cumplimiento:
  - El proveedor tiene que asegurar el cumplimiento de la política de accesibilidad en los proyectos a través de los mecanismos definidos en el ámbito del gobierno
  - Establecer y mantener los elementos que permitan operativizar la verificación, como por ejemplo
    - Un certificado de accesibilidad
    - Un Xrail ejecutado con éxito
    - Una verificación automática con una herramienta especializada (habrá que indicar qué)
    - Una auditoría externa que realiza el propio proveedor (si se los hemos indicado en el encargo)
    - Una verificación externa encargada desde la UOC
    - Etc.

## Actividades

La Normativa de accesibilidad la conforman las actividades que se describen a continuación:

- Librar la propuesta de Solución
- Integrar accesibilidad en la fase diseño
- Ejecutar el heuristic de accesibilidad
- Definir la paleta de colores
- Definir estrategia de pruebas de Accesibilidad
- Ejecutar la test de Accesibilidad al TestRail
- creación informe resultado pruebas accesibilidad
- Declaración de accesibilidad

## **Librar la propuesta de solución**

Hay que tener en mente estos puntos a la hora de hacer la propuesta de solución:

- Tiene que considerarse durante la planificación hacer el contenido accesible y asegurarse que lo se, no como una etapa posterior: La accesibilidad no es un añadido. Hay que tenerla en cuenta desde el principio del proyecto.
- La accesibilidad no implica más coste de desarrollo. "Cuesta el mismo añadir una imagen de forma accesible que inaccesible. Cuesta el mismo hacer un formulario accesible que inaccesible"
- El buen desarrollo web empieza con la accesibilidad. Cada línea de código escrita sin pensar en ella se una línea que tendrá que ser reescrita más tarde: ES MUY COSTOSO ARREGLAR LOS ERRORES DESPUÉS. "Si basura un error, aquella línea de código habrá que reescribirla. Y en la mayoría de casos implica reescribir de nuevo la funcionalidad con problemas de accesibilidad".
- No es aceptable ningún error de accesibilidad en herramientas y espacios web que se hayan creado de nuevo. Cualquier error no solucionado tendrá que ir acompañado de su justificación para poder ser aceptado.
- En caso de tener que utilizar una herramienta de un tercero o servicio externo por un proyecto, ya sea en código abierto o propietario, habrá que escoger la más accesible.
- En la plantilla de checklist por Aceptación de la propuesta de solución hay un apartado de accesibilidad donde se tiene que recoger como se preten lograr esta accesibilidad.

A la hora de escribir los requerimientos hay que tener en cuenta las siguientes recomendaciones:

- [Accesible Rich Internet Applications \(WAI-ARIA\) 1.2 \(W3C Recommendation 6 jun 2023\)](#)
- [WAI-ARIA Authoring Practices](#)
- [Requerimientos de los dialegs modales](#)
- [Requerimientos de los formularios](#)
- [Requerimientos de las apps con angular](#)

- [Navegación con teclado y lector de pantalla](#)

## Integrar accesibilidad en la fase diseño

Se propone seguir la metodología siguiente por aconsseguir una buena integración accesibilidad. Estos pasos se tendrían que aplicar a la hora de crear los wireframes y diseños gráficos para poder generar una maqueta con un buen código html.

## Ejecutar el heurístico de accesibilidad

Cómo hemos indicado anteriormente la accesibilidad hay que tenerla en mente durante todas las fases del desarrollo de una aplicación. Cuanto más cerca de la posta en producción se detecta un error de accesibilidad más es el coste de solucionar este error.

Por esta razón, se recomienda (en este momento no se normativo) pasar el heurístico de accesibilidad en el entregable wireframe.

## Definir la paleta de colores

Dentro de la fase de diseño del proyecto hay que validar que todas las combinaciones de colores que se usan cumplen que el texto, o las imágenes de texto, tienen que tener una relación de contraste de al menos 4.5:1 . En el caso de texto grande o en negrita sería de 3:1.

Habrá que entregar, con el diseño gráfico, una lista de las combinaciones de colores empleadas y su valor de contraste para validar que son correctos. Se puede emplear entre otros los siguientes recursos:

- [Paleta de colores accesible](#)
- [Color Contraste Checker](#)
- [Otros recogidos al apartado recursos de accesibilidad](#)

## Definir estrategia de pruebas de Accesibilidad

Dentro del entregable de Estrategia de Pruebas del proyecto es necesario contemplar las pruebas de Accesibilidad para cada una de las aplicaciones implicadas en el proyecto.

En especial hay que tener en consideración el entregable Plano de Pruebas de Accesibilidad.

## Ejecutar la test de Accesibilidad a XRAY

Como ya hemos dicho, toda aplicación tiene que ser accesible a nivel doble A.

Se ha diseñado un Plan de pruebas de Accesibilidad a XRAY de la UOC. Cada una de las aplicaciones implicadas en un proyecto tienen que superar la test para que sea aceptada la entrega del proyecto.

Justo es decir que la Test Plan de Accesibilidad definido a XRAY, no contempla todas las validaciones que serían necesarias para garantizar doble A. Acontece una compilación que nos parece esencial que todo proyecto cumpla. Por lo tanto, esto quiere decir que hay que cumplir con todas las validaciones propuestas por la WAI, no solo las recogidas a la Test Plan.

El primer paso de la test suite consiste al hacer una test automática. Hay herramientas automáticas que nos pueden ayudar en esta tarea pero, hay que tener en cuenta que no son totalmente fiables. Con una revisión automática obtendremos un nivel aproximado de revisión de accesibilidad del 32 % (ver what we found when we tested tools donde the worlds least accesible webpage), por el que se recomienda una combinación de test manual y herramientas automáticas. El W3C tiene una llanura donde recoge las diferentes herramientas para hacer una test automática.

### **Creación informe resultado pruebas accesibilidad**

Cuando pasamos la test de accesibilidad ya sea vía Pla de pruebas de accesibilidad a confluence o siguiendo nuestra metodología (del proveedor) iremos recogiendo por cada criterio si pasa, falla o no aplica. Además de ejemplos donde la test falla (ver Generación informe cumplimiento accesibilidad para ver una lista de herramientas para recoger los datos de pasar la test).

Toda esta información (si pasa o no el criterio y los comentarios con las evidencias de los errores detectados) se tienen que recoger en un informe de Resultado de Pruebas de accesibilidad.

Si se detectan errores se tendrían que solucionar para poder cumplir el primer requerimiento de la normativa que la herramienta tiene que ser accesible. No tendría que pasar a mantenimiento ninguna herramienta si avance no se han solucionado los errores de accesibilidad detectados en la auditoría.

### **Declaración de accesibilidad**

Cuando la herramienta se dé como correcto desde el punto de vista de la accesibilidad se hace uso del informe de resultados de pruebas de accesibilidad para poder llenar la "declaración de accesibilidad"

Esta declaración de accesibilidad se tendría que mostrar en el enlace "accesibilidad" que hay que añadir en el pie de toda aplicación web. Si se trata de una aplicación móvil si tiene que poder acceder desde la app.

### **Herramientas y documentos**

El proveedor tiene que librar conforme ha pasado el plan de pruebas de accesibilidad a XRAY sobre los siguientes entregables:

- Frente end
- Producto en en torno preproducción

## Anexo - Normativa DevOps

A continuación se muestra la normativa vigente en el momento de redacción del pliego. Esta Normativa se revisa 2 golpes al año a la Release Normativa. Hay que tener en cuenta que se tiene que cumplir la normativa vigente en cada momento.

### Introducción

La Normativa DEVOPS describe como Construir y Desplegar Aplicaciones en la UOC

### Alcance

La Normativa es aplicable a todo el ciclo de vida de un proyecto o aplicación y durante todo el periodo que dura el contrato con un proveedor.

### Propósito

EL OBJETIVO DE ESTE DOCUMENTO ES DAR A CONOCER A LOS COLABORADORES DEL ÁREA DE TECNOLOGÍA LA NORMATIVA QUE APLICA A LOS PROYECTOS EN TÉRMINOS DE CONSTRUCCIÓN Y DESPLIEGUE DE APLICACIONES CON EL OBJETIVO DE UNIFICAR LA FORMA DE TRABAJAR DE TODOS LOS PROVEEDORES.

### Audiencia

Este Normativo está dirigido a todos aquellos proveedores del Área de Tecnología de la UOC que estén ofreciendo sus servicios actualmente o lo quieran hacer en un futuro y evidentemente, para los miembros del equipo propio del Área de Tecnología UOC.

### Aplicaciones Cloud Ready

Hay que seguir estas indicaciones normativas sobre construcción de aplicaciones "cloud-ready" para poder asegurar que en el momento de querer desplegar la aplicación por el Servicio de CICD de la UOC, será compatible. Los nuevos sistemas tendrán que ir al Cloud y ser desplegados vía el CICD - Continous Integrations and Continous deployment, esto implica que se tienen que construir siguiendo una serie de premises:

## Desarrollo

- Desarrollar siguiendo se los Principios Tecnológicos y la Hoja de ruta de Tecnología UOC.
- El código fuente de las aplicaciones tienen que estar adecuadamente "componetitzades". Esto quiere decir que las aplicaciones se tienen que estructurar en diferentes Componentes Técnicos.
- Evitar almacenar el estado de la sesión a la aplicación, tienen que ser stateless.
- Evitar dependencias con el sistema operativo y la infraestructura subyacente mejorando su portabilidad.
- Los logs se tienen que escribir a salida estándares, no a fichero. La plataforma Cloud donde se despliega se encargará de recolectarlos por su visualización.
- Utilizar protocolos resilientes (http, ssl, standard database, queuing).
- A la fase de diseño técnico de la aplicación, y como complemento del Documento de Arquitectura, el integrador tendrá que llenar una Receta Cloud donde se certificarán estos puntos.
- En el caso de aplicaciones web, exponen su funcionalidad vía HTTP sin necesidad otras runtimes (aplicaciones autocontingudes).

## Código Font

- El Código Font se guarda al GITLAB de la UOC y en caso de que este código fuente, sean artefactos, al ARTIFACTORY
- El código fuente hay que versionarlo siguiendo la Normativa de Versionado de Código Font y Ramas.

## Construcción

- Se construyen una única vez y se promocionan entre entornos. Esto implica que cualquier configuración tiene que realizarse mediante variables de entorno, no "harcoded" al código de la aplicación
- No tienen que verse afectados en caso de escalado dinámico. Esto quiere decir que se tienen que poder añadir/sacar nodos de ejecución sin que el comportamiento de la aplicación se vea alterado.
- Su instalación tiene que estar automatizada, siguiendo CICD (Continous Integrations and Continous deployment) no puede requerir tareas manuales.
- Tienen que poder ser monitorizadas. Por lo tanto, hace falta que expongan endpoints de

healthcheck que permitan testar su correcto funcionamiento.

### Calidad

- Para algunas tecnologías como Java, se ejecutan Pruebas Unitarias y hace falta que la cobertura de código llegue al % establecido por el indicador Quality Gate
- A todo el código que pasa por el CICD - Continous Integrations and Continous deployment se le mide la Calidad. Tiene que cumplir con la normativa para el código fuente, " y se recoge a la Normativa de Calidad.
- Usar lints de SonarQube para liberar el código a los repositorios de la UOC con la calidad adecuada a los umbrales UOC. Ver SonarLint - Guía de instalación y configuración
- Documentación detallada de la aplicación (documento de arquitectura, documento funcional, manual de usuario, manual de administración, ...) Ver Normativa de Entregables
- Más información sobre la Normativa de Calidad

### Despliegue de la nueva Infraestructura

- Habiendo tenido en cuenta la Normativa, si ya se dispone del Documento de Arquitectura, y de la Receta Cloud, habrá que abrir una issue de tipo Primero Despliegue Infra Cloud al Jira de la aplicación.

### Incumplimiento de la Normativa

El incumplimiento de esta Normativa NO permitirá desplegar las aplicaciones. Además, generará No conformidad en los proyectos, las aplicaciones, los servicios o los programas correspondientes.

### Control de aplicación de la Normativa

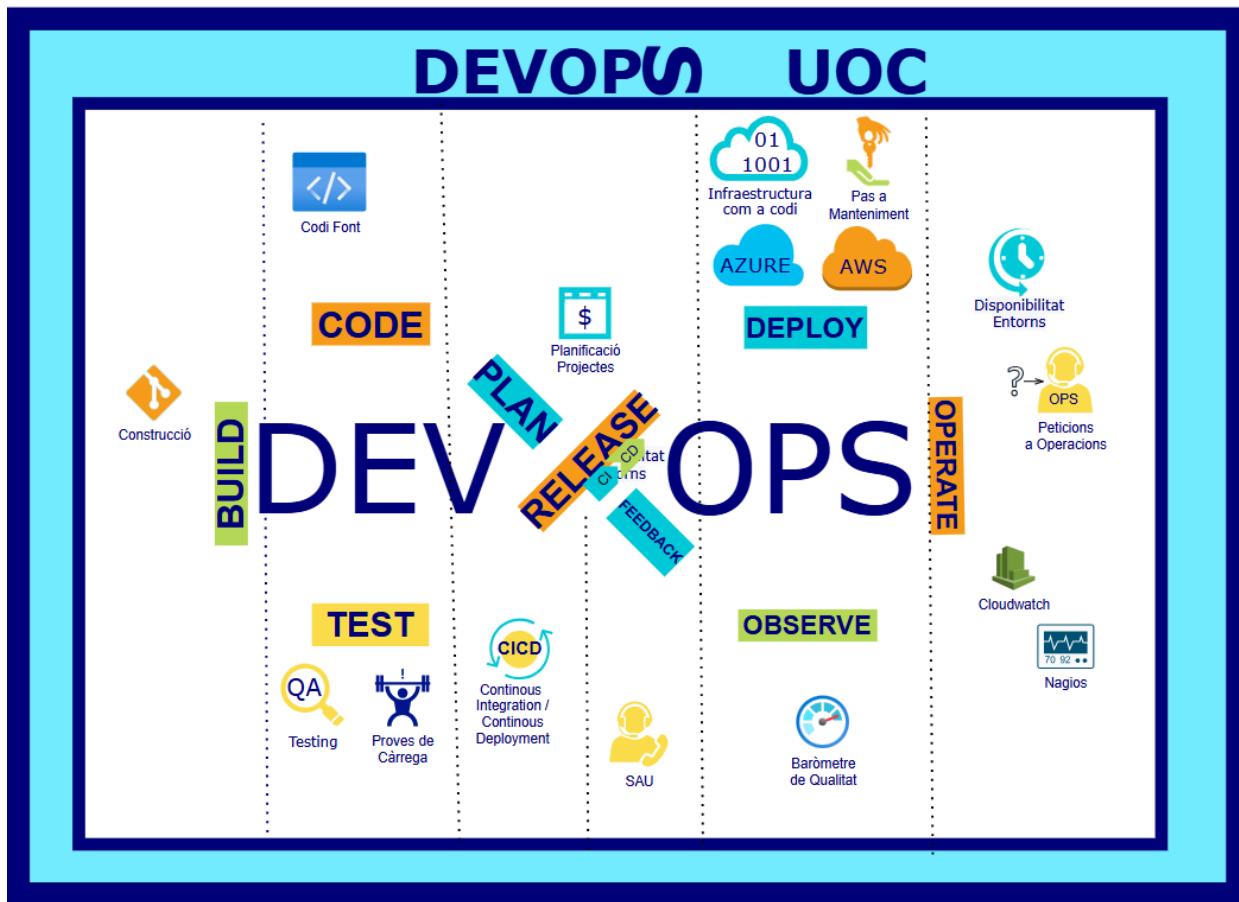
Principalmente será el responsable de Operaciones o el responsable de Arquitectura quien velará por el cumplimiento de la Normativa. Adicionalmente, el equipo de calidad también podrá hacer comprobaciones y auditorías.

### Funcionamiento del CI/CD

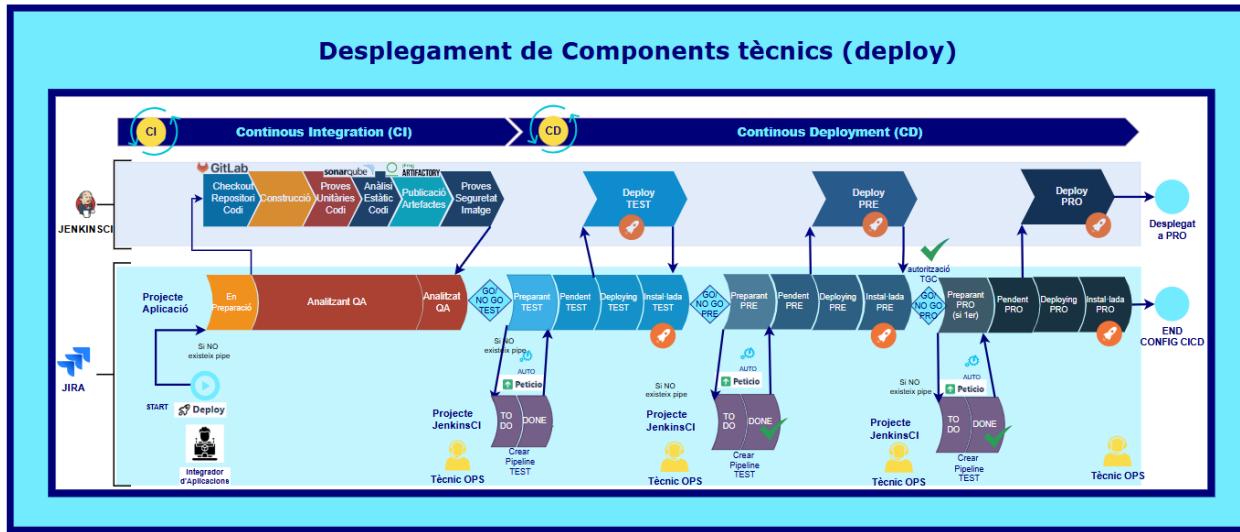
Todas las aplicaciones que se desplieguen en plataformas Cloud de la UOC tienen que tener automatizado su ciclo de vida. Esto quiere decir que no se realizará ninguna acción manual en el proceso de integración continua y despliegue.

Por toda aplicación que tenga que ser integrada al CI/CD de la UOC hará falta antes (en el contexto de su proyecto) haber llenado la Receta Cloud, en concreto la sección CI/CD, proporcionando el detalle necesario de sus componentes técnicos.

Este es el ecosistema de elementos que conforman el CI/CD de la UOC:



Y la pipeline de CI/CD:



## Descripción

Dentro de DEVOPS, El sistema de Continuous Integration y Continuous deployment (CICD: Integración Continua y Despliegue Continuo) es el responsable de la automatización del ciclo de vida de las aplicaciones a la UOC para liberar versiones de código (Release)

- El CICD permite automatizar el ciclo de vida de las aplicaciones.
- Jira y Jenkins son los principales actores para los despliegues.
- A partir de un job único, se ejecutan las pasos necesarias para desplegar las aplicaciones a los entornos de producción.
- El mecanismo CICD se encargado de:
  - Construir los artefactos a partir del código fuente y librerías repositadas a la Artifactory, ejecutar los controles de calidad, seguridad y las pruebas necesarias para asegurarse que el código que se construye ofrece las garantías exigidas por la universidad
  - Ofrecer las herramientas para desplegar estos artefactos a las diferentes infraestructuras y entornos cloud, en todos los casos mediante siempre el uso de la automatización.

## Funcionamiento General

- Por cada componente técnico de la aplicación existe una pipeline de CI/CD al Jenkins. Esta pipeline se ejecuta cuando se invoca desde el workflow de JIRA de Deploy. El/s stage/s que se ejecutan vienen determinados por el paso dentro del workflow (Ej. "Desplegar a Test")
- El componente técnico se construye una única vez dentro del workflow de Deploy de JIRA. Esto quiere decir que su construcción tiene que ser independiente de la en torno a despliegue. La

configuración dependiente del entorno se carga en tiempo de despliegue o ejecución a partir de variables de entorno, no en tiempo de construcción. El artefacto generado se promociona entre entornos, tiene que ser inmutable. Este punto es muy importante, puesto que el componente no puede ser integrado dentro del CI/CD si no cumple este requisito

- A pesar de que por algunas tecnologías se ejecutan macetas unitarias en la actualidad, como por ejemplo JUnit por Java, en caso de error no se para el proceso de CICD. Pronto no será así, para todas las tecnologías se exigirán macetas unitarias con un cierto grado de cobertura, así como su correcta ejecución para poder desplegar a los diferentes entornos
- También se da la posibilidad de no pasar por ciertos stages (Ej. Deploy Test, ...) de la pipeline de CICD, siempre quedando registrado y apareciendo a los indicadores de DevOps. Por casos excepcionales, bugfixes urgentes que tengan que llegar a producción lo más rápido posible, puede ser necesario. Estas excepciones se tienen que solicitar explícitamente al equipo de CICD puesto que requieren de cierta configuración adicional. Ver apartado "Peticiones al servicio"

### Componentes del Sistema CICD

Componentes del Sistema CICD	Breve descripción	Más Info
<b>JIRA</b>	<p>El Jira es la herramienta de ticketing que se usa en el Área de Tecnología para las gestiones internas tanto de ejecución de proyectos, mantenimiento de aplicaciones, incidencias...</p> <p>En este servicio el Jira hace la función de orquestador de los despliegues a los diferentes entornos y dependiendo de las issue que usamos para desplegar (Ver <a href="#">Tipo de Issues de Despliegue</a> ) Se activarán un stages u otros de la pipe.</p>	<a href="#">Ver detalle sobre JIRA</a>
<b>Jenkins (JENKINSCI)</b>	<p>El Jenkins (JenkinsCI) es el orquestador del servicio de integración continua (CICD), se crea una pipeline por cada componente técnico que se gestionado por el servicio.</p> <p>Jenkins trabaja con una arquitectura de estrella, está formado por una instancia master y una serie</p>	<a href="#">Ver detalle sobre JENKINSCI</a>

	<p>de instances slave que cumplen diferentes funciones dentro del servicio de CICD. Los Slaves se pueden dividir en dos grandes categorías, agentes de construcción o agentes de despliegue.</p>	
<b>Gitlab</b>	<p><a href="#">Gitlab.uoc.es</a> es el Repositorio oficial de la UOC, donde se custodia el código de las aplicaciones. No se trata solo de un repositorio estático, sinò que ofrece otras funcionalidades, como desarrollo de software colaborativo basado en Git, además de un gestor de ramas por lo tanto todas las aplicaciones de la UOC tienen que tener su código fuente en un repositorio de la herramienta.</p>	Ver detalle sobre <a href="#">GITLAB</a>
<b>JFrog Artifactory</b>	<p>El Artifactory es repositorio de artefactos de la UOC. ES LA HERRAMIENTA DONDE SE CUSTODIAN LOS ARTEFACTOS DE LAS APLICACIONES DE LA UOC.</p> <ul style="list-style-type: none"><li>• ES EL ADMINISTRADOR CENTRAL DE BIBLIOTECAS QUE FACILITA LA COLABORACIÓN EFICIENTE ENTRE LOS DIFERENTES COLABORADORES Y EQUIPOS IMPLICADOS Y OFRECE UN REPOSITORIO UTILIZADO POR LA HERRAMIENTA DE CICD PARA RESOLVER LAS DEPENDENCIAS EN LA CONSTRUCCIÓN DE LAS APLICACIONES.</li><li>• Repositorio de artefactos generados por la herramienta de CICD, que sueño utilizados para desplegar a los diferentes entornos de ejecución.</li><li>• Ein por la custodia de dependencias</li><li>• Custodia de librerías de terceros</li><li>• Versionado de librerías comunes</li></ul>	Ver detalle sobre <a href="#">ARTIFACTORY</a>

<b>SonarQube</b>	<p>SonarQube (conocido anteriormente como Sonar) es una plataforma para evaluar código fuente. ES SOFTWARE LIBRE Y USA VARIAS HERRAMIENTAS DE ANÁLISIS ESTÁTICO DE CÓDIGO FUENTE COMO CHECKSTYLE, PMD O FINDBUGS PARA OBTENER MÉTRICAS QUE PUEDEN AYUDAR A MEJORAR LA CALIDAD DEL CÓDIGO DE UNA APLICACIÓN.</p> <p>SonarQube devuelve métricas y define un umbral (Quality Gate) que permiten evaluar la calidad del código desarrollado y tomar decisiones según el resultado obtenido.</p> <p><b>Guías principales</b></p> <p><a href="#">SONAR_ Uso del archivo sonar-project.properties</a></p> <p><a href="#">SonarIDE (sonarlint) - Guía de instalación y configuración</a></p>	Ver detalles sobre <a href="#">SonarQube</a>
<b>XRAY</b>	XRAY es una solución de gestión de casos de prueba para aseguramiento de la calidad (QA) y equipos de desarrollo, que está diseñada para ayudar a los equipos a organizar, gestionar y hacer seguimiento del proceso de pruebas.	
<b>Owasp ZAP</b>	Análisis de seguridad, encargado de analizar el código de las aplicaciones para buscar vulnerabilidades de seguridad.	
<b>Anchore-engine</b>	Componente que se encarga de hacer checks de seguridad de los contenedores y reportar si encuentra bugs o vulnerabilidades en las imágenes generadas desde el build.	

# Anexo - Normativa de Seguridad

A continuación se muestra la normativa vigente en el momento de redacción del pliego. Esta Normativa se revisa 2 golpes al año a la Release Normativa. Hay que tener en cuenta que se tiene que cumplir la normativa vigente en cada momento.

## Plataformas SaaS

Para las plataformas SaaS se definen los criterios siguientes:

### Ámbito Seguridad:

- Certificación ISO 27001 o ENTE por el SaaS
- Garantiza la posibilidad de configuración de roles y permisos en función de usuarios y tareas.
- Registra los datos de cada intento de acceso, con información relativa a la identificación única de usuario, tipo de acceso, fecha y hora, información accedida y las conserva un mínimo de 12 meses
- Permite tener un registro actualizado de usuarios que refleje la asociación de cada código de usuario con la persona que lo tiene asignado, su perfil y los accesos autorizados. Este registro tiene que reflejar todos los cambios en el mapeo: altas, bajas y posibles modificaciones.
- La solución tiene que disponer de mecanismos necesarios que permitan registrar la actividad de los usuarios y custodiar estos registros durante un mínimo de 12 meses
- La solución tiene que tener mecanismos de logging y de control de cambio en los datos, registrando toda la información necesaria que permita la trazabilidad de los sucesos.
- La solución puede procesar inmediatamente las bajas de los usuarios mediante las herramientas de administración de las aplicaciones, inhabilitando el acceso a las mismas. La baja de un usuario implica su bloqueo temporal, antes de proceder a su eliminación definitiva.
- La solución tiene que garantizar que toda la información viaja y se almacena cifrada (con clave mínima 256 bits)
- Dispone de medidas de seguridad adecuadas en todos los entornos para garantizar la confidencialidad, integridad y disponibilidad de los datos (y.e.: Firewall, Sistemas de Detección y Prevención de Intrusos (IDS/IDPS), Zona Desmilitarizada (DMZ), Redes Privadas Virtuales (VPN) y Proxy a la infraestructura.
- Disponen de los logs o registros de trazabilidad suficientes para analizar una intrusión o fuga de información en caso necesario.
- El prestamista del servicio tiene que actuar proactivamente instalando las actualizaciones y parches de seguridad publicados por los correspondientes fabricantes. Tiene que tener una política de vigilancia de alertas de seguridad y detección de vulnerabilidades.
- El Proveedor tiene que realizar todas aquellas auditorías legalmente exigibles, tanto de manera interna como externa, sobre aquellos sistemas involucrados en el servicio prestado a la UOC, dejando a disposición de la UOC los informes de auditoría generados.
- Posibilidad de definir IP Blacklist

- Caducidad de sesión en caso de inactividad (configurable)
- Disponer de herramientas de revisión de análisis forense
- Dispone de un sistema de mitigación de riesgos e impactos por ataques DDOS
- Posibilidad de integrar doble factor de autenticación en diferentes roles de la herramienta, haciendo énfasis en los administradores
- La solución tendrá que garantizar que se emplearán mecanismos de eliminación segura de información. Estos incluyen los casos de reciclaje de apoyos y de finalización del Servicio.

**Ámbito RGPD:**

- Los datos tienen que estar almacenadas en una ubicación de la UE
- Se tratan datos de categorías especiales?
- Permite definir criterios de conservación y eliminación periódica de datos
- Da cumplimiento a toda la normativa de protección de datos personal RGPD, LOPDGDD
- Dispone de un sistema de atención al ejercicio de derechos en materia de protección de datos
- Dispone de Delegado de Protección de Datos
- Dispone de un protocolo ante incidentes y comunicación de brechas de seguridad
- Todos los subcontratistas que utiliza el SaaS se ubican en la UE (incluye los servicios cloud)

**Desarrollos y configuraciones**

Por los desarrollos y configuraciones hay que aplicar como mínimo los criterios siguientes. Estos criterios se tienen que tener en cuenta a la hora de elaborar cualquier de los artefactos del proyecto, en caso contrario, generarán penalidades.

Seguridad Arquitectural
La aplicación garantiza que todos los componentes que están presentes a la misma están identificados (bien sea un componente individual o bien sean grupos de archivos de código fuente, librerías y/o ejecutables)
La aplicación garantiza que todos los componentes que no forman parte de la misma aplicación, pero en los que esta confía, están identificados
Existe una arquitectura de alto nivel para la aplicación
Autenticación
La aplicación garantiza que todos los campos de contraseña tienen el "eco" desactivado y que los campos de contraseña o los formularios que contienen estos campos tienen el atributo "de autocompletat" desactivado

**Seguridad Arquitectural**

La aplicación garantiza el control en si se ha excedido el número máximo de intentos durante el proceso de autenticación, la cuenta quedará bloqueado durante un periodo basta largo de tiempo que impida los ataques a la fuerza sucia

La aplicación garantiza que todos los controles de autenticación se aplican a la banda servidora

La aplicación garantiza que todos los controles de autenticación (incluidas las librerías que gritan a servicios de autenticación externos) tienen una aplicación centralizada

La aplicación garantiza que todos los controles de autenticación fallan de forma segura

La aplicación garantiza que la robustez de las credenciales de autenticación son suficientes para resistir los ataques típicos

La aplicación garantiza que todas las funciones de gestión de las cuentas de usuarios son al menos tan resistentes a un ataque (de fuerza bruta o de ingeniería inversa) como el mecanismo de autenticación principal

La aplicación garantiza que los usuarios pueden cambiar de forma segura sus credenciales y con algún mecanismo que sea, al menos, tanto resistente como el mecanismo de autenticación principal

La aplicación garantiza que hace falta una re-autenticación antes de permitir operaciones "realmente" sensibles

La aplicación garantiza que todas las decisiones (tanto el que hace a los éxitos como a los fracasos) de autenticación se están grabando

La aplicación garantiza que que todas las contraseñas de las cuentas de usuario se "saltan" con una "sal" (en criptografía, la sal compran bits aleatorios que son usados como una de las entradas en una función derivadora de claves) exclusiva de esta cuenta (pex. el ID de usuario, la fecha de creación de la cuenta, ... etc) y que se almacena un hash y no la contraseña en claro

La aplicación garantiza que todas las credenciales de autenticación para acceder a servicios externos (accesos a la DB, WX, ... ect) están cifradas y que se almacenan en un lugar protegido (y no en el código fuente o ficheros accesibles de configuración)

**Gestión de la sesión**

La aplicación utiliza algún "framework" de gestión de las sesiones

La aplicación garantiza que las sesiones se invalidan cuando el usuario cierra la sesión

La aplicación garantiza que las sesiones caducan (expiran) después de un periodo de inactividad

**Seguridad Arquitectural**

La aplicación garantiza que todas las páginas que necesitan autenticación para acceder a ellas tienen vínculos (anchors, buttons, imágenes, ... etc) de cierre de sesión

La aplicación garantiza que el identificador de la sesión no se "divulga o publica", particularmente en las direcciones de las URL y mensajes de error. Esto incluye garantizar que la aplicación no admite la reescritura de la URL ni las cookies de sesión

La aplicación garantiza que el identificador de sesión se cambia al inicio de la sesión

La aplicación garantiza que el identificador de sesión se cambia cuando se hace una re-autenticación

La aplicación garantiza que el identificador de sesión se cambia o se borra al salir de la aplicación o servicio

La aplicación garantiza que solo los identificadores de sesión generados por el "framework" de la aplicación son reconocidos como válidos por la aplicación

**Control de acceso**

La aplicación garantiza que los usuarios solo pueden acceder a las funciones protegidas para las que tengan una autorización específica

La aplicación garantiza que los usuarios solo pueden acceder a las direcciones URL para las que tengan una autorización específica

La aplicación garantiza que los usuarios solo pueden acceder a los archivos de datos por los que tengan una autorización específica

La aplicación garantiza que las referencias directas a objetos están protegidas, de tal manera que un usuario solo tiene acceso a los objetos a los que ha sido autorizado

La aplicación garantiza que está deshabilitada la vista de contenido de los directorios del servidor Web, exceptuando que se quiera hacer de forma deliberada

La aplicación garantiza que los usuarios solo pueden acceder a los servicios para los que tienen una autorización específica

La aplicación garantiza que los usuarios solo pueden acceder a los datos para las que tienen una autorización específica

La aplicación garantiza que el control de acceso falla de forma segura

La aplicación garantiza que se aplican las mismas reglas de acceso a la capa de presentación que en la banda del servidor

**Seguridad Arquitectural**

La aplicación garantiza que todos los atributos de usuario y de datos y que la política de información usados en el control de accesos no pueden ser manipulados por usuarios finales sin una autorización expresa

La aplicación garantiza que todos los controles de acceso se aplican en el servidor

La aplicación garantiza que hay un mecanismo centralizado para proteger el acceso a cada tipo de recurso protegido (incluidas las librerías que necesitan de autorización de servicios externos)

La aplicación garantiza que las limitaciones impuestas por "las reglas de negocio", tanto en cuanto a la entrada de datos como en el acceso a la aplicación (pex. límites diarios de transacción) no pueden ser anuladas o ignoradas

La aplicación garantiza que todas las decisiones de control de acceso se pueden grabar y que todos los fallos en decisiones de control se están grabando

**Validación de la entrada de datos**

La aplicación garantiza que en torno a ejecución del software no se susceptible a desbordamientos de buffer. O que en su defecto, los controles de seguridad previenen de los desbordamientos de buffer

La aplicación garantiza que se aplica un patrón de validación positiva para todas las entradas de datos de la aplicación

La aplicación garantiza que todos los fallos de validaciones de entrada dan como resultado el rechazo de la entrada o el "saneamiento" de la misma

La aplicación garantiza que se especifica una conjunto de caracteres, como puede ser UTF8, para todas las posibles fuentes de entrada de datos

La aplicación garantiza que la validación de todas las entradas de datos se realizan a la parte servidora

**Codificación y escape de los caracteres de salida**

La aplicación garantiza que todos los datos emitidos por la salida HTML y que no son de confianza están debidamente escapados (incluyendo los elementos HTML, los atributos, valor de los datos Javascript, bloques CSS y atributos URL)

La aplicación garantiza que todos los controles de codificación y escape de los caracteres de salida están implementado a la parte servidora

La aplicación garantiza que todos los controles de codificación y escape de los caracteres de salida codifican estos caracteres como "no seguros" para el lenguaje intérprete (HTML, Javascript, CSS,

**Seguridad Arquitectural**

XML, JSON, ... etc)

La aplicación garantiza que todos los datos que "no son de confianza" emitidas para ser usadas por un intérprete SQL, utilizan interfaces parametrizadas, del tipo "prepared statement", o bien están correctamente escapados

La aplicación garantiza que todos los datos que no son de confianza emitidas para ser usadas con un intérprete XML utilizan interfaces con parámetros o caracteres de escape apropiados

La aplicación garantiza que todos los datos que no sean de confianza utilizados en consultas LDAP están escapados correctamente

La aplicación garantiza que todos los datos que no sean de confianza y que pueden ser parámetros de mandos del sistema operativo están escapados correctamente

La aplicación garantiza que todos los datos que no son de confianza para cualquier intérprete (de los no mencionados anteriormente) están escapados correctamente

**Criptografía y cifrado**

La aplicación garantiza que todas las funciones criptográficas y de cifrado usadas para proteger los "secretos" de la aplicación se ejecutan a la banda servidora

La aplicación garantiza que todos los módulos criptográficos fallan de forma segura

**Gestión de errores y gestión de logs**

La aplicación garantiza que no se generan mensajes de error ni trazas que contengan datos sensibles que un atacante podría usar (como por ejemplo la información personal)

La aplicación garantiza que todos los errores de la parte servidora están gestionados desde la misma parte servidora

La aplicación garantiza que todos los controles de registro están implementados en la parte del servidor

La aplicación garantiza que la lógica de controles de errores en los controles de seguridad deniega el acceso por defecto

La aplicación garantiza que el registro de seguridad ofrece la posibilidad de grabar tanto los acontecimientos de éxito como los de fracaso o error (entre la serie de acontecimientos identificados como relevantes)

**Seguridad Arquitectural**

La aplicación garantiza que cada acontecimiento de registro incluye: un sello de tiempo de una fuente confiable (pex. NTP), el nivel de gravedad del acontecimiento, una indicación de que se trata de un acontecimiento de seguridad pertinente (si se mezcla con otros registros), la identidad del usuario que ha ocasionado el acontecimiento (en el supuesto de que haya un usuario asociado al acontecimiento), la dirección IP del origen de la solicitud asociada al acontecimiento, el grado de éxito del acontecimiento y una descripción del acontecimiento

La aplicación garantiza que todos los acontecimientos que incluyen datos que no son de confianza no se ejecutará como "código" en el software de visualización de logs

La aplicación garantiza que los acontecimientos (registros de actividad, logs, ...) están protegidos contra el acceso no autorizado y la modificación

La aplicación garantiza que solo hay una única implementación de grabación de acontecimientos y que es la que usa la aplicación

La aplicación garantiza que no graba datos sensibles que pudieran ayudar o dar pistas a un atacante, como por ejemplo información personal o sensible del entorno o la aplicación

La aplicación garantiza que hay una herramienta disponible que permita al analista realizar búsquedas de registros de acontecimientos basado en la combinación de criterios de busca

**Protección de los datos**

Todo el sistema está construido bajo el principio de privacidad por defecto y desde el diseño para garantizar pleno cumplimiento de todas las disposiciones del RGPD y normativa adicional.

La aplicación garantiza que todos los formularios que contienen información sensible tienen deshabilitada la "cache" de información en la banda del navegador, o cliente, incluyendo las características de autocompletat

La aplicación garantiza que hay una lista identificada de datos confidenciales procesados por cada una de las aplicaciones, y que hay una política explícita de acceso a estos datos, y cuando estos datos tienen que estar cifradas (tanto en el almacenamiento como en el tráfico)

La aplicación garantiza que todos los datos confidenciales se envían al servidor en el cuerpo del mensaje o protocolo HTTP, se a decir, no se utilizan parámetros de la URL para transmitir datos

La aplicación garantiza que las copias en cache o temporales de datos confidenciales enviados a los clientes están protegidas contra el acceso no autorizado y que estas caches son porgades o invalidadas después de que el propio usuario haya accedido a datos sensibles (es decir: las cabeceras HTTP tienen establecidos los parámetros de No-cache y No-Store Cache-Control)

**Seguridad Arquitectural**

La aplicación garantiza que todas las copias en cache o temporales de los datos almacenados a la parte servidora están protegidas contra el acceso no autorizado y que estas cahes son porgades o invalidadas después de que el usuario haya accedido a datos sensibles

**Seguridad de las comunicaciones**

La aplicación garantiza que para cada certificado SSL se puede construir una ruta de verificación hacia una CA (Certification Authority) de confianza, y que cada certificado de servidor es válido

La aplicación garantiza que un fallo en la conexión SSL no compuerta que se establece una conexión NO segura

La aplicación garantiza que se utiliza SSL a todas las conexiones autenticadas o referidas a datos o funciones sensibles (tanto en cuanto a las conexiones externas como hacia las da back-end)

La aplicación garantiza que el back-end de conexiones SSL graba los fallos de conexión

La aplicación garantiza que los certificados de cliente utilizan anclajes ("anchors") de confianza (para construir la ruta hacia la CA) e información de revocación

La aplicación garantiza que todas las conexiones a sistemas externos que manipulan información o funciones sensibles están autenticadas

La aplicación garantiza que todas las conexiones a sistemas externos que manipulan funciones o información sensible utilizan una cuenta que ha sido creada para tener los privilegios mínimos para que la aplicación funcione correctamente

**Seguridad HTTP**

La aplicación garantiza que las redirecciones NO incluyen datos no validados

La aplicación garantiza que solo acepta un conjunto definido de métodos de solicitud HTTP, como pueden ser GET y TABLA

La aplicación garantiza que cada respuesta HTTP contiene un encabezamiento de tipo de contenido que especifica un conjunto de caracteres seguros (pex. UTF-8)

La aplicación garantiza que el indicador **HTTPOnly** se utiliza a todas las cookies que no precisan específicamente de acceso Javascript

La aplicación garantiza que el indicador "secure" se utiliza a todas las cookies que contienen información confidencial, incluyendo la cookie de sesión

La aplicación garantiza que los encabezamientos HTTP, tanto a las peticiones como las respuestas, contienen solo caracteres ASCII imprimibles

**Seguridad Arquitectural****Verificación de los requerimientos de Seguridad**

La aplicación garantiza que toda la información de configuración relevante para la seguridad, se almacena a lugares que están protegidos accesos no autorizados

La aplicación garantiza que se deniegan todos los accesos, si la aplicación no puede acceder a su información sobre la configuración de seguridad

Como la aplicación mujer cumplimiento del RGPD y a la LOPDGDD

**Protección de datos personales**

El adjudicatario tendrá que dar cumplimiento con la normativa aplicable en relación a la protección de datos de carácter personal, así como al que se establece al pliego de cláusulas particulares.

**Protección WAF (Web Application Firewall)**

Los nuevos desarrollos de aplicaciones web con exposición a internet tendrán que implementar la solución WAF de la UOC.

El adjudicatario tendrá que implementar la solución waf, siguiendo el procedimiento de la uoc y realizar las validaciones de seguridad necesarias para garantizar el correcto funcionamiento de la aplicación.

**Desarrollo seguro del software****1. Introducción**

Este capítulo establece el compromiso de acuerdo entre la UOC y el adjudicatario para maximizar la seguridad del software de la aplicación (de ahora en adelante la aplicación) de acuerdo con los siguientes términos.

**2. Filosofía**

Este anexo está destinado a clarificar los derechos y obligaciones relacionadas con la seguridad de las relaciones comerciales de las partes involucradas en el desarrollo de la aplicación. Al más alto nivel, estas partes acuerdan que:

- Las decisiones de seguridad estarán basadas en el riesgo: Las decisiones de seguridad se

tomarán conjuntamente entre UOC y el adjudicatario basándose en una alta comprensión del riesgo que puede tener la aplicación.

- Las actividades dedicadas a la seguridad estarán balanceadas: Los esfuerzos dedicados a la seguridad de la aplicación estarán fuertemente distribuidos durante todo el ciclo de vida de su desarrollo.
- Se integrarán las actividades dedicadas a la seguridad: Todas las actividades y la documentación generada aquí hará falta que se integre en el ciclo de vida de desarrollo del software de la aplicación y no podrán en ningún caso estar separadas del resto del proyecto. Nada en este anexo implica un desarrollo especial de software.
- Podrán aparecer vulnerabilidades: Este anexo asume que todo software puede tener errores y que algunos de estos pueden crear incidentes de seguridad. Tanto UOC como el adjudicatario se esforzarán al identificar las vulnerabilidades con la mayor celeridad posible durante el ciclo de vida de la aplicación.
- La seguridad de la información será completamente revelada: Toda información relacionada con la seguridad de la información de la aplicación, será compartida entre UOC y el adjudicatario de forma inmediata y completa. El proveedor especificará por escrito qué aspectos de la seguridad relacionada con el desarrollo piensa no cumplir.
- Solo se requiere aquella documentación que sea útil en aspectos de seguridad: La documentación de seguridad no tiene que ser extensa en exceso en orden a clarificar, describir el diseño de la seguridad, el riesgo, el análisis y los incidentes

### **3. Actividades del ciclo de vida de la aplicación**

- Comprensión del riesgo: UOC y el adjudicatario acuerdan trabajar de manera conjunta para comprender y documentar los riesgos que pueden afectar a la aplicación. Este esfuerzo se enfocará a identificar posibles vulnerabilidades que afecten a los activos y las funcionalidades de la aplicación. Habrá que considerar cada uno de los puntos, funcionalidades y casos de uso definidos en la parte de los requerimientos de la aplicación.
- Requerimientos: Basándose en el riesgo, UOC y el adjudicatario acuerdan trabajar de manera conjunta para definir unos requerimientos de seguridad como parte de las especificaciones del software a desarrollar. Cada uno de los puntos que se hayan enumerado en la sección de requerimientos hará falta que sean discutidos y evaluados conjuntamente por UOC y el adjudicatario. Estos requerimientos tendrán que ser satisfechos en su totalidad por la aplicación.
- Diseño: el adjudicatario acuerda proporcionar una documentación que explique de forma clara el diseño que se implementa para lograr los requerimientos de seguridad. este diseño explicará de forma clara si el apoyo a los requerimientos de seguridad provienen del software desarrollado a la aplicación, de software de terceras partes o de la plataforma en la que se implementa el desarrollo.
- Implementación: el adjudicatario acuerda proporcionar y seguir un conjunto de buenas prácticas de programación. estas guías indicarán como se tiene que formatear el código, como se tiene que estructurar y comentar. aun así el código podrá ser revisado por alguna persona responsable

de la ejecución del proyecto de la uoc que validará los requerimientos de seguridad y las guías de programación antes de que el código de la aplicación sea considerado preparado por test.

- Seguridad del análisis y pruebas: el adjudicatario acuerda proporcionar y seguir un plan de pruebas de seguridad el cual definirá la aproximación que se haya hecho a cada uno de los requerimientos de seguridad. El nivel del rigor de esta actividad tiene que estar considerada y detallada en el plan del proyecto. El adjudicatario ejecutará el plan de pruebas de seguridad y proporcionará los resultados a la uoc.
- Despliegue seguro: el adjudicatario acuerda proporcionar las guías para configurar la aplicación de manera segura, muy documentadas y de tal manera que cubran todos los requerimientos especificados en el plan de proyecto. El adjudicatario incluirá en esta guía una completa descripción de las dependencias de la plataforma soportada, incluyendo el sistema operativo, la base de datos, el servidor web y el servidor de aplicaciones y como estos tendrán que ser configurados para cumplir con los requerimientos de seguridad. El adjudicatario acuerda que la configuración por defecto de la aplicación que librará a la uoc será segura.

#### 4. Áreas de seguridad

el adjudicatario acuerda que las siguientes áreas serán consideradas durante la comprensión del riesgo y las actividades de definición de los requerimientos de la aplicación.

- Validación y codificación: Todo el conjunto de requerimientos que especifican las reglas por canonicalizar, validar y codificar cada entrada de la aplicación, tanto desde los usuarios, sistema de ficheros, directorios o sistemas externos. La regla por defecto será que todas las entradas son inválidas a menos que cumplan una determinada especificación que lo permita. Adicionalmente, los requerimientos de la aplicación especificarán la acción a emprender cuando se reciba una entrada no válida. Específicamente, la aplicación no será susceptible a inyección, desbordamiento (overflow), manipulación (tampering) u otros ataques de corrupción de entrada de datos.
- Autenticación y gestión de la sesión: Los requerimientos de la aplicación especificarán como las credenciales de autenticación y los identificadores de sesión sueño protegidos en todo su ciclo de vida. Se incluirán los requerimientos referidos en el encabezamiento de este párrafo de todas las funciones relacionadas, incluyendo la pérdida de contraseña, los cambios de contraseña, los recordatorios de contraseña, la desconexión por inactividad y la posibilidad del uso de múltiples conexiones simultáneas (multiple login).
- Control de acceso: Los requerimientos de la aplicación incluirán una descripción detallada de todos los roles y perfiles usados (grupos, privilegios, autorizaciones) en la aplicación. Los requerimientos habrán también de indicar todo el conjunto de activos y funcionalidades proporcionadas por la aplicación. Los requerimientos tendrán que explicar completa y exactamente los derechos de acceso a cada uno de los activos y funciones de la aplicación por cada uno de los roles. Se sugiere construir una matriz de control para explicar el formato de estas reglas.

- **Manipulación y gestión de errores:** Los requerimientos de la aplicación hará falta que detallen como se gestionan los errores de la aplicación durante el proceso de funcionamiento de la misma. Se entiende que en algunas funcionalidades se tendrá más cura en el tratamiento de errores, sobre todo las que tengan más incidencia en el usuario final, mientras que otros puede ser que acaben en la finalización del proceso inmediatamente.
- **Registro (Logging):** Dentro de los requerimientos hará falta que se especifiquen qué acontecimientos son relevantes por la seguridad y las conexiones de la aplicación, como por ejemplo los ataques detectados, los intentos fallados de acceso y los intentos de ganar privilegios. Los requerimientos también especificarán qué tipo de información queda grabada para cada uno de los acontecimientos generados, incluyendo la fecha, la descripción del acontecimiento, detalles de la aplicación y de otra información forense de utilidad.
- **Conexión a sistemas externos:** Los requerimientos especificarán como se trata la autenticación hacia sistemas externos a la aplicación, como por ejemplo las bases de datos, directorios y servicios web. Todas las credenciales necesarias para la comunicación con estos sistemas que se almacenen en ficheros de configuración lo harán de forma protegida.
- **Encriptación:** Los requerimientos de la aplicación especificarán qué datos tienen que ser cifrados, como se han cifrado y como se manipularán los certificados y otras credenciales. Las especificaciones se referirán siempre a un algoritmo estándar de alguna biblioteca abastecimiento probada y utilizada.
- **Disponibilidad:** Los requerimientos tendrán que especificar como se protegerá la aplicación de los ataques de denegación de servicio. Englobando los ataques sobre el bloqueo de autenticación, agotamiento de las conexiones y en general otros ataques de exhaustiment de recursos.

## 5. Personal y organización

- **Arquitecto de seguridad:** el adjudicatario asignará responsabilidades en materia de seguridad a un recurso, que pase a ser el arquitecto de seguridad del proyecto. este certifica la seguridad de cada entrega.
- **Formación en seguridad:** el adjudicatario será responsable de asegurar que todos sus miembros del equipo de desarrollo están capacitados en las técnicas de desarrollo seguro.

## 6. Bibliotecas, frameworks de aplicación y productos

- **Divulgación:** el adjudicatario hará público el software de terceras partes usado en el desarrollo de la aplicación, incluyendo las librerías, frameworks, componentes y otros productos, sean comerciales, libres, en código abierto o en código cerrado.
- **Evaluación:** el adjudicatario realizará los esfuerzos razonables para asegurar que todo el software de terceras partes usado en la aplicación, cumple los términos de este anexo. Si las librerías tienen vulnerabilidades publicadas en el momento de la firma del contrato, el adjudicatario se comprometerá a cambiarlas por la versión que no presente vulnerabilidades o por otro producto que ofrezca mayores garantías de seguridad siempre y cuando sea compatible con el proyecto.

## 7. Revisiones de seguridad

- Derecho a la revisión: UOC tiene el derecho a revisar el código por fallos de seguridad. el adjudicatario acuerda proporcionar un apoyo razonable posteriormente a la fecha de entrega y durante el periodo de garantía.
- Cobertura de las revisiones: las revisiones de seguridad incluirán todos los aspectos relacionados con la distribución de la aplicación, incluyendo el código desarrollado, los componentes, productos y la configuración del sistema.
- Ámbito de la revisión: como mínimo, la revisión alcanzará todos los requerimientos de seguridad y busca de otras vulnerabilidades. El examen puede incluir una combinación de escaneos de vulnerabilidades, macetas de penetración, análisis del código fuente, revisiones por terceras partes y revisión del código.
- Problemas de seguridad descubiertos: serán reportados tanto por UOC como el adjudicatario. todas estas cuestiones serán rastreadas y solucionadas tal y como se especifica en la sección gestión de los problemas de seguridad de este anexo.

## 8. Gestión de los problemas de seguridad

- Identificación: el adjudicatario hará un seguimiento de todos los problemas de seguridad descubiertos durante todo el ciclo de vida de la aplicación, diseño, ejecución, ensayos, despliegue y garantía. el riesgo asociado a cada problema de seguridad será evaluado, documentado e informado en la uoc tan rápido como sea posible después de su descubrimiento.
- Protección: el adjudicatario protegerá adecuadamente la información relativa a cuestiones relacionadas con la seguridad y la documentación asociada, para ayudar a que queden expuestas en el mínimo las posibles vulnerabilidades de la aplicación.

## 9. Fiabilidad

- Fiabilidad: el adjudicatario ofrecerá un "paquete de certificación" que constará de la documentación de seguridad creada a lo largo del proceso de desarrollo. este paquete de certificación establecerá y describirá cómo los requerimientos de seguridad, diseño, implementación y los resultados de las pruebas fueron debidamente agasajados y que todas las cuestiones de seguridad se resolvieron adecuadamente.
- Certificación: el adjudicatario certifica que la aplicación cumple con los requerimientos de seguridad, que en el desarrollo se han realizado todas las actividades relacionadas con la seguridad y que todos los problemas de seguridad identificados se han documentado y resuelto. cualquier excepción a la certificación de la aplicación estará plenamente documentada en la entrega. Esta certificación estará incluida dentro del "paquete de certificación".
- Libre de código malicioso: el adjudicatario garantiza que la aplicación no contiene ningún código que no sea compatible con los requerimientos de la aplicación y debilite la seguridad de la aplicación, incluidos los virus, gusanos, bombas de tiempos, puertas secretas, troyanos, huevos de pascua y todas las demás formas conocidas de código malicioso.

## 10. Aceptación de la seguridad y garantía

- Aceptación: el software de la aplicación no se considerará aceptado hasta que el "paquete de certificación" esté completo y todas las cuestiones de seguridad se hayan resuelto.
- Investigación de los problemas de seguridad: después de la aceptación, si se descubren problemas de seguridad o existe una sospecha razonable, el adjudicatario ayudará en UOC a llevar una investigación para determinar la naturaleza de la cuestión. los hechos se considerarán una novedad.
- Cuestiones de seguridad nuevas: el adjudicatario y uoc acuerdan dentro del ámbito de aplicación, que habrá un esfuerzo necesario para resolver los nuevos problemas de seguridad y negociarán de buena fe un acuerdo para realizar las tareas necesarias para corregirlos.

## Anexo - Hoja de ruta de Software

A continuación se muestra la Hoja de ruta de Software vigente en el momento de redacción del pliego. Esta Hoja de ruta se revisa 2 golpes al año a la Release Normativa. Hay que tener en cuenta que se tiene que cumplir la normativa vigente en cada momento.

Criterios de selección de versiones:

1. La versión tiene que estar soportada por el fabricante en el momento del diseño del sistema (si la versión "soportada" UOC no estuviera alineada con el fabricante, se selecciona la última soportada por el fabricante)
2. Preferiblemente se tienen que utilizar "versiones recomendadas": seleccionamos la última versión LTS y en caso de que no exista el concepto en el software en cuestión, última estable

		Obsoleto	Soportado	Recomendado UOC	Distribución / Observaciones	
<b>Lenguajes y runtimes</b>	Golang	<= 1.20	>= 1.21	1.24		
	Java	< 11	11, 17	21	Solo se aceptan versiones LTS	
	NodeJS	<= 16	18, 20	22.14	Solo se aceptan versiones LTS	
	Python	<= 3.8	>= 3.9	3.13	(AWS lambda soporta de la 3.9 hasta la 3.12 a 26/02/25)	
	PHP	Solo mantenimiento	<= 8.1	>= 8.2	8.4	Solo para WordPress o Drupal que estén en mantenimiento, no para aplicaciones nuevas. Ni WP ni Drupal son los gestores de contenidos recomendados por el Área de Tecnología
<b>Frontal web</b>	Angular	<= 17	18	19		
	React	<= 16	17	19		
	Vue	<= 2.6.x	>= 2.7.x	3.5.x		
<b>Gestión de dependencias</b>	Maven (Java)	<= 3.6	>= 3.8	3.9		
	NPM (NodeJS)	< 7.24	>= 7.24	11.1.0		
	Go Modulas (Golang)	<= 1.21	>= 1.22	1.24		
	PiP (Python)	<= 22.x	23.x	25.0.1		

<b>Frameworks de Desarrollo</b>	Spring Framework (Java)		<= 4.3.x, <= 5.2.x	5.3.x	6.2.x	Alineadas con las version de JDK 11-17
	Flutter (desarrollo apps nativas)		< 3.0	>= 3.0.x	3.29.0	SDK de Google basado en Dart para acelerar el desarrollo de apps Android e iOS. Recomendada la última "Stable channel"
	Serverless Framework		< 3.19.x	>= 3.19.x	3.39.x	Se toma de referencia la última minor version en fecha 26/02/2025. Existe la v4 pero no se contempla porque incorpora un nuevo modelo de licensing
	Hugo (webs estáticas)		< 0.110.0	>= 0.110.0	0.145.0	Se toma de referencia la última minor version en fecha 06/02/2025
	Gatsby (webs estáticas)		< 5.0.x	5.x.x	5.14	V4.x finaliza el soporte en q4 del 23
<b>Gestión de versiones de DB</b>	Flyway (Java)		< 9.14.x	>= 9.14.x	11.3.3	Soportadas versiones que han tenido alguna release durante los últimos 2 años
	Liquibase (Java)		< 4.20	>= 4.20.x	4.31.1	Soportadas versiones que han tenido alguna release durante los últimos 2 años
	SQLAlchemy Alembic (Python)		< 1.10.0	>= 1.10.0	1.14.x	Soportadas versiones que han tenido alguna release durante los últimos 2 años
	migrate (Golang)		< 4.16	>= 4.16.x	4.18.x	Soportadas versiones que han tenido alguna release durante los últimos 2 años
	Sequelize (NodeJS)		< 6.28	>= 6.28.x	6.37.x	Soportadas versiones que han tenido alguna release durante los últimos 2 años

Aceleradores dev / Generadores de código / Low Code	JHipster		< 7.9.3	>= 7.9.3	8.9.x	JHipster, además del generador de código, proporciona un en torno a desarrollo y otros tools de apoyo (IDE Tools, JDL Studio). Para estar alineado con el modelo de CI/CD de la UOC, hace falta que se generen frontend y backend por separado.
LowCode Application Platform	OutSystems		-	-	-	ES EL MÉTODO PREFERENTE DE DESARROLLO DE APLICACIONES A LA UOC. Se tienen que analizar las características del sistema antes de decidir ir a lowcode o desarrollo tradicional. La plataforma lowcode tiene limitaciones relativas al licenciamiento (volumen de usuarios, nivel de apoyo, ...).
Infraestructura como Código	Terraform		< 1.7.5	>= 1.7.5	1.10.x	Soportadas versiones minor por las cuales la fecha de liberación de la última release tenga una antigüedad inferior a 1 año
Bases de datos (Relacional)	Oráculo	Solo mantenimiento	<=11g <19c	19c	23ai	Solo por mantenimiento de aplicaciones legacy
	MySQL		< 8.0.36	>=8.0.36	8.0.41	RDS end of standard support dato 5.7 29 February 2024
	PostgreSQL	Opción recomendada	<13.8	>=13.8	16.3	A AWS recomendamos Aurora
Bases de datos (Documental)	MongoDB		<6.0	>=6.0	7.0	A AWS recomendamos documentdb
	AWS DynamoDB		<=2017.11.29	2019.11.21	2019.11.21	AWS Global Tables. Engine version is internal manage by aws

	AWS DocumentDB				5.0	AWS no ha deprecat ninguna versión. Tenemos alguna applicació con DocumentDB?
<b>Bases de datos (Clave-Valor)</b>	Redis		< 6.0	>=6.0	7.2	A AWS recomendamos Elasticache
<b>CMS y Portales</b>	Liferay	Solo mantenimiento	<= 7.3.x	7.4.x	7.4.x	Se consideran como soportadas las versiones de Liferay dentro de "Premium Support Phase"
	OpenCMS	Solo mantenimiento	< 12.x	12.x, 13.x, 14.x, 15.x 16.x, 17x	18.x	
	WordPress	Solo mantenimiento	<= 5.x	>= 6.7	>= 6.7	No se pueden desarrollar sistemas de información con WP, solo para servicios existentes que ya se basan en este producto
	DXP					SaaS
<b>CRM</b>	Salesforce					SaaS
<b>Runtimes Docker</b>	Java		eclipso-te murin:18-j re-alpine eclipso-te murin:19-j re-alpine eclipso-te murin:20-j re-alpine	eclipso-te murin:8-jre-alpine eclipso-te murin:11-j re-alpine eclipso-te murin:17-j re-alpine eclipso-te murin:21-j re-alpine	eclipso-temurin:2 1-jre-alpine	Alpine como distribución recomendada. En caso de problemas (incompatibilidades) en fase de proyecto se decidirá qué otro utilizar.

	NodeJS		nodo:8.16 .0-alpine nodo:10-a lpine nodo:12-a lpine nodo:14-a lpine nodo:16-a lpine	nodo:18-al pine nodo:20-al pine	nodo:20-alpine	Alpine como distribución recomendada. En caso de problemas (incompatibilidades) en fase de proyecto se decidirá qué otro utilizar.
	Golang		golang:1.2 0-alpine	golang:1.2 1-alpine golang:1.2 2-alpine	golang:1.22-alpin e	Alpine como distribución recomendada. En caso de problemas (incompatibilidades) en fase de proyecto se decidirá qué otro utilizar.
	Python		python:2. 7-slim-bus ter python:3. 6-slim-bus ter python:3. 7-slim-bus ter	python:3.8 -slim-buste r python:3.9 -slim-buste r python:3.1 0-slim-bust er python:3.1 1-slim-bust er python:3.1 2-slim-bust er	python:3.12-slim- buster	Debian como distribución recomendada. En caso de problemas (incompatibilidades) en fase de proyecto se decidirá qué otro utilizar.
<b>Servidor de aplicaciones</b>	WildFly	Solo mantenimiento	<= 22.x	>= 23.x	35.x	Se recomienda ir a runtime Docker de Java y Spring Boot
	JBoss EAP	Solo mantenimiento	<= 7.3.x	7.4.x	8.x	Se recomienda ir a runtime Docker de Java y Spring Boot
	Tomcat	No recomendado	<= 7.x,8.0,8.5 ,10.0.x	9.0.x, 10.1.x	11.0.x	Se recomienda ir a runtime Docker de Java y Spring Boot
	Jetty	No recomendado	<= 11.x	12.x	12.x	Se recomienda ir a runtime Docker de Java y Spring Boot
<b>Servidor web</b>	Apache	Solo mantenimiento	<= 2.2	2017-04-0 1T22:00:00 .000Z	2017-04-01T22:00 :00.000Z	

	NGinx		<=1.24.x	1.26.x	1.26.x	
<b>Sistemas operativos</b>	Ubuntu	Solo mantenimiento	<17.x			
	RHEL/CentOS	Solo mantenimiento	<7	7.x	8.x	
	Amazon Linux 2	Opción recomienda AWS				Utilizar la AMI corporativa "TEMPLATE_UOC"
	Suse	Solo mantenimiento	<12.x			
	Windows Server		<= 2012 R2	2016, 2019	2022	Solo para servidores situados en la Red de Área Local
	Windows Desktop		<10	10	11	Solo para equipos de usuario, no para servicios en producción
	Solaris	Solo mantenimiento	<10	11		
<b>Bedrock</b>	Claude Sonnet		<=3.0	>=3.5	3.5	
	Claude Haiku		<=3.0	>=3.5	3.5	
	Titan Texto Embeddings		<2.0	>=2.0	2.0	

#### Criterios de selección de versiones:

- La versión tiene que estar soportada por el fabricante en el momento del diseño del sistema (si

la versión "soportada" UOC no estuviera alineada con el fabricante, se selecciona la última soportada por el fabricante)

- Preferiblemente se tienen que utilizar "versiones recomendadas": seleccionamos la última versión LTS y en caso de que no exista el concepto en el software en cuestión, última estable

El Grupo Operativo de Arquitectura Tecnológica actúa como dos tipos de arquitectos:

- Arquitectos de Empresa (Enterprise Architects): Ayudamos a definir la estrategia tecnológica, la visión (TONO BE), los principios de arquitectura, los riesgos a nivel de programas y servicios, etc. Es decir todo aquello de arquitectura que no tiene una relación directa con la implantación o el desarrollo de una tecnología concreta.
- Arquitectos de soluciones (Solution o Domino Architects): Ayudamos a definir la mejor solución para implantar o desarrollar una tecnología, una aplicación o un componente y participamos de las reuniones de seguimiento del proyecto para apoyar al redisseny de esta solución si es necesaria a lo largo del proyecto. También hacemos pruebas de concepto y documentamos el uso de nuevas tecnologías para poderlas conocer y anticiparnos a las necesidades de implantación.