



**PLEC DE PRESCRIPCIONS TÈCNiques QUE HA DE REGIR LA  
CONTRACTACIÓ DEL SERVEI D'IMPLEMENTACIÓ DE LA GESTIÓ  
INTEGRAL DE LES UNITATS TRIBUTÀRIES (1-2026/0024-GEC)**

**ÍNDEX**

<b>PRIMERA. OBJECTE DEL CONTRACTE.....</b>	<b>2</b>
<b>SEGONA. DESCRIPCIÓ DEL SERVEI.....</b>	<b>2</b>
2.1. Definicions.....	2
2.2. Objectius .....	4
2.3. Tipus de servei .....	6
2.4. Descripció funcional del servei .....	6
2.4.1. HU01. Altes automàtiques.....	6
2.4.2. HU02. Altes manuals via BASE (Empleat i Ens).....	8
2.4.3. HU03. Conciliació de dades .....	9
2.4.4. HU04. Vialer.....	10
2.4.5. HU05. Consultes .....	10
2.4.6. HU06. Titularitat .....	11
2.4.7. HU07. Paràmetres i atributs de les Unitats Tributàries .....	12
2.4.8. HU08. Comunicació de noves Unitats Tributàries.....	14
2.5. Requeriment tècnics .....	15
2.6. Equip .....	17
2.6.1. Interlocutor de l'adjudicatari.....	17
2.6.2. Analista funcional/orgànic.....	18
2.6.3. Programador sènior.....	20

## PRIMERA. OBJECTE DEL CONTRACTE

L'objecte del present contracte és el servei d'implementació de la gestió integral de les Unitats Tributàries (UT) que un Municipi disposa en l'àmbit de l'Import de Bens Immobles Urbans.

Aquesta gestió ha de permetre realitzar altes, baixes, modificacions i consultes de tots els immobles urbans de que disposa d'un determinat municipi, aconseguint així la base necessària per la confecció dels padrons de la pròpia Urbana i de les taxes d'Escombraries, Clavegueram i Guals.

Així mateix, l'objecte del contracte comprèn totes les tasques necessàries per garantir el correcte desplegament i funcionament de l'aplicació, incloent-hi la seva integració amb els sistemes existents i la interoperabilitat amb sistemes externs.

## SEGONA. DESCRIPCIÓ DEL SERVEI

### 2.1. Definicions

- **Archetype.** És l'eina de Maven que permet construir una aplicació a partir d'un esquelet o estructura de model.
- **Commit.** És l'enviament dels canvis realitzats des de l'entorn de treball, en local, al repositori. En el nostre cas, el repositori que utilitzem és el GIT.
- **Codi segur de verificació (CSV).** És un codi que es troba en cada document i és una referència unívoca que permet accedir al mateix document en format electrònic per verificar-ne l'autenticitat.
- **Entorn.** A BASE, tenim tres llocs web on es disposen de diferents funcionalitats de les aplicacions. Aquestes són: BASEon-line (solament accés els empleats), BASEens (accés els ens) i Seu electrònica (accés el ciutadà).

A les explicacions funcionals del servei, ens referim a l'entorn empleat, ens o ciutadà segons on afecti el desenvolupament a realitzar.

- **Foliar.** La foliació dels expedients electrònics s'ha de portar a terme mitjançant un índex electrònic, signat per una entitat, que verifica mitjançant les dades d'aquest índex que no hi ha modificacions sobre l'expedient.



- **Formulari.** És el conjunt de camps sol·licitats per un determinat programa mitjançant diferents pantalles, els quals s'emmagatzemen per al seu posterior ús.
- **Integració.** Acció d'integrar un component A amb un component B.
- **Interfície d'usuari (UI).** El disseny UI es defineix com a una nova tendència en el disseny web que s'enfoca en crear una interfície entre les persones i els diferents dispositius digitals.
- **JasperReports.** Software per generar les plantilles per ajudar a crear documents de tipus pàgina, preparats per imprimir d'una forma simple i flexible.
- **Javadoc.** Javadoc és una utilitat d'Oracle per a la generació de documentació d'APIs en format HTML a partir de codi font Java. Javadoc és l'estàndard de la indústria per documentar classes de Java. La majoria dels IDEs (entorns integrats de desenvolupament) els generen automàticament.
- **JEE.** És una plataforma de programació per desenvolupar i executar programari escrit amb el llenguatge Java amb una arquitectura distribuïda amb nivells, basada en components de programari, tot plegat executant-se en un servidor d'aplicacions.
- **Log.** És l'enregistrament seqüencial en un arxiu de tots els esdeveniments (esdeveniments o accions) que afecten a un procés particular (aplicació, activitat d'una xarxa informàtica, ...). D'aquesta manera es veu el comportament del sistema.
- **Lògica.** Capa amb les regles de negoci de l'aplicació.
- **Plantilla.** Es crea mitjançant el JasperReport i serveix per poder generar documents en un mateix format amb les diferents dades aportades.
- **Redmine.** És una eina per a la Gestió de projectes, que amb les seves diverses funcionalitats permet als usuaris de diferents projectes realitzar el seguiment i organització dels mateixos.

- **Subversion (SVN).** És una eina de control de versions del codi desenvolupat.
- **VÀlid.** Integrador de serveis d'identitat digital del Consorci Administració Oberta de Catalunya (Consorci AOC o simplement AOC) (<https://www.aoc.cat/serveis-aoc/valid/>).
- **XML.** Es proposa com a un estàndard per a l'intercanvi d'informació estructurada entre diferents plataformes. XML és una tecnologia rematadament senzilla que té al seu voltant altres que la complementen i la fan notablement més extensa, a més de proporcionar-li unes possibilitats molt més grans. En l'actualitat té un paper molt important ja que permet la compatibilitat entre sistemes compartint informació d'una manera segura, fiable i fàcil.

## 2.2. Objectius

Descriurem els objectius d'aquest servei en les següents històries d'usuari:

- **HU01. Altes automàtiques**

La càrrega automatitzada dels fitxers XML (de Cadastre i Notaris) ha d'incloure altes automàtiques d'Unitats Tributàries (UT), indexant tota la informació del vialer i permetent la seva posterior validació en cas de discrepàncies amb possibles dades existents.

- **HU02. Altes manuals via BASE (Empleat i Ens)**

Davant la possible necessitat immediata de disposar de noves Unitats Tributàries (UT) i no disposar de les dades prèvies de Cadastre o Notaris, s'han de poder fer les altes de forma manual a través de BASE-Gestió d'Ingressos.

- **HU03. Conciliació de dades**

Les possibles altes manuals fetes amb anterioritat a la recepció dels fitxers de Cadastre o Notaris s'han de conciliar (creuar) en posterioritat i verificar que no es generen duplicats en el cens de les Unitats Tributàries (UT).



- **HU04. Vialer**

Totes les Unitats Tributàries (UT), amb independència de la seva procedència, han d'estar indexades amb el vialer que disposa el municipi de la Unitat Tributària.

- **HU05. Consultes**

S'ha de disposar d'una consulta parametritzable per consultar les Unitats Tributàries (UT) d'un determinat municipi.

- **HU06. Titularitat**

Qualsevol Unitat Tributària (UT) ha de poder disposar de tres tipus de titularitat: titularitat de l'Impost de Bens Immobles, titularitat de les possibles taxes sobre l'Immoble i titularitat cadastral. Totes aquestes dades han d'estar indexades amb la base de dades de Subjectes que ja disposa BASE-Gestió d'Ingressos.

- **HU07. Paràmetres i atributs de les Unitats Tributàries**

L'aplicació ha de permetre la definició i gestió de paràmetres personalitzables per municipi, així com la configuració d'atributs associats a les Unitats Tributàries (UT), amb la finalitat de permetre la parametrització específica necessària per al càlcul de les taxes municipals, especialment la Taxa Justa de Residus.

- **HU08. Comunicació de noves Unitats Tributàries**

Qualsevol nova alta d'una Unitat Tributària (UT), massiva o manual, degudament registrada s'ha de comunicar a l'empresa concessionària del municipi amb les següents finalitats:

- Que quedi degudament registrada i sincronitzada en el seu sistema d'informació.
- Que es pugui procedir a fer les tasques pertinents de gestió de la Taxa.

### **2.3. Tipus de servei**

Els tipus de servei demanats són els d'anàlisi, desenvolupament i implantació de les històries d'usuari que s'han enumerat en la prescripció **2.2**.

### **2.4. Descripció funcional del servei**

En aquesta secció es presenta una descripció funcional de cadascun dels objectius per tal que el licitador pugui comprendre l'activitat que ha de desenvolupar, a la vegada que quantificar-ne l'esforç.

D'altra banda, s'ha de tenir en compte que BASE–Gestió d'Ingressos ja disposa d'una sèrie de components transversals amb l'objectiu d'agilitzar la implantació de noves aplicacions en el seu Sistema d'Informació. Per tant, per garantir l'èxit del projecte i no replicar funcionalitats que ja existeixen, és necessari integrar amb els components que interactuen amb l'Ordenança general de convivència. En l'*Annex\_components\_sistema* es poden trobar la descripció d'aquests components.

#### **2.4.1. HU01. Altes automàtiques**

##### **Lògica/domini**

La càrrega automatitzada dels fitxers XML (de Cadastre i Notaris) ha d'incloure altes automàtiques d'Unitats Tributàries (UT), indexant tota la informació del vialer i permeten la seva posterior validació en cas de discrepàncies amb possibles dades existents.

L'aplicació ha de:

- Implementar el model de dades necessari per a la creació i gestió d'Unitats Tributàries (UT) i la seva vinculació amb la informació del vialer.
- Generar automàticament noves d'Unitats Tributàries (UT) a partir de la informació continguda en els fitxers XML.
- Detectar l'existència prèvia d'Unitats Tributàries (UT) mitjançant criteris d'identificació unívoca.



- Comparar la informació entrant amb les dades existents i identificar possibles discrepàncies.
- Garantir la traçabilitat completa del procés, incloent origen de les dades, data i resultat del tractament.
- Indexar els camps rellevants per assegurar la seva posterior consulta i explotació.

També ha d'incorporar mecanismes automatitzats que permetin:

- El processament de fitxers XML i la validació prèvia de l'estructura i integritat del fitxer.
- El tractament massiu de registres amb control d'errors per element.
- La creació transaccional de les Unitats Tributàries (UT).
- La generació d'informes de resultat del procés de càrrega.
- La implementació de mecanismes per evitar altes duplicades.

### **Interfície gràfica**

La interfície gràfica ha de disposar de funcionalitats que permetin:

- Executar la càrrega dels fitxers XML de Cadastre i Notaris, amb selecció del fitxer corresponent i identificació automàtica del seu origen. L'aplicació ha d'impedir la càrrega de fitxers amb formats no admesos o estructures incorrectes.
- Cercar i consultar les càrregues realitzades per estat, data de càrrega i origen del fitxer.
- Visualitzar el resultat de l'operació càrrega. En cas d'error, ha de mostrar el detall dels errors detectats.

## **2.4.2. HU02. Altes manuals via BASE (Empleat i Ens)**

### **Lògica/domini**

Davant la possible necessitat immediata de disposar de noves Unitats Tributàries (UT) i no disposar de les dades prèvies de Cadastre o Notaris, s'han de poder fer les altes de forma manual a través de BASE-Gestió d'Ingressos.

- Permetre la introducció manual de totes les dades obligatòries i opcionals associades a la Unitat Tributària (UT).
- Validar en temps real la informació introduïda.
- Detectar l'existència prèvia d'Unitats Tributàries (UT) mitjançant criteris d'identificació unívoca per evitar duplicitats.
- Comparar la informació introduïda amb dades existents i advertir de possibles discrepàncies.
- Assignar l'estat corresponent a la UT en el moment de la seva creació.
- Garantir la traçabilitat completa del procés.
- Indexar els camps rellevants per assegurar la seva posterior consulta i explotació.
- Establir mecanismes de control de concurrència per evitar duplicacions en escenaris d'edició simultània.

### **Interfície gràfica**

La interfície gràfica ha de disposar de funcionalitats que permetin:

- Crear una nova Unitat Tributària (UT) mitjançant un formulari estructurat que permeti introduir les dades de la UT, amb indicació clara de camps obligatoris i opcionals.
- Validar la informació en temps real, mostrant missatges d'error o advertiment quan correspongui.
- Confirmar o cancel·lar l'operació abans del registre definitiu.



- Visualitzar el resultat de l'operació d'alta, indicant si s'ha realitzat correctament o si s'han detectat errors que impedeixen la seva creació.

### **2.4.3. HU03. Conciliació de dades**

#### **Lògica/domini**

Les possibles altes manuals fetes amb anterioritat a la recepció dels fitxers de Cadastre o Notaris s'han de conciliar (creuar) en posterioritat i verificar que no es generin duplicats en el cens de les Unitats Tributàries (UT).

L'aplicació ha de:

- Implementar mecanismes de creuament automàtic entre les UT creades manualment i les provinents de fitxers o que ja estiguin donades d'alta prèviament en el sistema d'informació.
- Detectar possibles duplicats i classificar els resultats de la conciliació.
- Mantenir registre d'històric de decisions adoptades en processos de conciliació.

També ha d'incorporar mecanismes que permetin:

- El tractament massiu de registres amb control d'errors per element.
- La gestió transaccional del procés per garantir la integritat de les dades.
- La implementació de regles per evitar la creació o manteniment de duplicats.

#### **Interfície gràfica**

La interfície gràfica ha de disposar de funcionalitats que permetin:

- Consultar els resultats dels processos de conciliació realitzats per identificar clarament possibles duplicats o discrepàncies.
- Permetre la resolució manual de la conciliació.

#### **2.4.4. HU04. Vialer**

##### **Lògica/domini**

Totes les Unitats Tributàries (UT), amb independència de la seva procedència, han d'estar indexades amb el vialer que disposa el municipi de la Unitat Tributària.

L'aplicació ha de:

- Implementar la vinculació obligatòria entre la UT i el vialer municipal.
- Validar que no es pugui crear ni mantenir una UT sense associació a un element vàlid del vialer.
- Garantir la coherència entre les dades d'adreça de la UT i la informació estructurada del vialer.
- Permetre l'actualització de la vinculació en cas de modificació del vialer.
- Garantir la traçabilitat de qualsevol modificació en la vinculació amb el vialer.

També ha d'incorporar mecanismes que permetin:

- La validació automàtica de la vinculació amb el vialer en qualsevol procés de creació o modificació d'UT.
- La detecció de UT existents sense indexació correcta i la generació d'incidències.
- L'execució de processos de reindexació en cas d'actualització del vialer municipal.

#### **2.4.5. HU05. Consultes**

##### **Lògica/domini**

S'ha de disposar d'una consulta parametrizable per consultar les Unitats Tributàries (UT) d'un determinat municipi.

L'aplicació ha de:



- Implementar els criteris de filtratge necessaris per recuperar les UT vinculades a un municipi concret per garantir que la consulta es basi en la vinculació estructurada entre la UT i el municipi.
- Permetre la combinació del municipi amb altres paràmetres de filtratge.
- Garantir la coherència i integritat de la informació retornada.
- Aplicar els controls de seguretat i permisos d'accés segons el perfil d'usuari (empleat o ens).

### **Interfície gràfica**

La interfície gràfica ha de disposar de funcionalitats que permetin:

- Seleccionar el municipi objecte de consulta i definir filtres addicionals per refinar la cerca.
- Executar la consulta i visualitzar el llistat de resultats (ordenar i pàginar els resultats).
- Mostrar, per cada UT, la informació rellevant (identificador, adreça, estat, origen, data d'alta, etc.).
- Accedir al detall d'una UT concreta des del llistat.

#### **2.4.6. HU06. Titularitat**

##### **Lògica/domini**

Qualsevol Unitat Tributària (UT) ha de poder disposar de tres tipus de titularitat: titularitat de l'Impost de Bens Immobles, titularitat de les possibles taxes sobre l'Immoble i titularitat cadastral. Totes aquestes dades han d'estar indexades amb la base de dades de Subjectes que ja disposa BASE-Gestió d'Ingressos.

L'aplicació ha de:

- Implementar el model de dades necessari per gestionar els tres tipus de titularitat de manera independent però vinculada a la mateixa UT.
- Permetre que una UT disposi d'un o diversos titulars per cada tipus de titularitat, d'acord amb la normativa aplicable.
- Garantir la integritat referencial entre la UT i els titulars associats.
- Indexar obligatòriament totes les titularitats amb la base de dades de Subjectes corporativa ja existent a BASE-Gestió d'Ingressos.

### **Interfície gràfica**

La interfície gràfica ha de disposar de funcionalitats que permetin:

- Mostrar les titularitats associades a una Unitat Tributària (UT), diferenciant clarament els tres tipus.
- Visualitzar per cada titular la informació bàsica del Subjecte associat (identificador, nom/raó social, etc.).

#### **2.4.7. HU07. Paràmetres i atributs de les Unitats Tributàries**

##### **Lògica/domini**

L'aplicació ha de permetre la definició i gestió de paràmetres personalitzables per municipi, així com la configuració d'atributs associats a les Unitats Tributàries (UT), amb la finalitat de permetre la parametrització específica necessària per al càlcul de les taxes municipals, especialment la Taxa Justa de Residus.

L'aplicació ha de:

- Permetre la definició de paràmetres específics per a cada municipi. Per a cada paràmetre, cal definir:
  - La seva descripció.
  - El tipus de valor (enter, decimal, booleà, llista predefinida, etc).
  - Vigència (data d'inici i, si escau, data de fi).
- Associar un o diversos paràmetres a cada Unitat Tributària (UT).



- Validar que els valors assignats a cada paràmetre siguin coherents amb el seu tipus de dada.
- Permetre la configuració d'atributs vinculats al tipus de tribut (m<sup>2</sup>, ús, categoria, etc.) i definir el tipus de valor (fixe o per intervals).
- Garantir que els atributs aplicables depenguin del tipus de tribut associat a la UT.
- Permetre que la combinació de paràmetres i atributs constitueixi la base de càlcul de les Taxes municipals.
- Garantir la traçabilitat completa de les configuracions i modificacions realitzades.

També ha d'incorporar mecanismes automatitzats que permetin:

- Validar automàticament la coherència de tipus de dada en la introducció o actualització massiva de valors.
- Gestionar canvis de vigència amb aplicació automàtica en períodes impositius futurs.
- Permetre càrregues o actualitzacions massives de paràmetres i atributs.

Mitjançant aquesta configuració, cada municipi ha de poder parametritzar les seves Unitats Tributàries (UT) de forma autònoma per adaptar-les als criteris de càlcul de les seves Taxes, especialment en el marc de la Taxa Justa de Residus.

### **Interfície gràfica**

La interfície gràfica ha de disposar de funcionalitats que permetin:

- Crear i mantenir paràmetres específics per municipi.
- Seleccionar i configurar atributs vinculats a cada tipus de tribut.
- Associar paràmetres i atributs a una Unitat Tributària (UT) i introduir-ne els valors.

- Visualitzar tots els valors configurats (paràmetres i atributs) que intervenen en el càlcul de la taxa d'una Unitat Tributària (UT).
- Consultar l'històric de configuracions i modificacions.

#### **2.4.8. HU08. Comunicació de noves Unitats Tributàries**

##### **Lògica/domini**

Qualsevol nova alta d'una Unitat Tributària (UT), massiva o manual, degudament registrada s'ha de comunicar a l'empresa concessionària del municipi amb les següents finalitats:

- Que quedi degudament registrada i sincronitzada en el seu sistema d'informació
- Que es pugui procedir a fer les tasques pertinents de gestió de la Taxa.

L'aplicació ha de:

- Detectar qualsevol nova alta efectiva d'UT.
- Determinar el municipi corresponent i la concessionària associada.
- Preparar el missatge amb la informació necessària per a la correcta identificació i gestió de la UT.
- Garantir que les dades comunicades siguin coherents, completes i consistents.
- Assegurar la traçabilitat de cada comunicació realitzada.
- Mantenir l'estat de sincronització de cada UT (pendent d'enviament, enviada, confirmada, amb error).
- Permetre la reemissió controlada de comunicacions en cas d'error.
- Garantir la integritat i confidencialitat de la informació transmesa.

L'aplicació ha d'incorporar mecanismes automatitzats que permetin:

- L'enviament automàtic de la informació en el moment de l'alta efectiva de la UT.
- La gestió de cues per garantir l'entrega encara que hi hagi incidències temporals de connectivitat.



- La recepció i registre de la resposta.
- La implementació de mecanismes de reintent automàtic en cas d'error tècnic.

### **Aspectes generals aplicables a totes les Històries d'Usuari**

Per a totes les Històries d'Usuari, cal tenir en compte que:

- **Entorns:** totes les funcionalitats s'han de desplegar en dos entorns diferenciats: Empleat (amb capacitat de gestió multi-municipi) i Ens (restringit exclusivament a la gestió del propi municipi).
- **Proves:** s'han de contemplar, com a mínim, els següents tipus de proves: unitàries, d'integració, de rendiment, funcionals i d'usuari.
- **Interfície gràfica:** BASE entregará els prototips de totes les interfícies gràfiques indicades en aquest *Plec*.

## **2.5. Requeriment tècnics**

A continuació es detallen els requeriments tecnològics:

- Entorn de desenvolupament JEE (BASEFrame – Framework JEE de BASE).
- Gestor documental: Documentum.
- Servidors d'aplicacions: Red Hat – Jboss.
- Servidors de base de dades: Informix – Oracle.
- Gestió de projectes/ticketing: Redmine.

Per a la implementació de les històries d'usuari, s'ha de treballar amb:

- Java JDK 1.6.0\_29.

- Maven 3.5.2.
- Encoding: ISO-8859-15.
- Utilitzar la versió d'Spring posada a les llibreries de BASE.
- Per generar les plantilles: JasperReports.
- Per generar la documentació UML: Papyrus.

L'estructura de l'aplicació ha de complir amb l'esquelet definit en l'*archetype* de BASE, que consta de:

- **Model de domini.** Artefacte on incloem el codi referent al domini. Aquí podem trobar les classes entitats i les interfícies dels mètodes que la capa de presentació utilitza per cridar als mètodes de la lògica de negoci. Aquest projecte ha de ser compartit per la capa de presentació i la lògica de negoci.
- **Lògica.** Aquí es troba la lògica de negoci que implementa la interfície que es troba al domini i els mètodes DAO que fan els accessos a la base de dades.
- **Recursos.** Conté els arxius de recursos per a la resolució de la connexió amb el servei (EJB, Enterprise JavaBean) i els arxius de missatges, si n'hi ha, dels errors específics de la lògica. Aquest mòdul és opcional i només és necessari en cas que el projecte publiqui un servei o hi hagi missatges d'error específics.
- **UI.** Conté la capa de presentació.
- **EAR.** Conté la resta d'artefactes per a la seva distribució unificada.
- **Test.** Subprojecte opcional i que depèn del tipus de projecte. Normalment, en els projectes de serveis\_d'arquitectura, s'incorpora aquest subprojecte per fer simulacions del servei en situacions en què no és pràctic definir-les com a un test unitari/integració. Aquest tipus de projectes de test estan orientats a poder invocar un servei també amb el propòsit d'iniciar una tasca de la qual no es té encara desenvolupat el *front-end*. El projecte normalment consta d'una única classe principal *main* i aconsegueix les referències als servei a fer les proves per el JNDI (Java Naming and Directory Interface).



## 2.6. Equip

L'equip requerit per BASE consisteix en tres perfils clau: el d'interlocutor de l'adjudicatari, el d'analista funcional/orgànic i el de programador sènior.

La composició de l'equip i la distribució orientativa dels perfils bàsics del projecte ha de ser de la següent manera:

Perfil	Nombre
Interlocutor de l'adjudicatari	1
Analista funcional/orgànic	1
Programador sènior	2

El cap de projecte de BASE és l'encarregat de vetllar que la documentació i la planificació compleixen amb els requisits establerts.

És responsabilitat de l'empresa adjudicatària l'assoliment de totes les fites requerides per executar cadascuna de les tasques comunicades i prioritzades pel cap de projecte de BASE, des de la presa de requeriments fins a la coordinació del desplegament a producció.

Tots els treballs s'han de realitzar de forma remota via VPN en les pròpies oficines de l'adjudicatari.

### 2.6.1. Interlocutor de l'adjudicatari

L'interlocutor de l'adjudicatari és la figura clau en la planificació, execució i control del projecte i és qui ha d'impulsar l'avenç d'aquest mitjançant la presa de decisions necessàries per assolir els objectius. De la mateixa manera és l'encarregat de coordinar-se i trobar les solucions adequades, adaptades a l'estructura actual de BASE i acceptades pel cap de Serveis Informàtics.

Tasques a realitzar per l'interlocutor de l'adjudicatari:

- Col·laborar amb el cap de projecte de BASE en la definició i concreció dels objectius del projecte.
- Planificar el projecte en tots els seus aspectes, identificant les activitats a realitzar, els recursos a destinar, els terminis i els costos previstos.
- Dirigir i coordinar tots els recursos destinats al projecte. En cas de mancances, elevar-les al cap de Serveis Informàtics per trobar una solució.
- Gestionar la demanda de peticions mitjançant l'eina Redmine.
- Ser el responsable d'obtenir l'aprovació de les definicions per part del cap de Serveis Informàtics.
- Prendre les decisions necessàries per conèixer en tot moment la situació en relació amb els objectius previstos.
- Adoptar les mesures correctores pertinents per redireccionar les desviacions que s'hagin detectat.
- En cas que el cap de projecte de BASE hagi sol·licitat reunions de seguiment, l'interlocutor de l'adjudicatari ha de realitzar-ne les actes.
- En cas que el cap de projecte de BASE requereixi un Informe per detallar els motius d'una desviació o d'alguna problemàtica associada al projecte, l'interlocutor de l'adjudicatari és qui l'ha de redactar.

### **2.6.2. Analista funcional/orgànic**

L'analista funcional/orgànic ha d'identificar i donar resposta a les necessitats funcionals de BASE amb l'elaboració i execució del projecte, complint amb els estàndards i la documentació requerida per la Metodologia de gestió de projectes de BASE (especificada a BASEFrame), seguint la normativa de control de la qualitat determinat pel cap de projecte de BASE.

En els requeriments funcionals i en els requeriments tècnics definits en aquest *Plec* i en el seu *Annex*, BASE especifica què vol aconseguir, com ho vol aconseguir i amb quin objectiu ho vol aconseguir. A partir d'aquesta documentació, el perfil d'analista funcional/orgànic ha de ser capaç de complir amb les següents competències:

- Entendre els requeriments funcionals i requeriments tècnics sent-hi crític amb l'objectiu d'aportar qualsevol millora.



- Realitzar l'anàlisi tècnic des d'una perspectiva funcional, és a dir, realitzar diagrames d'arquitectura lògics, diagrames de seqüència de les operacions, diagrames conceptuals...
- Aplicar tot el que ha estat documentat i diagramat en la implantació de la solució, és a dir, que la documentació s'ha de correspondre 100x100 amb la realitat.

Més específicament i associades a les competències esmentades, aquest mateix perfil ha de realitzar les següents tasques:

- Ser l'encarregat d'agafar els requeriments del cap de projecte de BASE i plasmar-los en la solució *objectiu*. Per realitzar aquesta tasca, és important adequar el vocabulari i el nivell de detall del discurs (menys tecnicismes, menys nivell de detall...) amb l'objectiu de facilitar la comprensió.
- Gaudir d'autonomia de decisió, capaç de decidir si una petició es pot realitzar, si es disposen de les eines necessàries i apropiades i, finalment, si és la millor manera de donar-li cobertura.
- Ha de realitzar una completa descripció de cada unitat funcional i en conseqüència orgànica. També ha de definir els mètodes de control per evitar errades de la solució proposada.
- Detectar, en la mesura del possible, eventuais omissions en la petició del cap de projecte de BASE i comunicar-les per consignar-les en la petició identificada.
- Validar i obtenir l'aprovació per part del cap de projecte de BASE.
- Liderar les reunions amb BASE exposant i plantejant els punts a tractar i, un cop acabades les reunions, assignar les tasques a cadascuna de les parts implicades, si escau.
- Revisar les estimacions.
- Entendre els conceptes de la metodologia àgil de BASE i documentar els diferents aspectes del projecte segons els principis d'aquesta i seguint els requisits del cap de projecte de BASE.
- Definir els casos de prova.
- Guiar l'usuari en l'execució dels casos de prova.
- Proposar les necessitats de millores.
- Administrar els canvis.

- Analitzar alternatives de les possibles implementacions.
- Definir les parametritzacions.
- Implementar la solució tot buscant el consens amb l'equip de desenvolupament de BASE.
- Fer els canvis de codificació i de configuració necessaris per satisfer els objectius del contracte.
- Definir proves funcionals.
- Realitzar proves funcionals.
- Donar el suport post implantació.

### 2.6.3. Programador sènior

El programador sènior ha d'escriure, depurar i mantenir el codi font complint amb els estàndards de BASE, fent ús del *framework* de desenvolupament de BASE i seguint la normativa de qualitat determinat pel cap de projecte de BASE.

Tasques a realitzar pel programador sènior:

- Implementar les especificacions de l'analista funcional/orgànic en codi executable.
- Fer ús de les funcionalitats del *framework* de desenvolupament de BASE.
- Definir les proves unitàries del software.
- Implementar i executar les proves unitàries.
- Detectar eventuais omissions en les definicions de l'analista funcional/orgànic.
- Comunicar-se constantment amb l'analista funcional amb l'objectiu d'evitar desviacions innecessàries.

Carles Casas Vidal  
Cap de Servei de Serveis Informàtics



**DILIGÈNCIA:** Per fer constar que el present plec de prescripcions tècniques que ha de regir la contractació servei d'implementació de la gestió integral de les Unitats Tributàries de BASE-Gestió d'Ingressos, es va aprovar per decret de la Presidència en data 27 de maig de 2026.

El secretari

<signatari1>

Contra aquests plecs es podrà interposar recurs contenciós administratiu, davant del Tribunal d'Instància de Tarragona, Secció contenciosa administrativa, en el termini de dos mesos o bé, alternativament i de forma potestativa, recurs especial davant l'òrgan de contractació o del Tribunal Català de Contractes del Sector Públic, en el termini de 15 dies hàbils. En ambdós supòsits el termini d'impugnació es comença a computar des de l'endemà de la publicació dels plecs en el perfil de contractant.