

PLEC DE PRESCRIPCIONS TÈCNIQUES PER LA CONTRACTACIÓ DEL DESENVOLUPAMENT D'UN SOFTWARE PER L'EDICIÓ DE LAYOUTS DEL PORTAL WEB I APLICACIONS DIGITALS DE LA SOCIETAT MERCANTIL CORPORACIÓ CATALANA DE MITJANS AUDIOVISUALS, SA

- PROCEDIMENT OBERT-

EXPEDIENT NÚM. 2406OB01

1. INTRODUCCIÓ A LA PLATAFORMA DE PUBLICACIÓ DE LA CCMA,S.A.

Es requereix evolucionar el portal web actual de la CCMA,S.A. per tal de poder accedir a un mode edició i així poder editar la disposició dels components que componen una pàgina del portal web de la ccma, screens de les aplicacions mòbils o seccions d'aplicacions SmartTV o HbbTV, tots ells generats amb la plataforma de publicació JIT-Pub2 pròpia de la CCMA.

El funcionament de la plataforma JIT-Pub2 està basat en que hi ha un catàleg de components amb els quals es pot compondre una pàgina. Per cada tipus de pàgina hi ha un fitxer json que indica

- quins components hi ha
- quina jerarquia hi ha entre els components
- disposició que tenen dins la pàgina
- quins fitxers de dades necessita la pàgina
- quines dades agafa de cada fitxer a cada component
- quina configuració tenen els components

Aquests fitxers jsons que indiquen la configuració de les pàgines són els **layouts** del portal.

La CCMA,S.A. està treballant en habilitar un mode edició de les pàgines del portal web, on hi ha disponible, via API, un catàleg de mòduls que indica quins atributs pot personalitzar cada component, i aquest mode ofereix la opció de llegir un layout ja existent per la seva visualització o edició.

L'objectiu d'aquesta licitació és el desenvolupament requerit per poder realitzar l'edició dels layouts de manera visual afegint funcionalitats a la plataforma de publicació JIT-Pub2.

2. NECESSITATS DE L'EDITOR DE LAYOUTS - REDL2

El portal web de la CCMA corre sobre un servidor nodejs versió 20 implementat amb el framework NextJS versió 12. Els evolutius del codi per implementar l'editor de layouts podran ser codi Javascript o Typescript. Aquests evolutius seran anomenats a partir d'ara **REDL2** (Responsive Editor De Layouts 2).

En el supòsit que l'empresa adjudicatària hagi de realitzar qualsevol canvi pel correcte funcionament de l'REDL2 a qualsevol dels entorns de la CCMA,S.A. l'empresa adjudicatària ha de validar prèviament amb l'equip d'Enginyeria de la CCMA,S.A. la proposta i adaptar els scripts de compilació, testing i desplegament actuals.

El codi està allotjat en un repositori GIT - Bitbucket de la CCMA,S.A. en una branca de desenvolupament, i el personal tècnic de la CCMA haurà de poder provar els evolutius als seus entorns de desenvolupament actual, desplegar als entorns d'integració/test, i un cop acabat el desenvolupament, a pre-producció i producció.

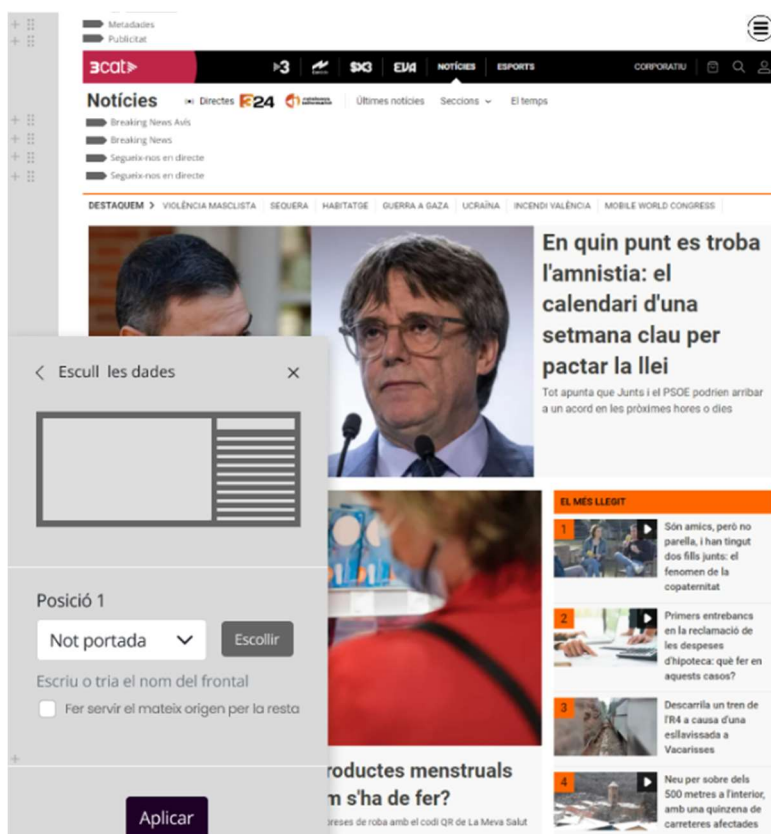
Per editar els layouts, l'empresa adjudicatària

- Haurà d'aplicar funcionalitat de drag and drop per reordenar blocs horitzontals de components del portal
- Haurà d'afegir formularis d'edició dinàmics, en funció dels atributs de cada component dins del bloc de components, per poder modificar-los amb els valors vàlids segons indica l'API de la CCMA S.A.
- S'haurà d'oferir formularis dinàmics per afegir i treure blocs de components, parametritzant segons indica l'API de la CCMA S.A.

Per indicar amb quines dades treballa cada component, existeix un llistat de dades disponibles (crides a API de dades) amb les quals poden treballar els components. Amb aquest llistat de dades s'afegeixen i treuen orígens de dades, i s'associen als diferents components.

A l'hora de visualitzar un canvi en el layout, l'editor s'haurà de recolzar en la plataforma de publicació JIT-PUB2. Un cop s'hagi fet un canvi, se li enviarà la configuració del layout provisional resultant d'aquesta modificació a la plataforma. La crida retornarà la plana ja renderitzada amb els canvis de layout.

Per tal de poder interpretar la segmentació en blocs de mòduls dins de JIT-Pub2, el sistema de render afegeix metainformació extra quan estem en mode edició. Amb aquesta metainformació, l'REDL2 haurà de ser capaç d'oferir eines visuals per reordenar components, afegir-ne de nous, treure'n de la pàgina, editar-los, etc.



Exemple d'esquema

2.1. Entorn de desenvolupament

L'entorn de desenvolupament de la CCMA és Windows10 amb nodejs versió 20. Per executar l'aplicació es proporcionaran scripts .bat que aixequen el servidor en la màquina del desenvolupador i es connecten als entorns de desenvolupament de la CCMA, S.A. per aconseguir les dades i configuracions necessàries.

2.2. Entorn d'integració / test

Per desplegar a l'entorn d'integració / test, el codi s'ha de pujar al Bitbucket de la CCMA, S.A., i executar una pipeline de Jenkins que s'executa sobre un servidor CentOS7 per generar un instal·lable. Aquesta pipeline ja està configurada per agafar el codi amb l'arquitectura actual, per tant, s'ha de fer servir l'actual arquitectura de directoris.

Els servidors de l'entorn d'integració / test són CentOS7 amb Nodejs 16 i Alma Linux 9 amb Nodejs versió 20.

2.3. Entorns de PRE Producció i Producció

Per poder desplegar els desenvolupaments als entorns de PRE Producció i Producció s'ha de realitzar un Pull Request a la branca Master de desenvolupament de la CCMA, S.A. i els desenvolupadors d'Enginyeria de la CCMA S.A. validaran i aprovaran la integració. Si cal, es farà una sessió d'explicació i la revisió es farà conjunta amb l'equip tècnic de l'empresa adjudicatària.

Els servidors de PRE i Producció són servidors CentOS7 amb Nodejs 16 i Alma Linux 9 amb Nodejs versió 20

Un cop integrats els funcionaments i desplegats a PRE Producció, s'acabarà de validar el desenvolupament amb dades reals, i un cop acceptat per part de la CCMA es donarà per validat el desenvolupament.

3. DESENVOLUPAMENTS A IMPLEMENTAR

Tots els desenvolupaments realitzats per l'empresa adjudicatària han de disposar dels tests automàtics (unitaris o integració) per validar el seu funcionament en cada procés de desplegament a l'entorn d'integració. Aquests tests s'han d'especificar amb llenguatge Gherkin i s'executen sobre Cypress amb una extensió Cucumber.

A continuació es detallen funcionalitats a implementar (Backlog) per dotar el portal de l'editor de layouts que es demana sobre el portal actual de la CCMA.

Base de desenvolupament

La CCMA, S.A. entregarà una versió (branch) de la plataforma de publicació de la CCMA on hi ha un accés en mode edició al portal web amb una parametrització especial a la URL. Si un usuari demana accedir en mode edició, requereix d'user/password que valida contra una API privada de la CCMA. Si l'usuari demana aquest accés des de fora de la CCMA (VPN) es denega l'accés. Si les credencials no són correctes, també es denega l'accés. En cas d'accés correcte, es guarda el JWT d'accés en SessionStorage. A partir

d'aquest moment, tota navegació del portal en mode edició comprova que tenim aquest JWT disponible. En cas contrari torna a demanar les credencials.

Quan estem en mode edició:

- Les planes disposen de la metainformació de cada bloc del portal web per tal que la interfície client del portal tingui la informació necessària per afegir eines visuals per reordenar i eliminar blocs de components (tota la informació del layout actual).
- Generalment, les pàgines web de la CCMA poden llegir la configuració (layout) de la pàgina segons la URL perquè tenen associades un layout que demana via http. En mode edició, JITPub2 detecta si arriba el layout a la request per poder carregar una configuració que encara no s'ha guardat i s'està treballant.

A partir d'aquí es demanen les següents evolucions per aconseguir disposar de l'REDL2.

3.1. Incorporar components visuals en mode edició per eliminar i reordenar blocs frontend

Quan entrem en mode edició afegirem botons per moure, editar i eliminar un bloc frontend. Aquí treballaríem la maquetació sense les funcionalitats.

3.2. Implementar eliminació de bloc frontend

Al clicar el botó d'eliminar un bloc horitzontal, modificarem el layout de la pàgina que tinguem en memòria eliminant el bloc, i verificarem que la plana es recarrega automàticament enviant per POST el layout modificat, mantenint la sessió d'usuari (via cookie o passant l'authorization per headers)

3.3. Implementar funcionalitat de desar

Farem una petició un endpoint de l'API de publicació que guardarà els canvis del layout passant el layout i les credencials de l'usuari. Si hi ha algun error, informarem a l'usuari i no canviarem de pàgina. Si es guarda correctament, recarregarem la pàgina fent servir el layout que acabem de guardar sense passar el layout per POST, només mantindrem les credencials.

3.4. Implementar reordenació de bloc frontend via drag and drop

Afegirem la funcionalitat drag and drop de reordenar tot un bloc horitzontal de la pàgina. Això haurà d'invocar a la funció que recol.loca el bloc a la nova posició. Haurem de comprovar que es reordenarà el layout en memòria de la pàgina reflectint el canvi, i es recarregarà la pàgina actual enviant per POST la informació del layout modificat, mantenint la sessió d'usuari (via cookie o passant l'authorization per headers)

3.5. Implementar poder afegir un bloc frontend d'un catàleg preconfigurat

Amb un botó per afegir un bloc horitzontal, mostrarem una llista senzilla de blocs per afegir a la pàgina. Aquesta llista la proveirà la CCMA, S.A. amb un fitxer json que indicarà el nom del bloc, una imatge thumbnail i un path amb el que s'haurà de construir la url que ens retorna el codi que haurà d'afegir al layout. Aquest json es carregarà via petició http quan l'usuari demani d'afegir un bloc.

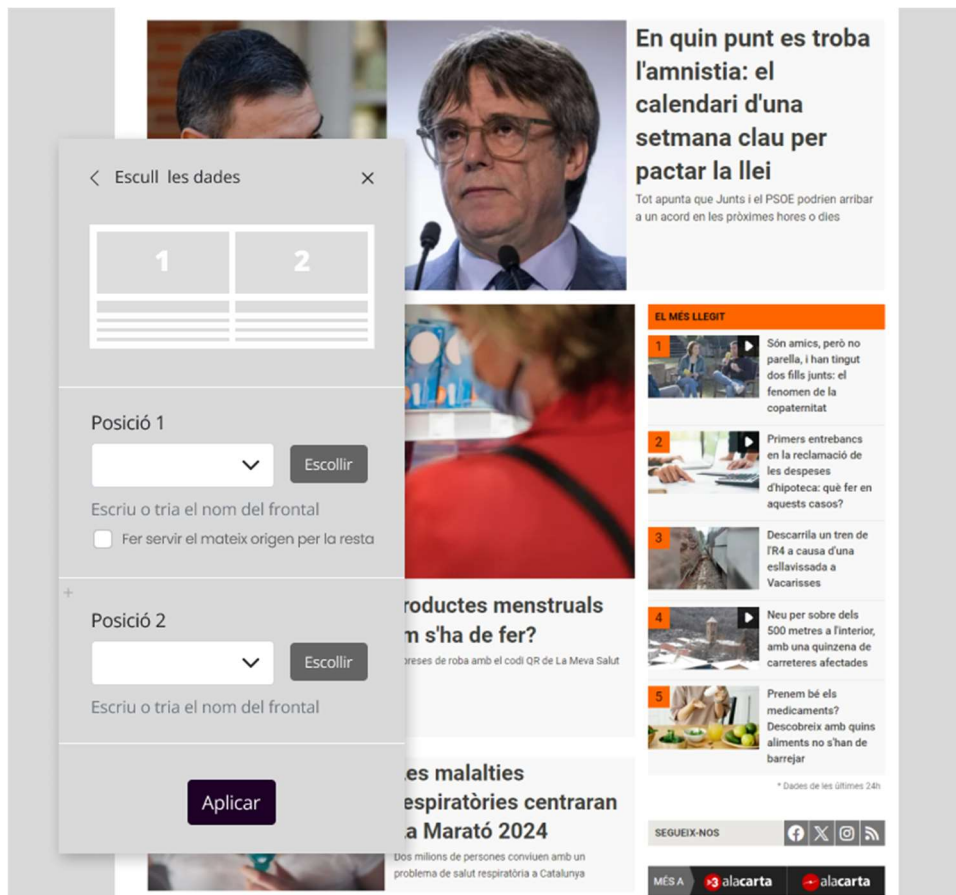
3.6. Implementar personalització del bloc frontend a l'hora d'afegir un bloc

Quan afegim un bloc, el json que conté el codi del mòdul indicarà la parametrització que té el bloc, i el frontend haurà de mostrar un formulari per personalitzar aquests paràmetres per part de l'usuari. Exemple: Si afegim un bloc que té component amb un atribut "path" el json que conté el codi indicarà que el path és editable, i l'usuari ha de poder escriure el valor del nou path.

3.7. Implementar desplegable com a ajudar a l'hora d'escollir origen de dades

Els formularis per parametritzar els blocs de components fins aquest moment tenen camps de text oberts. S'evolucionarà per tal que ofereixin ajudes visuals per omplir la informació amb crides a l'API del gestors de continguts de la CCMA.

En primera instància tindrem un desplegable amb els orígens de dades que ja tenim present al layout.



El camp serà un desplegable que accepta escriptura oberta amb autocompletat. Si no hi ha match, l'usuari podrà continuar escrivint completament l'origen de dades. En aplicar els canvis es modificarà el layout de la pàgina que tinguem en memòria modificant el valor de l'atribut de l'origen de dades, i comprovarem que la pàgina actual es recarrega enviant per POST la informació del layout modificat, mantenint la sessió d'usuari (via cookie o passant l'authorization per headers).

3.8. Implementar afegir origen de dades

Afegirem un component visual que mostrarà els orígens de dades que disposa el layout. Sobre això, facilitarem l'acció d'afegir un nou origen de dades. Aquest formulari ha d'estar pensat per usuaris no tècnics (editors del portal).

3.9. Esborrar origen de dades que no es fan servir quan es desa un layout

Quan un usuari fa l'acció de Desar un layout, es comprovarà si algun origen de dades no s'està fent servir a cap component o altres orígens de dades. En aquest cas, es mostrarà un modal informant que aquests orígens de dades que no es fan servir seran esborrats, i abans d'enviar el layout a l'API, prèvia confirmació, es trauran els orígens de dades que no es facin servir.

3.9. Implementar edició de blocs frontend

Els blocs que ja estan en pàgina mostraran els mateixos formularis per modificar les dades que es mostren quan s'afegeix un bloc nou. Per saber quins atributs són editables es pot consultar el json que conté el codi de configuració dels components.

3.10. Implementar la visualització en 2 nivells del catàleg de blocs per afegir

A l'hora d'afegir un bloc de components nou, es crida una URL que retorna un llistat de blocs per afegir. Evolucionarem aquesta funcionalitat per tal que el frontend accepti la jerarquia/agrupació dels blocs, és a dir, els blocs podran estar agrupats en categories, simulant carpetes. Quan es seleccioni un bloc, s'afegirà com fins ara. Quan es seleccioni una agrupació, es tornarà a cridar l'API perquè retorni un nou llistat que pot tenir, de nou, blocs o agrupacions. L'API informará si un node és un bloc de components o una agrupació.

4. REQUERIMENTS TÈCNICS PER LA IMPLEMENTACIÓ

L'entrega de l'aplicació ha de complir els següents requeriments tècnics:

- El desenvolupament i scripts necessaris han de ser compatibles amb Node v20 LTS
- Per carregar dades dinàmiques XHR, s'ha de fer servir react-query (TanStack Query) amb Axios.
- No es podran fer servir llibreries de software de pagament. Totes les llibreries que s'hagin d'afegir, s'han de pactar amb la CCMA, S.A. i hauran de ser de codi obert.
- La maquetació ha de respectar i continuar els estils entregats al portal de la CCMA. Si s'han d'afegir llibreries (com dnd-kit peel drag and drop) cal validació amb l'equip tècnic de la CCMA.
- El codi es pujarà al servidor GIT de la CCMA (BitBucket).
- La compilació i empaquetat del codi s'ha de realitzar amb maven, projecte ja configurat pel desplegament del portal de la CCMA
 - La compilació, empaquetat, testing i desplegament es validarà a l'entorn d'integració / test de la CCMA
 - Es farà mitjançant l'execució d'una pipeline de Jenkins (jenkinsfile)
- Pel testing, es farà servir el llenguatge Gherkin per definir els tests, i s'executarà amb Cypress
 - La CCMA proveirà d'una configuració mínima d'exemple de testing

A nivell de rendiment client es faran validacions per comprovar que l'experiència d'usuari és acceptable segons els estàndards de qualitat recomanats per la indústria fent servir

Core Web Vitals. Es miraran paràmetres que proporciona l'eina de profiler de Google Chrome dev-tools comprovant que no es donen moments amb frame-dropping provocant indicadors vermells. Documentació de referència per validació:

- <https://web.dev/articles/rail>
- <https://developer.chrome.com/docs/devtools/performance/reference#rendering>

L'empresa adjudicatària tindrà accés via VPN a

- Repositori BitBucket corporatiu (limitat al projecte en concret)
- Servidor CentOS 7 de l'entorn d'integració / test amb nodejs v16 i Alma Linux 9 amb nodejs v20
- Sonarqube (QA), Nexus (repositori llibreries), RPM repo (repositori RPMs)
- API de dades de la CCMA (entorn de test)
- API de lectura i escriptura de layouts (entorn de test)

A més, la CCMA proveirà d'una màquina virtual Vagrant CentOS 7 amb node 16 instal·lat per validar l'empaquetat amb maven.

A l'hora de configurar i arrencar el projecte es donarà suport per part de la CCMA S.A. per assegurar connectivitat i entorn preparat per treballar.

5. LLIURAMENT DE SOFTWARE I VALIDACIONS

L'entrega del desenvolupament ha de complir els següents requisits:

- S'entregarà el codi via el repositori de BitBucket corporatiu
- El codi ha de ser 100% funcional.
- L'execució dels scripts de compilació i empaquetats ha de funcionar en un entorn Windows 10 amb nodejs v20 i al jenkins de la CCMA per desplegar-ho a l'entorn de test. S'ha de poder aixecar l'aplicació i poder fer-la funcionar en un entorn de desenvolupament windows (dins de la CCMA o amb VPN)
- El codi ha de complir criteris de clean code (noms de variables i funcions descriptives, funcions curtes amb pocs arguments d'entrada, etc).
- El rendiment de les accions de mostrar noves capes i reordenació de components ha de ser correcte revisant les eines de Core Web Vitals
- Es farà una sessió de traspàs de coneixement a un equip tècnic de la CCMA S.A. de manera telemàtica (via MS Teams), màxim 2h de durada.

La validació final es farà en un equip de desenvolupament, entorn Windows 10 amb node 20 instal·lat, i comprovant el correcte funcionament de les funcionalitats:

- Aixecant l'entorn per desenvolupar i fent una prova senzilla (per exemple, reordenar blocs) i veure que els canvis es mostren en els layouts.
- Generant el paquet RPM des del Jenkins corporatiu, sense errors, i desplegant-ho a l'entorn d'integració / test, validant les funcionalitats descrites a l'apartat de desenvolupament d'aquest mateix document.
- Passant les proves de rendiment amb Core Web Vitals

Tot el codi entregat serà propietat de CCMA per poder ser evolucionat posteriorment.

6. SEGUIMENT DEL PROJECTE

El desenvolupament es farà íntegre a l'entorn de l'empresa adjudicatària, que haurà d'anar pujant el codi periòdicament al servidor BitBucket corporatiu de la CCMA S.A.,

com a mínim un cop per setmana durant la duració del desenvolupament. Després de l'entrega final, es farà una sessió explicativa (màxim 2 hores) de com s'ha estructurat el projecte on es farà la validació final.

Es faran reunions telemàtiques d'arrencada i seguiment del projecte convocades per la CCMA S.A. quan es consideri necessari, com a mínim 1 d'arrencada i 1 cada 3 setmanes de seguiment a partir de l'inici del projecte.

El contracte finalitzarà a l'entrega dels desenvolupaments inclosos en el projecte, degudament validats per la CCMA,S.A. en un termini estimat de 4 mesos.

Un cop l'equip d'Enginyeria de la CCMA, S.A. hagi validat i acceptat l'entrega final de l'aplicació, hi haurà un període de 2 mesos per fer un màxim de 4h de consultes telemàtiques (mail, teams) per resoldre dubtes que puguin sorgir durant l'exploració per part de l'equip tècnic de la CCMA.

7. PROCÉS DE FACTURACIÓ

L'adjudicatari facturarà un 20% a l'inici del projecte, un 30% més després de l'entrega dels 4 primers punts del desenvolupament (3.1, 3.2, 3.3 i 3.4) i el 50% restant a la finalització del projecte.

Les factures es pagaran d'acord amb la clàusula DIVUITENA del plec administratiu.

8. IDIOMES ADMESOS PER LA RECEPCIÓ D'OFERTES I LA COMUNICACIÓ

El licitador haurà de presentar la seva oferta en català. Les comunicacions entre CCMA,SA i el licitador, només s'acceptaran en català i castellà.

Un cop signat el contracte, la documentació que es derivi de l'execució del subministrament, caldrà que sigui en català.

9. REUNIÓN EXPLICATIVA A LES INSTAL·LACIONS DE LA CCMA,SA

Les empreses licitadores hauran d'assistir a una reunió explicativa a les instal·lacions de CCMA, SA., a la seu de la televisió a Sant Joan Despí.

A tal efecte, hauran de confirmar, per correu electrònic i abans del **dia 24 de juny de 2024**, la seva assistència a l'adreça: administracióiserveiscompres@ccma.cat indicant a l'assumpte del correu: **CPO 2406OB01** assistència a la reunió explicativa.

En aquest correu haurà de figurar, necessàriament, el nom de l'empresa, el nom i el número de DNI de la persona o persones que hi assistiran (màxim dues), i el nom de la persona de contacte, el seu telèfon i l'adreça de correu electrònic.

La visita tindrà lloc **el dimarts dia 25 de juny de 2024 a les 10:00 hores** a les instal·lacions de televisió de la CCMA, SA a Sant Joan Despí, al carrer de Oriol Martorell 1.

Qualsevol informació facilitada a la visita que complementi i actualitzi la informació continguda en aquest plec de prescripcions tècniques haurà de ser tinguda en compte.

Sant Joan Despí, juny de 2024