

Sistema d'Informació de Gestió d'Activitats i Recursos per Operació i Sistemes de la FIB (GAROS)

Memòria Final

Treball de Final de Grau - Modalitat B
Especialització en Sistemes d'Informació
Presentació: Entre el 27 Juny a 1 de Juliol de 2022

20 de maig de 2022

GEI Curs 21-22 Q2

Autor: Nahúm Manuel Martín Vegas

Directora: Sandra Marsà Hernández

Ponent: Joan Antoni Pastor Collado



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Facultat d'Informàtica de Barcelona



El més sincer agraïment a totes les persones que han col·laborat en aquest projecte per a que es dogués a terme. A la directora **Sandra Marsà**, al ponent **Joan Antoni Pastor**, als companys d'operació **Elvira Pallàs** i **Ignasi Herranz** per les seves valoracions i propostes, i a **Jaume Moral** per la seva ajuda en molts àmbits del projecte.

Resum

Existeixen llocs de treball on s'ha de supervisar i vigilar diversos sistemes d'informació a la vegada i molts d'aquests sistemes no donen les facilitats per ser supervisats en grup. Això vol dir que si tenim diversos sistemes d'informació i necessitem veure canvis o avisos a cada un d'ells hem de anar canviant constantment d'entorn a entorn a revisar aquests canvis o avisos. Un mètode antiquat de rebre avisos en temps real són els correus electrònics. Per cada activitat que requereixi intervenció o coneixement rebem un correu electrònic informant d'allò i acumulant-se a la bústia. Actualment s'estilen els sistemes de «notificacions», utilitats al sistema operatiu (escriptori o mòbil) que ens avisa de forma visual i/o acústica de que tenim algun tipus d'avís. Aquests mètodes tenen l'inconvenient de que funcionen només per aplicacions instal·lades al propi sistema i no per sistemes d'informació via web.

Aquest projecte ve a donar solució a aquest tipus de problema, notificar i mostra informació de diferents sistemes d'informació en un sol entorn visual que ens pugui ajudar a estar al cas de les activitats que succeeixen al lloc de treball. Un entorn clar, fàcil, dinàmic i automàtic que rebí i mostri la informació i ens notifiqui d'allò que és rellevant i ens resumeixi la informació que considerem important. A més, aquest sistema ajuda a accedir als sistemes on està connectat per fer les intervencions pertinents.

L'entorn en el que el projecte està concebut és en l'àrea de Operació i Sistemes de la Facultat d'Informàtica de Barcelona (FIB), dintre de la Unitat Transversal de Gestió de l'Àmbit TIC del Campus Nord (UTGCNTIC). Utilitza eines, protocols i entorns de la universitat per conceptualitzar l'entorn visual i les comunicacions però el concepte global és fàcilment exportable a qualsevol àmbit on sigui necessari.

En l'àmbit tècnic podem resumir que està concebut en un sistema Front-end i Back-end dintre de contenidors (Docker amb Linux i servidors web NGINX). Això es denomina com «Conceptualització en micro-serveis» on es separa l'entorn visual (Front-end) de l'entorn proveïdor i gestor de dades (Back-end). Son les arquitectures que es fan servir avui dia per estructurar un Sistema d'Informació per petit que sigui per a que tingui les característiques de versatilitat, seguretat, ampliabletat i interconnectabilitat. Més en concret tècnicament parlant, per al Front-end es fa servir Vue.js amb llenguatge JavaScript dintre d'una interfície marc de la UPC, i al Back-end es fa servir Django amb llenguatge Python. Pel que fa a les dades, es fa servir un servidor MariaDB pre-existent on s'allotja la base de dades d'aquest Sistema.

Així doncs, aquest projecte formalitza i construeix un PMV^[17] (Producte Mínim Viable o MVP en anglès) d'on partir i ampliar un sistema que pot ser tant gran com puguem concebre'l, ja que es pot ampliar amb moltes altres connexions a serveis i emmarcar a molts altres departaments tant dins com fora de la UPC.

Índex

Resum	3
1. Introducció i contextualització	8
1.1 Context	8
1.1.1 Introducció	8
1.1.2 Problema	8
1.1.3 Actors implicats	8
1.2. Justificació	10
1.2.1 Situació actual	10
1.2.2 Tendència global dels SI	11
1.3. Visió del projecte	11
1.3.1 Alternatives plantejades	11
1.3.2 Visió final	12
2. Planificació del projecte	13
2.1. Abast	13
2.1.1 Objectiu general	13
2.1.2 Objectius PMV	13
2.1.3 Requisits inicials	13
2.1.3.1 Requisits funcionals	13
2.1.3.2 Requisits no funcionals	14
2.1.4 Objectius de futur	14
2.1.5 Metodologia i rigor	15
2.1.5.1 Iteracions periòdiques	15
2.1.5.2 Eines de desenvolupament	16
2.1.5.3 Alternatives descartades	16
2.1.5.4 Mètode de validació	17
2.2. Tasques i planificació temporal	18
2.2.1 Definició de tasques	18
2.2.2 Hores, dependències i recursos per tasca	19
2.2.3 Diagrama de dependències	21
2.2.4 Gantt	22
2.2.5 Riscos	24
2.3. Pressupost	25
2.3.1 Identificació i estimació de costos	25
2.3.1.1 Recursos Humans	25
2.3.1.2 Recursos materials	26

2.3.1.3 Amortitzacions	27
2.3.1.4 Contingències i imprevistos	28
2.3.1.5 Pressupost Final	28
2.3.2 Control de gestió	29
2.4. Sostenibilitat	30
2.4.2 Dimensió econòmica	30
2.4.3 Dimensió ambiental	30
2.4.4 Dimensió Social	30
3. Desenvolupament	31
3.1. Tecnologies, eines i coneixements aplicats	31
3.1.1 Llenguatges de programació	31
3.1.2 Software extern i Frameworks	31
3.1.3 IDE's, editors i eines	33
3.2. Anàlisi de requisits i especificació final	34
3.2.1 Requisits funcionals	34
3.2.2 Diagrama de casos d'ús	34
3.2.3 Requisits no funcionals	37
3.2.4 Esquema de serveis	38
3.3. Disseny de projecte	39
3.3.1 Arquitectura i components	39
3.3.1.1 Contenedors	39
3.3.1.2 Components i mòduls	43
3.3.1.3 Base de dades	46
3.3.1.4 Seguretat	51
3.3.2 Patrons i disseny	52
3.3.3 Interfície	54
3.4 Experiència d'usuari	55
3.4.1 Accés i primera vista	55
3.4.1 Calendaris	57
3.4.1 Tiquets	59
3.4.1 Préstecs	60
3.4.1 Equips	62
4. Conclusions	64
4.1 Assoliment dels objectius inicials	64
4.2 Assoliment dels objectius personals	64
4.3 Problemes o dificultats	65
4.4 Feina futura	65

5. Bibliografia

Figures

Figura 1: Concepte de sprints al mètode Agile	15
Figura 2: Concepte d'iteració a Extreme Programming (XP)	16
Figura 3: Diagrama de dependències	21
Figura 4: Diagrama de Gantt amb les tasques detallades a partir del 28 de març	23
Figura 5: Diagrama de casos d'ús	35
Figura 6: Esquema conceptual de servei	38
Figura 7: Esquema de contenidors de Front-End	39
Figura 8: Esquema de contenidors de Back-End	41
Figura 9: Esquema de components i connexions entre Front-End i Back-End	43
Figura 10: Esquema de base de dades	46
Figura 11: Diagrama Model-Vista-Controlador	52
Figura 12: Diagrama Agregator	52
Figura 13: Distribució de la interfície	54
Figura 14: Autenticació SSOLab	55
Figura 15: Distribució dels elements del panell	56
Figura 16: Visual de la secció de Calendaris	57
Figura 17: Visual de la sub-secció de tots els calendaris	57
Figura 18: Visual de la secció de Tiquets	59
Figura 19: Visual de la secció de Préstecs a la sub-secció de Tots	60
Figura 20: Visual de la secció de Préstecs a la sub-secció de Canvis	61
Figura 21: Visual de la secció d'Equips	62
Figura 22: Visual de la secció d'Equips a la sub-secció de l'Historial	63

Taules

Taula 1: Resum de tasques del projecte	18
Taula 2: Costos en hores, dependències i recursos de les tasques	19
Taula 3: Temporització del sprints i la gestió del projecte G1	22
Taula 4: Temporització de les tasques sobre els sprints planificats i el temps de Gestió de Projecte Gx	22
Taula 5: Salaris dels treballadors implicats	25
Taula 6: Costos per treballador i totals en recursos humans	26
Taula 7: Costos de material per lloc de treball i total	27
Taula 8: Costos amb contingències	28
Taula 9: Costos amb imprevistos	28
Taula 10: Costos totals	28
Taula 11: Contingut del fitxer docker-compose.pro.yml del Front-End	39

Taula 12: Contingut del fitxer Dockerfile.pro del Front-End	40
Taula 13: Contingut del fitxer docker-compose.pro.yml del Back-End	41
Taula 14: Contingut del fitxer Dockerfile.api a la esquerra i Dockerfile.proxy a la dreta	42
Taula 15: Taules de la base de dades per al mòdul de Préstecs	47
Taula 16: Taules de la base de dades per al mòdul d'Equips	48
Taula 17: Taules de la base de dades per al mòdul de Tiquets	49
Taula 18: Taula de la base de dades per al mòdul de Calendari	50

1. Introducció i contextualització

1.1 Context

1.1.1 Introducció

Sobre l'any 2017, el consell de govern de la UPC va prendre la decisió de re-assignar alguns departaments de les facultats i la gestió dels equipaments de la universitat^[1]. Un d'aquells moviments implicava que les branques d'Operació i Sistemes de la FIB, que abans formaven part de la branca inLab FIB^[2]; ara formarien part de la UTG CNTIC^[3] (Unitat Transversal de Gestió de l'àmbit TIC del Campus Nord). Aquests canvis impliquen que algunes eines de gestió existents al període inLab s'han replicat per ambdós departaments o directament s'estan compartint.

Tenim llavors una re-distribució de departaments amb recursos que es van escollir a un context diferent i uns criteris que no responen a la situació actual. Unes eines de treball que, si bé són funcionals, requereixen actualitzacions i adequacions als nous àmbits i requeriments distributius.

1.1.2 Problema

Com es pot intuir a la introducció anterior, a Operació i Sistemes tenim un nombre considerable de sistemes d'informació on accedir per fer diferents tipus de tasques: un per inventari, un per tickets, un per documentació, quatre per sistemes operatius d'aules, un per préstecs i assignació d'aules, etc. Una amalgama de sistemes d'informació compartits amb altres departaments tots independents per diferents motius.

Per tant, resulta necessari un sistema que connecti amb tots els altres sistemes d'informació, informi de canvis o alertes a algun d'ells, mostri informació organitzada de la forma més adient al lloc de treball i que enllaci aquesta informació rebuda amb la informació concreta per enviar directament a l'usuari al sistema pertinent amb la dada o registre concret. I que també organitzi tota aquesta informació de forma personalitzada. Tot això des d'un sol entorn visual àgil, segur, ampliable i personalitzable per a cada usuari.

Amb aquesta premisa resulta clar que necessitem un nou sistema de tipus Web, com marquen els temps actuals, amb una petita base de dades per guardar la informació temporal rebuda i la dels usuaris, i una interfície de comunicació amb els sistemes requerits.

1.1.3 Actors implicats

Aquest treball de final de grau (TFG) es realitzarà en modalitat B i ha estat plantejat com projecte intern de la UTG CNTIC^[3]. L'equip que treballarà per realitzar aquest projecte serà:

- Nahum Manuel Martín Vegas. Estudiant i desenvolupador del Projecte a mes de Personal d'Administració i Serveis (PAS) al departament Operació de la Facultat d'Informàtica de Barcelona (FIB - UPC).
- Sandra Marsà Hernández. Directora del projecte i Personal d'Administració i Serveis (PAS) al departament de Sistemes de la Facultat d'Informàtica de Barcelona (FIB - UPC).

- Joan Antoni Pastor Collado, Ponent del Projecte a més de Professor en diverses assignatures i càrrecs directius a la Facultat d'Informàtica de Barcelona (FIB - UPC).

Les persones afectades pel problema són tot el personal de Sistemes i Operació. Però per aportar una solució a això caldrà la implicació de responsable d'altres departaments que facilitin l'accés a les dades que requerirà el nou sistema d'informació que es planteja en aquest projecte. Entre ells els responsables de les aplicacions de Racó^[4], gestió de tickets^[5], actual aplicació de projectes Redmine^[6], etc.

1.2. Justificació

1.2.1 Situació actual

Les eines com Redmine^[6] es van escollir al seu dia amb criteri d'utilització per l'inLab^[2], molt centrades en la gestió de projectes i com utilitat secundària la documentació per Operació i Sistemes; la qual està tota creada amb text i hipervincles sense cap tipus de menú àgil de cerca i organització. Per trobar un document només hi han dues vies, recordar-se de l'alçada on estava si ja el coneixes o fer una cerca per paraules si se sap quin nom tenia. A més, si l'escrit de Redmine és massa llarg es troben molts problemes per gestionar el text, l'editor de text integrat és antic i no funciona adequadament. El calendari d'absències del servei d'Operació també es troba a Redmine i cada any s'ha de referir nou manualment, ja que es fa amb l'editor de text integrat.

També es troba que eines com el Racó^[4] tenen utilitats per sistemes i operació. Es van crear allà perquè ja era un entorn funcional i no requeria crear un de nou amb còpia de dades d'usuaris, sobretot per coses com el registre de préstecs on es requeria accés al llistat d'alumnes i professors. Es troba per tant que si es volen afegir funcionalitats o corregir falles d'aquestes funcions per Sistemes i Operació s'han de demanar al departament que gestiona el Racó. També es troba que el Racó, al ser una plataforma d'ús extern per a estudiants i professors amb un disseny específic no aporta la llibertat d'afegir components interactius amb la velocitat i facilitat que altres mètodes avui dia gaudeixen.

Es pot comentar també altres serveis de registre com l'Inventari que tenen el seu propi portal i que no està enllaçat amb el gràfic d'aules que hi ha al Racó ni amb el sistema de gestió de Sistemes Operatius d'aules OpenGNSys. Tres portals d'informació on apareixen el mateix equipament i que no estan enllaçats per enriquir-los amb l'intercanvi d'informació.

Per altre banda també indicar que per accedir als servidors s'està fent servir eines força convencionals i no hi ha un lloc central on poder gestionar l'accés a ells independentment dels seus sistemes de seguretat o metodologia d'accés remot.

A la vegada, es té un sistema de tickets^[5] per UTG que genera molt de correu per cada acció que es fa a cada un dels tickets als que es té accés. Una rebuda massiva d'informació molt poc estructurada.

Per aquests motius és important plantejar-se el tenir una plataforma nova per Sistemes i Operació que agrupi tota la informació possible dins d'una sola plataforma. Que tingui accés a totes les dades dels diferents sistemes d'informació per facilitar la feina dels usuaris. Que organitzi adequadament la informació que requereix el dia a dia i que sigui fàcil de fer servir així com segura.

Resulta impossible trobar una eina que contingui tots els connectors d'informació als diferents sistemes amb els que es treballa. Alguns d'aquests sistemes d'informació a connectar són productes comercials, altres de codi lliure i altres programari fet per la pròpia universitat i que només es fa servir en aquest entorn. Els seus orígens són tant heterogenis que la millor manera és plantejar-se una solució a mida.

1.2.2 Tendència global dels SI

En els darrers temps s'ha vist que la tendència dels sistemes d'informació canvien a una distribució de microserveis on cada funció o base de dades és independent del servei d'interacció (Front-end^[7]) que utilitza l'usuari per accedir a les dades i funcions. Ja es comentava al 2016 a revistes especialitzades <<Los desarrolladores están cambiando las arquitecturas de sistemas masivos en función de aplicaciones pequeñas e independientes que trabajan juntas para resolver problemas complejos>>^[8]. I al desembre de 2021 continuaven comentant la tendència de les aplicacions cap a microserveis <<Afrontar los retos de la adopción de microservicios a gran escala>>^[9]. El paradigma de microserveis permet atomitzar i controlar cadascuna de les funcions del sistema d'informació amb més seguretat. Tota aquesta atomització s'està suportant gràcies a les tecnologies de contenidors com Docker^[10] o Kubernetes^[11] que, juntament amb les tecnologies del núvol, conformen la tendència del sector de sistemes d'informació. Un petit pas cap a la migració a aquesta distribució atomitzada de recursos i funcions és el projecte que presentem que vol crear un Front-end connectat a tots els recursos possibles per integrar les diferents funcions de treball en un sol entorn per l'usuari especialitzat a cada usuari i al seu lloc de treball.

1.3. Visió del projecte

Per tal de desenvolupar de forma correcta i directa qualsevol projecte s'ha de tenir clara la idea i els conceptes implicats per endavant. Així podem marcar un camí per avançar que sigui el més clar i directe possible.

1.3.1 Alternatives plantejades

Com a concepte resulta impossible trobar una eina que ja faci tota la feina que nosaltres requerim, així que les qüestions que ens plantejem es basen principalment en quins Frameworks (marc de treball) escollim, quins llenguatges de programació, quina arquitectura de desenvolupament, etc.

Un Framework és un conjunt de regles i eines que marquen els procediments i normatives d'un treball.

Pel que fa a l'arquitectura, en aquests temps i mirant cap el futur es veu millor una estructura de contenidors amb diferenciació del Front-End^[7], el Back-End^[7] i la base de dades en front al concepte antiquat de sistema monolític que agrupa entorn visual, controlador i dades dintre de un mateix sistema operatiu. Si bé encara queden situacions on l'entorn monolític és l'adient, no ho és per aquest projecte ja que requereix una versatilitat i eficiència que no té el monòlit.

Inicialment es tenia la idea de fer servir un Framework d'administració però la universitat ja disposa d'un Framework comú d'entorn visual basat en Vue.js i JavaScript que ens dona un marc inicial pel que fa al Front-end. En quant al Back-end hi han diverses opcions però novament, la universitat ha agafat Django com l'entorn Back-End recomanable, opció que veiem del tot adient per aquest projecte. Pel que fa a la base de dades només es requereix que sigui una base de dades relacional, tot i que una vegada finalitzat el projecte també es podria haver utilitzat una base de dades no relacional

degut a la poca interconnectivitat de les dades. En aquest sentit la universitat ens ofereix una base de dades MariaDB que ja està en funcionament i on podem connectar el nostre projecte.

En definitiva, poques alternatives de cara al plantejament del projecte ja que la pròpia Universitat suggereix les eines més adients i modernes per al tipus de projecte que es vol desenvolupar. Les alternatives descartades es tracten més endavant, a l'apartat 2.1.5.3.

1.3.2 Visió final

El sistema concebut es basa en les tècniques de micro-serveis, contenidors que són com càpsules de software amb un sistema operatiu mínim que permeten copiar i restaurar cada servei de forma quasi bé immediata amb poques instruccions. Dividit en tres parts (Front-End^[7], Back-end^[7] i Base de dades), cada una es pot manegar i gestionar de forma independent fins i tot en mode de producció.

El Front-End es basa en un Framework de JavaScript denominat Vue.js 3 i que incorpora una plantilla de disseny de la UPC `upc-webapp-ui-vue3`. A més es fan servir plug-ins de visualització i tractament de dades com `Charts.js`^[12], `Bootstrap`^[13], `Vuex`^[14] i `Element-Plus`^[15] entre d'altres.

En quant al Back-end està basat en una versió base de Django^[27], un Framework de Python enfocat al serveis de REST API amb gestió d'usuaris i un mínim entorn web. Aquí també fem servir llibreries com `mysqlclient`, `apscheduler`, `django-jwt-oidc` o `suds-community` entre altres.

Pel que fa a la base de dades es fa servir un servidor MariaDB proporcionat per la pròpia universitat i que ells mateixos gestionen totes les qüestions de manteniment, seguretat i backups.

En definitiva, un conjunt d'eines molt actual que utilitza les tècniques més modernes en quant a l'oferiment de serveis web i que compleix sobradament els requeriments del sistema. A més, és important recalcar que tot el software escollit per al sistema d'informació és programari lliure i no té cap cost d'adquisició o quota de llicències. També cal recalcar que anar en consonància amb les aplicacions que la UPC ja desenvolupa afavoreix que qualsevol programador d'elles pugui entendre i mantenir aquesta mateixa aplicació.

Existeix un protocol de bones pràctiques i recomanacions per l'àmbit de les TIC que se'n diu ITIL^[16] (Information Technology Infrastructure Library). Ja està per la versió 4 i aquest projecte es pot encabir en alguns dels seus conceptes i recomanacions.

Proclama un model de producte o servei basat en quatre dimensions: Organització i persones, informació i tecnologia, socis i proveïdors, i fluxos de valor i processos.

Els seus principis es basen en: Enfocar-se al valor, començar des de on estem, Avançar iterativament i retroalimentar-se, col·laborar i promoure la visibilitat, visió holística, mantindre-ho simple i pràctic, i optimitzar i automatitzar.

Amb aquestes premisses podem encaixar aquest projecte com un factor d'optimització que simplifica i practifica la feina diària dels departaments, i que agafa una vista global dels serveis. També la podem emmarcar en les dimensions d'organització i persones, informació i tecnologia, i fluxos de valor i processos.

2. Planificació del projecte

2.1. Abast

2.1.1 Objectiu general

L'abast del projecte abarca inicialment Operació i Sistemes de la FIB (Facultat d'informàtica de Barcelona) encara que s'ha plantejat ampliar-ho a Campus Nord si es troba oportú; i si acaba sent un sistema desenvolupat per la pròpia facultat es podria considerar oferir-ho com a producte UPC.

2.1.2 Objectius PMV

Com que en tot projecte s'ha de plantejar una meta factible de la qual poder continuar creixent. Aquesta meta inicial és el PMV^[17] on es poden posar en producció funcionalitats bàsiques i es pot anar avaluant el nou sistema a mesura que s'utilitza.

El PMV contempla els següents objectius:

- Tenir perfils propis en base a la identitat digital de la UPC^[18].
- Connexió del nou sistema amb el Racó^[4] per administrar els préstecs de material. Afegint funcionalitats com historial, filtratge, alertes, etc.
- Crear un calendari compartit per millorar el que ara es fa servir a Redmine^[6] que pugui ser dinàmic, fàcil d'actualitzar i amb alertes.
- Traspasar la supervisió dels equips que hi ha actualment al Racó per un entorn més fiable i exacte del control de l'equipament d'aules (POA).
- Connexió del sistema amb el sistema de tickets^[5] de la UTG per tenir avisos diferenciats entre tickets nous i diferents tipus d'actualitzacions de tickets.

2.1.3 Requisits inicials

2.1.3.1 Requisits funcionals

Segons la metodologia Agile els requisits funcionals poden variar al llarg del desenvolupament en funció de les oportunitats o inconvenient que es vagin trobant.

- Un sistema integrat des d'on es puguin rebre avisos, programar notificacions i configurar avisos personalitzats amb totes les connexions possibles dels altres sistemes d'informació.
- Ha de tenir un perfil personalitzat per a cada usuari que pugui permetre veure totes les notificacions pendents i deixar de veure avisos d'aquelles que ja estan revisades.
- Per cada notificació ha de poder tenir un enllaç al sistema d'on prové l'avis per si cal veure detalls o realitzar accions a aquell sistema.
- Els usuaris han de poder accedir mitjançant la identitat digital de la UPC i accedir directament al seu perfil o dashboard.
- El dashboard principal ha de poder mostrar tota la informació rellevant de la feina diària.
- Si algun servei no està disponible ha de poder mostrar aquesta informació clarament i la resta que sí siguin disponibles no han de tenir inconvenient si no estan relacionats.

- Els avisos han de ser tant visuals com sonors podent-se desactivar si calen en funció de les preferències de cada usuari.
- La informació que trobem en taules ha de poder mostrar-se ordenada pel component que sigui necessari en l'ordre més convenient segons la necessitat. Ha d'estar ben distribuïda i amb opcions de personalització de vista.

2.1.3.2 Requisits no funcionals

Els requisits no funcionals difícilment poden variar al llarg dels diferents sprints. Són conceptes generals que es poden assumir des de un bon començament de la gestió del projecte.

- El sistema ha de ser àgil i funcional per a totes les funcions que tingui.
- L'entorn gràfic ha de complir uns mínims d'accessibilitat per a usuaris amb fatiga visual o problemes de visió comuns. Han d'haver opcions de configuració visual per aquests mínims d'accessibilitat variats.
- S'han de seguir els colors corporatius de la UPC sempre i quant no entrin en contradicció amb les necessitats d'accessibilitat visual.
- La comunicació entre serveis ha de ser directa i no ha d'haver necessitat de que l'usuari realitzi cap acció per a que la comunicació es realitzi, s'ha de mirar que les dades requerides s'actualitzin periòdicament de forma automàtica en la mesura del que sigui possible.
- El sistema no ha de interferir ni provocar cap tipus d'error a aquells serveis d'informació on es connecti. S'han de minimitzar al màxim les possibles modificacions de dades remotes.

2.1.4 Objectius de futur

Una vegada assolit el PMV^[17] es planificaran les següents funcionalitats:

- Creació de perfils i permisos de grups a fi de gestionar per defecte els permisos a cada secció del sistema.
- Traspasar i organitzar la documentació d'operació i sistemes de forma adequada facilitant l'actualització i modificació.
- Integració d'accessos als servidors i equipaments remots sigui amb enllaços externs o amb clients embeguts al sistema (SSH, RDP, VNC, ...) guardant bé la seguretat a tots ells.
- Connexió de monitorització d'aules amb l'inventari.
- Registre d'autenticació de cada equip d'aules a fi d'identificar el sistema operatiu i l'usuari que ha accedit en cada moment.
- Anàlisi de dades sobre utilització d'aules, sistemes operatius, utilització d'equips per assignatura, control d'usuaris fora d'hores de classe, etc.
- Gestió de projectes adequats a les necessitats i organització d'Operació i Sistemes millorant la operativa actual i esmenant les mancances de Redmine^[6].
- Generador automàtic de la memòria anual d'activitats per Operació i Sistemes.
- Connexió amb l'aplicació de recursos de UTGCNTIC^[3] basada en Booked Scheduler^[19].
- Un sistema de recollida i filtratge de logs dels diferents serveis que sigui accessible, fàcil d'utilitzar, amb filtrat dinàmic del registres i alertes personalitzables.

És molt probable que a mida que el projecte vagi agafant forma vagin sortint noves necessitats o inconvenients al desenvolupament que canviïn el programa previst inicial.

2.1.5 Metodologia i rigor

Per a tot projecte de sistemes d'informació un punt clau és definir com es treballarà, quina metodologia de desenvolupament de software existents, o noves, es faran servir. Aquest projecte s'ha plantejat per a que es pugui desenvolupar dintre del mateix departament amb un personal reduït, així que molts rols de la metodologia seran compartits per les mateixes persones.

2.1.5.1 Iteracions periòdiques

Un dels mètodes de desenvolupament més apropiats per aquest tipus de projecte és l'Agile^[20]. Consisteix en marcar micro objectius, habitualment setmanals o bi-setmanals, on en aquest període s'ha de poder assolir el repte plantejat. D'aquests períodes d'assoliment de micro-objectius setmanals o bi-setmanals s'anomenen sprints. Podem veure a la figura 1 com es duen a terme tots els processos de desenvolupament sprint a sprint. Aquest mètode ens permet poder re-plantejar els objectius generals en cas de trobar inconvenients o noves funcionalitats sortides durant un sprint.



Figura 1: Concepte de sprints al mètode Agile

Font: <https://kruschecompany.com/agile-software-development/>

D'entre tots els estils (frameworks) d'Agile que existeixen s'ha trobat que el Extreme Programming (XP)^[21] és el que més s'ajusta a la filosofia del projecte. Un mètode dinàmic, d'objectius curts i variants. A la figura 2 podem apreciar el diagrama d'una iteració de l'estil Extreme Programming. Aquest mètode de treball ens permet desenvolupar ràpidament les funcionalitats bàsiques i adaptar-les als inconvenients que es poden anar trobant a mida que el projecte va agafant forma.

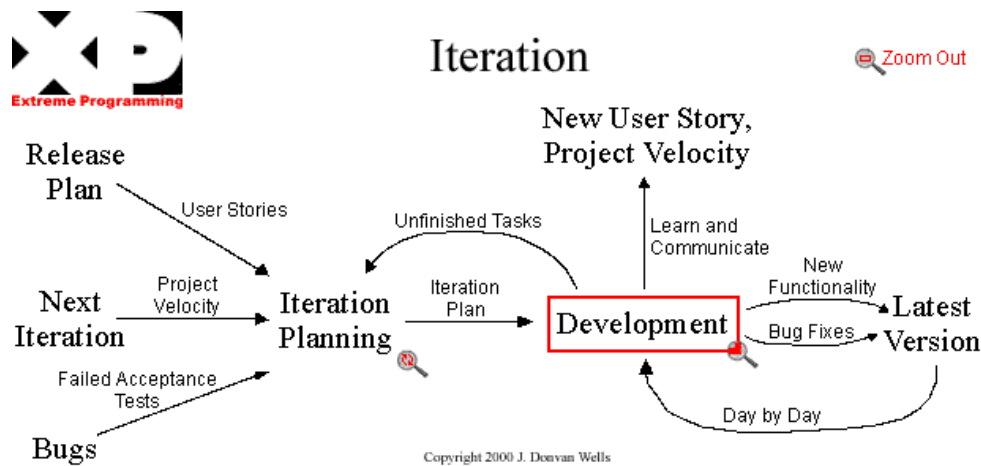


Figura 2: Concepte d'iteració a Extreme Programming (XP)
 Font: <http://www.extremeprogramming.org/>

2.1.5.2 Eines de desenvolupament

Degut a la dificultat de trobar un entorn amb connexions o plug-ins amb sistemes d'informació interns de la UPC s'ha optat per fer servir un Framework Front-end^[7] de programació per facilitar la implantació d'un entorn d'administració personalitzable. Per aquest cas s'ha optat per fer servir el Framework UPC webapp ui Vue 3^[22] basat en Vue.js^[23] amb Node.js^[24] i programat amb JavaScript^[25] per el Front-end on coses com taules, botons, menús, barres i etcétera ja venen definides i llestes per posar allà on es necessiti ràpidament i sense gaire programació. Per el Back-end^[7] s'ha decidit fer servir un llenguatge de programació conegut als departaments de Sistemes com és Python^[26]. Amb aquesta premisa s'ha escollit Django^[27] com servidor Back-end que compleix amb els requeriments esmentats.

Les eines de desenvolupament que es faran servir són un IDE^[28] com Microsoft Visual Studio CODE^[29] que a part de ser gratuït és un dels més utilitzats arreu del món. A més ens permet fer servir plug-ins d'execució web local per poder veure els resultats abans de pujar els canvis al servidor.

Per al control de versions i canvis es planteja fer servir GitLab^[30] de UPC pel fet de què és un entorn destinat a aquest efecte i on tots els interessats poden tenir accés amb el seu compte d'usuari de la UPC.

Per al control de projecte s'utilitzarà Redmine^[6], què és l'actual sistema de control de projectes.

2.1.5.3 Alternatives descartades

A l'hora d'escollir el Framework de Front-end^[7] s'ha decidit per UPC webapp ui Vue 3^[22] ja que ja porta de sèrie els colors corporatius i també es fa servir a altres aplicacions de la UPC. Les alternatives que s'havien estudiat eren AdminLTE^[31], CoreUI^[32] o Table^[33] basats en Bootstrap^[13], un Framework compatible amb Vue.js^[23]. Tots tres poden ser perfectament vàlids si l'opció escollida mostra inconvenients per al desenvolupament.

Pel que fa al Back-end^[7] s'ha escollit Django^[27] perquè compleix amb els requisits comentats anteriorment. A més integra totes les opcions que necessitem per guardar les dades personals dels usuaris i la seguretat requerida. Com alternativa podem trobar Flask^[34], que no te tots els complementos que necessitaríem però es poden afegir de forma externa si per algun inconvenient de

desenvolupament no podem ser servir la opció per defecte. Altres alternatives per a Back-end com Ruby on Rails^[35] o Laravel^[36] incompleixen moltes més condicions de les requerides per al personal de sistemes i desenvolupament, i per això es descarten directament i no es plantegen com alternatives tot i que podrien ser perfectament funcionals.

En quant a les eines de desenvolupament és més una qüestió de preferències del programador. Tot i que comptar amb un IDE que ajudi a informar de falles i errades del codi com per automatitzar la previsualització local sempre és un afegit que pot aportar una velocitat de desenvolupament al projecte. IDEs alternatius a Microsoft Visual Studio CODE^[29] hi han molts, així com programar directament amb editor de text però l'escollit és un dels més utilitzats per tots els programadors arreu del món per qualsevol llenguatge de programació o sistema operatiu.

GitLab^[30] de UPC és on resideix l'actual codi de UPC webapp ui Vue 3. Per tant és el lloc ideal pel repositori del projecte. Si fos necessari es podria portar a altres repositoris externs a la UPC com GitHub^[37] o GitLab^[30] públic.

En quant a alternatives a Redmine^[6] podem trobar Microsoft Project^[38] o Trello^[39], totes dues de pagament. Però si ja es té un software de gestió de projectes que coneix la empresa no té gaire sentit plantejar una alternativa de moment.

2.1.5.4 Mètode de validació

Amb la metodologia de treball escollit podem fer avaluacions contínues dels micro-objectius de cada sprint tant amb la gent directament involucrada com amb els voluntaris dels departaments que vulguin participar de les proves de funcionament.

El seguiment del projecte es durà a per les persones involucrades de forma directa:

- Nahum Manuel Martín Vegas que farà els rols de gestor de projecte, programador, analista d'interfícies gràfiques i enginyer de sistemes.
- Sandra Marsà Hernández que farà els rols de Directora de projecte supervisant la feina i el compliment dels temps i objectius previstos.
- Joan Antoni Pastor Collado que farà el rol de Ponent i revisarà la documentació i l'estat del projecte periòdicament.

Es marcaran reunions setmanals al final de cada sprint per revisar els objectius realitzats i plantejar els objectius del següent sprint segons la planificació o aplicant canvis si es veu oportú i necessari.

2.2. Tasques i planificació temporal

Les tasques s'han identificat a nivell global marcant els objectius, sobretot enfocats als requisits funcionals com són les connexions amb les diferents aplicacions i serveis. Les tasques identificades amb Gx fan referència a la Gestió del Projecte, les identificades amb Px fan referència a la instal·lació i configuració del programari, les identificades amb Dx fan referència a Disseny d'interfície, i les identificades amb Cx fan referència a les diferents connexions amb els servidors o serveis que ha realitzar l'aplicatiu.

2.2.1 Definició de tasques

Les tasques es nombren amb dos caràcters, un que identifica el tipus i altre que la enumera. A la taula 1 podem veure també el títol i una petita descripció de la feina que comporta cada tasca.

G1	Gestió de projecte Contextualització, abast definició de funcionalitats i requisits, definició de tasques a realitzar, planificació de tasques, pla econòmic, informe de sostenibilitat i reunions amb Stakeholders.
G2	Documentació de projecte Documentació del projecte durant la gestió i tots els sprints. S'estima que es faran servir 3 hores de documentació per a cada sprint i 20h a la gestió del projecte. La documentació als diferents sprints es farà paral·lelament amb les tasques a fer.
G3	Reunions de control Reunions de control programades al final de cada sprint per revisar objectius complerts i possibles variacions. Cada sprint tindrà una reunió d'aproximadament 1 hora.
P1	Instal·lació de contenidors de Front-end i Back-end Preparació i instal·lació dels contenidors de Front-end i Back-end per als servidors de desenvolupament dedicats. També s'habilitaran els permisos de xarxa pertinents per poder treballar al desenvolupament. També serà el període per avaluar les versions de servidors que es faran servir al projecte.
P2	Configuració de UPC webapp ui Vue 3 i Servei API Django Adaptar el Framework UPC webapp ui Vue 3 per al treball de disseny i configurar el servidor Python com API de consulta de dades de l'aplicació.
P3	Preparació de l'entorn de desenvolupament, repositori i area de testing Instal·lació del programes de desenvolupament com l'IDE o creació del repositori on s'allotjaran els codis del desenvolupament. També preparar l'àrea de proves local per a una visualització més ràpida.
D1	Disseny i distribució de l'entorn Front-end Disseny de tota la interfície gràfica i treball de distribució de les diferents funcions dintre del marc visual del Front-end.

D2	Disseny i distribució de les funcions connectades Disseny de l'entorn visual de cada funció. Estimació de 5 hores per a cada funció, a cada sprint dedicat a connectivitat. Les dependències esdevenen segons es realitza cada funció (C1, C2, C3, C4, C5) de connexió i el disseny global de l'aplicació fet a la tasca D1.
C1	Connexió amb aplicació de login amb Identitat Digital UPC Connexió amb Identitat Digital de la UPC i obtenció de dades d'usuari de cara a la utilització al programari.
C2	Connectar Calendaris d'Operació Connexió amb calendaris de Google Calendar i traspàs de la informació actual, així com la metodologia i nomenclatura per als calendaris compartits.
C3	Connectar informació de la API de POA Connexió amb l'aplicació POA de control d'equips per a una visualització i control adients dins de l'aplicació.
C4	Connectar API de Tickets Connexió amb l'aplicació de tickets i programació de les diferents tipus d'alertes que es poden obtenir a partir de les dades rebudes de la API.
C5	Connectar Reserves i Préstecs per API del Racó Connexió amb l'aplicació de reserves i préstecs del Racó per a una ordenació i notificació d'alertes personalitzades, així com mecanismes de cerca i categorització.

Taula 1: Resum de tasques del projecte

Font: Elaboració pròpia

2.2.2 Hores, dependències i recursos per tasca

A la taula 2 podem veure les hores que previsiblement necessitarà cada tasca, les seves dependències i els recursos necessaris per dur-les a terme. Cal remarcar que la tasca de documentació es farà en paral·lel amb d'altres i per aquest motiu, tot i que no tenen un punt d'inici i final continu, es poden dur a terme durant un sprint a diferents moments segons sigui convenient. La suma total d'hores és de 302 repartit en una dedicació aproximada de 4 hores diàries, unes 20 hores setmanals o per sprint.

Tasca	Hores	Dependències	Recursos
G1	70	-	Recursos humans: Cap de projecte Recursos materials: Ordinador amb internet, Google Docs.
G2	92	Tota la resta	Recursos humans: Cap de projecte Recursos materials: Ordinador amb internet, Google Docs.
G3	8	P1, P3, D1, D2	Recursos humans: Cap de projecte Recursos materials: Ordinador amb internet, Google Docs.

P1	16	G1	Recursos humans: Programador i administrador de sistemes Recursos materials: Ordinador amb xarxa, Servidor de desenvolupament, software a instal·lar.
P2	10	P1	Recursos humans: Programador Recursos materials: Ordinador amb xarxa
P3	8	P2	Recursos humans: Programador Recursos materials: Ordinador amb xarxa
D1	18	P3	Recursos humans: Programador i Analista d'interfícies gràfiques Recursos materials: Ordinador amb xarxa
D2	20	D1, C1, C2, C3, C4, C5	Recursos humans: Programador i Analista d'interfícies gràfiques Recursos materials: Ordinador amb xarxa
C1	12	P3	Recursos humans: Programador Recursos materials: Ordinador amb xarxa
C2	12	P3	Recursos humans: Programador i Analista d'interfícies gràfiques. Recursos materials: Ordinador amb xarxa
C3	12	P3	Recursos humans: Programador i Analista d'interfícies gràfiques Recursos materials: Ordinador amb xarxa
C4	12	P3	Recursos humans: Programador i Analista d'interfícies gràfiques Recursos materials: Ordinador amb xarxa
C5	12	P3	Recursos humans: Programador i Analista d'interfícies gràfiques Recursos materials: Ordinador amb xarxa

Taula 2: Costos en hores, dependències i recursos de les tasques

Font: Elaboració pròpia

2.2.3 Diagrama de dependències

A la figura 3 podem veure el diagrama de dependències on els nodes en vermell són les tasques recurrents que depenen de diverses tasques i que estan repartides en diferents sprints amb nomenclatura afegida <<-Sx>> que fan referència al sprint concret on la x marca el número de sprint. La documentació de gestió del projecte es designa com <<G2-G1>>.

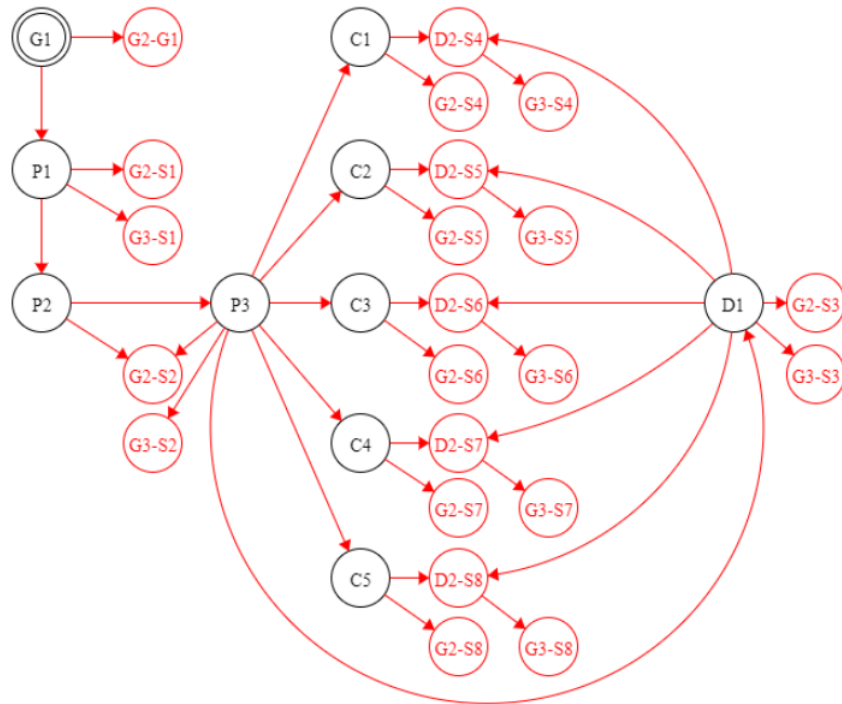


Figura 3: Diagrama de dependències
Font: Elaboració pròpia

2.2.4. Gantt

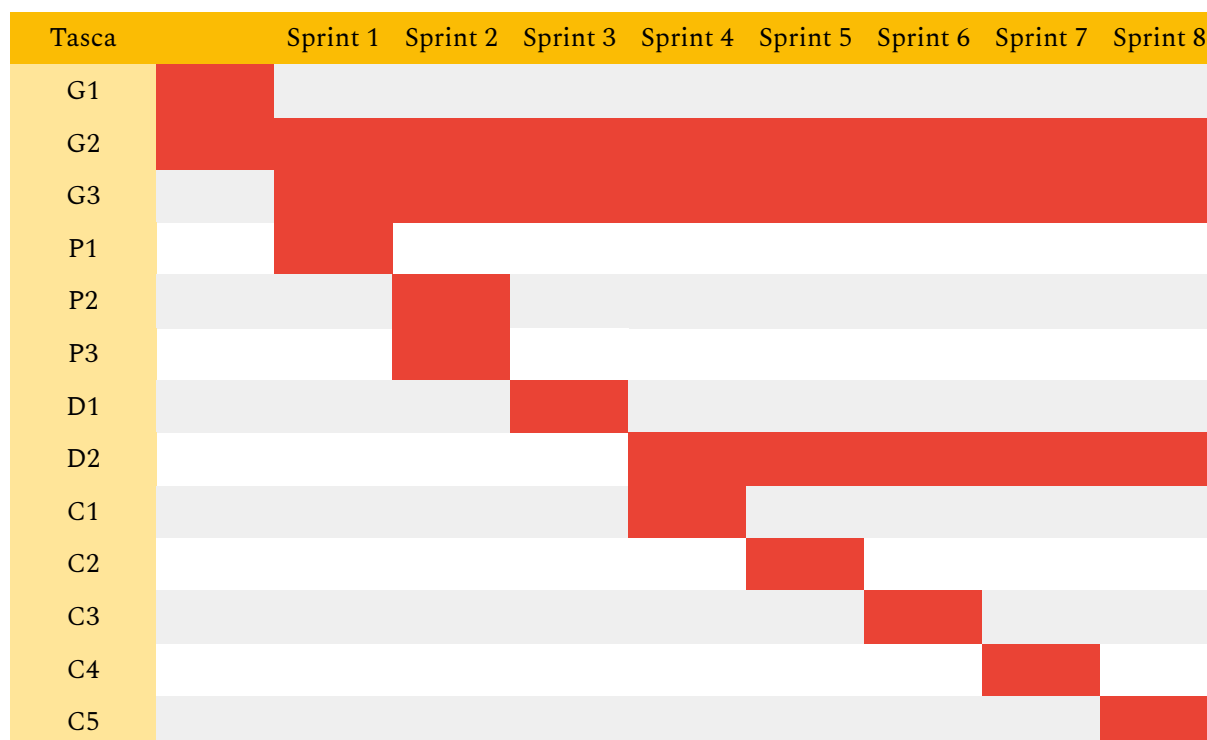
S'han definit els sprints i hem determinat la data d'inici i acabament de cada un. També hem definit el període necessari per realitzar la gestió del projecte G1.

Sprint:	G1	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Sprint 6	Sprint 7	Sprint 8
Dia d'inici	22/02	04/04	11/04	18/04	25/04	02/05	09/05	16/05	23/05
Dia final	01/04	08/04	15/04	22/04	29/04	06/05	13/05	20/05	27/05

Taula 3: Temporització del sprints i la gestió del projecte G1

Font: Elaboració pròpia

Un cop tenim els sprints definits ja podem mostrar l'estimació de realització de cada tasca definida per a cada sprint d'acord amb les dependències i el temps que es requereix per a cada una. Moltes de les tasques de connectivitat es podrien solapar però s'estima que el grup de programació serà força reduït i per aquest motiu es defineix com un sprint per cada una de les tasques de connectivitat. La documentació s'ha de anar fent a mida que es va desenvolupant, per això la tasca G2 apareix a tots els sprints i durant la tasca de Gestió del Projecte G1. Es determina que cada tasca de connectivitat requerirà certa part d'implantació al disseny de l'aplicació, concretament el que fa referència a D2, per això s'ha repartit per a cada un dels sprints necessaris.



Taula 4: Temporització de les tasques sobre els sprints planificats i el temps de Gestió de Projecte Gx

Font: Elaboració pròpia

Seguidament a la figura 3 es mostra el diagrama de Gantt més detallat a partir del 28 de març per a que els detalls es puguin veure adequadament. Del 22 de febrer al 1 de març es realitzen paral·lelament les tasques G1 i G2 tal i com es pot veure a la gràfica des del 28 de març. Les tasques que es realitzen a diferents sprints tenen la nomenclatura de la tasca seguit d'un guió i el sprint al que pertanyen.

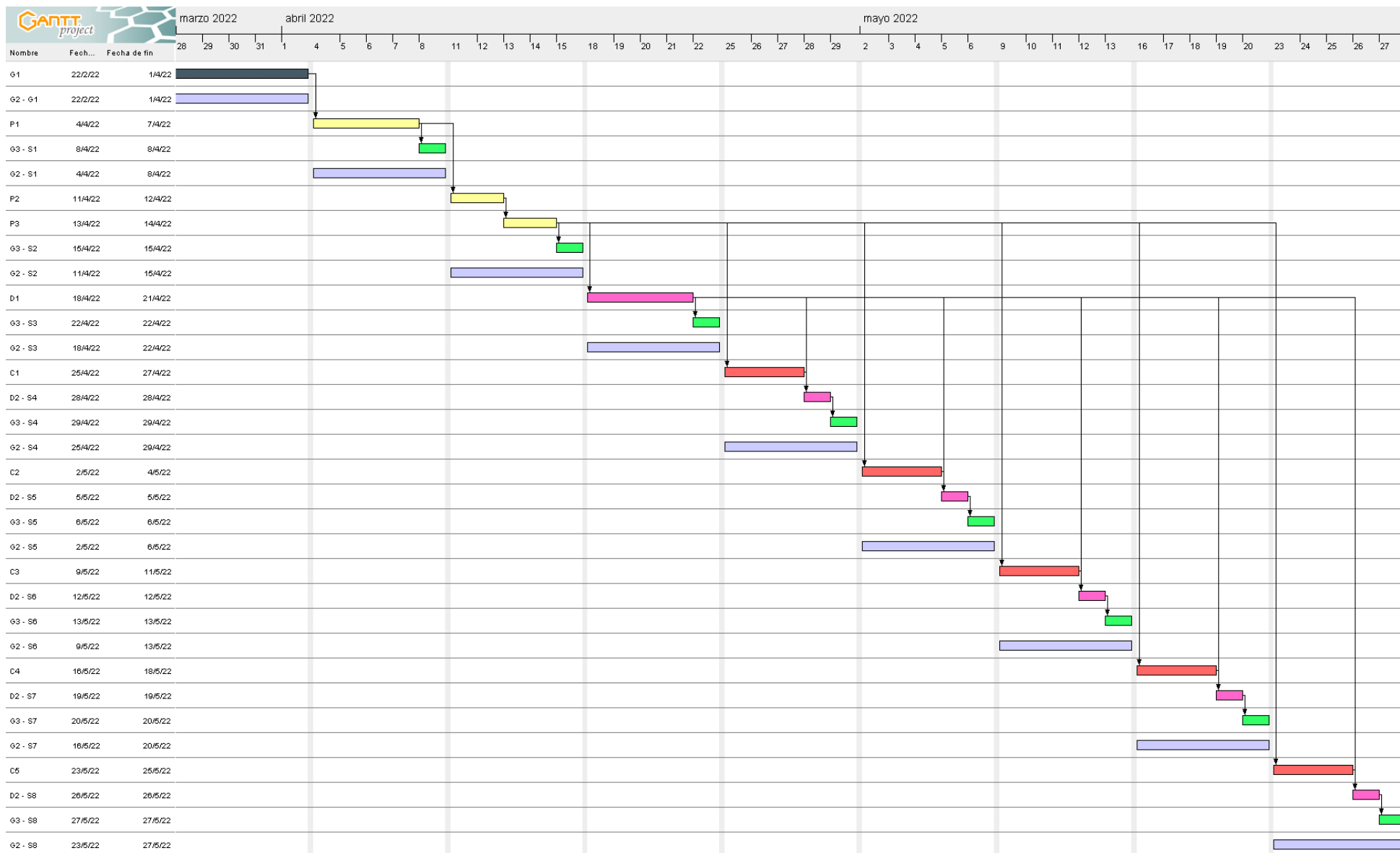


Figura 4: Diagrama de Gantt amb les tasques detallades a partir del 28 de març
 Font: Elaboració pròpia

2.2.5 Riscos

En quant a la base del projecte que són el Front-end^[7] i el Back-end^[7] els riscos que podem trobar són molt baixos. Són entorns aïllats i provats que estan dedicats per al desenvolupament, llevat d'errors de servidors o permisos de xarxa per connectar als diferents serveis no es poden trobar gaires inconvenients més. A més, les tecnologies de contenidors i el programari a utilitzar estan sobradament provats i no s'esperen incompatibilitats de cap tipus. Tot i així, si es trobessin inconvenients amb el programari escollit hi han alternatives idèntiques de sobres per canviar sense cap canvi en la planificació.

Sobre les tasques de disseny i connectivitat s'ha estimat el temps per sobre perquè és conegut que difícilment les coses funcionen a la primera i l'intercanvi d'informació i errors de connectivitat sempre es trobaran. També es poden trobar diferents criteris de disseny i ser possible que es necessitin consultes dels usuaris per a trobar consens sobre el disseny.

És força probable que trobem retards en la comunicació amb els diferents Stakeholders que han de participar en l'intercanvi d'informació però es té present que és possible que algunes tasques de connectivitat s'hauran de solapar a algunes hores. En tota aquesta part (cap2) trobo a faltar una reflexió de com han anat les coses, s'han fet les reunions previstes, ha hagut desviacions, quins problemes t'has trobat, etc.

2.3. Pressupost

2.3.1 Identificació i estimació de costos

Per a cada projecte informàtic podem diferenciar dos tipus de costos generals, els materials i el humans. Per aquest projecte considerem que tenim quatre perfils de persones que han de formar part del desenvolupament del projecte: Cap de Projecte (CP), Programador (PR), Enginyer de Sistemes (ES) i Analista d'Interfícies Gràfiques (IG). Cada un d'ells participa en un període diferent del projecte amb responsabilitats ben acotades. En quant als costos materials tant físic com digital no es tindrà en compte l'amortització feta dels materials ja que es pot donar el cas que fem servir material adquirit per una altra funció o projecte previ, però sí que es reflectirà si aquest material té funcions compartides amb altres projectes o funcions per establir el cost parcial.

2.3.1.1 Recursos Humans

En quant als recursos humans tenim que les persones que intervenen al projecte són:

El Cap de projecte s'encarregarà de la gestió del projecte i supervisarà que els sprints i objectius s'estan assolint. El programador es dedicarà a fer tota la lògica de l'aplicació segons les indicacions esmentades a la documentació del projecte. L'administrador de sistemes participarà en la posada en marxa dels contenidors de Front-end^[7] i Back-end^[7] junt amb el programador que indicarà els requeriments de l'aplicació. L'analista d'interfícies gràfiques dissenyarà la distribució d'elements de l'aplicació i supervisarà la funcionalitat a nivell d'usuari dels diferents components de l'aplicació.

Segons la web Glassdoor els sous base promig dels treballadors a Barcelona son:

Lloc de treball	Brut anual (€/any)	Brut anual amb SS (€/any)	Hora Bruta amb SS (€/h)
Cap de projecte (CP) Project Director	65.769€	85.500€	41,11€
Programador (Front-end i Back-end) (PR) Full Stack Developer	36.731€	47.750€	22,96€
Enginyer de Sistemes (ER) Systems Engineer	43.365€	56.375€	27,10€
Analista d'Interfícies Gràfiques (IG) Interaction Designer	48.077€	62.500€	30,05€

Taula 5: Salaris dels treballadors implicats

Font: <https://www.glassdoor.es>

Per a cada tasca hi han assignats alguns d'aquests treballadors i participaran en funció de les necessitats de cada tasca. També és necessari conèixer els totals de costos tant per lloc de treball com els costos totals en recursos humans. Tot això ho tenim reflexat a la següent taula:

Tasques	Hores	Costos per treballador				Totals per tasques
		CP	PR	ER	IG	
G1	70	2.877,70 €				2.877,70 €
G2	34	1.397,74 €				1.397,74 €
P1	16	367,36 €		433,60 €		800,96 €
P2	12	275,52 €				275,52 €
P3	8	183,68 €				183,68 €
D1	20	459,20 €		601,00 €		1.060,20 €
D2	25	574,00 €		574,00 €		1.148,00 €
C1	12	275,52 €				275,52 €
C2	12	275,52 €				275,52 €
C3	12	275,52 €				275,52 €
C4	12	275,52 €				275,52 €
C5	12	275,52 €				275,52 €
Totals:	245	4.275,44 €	3.237,36 €	433,60 €	1.175,00 €	9.121,40 €

Taula 6: Costos per treballador i totals en recursos humans

Font: Elaboració pròpia

2.3.1.2 Recursos materials

En quant al material físic que es necessita per desenvolupar el projecte tenim dos conceptes de requeriments físics, els servidors on rodaran els serveis i els llocs de treball del personal que participa al projecte. En aquest cas concret, els servidors físics es fan servir per molts altres serveis més, en total hi han 34 màquines virtuals funcionant. Per aquest motiu la participació dels servidors al projecte serà de 1/34 part. En el cas de les estacions de treball per els treballadors només les faran servir com màxim 2 a la vegada, per això s'ha pressupostat només 2 equips que compartiran ja que no està planificat que treballin més de dos persones a la vegada. Un serà exclusiu per al programador i l'altre el faran servir la resta de treballadors del projecte quan estigui programat per fer-ho servir.

Els costos de taules, cadires, espai de treball i etcètera ja existeix a les instal·lacions i està molt amortitzat, per aquest motiu s'obvia el seu cost.

Materials	Servidors: 2 x Servidor Dell PowerEdge R640 Compartit amb 34 serveis més	GP, PR, ER o IG. Màxim 2 a l'hora Dell Optiplex 5490 AIO
Hardware / Maquinari	11.000,00 €	1.110,00 €
Sistema Operatiu	4.400,00 € (VMWare ESX)	0,00 € (W10 inclòs)
Participació	2,94% 1/34 unitats	200,00% 2 unitats
Cost Maquinari Utilitzat	452,94 €	2.220,00 €
IDE		0,00 €
Framework Front-end	0,00 €	
API Back-end	0,00 €	
Totals per llocs	452,94 €	2.220,00 €
Costos de material:	2.672,94 €	

Taula 7: Costos de material per lloc de treball i total

Font: Dades de servidor cedides pel departament de Sistemes de UTG CNTIC^[3] a la FIB-UPC

2.3.1.3 Amortitzacions

Pel que fa als servidors, són dos equips adquirits a l'any 2021 i posats en producció al Juny d'aquell mateix any. No es va estimar la vida útil de l'equipament però podem fer un paral·lelisme amb el cost d'un servei similar. Contem que ja des d'un inici tenia en marxa les **34 màquines virtuals (VM)** funcionant. Des del Juny fins la utilització d'aquest projecte hauran passat **9 mesos**. Si estimem el cost d'una màquina virtual d'un servei com **Amazon Web Services AWS^[40] té un cost aproximat de 620€ al mes** tenim que:

$$34 \text{ VM} * 620 \text{ €/mes} * 9 \text{ mesos} = 189.720 \text{ € d'amortització}$$

Si contem que l'equipament suma 11.000 € de hardware més 4.400 € de sistema operatiu tenim que:

$$189.720 \text{ €} / (11.000 \text{ €} + 4.400 \text{ €}) = 12,3 \text{ vegades amortitzat}$$

Assumint que les instal·lacions també fa temps que son amortitzades podem afirmar que el cost efectiu d'amortització del hardware de servidors i els seus sistemes operatius estan molt amortitzats de fa mesos. Així doncs, es podria considerar que el cost del servidor seria pràcticament zero.

En quant a l'equipament d'estacions de treball, si considerem que no s'utilitzarà equipament ja present a l'empresa podem dir que:

Tenint en compte que la vida útil d'una **estació de treball** solen ser **4 anys**. Es necessiten un mínim de **2 equips**, tenint present que no es tractaran les hores en funcionament de cadascun si no la seva disponibilitat. Un dels equips ha de estar disponible **3 mesos i l'altre només 2**.

$$4 \text{ anys} * 12 \text{ mesos} = 48 \text{ mesos d'amortització}$$

$$1.100\text{€ a amortitzar per equip} / 48 \text{ mesos} = 22.92 \text{ € d'amortització per equip al mes}$$

$$3 \text{ mesos d'amortització} * 22.92 \text{ € d'amortització mensual} = 68,76 \text{ d'amortització en un dels equips}$$

$$2 \text{ mesos d'amortització} * 22.92 \text{ € d'amortització mensual} = 45,84 \text{ d'amortització per l'altre equip}$$

El percentatge d'amortització total sobre el temps de durada del projecte serà:

$$(68,76 \text{ €} + 45,84 \text{ €}) / (2 \text{ equips} * 1.100 \text{ € per equip}) = 0,052$$

Només s'amortitzarà el 5,2% de l'equipament d'estacions de treball per aquest projecte. Seria sensat fer servir aquest equipament per altres projectes un cop acabat aquest o, com ja s'ha esmentat a altres punts, fer servir equips que ja estiguin a l'empresa.

2.3.1.4 Contingències i imprevistos

Es preveu que poden haver contingències degudes a negociacions o canvis de preu que es poden assumir fins a un 10% del valor pressupostat.

En quant als imprevistos que podem trobar podem afirmar que només poden ser en hores dels treballadors del projecte ja que el material te manteniment i garanties de funcionament inclosos. Així doncs, per imprevistos de complicacions al temps de programació, falles de maquinari físic o de software podem assumir que els temps de treball poden augmentar un 20%. Per tant el cost dels recursos humans pot pujar per sobre dels dotze mil Euros.

Tipus de cost	Original	10%	Totals per costos
Costos Recursos Humans	9.121,40 €	912,14 €	10.033,54 €
Costos Materials	2.672,94 €	267,29 €	2.940,24 €
Totals:	11.794,34 €	1.179,43 €	12.973,78 €

Taula 8: Costos amb contingències

Font: Elaboració pròpia

Tipus de cost	Amb contingències	20%	Totals per costos
Costos Recursos Humans	10.033,54 €	2.006,71 €	12.040,25 €
Costos Materials	2.940,24 €	-	2.940,24 €
Totals:	12.973,78 €	2.006,71 €	14.980,48 €

Taula 9: Costos amb imprevistos

Font: Elaboració pròpia

2.3.1.5 Pressupost Final

Aquí faltaria també la desviació, quantes hores Jaume, quantes hores meves, s'han ajustat correctament?

Els costos finals desglossats per conceptes són els següents:

Tipus de cost	Costos totals
Costos Recursos Humans	9.121,40 €
Costos Materials	2.672,94 €
Contingències	1.179,43 €
Imprevistos	2.006,71 €
Total:	14.980,48 €

Taula 10: Costos totals

Font: Elaboració pròpia

2.3.2 Control de gestió

Una qüestió important és controlar les desviacions del pressupost al tancament de cada tasca per revertir al pressupost les hores o imprevistos trobats. Farem servir una serie de fórmules detallades a continuació que calcularan aquestes desviacions de costos reals amb els costos estimats inicialment.

- Desviació d'hores consumides per tasca
(Hores estimades - Hores reals) * Cost estimat
- Desviació dels costos segons les hores consumides per tasca
(Hores estimades - Hores reals) * Cost real
- Desviació dels costos en recursos humans per tasca
(Costos estimats - Costos reals) * Hores reals
- Desviació total dels costos materials
Costos de materials estimats - Costos de materials reals
- Desviació total dels costos de contingències
Costos de contingències estimats - Costos contingències reals
- Desviació total dels imprevistos
Costos imprevistos estimats - Costos imprevistos reals
- Desviació total dels costos de personal
Cost personal estimat - Cost personal real
- Desviació total d'hores
Hores estimades - Hores reals
- Desviació total del costs
Cost total estimat - Costos total real

2.4. Sostenibilitat

Resulta difícil avaluar l'impacte de sostenibilitat respecte al consum energètic a un equip que comparteix consums amb altres 33 serveis més amb diferents nivells requisits computacionals. Tot i així es considera que executar l'aplicació en un entorn de maquinari compartit comporta una millora en sostenibilitat.

2.4.2 Dimensió econòmica

El cost del projecte es considera prou ajustat si es té en compte que es poden estalviar despeses re-utilitzant equipament. Es considera un projecte necessari demandat pels usuaris, sobre tot per operació, que havien de revisar contínuament diverses webs per verificar canvis en lloc de tenir centralitzada la supervisió del serveis.

Normalment aquest problema es sol solucionar amb un paquet integrat de totes les tasques que es necessiten, una solució molt cara i que difícilment s'ajusta a tots els requeriments dels llocs de treball. Amb la solució aportada d'aquest projecte s'obté una solució més econòmica que integra totes les eines existents amb aquelles funcionalitats que no porten per defecte.

2.4.3 Dimensió ambiental

El fet de fer servir contenidors i servidor compartit implica un estalvi tant per la part de maquinari físic com de sistema operatiu i serveis addicionals com còpies de seguretat del servidor sencer. Al projecte ja es comenta que és probable que es reutilitzin equips ja presents als departaments per fer la feina, cosa que estalviarà la compra dels entorns de treball amb una reducció substancial dels residus ambientals. Comparat amb altres solucions, el fet de executar l'aplicació a un servidor compartit i la reutilització d'equipament implica una millora mediambiental notòria respecte a noves instal·lacions amb equipament nou. El fet que les estacions de treball siguin equips integrats en un sol conjunt (monitor, teclat i ratolí) també és un afegit en reducció de residus.

2.4.4 Dimensió Social

S'estima que l'estalvi en temps i la comoditat del treball del personal que el farà servir seran considerables. Encara que hem de tenir en compte que tot canvi quasi bé sempre obté un rebuig, en aquest cas la metodologia de treball pot continuar sent com era abans però amb la nova eina que aporta un afegit funcional al conjunt d'aplicacions. Respecte a d'altres solucions que canvien tota la metodologia de treball es troba que aquesta és la forma menys traumàtica introduir canvis, on les aplicacions continuen sent-hi però es té una gestió centralitzada d'elles. Aquest projecte ve d'una demanda que es va trobar i que amb la tecnologia actual es pot arribar a resoldre de forma econòmica, ràpida, funcional i ben rebuda pels usuaris.

3. Desenvolupament

3.1. Tecnologies, eines i coneixements aplicats

Els coneixements per desenvolupar el projecte són diversos i abasten tant feines de desenvolupador, disseny d'interfícies, sistemes i seguretat informàtica.

3.1.1 Llenguatges de programació

Avui dia parlar de llenguatges de programació no és tan rellevant com fa un temps on el llenguatge i les particularitats de cada un marcaven les característiques de tot sistema. Ara el concepte de desenvolupament es basa més en Frameworks i llibreries on és més important dominar un Framework de desenvolupament que no el llenguatge de programació que fa servir. Si bé l'àmbit del projecte ja determina certs llenguatges de programació la tendència global marca que per a un projecte de servei web com aquest els llenguatges habituals són pocs, tant per al Front-End com per al Back-end. És possible desenvolupar el projecte en altres llenguatges no habituals per l'entorn però sempre trobarem molts més inconvenients i molt poca informació de cara a emmarcar-ho en aquest àmbit concret.

Els llenguatges de programació són bàsicament dos, JavaScript per al Front-End i Python per al Back-end. Tot i que també s'han utilitzat scripts en Shell per automatitzar processos de sistema operatiu.

Si bé Python s'ha fet servir en assignatures com Robòtica i algunes de l'especialitat de Computació, no es pot dir el mateix de JavaScript con no s'ha trobat cap assignatura cursada que el faci servir. Cal comentar que també han sigut molt útils les assignatures de Sistemes Operatius en quant a Linux es refereix i els scripts fets servir.

3.1.2 Software extern i Frameworks

Un Framework és un concepte molt ampli, es podria denominar com una estructura de fitxers i llibreries que conformen la base d'una aplicació sigui el tipus que sigui. No ha sigut fins ben entrat el mil·lenni que el Frameworks no han esdevingut el mètode de producció de software per internet més habitual. Un Framework proveeix una estructura i llibreries que faciliten el desenvolupament enormement a més de donar una capa de seguretat en quant a la estructura es refereix. Avui dia, crear un servei de xarxa o una pàgina web dinàmica, segura, modularitzable i fàcilment sostenible no té sentit si no es fa amb un Frameworks que estandaritzen i agilitzen el seu desenvolupament.

Tot el projecte està basat principalment en Frameworks, per una banda tenim el Framework Vue.js sota l'entorn Node.js utilitzant JavaScript com llenguatge de programació, i podem considerar també l'entorn visual de la UPC com un petit Framework pel que fa a l'aspecte gràfic. Per altra banda tenim el Framework Django^[27] que utilitza Python com llenguatge de programació. Tenim també la base de dades MariaDB que és una base de dades relacional tipus SQL i la base de sistema operatiu Linux dins de contenidors Docker que conformen l'entorn d'execució del Sistema d'Informació.

Tots els serveis estaran funcionant sota la tecnologia Docker amb les funcionalitats addicionals de Docker-compose.

Més en detall:

- **Docker:** És una tecnologia de encapsulament de software on tenim una micro-màquina virtual que dona un servei. Normalment Docker s'emmarca en la filosofia de micro-serveis on micro-màquines virtuals s'executen per separat comunicar-se entre elles per formar un sistema global. Les avantatges dels contenidors, i Docker en aquest cas, són que es poden moure, clonar i reiniciar molt ràpidament. Son molt lleugers en quan a memòria i ben preparades es poden escalar ràpidament sense perdre servei. Un contenidor té totes les dependències de sistema operatiu incorporades, per aquest motiu és fàcilment portable a altres servidors Docker o al seu anàleg evolucionat Kubernettes^[11]. Funciona a base d'imatges oficials amb un sistema operatiu Linux molt reduït amb bases de llenguatges i Frameworks bàsics. En el cas d'aquest projecte s'han fet servir imatges de Node.js, Python i NGINX.
- **Node.js i Vue.js:** Node es un Framework basat en Javascript que serveix com base per a altres Frameworks de més alt nivell com Vue.js. La seva característica principal és que té un gestor de paquets des de el qual podem instal·lar multitud de llibreries per al nostre projecte Javascript. Per altre banda, Vue.js és un Framework estructural que defineix unes estructures de fitxers, directoris i configuracions pròpies amb la seva pròpia llibreria de compilació. Node.js amb Vue.js te la principal avantatge de que es poden executar directament des de el directori d'arxius amb una simple comanda que aixeca un servidor web local, ideal per a desenvolupar. La estructura de Vue.js es centra principalment en el concepte de "Component" on els components s'enllacen entre ells per crear l'arbre estructural de una web. També ens permet afegir Plug-ins i llibreries per enriquir les funcionalitats del nostre projecte Vue.js.
- **Django**^[27]: És un Framework basat en Python que dota d'una estructura de servei REST API al projecte. S'instal·la directament des dels repositoris de Python i també conté una gran varietat de Plug-ins per enriquir el seu funcionament. El concepte principal de Django son els mòduls i les "apps", dons dintre d'una estructura Django podem tenir diferents aplicacions independents diferenciades per la adreça URL d'accés des de el client. Internament conté una estructura de directoris molt coherents i si es segueix la estructura la càrrega de mòduls és automàtica. Django també té l'avantatge de que pot connectar a una base de dades i transformar les crides a objectes (models a títol intern) a crides SQL estalviant-nos construir una crida directa a la base de dades. Tant les taules de la base de dades com les vistes de les dades s'estableixen directament sobre Django i és el propi Framework el que crea les taules a la base de dades i les rutes d'accés de la API per les visualitzacions.
- **NGINX:** És un dels servidors Webs més simples i versàtils que pode trobar. Ens permet tant publicar una web senzilla com redirigir tràfic entre diferents serveis. Per aquest projecte l'hem fet servir com servidor Web per al Front-End i com proxy SSL al Back-End.
- **MariaDB:** La pròpia universitat ens ha proporcionat accés a un servidor de bases de dades per poder encabir la base de dades d'aquest projecte. Aquest servidor és MariaDB, un servidor SQL de codi obert molt popular i fiable. Pot albergar dades relacionals, crear vistes, definir restriccions i la resta de propietats de qualsevol altre servidor SQL.
- **UPC Webapp ui Vue 3:** És un Framework d'interfície sobre Vue 3. És un dels nivells més alts en l'escala de programació. Dota d'un entorn preparat amb un menú lateral, una barra de títol, una secció d'usuari, un apartat de selecció d'idioma funcional i un peu informatiu. A més

incorpora el Plug-in Vuex que permet guardar dades per poder fer-les servir en qualsevol punt de l'estructura web. Conté els logotips i colors corporatius de la UPC i la seva documentació conté suggerències dels Plug-ins i contingut visual recomanat.

3.1.3 IDE's, editors i eines

Les eines de desenvolupament com els IDE's moltes vegades s'injurien perquè acaben consumint molts recursos de l'equip del desenvolupador però en els últims temps això ha deixat de ser així i han esdevingut una eina molt útil si s'ha de fer servir eines diverses i habituals.

Per aquest projecte s'ha fet servir l'IDE Microsoft Studio Code amb Plug-ins per a GIT, Docker i Requests on des de la mateixa interfície es poden actualitzar versions al repositori^[41], manegar els contenidors Docker locals i fer proves de crides a les APIs amb les que es treballa. Per a provar les versions en desenvolupament ha sigut necessari tenir instal·lat Docker a l'equip de desenvolupament, sobre tot al començament quan s'han d'establir els paràmetres dels contenidors. De forma puntual també s'han fet servir altres editors com Nano, Notepad++ o Kate segons les circumstàncies. També caldria anomenar el repositori GITLab de la UPC^[41] des de on es carregaven i descarregaven les versions als diferents equips de desenvolupament i producció.

3.2. Anàlisi de requisits i especificació final

A l'apartat de planificació del projecte havíem projectat una serie de requisits i especificacions previs que s'havien de mira d'aconseguir. Però com que tot canvia aquells requisits s'han de tornar a revisar i veure què es pot implementar i què no segons el procés del projecte.

3.2.1 Requisits funcionals

Els requisits funcionals són aquells que determinen les funcions que poden fer els usuaris al sistema. A l'apartat de planificació (2.1.3.1) es projectaven una serie de requisits força generals que, una vegada està el projecte en marxa, han variat amb el temps en funció de les circumstàncies que s'han trobat en el seu desenvolupament.

Els requisits funcionals en ple desenvolupament trobem que són:

- **Iniciar sessió:** Un usuari ha de poder iniciar sessió amb el seu usuari de la UPC i poder accedir a l'aplicació.
- **Veure les dades del perfil de l'usuari:** Cada usuari ha de poder veure la informació del seu perfil en la mesura en que els protocols de seguretat de la UPC ho permetin.
- **Visualitzar notificacions:** Un usuari ha de poder veure les notificacions d'incidències de cada mòdul per estar informat de forma immediata dels esdeveniments rellevants del lloc de treball.
- **Cada unitat informativa (dada) ha de poder ser enllaçada al seu origen:** Per cada dada visualitzada al sistema l'usuari ha de poder accedir al sistema d'on prové aquesta informació si el disseny així l'especifica.
- **Panell (Dashboard) de resum:** Tot usuari ha de poder accedir al Panell principal de l'aplicació que resumeix en una sola pàgina la informació dels diferents mòduls del sistema.
- **Ordenació i cerca de dades:** Tota informació rebuda ha de poder ser filtrada i ordenada per cada fragment informatiu sempre que sigui possible.

3.2.2 Diagrama de casos d'ús

El diagrama de casos d'ús mostra les accions que pot realitzar un usuari dins del sistema i les relacions entre elles. Per accedir al sistema la porta d'entrada és el Panell tot i que es pot accedir a les diferents seccions per la URL. Totes les accions requereixen autenticació de la UPC per poder accedir i algunes accions depenen d'altres tal i com indica la etiqueta "includes".

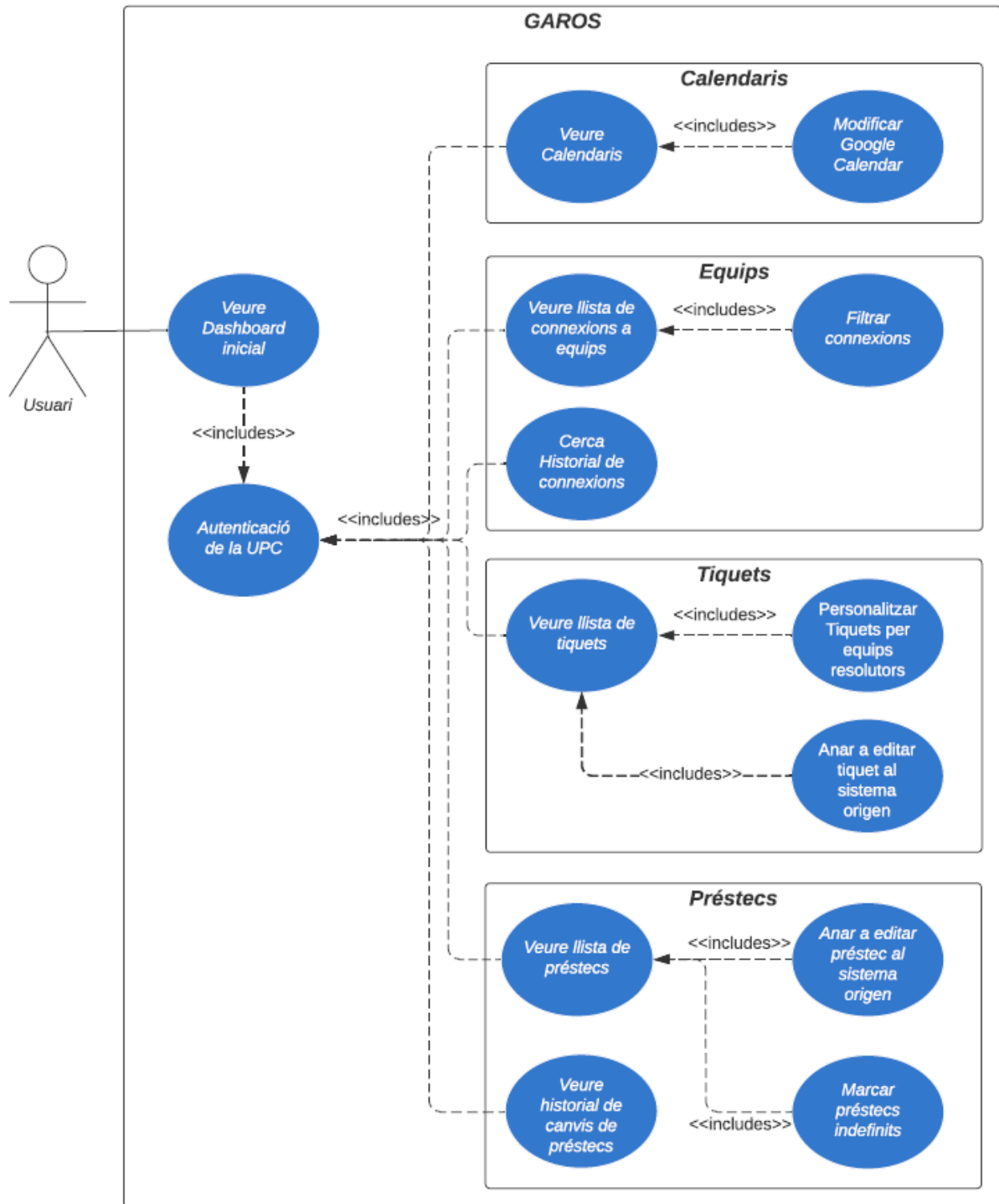


Figura 5: Diagrama de casos d'ús
 Font: Elaboració pròpia

El diagrama mostra els casos d'ús següents:

- **Veure el Dashboard inicial:** Mostra els resums dels mòduls del sistema amb petits llistats i gràfiques. Aquest cas d'ús inclou el cas de "Autenticació de la UPC" prèviament.
- **Autenticació de la UPC:** Mostra una pantalla d'autenticació SSO Lab on hem d'introduir el nostre nom d'usuari i contrasenya, i si és la primera vegada ens demanarà permís de confiança amb l'aplicació GAROS.
- **Veure Calendaris:** Mostra els diferents calendaris en diferents pestanyes i la pestanya final té una visió mensual de tots els calendaris. Aquest cas d'ús inclou el cas previ de "Autenticació de la UPC".
- **Modificar Google Calendar:** A sota de cada calendari tenim un botó que ens obre una pestanya nova al navegador amb Google Calendar. Aquest cas d'ús inclou el cas previ de "Veure Calendaris".
- **Veure llista de connexions a equips:** Ens permet veure les connexions actives als equips dels aularis. Aquest cas d'ús inclou el cas previ de "Autenticació de la UPC".
- **Filtrar connexions:** Ens permet filtrar les connexions actives mitjançant el cercador o anant a les pestanyes de pre-filtratge per edificis. Aquest cas d'ús inclou el cas previ de "Veure llista de connexions a equips".
- **Cerca Historial de connexions:** Ens permet fer cerques de totes les connexions registrades encara que no estiguin actives. Aquest cas d'ús inclou el cas previ de "Autenticació de la UPC".
- **Veure llista de tiquets:** Podem veure un llistat de tiquets en actiu. Aquest cas d'ús inclou el cas previ de "Autenticació de la UPC".
- **Personalitzar tiquets per equip resolutor:** Podem personalitzar els tiquets que veiem activant o desactivant els equips resolutors que necessitem. Aquest cas d'ús inclou el cas previ de "Veure llista de tiquets".
- **Anar a editar tiquet al sistema origen:** Per poder editar o interactuar amb un tiquet podem prémer el botó d'Obrir que conté la fila informativa del tiquet. Aquest cas d'ús inclou el cas previ de "Veure llista de tiquets".
- **Veure llista de préstecs:** Permet veure els préstecs vigents en llistats. Aquest cas d'ús inclou el cas previ de "Autenticació de la UPC".
- **Anar a editar préstec al sistema origen:** Ens permet editar o retornar un préstec al Racó^[4] mitjançant els botons d'acció que conté cada línia d'informació de tiquet. Aquest cas d'ús inclou el cas previ de "Veure llista de préstecs".
- **Marcar préstecs indefinits:** Per poder marcar un préstec com indefinit només s'ha de activar el switch de la columna de indefinit que apareix a cada fila informativa de tiquet. Aquest cas d'ús inclou el cas previ de "Veure llista de préstecs".
- **Veure historial de canvis de préstecs:** Podem veure l'historial de canvis de préstecs per veure quins canvis han esdevingut en els últims préstecs i fer cerques al llistat. Aquest cas d'ús inclou el cas previ de "Autenticació de la UPC".

3.2.3 Requisits no funcionals

El requisits no funcionals son aquells que tenen un caràcter general que no es solen referir a funcions concretes. Dels requisits no funcionals detallats en la planificació del projecte (2.1.3.2) es projectaven una serie de requisits que no han variat gaire en quant al recorregut del desenvolupament del sistema.

- El sistema ha de ser àgil i funcional per a totes les funcions que tingui.
- S'han de seguir els colors corporatius de la UPC.
- La comunicació entre serveis ha de ser directa i no ha d'haver necessitat de que l'usuari realitzi cap acció per a que la comunicació es realitzi, s'ha de mirar que les dades requerides s'actualitzin periòdicament de forma automàtica en la mesura del que sigui possible.
- El sistema no ha de interferir ni provocar cap tipus d'error a aquells serveis d'informació on es connecti. S'han de minimitzar al màxim les possibles modificacions de dades remotes.
- La informació rellevant ha de mostrar-se clarament diferenciada de la resta.
- S'han de complir tots els criteris de seguretat que marca la UPC per aquests tipus de serveis.

3.2.4 Esquema de serveis

Al diagrama de la figura següent es poden veure a la línia inferior les fonts de dades de les que es nodreix el sistema, inclòs el servei de Google Calendar embegut de la secció de calendaris.

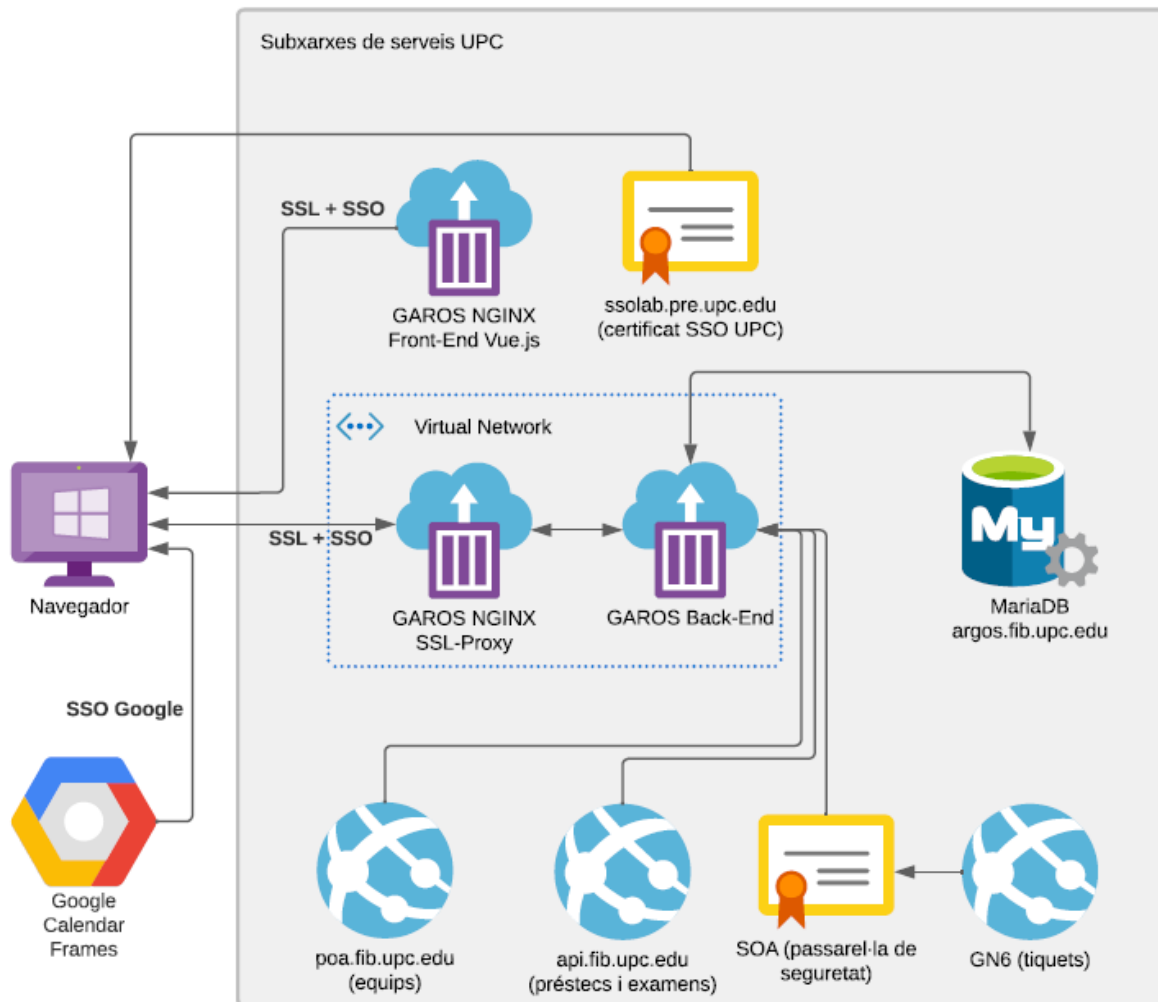


Figura 6: Esquema conceptual de servei
Font: Elaboració pròpia amb Lucidchart

3.3. Disseny de projecte

El disseny és l'apartat clau per a que un servei esdevingui funcional, segur i escalable. És important tenir tota la concepció ben estructurada des d'un inici per poder desenvolupar de forma correcta i directa minimitzant els canvis circumstancials que poden aparèixer.

3.3.1 Arquitectura i components

A aquesta secció entrarem una mica més en detall en les diferents parts que hem vist a la figura del esquema conceptual anterior.

3.3.1.1 Contenedors

Tenim tres contenidors Docker en funcionament pel sistema en producció tot i que per compilar la web Vue.js cal prepara un quart que realitzarà aquesta compilació i copiarà el resultat al servidor NGINX que donarà el servei Web. A la següent figura mostrem el contenidor Node en un color diferent per indicar que després no donarà servei i serà esborrat.

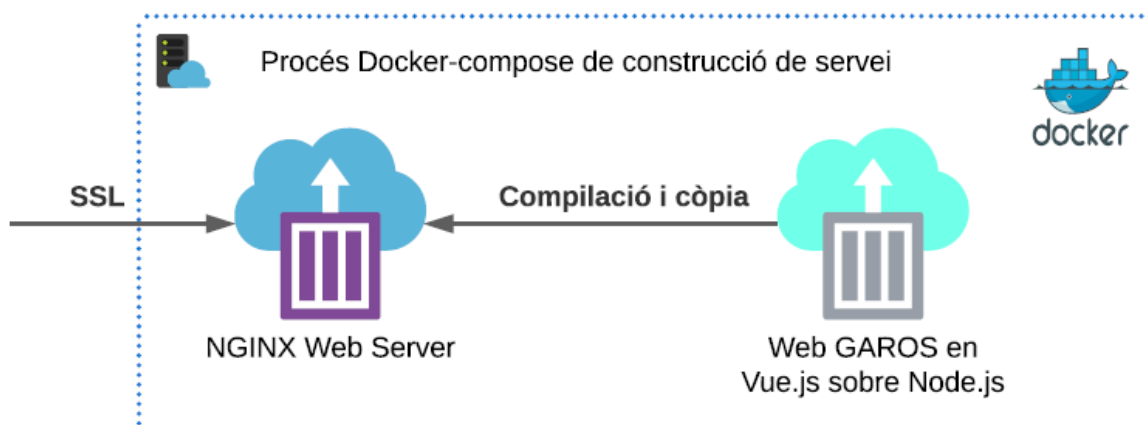


Figura 7: Esquema de contenidors de Front-End
Font: Elaboració pròpia amb Lucidchart

Podem veure un exemple d'això a un dels fitxers docker-compose que utilitzem. Concretament el "docker-compose.pro.yml" del Front-end.

```
version: "3"

services:
  web:
    container_name: garos-front
    build:
      dockerfile: Dockerfile.pro
      context: .
    volumes:
      - /home2/users/lcfib/garos/ssl:/etc/nginx/ssl:ro
      - /home2/users/lcfib/garos/ssl/nginx-default-web.conf:/etc/nginx/conf.d/default.conf:ro
      - /home2/users/lcfib/garos/ssl/auth.conf.js:/public/conf/auth.conf.js:ro
```

```

environment:
- VUE_APP_GAROS_API_SRV=${VUE_APP_GAROS_API_SRV}
- VUE_APP_GAROS_API_PORT=${VUE_APP_GAROS_API_PORT}
- VUE_APP_BACKEND_URL=${VUE_APP_BACKEND_URL}
- VUE_APP_OIDC_CLIENT_ID=${VUE_APP_OIDC_CLIENT_ID}
- VUE_APP_AUTHORITY=${VUE_APP_AUTHORITY}
ports:
- "147.83.41.32:80:80"
- "147.83.41.32:443:443"
dns:
- 147.83.41.36
- 147.83.41.104
dns_search:
- fib.upc.es

```

Taula 11: Contingut del fitxer `docker-compose.pro.yml` del Front-End
Font: Elaboració pròpia amb DBEaver

Al fitxer podem veure com tenim designat el fitxer `Dockerfile.pro` que ens executarà la compilació i copia entre contenidors, a “volumes” podem veure com es mapejen la carpeta `SSL` i els arxius d'autenticació `SSO`. A la secció “environment” podem veure com es designen les variables d'entorn que després faran servir els contenidors a dins. I a “ports”, “dns” i “dns_search” tota la configuració de xarxa.

També podem mostrar el fitxer `Dockerfile.pro` per veure com realitza aquesta operació de compilació i copia.

```

# build environment
FROM node:16-alpine as build-stage
LABEL description="Garos Front-End build"
WORKDIR /app
COPY ./app/. .
COPY .env .
RUN npm install
RUN npm rebuild node-sass
RUN npm run build

# production stage
FROM nginx:stable-alpine as production-stage
LABEL description="Garos Front-End"
COPY --from=build-stage /app/dist /usr/share/nginx/html
EXPOSE 443
CMD ["nginx", "-g", "daemon off;"]

```

Taula 12: Contingut del fitxer `Dockerfile.pro` del Front-End
Font: Elaboració pròpia amb DBEaver

Podem veure com dissenyem la imatge base del sistema (node 16-alpine, li afegim un nom descriptiu per diferenciar-ho al llistat d'imatges i contenidors en execució, realitzem operacions de còpia d'arxius, executem la descàrrega de llibreries, i executem la compilació. A l'apartat del servidor web veiem que partim d'una imatge `NGINX stable-alpine`, dissenyem nom descriptiu, copiem la compilació a dins des de el contenidor anterior, obrim el port d'accés `HTTPS` i executem el servidor web.

El server API del Back-End si que manté els dos contenidors perquè el servidor `NGINX` només fa de pasarella `SSL` dins d'una xarxa virtual entre els dos contenidors. A la figura següent podem apreciar

com la connexió externa és per SSL però la interna és HTTP. Les connexions a les APIs de dades de la UPC també es realitzen amb protocol SSL.

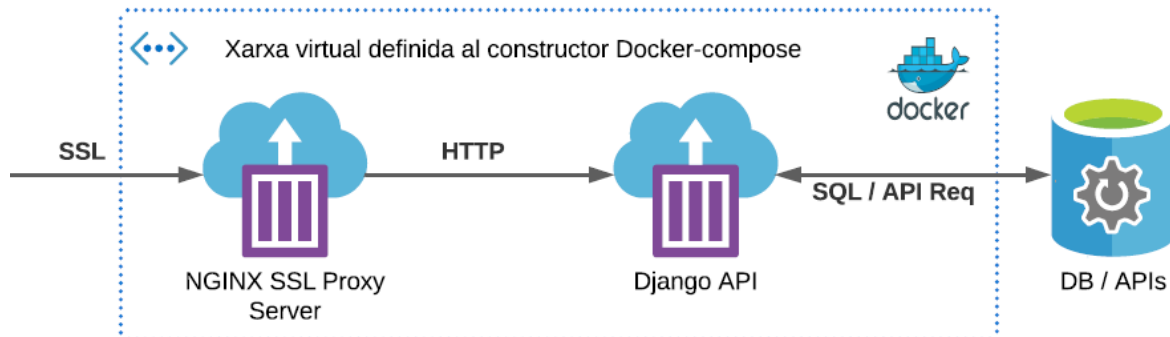


Figura 8: Esquema de contenidors de Back-End
 Font: Elaboració pròpia amb Lucidchart

Podem veure la seva configuració al fitxers “docker-compose.pro.yml”, “Dockerfile.api” i “Dockerfile.proxy”.

El primer de tots designa la configuració interna dels dos contenidors i la xarxa virtual interna entre els dos:

```
version: '3.9'

services:
  api:
    container_name: garos_back_api
    build:
      dockerfile: Dockerfile.api
    context: .
    command: python manage.py runserver 0.0.0.0:8000
    env_file:
      - sql-variables.env
    environment:
      - SQL_SRV=${SQL_SRV}
      - SQL_PASSWORD=${SQL_PASSWORD}
      - OIDC_CLIENT_ID=${OIDC_CLIENT_ID}
      - AUTHORITY=${AUTHORITY}
      - BACKEND_URL=${BACKEND_URL}
      - FRONTEND_URL=${FRONTEND_URL}
      - OIDC_SERVER_TYPE=${OIDC_SERVER_TYPE}
      - SOA_PASS=${SOA_PASS}
      - GN6_PASS=${GN6_PASS}
    dns:
      - 147.83.41.36
      - 147.83.41.104
    dns_search:
      - fib.upc.es
    networks:
      back:
        ipv4_address: 172.88.0.10

  proxy:
    container_name: garos_back_proxy
```

```

image: nginx:stable-alpine
build:
dockerfile: Dockerfile.proxy
context: .
volumes:
-
/home2/users/lcfib/garos/ssl/nginx-default-api-proxy.conf:/etc/nginx/conf.d/default.conf:ro
- /home2/users/lcfib/garos/ssl/server.pem:/etc/nginx/ssl/server.pem:ro
- /home2/users/lcfib/garos/ssl/server.key:/etc/nginx/ssl/server.key:ro
ports:
- "147.83.41.32:8443:443"
depends_on:
- api
dns:
- 147.83.41.36
- 147.83.41.104
dns_search:
- fib.upc.es
networks:
back:
ipv4_address: 172.88.0.2

networks:
back:
driver: bridge
ipam:
driver: default
config:
- subnet: 172.88.0.0/16
gateway: 172.88.0.1

```

Taula 13: *Contingut del fitxer docker-compose.pro.yml del Back-End*
Font: *Elaboració pròpia amb DBEaver*

Com es pot apreciar, la configuració és molt semblant a la que hem vist abans però ara establim dos serveis i una xarxa que els connectarà.

<pre> FROM python:3 as api-stage LABEL description="Garos Back-End" ENV PYTHONDONTWRITEBYTECODE=1 ENV PYTHONUNBUFFERED=1 WORKDIR /garos_api COPY requirements.txt . RUN pip install -r requirements.txt COPY . . WORKDIR /garos_api/garosBackEnd </pre>	<pre> FROM nginx:stable-alpine as proxy-stage </pre>
---	--

Taula 14: *Contingut dels fitxers Dockerfile.api a la esquerra i Dockerfile.proxy a la dreta*
Font: *Elaboració pròpia amb DBEaver*

Aquí podem veure que els fitxer Dockerfile són força més senzills ja que no han de realitzar cap operació entre ells ni activitats complexes.

En l'apartat de desenvolupament no calen tots els contenidors ja que Vue.js i Node.js ens permet executar un servidor web local, així que en l'equip de desenvolupament només cal carregar el contenidors del Back-end mentre que el Front-End s'executa directament des de la línia de comandes.

Al propi repositori de GitLab del projecte^[41] també hem proporcionat les instruccions per poder executar l'aplicació tant en local com en producció. És important tenir en compte que hi ha informació que no hi és al repositori com contrasenyes, adreces de connexió, etc. Aquestes dades addicionals s'han de preparar a part en fitxers d'entorn per a que es carreguin directament quan executem els fitxers Docker-compose. Concretament en fitxers “.env” just al mateix directori que el fitxer “docker-compose.????.yml”. Això és bàsicament per seguretat, ja que no es poden posar les contrasenyes o segons quines adreces dins al repositori públic.

3.3.1.2 Components i mòduls

En quant a la estructura interna tenim els components dividits en mòduls i aquests s'estructuren de forma independent dins de cada servei.

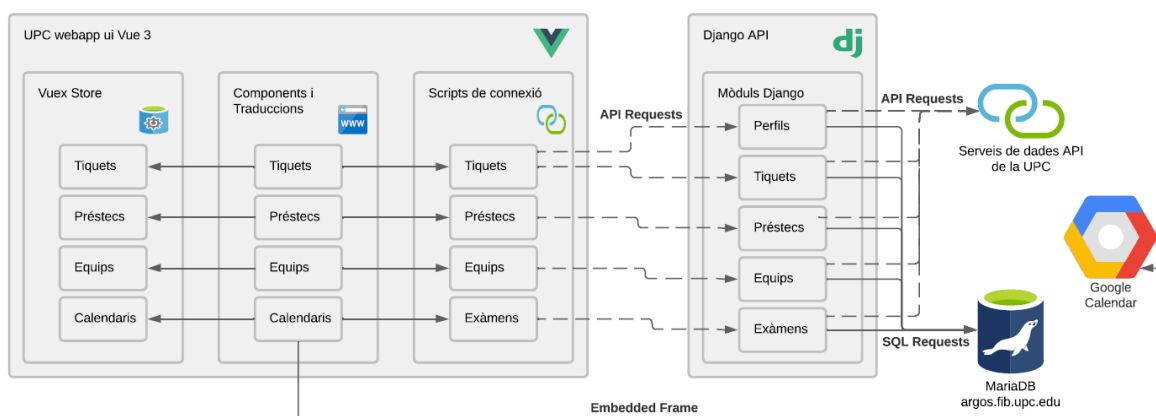


Figura 9: Esquema de components i connexions entre Front-End i Back-End
 Font: Elaboració pròpia amb Lucidchart

És important remarcar que els scripts de connexió actualitzen les dades periòdicament així com les crides API del Back-End a les fonts de dades de la UPC. El fet de que la web s'actualitzi automàticament és gràcies al caràcter reactiu del Framework Vue.js que permet actualitzar dades internes i mostrar-les automàticament. Per a la connexió dels frames embeguts del calendari de Google cal remarcar que es necessari estar autenticat a Google amb l'usuari de la UPC per tenir accés als calendaris dels departaments, i sempre i quan es tinguin permisos.

Abans de detallar cada un dels mòduls explicarem en què consisteix un mòdul. Cada mòdul està dividit en dos parts, la part de Front-End i la part de Back-end. Al Front-End tenim que un mòdul té diverses parts. Una és la d'emmagatzemament global Vuex on es pot recurrir a ella amb funcions de Setters i Getters pròpies des de qualsevol part de la web. Una altra són les de recull de dades del

Front-End i guarda aquesta informació a la seva part de emmagatzemament Vuex. I per últim al Front-End tenim l'apartat Web on Vue es denomina "Component" i que conté les estructures de visualització de les dades recollides i emmagatzemades en memòria. Recordem que Vue s'executa plenament al navegador del client, totes aquestes funcions queden "compilades" en JavaScript al servidor Web i el navegador és qui les executa.

La part del Back-end és molt similar. Existeix una aplicació principal Django^[27] que conté la configuració i administrador del servidor Back-End i tenim una sub-aplicació API que és la que conté el nostre aplicatiu. A la sub-aplicació API hi han diversos directoris anomenats Mòduls on cada un té un arxiu "models" que conté l'esquema de base de dades d'aquell mòdul, un arxiu "views" que conté les vistes de la informació per a les crides GET, PUT, etc. I arxius addicionals que fan les funcions de actualitzadors d'informació, què són les funcions que connecten amb els proveïdors d'informació de la UPC. És important saber que des de Django^[27] no fem crides directes SQL a la base de dades sinó que és el propi Django^[27] qui interpreta les crides i filtres a objectes propis definits als models convertir-les a crides SQL de forma automàtica. Així que podem treballar en llenguatge Python en tot moment sense haver de construir crides SQL quan demanem dades.

En quant als mòduls del nostre sistema podem dividir l'aplicació en cinc parts on la primera es un resum de tots els mòduls.

En primer lloc i com a entrada directa tenim el Panell, que és el que forma el resum dels mòduls de l'aplicació. Dins del Framework de Vue està designat com "home_page" per ser la primera plana que es veu.

Els mòduls pròpiament dits són:

Tiquets: Conté una caché dels tiquets en actiu al qual tenim accés des de Operació i Sistemes. La dada més rellevant d'un tiquet al sistema és el grup resolutor, que és el punt des de on es filtren els tiquets en funció del departament o persona que han de tractar el tiquet. Hi ha un grup resolutor crític que és el de "Dispatcher" que són tots aquells tiquets que no tenen grup resolutor i s'han d'assignar manualment a un en funció del seu contingut. Aquests es ressalten de forma especial per a que es tractin el més ràpid possible. Aquestes dades es recullen des del el servidor GN6^[5] mitjançant la passarel·la de seguretat SOAP.

Préstecs: Conté la caché dels préstecs en vigor i els categoritza entre: Indefinits, que són els préstecs sense data de retorn; Pendants, que són els que estan pendants de retornar en temps correcte; Caducats, que són els que ha passat la data de retorn i s'han de reclamar. A més, el mòdul conté un historial de canvis on queda registrat tot canvi als préstecs com creació, retorn, modificació d'algun camp, o canvi de caràcter indefinit. És important saber que tant la marca d'indefinit com l'historial de canvis són registres locals de GAROS, no són dades provinents del Racó^[4]. El mòdul també informarà en forma d'avís dels préstecs caducats per considerar-los urgents. Aquestes dades s'obtenen del server api.fib.upc.edu que requereix un identificador "client_id".

Equips: Conté les connexions actuals dividides per localitzacions de edificis A5, B5 i C6. A més es poden veure totes juntes com entrada al mòdul. També conté un historial de connexions que registra tant les connexions actives com les que han tancat. Aquesta última informació es sol·licita sota demanda marcant un rang temporal per a les connexions. El mòdul també informarà de les

connexions que porten actives més de 4 hores per a que els usuaris puguin contemplar si un usuari s'ha deixat un equip connectat a un aula. Aquestes dades es recullen del servei POA que no requereix autenticació dins de la xarxa on està tot i que ha sigut necessari fer-hi una modificació per donar-nos les dades necessàries per al nostre projecte.

Calendaris: Conté els sis calendaris habituals de Operació i Sistemes: A5, B5 i C6 que conté les reserves i activitats per aquells aularis; Presentacions i esdeveniments, Recursos Humans i Esdeveniments de Sistemes que contenen tasques planificades, dies d'absències de personal, reunions, etc. El mòdul també informa de les reserves que resten en el dia present. També podem prendre una visió mensual de tots els calendaris. Els Calendaris són frames embeguts de Google Calendar tret dels avisos de reserves que provenen el servei `api.fib.upc.edu` que requereix identificació d'usuari amb un "client_id".

3.3.1.3 Base de dades

En quant a la base de dades podem dir de primeres que no te molta complexitat, ja que es tracta principalment d'un servei de "cache" i tractament temporal de les dades originals. Només podem considerar dades pròpies els perfils d'usuari, l'historial de canvis dels préstecs i l'historial de connexions als equips. És necessari guardar una caché de les dades per poder analitzar els canvis que es produeixen entre actualitzacions.

Seguidament adjuntem els esquemes de les bases de dades extretes del software DBeaver^[42].

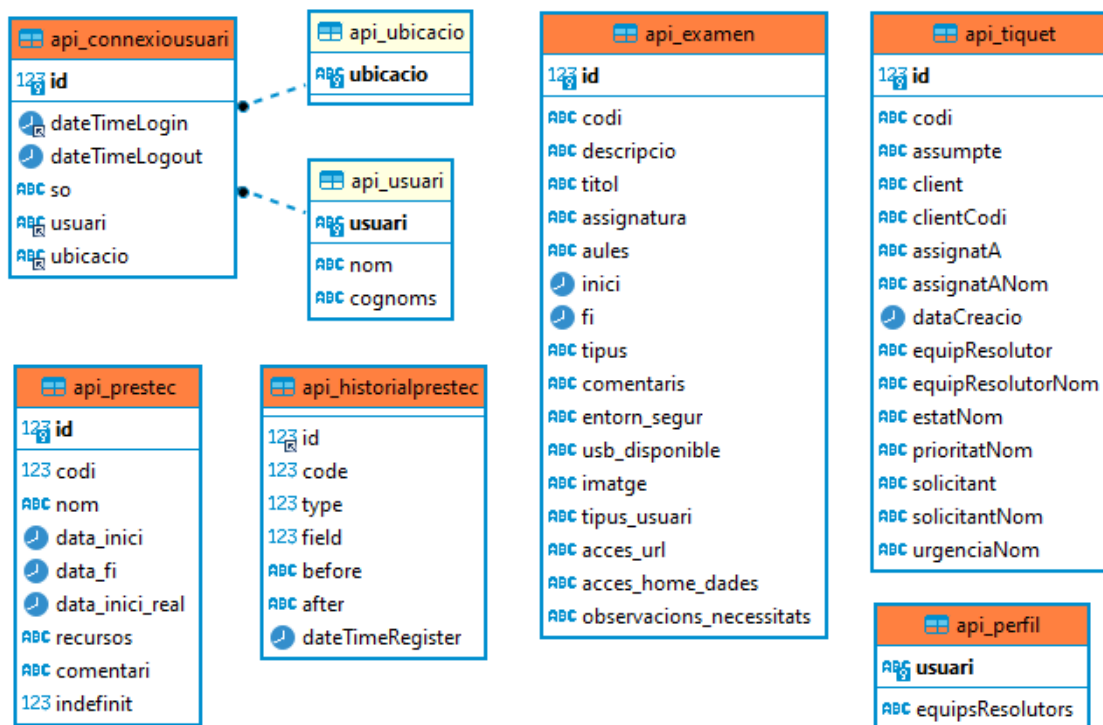
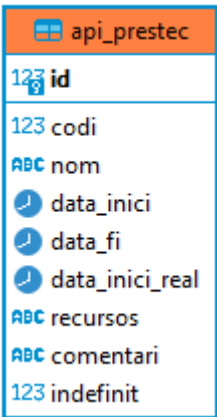
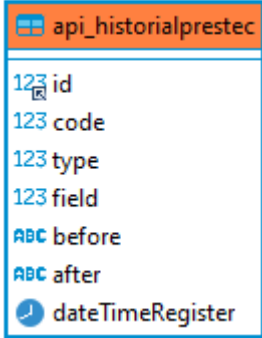


Figura 10: Esquema de la base de dades
Font: Elaboració pròpia amb DBeaver

Cada mòdul del sistema té la seva taula caché, i alguns com el mòdul de préstecs o els equips tenen apartats permanents de dades. Passem a detallar cada taula en cada un dels mòduls:

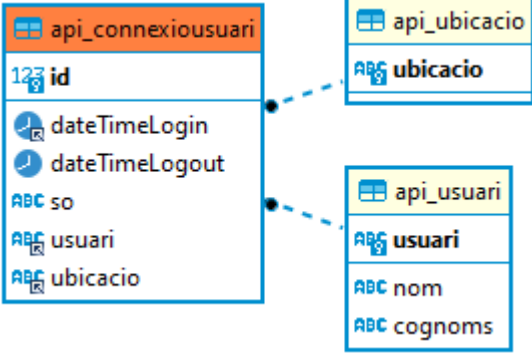
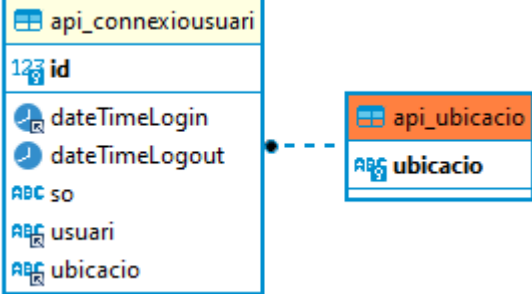
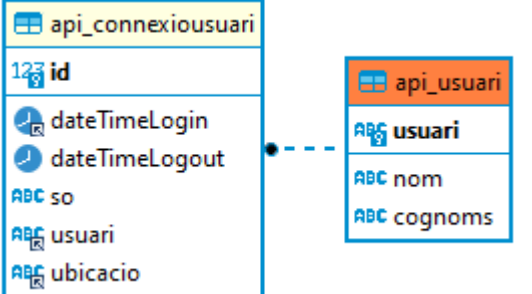
Mòdul de Préstecs

	<p>api_prestec: Copia tal qual les dades dels préstecs provinents del racó^[4] i afegeix un valor Booleà de indefinit per marcar si aquest préstec no s’ha de reclamar. Aquesta dada addicional només es troba en aquest sistema i facilita la catalogació dels préstecs en l’ús diari.</p> <p>Queda registrat el Codi del préstec donat al Racó, el nom de la persona que ha sol·licitat el préstec, la data inicial del préstec, la data final, la data de registre, un llistat de recursos prestats, els comentaris afegits i el caràcter indefinit que el nostre sistema afegeix.</p> <p>Per cada entrada es genera un id únic de préstec independent del codi provinent del Racó.</p> <p>Cada préstec nou que arriba crea una entrada nova i cada préstec que desapareix del llistat d’entrada elimina els préstecs desapareguts, recordem que és un registre caché.</p>
	<p>api_historialprestec: Cada modificació de préstecs queda registrada a aquesta taula. Es calcula revisant les dades entre una actualització anterior i la present i si hi han préstecs nous, préstecs eliminats o préstecs modificats queden registrats per poder fer servir un historial de què ha passat amb cada préstec. També queda registrat si canvia el estatus de indefinit del préstec.</p> <p>Queden registrats el codi (code), el tipus de modificació (type), el camp modificat (field), l’estat anterior del camp (before) i el nou estat del camp (after), a més es guarda el moment actual on es detecta la modificació.</p> <p>Cada modificació queda identificada amb un id únic.</p> <p>Cada entrada que es genera al historial és permanent a la base de dades.</p>

Taula 15: Taules de la base de dades per al mòdul de Préstecs

Font: Elaboració pròpia amb DBeaver

Mòdul d'Equips

 <p>The diagram shows three database tables. The first table, api_connexiousuari, has fields: id (primary key), dateTimeLogin, dateTimeLogout, so, usuari, and ubicacio. The second table, api_ubicacio, has a primary key ubicacio. The third table, api_usuari, has primary keys usuari, nom, and cognoms. Dashed lines with dots at the end indicate foreign key relationships from api_connexiousuari to api_ubicacio and api_usuari.</p>	<p>api_connexiousuari: Copia tal qual les dades provinents de POA amb claus foranes per a la ubicació i l'usuari. Cal remarcar que totes aquestes dades venen juntes des de l'aplicació POA i al Back-End es distribueixen en les tres classes.</p> <p>Queden registrats els moments de Login i Logout, el sistema operatiu (so) fet servir, l'usuari que ha accedit i la seva ubicació. Aquestes dos últimes dades són claus foranes. Cada terna dateTimeLogin, usuari i ubicació és única.</p> <p>Cada connexió nova rebuda crea una entrada nova però cada connexió que desapareix estableix la variable dateTimeLogout al temps actual i ho guarda a la base de dades.</p>
 <p>The diagram shows two database tables. The first table, api_connexiousuari, has fields: id (primary key), dateTimeLogin, dateTimeLogout, so, usuari, and ubicacio. The second table, api_ubicacio, has a primary key ubicacio. A dashed line with a dot at the end indicates a foreign key relationship from api_connexiousuari to api_ubicacio.</p>	<p>api_ubicacio: Cada ubicació és única i ve identificada amb la seva denominació. És possible que en un futur aquesta classe es pugui arribar a ampliar. Cada ubicació és única i conforma la seva clau primària.</p> <p>Cada ubicació nova que arriba crea una nova entrada i és permanent a la base de dades</p>
 <p>The diagram shows two database tables. The first table, api_connexiousuari, has fields: id (primary key), dateTimeLogin, dateTimeLogout, so, usuari, and ubicacio. The second table, api_usuari, has primary keys usuari, nom, and cognoms. A dashed line with a dot at the end indicates a foreign key relationship from api_connexiousuari to api_usuari.</p>	<p>api_usuari: Cada connexió es produeix per un usuari i POA proveeix del seu nom i cognoms. Per cada usuari es guarda el nom d'usuari del sistema, el seu nom personal i els seus cognoms. Cada nom d'usuari és únic i conforma la pròpia clau primària.</p> <p>Cada nou usuari nou que arriba es crea una nova entrada i és permanent a la base de dades.</p>

Taula 16: Taules de la base de dades per al mòdul d'Equips

Font: Elaboració pròpia amb DBeaver

Mòdul de Tiquets

 <p>api_tiquet</p> <ul style="list-style-type: none"> id codi assumpte client clientCodi assignatA assignatANom dataCreacio equipResolutor equipResolutorNom estatNom prioritatNom sol·licitant sol·licitantNom urgenciaNom 	<p>api_tiquet: Copia gran part de la informació rebuda de cada tiquet, queden fora de la taula el contingut i altres dades no rellevants per la informació que es requereix.</p> <p>Per cada tiquet es guarda el codi, l'assumpte, el nom del client d'on prové el tiquet (aplicació origen), el codi del client (codi d'aplicació origen), a qui ha sigut assignat el tiquet si existeix assignació, el nom de la persona o departament assignat, la data de creació del tiquet, el codi de l'equip que ha de resoldre el tiquet, el nom de l'equip resolutor, el nom descriptiu de l'estat del tiquet (no guarda el seu codi), el nom descriptiu de la seva prioritat (no guarda el seu codi), el nom d'usuari del sol·licitant del tiquet, el nom de l'usuari sol·licitant i el nom descriptiu de la urgència (no el seu codi)</p> <p>Cada Tiquet ve identificat amb una id que és clau primària.</p> <p>Quan un nou tiquet apareix es crea una nova entrada i quan desapareix de la llista rebuda també s'elimina la entrada de la taula. Recordem que només fa de caché d'informació.</p>
 <p>api_perfil</p> <ul style="list-style-type: none"> usuari equipsResolutors 	<p>api_perfil: Cada usuari que entra al sistema té una sèrie d'equips resolutors visualitzables per defecte i altres equips que pot veure de forma opcional. Per habilitar i deshabilitar la visualització de cada equip es guarden en aquest perfil on cada usuari pot personalitzar la seva visualització en funció dels tiquets que necessita veure.</p> <p>Cada perfil guarda un text que representa un objecte en format JSON que llista els equips resolutors visibles.</p> <p>Cada perfil ve identificat amb un usuari que és clau primària.</p>

Taula 17: Taules de la base de dades per al mòdul de Tiquets

Font: Elaboració pròpia amb DBeaver

Mòdul de Calendaris

api_examen	
123	id
ABC	codi
ABC	descripcio
ABC	titol
ABC	assignatura
ABC	aules
🕒	inici
🕒	fi
ABC	tipus
ABC	comentaris
ABC	entorn_segur
ABC	usb_disponible
ABC	imatge
ABC	tipus_usuari
ABC	acces_url
ABC	acces_home_dades
ABC	observacions_necessitats

api_examen: A fi de poder veure les notificacions dels exàmens pendents del dia l'aplicació carrega la informació de les reserves realitzades prèviament filtrades. Cada reserva nova crea una entrada nova i cada reserva desapareguda elimina la entrada, recordem que només fa de caché.

Per cada examen es guarda el codi de reserva, la descripció de la reserva, el títol de la reserva, l'assignatura de l'examen, les aules que farà servir, la hora inicial, la hora final, el tipus de reserva, els comentaris afegits, si requereix entorn segur o no, si l'USB estarà disponible, el sistema operatiu que farà servir (imatge), el tipus d'usuari que entrarà als equips, els accessos permesos a webs (acces_url), si es podrà accedir a home_dades, i observacions afegides.

Cada examen queda identificat amb una id independentment del codi que proporciona i que forma la clau primària.

Es té previst fer servir aquesta informació en un futur, per això es guarda informació ara innecessària.

Taula 18: Taula de la base de dades per al mòdul de Calendari

Font: Elaboració pròpia amb DBeaver

3.3.1.4 Seguretat

En qüestions de seguretat és important dir que ha sigut el procés més complicat ja que quasi bé cada font de dades té el seu mètode d'autenticació o registre d'usuari. Aquí detallem els mètodes d'autenticació i les seves particularitats i justificacions:

- **SSOLab Pre:** Per tal de tenir un usuari autenticat al sistema cal tenir un sistema d'autenticació de la UPC. Aquest servei d'autenticació és una versió provisional que serà el futur mètode SSO de la UPC però encara està en desenvolupament. Com és el mateix cas d'aquesta aplicació i també està connectat al sistema d'autenticació SSO actual de la UPC ja ens serveix com mètode d'autenticació del servei, i en un futur la transició serà més ràpida.
- **Api UPC:** Per a la API de la UPC hi han diversos àmbits d'autenticació però el que nosaltres requirim és el públic. Aquest mètode només requereix un registre del sistema dins del servei API per poder fer crides. El registre en qüestió ens proporciona un Client_ID que hem de donar en cada crida a la API.
- **SOAP - GN6:** El sistema d'autenticació per recollir dades dels tiquets conté una pasarel·la SOA d'autenticació prèvia, per aquest motiu calen dos usuaris i dos contrasenyes per fer la crida. El mètode de sol·licitud de dades no és un Request convencional com es fa entre Front-End i Back-End o entre Back-end i API UPC sino que es requereix una libreria de client SOAP per a Python anomenada SUDS.
- **Google:** De cara a poder accedir als calendaris de Google hi han dos vies, una és registrar l'aplicació a Google Cloud per fer crides directament a la API de Calendar i una altre és embebir directament un frame de Google Calendar a la web. Es va triar la segona opció perquè no es requeria tractament de les dades i simplificava enormement el desenvolupament del mòdul.
- **SSL:** Tan per la connexió entre el navegador de l'usuari i la web de GAROS com entre el navegador i la API de GAROS la seguretat de la comunicació és SSL amb un certificat proporcionat per la UPC. Ambdós tenen el mateix certificat ja que treballen amb el mateix domini garos.fib.upc.edu. També existeix seguretat SSL entre el Back-End i la API de la UPC, aquesta última amb el seu propi certificat.

3.3.2 Patrons i disseny

Si mirem el esquema global podem apreciar que és semblant, però no encertat, al concepte del patró Model-Vista-Controlador però amb interfícies API en la comunicació entre la Vista (Front-End) i el Controlador (Back-end) i comunicació SQL entre el controlador (Back-End) i el Model de base de dades (MariaDB). Aquest mateix patró l'hem pogut estudiar a l'assignatura de PROP on es tracten directament aquestes qüestions.

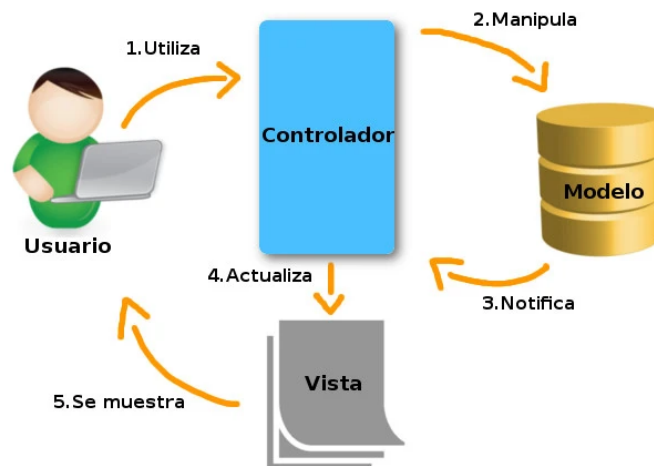


Figura 11: Diagrama Model-Vista-Controlador
Font: Platzi.com

Però en l'esquema de micro-serveis considerat està definit el patró de Agregador amb una capa de visualització i base de dades pròpia com el mencionat anteriorment. El patró Agregador es caracteritza per ser un consumidor de dades API per fer anàlisi, visualització, càlculs, etc. És precisament el que fa el nostre Back-End però amb un Model i Vista afegits.

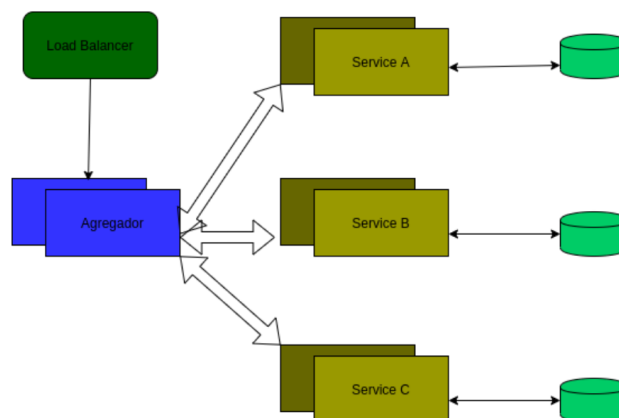


Figura 12: Diagrama Agregador
Font: <https://refactorizando.com/arquitectura-de-microservicios/>

Estrictament parlant i seguint la filosofia de micro-serveis aquest projecte hauria d'haver implementat els dos patrons per separat, un contenidor Python només per recollir dades dels proveïdors informació de la UPC i enviar-les al servidor Django^[27]. I el contenidor amb Django^[27] que faria només les funcions de API per consulta i modificació de dades de la base de dades. Però com que el model Aggregator simple té poques fonts d'informació i encaixen directament amb els mòduls, i aprofitant que la API ja funciona amb el llenguatge que fa servir, hem decidit fer els dos patrons en un de sol.

Potser en un futur cap la possibilitat de diferenciar el dos patrons i separar en dos contenidors aquestes tasques.

3.3.3 Interfície

El disseny d'una bona interfície intuïtiva, agradable i fàcil és clau per l'èxit d'un sistema que fa servir una persona física i per això s'ha posat molta cura de que la distribució, l'accessibilitat, la coherència i la facilitat d'ús estigui dins el medi d'interacció amb l'usuari.

Presentem a continuació una captura de pantalla d'una pàgina inicial per mostrar els camps generals de la interfície. Aquesta interfície ve donada pel Framework de la UPC Webapp Vue 3 que s'ha fet servir de base per la interfície.

A la barra superior podem veure d'esquerra a dreta el logotip de la UPC, seguit de una icona per minimitzar el menú lateral, El títol del sistema "GAROS", una icona informativa que conté un enllaç a la web de <https://www.fib.upc.edu/ca/la-fib/serveis-tic/cataleg-de-serveis>, una icona per escollir el l'idioma del sistema, i un accés al nostre perfil d'usuari autenticat. A la barra lateral esquerra podem veure el menú on trobem les seccions de Panell, Calendaris, Tiquets, Préstecs i Equips. A la seva dreta tindrem el contingut de cada una de les seccions (mòduls), i a la part de sota (peu) tenim una barra que ens indica la propietat "© UPC Universitat Politècnica de Catalunya · BarcelonaTech", un enllaç de contacte que ens informa que podem contactar amb ATIC en cas d'incidències, un enllaç d'accessibilitat que ens porta a la web <https://www.upc.edu/ca/avis-legal/accessibilitat>, i un últim enllaç d'avis legal que ens porta a la web <https://www.upc.edu/ca/avis-legal>.



Figura 13: Distribució de la interfície
Font: Elaboració pròpia

Els continguts de cada mòdul i del panell es detallen a la secció "Experiència d'usuari".

3.4 Experiència d'usuari

És important que cada usuari experimenti el sistema de forma còmoda, informativa, útil i que s'adapti a les seves necessitats. També és important que augmenti el rendiment del treballador facilitant la feina diària i avisar només de la informació rellevant sense perdre tota la informació útil.

3.4.1 Accés i primera vista

De primeres, si introduïm l'adreça de `garos.fib.upc.edu` ens ha de mostrar la pantalla d'autenticació en la qual s'ha d'introduir el nom d'usuari i contrasenya de la UPC. Certament no és igual que la pantalla de Login habitual de la UPC però ja hem mencionat anteriorment que aquest mètode d'autenticació està en desenvolupament i que serà el que acabi donant servei a la UPC.

adAS SSO

Identificació d'usuaris

Autenticació centralitzada

USUARI

CONTRASENYA

ACCEPTAR

Vull recuperar la meua contrasenya

ALTRES MÈTODES D'AUTENTICACIÓ

Certificat digital

Una vegada t'hagis identificat, no caldrà que et tornis a identificar per a accedir a nous recursos.
Per a desconnectar-te, recomanem que tanquis el teu navegador (tancant totes les finestres).

adAS SSO desenvolupat per PRISE

Figura 14: Autenticació SSOLab
Font: Elaboració pròpia

Un cop autenticats entrarem directament a la secció del Panell que ens mostrarà un resum dels diferents mòduls de l'aplicació.



Figura 15: Distribució dels elements del panell

Font: Elaboració pròpia

Al Panell podem veure a la part superior una gràfica de barres que ens indica el nombre d'hores que porta un equip en marxa. Concretament, quantes hores està un usuari logejat. Això és útil de cara a saber si algun usuari s'ha deixat l'equip encès i quantes persones hi han treballant a les aules. Les connexions inferiors a quatre hores con mostrades en verd i les de més de quatre són mostrades en vermell.

A la part inferior tenim el resum de la resta de mòduls. El primer és un llistat resumit dels tiquets on els tiquets amb equip resolutor "Dispatcher" es mostren en vermell. Això és perquè són tiquets que s'han de redirigir a altres departaments i són la feina més urgent.

El segon és un gràfic de barra vertical dels préstecs actuals i les quantitats de cada tipus on el blau són el nombre de préstecs indefinits, els verds són el nombre de préstecs pendents d'entregar a temps, i el vermell són els préstecs caducats pendents d'entregar. Aquest últim grup són els urgents i que s'han de reclamar a la persona que es va registrar el préstec. Per últim veiem un resum de tots els calendaris que fa servir Operació i sistemes on cada calendari està representat en un color diferent.

Al menú lateral esquerra podem veure que les diferents seccions que representen els mòduls tenen indicatius en vermell. Aquests indicatius marquen la informació rellevant que hi ha a cada mòdul. Per exemple: A la gràfica de connexions veiem que hi ha una connexió en vermell i al menú lateral la secció d'equips porta un número 1 en vermell. La secció de préstecs al menú porta un número 8 que indica el nombre de préstecs caducats pendents de reclamar. A la secció de tiquets apareixen el nombre de tiquets de tipus "Dispatcher". I als calendaris indica el nombre de reserves d'exàmens que queden el dia d'avui.

3.4.1 Calendaris

A la secció de Calendaris inicialment podem veure els tres calendaris embeguts dels aularis A5, B5 i C6 dins de la primera pestanya “Exàmens”. A la segona pestanya “Departaments i esdeveniments” trobem els calendaris de “Presentacions i esdeveniments”, “Recursos Humans” i “Esdeveniments Sistemes”.

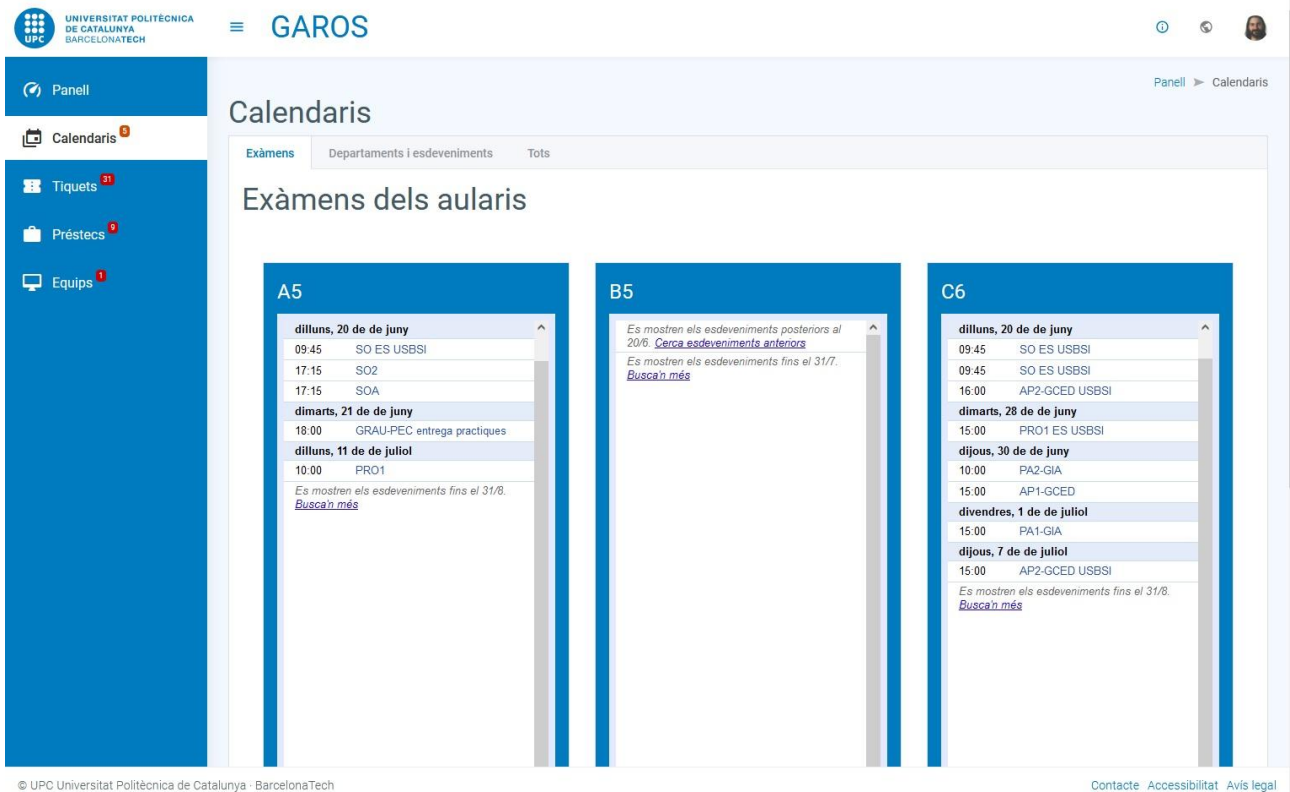


Figura 16: Visual de la secció de Calendaris

Font: Elaboració pròpia

A la última pestanya podem trobar tots els calendaris en una visió mensual per poder tenir una perspectiva temporal de totes les activitats.

Panel

Calendaris 5

Tiquets 31

Prèstecs 9

Equips 1

Calendaris

Exàmens Departaments i esdeveniments Tots

Vista completa de tots els calendaris

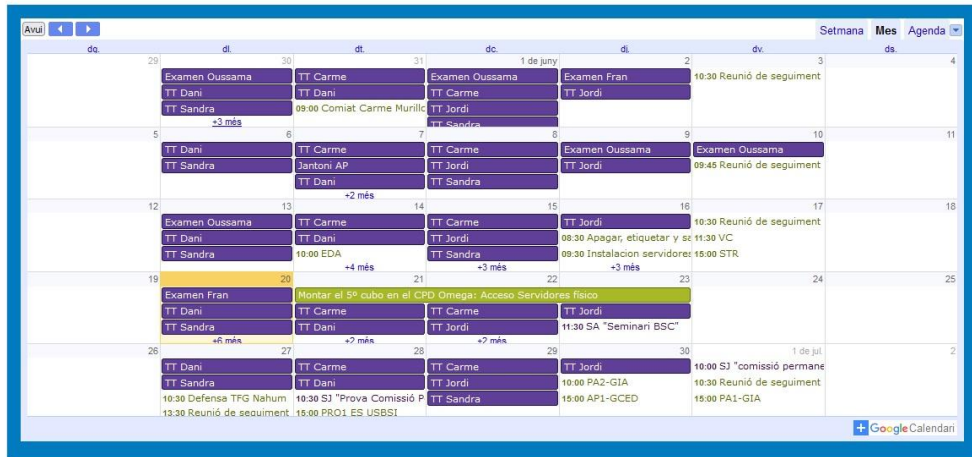


Figura 17: Visual de la sub-secció de tots els calendaris

Font: Elaboració pròpia

Cada calendari té la opció de expandir la informació de cada entrada picant en ella, obrir una nova pestanya directa al calendari de Google picant la icona de sota de cada calendari. A més, podem realitzar recorreguts d'entrades anteriors només fent rodar la roda del ratolí cap endavant si posem el cursor sobre d'un calendari.

Recordem que l'indicador del menú lateral comptabilitza les "Reserves" vigents del dia, i no té perquè coincidir amb les entrades de calendaris dels Aularis.

3.4.1 Tiquets

Només tenim una secció global de tiquets tot i que és dinàmica. Primerament trobem que els tiquets amb equip resolutor “Dispatcher” apareixen en una primera llista emmarcada en vermell ja que són els més urgents a tractar. Els marcats com “Dispatcher” són aquells tiquets on no s’han especificat un grup resolutor en la seva creació, habitualment perquè l’usuari desconeix a quin equip s’ha d’assignar. Si no hi ha cap tiquet de “Dispatcher” tot el requadre vermell i el seu contingut desapareixen de la vista i trobem només la llista que hi ha just a sota, la resta de tiquets.

A la secció de tiquets habituals podem veure que a sobre de la llista hi han dos elements, un botó rodó i un requadre de cerca. El botó ens permet activar o desactivar grups resolutors del llistat, inclosos els “Dispatcher”, amb això podem pre-filtrar els tiquets que l’usuari necessita per al seu lloc de treball. Cal dir que aquesta opció queda guardada al sistema dintre del seu perfil d’usuari. El requadre de cerca podem filtrar els tiquets pels seus camps, el filtre cercarà coincidències en tots els camps visibles del tiquet. Al ser una aplicació “Reactiva” el requadre de cerca s’activa i actualitza automàticament a mida que anem escrivint en ell. La llista també ens permet ordenar-la per quasi tots els camps visibles en ordre ascendent o descendent tret del camp de assumpte que al ser tan heterogeni no té sentit.

Cada entrada de tiquet conté un botó “Obrir” que obre un enllaç cap al sistema GN6^[5] directament al ticket en qüestió.

The screenshot shows the GAROS interface for managing tickets. The left sidebar contains navigation options: Panell, Calendaris (3), Tiquets (1), Préstecs (6), and Equips (1). The main content area is titled 'Tiquets' and features a search bar and a list of tickets. A red box highlights the 'Dispatchers' section, which contains one ticket with the following details:

Codi	Creació	Sol·licitant	Assumpte	Estat	Resolutor	Urgència	Accions
1239906	20/6/2022 11:03:00	Nahum Manuel Martin Vegas	Tiquet de prova no esborrar	Obert	UTIC Docència FIB Dispatchers	Mitja	Obrir

Below this, a search bar is present with the text 'Escriu per cercar'. The main list of tickets includes:

Codi	Creació	Sol·licitant	Assumpte	Estat	Resolutor	Urgència	Accions
1239651	17/6/2022 14:01:00	Jordi Petit Silvestre	Re: [professors-fib] Proves imatges aules informàtiques curs 2022-2023	Pendent	UTIC Docència FIB Suport Estudiants i Professors	Mitja	Obrir
1239569	17/6/2022 12:33:00	Marta Cuatrecasas Capdevila	Canvi roseta PC51362	Obert	UTIC Docència FIB Sistemes, Seguretat i Xarxa	Mitja	Obrir
1239573	17/6/2022 12:17:00	Xavier Martorell Bofill	Sala Actos "MANA-2.0" 01/07/2022 10:00	Obert	UTIC Docència FIB Audiovisuals	Mitja	Obrir
1239336	16/6/2022 19:00:00	Jordi Delgado Pin	Petició -fora de termini- d'instal·lació de programari	Pendent	UTIC Docència FIB Suport Estudiants i Professors	Mitja	Obrir
1239143	16/6/2022 11:54:00	Jordi Petit Silvestre	Re: [professors-fib] Proves imatges aules informàtiques curs 2022-2023	Pendent	UTIC Docència FIB Suport Estudiants i Professors	Mitja	Obrir
1238793	15/6/2022 12:07:00	Marta Cuatrecasas Capdevila	Canvi roseta PC51362	Obert	UTIC Docència FIB Sistemes, Seguretat i Xarxa	Mitja	Obrir
1238447	14/6/2022 14:03:00	Fib Vd Postgrau	canvi de compte fib.vd.postgrau a format vd.postgrau.fib	Obert	UTIC Docència FIB Sistemes, Seguretat i Xarxa	Mitja	Obrir

Figura 18: Visual de la secció de Tiquets
Font: Elaboració pròpia

3.4.1 Préstecs

La pestanya inicial del mòdul de préstecs és la dels préstecs pendents, que són els que es tracten més sovint. També tenim la pestanya de “Caducats” que mostra els préstecs on ha passat la data de tornada. La pestanya de “Indefinits” que són aquells que no tenen data de retorn que no s’han de reclamar. La pestanya de “Tots” que ens mostra tots els préstecs junts.

A sobre de cada llista tenim un botó a la esquerra que ens permet crear un nou tiquet enviant-nos al racó^[4] just a la secció de crear tiquets, i a la dreta un requadre de cerca. El requadre de cerca permet filtrar els tiquets pel contingut de cada tiquet, tots els camps inclosos. Al ser una aplicació “Reactiva” el filtre s’activa i actualitza automàticament a mida que escrivim en ell. El llistat també ens permet ordenar el contingut per columnes en ordre ascendent o descendent en tots els camps tret de els comentaris, la designació d’indefinit i els botons d’accions. No té sentit ordenar per els comentaris ja que són molt heterogenis, pel caràcter indefinit ja tenim una pestanya dedicada i els botons d’acció són tots iguals. Els botons d’acció ens permeten anar a les accions de editar un préstec o marcar el seu retorn, ambdós enviant-nos al racó directament al préstec i la acció que volem fer. La columna de indefinits es pot modificar per marcar o desmarcar el caràcter indefinit de cada préstec. Cal recordar que aquesta dada només es troba a GAROS.

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

GAROS

Panel > Préstecs

Gestió de préstecs

Pendents Caducats Indefinits **Tots** Canvis

Tots els préstecs registrats

Crear nou préstec

Nom i cognoms	Recursos	Data d'inici	Data de fi	Comentaris	Indefinit	Accions
J. IGNACIO HERRANZ SANTILLAN	APALACHE04 MASAI5	18/10/2021 14:34:00	30/12/2022 20:00:00	Master para clonado LENOVO's. NO es APALACHE sino SIOUX15 con pruebas de windows	<input checked="" type="checkbox"/>	Editar Retornar
FRANCESC MARGARIT RUIZ	MASAI9	21/10/2021 8:35:00	17/1/2022 8:00:00		<input type="checkbox"/>	Editar Retornar
TANIA POLONIO ALAMINO	MASAI18	17/12/2021 11:35:00	10/1/2022 11:00:00	BOSSA + CARREGADOR	<input type="checkbox"/>	Editar Retornar
JOAN ANTONI TORRES COBAS	MASAI17	8/4/2022 9:43:00	28/2/2023 21:00:00	Portatili pel geoticon01. No te data de retorn. També te un maletí	<input checked="" type="checkbox"/>	Editar Retornar
HOUDA EL FEZZAK BEKKOURI	MASAI16	1/6/2022 16:53:00	10/6/2022 19:00:00	Funda + Ratoli + Carregador	<input type="checkbox"/>	Editar Retornar

© UPC Universitat Politècnica de Catalunya · BarcelonaTech [Contacte](#) [Accessibilitat](#) [Avis legal](#)

Figura 19: Visual de la secció de Préstecs a la sub-secció de Tots

Font: Elaboració pròpia

Per últim tenim una pestanya addicional on podem veure l’historial de canvis als préstecs. Aquest llistat temporal ens indica quins préstecs han sofert canvis i cada canvi ens els senyala amb una icona

característica. A més, poden realitzar cerques en tot el contingut de cada entrada igual que podem fer amb el llistat de tiquets anterior. Aquest llistat també ens permet anar al Racó a veure el detall del tiquet ja que si un tiquet ha sigut retornat no apareixerà en les seccions anteriors.

The screenshot shows the 'Gestió de préstecs' (Loan Management) interface. The 'Canvis' (Changes) sub-section is active, displaying a list of loan events. The interface includes a sidebar with navigation options: Panell, Calendaris (5), Tiquets (1), Préstecs (2), and Equips (1). The main content area has tabs for Pendants, Caducats, Indefinits, Tots, and Canvis. A search bar is present with the placeholder 'Escriu per cercar'. The list of changes includes:

Icona	Data i Hora	Descripció
🕒	16/6/2022 11:04:37	Creació del préstec amb codi 82021
🔄	14/6/2022 17:26:16	Retornat el préstec amb codi 82001
🕒	14/6/2022 16:04:35	Creació del préstec amb codi 82001
🕒	14/6/2022 10:33:30	Creació del préstec amb codi 81981
🕒	10/6/2022 10:42:56	Creació del préstec amb codi 81961

© UPC Universitat Politècnica de Catalunya - BarcelonaTech Contacte Accessibilitat Avis legal

Figura 20: Visual de la secció de Préstecs a la sub-secció de Canvis
Font: Elaboració pròpia

3.4.1 Equips

A Equips poden veure inicialment totes les connexions a equips actuals a la pestanya de Ara. Les pestanyes de Edifici A5, Edifici B5, Edifici C6 i Altres són contingut pre-filtrat per veure més en concret les connexions d'aquells àmbits. Dintre de cada una d'aquestes pestanyes podem trobar un quadre de cerca que filtra el llistat en funció del contingut de cada camp. Com que és una cerca "Reactiva" el filtre s'activa i actualitza automàticament a mida que anem escrivint en ell. També podem ordenar de forma ascendent o descendent cada una de les columnes que trobem al llistat.

Supervisió d'equips

Connexions actuals a les aules

Nom d'usuari	Ubicació	Hora de login	Durada	Nom i cognoms	Sistema
nahum.manuel.martin	A5S109PC22	20/6/2022 6:21:03	4h 56m 32s	Nahum Manuel Martin Vegas	Windows 10
nahum.manuel.martin	A5S109PC11	20/6/2022 7:53:54	3h 23m 41s	Nahum Manuel Martin Vegas	Linux
nahum.manuel.martin	A5S109PC12	20/6/2022 8:56:12	2h 21m 23s	Nahum Manuel Martin Vegas	Windows 10
nahum.manuel.martin	A5S109PC13	20/6/2022 9:59:58	1h 17m 37s	Nahum Manuel Martin Vegas	Linux
nahum.manuel.martin	A5S109PC14	20/6/2022 10:00:57	1h 16m 38s	Nahum Manuel Martin Vegas	Windows 10
nahum.manuel.martin	A5S109PC21	20/6/2022 10:05:02	1h 12m 33s	Nahum Manuel Martin Vegas	Linux
xavier.pina	A5S105PC25	20/6/2022 10:20:06	0h 57m 29s	Xavier Pina Suarez	Linux

Figura 21: Visual de la secció d'Equips
Font: Elaboració pròpia

A l'última pestanya del mòdul trobem l'historial que inicialment estarà buit. Per carregar dades hem de seleccionar un rang de dates mitjançant el Plug-in superior dretà que conté una icona d'un rellotge i què conté el text "Data inici - fins a - Data fi". Un cop seleccionat el rang temporal podem pressionar el botó inferior amb el text "Carregar dades" i realitzarà la càrrega de dades des de el Back-end. Un cop carregades les dades podem realitzar les mateixes operacions de cerca i filtratge que en les pestanyes anteriors utilitzant el requadre de cerca i els ordinadors de columnes.

- 🏠 Panell
- 📅 Calendaris ⁵
- 🎫 Tiquets ¹
- 📁 Préstecs ⁹
- 🖨️ Equips ¹

Supervisió d'equips

Panell ▶ Equips

Ara Edifici A5 Edifici B5 Edifici C6 Altres **Historial**

Historial de connexions

🕒

🕒

🕒

Carregar dades

Nom d'usuari	Ubicació	Hora de login	Hora de logout	Durada	Nom i cognoms	Sistema
angela.grau	ASS108PC11	14/6/2022 8:56:21	14/6/2022 13:31:33	4h 35m 12s	Maria Angela Grau Gotes	Windows 10
joel.josep.dill	ASS108PC26	14/6/2022 8:59:22	14/6/2022 11:52:03	2h 52m 41s	Joel Josep Dill Sarroca	Windows 10
xavier.caricchio	ASS108PC13	14/6/2022 9:00:29	14/6/2022 10:12:03	1h 11m 34s	Xavier Caricchio Hernandez	Windows 10
ferran.villar.rodriiguez	ASS108PC14	14/6/2022 9:04:24	14/6/2022 10:28:34	1h 24m 10s	Ferran Villar Rodriguez	Windows 10
david.garcia.tejeda	ASS108PC22	14/6/2022 9:15:40	14/6/2022 10:29:04	1h 13m 24s	David Garcia Tejeda	Linux
marta.granero.i	ASS108PC32	14/6/2022 9:15:59	14/6/2022 10:32:03	1h 16m 4s	Marta Granero I Marti	Windows 10
david.galvez.alcantara	ASS108PC44	14/6/2022 9:16:17	14/6/2022 10:32:33	1h 16m 16s	David Galvez Alcantara	Linux
alvaro.francesc.budria	ASS108PC10	14/6/2022 9:16:17	14/6/2022 10:32:33	1h 16m 16s	Alvaro Francesc Budria	Linux

Figura 22: Visual de la secció d'Equips en la sub-secció de l'Historial

Font: Elaboració pròpia

4. Conclusions

A data del redactat d'aquesta memòria i havent complert quasi bé tots els objectius funcionals de l'aplicació queda repassar la feina feta i treure conclusions de tot el procés de planificació i desenvolupament del projecte.

4.1 Assoliment dels objectius inicials

Si bé no tots els objectius planificats han resultat complerts cal dir que la majoria amplia s'han aconseguit. Tenim un sistema que recapta informació d'altres sistemes d'informació, ordena i resumeix les dades rebudes, avisa a l'usuari de la informació rellevant i guarda informació addicional per completar el servei. A més, proporciona una interfície clara, àgil, que s'actualitza automàticament i que compleix amb els requeriments modulars planejats des d'un inici.: Préstecs, tiquets, calendaris, connexions a equips i resum global.

A més el projecte s'ha dut a terme dintre d'un marge de temps raonable tot i els inconvenients trobats i retards. La concepció del sistema és quasi bé la inicial i va en consonància amb la resta de sistemes d'informació nous que s'estan creant a la UPC avui dia. De vuit setmanes planificades totals el projecte ha trigat deu en tenir tots els mòduls en funcionament, tot i que sempre poden sortir correccions que formen part de la fase de manteniment continu.

En quant a les hores i personal dedicats al projecte tenim que:

- Pel departament de Sistemes la Sandra Marsà ens reporta 10 hores en: creació d'un nou servidor, backups, instal·lació dockers, parches, etc.
- Jaume Moral en el suport de connexió i recollida de dades de les fonts de la UPC ens reporta unes 5 hores de feina.
- Per part d'un servidor les hores dedicades al projecte sobrepassen de molt les estimades, fins i tot es podien arribar a duplicar.

Al punt de "Problemes i dificultats" podem trobar les causes de l'augment d'hores dedicades.

4.2 Assoliment dels objectius personals

A nivell personal crec que el projecte ha suposat un gran esforç en quant a dedicació i aprenentatge però el resultat ha sigut molt satisfactori. Crec que aquest projecte té la capacitat d'escalar amb més serveis connectats i assumir més càrrega de serveis propi, és a dir, no provinent d'altres sistemes d'informació. El fet de fer servir una plataforma estàndard de la UPC ha facilitat la feina en certa mesura per què algunes decisions de disseny ja venen suggerides i no s'ha tingut la necessitat d'experimentar.

Veig GAROS com un projecte de futur que pot esdevenir la pedra angular del servei de Operació de la FIB i part de la de Sistemes; i potser també d'altres departaments de la UPC en un futur.

4.3 Problemes o dificultats

Com tot projecte no ha estat exempt de dificultats, le més destacables han sigut la formació en els Frameworks i els sistemes d'autenticació. Aquest últim ha sigut força farragós per què implicava altres departaments i calia que dediquessin temps al meu projecte per esbrinar alguns errors o mostrar-me el seu funcionament. La cerca d'informació i recursos a internet ha sigut clau per resoldre la majoria de problemes i mancances de formació, i moltes vegades ha sigut clau en la presa de decisions veure conceptes o mètodes fets de diferents formes.

També ens hem trobat que alguns continguts no tenien prou documentació i s'ha hagut de trucar i parlar amb persones que han tocat el tema. Altres vegades els procediments no estaven prou clars i s'ha tingut que investigar i realitzar proves de funcionament.

Una dificultat trobada força incòmoda era l'ordinador de la UPC destinat a ser l'equip de desenvolupament que per ser massa antic i no tenir suficient memòria suposar una veritable desesperació quan s'havien de posar en marxa els contenidors o realitzar la compilació. Per sort es va canviar a temps per un portàtil amb més potència i memòria que podia realitzar tota la feina de forma sobrada.

4.4 Feina futura

GAROS està pensat per ser la pedra angular de tota la informació que que requereix un usuari d'Operació a la FIB i per la part de Sistemes relacionat amb aquest. Com podem llegir a un apartat anterior "Objectius de futur" queden moltes funcionalitats i ampliacions per ficar dins del sistema. És més, a mida que avançava el projecte han sortit algunes de noves que es poden plantejar en un futur. Unes de les qüestions més útils es poder realitzar estadístiques d'ús i activitats en tots els mòduls del sistema com veure quins equips o aules són les més utilitzades dins i fora d'horari, analitzar el grau de morositat als préstecs, el grau de presencialitat i utilització de l'equipament, etc. Aquests anàlisi estadístics poden ser molt útils de cara a la presa de decisions, el que s'anomena Intel·ligència Empresarial^[43] o Bussiness Intelligence (BI) en anglès.

El fet de fer servir un Framework estès a la UPC facilita el fet de que qualsevol persona que domini Vue i Django^[27] sigui capaç de fer-se càrrec del manteniment i ampliació del sistema. S'ha tingut en compte de que totes les variables i funcions tinguin noms autodescriptius per facilitar la navegació pel codi a fi de facilitar l'accés a qualsevol nou programador futur. També està pensat fer un manual tècnic per al programador per el cas en que es vulgui exportar el sistema a altres departaments anàlegs.

4.5 Assoliment de les competències tècniques

Les competències tècniques són uns requeriments mínims per demostrar l'assoliment formatiu de la universitat i com s'ha aplicat al projecte de final de grau. Mirarem de resumir les influències formatives sobre cada una de les competències del projecte.

CSI2.1: Demostrar comprensión y aplicar los principios y las técnicas de gestión de calidad y de innovación tecnológica en las organizaciones. [Un poco]

S'ha tingut cura de la metodologia, els temps d'execució, la seguretat corporativa i l'eficiència en tots els aspectes, tant en temps com en diners guardant sempre la qualitat promesa.

CSI2.2: Concebir, desplegar, organizar y gestionar sistemas y servicios informáticos, en contextos empresariales o institucionales, para mejorar sus procesos de negocio, responsabilizarse y liderar su puesta en marcha, y su mejora continua; y valorar su impacto económico y social. [En profundidad]

Queda clar que un sistema que facilita la resposta dels serveis i agilitza la rebuda d'informació rellevant sempre és benvingut a qualsevol entorn de treball. Aquest era l'objectiu i s'ha complert com s'esperava. S'ha planificar, organitzat, gestionat i desenvolupat un sistema que millora la eficiència en l'entorn laboral a més de rebaixar l'estrès als llocs de treball al reduir la preocupació constant de la informació rellevant.

CSI2.3: Demostrar conocimiento y capacidad de aplicación de los sistemas de extracción y de gestión del conocimiento. [Bastante]

Haver aconseguit connectar i comunicar sistemes d'informació diversos en una sola plataforma informativa que analitza, organitza i cataloga totes les dades es mostra suficient per considerar la competència com a complerta.

CSI3.1: Demostrar comprensión de los principios de la evaluación de riesgos y aplicarlos correctamente en la elaboración y la ejecución de planes de actuación. [Bastante]

S'han complert les planificacions del projecte dintre dels terminis i pressupostos determinats fins a cert punt, ja que s'havia previst un cost addicional del 10% i finalment ha sigut del 20% només en l'àmbit temporal. Llevat dels endarreriments puntuals les previsions inicials han anat força a temps assolint quasi bé tots els objectius a nivell general.

CSI4.1: Participar activamente en la especificación de los sistemas de información y de comunicación. [Bastante]

L'especificació inicial és clau i al ser un projecte personal per la UPC ha sigut quasi bé completament pròpia.

CSI4.3: Administrar bases de datos (CES1.6). [Un poco]

Si bé l'àmbit de bases de dades pròpies no ha sigut massa extens, el accedir a diverses fonts d'informació podria veure's com interaccions amb altres bases de dades.

CSI3.3: Evaluar ofertas tecnológicas para el desarrollo de sistemas de información y gestión. [Bastante]

Pel que fa a l'estudi d'alternatives cal comentar que s'han mirat propostes alternatives en quant a Frameworks de desenvolupament tot i que des de un inici la suggerència de la UPC ha sigut la més encertada. No s'ha trobat cap sistema concret que pogués fer la feina d'aquest mateix sistema amb fonts d'informació tan particulars i pròpies com hi han a la UPC.

CSI3.4: Desarrollar soluciones de negocio mediante la implantación y la integración de hardware y software. [Bastante]

El fet de determinar la divisió contenidors, la diferenciació hardware de la base de dades i l'estructura comunicativa interna i externa del sistema pot demostrar aquesta competència.

CSI3.5: Proponer y coordinar cambios para mejorar la explotación del sistema y de las aplicaciones. [En profunditat]

El propi fet de proposar un sistema d'informació per agilitzar el treball esdevé en si una millora per a l'explotació del sistema i la resta d'aplicacions dels llocs de treballs implicats.

CSI1: Demostrar comprensión y aplicar los principios y las prácticas de las organizaciones, de manera que puedan ejercer como enlace entre las comunidades técnica y de gestión de una organización, y participar activamente en la formación de los usuarios. [Bastante]

El fet de facilitar i agilitzar els serveis del lloc de treball implica una agilització dels procediments laborals. Aquestes millores s'han provat amb els usuaris finals els que han donat la seva opinió en tot moment i han ajudat a millorar el sistema.

CSI4.2: Participar activamente en el diseño, la implementación y el mantenimiento de los sistemas de información y de comunicación. [En profunditat]

Com ja hem comentat es un projecte propi per a la UPC i tot el disseny i implementació han sigut pròpies. A més, també s'espera que faci el manteniment de tot el sistema d'informació mentre hi sigui al lloc de treball.

5. Bibliografia

- [1] Creació de la Unitat Transversal de Gestió de l'àmbit TIC, 2017. [online], Barcelona : Consell de Govern de la Universitat Politècnica de Catalunya, Disponible: <https://www.upc.edu/normatives/ca/documents/consell-de-govern/11-29-aprovacio-de-la-creacio-utg-a-mbit-tic.pdf>
- [2] inLab FIB. inLab FIB [online], [Accedit 1 Març 2022]. Disponible: <https://inlab.fib.upc.edu/>
- [3] CATALUNYA, UPC, Unitat Transversal de Gestió de l'Àmbit TIC Campus Nord (UTG CNTIC). Unitat Transversal de Gestió de l'Àmbit TIC Campus Nord [online]. [Accedit 1 Març 2022]. Disponible: <https://utgcntic.upc.edu/ca/la-utg>
- [4] Racó de la FIB. Raco.fib.upc.edu [online], [Accedit 1 Març 2022]. Disponible: <https://raco.fib.upc.edu>
- [5] eATIC Atenció als usuaris. Gn6.upc.edu [online], [Accedit 1 Març 2022]. Disponible: <https://gn6.upc.edu>
- [6] Redmine, 2006. Redmine.org [online], [Accedit 1 Març 2022]. Disponible: <https://www.redmine.org/>
- [7] Front end y back end - Wikipedia, la enciclopedia libre. Es.wikipedia.org [online], [Accedit 1 Març 2022]. Disponible: https://es.wikipedia.org/wiki/Front_end_y_back_end
- [8] Lambert, N., 2016. Microservicios: rompiendo el software monolítico. [online] Computerworld.es. [Accedit 1 Març 2022]. Disponible: <https://www.computerworld.es/innovacion/microservicios-rompiendo-el-software-monolitico>
- [9] Group, I., 2021. ¿Cómo evolucionará la adopción de tecnologías emergentes en 2022?. [online] Ituser.es. [Accedit 1 Març 2022]. Disponible: <https://www.ituser.es/estrategias-digitales/2021/12/como-evolucionara-la-adopcion-de-tecnologias-emergentes-en-2022>
- [10] Empowering App Development for Developers | Docker. Docker [online], [Accedit 1 Març 2022]. Disponible: <https://www.docker.com/>
- [11] Production-Grade Container Orchestration. Kubernetes [online], Disponible: <https://kubernetes.io>
- [12] Charts.js | Open source HTML5 charts for your website. [online], Disponible: <https://www.chartjs.org/>
- [13] MARK OTTO, AND BOOTSTRAP CONTRIBUTORS, Bootstrap. Getbootstrap.com [online]. [Accedit 1 Març 2022]. Disponible: <https://getbootstrap.com/>
- [14] What is Vuex? | Vuex. [online], Disponible: <https://vuex.vuejs.org/>
- [15] A Vue 3 UI Framework | Element Plus. [online], Disponible: <https://element-plus.org/>
- [16] Information Technology Infrastructure Library (ITIL) - Wikipedia.es. [online], Disponible: https://es.wikipedia.org/wiki/Information_Technology_Infrastructure_Library
- [17] Producte mínim viable - Viquipèdia, l'enciclopèdia lliure, 2022. Ca.wikipedia.org [online], [Accedit 1 Març 2022]. Disponible: https://ca.wikipedia.org/wiki/Producte_m%C3%ADnim_viable
- [18] CATALUNYA, UPC, Identitat Digital UPC. - Universitat Politècnica de Catalunya. Identitat Digital UPC [online]. [Accedit 1 Març 2022]. Disponible: <https://www.upc.edu/identitatdigital/ca>
- [19] Booked - Simply Powerful Scheduling. Bookedscheduler.com [online], [Accedit 1 Març 2022]. Disponible: <https://www.bookedscheduler.com/>
- [20] Metodologia àgil - Viquipèdia, l'enciclopèdia lliure, 2022. Ca.wikipedia.org [online], [Accedit 1 Març 2022]. Disponible: https://ca.wikipedia.org/wiki/Metodologia_%C3%A0gil
- [21] Extreme Programming: A Gentle Introduction. Extremeprogramming.org [online], [Accedit 1 Març 2022]. Disponible: <http://www.extremeprogramming.org/>

- [22] UPC webapp ui Vue 3. gitlab.upc.edu [online], [Accedit 1 Març 2022].
Disponible: <https://gitlab.upc.edu/f1/upc-webapp-ui-vue3>
- [23] Vue.js - The Progressive JavaScript Framework | Vue.js. [Vuejs.org](https://vuejs.org) [online], [Accedit 1 Març 2022].
Disponible: <https://vuejs.org/>
- [24] Node.js. [Node.js](https://nodejs.org) [online], [Accedit 1 Març 2022]. Disponible: <https://nodejs.org>
- [25] JavaScript - Viquipèdia, l'enciclopèdia lliure. [Ca.wikipedia.org](https://ca.wikipedia.org/wiki/JavaScript) [online], [Accedit 1 Març 2022].
Disponible: <https://ca.wikipedia.org/wiki/JavaScript>
- [26] Welcome to Python.org. [Python.org](https://www.python.org/) [online], [Accedit 1 Març 2022]. [Accedit 1 Març 2022].
Disponible: <https://www.python.org/>
- [27] The web Framework for perfectionists with deadlines | Django. [Djangoproject.com](https://www.djangoproject.com/) [online], [Accedit 1 Març 2022]. Disponible: <https://www.djangoproject.com/>
- [28] Entorn integrat de desenvolupament - Viquipèdia, l'enciclopèdia lliure. [Ca.wikipedia.org](https://ca.wikipedia.org/wiki/Entorn_integrat_de_desenvolupament) [online], [Accedit 1 Març 2022]. Disponible:
https://ca.wikipedia.org/wiki/Entorn_integrat_de_desenvolupament
- [29] Visual Studio Code - Code Editing. Redefined. [Code.visualstudio.com](https://code.visualstudio.com/) [online], [Accedit 1 Març 2022]. Disponible: <https://code.visualstudio.com/>
- [30] Iterate faster, innovate together | GitLab. [gitlab.com](https://about.gitlab.com/) [online], [Accedit 1 Març 2022]. Disponible:
<https://about.gitlab.com/>
- [31] Free Bootstrap Admin Template - AdminLTE.IO. [AdminLTE.IO](https://adminlte.io/) [online], [Accedit 1 Març 2022].
Disponible: <https://adminlte.io/>
- [32] Bootstrap Admin & Dashboard Template · CoreUI. [Coreui.io](https://coreui.io/) [online], [Accedit 1 Març 2022].
Disponible: <https://coreui.io/>
- [33] Premium dashboard template with responsive and high quality UI. [Tabler.io](https://tabler.io/) [online], [Accedit 1 Març 2022]. Disponible: <https://tabler.io/>
- [34] Welcome to Flask — Flask Documentation (2.0.x). [Flask.palletsprojects.com](https://flask.palletsprojects.com/) [online], [Accedit 1 Març 2022]. Disponible: <https://flask.palletsprojects.com>
- [35] Ruby on Rails. [Ruby on Rails](https://rubyonrails.org/) [online], [Accedit 1 Març 2022]. Disponible: <https://rubyonrails.org/>
- [36] Laravel - The PHP Framework For Web Artisans. [Laravel.com](https://laravel.com/) [online], [Accedit 1 Març 2022].
Disponible: <https://laravel.com/>
- [37] GitHub: Where the world builds software. [GitHub](https://github.com/) [online], [Accedit 1 Març 2022]. Disponible:
<https://github.com/>
- [38] Software de administració de proyectos | Microsoft Project. [Microsoft.com](https://www.microsoft.com/es-es/microsoft-365/project/project-management-software) [online], [Accedit 1 Març 2022]. Disponible:
<https://www.microsoft.com/es-es/microsoft-365/project/project-management-software>
- [39] Trello. [Trello.com](https://trello.com/) [online], [Accedit 1 Març 2022]. Disponible: <https://trello.com/>
- [40] Cloud Computing Services - Amazon Web Services (AWS). [Amazon Web Services, Inc.](https://aws.amazon.com) [online],
Disponible: <https://aws.amazon.com>
- [41] [nahum.manuel.martin / GAROS · Gitlab.](https://gitlab.upc.edu/nahum.manuel.martin/garos) [online], Disponible:
<https://gitlab.upc.edu/nahum.manuel.martin/garos>
- [42] DBeaver Community | Free Universal Database Tool. [online], Disponible: <https://dbeaver.io/>
- [43] Intel·ligència empresarial - Viquipèdia, l'enciclopèdia lliure. [online], Disponible:
https://ca.wikipedia.org/wiki/Intel%C2%B7lig%C3%A8ncia_empresarial