



UNIVERSITAT DE
BARCELONA

Treball de Fi de Grau

GRAU D'ENGINYERIA INFORMÀTICA

**Facultat de Matemàtiques i Informàtica
Universitat de Barcelona**

**DESENVOLUPAMENT D'UNA WEB DE GESTIÓ
D'EQUIPS DE FUTBOL AMATEURS**

Marta Bernadas Portas

Director: Daniel Geruous Fares
Realitzat a: Departament de
Matemàtiques i Informàtica
Barcelona, 10 de juny de 2025

Agraïments

En primer lloc, voldria donar les gràcies al meu tutor, Daniel Geruous, per la seva guia i suport al llarg de tot aquest procés. Els seus consells i orientacions han estat fonamentals per al desenvolupament d'aquest projecte.

També vull agrair a la meva família per haver estat sempre al meu costat i pel suport proporcionat al llarg de tot aquest procés.

Resum

En el fútbol amateur, els entrenadors han d'assumir moltes tasques organitzatives amb recursos limitats, poc temps i sense suport tecnològic. Encara que existeixen aplicacions per facilitar aquesta feina, la majoria estan pensades per a clubs professionals i resulten massa complexes o costoses per a equips petits.

Aquest treball de final de grau busca desenvolupar una aplicació web senzilla i intuïtiva per donar resposta a les necessitats dels entrenadors d'equips amateurs. L'eina permet gestionar la plantilla, programar entrenaments i partits, controlar l'assistència, preparar alineacions, consultar estadístiques bàsiques dels jugadors i utilitzar una pissarra tàctica visual per planificar exercicis de manera gràfica.

El projecte s'ha desenvolupat seguint una metodologia incremental, estructurada en les fases d'anàlisi, disseny, implementació, proves i desplegament. Per a la seva implementació, s'ha aplicat una arquitectura modular basada en capes per al backend (FastAPI) i una arquitectura basada en components per al frontend (Vue.js). Les funcionalitats s'han desenvolupat mitjançant una estratègia iterativa, i s'han realitzat proves unitàries, d'integració, d'usabilitat i end-to-end per validar el correcte funcionament del sistema. També s'ha mesurat la cobertura del codi i s'ha dut a terme el desplegament a AWS utilitzant una infraestructura escalable.

Els resultats obtinguts mostren que l'aplicació és funcional, accessible i útil per als equips modestos, i representa una solució realista que permet facilitar el dia a dia als seus entrenadors.

Resumen

En el fútbol amateur, los entrenadores deben asumir muchas tareas organizativas con recursos limitados, poco tiempo y sin apoyo tecnológico. Aunque existen aplicaciones para facilitar este trabajo, la mayoría están pensadas para clubes profesionales y resultan demasiado complejas o costosas para equipos pequeños.

Este trabajo de final de grado tiene como objetivo desarrollar una aplicación web sencilla e intuitiva que responda a las necesidades de los entrenadores de equipos amateurs. La herramienta permite gestionar la plantilla, programar entrenamientos y partidos, controlar la asistencia, preparar alineaciones, consultar estadísticas básicas de los jugadores y utilizar una pizarra táctica visual para planificar ejercicios de forma gráfica.

El proyecto se ha desarrollado siguiendo una metodología incremental, estructurada en las fases de análisis, diseño, implementación, pruebas y despliegue. Para su implementación, se ha aplicado una arquitectura modular basada en capas para el backend (FastAPI) y una arquitectura basada en componentes para el frontend (Vue.js). Las funcionalidades se han desarrollado mediante una estrategia iterativa, y se han realizado pruebas unitarias, de integración, de usabilidad y end-to-end para validar el correcto funcionamiento del sistema. También se ha medido la cobertura del código y se ha llevado a cabo el despliegue en AWS utilizando una infraestructura escalable.

Los resultados obtenidos muestran que la aplicación es funcional, accesible y útil para equipos modestos, y representa una solución realista que facilita el día a día de sus entrenadores.

Abstract

In amateur football, coaches must take on many organizational tasks with limited resources, little time, and no technological support. Although there are applications designed to assist with this work, most are intended for professional clubs and are too complex or expensive for small teams.

This final degree project aims to develop a simple and intuitive web application that meets the needs of amateur team coaches. The tool allows for squad management, training and match scheduling, attendance tracking, lineup preparation, viewing basic player statistics, and using a visual tactical board to graphically plan exercises.

The project was developed following an incremental methodology, structured in the phases of analysis, design, implementation, testing, and deployment. For its implementation, a modular layered architecture was used for the backend (FastAPI) and a component-based architecture for the frontend (Vue.js). The functionalities were developed using an iterative strategy, and unit, integration, usability, and end-to-end testing was carried out to ensure the system's correct operation. Code coverage was also measured, and the deployment was carried out on AWS using a scalable infrastructure.

The results show that the application is functional, accessible, and useful for small teams, representing a realistic solution that helps simplify the daily tasks of their coaches.

Índex

1.	Introducció	1
1.1.	Context	1
1.2.	Estat de l'art.....	1
1.2.1.	SportEasy.....	1
1.2.2.	360Player	1
1.2.3.	Clupik	1
1.2.4.	Sportlyzer	2
1.3.	Motivació	2
2.	Objectius	3
3.	Planificació	4
4.	Costos	6
4.1.	Recursos humans.....	6
4.2.	Equipament	6
4.3.	Infraestructura	7
5.	Anàlisi	9
5.1.	Entrevistes.....	9
5.2.	Anàlisi de requisits.....	11
5.2.1.	Requisits Funcionals	11
5.2.2.	Requisits no funcionals.....	11
5.3.	Model de domini	12
5.4.	Casos d'ús i històries d'usuari	13
5.4.1.	Autenticació i gestió d'equips	13
5.4.2.	Gestió de jugadors	14
5.4.3.	Gestió de calendari, entrenaments i partits	15
5.4.4.	Descripció de cas d'ús	18
5.4.5.	Història d'usuari detallada	19
6.	Desenvolupament	21
6.1.	Tecnologies	21
6.1.1.	Backend	21
6.1.2.	Frontend	21
6.1.3.	Proves	22
6.1.4.	Desplegament al núvol	22
6.2.	Arquitectura del sistema	23
6.2.1.	Arquitectura per capes	23
6.2.2.	Arquitectura basada en components.....	24

6.3.	Model de dades	25
6.3.1.	Entitats	25
6.3.2.	Taules d'enllaç	27
6.3.3.	Relacions.....	28
6.3.4.	Diagrama entitat-relació	29
6.4.	Implementació del backend	29
6.4.1.	Estructura del backend.....	29
6.4.2.	Diagrames d'arquitectura	30
6.4.3.	Correspondència amb l'arquitectura per capes	33
6.4.4.	Diagrames de seqüència de les operacions HTTP	34
6.5.	Implementació del frontend	39
6.5.1.	Estructura del frontend	40
6.5.2.	Diagrama de components	40
6.5.3.	Navegació	42
6.6.	Validació del sistema	43
6.6.1.	Proves unitàries.....	43
6.6.2.	Proves d'integració	44
6.6.3.	Proves end-to-end.....	44
7.	Resultats	45
7.1.	Cobertura de proves	45
7.1.1.	Mòdul api.....	45
7.1.2.	Mòdul core.....	46
7.1.3.	Mòdul crud.....	46
7.1.4.	Mòdul models	46
7.1.5.	Mòdul app.....	47
7.2.	Prova d'usabilitat	47
8.	Desplegament	49
9.	Conclusions	51
9.1.	Treball futur	51
10.	Referències	52
11.	Annexos	55
	Annex 1. Descripció dels casos d'ús	55
	Annex 2. Descripció de les històries d'usuari	73

1. Introducció

1.1. Context

El futbol és actualment un dels esports més practicats i seguits arreu del món. La gestió eficient d'un equip, fins i tot en l'àmbit amateur, representa una dificultat important per als entrenadors, ja que han de coordinar entrenaments, planificar partits, fer seguiment del rendiment dels jugadors i mantenir una bona comunicació amb tots els membres de l'equip.

En els darrers anys, la digitalització també ha arribat a la gestió esportiva, i cada cop és més habitual l'ús de plataformes tecnològiques per facilitar aquestes tasques. No obstant això, moltes d'aquestes solucions estan pensades per a clubs professionals o semiprofessionals, amb més recursos i cos tècnic. En conseqüència, aquestes plataformes poden resultar massa complexes o inaccessibles per amateurs.

Els equips amateurs acostumen a disposar d'un pressupost molt reduït i sense personal especialitzat en tecnologia. Per tant, requereixen eines senzilles, intuïtives i de baix cost (o gratuïtes), que cobreixin les seves necessitats reals sense requerir coneixements tècnics avançats.

1.2. Estat de l'art

En els darrers anys, l'ús de la tecnologia en el món del futbol ha anat en augment. Aquestes eines s'han convertit en una solució útil per a la coordinació d'activitats, el seguiment del rendiment dels jugadors i la millora de la comunicació interna dels equips. No obstant això, gran part de les aplicacions disponibles al mercat estan pensades per a clubs professionals o semiprofessionals, que compten amb un cos tècnic més ampli i disposen de més recursos econòmics. Aquesta orientació professional fa que aquestes plataformes siguin poc pràctiques o inaccessibles per a equips amateurs.

1.2.1. SportEasy

SportEasy [1] és una de les solucions més conegudes i destaca per ser dirigida a equips no professionals. L'aplicació permet gestionar entrenaments, partits, convocatòries i comunicacions dins de l'equip. La seva interfície és clara i fàcil d'utilitzar, fet que facilita l'ús de la plataforma a tots els entrenadors independentment dels seus coneixements tecnològics. Tot i això, la versió gratuïta és força limitada i només ofereix una part de les funcionalitats disponibles. Això suposa un inconvenient per a equips amb recursos limitats, que es veuen obligats a fer servir el pla de pagament per a cobrir les necessitats bàsiques.

1.2.2. 360Player

360Player [2] ofereix una proposta centrada en l'anàlisi esportiva i el desenvolupament individual dels jugadors. Inclou eines com el seguiment estadístic detallat, l'anàlisi de vídeo i la planificació dels entrenaments. És una eina pensada per a equips professionals, ja que compta amb funcionalitats que busquen millorar el rendiment esportiu de manera metòdica. Aquests equips disposen de personal especialitzat en aquestes tasques. La seva complexitat i el fet que només estigui disponible mitjançant subscripció fan que, per a molts equips amateurs, sigui una opció inviable.

1.2.3. Clupik

Clupik [3] és una plataforma que ofereix una gestió completa del club, més enllà de l'àmbit esportiu. A més de gestionar entrenaments i partits, també permet controlar inscripcions, pagaments, botiga en línia, i comunicacions amb famílies i socis. Es tracta d'una plataforma

molt útil per a entitats amb diversos equips i una estructura consolidada. Tanmateix, per a un equip amateur, amb menys necessitats i una gestió més simple, Clupik pot resultar excessivament carregada de funcionalitats. A més, moltes de les opcions més útils només estan disponibles a través de plans de pagament, fet que pot condicionar-ne l'ús.

1.2.4. Sportlyzer

Finalment, cal destacar **Sportlyzer** [4], una aplicació que se centra en el seguiment del progrés dels jugadors, especialment útil en l'àmbit formatiu. Inclou funcions per controlar l'assistència, monitorar l'evolució individual i facilitar la comunicació entre entrenadors, jugadors i famílies. Aquesta orientació pedagògica fa que Sportlyzer sigui especialment adequada per a equips de futbol base. Tot i això, la seva estructura està pensada per a clubs amb més d'un equip i amb una certa estructura, cosa que pot limitar-ne l'ús en clubs més petits. Com en altres casos, les funcionalitats més avançades queden restringides a les versions de pagament.

1.3. Motivació

Aquest projecte sorgeix de la meva experiència dins del futbol amateur. Com a jugadora en actiu d'un equip de Primera Catalana femenina, he pogut comprovar en primera persona les dificultats amb què es troben molts entrenadors a l'hora de gestionar l'equip. Tasques com organitzar convocatòries, planificar entrenaments o fer seguiment de l'assistència, que poden semblar senzilles, acaben suposant una càrrega de temps i esforç per als entrenadors, especialment quan es disposa de recursos limitats, tant humans com materials. Tot i que existeixen diverses aplicacions dedicades a la gestió esportiva, la majoria estan pensades per a clubs professionals o amb estructures consolidades. Moltes de les funcionalitats més útils només es troben en plans de pagament i, a més, aquestes plataformes poden resultar massa complexes per a entrenadors d'equips no professionals. Aquesta situació deixa molts equips amateurs, que tenen necessitats clares però més senzilles, sense cap eina que s'adapti realment a la seva realitat.

2. Objectius

L'objectiu principal d'aquest projecte és dissenyar i desenvolupar una eina digital senzilla, intuïtiva i accessible que ajudi als entrenadors d'equips de futbol amateur gestionar de manera eficient les seves tasques habituals.

Per assolir aquest objectiu general, es defineixen els següents objectius específics:

- Identificar les necessitats reals dels equips amateurs, mitjançant l'anàlisi de la seva dinàmica de treball.
- Dissenyar una interfície d'usuari clara i accessible, adaptada a entrenadors amb diferents nivells de competència digital.
- Desenvolupar funcionalitats essencials per al dia a dia dels equips, com la gestió de plantilles i jugadors, la planificació d'entrenaments i partits i el control d'assistència.
- Garantir l'eficiència i la qualitat tècnica de l'eina mitjançant la realització de proves unitàries, d'integració i d'usabilitat que assegurin un bon funcionament i una experiència d'ús satisfactòria.
- Publicar i desplegar l'aplicació en un entorn accessible en línia perquè qualsevol equip amateur pugui accedir-hi i fer-ne ús.

3. Planificació

El projecte s'ha dividit en cinc fases que han marcat les diferents etapes del desenvolupament, des de la definició inicial fins al lliurament final. A continuació es detallen cadascuna de les fases:

1. Fase d'anàlisi (del 3 al 19 de març de 2025)

Durant aquesta etapa es duu a terme la recollida d'informació sobre les necessitats reals dels entrenadors d'equips amateurs. Es realitzen entrevistes, es defineixen els requisits funcionals i no funcionals del sistema, es redacten els casos d'ús i les històries d'usuari.

2. Fase de disseny (del 20 al 26 de març de 2025)

En aquesta fase es defineix l'arquitectura general del sistema, el model de dades, els prototips de la interfície d'usuari i el model de domini.

3. Fase d'implementació (del 27 de març al 7 de maig de 2025)

Aquesta fase, organitzada en sprints setmanals, es desenvolupen les funcionalitats principals del sistema: gestió d'usuaris, equips, jugadors, entrenaments i partits. Seguint una estratègia incremental, amb entregues parcials i funcionals al final de cada sprint.

4. Fase de proves (del 8 al 21 de maig de 2025)

Un cop completades les funcionalitats bàsiques, es realitzen una bateria de proves per assegurar la qualitat i el correcte funcionament del sistema:

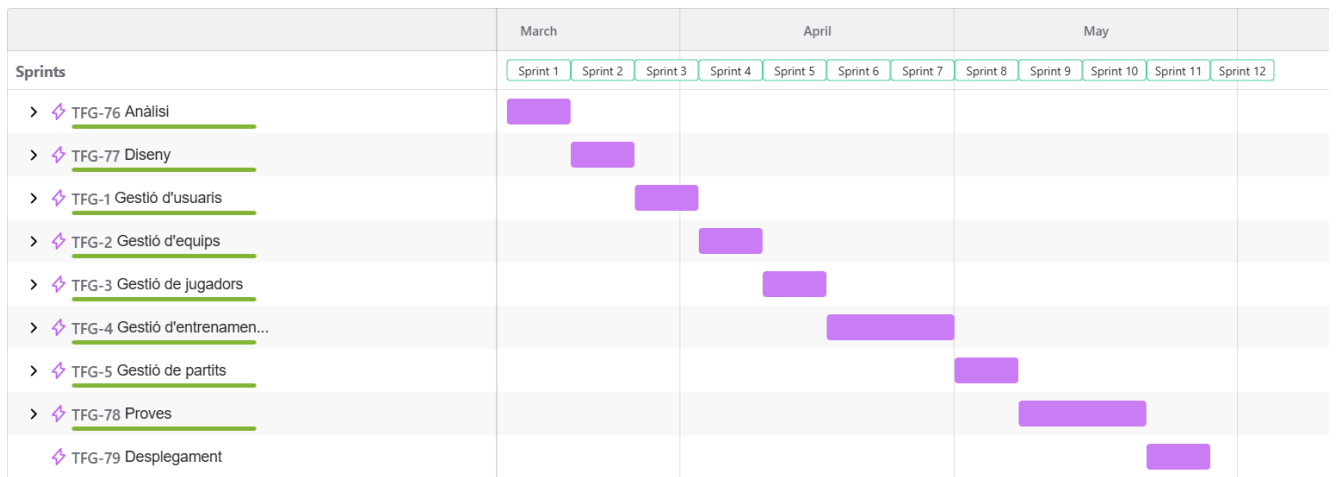
- Proves unitàries per comprovar el correcte funcionament de funcions específiques de la lògica del sistema.
- Proves d'integració per assegurar la coherència entre mòduls.
- Proves end-to-end per verificar el comportament del sistema en escenaris reals d'ús.
- Proves d'usabilitat, per validar la facilitat d'ús de la interfície.

5. Fase de desplegament (del 22 al 28 de maig de 2025)

Finalitzades les proves, es procedeix amb el desplegament de l'eina en l'entorn de producció.

El desenvolupament del projecte ha seguit de manera fidel la planificació prevista, amb totes les fases completades dins dels terminis previstos. Aquesta organització ha permès un bon ritme de treball i una execució ordenada de les tasques.

A continuació, es mostra un diagrama de Gantt que resumeix visualment tota la planificació del projecte. En aquest es poden identificar els sprints setmanals, les diferents fases del desenvolupament i reflecteix clarament la seva durada i ordre d'execució al llarg del temps.



Il·lustració 1: Diagrama de Gantt corresponent a la planificació inicial del projecte

4. Costos

El desenvolupament i posada en marxa d'aquesta aplicació web implica una sèrie de costos. Tot i que es tracta d'un projecte de final de grau, s'ha realitzat una estimació dels recursos necessaris i els costos que suposaria empresarialment realitzar aquest desenvolupament. El pressupost s'estructura en tres grans blocs: recursos humans, equipament i infraestructura.

4.1. Recursos humans

En un entorn empresarial, el desenvolupament d'una aplicació com aquesta requeriria la participació de diversos perfils professionals especialitzats. S'ha fet una estimació del cost, tenint en compte els salaris mitjans del sector tecnològic a Espanya [5] [6] [7]. Els perfils considerats són els següents:

- **Desenvolupador/a full-stack:** responsable de dissenyar i implementar tant el backend com el frontend, incloent l'arquitectura del sistema, la connexió amb la base de dades, la lògica de negoci i la integració de serveis.
- **Dissenyador/a UX/UI:** intervé en les fases inicials, creant prototips interactius i desenvolupant el disseny visual de la interfície, garantint que l'aplicació sigui intuïtiva, usable i coherent.
- **Tècnic/a de QA (Quality Assurance):** participa en la fase final per validar que l'aplicació compleixi els requisits, detectar possibles errors i assegurar la seva estabilitat.

A continuació es presenta una taula amb la dedicació estimada per a cada perfil, el cost per hora i el cost total:

Concepte	Hores	Preu/hora (€)	Cost total (€)
Desenvolupador/a full-stack	250 h	20 €	5.000 €
Dissenyador/a UX/UI	20 h	16 €	320 €
QA	10 h	15 €	150 €
Subtotal recursos humans			5.470 €

Taula 1: Costos estimats de recursos humans per al desenvolupament del projecte

4.2. Equipament

Per al desenvolupament del projecte, es requereixen tres ordinadors portàtils, un per a cada membre de l'equip. El model escollit és l'ASUS Vivobook 15 F1504ZA-NJ1447W [8], ja que ofereix una bona relació qualitat-preu i unes prestacions adequades per a tasques de programació, disseny i proves. Aquest portàtil disposa d'una pantalla de 15,6 polzades amb resolució Full HD, un processador Intel® Core™ i7-1255U de 10 nuclis, 16 GB de memòria RAM, un disc SSD d'1 TB per a l'emmagatzematge, sistema operatiu Windows 11 Home i gràfics integrats Intel Iris Xe.

El cost total de l'equipament és el següent:

Concepte	Quantitat	Preu unitari (€)	Cost total (€)
Ordinador portàtil	3 unitats	629 €	1.887 €
Subtotal equipament			1.887 €

Taula 2: Costos estimats d'equipament per al desenvolupament del projecte

4.3. Infraestructura

Per desplegar l'aplicació al núvol, s'han utilitzat diversos serveis d'Amazon Web Services (AWS) [\[9\]](#) [\[10\]](#) [\[11\]](#) [\[12\]](#) [\[13\]](#), una plataforma àmpliament utilitzada en entorns professionals per la seva escalabilitat i disponibilitat. El càlcul dels costos es basa en un ús moderat, adequat per a una aplicació web en fase inicial. Els serveis i funcionalitats considerats són:

- **Amazon S3:** per emmagatzemar fitxers estàtics i gestionar les sol·licituds corresponents.
- **CloudFront:** com a CDN per millorar el rendiment global de l'aplicació.
- **AWS Lambda i API Gateway:** per implementar la part lògica del backend en un entorn serverless.
- **Amazon RDS:** per gestionar una base de dades relacional (PostgreSQL) amb alta disponibilitat.
- **Domini web** propi i certificat SSL gratuït a través d'**AWS Certificate Manager**.

El desglossament detallat dels costos estimats anuals és:

Servei	Ús estimat mensual	Preu unitari (€)	Cost mensual (€)	Cost anual (€)
Amazon S3 (Emmagatzematge)	5 GB	0,023 €/GB	0,12 €	1,44 €
Amazon S3 (Sol·licituds)	1.000.000 sol·licituds	0,00000035 €/sol·licitud	0,35 €	4,20 €
Amazon CloudFront (Transferència sortint regional al origen)	50 GB	0,018 €/GB	0,90 €	10,80 €
Amazon CloudFront (Sol·licituds HTTPS)	1.000.000 sol·licituds	0,011 €/sol·licitud	1,10 €	13,20 €
AWS Lambda (Execucions)	1.000.000 execucions	0,176 €/milió	0,18 €	2,16 €
AWS Lambda (Temps d'execució)	128 MB, 100 ms	0,00001467 €/GB-s	0,19 €	2,28 €
Amazon API Gateway	1.000.000 sol·licituds	1,03 €/milió	1,03 €	12,36 €
Amazon RDS (Instància db.t3.micro)	730 h al mes	0,018 €/h	13,14 €	157,68 €
Amazon RDS (Emmagatzematge)	20 GB	0,106 €/GB	2,12 €	25,44 €
Domini web	1 domini anual	15,00 €/any	1,25 €	15,00 €
Certificat SSL (ACM)	Inclòs amb AWS	0,00 €	0,00 €	0,00 €
Subtotal infraestructura			20.38 €	244,56 €

Taula 3: Costos estimats d'infraestructura per al desenvolupament del projecte

Aquest import representa una estimació del cost que tindria el desenvolupament i posada en marxa de l'aplicació en un entorn professional, tenint en compte una càrrega de treball moderada i un equip reduït. Cal destacar que la major part del cost està relacionada amb els recursos humans, ja que són els qui dediquen més temps i esforç a dissenyar, programar i assegurar que tot funcioni correctament.

Categoria	Cost (€)
Recursos humans	5.470 €
Equipament	1.887 €
Infraestructura	244,56 €
Total projecte	7.601,56 €

Taula 4: Costos estimats totals per al desenvolupament del projecte

5. Anàlisi

5.1. Entrevistes

Per tal d'entendre millor les necessitats i els reptes que afronten els entrenadors de futbol amateur en la gestió dels seus equips, es van realitzar entrevistes semiestructurades a tres entrenadors. Aquests entrenadors dirigeixen equips de diferents categories, concretament quarta catalana masculina, segona divisió femenina juvenil i primera catalana femenina, i compten amb diferents contextos esportius. Aquesta diversitat permet obtenir una visió més àmplia, ja que inclou tant l'àmbit masculí com femení i diferents nivells de competició.

Durant les entrevistes, es va presentar la finalitat del projecte i es van plantejar preguntes enfocades a conèixer com gestionen actualment els seus equips, quines eines utilitzen, quines tasques consideren més importants i quins problemes o dificultats troben en el dia a dia. També es va explorar l'interès per funcionalitats específiques que podrien facilitar la gestió, com la planificació d'entrenaments, la preparació d'alineacions, el seguiment de la disponibilitat i estat físic dels jugadors, la comunicació interna, i l'accés a estadístiques i documents.

A partir de les respostes recollides, es van definir tres perfils d'usuari que representen les característiques, necessitats i objectius principals dels possibles usuaris de la plataforma. Aquests perfils ajuden a comprendre millor les diferents realitats i reptes als quals s'enfronten els entrenadors segons la categoria, el nivell i el context esportiu.

A continuació, es presenten els perfils d'usuari definits a partir de les entrevistes realitzades:

 JORDI LÓPEZ Entrenador de futbol Edat: 42 Sexe: Hombre Llicència: UEFA B Categoria: Segona Divisió Juvenil Femenina	CONTEXT En Jordi dirigeix un equip juvenil a segona divisió femenina format per joves jugadores que concilien el futbol amb els estudis. L'equip entrena tres vegades per setmana amb l'objectiu clar d'aconseguir un ascens de categoria.	OBJECTIUS • Analitzar el rendiment de cada jugadora per adaptar els entrenaments a les seves necessitats i potenciar el seu desenvolupament tècnic i tàctic. • Crear sessions més efectives amb l'objectiu de millorar el rendiment global de l'equip.
	CARACTERÍSTIQUES • Fomenta un bon ambient de cohesió dins de l'equip • S'adapta a les necessitats individuals de les jugadores per afavorir la seva evolució personal i col·lectiva. • Treballa per mantenir una dinàmica positiva, on cada jugadora se senti valorada	NECESSITATS • Centralitzar la informació de convocatòries, assistències, entrenaments, partits • Obtenir dades concretes del rendiment de les jugadores després dels partits per poder treballar amb elles de manera més personalitzada.
	TECNOLOGIES • Ús de WhatsApp per a comunicacions • Planificació dels entrenaments i partits de manera manual, en paper o , en un document digital	DIFICULTATS • Gestionar la càrrega de minuts, ja que algunes jugadores juguen dos partits durant el cap de setmana. • Falta de jugadores per motius d'estudi, fet que complica la planificació i la preparació de l'equip.

Il·lustració 2: Perfil d'usuari de l'entrenador de Segona Divisió Juvenil Femenina



MARC PÉREZ
Entrenador de futbol

Edat: 46
Sexe: Hombre
Llicència: UEFA B
Categoria: Quarta Catalana Masculina

CONTEXT

En Marc és entrenador d'un equip de Quarta Catalana format per jugadors amateurs que compaginen el futbol amb estudis i feines. L'equip entrena dues vegades per setmana i juga els caps de setmana. No compta amb un cos tècnic ampli, i per tant, ha d'assumir múltiples tasques: planificació d'entrenaments, gestió de la plantilla, comunicació amb els jugadors i seguiment del rendiment.

OBJECTIUS

- Organitzar millor els entrenaments i la preparació dels partits.
- Disposar d'un espai únic on recollir tota la informació de l'equip.
- Reduir el temps dedicat a tasques de coordinació per centrar-se en la part esportiva.

CARACTERÍSTIQUES

- Metòdic i implicat, s'adapta a les necessitats de l'equip.
- Bona visió tàctica i compromís amb els entrenaments.
- Fomenta una bona dinàmica i la millora continua individual i col·lectiva.
- Obert a noves tecnologies per millorar la gestió.

NECESSITATS

- Una eina econòmica i intuïtiva que centralitzi la informació de l'equip.
- Funcionalitats clau que reagrupin comunicació, planificació d'entrenaments i seguiment de dades bàsiques.

TECNOLOGIES

- WhatsApp per a comunicacions amb els jugadors.
- Fulls d'Excel per portar el control d'assistències, convocatòries i estadístiques bàsiques.

DIFICULTATS

- Manca de temps per fer un seguiment detallat dels jugadors i de l'evolució de l'equip.
- Coordinar un grup amb disponibilitat canviant.
- Excés de tasques manuals, que fa que la informació no sempre estigui actualitzada ni sigui fàcilment accessible.

Il·lustració 3: Perfil d'usuari de l'entrenador de Quarta Catalana Masculina



JOAN MARTÍNEZ
Entrenador de futbol

Edat: 43
Sexe: Hombre
Llicència: UEFA A
Categoria: Primera Catalana Femenina

CONTEXT

En Joan dirigeix un equip sènior a primera Catalana femenina en un club modest amb recursos limitats. L'equip entrena tres vegades per setmana i competeix els caps de setmana, combinant jugadores amb diferents nivells d'experiència i disponibilitat. Sense un cos tècnic ampli, ha d'assumir múltiples responsabilitats, des de la preparació tàctica fins a la gestió del grup.

OBJECTIUS

- Organitzar entrenaments adaptats als punts febles dels rivals i a la disponibilitat dels jugadors.
- Definir alineacions i estratègies tàctiques de cara als partits
- Millorar la coordinació i comunicació interna

CARACTERÍSTIQUES

- Metòdic i analític, amb una forta orientació tàctica.
- Organitzat però flexible davant imprevistos.
- Interessat en eines digitals que li permetin organitzar millor l'equip sense incrementar la seva càrrega de feina.

NECESSITATS

- Automatitzar el seguiment de disponibilitats i assistències.
- Disposar d'un espai per tenir documents i vídeos.
- Integrar un sistema de missatgeria per estructurar millor la comunicació.

TECNOLOGIES

- Ús de WhatsApp per a comunicacions
- Utilitza una llibreta per anotar aspectes tàctics, físics i observacions dels partits.

DIFICULTATS

- La informació es dispersa en converses de WhatsApp.
- El seguiment manual de les dades consumeix temps i pot provocar errors.
- Les absències imprevistes dels jugadors alteren la planificació dels entrenaments.

Il·lustració 4: Perfil d'usuari de l'entrenador de Primera Catalana Femenina

Aquests perfils d'usuari permeten posar en perspectiva les diverses necessitats i reptes que afronten els entrenadors de futbol amateur en diversos contextos. Tot i que cada entrenador prioritza aspectes tàctics diferents segons la seva categoria i nivell competitiu, i mostra

preferències específiques respecte a les funcionalitats que consideren més útils, es poden identificar dificultats comunes entre ells.

Principalment, tots coincideixen en la necessitat d'automatitzar les tasques rutinàries per tal d'estalviar temps. A més, expressen la importància de disposar d'una eina que centralitzi tota la informació relacionada amb l'equip per evitar la dispersió de dades i facilitar la presa de decisions.

Finalment, també coincideixen en que, per als seus jugadors, el futbol no és l'activitat principal, ja que el compaginen amb la feina o els estudis, fet que condiciona la seva disponibilitat i provoca habitualment absències de jugadors.

5.2. Anàlisi de requisits

A partir de les entrevistes realitzades als entrenadors, s'han identificat les funcionalitats i característiques que la plataforma hauria d'oferir per adaptar-se a les seves necessitats i facilitar la gestió dels seus equips.

5.2.1. Requisits Funcionals

Els requisits funcionals recullen les accions que els usuaris han de poder realitzar dins la plataforma. Aquestes funcionalitats responen a necessitats expressades pels entrenadors i són les següents:

- Registre i inici de sessió d'usuari.
- Visualització i gestió dels equips: afegir, editar, eliminar.
- Visualització de la plantilla de cada equip.
- Gestió de jugadors.
- Visualització d'estadístiques de jugadors.
- Visualització i gestió d'entrenaments
- Visualització i gestió de partits

Aquestes funcionalitats responen als punts destacats pels entrenadors, especialment pel que fa a centralitzar la informació, facilitar el seguiment de la plantilla i optimitzar la planificació.

5.2.2. Requisits no funcionals

Els requisits no funcionals tenen com a objectiu garantir una experiència còmoda i eficient per part dels entrenadors de futbol amateur, tenint en compte la seva realitat: limitacions de temps, coneixements tecnològics molt diversos i recursos limitats.

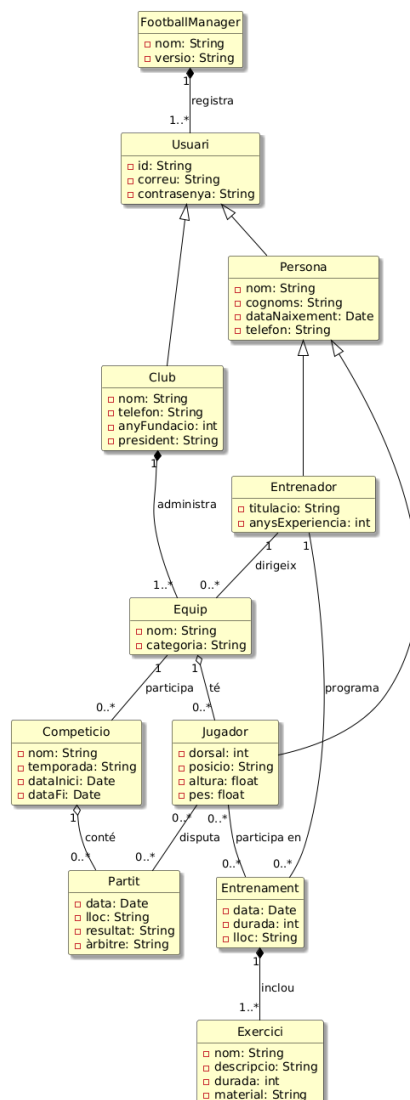
Els requisits no funcionals identificats són els següents:

- **Seguretat:** xifratge de contrasenyes i protecció de dades personals.
- **Usabilitat:** interfície intuïtiva i senzilla, adaptada a perfils no tècnics.

- **Accessibilitat:** sistema responsiu que funcioni en diferents dispositius (ordinador, tauleta, mòbil).
- **Compatibilitat:** funcionament correcte als principals navegadors web.
- **Mantenibilitat:** codi estructurat i documentat per facilitar futures ampliacions.
- **Rendiment:** capacitat per gestionar múltiples usuaris alhora sense pèrdua de rendiment.
- **Escalabilitat:** arquitectura preparada per créixer si augmenta el nombre d'usuaris.
- **Comunicació clara amb l'usuari:** missatges explícits per indicar errors o confirmacions.

5.3. Model de domini

El model de domini representa les entitats principals del sistema i les relacions que s'estableixen entre elles. Aquest model reflecteix l'estructura conceptual de l'aplicació **FootballManager**, orientada a facilitar la gestió esportiva i organitzativa de clubs de futbol.



Il·lustració 5: Model de domini de l'aplicació **FootballManager**

El model de domini parteix de l'aplicació FootballManager, que és l'encarregada de gestionar els usuaris registrats al sistema. Aquests usuaris poden representar tant una entitat esportiva, com és el cas del club, com una persona física, com els entrenadors o els jugadors. Per facilitar la gestió i diferenciació dels diferents rols dins del sistema, s'utilitza un model d'herència.

La classe base Usuari conté les dades bàsiques d'autenticació, és a dir, el correu electrònic i la contrasenya. A partir d'aquesta, es deriven dues classes: Club i Persona. La classe Club inclou informació específica com el nom, el telèfon, l'any de fundació i el nom del president. En canvi, la classe Persona recull dades personals com el nom, els cognoms, la data de naixement i el telèfon de contacte. A partir de Persona, es defineixen dos tipus concrets d'usuaris: Entrenador, amb atributs com la titulació i els anys d'experiència, i Jugador, que incorpora informació relacionada amb l'activitat esportiva com el dorsal, la posició, l'altura i el pes.

Cada club pot administrar diversos equips mitjançant una relació de composició, ja que els equips no tenen sentit fora del club que els gestiona. Els equips poden tenir múltiples jugadors i poden ser dirigits per un o més entrenadors. A més, poden participar en competicions, les quals permeten organitzar els partits segons la temporada i les dates d'inici i finalització. Aquesta relació entre equips i competicions és d'agregació, ja que durant la pretemporada els equips poden disputar partits amistosos que no formen part de cap competició oficial.

Els partits recullen informació com la data, el lloc on es disputa, l'àrbitre i el resultat final. A més, els jugadors poden disputar diferents partits al llarg de la temporada.

Pel que fa als entrenaments, són programats pels entrenadors i hi participen diversos jugadors. Cada entrenament inclou un conjunt d'exercicis concrets, amb la seva descripció, durada i material necessari, establint una composició, ja que els exercicis no tenen sentit fora del context de l'entrenament.

5.4. Casos d'ús i històries d'usuari

En aquesta secció es presenten els casos d'ús i les històries d'usuari definits a partir de l'anàlisi dels requisits funcionals del sistema. Aquestes eines permeten descriure de manera clara i estructurada les principals funcionalitats del sistema i les interaccions que els diferents usuaris poden establir amb aquest.

Cada cas d'ús està vinculat a una història d'usuari corresponent, assegurant que totes les funcionalitats desenvolupades responguin als requisits i necessitats identificades durant la fase d'anàlisi. Aquests s'organitzen en diferents blocs en funció de l'àmbit funcional al qual pertanyen.

5.4.1. Autenticació i gestió d'equips

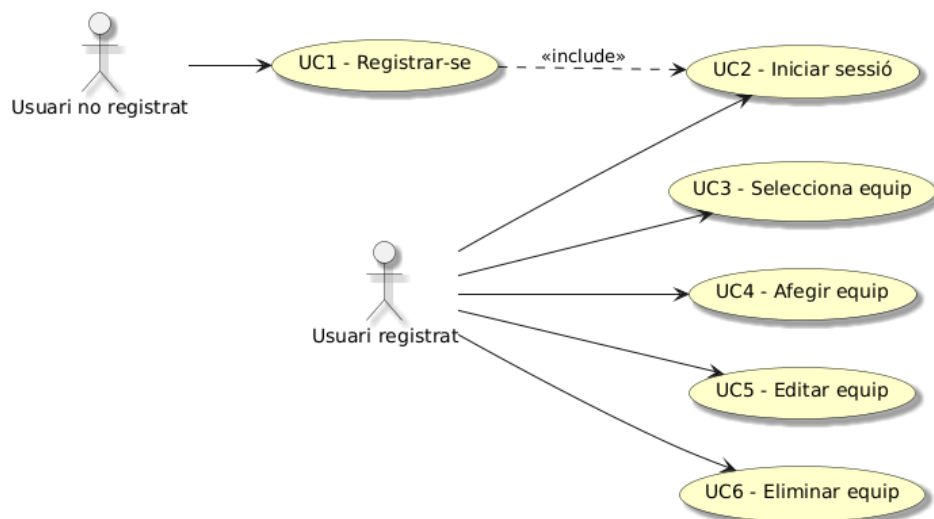
Aquest primer apartat agrupa les funcionalitats bàsiques relacionades amb l'accés al sistema i l'administració dels equips. Inclou accions com el registre, l'inici de sessió i les operacions de gestió associades als equips disponibles.

A continuació es presenta una taula que relaciona cada cas d'ús amb la seva corresponent història d'usuari:

Cas d'ús	Història d'usuari	Descripció
UC1 – Registrar-se	Com a usuari no registrat, vull poder crear un compte al sistema, per tal d'accedir a les funcionalitats reservades als usuaris autenticats.	Permet registrar un nou usuari per accedir al sistema.
UC2 – Iniciar sessió	Com a usuari registrat, vull poder iniciar sessió amb les meves credencials, per tal d'accedir al meu compte i a les funcionalitats del sistema.	Permet l'autenticació per accedir al sistema.
UC3 – Seleccionar equip	Com a usuari autenticat, vull poder veure una llista d'equips disponibles i seleccionar-ne un, per tal de consultar o gestionar-ne la informació.	Permet seleccionar un equip per veure o gestionar dades.
UC4 – Afegir equip	Com a usuari autenticat, vull poder afegir nous equips al sistema, per tal de registrar-los i gestionar-los.	Permet crear un nou equip.
UC5 – Editar equip	Com a usuari autenticat, vull poder editar la informació dels equips, per tal de mantenir les dades actualitzades i facilitar-ne la gestió.	Permet modificar dades dels equips existents.
UC6 – Eliminar equip	Com a usuari autenticat, vull poder eliminar equips del sistema, per tal de gestionar només els equips que són actuals.	Permet eliminar equips obsolets.

Taula 5: Casos d'ús relacionats amb l'autenticació d'usuaris i la gestió d'equips

També es presenta un diagrama de casos d'ús que mostra gràficament la relació entre aquestes accions i els usuaris.



Il·lustració 6: Diagrama de casos d'us corresponent als casos UC1 a UC6

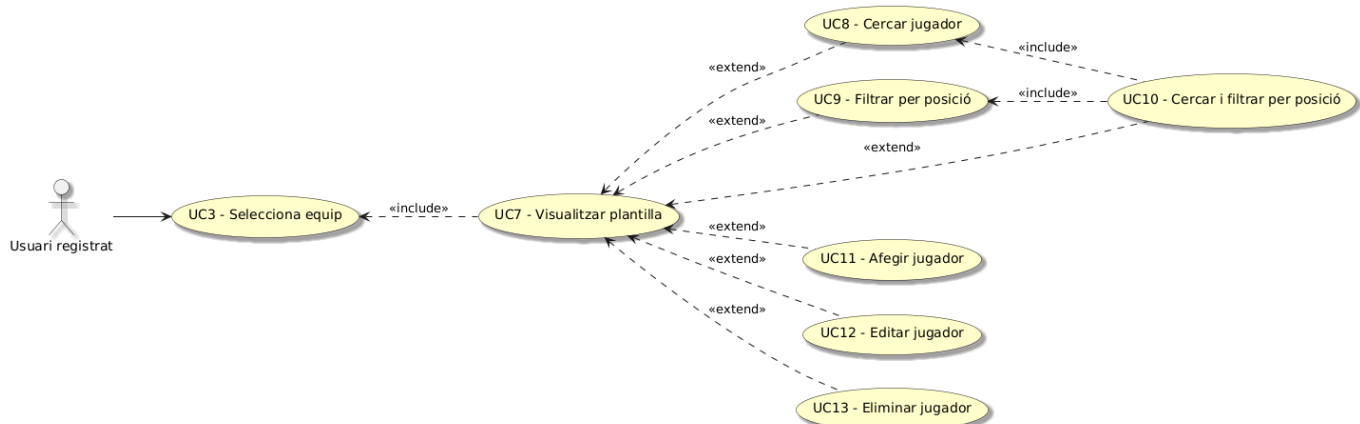
5.4.2. Gestió de jugadors

Aquest bloc mostra les accions que es poden realitzar sobre els jugadors d'un equip. Permet entendre quines accions estan disponibles per a gestionar la plantilla.

Cas d'ús	Història d'usuari	Descripció
UC7 – Visualitzar plantilla	Com a usuari autenticat, vull poder veure els jugadors de la plantilla, per tal de consultar-ne les dades i estadístiques.	Mostra la llista de jugadors d'un equip.
UC8 – Cerca de jugadors	Com a usuari autenticat, vull cercar un jugador pel seu nom o dorsal, per tal de trobar-lo ràpidament dins la plantilla.	Permet cercar jugadors per nom o dorsal.
UC9 – Filtrar per posició	Com a usuari autenticat, vull filtrar els jugadors per posició, per tal de veure només els d'una posició concreta.	Permet filtrar jugadors segons la seva posició.
UC10 – Cercar jugadors i filtrar per posició	Com a usuari autenticat, vull combinar la cerca i el filtre per posició, per tal de trobar jugadors específics dins un rol determinat.	Combina cerca i filtre per refinar la selecció.
UC11 – Afegir jugador	Com a usuari autenticat, vull poder afegir nous jugadors a la plantilla, per tal de tenir actualitzada la plantilla.	Permet registrar un nou jugador.
UC12 – Editar jugador	Com a usuari autenticat, vull poder editar la informació d'un jugador existent, per tal de corregir errors o actualitzar dades.	Permet modificar dades d'un jugador.
UC13 – Eliminar jugador	Com a usuari autenticat, vull poder eliminar jugadors del sistema, per tal de mantenir la llista actualitzada.	Permet eliminar jugadors.

Taula 6: Casos d'ús relacionats amb la gestió de jugadors

El següent diagrama mostra la relació entre usuaris i funcionalitats d'aquest bloc:



Il·lustració 7: Diagrama de casos corresponent als casos UC7 a UC13

5.4.3. Gestió de calendari, entrenaments i partits

Aquest bloc funcional cobreix les funcionalitats relacionades amb la planificació de l'activitat esportiva, des del calendari general fins als entrenaments i partits, incloent l'assistència, els exercicis i la cronologia dels esdeveniments.

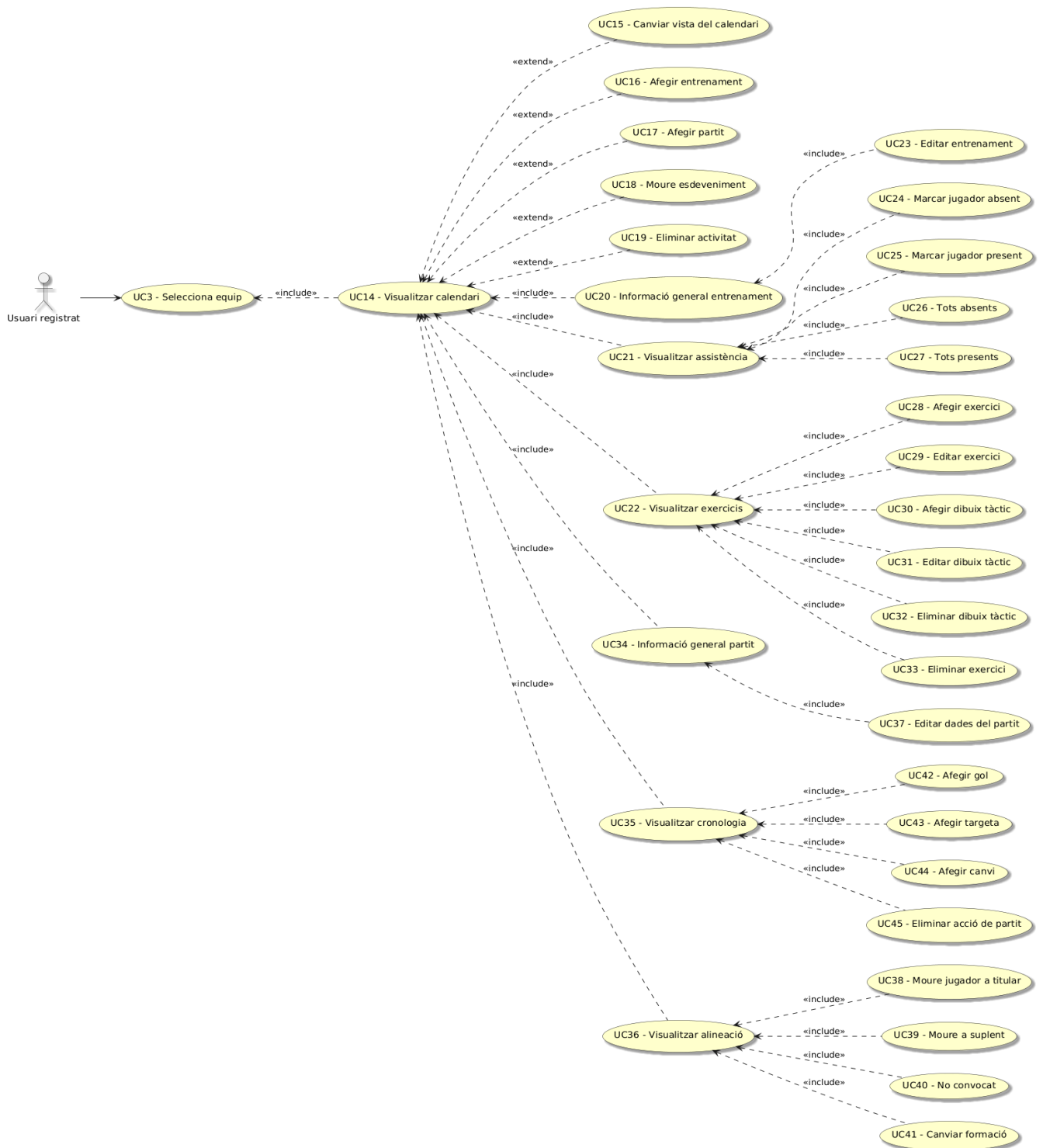
Cas d'ús	Història d'usuari	Descripció
UC14 – Visualitzar calendari	Com a usuari autenticat, vull poder veure el calendari del meu equip, per tal de planificar el meu temps i consultar entrenaments i partits.	Mostra calendari d'activitats.
UC15 – Canviar vista	Com a usuari autenticat, vull poder	Permet canviar format

del calendari	canviar la vista entre dia, setmana i mes, per tal d'adaptar la visualització.	de calendari.
UC16 – Afegir entrenament	Com a usuari autenticat, vull poder afegir un entrenament al calendari, per tal de planificar l'activitat esportiva.	Permet registrar un entrenament.
UC17 – Afegir partit	Com a usuari autenticat, vull poder afegir un partit al calendari, per tal de planificar l'activitat esportiva.	Permet registrar un partit.
UC18 – Moure esdeveniment	Com a usuari autenticat, vull poder moure un esdeveniment a una nova data o hora, per tal de reprogramar-lo.	Permet modificar la data o hora d'un esdeveniment.
UC19 – Eliminar esdeveniment del calendari	Com a usuari autenticat, vull poder eliminar un esdeveniment del calendari, per tal de gestionar-lo correctament.	Permet eliminar esdeveniments.
UC20 – Visualitzar informació general de l'entrenament	Com a usuari autenticat, vull visualitzar la informació general d'un entrenament, per tal de conèixer-ne els detalls.	Mostra informació bàsica d'un entrenament.
UC21 – Visualitzar l'assistència a l'entrenament	Com a usuari autenticat, vull veure quins jugadors han assistit a un entrenament, per tal de controlar l'assistència.	Permet consultar assistència.
UC22 – Visualitzar exercicis de l'entrenament	Com a usuari autenticat, vull veure els exercicis programats en una sessió, per tal de consultar la planificació.	Mostra la llista d'exercicis.
UC23 – Editar entrenament	Com a usuari autenticat, vull editar les dades de l'entrenament, per tal de reprogramar-lo.	Permet modificar informació d'un entrenament.
UC24 – Marcar un jugador com a absent	Com a usuari autenticat, vull poder marcar jugadors presents com a absents, per tal d'actualitzar l'assistència.	Permet marcar absències individuals.
UC25 – Marcar un jugador com a present	Com a usuari autenticat, vull poder marcar jugadors absents com a presents, per tal d'actualitzar l'assistència.	Permet marcar presències individuals.
UC26 – Marcar tots els jugadors com a absents	Com a usuari autenticat, vull poder marcar tots els jugadors presents com a absents, per tal d'actualitzar massivament l'assistència.	Permet registre massiu d'absències.
UC27 – Marcar tots els jugadors com a presents	Com a usuari autenticat, vull poder marcar tots els jugadors absents com a presents, per tal d'actualitzar massivament l'assistència.	Permet registre massiu de presències.
UC28 – Afegir exercici a l'entrenament	Com a usuari autenticat, vull poder afegir un exercici, per tal d'organitzar i adaptar els entrenaments.	Permet crear exercicis.
UC29 – Editar exercici de l'entrenament	Com a usuari autenticat, vull poder modificar un exercici, per tal d'ajustar-lo o millorar-lo.	Permet modificar exercicis.
UC30 – Afegir dibuix tàctic a l'exercici	Com a usuari autenticat, vull afegir un dibuix tàctic a un exercici, per tal de representar-lo visualment.	Permet afegir esquemes tàctics.
UC31 – Editar dibuix tàctic de l'exercici	Com a usuari autenticat, vull modificar un dibuix tàctic, per tal d'actualitzar-ne el	Permet modificar dibuixos tàctics.

	contingut.	
UC32 – Eliminar dibuix tàctic de l'exercici	Com a usuari autenticat, vull eliminar un dibuix tàctic, per tal de treure informació que ja no és útil.	Permet suprimir dibuixos tàctics.
UC33 – Eliminar exercici	Com a usuari autenticat, vull poder eliminar un exercici, per tal que l'entrenament només inclogui els exercicis que es realitzaran.	Permet eliminar exercicis.
UC34 – Visualitzar detalls del partit	Com a usuari autenticat, vull poder veure els detalls del partit, per tal de saber quan, on i el marcador.	Mostra informació completa del partit.
UC35 – Visualitzar cronologia del partit	Com a usuari autenticat, vull poder veure la cronologia del partit, per tal de saber com ha evolucionat.	Permet consultar accions del partit.
UC36 – Veure alineació del partit	Com a usuari autenticat, vull poder veure l'alineació, per tal de veure qui ha participat.	Mostra disposició de jugadors.
UC37 – Editar detalls del partit	Com a usuari autenticat, vull poder editar data, hora, ubicació o rival, per tal de corregir o actualitzar informació.	Permet modificar detalls del partit.
UC38 – Moure jugador com a titular	Com a usuari autenticat, vull poder assignar un jugador suplent o no convocat com a titular.	Permet canviar estatus de jugador.
UC39 – Moure jugador com a suplent	Com a usuari autenticat, vull poder assignar un jugador titular o no convocat com a suplent, per tal de configurar la banqueta.	Permet canviar estatus de jugador.
UC40 – Moure jugador a no convocat	Com a usuari autenticat, vull poder assignar un jugador titular o suplent com a no convocat, per tal de configurar la convocatòria.	Permet marcar jugadors com a no convocats.
UC41 – Canviar formació	Com a usuari autenticat, vull poder canviar la formació tàctica de l'equip, per tal d'adaptar l'estructura de l'alineació.	Permet modificar formació tàctica.
UC42 – Afegir gol	Com a usuari autenticat, vull poder afegir un gol a la cronologia, per tal de registrar qui i quan ha marcat.	Permet registrar gols.
UC43 – Afegir targeta	Com a usuari autenticat, vull poder afegir una targeta a la cronologia, per tal de registrar sancions.	Permet registrar targetes grogues o vermelles.
UC44 – Afegir canvi	Com a usuari autenticat, vull poder afegir un canvi a la cronologia, per tal de reflectir substitucions.	Permet registrar canvis de jugadors.
UC45 – Eliminar acció de partit	Com a usuari autenticat, vull poder eliminar un esdeveniment de la cronologia, per tal de mantenir-la actualitzada.	Permet suprimir registres erronis.

Taula 7: Casos d'ús relacionats amb la gestió del calendari, entrenaments i partits

El diagrama següent reflecteix la relació entre usuaris i funcionalitats d'aquest bloc:



Il·lustració 8: Diagrama de casos corresponent als casos UC14 a UC45

5.4.4. Descripció de cas d'ús

Cada cas d'ús esmentat anteriorment compta amb una descripció detallada que especifica els actors implicats, les precondicions, el flux bàsic d'execució, els possibles fluxos alternatius i les postcondicions. A continuació, es presenta un exemple del format utilitzat per a la descripció dels casos d'ús. La descripció detallada de tots els casos d'ús i les històries d'usuari es troba al [annex 1](#).

Descripció	UC1 - Registrar-se
Actors	Usuari
Precondicions	L'usuari no està registrat al sistema
Flux bàsic	1. L'usuari accedeix a la pàgina de registre. 2. El sistema mostra el formulari amb els camps obligatoris: nom, cognom, correu electrònic i contrasenya. 3. L'entrenador introdueix les dades requerides 4. El sistema valida en temps real i habilita el botó "Crea el teu compte" a. Que cap camp estigui buit. b. Que el correu electrònic tingui un format vàlid. c. Que la contrasenya compleixi amb els requisits de seguretat: mínim 12 caràcters, una majúscula, una minúscula, un número i un caràcter especial. 5. El sistema habilita el botó "Crea el teu compte" 6. L'usuari prem el botó "Crea el teu compte" 7. El sistema comprova que el correu electrònic no estigui registrat. 8. El sistema crea el compte i redirigeix a l'usuari a la pàgina d'inici
Flux alternatiu 1	4a. Algun dels camps està buit i el sistema mostra el missatge "El <i>camp a emplenar</i> és obligatori" 5. L'usuari ha de tornar al pas 3 del flux bàsic per introduir les dades correctament.
Flux alternatiu 2	4b. El correu electrònic no té un format vàlid i el sistema mostra el missatge "Correu electrònic no vàlid" 5. L'usuari ha de tornar al pas 3 del flux bàsic i introduir un correu electrònic vàlid.
Flux alternatiu 3	4c. La contrasenya no compleix amb els requisits de seguretat i el sistema mostra el missatge "Ha de tenir almenys 12 caràcters, una majúscula, una minúscula, un caràcter especial i un número" 5. L'usuari ha de tornar al pas 3 del flux bàsic i introduir una contrasenya vàlida
Flux alternatiu 4	7. El sistema detecta que el correu electrònic ja està registrat i mostra el missatge d'error "Ja existeix un usuari amb aquest correu" 8. L'usuari ha de tornar al pas 3 del flux bàsic i introduir un correu electrònic inexistent.
Postcondicions	L'usuari ha creat un compte i està autenticat al sistema

Taula 8: Descripció del UC1 - Registrar-se

5.4.5. Història d'usuari detallada

De la mateixa manera que els casos d'ús, les històries d'usuari també estan definides de manera detallada, descrivint el perfil de l'usuari, la funcionalitat que desitja i la raó per la qual la necessita. A més, es defineixen diversos escenaris que mostren les diferents situacions que poden sorgir durant l'ús i com ha de respondre el sistema. A continuació es presenta un exemple d'una història d'usuari amb els seus escenaris. La descripció completa de totes les històries d'usuari es pot trobar a l'[annex 2](#).

US1– Registrar-se

Com a usuari no registrat, vull poder crear un compte al sistema, per tal d'accedir a les funcionalitats reservades als usuaris autenticats

Escenari 1 – Validació de camps obligatoris

En cas que l'usuari accedeixi al formulari de registre

Quan omple parcialment els camps

El sistema mostra un missatge d'error: "El camp a emplenar és obligatori"

Escenari 2 – Format del correu electrònic

En cas que l'usuari escrigui un correu electrònic amb un format incorrecte

Mentre escriu el correu electrònic

El sistema mostra un missatge d'error: "Correu electrònic no vàlid"

Escenari 3 – Validació de contrasenya

En cas que l'usuari escrigui una contrasenya que no compleix els requisits de seguretat

Mentre escriu la contrasenya

El sistema mostra un error indicant els requisits: mínim 12 caràcters, una majúscula, una minúscula, un número i un caràcter especial

Escenari 4 – Correu ja registrat

En cas que l'usuari introdueixi un correu electrònic ja existent

Quan prem el botó "Crea el teu compte"

El sistema mostra un missatge d'error: "Ja existeix un usuari amb aquest correu"

Escenari 5 – Registre correcte

En cas que l'usuari introdueixi dades vàlides i un correu no existent

Quan prem el botó "Crea el teu compte"

El sistema crea el compte i redirigeix l'usuari a la pàgina d'inici

6. Desenvolupament

6.1. Tecnologies

Per dur a terme el desenvolupament del projecte, s'han utilitzat diverses tecnologies tant per al backend com per al frontend, així com per a la gestió de la base de dades, les proves i el desplegament al núvol. A continuació, es descriu cada tecnologia i el seu ús dins del projecte:

6.1.1. Backend

- **FastAPI:** Framework modern per crear APIs amb Python. Es caracteritza per la seva alta velocitat, facilitat d'ús i generació automàtica de documentació interactiva. S'ha escollit per la seva integració amb Python i la seva eficiència en entorns de producció [\[14\]](#).
- **Uvicorn:** És un servidor ASGI lleuger i ràpid que permet executar aplicacions FastAPI [\[35\]](#).
- **Swagger:** Eina que permet generar automàticament una interfície web per documentar i provar l'API. FastAPI ho integra de manera nativa [\[16\]](#).
- **SQLModel:** Llibreria que combina SQLAlchemy i Pydantic per definir models de dades amb validació i persistència. Permet interactuar amb bases de dades SQL utilitzant Python [\[17\]](#).
- **Alembic:** Eina per gestionar les migracions de l'esquema de la base de dades [\[18\]](#).
- **PostgreSQL:** Sistema de gestió de bases de dades relacional de codi obert. Escollit per la seva robustesa, fiabilitat i compatibilitat amb entorns de producció [\[19\]](#).
- **python-jose:** Llibreria que permet treballar amb JSON Web Tokens (JWT) [\[20\]](#), un sistema de generació de tokens per a l'autenticació i autorització d'usuaris [\[21\]](#).
- **Passlib:** Llibreria de Python per gestionar el hashing de contrasenyes. S'ha utilitzat amb l'algorisme bcrypt per garantir la seguretat [\[22\]](#).
- **OAuth2:** Protocol estàndard d'autorització per controlar l'accés segur a recursos protegits. S'ha integrat per gestionar l'inici de sessió [\[23\]](#).

6.1.2. Frontend

- **Vue.js:** Framework progressiu de JavaScript per construir interfícies d'usuari i que permet crear una pàgina web dinàmica en què el contingut es carrega i actualitza sense necessitat de recarregar tota la pàgina [\[24\]](#).
- **Vuex:** Llibreria per a la gestió de l'estat en aplicacions Vue.js. S'ha utilitzat per emmagatzemar el token d'accés i la informació de l'equip que es gestiona [\[25\]](#).
- **vuex-persistedstate:** Complement que permet persistir l'estat de Vuex entre recàrregues del navegador [\[26\]](#).

- **Vue Router:** Eina que gestiona la navegació entre pàgines en aplicacions Vue.js. Permet definir rutes i carregar components de manera dinàmica [\[27\]](#).
- **FullCalendar:** Llibreria per mostrar esdeveniments en forma de calendari. S'ha utilitzat per representar entrenaments i partits de manera visual i interactiva [\[28\]](#).
- **Bootstrap:** Framework de disseny responsiu amb components visuals i classes CSS predefinides [\[29\]](#).
- **Axios:** Llibreria per fer peticions HTTP des del frontend. S'ha utilitzat per comunicar el frontend amb el backend [\[30\]](#).
- **vue-toasted:** Plugin per mostrar notificacions emergents. Ha estat útil per mostrar els missatges d'error a l'usuari [\[31\]](#).
- **html2canvas:** Llibreria que permet capturar visualment un element HTML i convertir-lo en una imatge. S'ha fet servir per exportar els dissenys de la pissarra tàctica [\[32\]](#).
- **ESLint:** Eina d'anàlisi de codi estàtic que permet detectar errors, incoherències i males pràctiques en el codi [\[33\]](#).

6.1.3. Proves

- **pytest:** Framework de proves per Python. Utilitzat per a realitzar tests unitaris i d'integració [\[34\]](#).
- **MagicMock:** Eina per simular comportaments de funcions o mòduls durant les proves, útil per aïllar funcionalitats específiques [\[35\]](#).
- **TestClient:** Client HTTP que permet provar l'API simulant les peticions reals [\[36\]](#).
- **Playwright:** Eina de testing end-to-end que permet automatitzar navegació i interaccions en diferents navegadors [\[37\]](#).

6.1.4. Desplegament al núvol

- **Amazon S3:** Servei d'emmagatzematge d'objectes. Utilitzat per allotjar fitxers estàtics com imatges i recursos del frontend [\[13\]](#).
- **Amazon CloudFront:** Xarxa de distribució de contingut que millora el rendiment de càrrega del frontend [\[10\]](#).
- **Amazon API Gateway:** Servei per exposar APIs de forma segura i escalable. Permet integrar el backend amb altres serveis d'AWS [\[9\]](#).
- **AWS Lambda:** Servei de computació serverless que permet executar codi sense necessitat de gestionar servidors [\[11\]](#).
- **Amazon RDS:** Plataforma de bases de dades gestionades. S'ha utilitzat per allotjar la base de dades PostgreSQL del projecte amb alta disponibilitat [\[12\]](#).

6.2. Arquitectura del sistema

6.2.1. Arquitectura per capes

L'arquitectura per capes és un patró de disseny molt utilitzat, ja que permet dividir el sistema en blocs funcionals ben diferenciats [38] [39] [40]. Cada capa té una responsabilitat específica i es comunica amb les altres de manera controlada, fet que facilita la modularitat i el manteniment del projecte.

Els principis clau d'aquesta arquitectura són:

- **Separació de responsabilitats:** Cada capa té una funció concreta dins l'aplicació.
- **Encapsulació:** Les capes amaguen la seva implementació interna i només ofereixen allò necessari a la resta del sistema.
- **Abstracció:** Les capes superiors poden fer ús dels serveis de les inferiors sense conèixer-ne els detalls.
- **Acoblament dèbil:** Es busca que les capes siguin el més independents possible.

Les capes principals que formen part d'aquesta arquitectura són:

1. Capa de Presentació:

És la part del sistema amb què interactua l'usuari final. S'encarrega de mostrar la informació i de recollir les dades que introdueix l'usuari. Per tant, és important que aquesta capa sigui clara, intuïtiva i fàcil d'utilitzar. Normalment, s'implementa utilitzant tecnologies com HTML, CSS i JavaScript, i frameworks com React o Angular, segons si l'aplicació és web o mòbil.

2. Capa d'Aplicació:

Aquesta capa fa d'intermediària entre la presentació i la lògica de negoci. Controla el flux de l'aplicació, gestiona les peticions que arriben de la interfície d'usuari i s'encarrega de validar les dades abans d'enviar-les a la capa següent. També pot incloure funcionalitats com l'autenticació, la gestió de permisos o el maneig d'errors. Aquesta separació permet tenir un sistema més flexible i fàcil d'escalar.

3. Capa de Lògica de Negoci:

Aquí és on es defineixen les regles del sistema. Aquesta capa no depèn de cap tecnologia concreta, fet que la fa més fàcil de mantenir i testejar. Idealment, aquesta part del codi hauria de ser reutilitzable i independent de la presentació o de la base de dades.

4. Capa de Persistència:

Aquesta capa s'encarrega de comunicar-se amb la base de dades o qualsevol altre sistema d'emmagatzematge. És la responsable de llegir, escriure, actualitzar i eliminar dades. Per mantenir el sistema ben separat, és habitual utilitzar altres patrons de disseny, que ajuden a desacoblar la lògica de negoci dels detalls tècnics de l'accés a dades.

5. Capa de Dades:

És la capa on les dades es guarden de forma permanent. Pot estar formada per bases de dades relacionals com MySQL o PostgreSQL, bases NoSQL com MongoDB, o serveis al

núvol. Aquesta capa ha de garantir que les dades siguin consistents, segures i accessibles de manera eficient.



Il·lustració 9: Representació gràfica de les capes de l'arquitectura

Aplicar aquest model aporta diversos avantatges, especialment en projectes a mitjà i llarg termini:

- **Modularitat:** Cada capa pot ser desenvolupada i mantinguda de forma independent, fent que el projecte sigui més organitzat.
- **Reutilització:** Algunes capes poden ser reutilitzades en altres aplicacions amb mínimes modificacions.
- **Escalabilitat:** És més fàcil escalar parts del sistema segons les necessitats.
- **Mantenibilitat:** Els canvis en una capa tenen un impacte mínim en les altres, cosa que redueix el risc d'errors i la dificultat de les actualitzacions.
- **Testabilitat:** La separació clara de funcionalitats permet realitzar proves unitàries i d'integració de manera més eficient.

6.2.2. Arquitectura basada en components

L'arquitectura basada en components és un patró de disseny molt utilitzat en el desenvolupament d'interfícies d'usuari modernes [\[41\]](#). Aquesta estructura divideix l'aplicació en peces independents i reutilitzables, anomenades components, cadascun amb una funció concreta.

Els principis clau d'aquesta arquitectura són:

- **Modularitat:** El sistema es divideix en components reutilitzables que encapsulen funcionalitats específiques.
- **Encapsulació:** Cada component controla la seva pròpia lògica, estil i estructura de manera independent, i només exposa les interfícies necessàries per interactuar amb altres components.

- **Abstracció:** Els components amaguen la seva implementació interna, exposant només les interfícies necessàries.
- **Reusabilitat:** Els components es dissenyen per ser reutilitzables en diferents parts de l'aplicació o fins i tot en altres projectes, estalviant temps de desenvolupament i assegurant la consistència.
- **Separació de responsabilitats:** Cada component té una responsabilitat clara i específica, com ara la presentació, l'accés a dades o la lògica de negoci.

Adoptar una arquitectura basada en components aporta diversos beneficis:

- **Reutilització:** Els components poden ser reutilitzats en diferents parts de l'aplicació o en altres projectes, reduint la duplicació de codi i millorant la coherència.
- **Escalabilitat:** És més fàcil escalar l'aplicació afegint o modificant components sense afectar el sistema complet.
- **Mantenibilitat:** La divisió del sistema en components aïllats facilita la localització i correcció d'errors, així com la implementació de millores.
- **Desenvolupament paral·lel:** Diferents equips poden treballar simultàniament en components independents, accelerant el procés de desenvolupament.
- **Consistència i estandardització:** L'ús de components reutilitzables assegura una experiència d'usuari coherent i facilita l'estandardització del codi.
- **Facilitat de proves:** Els components poden ser provats de manera independent, simplificant el procés de proves i assegurant la qualitat del codi.

6.3. Model de dades

Per gestionar tota la informació esportiva, s'ha dissenyat una base de dades relacional. Aquesta base de dades està estructurada en diferents taules, cadascuna amb els seus atributs específics, i en relacions entre elles que assegurin la integritat i coherència de les dades.

6.3.1. Entitats

El model de dades es construeix a partir d'un conjunt d'entitats que representen els diversos elements implicats en la gestió esportiva. Cada entitat té la seva taula, on es defineixen les seves principals característiques. A continuació, es descriuen taules que conformen el model, juntament amb els seus atributs.

- **Taula Usuari (user):** Aquesta taula conté la informació dels usuaris del sistema.
 - id: Identificador únic de l'usuari.
 - email: Correu electrònic de l'usuari.
 - is_superuser: Indica si l'usuari té privilegis d'administrador.
 - full_name: Nom complet de l'usuari.
 - is_active: Indica si el compte de l'usuari està actiu.

- hashed_password: Contrasenya xifrada per garantir la seguretat.
- **Taula Equip (team):** Emmagatzema les dades dels equips esportius.
 - id: Identificador únic de l'equip.
 - nom: Nom de l'equip.
 - categoria: Categoria de competició de l'equip.
 - lletra: Subcategoria o grup dins la categoria.
 - partits_jugats: Nombre total de partits jugats per l'equip.
 - victories: Nombre de victòries aconseguides.
 - empats: Nombre d'empats.
 - derrotes: Nombre de derrotes.
 - gols_favor: Gols marcats per l'equip.
 - gols_contra: Gols rebuts per l'equip.
 - usuari_id: Clau forana que indica quin usuari gestiona l'equip.
- **Taula Jugador (player):** Registra la informació dels jugadors.
 - id: Identificador únic del jugador.
 - nom: Nom del jugador.
 - dorsal: Número que porta el jugador a la samarreta.
 - posicio: Posició del jugador en el camp.
 - team_id: Clau forana que indica a quin equip pertany.
 - partits_jugats: Total de partits jugats pel jugador.
 - minuts: Minuts totals jugats pel jugador.
 - gols: Nombre total de gols marcats pel jugador.
 - assistencies: Nombre total d'assistències realitzades.
 - targetesGrogues: Total de targetes grogues rebudes.
 - targetesVermelles: Total de targetes vermelles rebudes.
- **Taula Partit (match):** Inclou informació sobre cada partit disputat.
 - id: Identificador únic del partit.
 - start: Data i hora d'inici del partit.
 - end: Data i hora de finalització del partit.
 - location: Lloc on es juga el partit.
 - local: Booleà que indica si l'equip juga com a local.

- score_team: Gols marcats pel propi equip.
 - score_rival: Gols marcats per l'equip rival.
 - team_id: Clau forana que indica quin equip participa.
- **Taula Esdeveniment (event):** Registra els esdeveniments ocorreguts durant els partits.
 - id: Identificador únic de l'esdeveniment.
 - event_type: Tipus d'esdeveniment
 - minute: Minut en què es produeix l'esdeveniment.
 - player_name: Nom del jugador implicat.
 - match_id: Clau forana que indica el partit on passa l'esdeveniment.
 - player_id: Clau forana que indica quin jugador hi està involucrat.
- **Taula Entrenament (training):** Conté les sessions d'entrenament programades.
 - id: Identificador únic de l'entrenament.
 - start: Data i hora d'inici de la sessió d'entrenament.
 - end: Data i hora de finalització de la sessió.
 - location: Lloc on s'ha realitzat l'entrenament.
 - team_id: Clau forana que indica l'equip associat a l'entrenament.
- **Taula Exercici (exercise):** Emmagatzema exercicis individuals utilitzats durant els entrenaments.
 - id: Identificador únic de l'exercici.
 - nom: Nom de l'exercici.
 - durada: Duració en minuts de l'exercici.
 - descripcio: Explicació detallada de com es realitza.
 - image_base64: Imatge de l'exercici codificada en base64.
 - board_data: Informació tàctica gràfica de l'exercici.

6.3.2. Taules d'enllaç

El model de dades també incorpora diverses taules d'enllaç per representar les relacions de molts a molts (N:N) entre les diferents entitats. A continuació es descriuen cadascuna d'elles:

- **Jugador-Partit (matchplayerlink):** Representa la participació dels jugadors en cada partit.
 - match_id: Clau forana que identifica el partit.

- **player_id**: Clau forana que identifica el jugador.
- **is_starter**: Booleà que indica si el jugador ha estat titular.
- **minuts_jugats**: Minuts jugats pel jugador durant el partit.
- **Jugador-Entrenament (trainingplayerlink)**: Reflecteix l'assistència dels jugadors a les diferents sessions d'entrenament.
 - **player_id**: Clau forana que identifica el jugador.
 - **training_id**: Clau forana que identifica l'entrenament.
- **Entrenament-Exercici (trainingexerciselink)**: Associa cada sessió d'entrenament amb els exercicis que s'hi ha realitzat
 - **training_id**: Clau forana que identifica l'entrenament.
 - **exercise_id**: Clau forana que identifica l'exercici.

6.3.3. Relacions

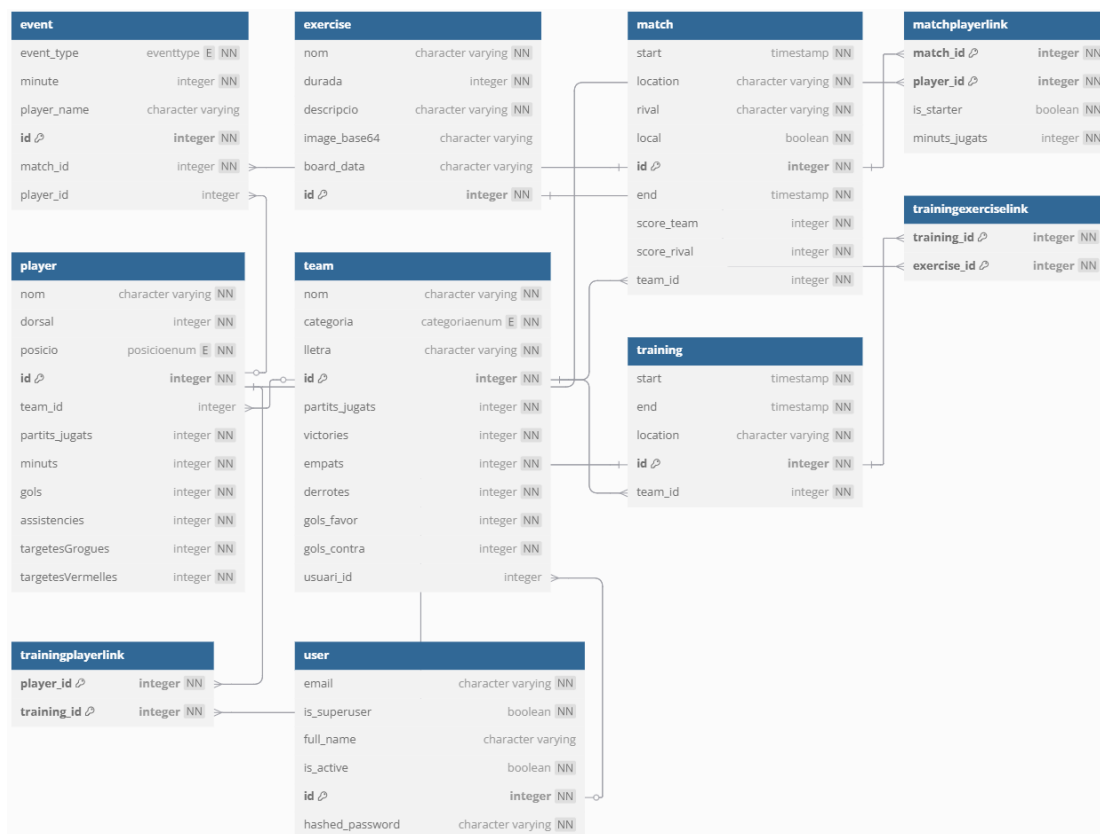
El model compta amb diferents tipus de relacions entre les entitats, tant de tipus un a molts (1:N) com de molts a molts (N:N), que es detallen a continuació:

- **Relacions de un a molts (1:N)**
 - **Usuari – Equip**: Un usuari pot gestionar diversos equips, però cada equip només pot estar assignat a un sol usuari.
 - **Equip – Jugador**: Cada equip pot tenir diversos jugadors, però cada jugador només pot formar part d'un sol equip alhora.
 - **Equip – Partit**: Un mateix equip pot disputar diferents partits, i cada partit queda vinculat a un sol equip.
 - **Equip – Entrenament**: Els equips poden tenir planificades diverses sessions d'entrenament al llarg de la temporada, però cada entrenament pertany a un únic equip.
 - **Jugador – Esdeveniment**: Un jugador pot generar múltiples esdeveniments durant els partits (com gols, targetes o assistències), però cada esdeveniment està relacionat amb un sol jugador.
 - **Partit – entrenament**: Un partit pot tenir molts esdeveniments, però cada esdeveniment només pot estar associat a un sol partit.
- **Relacions de molts a molts (N:N)**
 - **Jugador – Partit**: Un mateix jugador pot participar en diferents partits i, a la vegada, un partit pot incloure diversos jugadors. Aquesta relació es gestiona a través de la taula intermèdia *matchplayerlink*, que permet saber en quins partits ha participat cada jugador i si ha estat titular, així com els minuts jugats.
 - **Jugador – Entrenament**: Els jugadors poden assistir a diferents entrenaments, i cada entrenament pot incloure diversos jugadors. Aquesta relació es representa amb la taula *trainingplayerlink*, que enregistra quins jugadors han participat a cada sessió.

- **Entrenament – Exercici:** Cada entrenament pot incloure diversos exercicis, i un exercici pot repetir-se en més d'una sessió. Per reflectir això, s'utilitza la taula *trainingexerciselink*, que connecta entrenaments i exercicis de forma flexible.

6.3.4. Diagrama entitat-relació

Per facilitar la comprensió de l'estructura de la base de dades, es presenta el diagrama entitat-relació que inclou totes les entitats i les seves relacions.



Il·lustració 10: Estructura de la base de dades del projecte

6.4. Implementació del backend

En el desenvolupament del backend del projecte, s'ha aplicat el patró d'arquitectura per capes amb l'objectiu d'aconseguir una estructura modular, clara i fàcilment mantenible. Aquesta organització permet separar les diferents responsabilitats del sistema, afavorint la escalabilitat i la reutilització del codi.

6.4.1. Estructura del backend

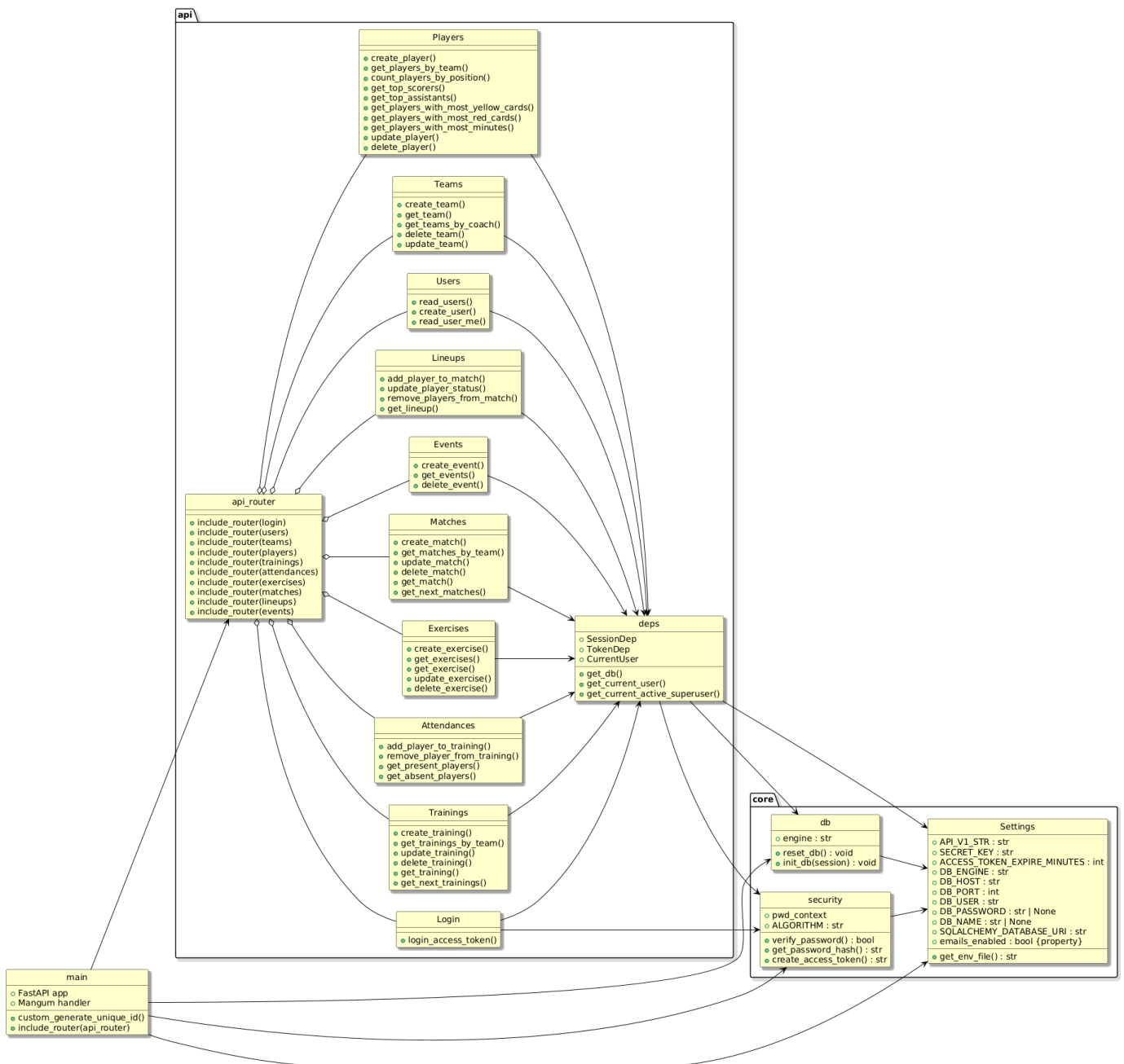
L'estructura principal del backend és la següent:

- **api/**: Conté la definició dels punts d'entrada de l'API i la lògica relacionada amb la gestió de les rutes.
 - **routes/**: Fitxers amb les rutes específiques per a diferents recursos.

- **deps.py**: Gestiona la creació de sessions de base de dades i la validació de l'autenticació i permisos dels usuaris mitjançant tokens JWT per a les rutes de l'API.
- **main.py**: Configura l'objecte FastAPI i integra les diferents rutes per estructurar l'API.
- **core/**: Agrupa la configuració i les funcionalitats comunes que s'utilitzen a tot el projecte.
 - **config.py**: Gestiona la configuració del sistema i les variables d'entorn.
 - **security.py**: Inclou funcions relacionades amb la seguretat, com la gestió de contrasenyes i tokens.
 - **db.py**: Configura la connexió a la base de dades
- **crud/**: Agrupa les operacions de creació, lectura, actualització i eliminació de les entitats del sistema.
- **models/**: Defineix els models ORM (Object Relational Mapper) que representen les entitats de la base de dades.
- **main.py**: Punt d'entrada principal del backend, on s'inicialitza l'aplicació i es configura l'entorn necessari per al funcionament del sistema.

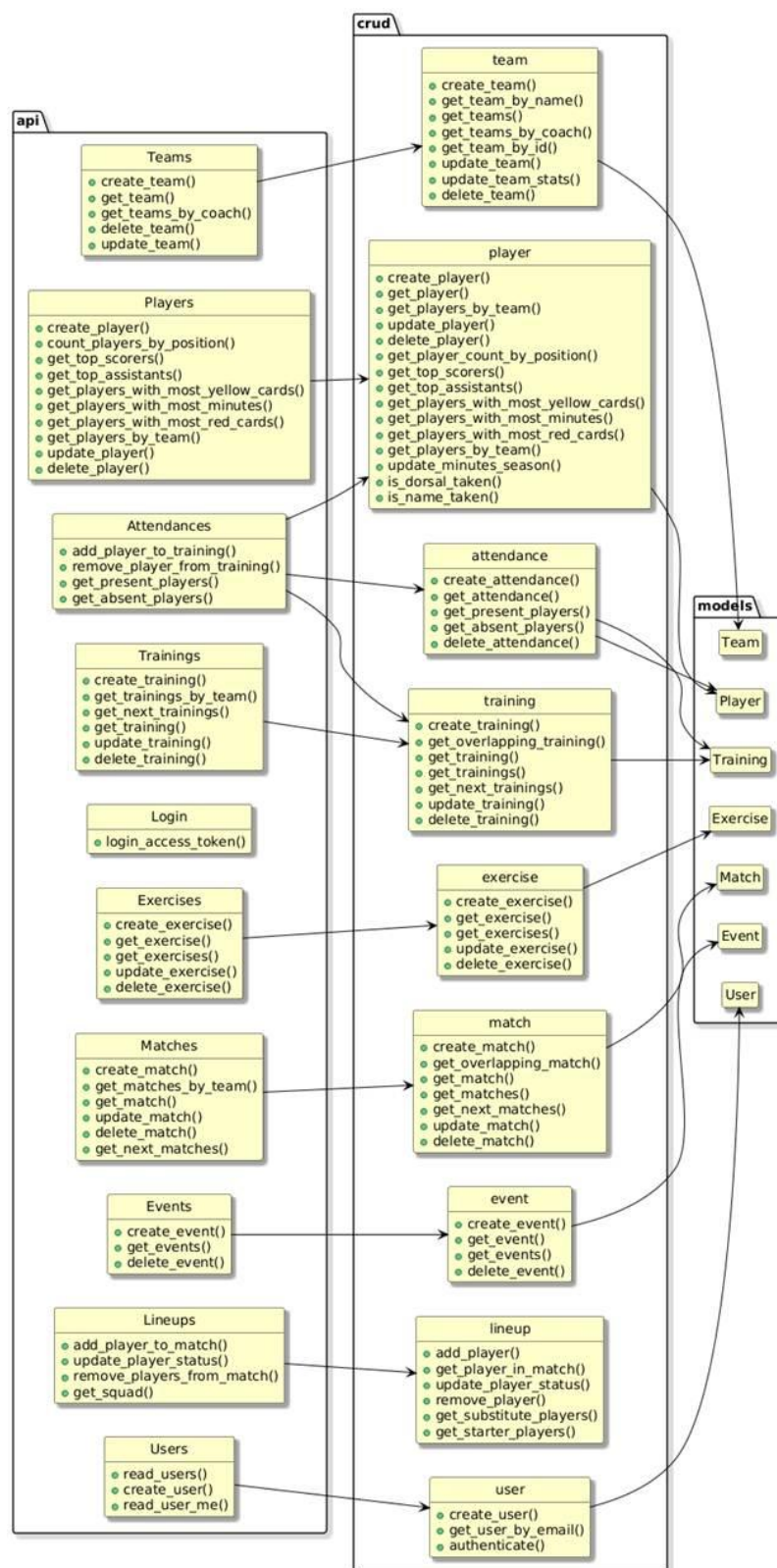
6.4.2. Diagrames d'arquitectura

A continuació es mostra un diagrama que representa com es connecta el fitxer main.py amb els mòduls api i core. Aquest fitxer actua com a punt d'entrada del backend i s'encarrega d'inicialitzar l'aplicació, carregar les rutes de l'API i configurar els components principals com la base de dades i la seguretat.



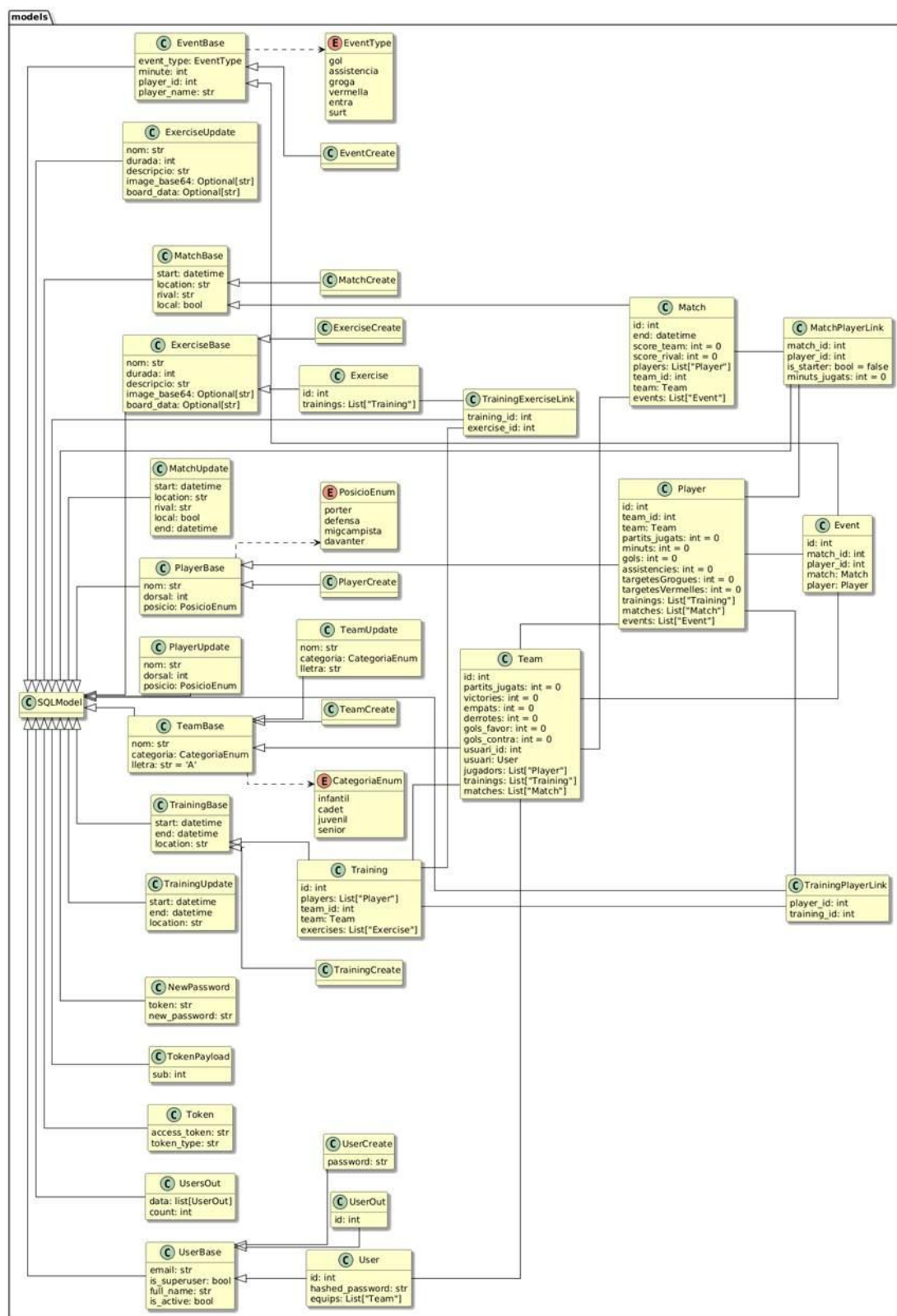
Il·lustració 11: Connexió entre main.py i els mòduls api i core

El següent diagrama representa la relació entre les rutes definides dins del mòdul api, les operacions de base de dades que es troben a crud/ i els models ORM definits a models/. Es pot observar com cada mòdul de l'API utilitza funcionalitats específiques del CRUD, les quals interactuen amb les entitats de la base de dades a través dels models.



Il·lustració 12: Relació entre les rutes, operacions CRUD i models ORM

Aquest últim diagrama mostra totes les entitats definides dins del mòdul models/, que són les que representen les taules de la base de dades i estructuren la informació del sistema. Totes aquestes classes hereten de `SQLModel`, que és la base comuna per crear models ORM en FastAPI amb `SQLModel`.



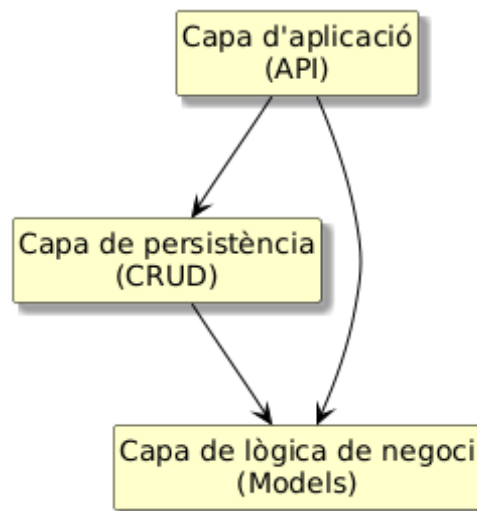
Il·lustració 13: Entitats del mòdul models/

6.4.3. Correspondència amb l'arquitectura per capes

Segons l'arquitectura descrita, es pot establir la correspondència amb l'arquitectura per capes:

- La carpeta api/ correspon a la capa d'aplicació, ja que gestiona les peticions i controla el flux entre l'usuari i el sistema, actuant com a pont entre la interfície externa i el backend.
- La carpeta models/ representa la capa de lògica de negoci, on es defineixen les entitats i regles que gestionen les dades, separant la lògica de l'emmagatzematge i la presentació.
- La carpeta crud/ constitueix la capa de persistència, encarregada de les operacions directes sobre la base de dades.

A continuació, es mostra un diagrama que il·lustra la relació entre aquestes capes:



Il·lustració 14: Representació gràfica de l'arquitectura per capes del backend

D'aquesta manera la capa d'aplicació (API) rep les peticions i connecta tant amb la capa de lògica de negoci (Models) com amb la capa de persistència (CRUD). Aquesta última és la responsable de l'accés directe a la base de dades i interactua amb els models per manipular les entitats.

6.4.4. Diagrames de seqüència de les operacions HTTP

Per mostrar el flux de comunicació entre l'usuari, l'API, la capa de persistència, els models de domini i la base de dades, es presenten a continuació els diagrames de seqüència corresponents a les quatre operacions fonamentals del protocol HTTP en un sistema RESTful: GET, POST, PUT i DELETE.

Operació GET

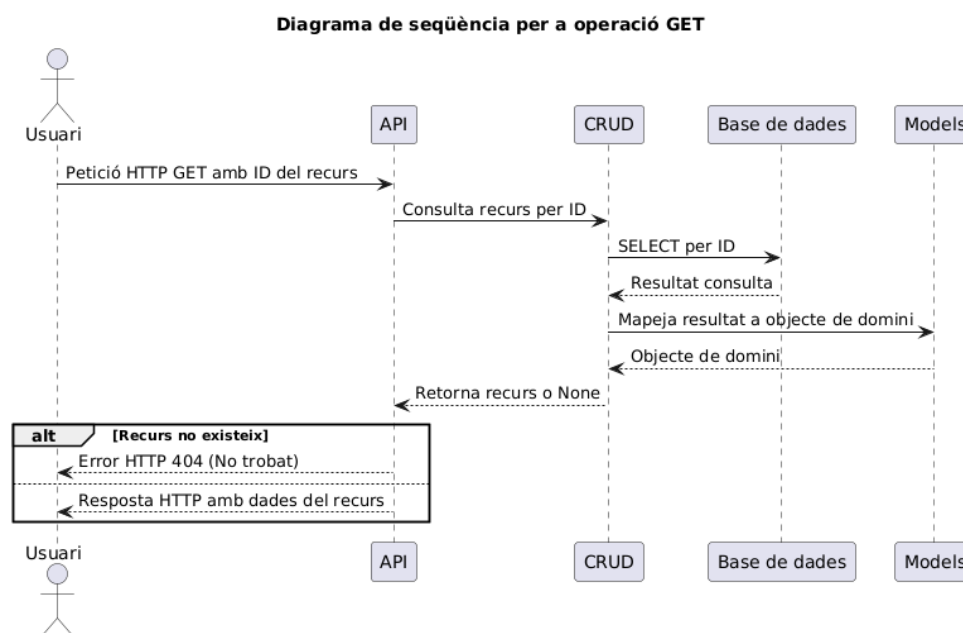
L'operació GET permet obtenir informació d'un recurs concret a partir del seu identificador únic. El procés que se segueix és el següent:

1. L'usuari fa una petició HTTP GET a l'API, indicant l'ID del recurs que vol consultar.
2. L'API rep la petició i la redirigeix a la capa de persistència (CRUD).

3. Aquesta capa executa una consulta SELECT a la base de dades per buscar el recurs corresponent.
4. Un cop obtingudes les dades, es transformen en un objecte de domini mitjançant la capa de models.
5. L'objecte es retorna a l'API, que respon a l'usuari amb un codi HTTP 200 i el recurs sol·licitat.

En cas que el recurs no existeixi:

6. La base de dades no retorna resultats i, per tant, l'API envia una resposta amb codi HTTP 404, indicant que el recurs no s'ha trobat



Il·lustració 15: Diagrama de seqüència de l'operació GET

Operació POST

L'operació POST s'utilitza per afegir un nou recurs al sistema. Abans de dur a terme la inserció, cal comprovar que les dades facilitades no entrin en conflicte amb registres ja existents. A continuació es descriu el procés pas a pas:

1. L'usuari envia una petició HTTP POST a l'API amb totes les dades necessàries per a crear un nou recurs.
2. L'API delega a la capa de persistència la tasca de verificar si ja existeixen recursos que puguin entrar en conflicte amb les dades rebudes.
3. La capa de persistència realitza una consulta SELECT amb les condicions corresponents per detectar possibles col·lisions.

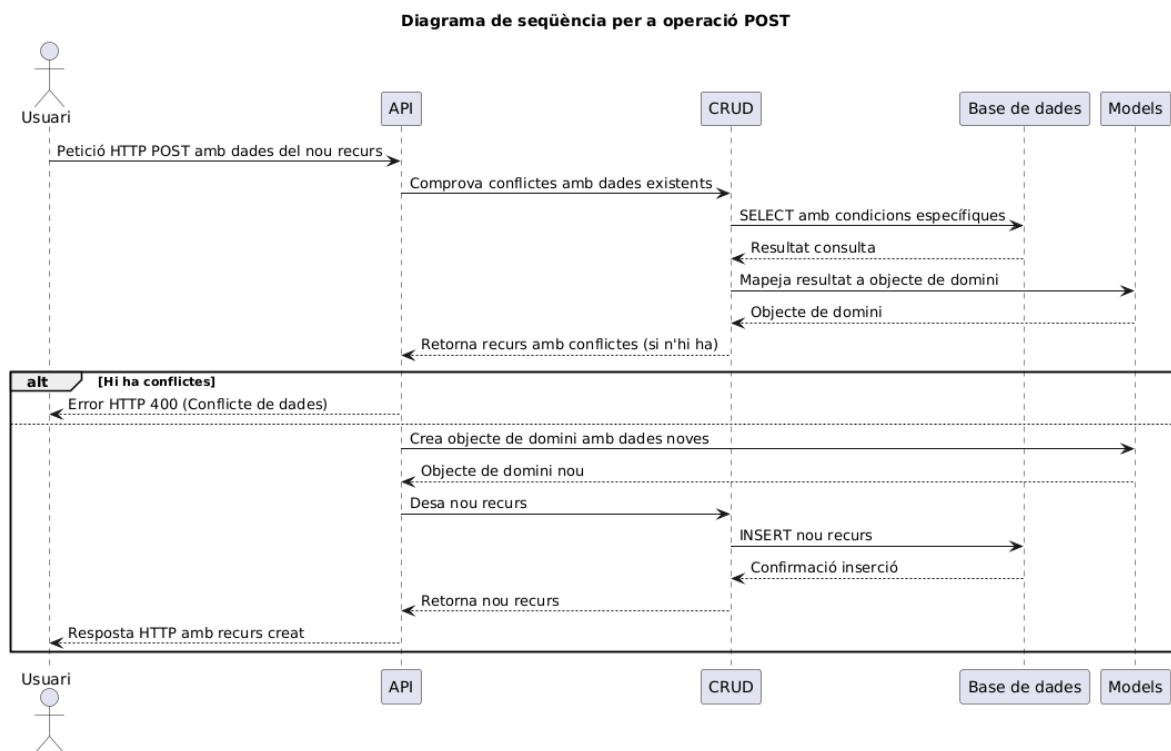
4. Els resultats de la consulta es mapegen a objectes de domini mitjançant la capa de models.
5. La capa de persistència retorna a l'API els possibles recursos amb què podria haver-hi conflicte.

En cas que es detecti un conflicte de dades:

6. L'API respon amb un codi HTTP 400, indicant que el recurs no s'ha pogut crear perquè les dades entren en conflicte amb un recurs existent.

Si no es detecta cap conflicte:

7. L'API sol·licita a la capa de persistència la creació d'un nou objecte de domini a partir de les dades de la petició.
8. La capa de persistència construeix l'objecte i executa una operació INSERT per desar-lo a la base de dades.
9. Un cop la base de dades confirma la inserció, l'objecte creat es retorna a l'API.
10. Finalment, l'API envia a l'usuari una resposta HTTP amb codi 200 i les dades del recurs creat.



Il·lustració 16: Diagrama de seqüència de l'operació POST

Operació PUT

L'operació PUT s'utilitza per modificar un recurs ja existent. Abans de fer l'actualització, és necessari comprovar que el recurs existeixi i que les noves dades no entrin en conflicte amb altres recursos del sistema.

1. L'usuari envia una petició HTTP PUT a l'API, incloent l'identificador del recurs que vol modificar i les noves dades.
2. L'API trasllada aquesta petició a la capa de persistència, que s'encarrega de verificar si el recurs existeix.
3. La capa de persistència consulta la base de dades mitjançant una operació SELECT filtrada per ID.
4. Si el recurs es troba, es fa un mapatge a objecte de domini mitjançant la capa de models.

En cas que el recurs no existeixi:

5. L'API respon amb un codi HTTP 404, indicant que el recurs no s'ha trobat.

Si el recurs existeix:

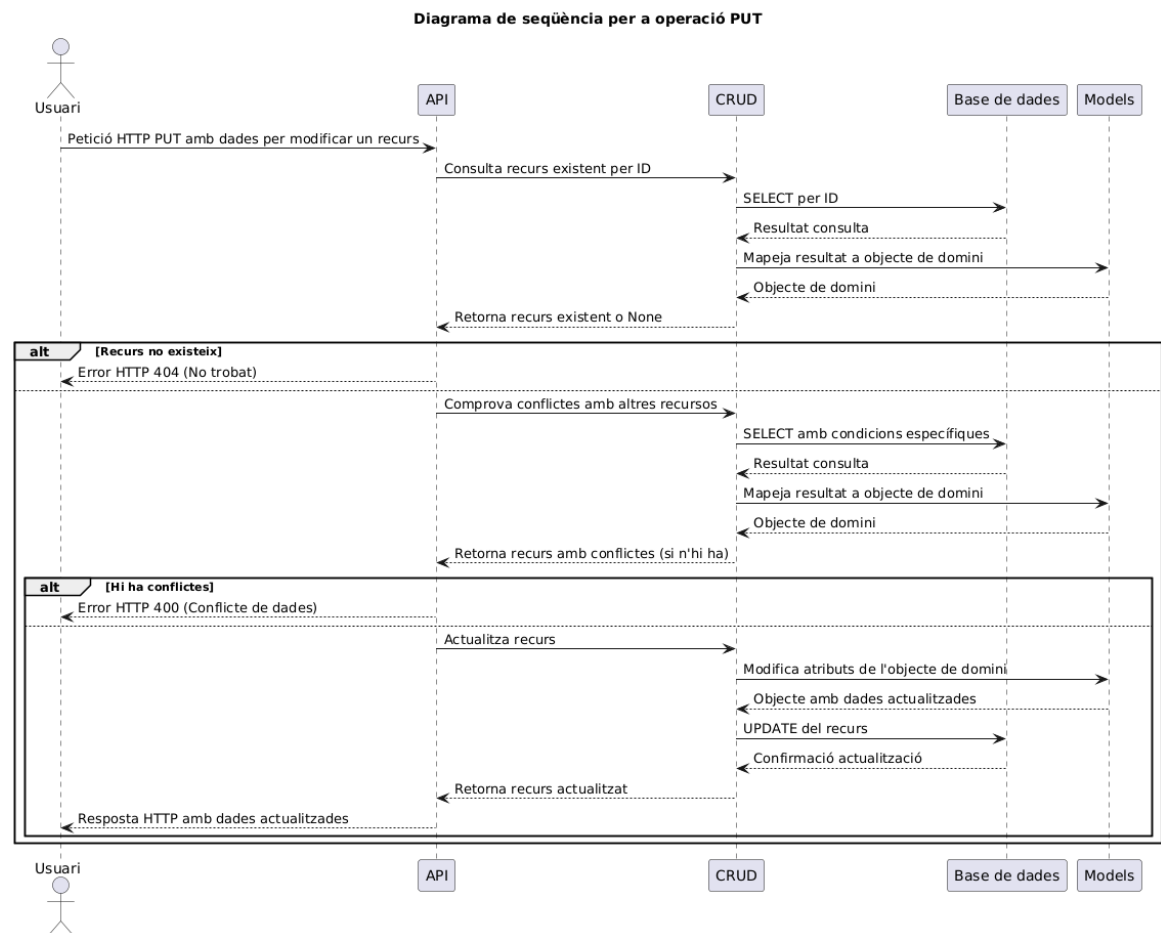
6. L'API sol·licita a la capa de persistència que comprovi si les noves dades podrien entrar en conflicte amb altres recursos del sistema (excloent el recurs actual).
7. La capa de persistència fa una nova consulta SELECT amb condicions específiques i, si cal, torna a fer el mapatge a objectes de domini.

En cas que es detecti un conflicte de dades:

8. L'API respon amb un codi HTTP 400, informant l'usuari que no s'ha pogut fer la modificació a causa del conflicte.

Si no es detecta cap conflicte:

9. L'API delega a la capa de persistència la modificació del recurs.
10. La capa de persistència actualitza l'objecte de domini amb les noves dades i executa l'operació UPDATE a la base de dades.
11. Quan la base de dades confirma l'actualització, l'API, que envia una resposta HTTP 200 a l'usuari amb les dades actualitzades.



Il·lustració 17: Diagrama de seqüència de l'operació PUT

Operació DELETE

L'operació DELETE s'utilitza per eliminar un recurs concret del sistema, a través d'un identificador únic. Abans d'executar-la, el sistema comprova que el recurs existeixi a la base de dades.

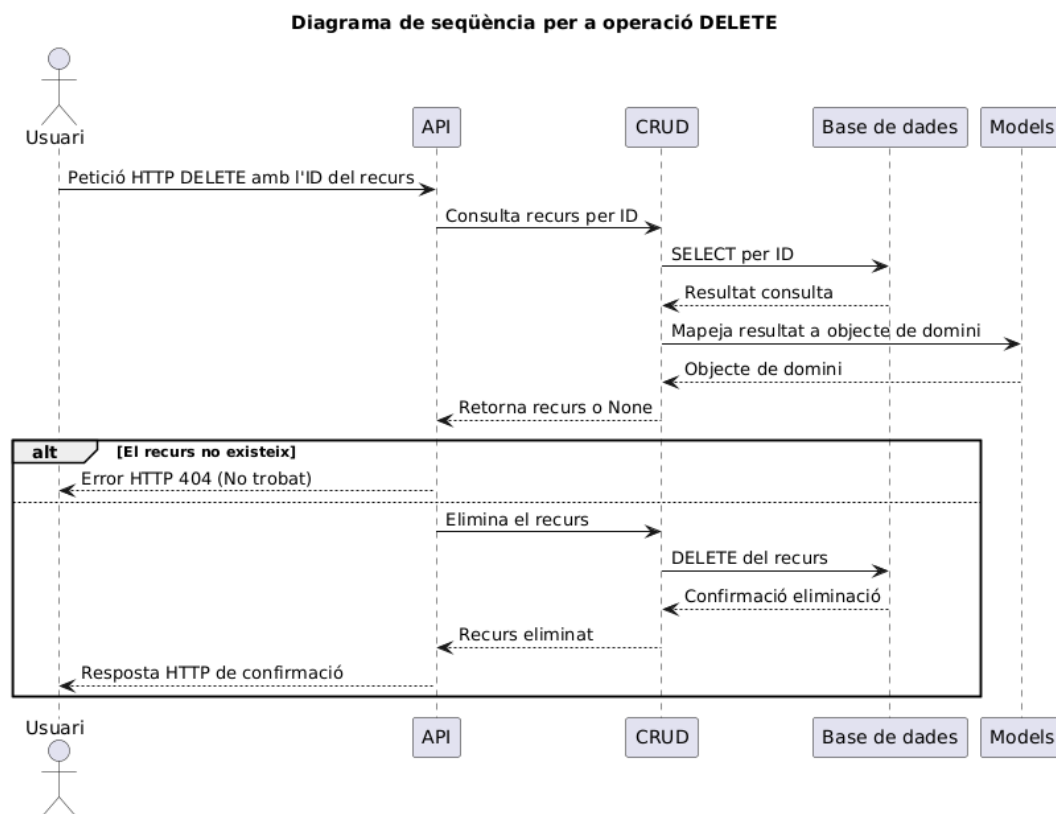
1. L'usuari envia una petició HTTP DELETE a l'API, amb l'identificador del recurs que es vol eliminar.
2. L'API trasllada aquesta petició a la capa de persistència, encarregada de comprovar l'existència del recurs.
3. La capa de persistència executa una consulta SELECT per ID a la base de dades.
4. El resultat obtingut es mapeja a un objecte de domini mitjançant la capa de models.

En cas que el recurs no existeixi:

5. L'API respon amb un codi HTTP 404, indicant que no s'ha pogut realitzar l'eliminació perquè el recurs especificat no existeix.

Si el recurs existeix:

6. L'API sol·licita a la capa de persistència que elimini el recurs.
7. La capa de persistència executa una operació DELETE a la base de dades.
8. Un cop la base de dades confirma que la supressió s'ha realitzat correctament, l'API envia a l'usuari una resposta HTTP 200, confirmant l'eliminació del recurs.



Il·lustració 18: Diagrama de seqüència de l'operació DELETE

6.5. Implementació del frontend

Per al desenvolupament del frontend s'ha utilitzat el framework Vue.js, que destaca per la seva simplicitat, flexibilitat i la seva arquitectura basada en components. Aquest enfocament permet dividir la interfície en peces independents i reutilitzables, cosa que facilita molt el manteniment i l'escalabilitat de l'aplicació.

Cada component de Vue es defineix habitualment en un fitxer amb extensió .vue, estructurat en tres seccions principals:

- **<template>**: Defineix l'estructura HTML del component, és a dir, com es mostra visualment a la pantalla.
- **<script>**: Conté la lògica del component, com ara dades, mètodes, imports d'altres components, etc.
- **<style>**: Inclou els estils CSS associats al component, que poden ser locals (només aplicats dins del component) o globals.

Aquesta separació ajuda a tenir un codi més ordenat, llegible i modular, ja que cada funcionalitat queda encapsulada en el seu propi component.

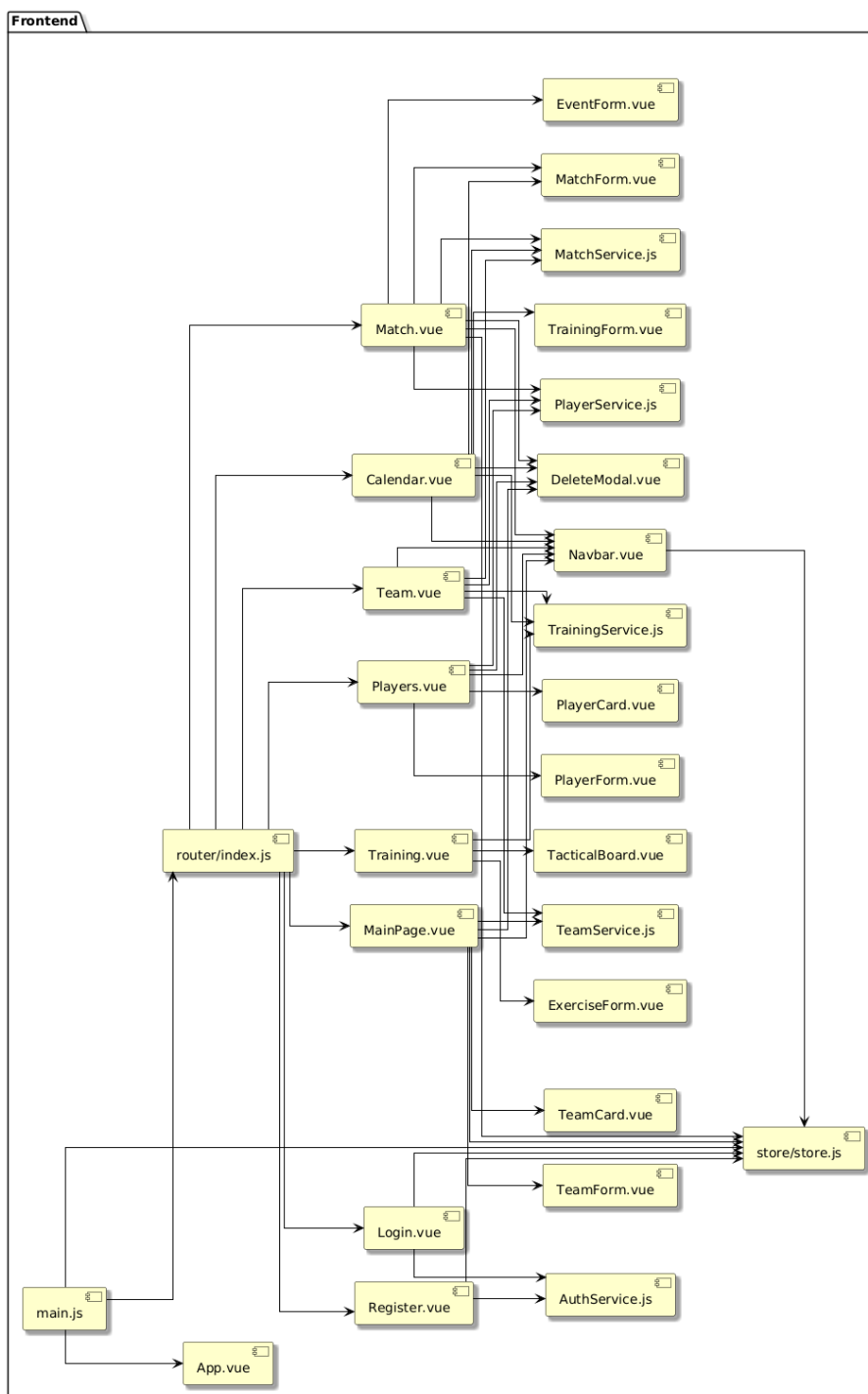
6.5.1. Estructura del frontend

El projecte està organitzat de manera que cada carpeta té una responsabilitat clara dins de l'aplicació. L'estructura principal dins la carpeta src és la següent:

- **Assets:** Arxius estàtics com imatges, icones o estils globals.
- **Components:** Components reutilitzables que es poden usar en diferents vistes.
- **Router:** Configuració del router de Vue (definició de rutes i navegació).
- **Services:** Funcions encarregades de fer crides HTTP a l'API.
- **Store:** Estat global de l'aplicació
- **Views:** Vistes o pàgines principals de l'aplicació, cadascuna associada a una ruta.
- **App.vue:** Component arrel de l'aplicació, que fa de contenidor principal
- **http-common.js:** Configuració comuna per a les peticions HTTP (com l'URL base).
- **main.js:** Punt d'entrada de l'aplicació, on es munta Vue i es carreguen els plugins.

6.5.2. Diagrama de components

Per representar de manera clara l'estructura del frontend i la relació entre les diferents parts de l'aplicació, s'ha elaborat un diagrama de components. Aquest diagrama mostra com es connecten els principals fitxers, components i serveis dins del projecte.



Il·lustració 19: Diagrama de components del frontend

El component arrel és main.js, que s'encarrega d'inicialitzar l'aplicació. Des d'aquí es carreguen els elements essencials com el component principal App.vue, el sistema de rutes (router/index.js) i la gestió de l'estat global (store/store.js).

Les vistes principals de l'aplicació són: Login.vue, Register.vue, MainPage.vue, Players.vue, Team.vue, Training.vue, Match.vue i Calendar.vue. Totes elles estan definides al sistema de rutes i representen les diferents seccions de l'aplicació.

- **Login.vue** i **Register.vue** són les pantalles d'autenticació de l'aplicació. Ambdues depenen del servei AuthService.js, que s'encarrega de gestionar les peticions

relacionades amb l'autenticació, com el login, el registre i la gestió de tokens. A més, ambdues utilitzen l'estat global gestionat per store/store.js per emmagatzemar informació de l'usuari i controlar l'accés a l'aplicació.

- **MainPage.vue** és la vista principal després de l'inici de sessió i mostra els equips que gestiona l'usuari. Integra components com TeamCard.vue, que mostra informació resumida dels equips, Navbar.vue per a la navegació, TeamForm.vue per crear o editar equips, i DeleteModal.vue per confirmar accions de supressió. Per gestionar la comunicació amb el backend, fa servir TeamService.js, i també interactua amb l'estat global del sistema a través del store.
- **Players.vue** és la vista on es gestiona tota la informació relacionada amb els jugadors. Incorpora components com PlayerCard.vue per mostrar els jugadors de manera individualitzada, PlayerForm.vue per afegir o modificar jugadors, i DeleteModal.vue per a la confirmació d'eliminacions. El servei PlayerService.js s'encarrega de les operacions relacionades amb la recuperació i actualització de dades dels jugadors.
- **Team.vue** gestiona la informació detallada d'un equip en concret. Aquesta inclou components com Navbar.vue per a la navegació consistent, i fa ús de serveis com TeamService.js, PlayerService.js, MatchService.js i TrainingService.js per mostrar estadístiques i un resum del calendari de l'equip.
- **Training.vue** està dedicada a la planificació i detalls d'un entrenament. Aquesta utilitza components especialitzats com ExerciseForm.vue per crear o editar exercicis i TacticalBoard.vue, que permet realitzar un dibuix representatiu d'un exercici. Aquesta vista s'aprofita del servei TrainingService.js per gestionar les dades associades als entrenaments.
- **Match.vue** mostra un resum dels partits i inclou components com MatchForm.vue per gestionar els detalls del partit, EventForm.vue per afegir esdeveniments dins del partit, i DeleteModal.vue per confirmar les accions. Els serveis MatchService.js i PlayerService.js són els responsables de les comunicacions amb el backend.
- **Calendar.vue** ofereix una visió de les activitats programades. Fa ús de components com TrainingForm.vue i MatchForm.vue per crear o modificar entrenaments i partits, Navbar.vue per la navegació, i DeleteModal.vue per a la confirmació d'eliminacions. Els serveis TrainingService.js i MatchService.js gestionen les dades associades als esdeveniments.

6.5.3. Navegació

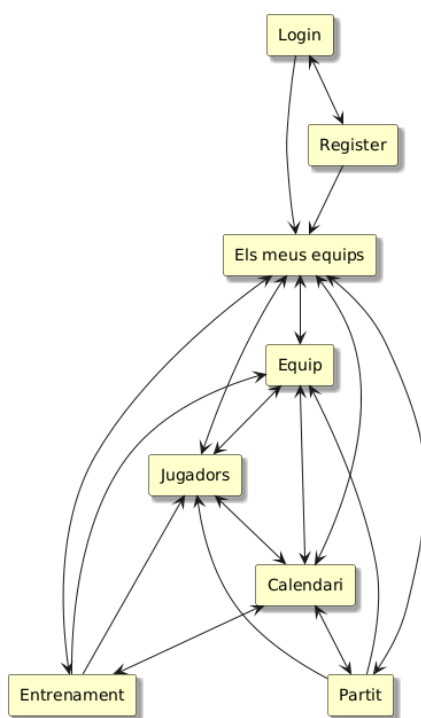
Pel que fa a la navegació, s'ha implementat un sistema de protecció de rutes que restringeix l'accés als usuaris no autenticats. D'aquesta manera, només poden accedir a les pantalles d'inici de sessió i de registre, les quals estan enllaçades entre elles per facilitar la creació d'un nou compte.

Quan l'usuari inicia sessió correctament, és redirigit a la pantalla principal (Els meus equips), que mostra els diferents equips associats al seu compte. Des d'aquesta vista, pot

seleccionar l'equip que vol gestionar. En fer-ho, la barra de navegació es modifica per mostrar tres seccions: el nom de l'equip, Jugadors i Calendari.

El nom de l'equip funciona com un enllaç per tornar a la vista principal de l'equip. A la secció de Jugadors, es pot accedir a la gestió i visualització de la plantilla, mentre que Calendari recull els entrenaments i partits de l'equip.

Si l'usuari selecciona un partit dins del calendari, es redirigeix a una pantalla amb els detalls corresponents. El mateix passa si selecciona un entrenament, accedint llavors a la informació concreta d'aquella sessió.



Il·lustració 20: Diagrama de navegació del sistema

6.6. Validació del sistema

Per tal de garantir el bon funcionament del sistema i verificar el compliment dels requisits, s'han realitzat tres tipus de proves: unitaris, d'integració i end-to-end.

6.6.1. Proves unitàries

Les proves unitàries serveixen per comprovar que les parts més petites del sistema, com ara funcions o mètodes, funcionen correctament de manera independent i sense dependre d'altres components.

En aquest projecte, s'han desenvolupat amb la llibreria **pytest** i amb **MagicMock** de **unittest.mock** per simular la base de dades. D'aquesta manera es pot comprovar la lògica interna dels mètodes sense utilitzar una base de dades real, ajudant a reduir el temps d'execució i fent els tests més fiables. S'ha implementat un test unitari per a cada mètode de la capa de persistència, cobrint les operacions bàsiques de creació, lectura, actualització i eliminació de dades. Aquests tests asseguren que les operacions amb la base de dades es realitzen correctament. Es tracta de proves white-box, ja que es fan amb accés i

coneixement del codi intern. Això permet comprovar detalladament el funcionament de cada mètode, validant que la seva lògica s'executa tal com s'esperava.

6.6.2. Proves d'integració

Les proves d'integració busquen validar que diferents components del sistema funcionen conjuntament com s'ha dissenyat. A diferència de les proves unitàries, que aïllen la lògica de cada mètode, aquestes comproven el comportament de funcionalitats completes, des de la capa d'aplicació fins a la interacció amb la base de dades.

Per fer-ho, s'ha utilitzat `pytest` amb el client de tests de FastAPI (**TestClient**), que simula peticions HTTP reals als diferents endpoints. Per evitar interferències amb dades reals, les proves fan servir una base de dades exclusiva per a testing, que es manté la mateixa estructura que la de l'aplicació, que es crea al començament de la sessió i s'elimina un cop finalitzen totes les proves.

Aquestes proves asseguren que l'API respon correctament en diferents situacions, que les dades es guarden i recuperen de manera adequada, i que les validacions funcionen correctament quan s'envien dades incorrectes. A més, es poden considerar de tipus *white-box*, ja que es desenvolupen amb coneixement de com estan definits els endpoints, la qual cosa permet validar-ne detalladament el funcionament.

6.6.3. Proves end-to-end

Les proves end-to-end serveixen per validar el funcionament complet de l'aplicació des de la perspectiva de l'usuari, comprovant que totes les capes del sistema interactuen correctament. Aquestes proves simulen la navegació real per la interfície i verifiquen que els processos complets funcionen tal com s'esperava.

Per dur a terme aquestes proves, s'ha utilitzat **Playwright**, una eina que permet automatitzar navegadors web per executar tests E2E de manera eficient. La configuració dels tests garanteix que tant el backend com el frontend s'iniciïn automàticament abans d'executar les proves, fent servir la base de dades específica per a les proves, i que es tanquin un cop finalitzades. Els tests cobreixen els diferents escenaris descrits en els criteris d'acceptació de les històries d'usuari del projecte. És a dir, s'ha implementat un test específic per a cada criteri, assegurant que es comproven totes les funcionalitats essencials.

Aquestes proves són de tipus *black-box*, ja que es dissenyen sense tenir en compte com està fet el sistema per dins. Es centren únicament a garantir que es compleixi el comportament esperat, independentment de la seva implementació interna.

7. Resultats

7.1. Cobertura de proves

Per identificar les parts del codi que s'han executat durant les proves, s'ha utilitzat `pytest-cov`. Aquesta eina genera un informe detallat que mostra el percentatge de línies de codi cobertes pels tests.

Un cop executades totes les proves, la cobertura global obtinguda ha estat del 88%, un percentatge alt que indica que s'han testejat la major part de les funcionalitats.

L'informe mostra que tots els models i gran part del mòdul CRUD tenen una cobertura del 100%, assegurant que les operacions de creació, lectura, modificació i eliminació han estat comprovades.

Pel que fa a les rutes de l'API, la cobertura és en general bona, amb valors d'entre el 75% i el 96%. Les rutes amb una cobertura menor, com ara `players.py` (75%) o `events.py` (83%), corresponen a aquelles amb una major complexitat i varietat de casos.

Dins la carpeta `core`, el fitxer `config.py` presenta un 76% de cobertura, mentre que `db.py` arriba al 56%, pel fet que inclou codi relacionat amb la configuració i la connexió a la base de dades que és més difícil de testejar.

Finalment, el mòdul `event.py` és el que mostra una cobertura més baixa, amb un 48%. Això s'explica perquè conté diversos casos condicionals i actualitzacions d'estat en funció del tipus d'esdeveniment i l'equip que l'executa, que no s'han cobert completament en els tests.

A continuació, es presenten les taules detallades per cada mòdul, amb les seves respectives línies totals, línies no cobertes i percentatge de cobertura.

7.1.1. Mòdul api

La següent taula mostra la cobertura detallada dels fitxers que formen part del mòdul API:

Fitxer	Línies totals	Línies sense cobrir	Percentatge de cobertura (%)
<code>api/deps.py</code>	34	9	74%
<code>api/main.py</code>	13	0	100%
<code>api/routes/attendances.py</code>	45	8	82%
<code>api/routes/events.py</code>	46	8	83%
<code>api/routes/exercises.py</code>	38	4	89%
<code>api/routes/lineups.py</code>	68	13	81%
<code>api/routes/login.py</code>	20	1	95%
<code>api/routes/matches.py</code>	54	6	89%
<code>api/routes/players.py</code>	63	16	75%
<code>api/routes/teams.py</code>	40	3	92%

api/routes/trainings.py	47	2	96%
-------------------------	----	---	-----

Taula 9: Cobertura de proves dels fitxers del mòdul API

7.1.2. Mòdul core

Cobertura dels fitxers relacionats amb la configuració i la base de dades:

Fitxer	Línies totals	Línies sense cobrir	Percentatge de cobertura (%)
core/config.py	98	24	76%
core/db.py	16	7	56%
core/security.py	16	0	100%

Taula 10: Cobertura de proves dels fitxers de configuració i base de dades

7.1.3. Mòdul crud

Detall de la cobertura dels fitxers que implementen la lògica CRUD:

Fitxer	Línies totals	Línies sense cobrir	Percentatge de cobertura (%)
crud/attendance.py	19	0	100%
crud/event.py	77	40	48%
crud/exercise.py	27	0	100%
crud/lineup.py	30	0	100%
crud/match.py	33	0	100%
crud/player.py	53	0	100%
crud/team.py	45	0	100%
crud/training.py	37	0	100%
crud/user.py	21	0	100%

Taula 11: Cobertura de proves dels fitxers del mòdul crud

7.1.4. Mòdul models

Cobertura dels fitxers que defineixen els models de dades:

Fitxer	Línies totals	Línies sense cobrir	Percentatge de cobertura (%)
models/base.py	1	0	100%
models/event.py	23	0	100%
models/exercise.py	20	0	100%
models/links.py	12	0	100%

models/match.py	27	0	100%
models/player.py	32	0	100%
models/team.py	33	0	100%
models/training.py	21	0	100%
models/user.py	27	0	100%

Taula 12: Cobertura de proves dels fitxers que defineixen els models de dades

7.1.5. Mòdul app

Finalment, es presenta una visió general de la cobertura per mòduls i fitxers principals:

Mòdul/Fitxer	Línies totals	Línies sense cobrir	Percentatge de cobertura (%)
api	468	70	85%
core	130	31	76%
crud	342	40	88%
models	196	0	100%
main.py	16	1	94%
Total	1196	142	88%

Taula 13: Resum global de la cobertura de proves per mòduls.

7.2. Prova d'usabilitat

Per assegurar que la plataforma fos fàcil d'utilitzar i que s'adaptés correctament a les necessitats dels entrenadors d'equips amateurs, es va fer un test d'usabilitat. Per dur-lo a terme, es va comptar amb els mateixos entrenadors que havien participat en les entrevistes durant la fase de recollida de requisits. La prova es va fer presencialment, i cada entrenador havia de completar les tasques més habituals a la plataforma mentre era observat. Per a cada tasca, els usuaris van valorar la dificultat en una escala d'1 (molt fàcil) a 5 (molt difícil). A més, un cop finalitzades les tasques, es van fer unes preguntes obertes per recollir les impressions sobre la seva experiència d'ús.

Les tasques proposades i les valoracions mitjanes obtingudes van ser les següents:

Tasca	Valoració mitjana
Registrar-se com a nou usuari	1
Crear un nou equip	1
Afegir almenys 18 jugadors a l'equip	2
Cercar un jugador pel seu nom	1
Buscar jugadors per posició	1,67
Programar un entrenament	1,33
Marcar l'assistència dels jugadors a l'entrenament	2,33
Afegir exercicis a la sessió d'entrenament	1
Programar un partit	1
Definir l'alineació del partit	1,33
Introduir esdeveniments al partit (gols, targetes, canvis)	1,33

Taula 14: Valoració mitjana de les tasques proposades durant la prova d'usabilitat

La majoria de les tasques van ser considerades fàcils o molt fàcils pels usuaris. No obstant això, algunes funcionalitats, com afegir jugadors o marcar l'assistència, van rebre puntuacions més altes, indicant que hi ha marge de millora.

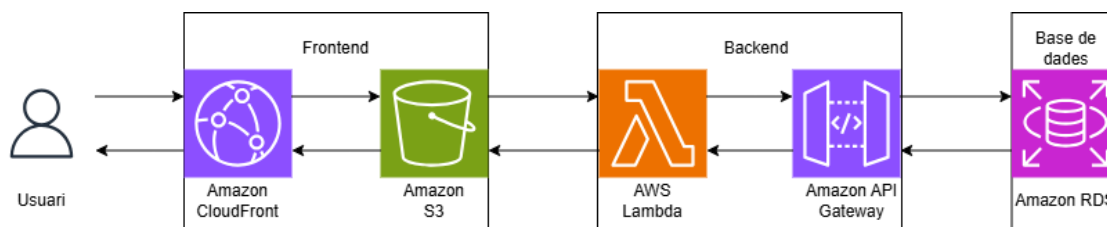
Els entrenadors van destacar que la plataforma és intuïtiva i còmoda d'utilitzar, amb noms de botons i menús molt clars i fàcils d'entendre. La funcionalitat més valorada va ser la gestió de l'alineació dels partits i la cronologia, per la seva claredat visual.

Pel que fa a les possibles millores, es va suggerir incorporar la possibilitat d'afegir jugadors de manera massiva per evitar haver de fer-ho de manera individual quan cal omplir una plantilla sencera, ja que aquest procés es va considerar lent i feixuc. També es va comentar que l'accés als detalls dels entrenaments i partits podria ser més accessible.

Finalment, tots els participants van afirmar que veuen la plataforma com una eina que farien servir de manera habitual, perquè facilita molt la gestió de l'equip i l'organització dels entrenaments i partits.

8. Desplegament

El desplegament de la plataforma s'ha realitzat utilitzant diversos serveis d'Amazon Web Services (AWS), una infraestructura al núvol àmpliament usada en entorns professionals per la seva flexibilitat, escalabilitat i fiabilitat. Compta amb una capa gratuïta que permet fer ús de determinats serveis sense cost durant els primers 12 mesos.



Il·lustració 21: Arquitectura del sistema a AWS

Per a la base de dades, s'ha utilitzat el servei **Amazon RDS** (Relational Database Service) amb el motor PostgreSQL. Concretament, s'ha fet servir una instància db.t4g.micro, inclosa dins del Free Tier, que ofereix:

- 750 hores mensuals d'instància, permetent-la mantenir activa de manera continuada.
- 20 GB d'emmagatzematge SSD.
- 20 GB addicionals per a còpies de seguretat automàtiques.

Durant la seva configuració, s'han definit paràmetres com el nom de la base de dades, les credencials de l'usuari administrador, la regió geogràfica, la zona de disponibilitat i el port 5432, predeterminat en PostgreSQL.

Un cop operativa la base de dades, s'ha desplegat el backend utilitzant **AWS Lambda** per executar el codi i **Amazon API Gateway** per exposar les funcionalitats com a API RESTful. Per tal que una aplicació en FastAPI fos compatible amb Lambda, s'ha fet servir la llibreria Mangum, que actua com a adaptador entre l'estructura ASGI de FastAPI i els esdeveniments que Lambda rep a través de l'API Gateway.

Des de la consola d'AWS s'ha creat una funció Lambda amb Python 3.12 com a entorn d'execució. S'hi ha pujat un arxiu .zip amb el codi de l'aplicació i les seves dependències, i s'ha indicat app.main.handler com a punt d'entrada. Per permetre la connexió amb la base de dades, la funció Lambda s'ha associat al mateix núvol virtual privat, compartint subxarxes i grups de seguretat amb la instància RDS. També s'hi han definit variables d'entorn amb les dades de connexió.

Per exposar el backend, s'ha creat una API REST mitjançant Amazon API Gateway, amb una ruta genèrica ANY /{proxy+} que redirigeix qualsevol petició HTTP cap a la funció Lambda. A més, s'ha configurat la política de CORS per permetre l'accés des del domini del frontend, especificant els mètodes i encapçalaments HTTP autoritzats.

Inicialment, el desplegament del frontend s'ha realitzat únicament amb **Amazon S3**, aprofitant la seva funcionalitat d'allotjament de llocs web estàtics. S'ha creat un bucket públic on s'han pujat els fitxers generats en compilar l'aplicació. També s'ha configurat perquè fes

servir l'index.html com a document principal, i s'han aplicat les polítiques necessàries per permetre'n la lectura des del navegador.

El punt d'enllaç obtingut només admetia connexions HTTP, fet que ha generat problemes amb alguns navegadors, que bloquejaven tant l'accés al lloc com les peticions cap al backend, que funcionava sobre HTTPS.

Per solucionar-ho, s'ha desplegat una distribució d'**Amazon CloudFront**, actuant com a xarxa de lliurament de contingut per sobre del bucket de S3. La distribució s'ha configurat de la següent manera:

- L'origen ha estat el bucket S3 amb accés públic habilitat.
- S'ha establert un comportament de rutes que redirigeix totes les peticions cap a index.html.
- S'ha forçat l'ús exclusiu del protocol HTTPS, redirigint automàticament qualsevol accés HTTP.
- S'ha utilitzat un certificat SSL gestionat per AWS, associat al domini per defecte *.cloudfront.net.
- S'ha aplicat la política de memòria cau Managed-CachingOptimized, que millora els temps de càrrega. A més, amb cada nou desplegament es realitzen invalidacions específiques per garantir que els canvis es reflecteixin de manera immediata.

D'aquesta manera, la plataforma queda accessible públicament a l'adreça <https://d304vcxbtmrf3.cloudfront.net> i l'API i la seva corresponent documentació està disponible a <https://sevn5xsfh1.execute-api.eu-west-3.amazonaws.com>

9. Conclusions

El projecte ha complert els objectius marcats, respectant la planificació inicial i donant lloc a una primera versió funcional de l'aplicació. Aquesta versió dona resposta a la problemàtica detectada: la manca d'eines digitals eficients per a la gestió d'equips de futbol amateur.

L'aplicació s'adapta bé a la realitat d'aquests equips i proporciona una solució útil, senzilla i accessible. L'aplicació destaca per ser fàcil d'utilitzar i disposa d'una interfície clara, pensada perquè qualsevol entrenador pugui fer-la servir sense dificultats, independentment del nivell digital que tingui. També cal esmentar que el codi s'ha estructurat de forma modular, de manera que facilitarà el manteniment i el desenvolupament de noves funcionalitats en el futur.

No obstant això, aquesta versió inicial només incorpora les funcionalitats més bàsiques que un entrenador necessita en el seu dia a dia. Per manca de temps, no s'han pogut desenvolupar algunes funcionalitats que alguns usuaris havien considerat útils i que podrien haver aportat un valor afegit a la plataforma, com ara un sistema de missatgeria interna o eines d'anàlisi tàctica dels rivals. Tampoc s'ha implementat la possibilitat d'afegir jugadors de manera massiva, una funcionalitat demanada pels usuaris durant les proves d'usabilitat.

A més, el procés de desplegament i l'execució dels testos no està automatitzat, de manera que moltes tasques s'han de fer manualment, augmentant el temps necessari per llançar noves versions i el risc d'errors.

9.1. Treball futur

Per tal de continuar millorant la plataforma, es planteja aquesta sèrie de millores i noves implementacions de cara al futur per solucionar les limitacions identificades:

- Afegir la possibilitat de carregar jugadors de manera massiva, per exemple mitjançant fitxers CSV o Excel, per agilitzar la configuració inicial dels equips.
- Crear una secció específica per a entrenaments i partits, que permeti accedir fàcilment als detalls de cada sessió o partit, en lloc de fer-ho només des del calendari.
- Automatitzar el procés de desplegament, incorporant pipelines de CI/CD per facilitar el llançament de noves versions.
- Automatitzar i ampliar la cobertura de proves, tant unitàries com d'integració, per assegurar que cada canvi en el codi sigui més fiable.
- Afegir funcionalitats per ajudar a l'anàlisi tàctica dels rivals, amb eines que recopilin informació i estadístiques que serveixin per preparar millor els partits.
- Oferir nous perfils d'usuari, com jugadors o representants del club, amb funcions adaptades per a cada rol.
- Desenvolupar un sistema de missatgeria interna que faciliti la comunicació entre l'entrenador i els jugadors.

10. Referències

- [1] SportEasy. Gestiona tu equipo de forma sencilla. [pàgina web]. [Consulta: 18 de març de 2025]. Recuperat de: <https://www.sporteasy.net/es/>
- [2] 360Player. Herramientas todo en uno para el deporte. [pàgina web]. [Consulta: 18 de març de 2025]. Recuperat de: <https://es.360player.com/>
- [3] Clupik. Software de gestión deportiva. [pàgina web]. [Consulta: 18 de març de 2025]. Recuperat de: <https://clupik.com/>
- [4] Sportlyzer. Herramienta para clubes deportivos. [pàgina web]. [Consulta: 18 de març de 2025]. Recuperat de: <https://www.sportlyzer.com/es/>
- [5] Talent.com. Salario de Full Stack Developer. [pàgina web]. [Consulta: 27 de maig de 2025]. Recuperat de: <https://es.talent.com/salary?job=Full+Stack+Development>
- [6] Talent.com. Salario de QA Tester. [pàgina web]. [Consulta: 27 de maig de 2025]. Recuperat de: <https://es.talent.com/salary?job=Qa+Test>
- [7] Talent.com. Salario de UX/UI Designer. [pàgina web]. [Consulta: 27 de maig de 2025]. Recuperat de: <https://es.talent.com/salary?job=Ux+Ui+Designer+>
- [8] MediaMarkt. Portátil ASUS Vivobook 15. [pàgina web]. [Consulta: 3 de juny de 2025]. Recuperat de: https://www.mediamaarkt.es/es/product/_portatil-asus-vivobook-15-f1504zanj1447w-156-full-hd-intelr-coretm-i7-1255u-16gb-ram-1tb-ssd-irisr-xe-windows-11-home-1579742.html
- [9] Amazon Web Services (AWS). Amazon API Gateway documentation. [pàgina web]. [Consulta: 26 de maig de 2025]. Recuperat de: <https://aws.amazon.com/api-gateway/>
- [10] Amazon Web Services (AWS). Amazon CloudFront documentation. [pàgina web]. [Consulta: 27 de maig de 2025]. Recuperat de: <https://aws.amazon.com/cloudfront/>
- [11] Amazon Web Services (AWS). Amazon Lambda documentation. [pàgina web]. [Consulta: 26 de maig de 2025]. Recuperat de: <https://aws.amazon.com/lambda/>
- [12] Amazon Web Services (AWS). Amazon RDS documentation. [pàgina web]. [Consulta: 26 de maig de 2025]. Recuperat de: <https://aws.amazon.com/rds/>
- [13] Amazon Web Services (AWS). Amazon S3 documentation. [pàgina web]. [Consulta: 21 de maig de 2025]. Recuperat de: <https://aws.amazon.com/s3/>
- [14] FastAPI. FastAPI documentation. [pàgina web]. [Consulta: 19 de març de 2025]. Recuperat de: <https://fastapi.tiangolo.com/>
- [15] Uvicorn. Uvicorn documentation. [pàgina web]. [Consulta: 19 de març de 2025]. Recuperat de: <https://www.uvicorn.org/>
- [16] Swagger. Swagger UI. [pàgina web]. [Consulta: 28 de març de 2025]. Recuperat de: <https://swagger.io/tools/swagger-ui/>

- [17] SQLAlchemy. SQLAlchemy documentation. [pàgina web]. [Consulta: 3 d'abril de 2025]. Recuperat de: <https://sqlmodel.tiangolo.com/>
- [18] Alembic. [pàgina web]. [Consulta: 21 de març de 2025]. Recuperat de: <https://alembic.sqlalchemy.org/>
- [19] PostgreSQL Global Development Group. PostgreSQL documentation. [pàgina web]. [Consulta: 2 d'abril de 2025]. Recuperat de: <https://www.postgresql.org/docs/>
- [20] JWT.io. Introduction to JSON Web Tokens. [pàgina web]. [Consulta: 26 de març de 2025]. Recuperat de: <https://jwt.io/introduction>
- [21] python-jose. python-jose GitHub repository. [pàgina web]. [Consulta: 26 de març de 2025]. Recuperat de: <https://github.com/mpdavis/python-jose>
- [22] Passlib. Passlib documentation. [pàgina web]. [Consulta: 27 de març de 2025]. Recuperat de: <https://passlib.readthedocs.io/>
- [23] OAuth.net. OAuth 2.0 authorization framework. [pàgina web]. [Consulta: 26 de març de 2025]. Recuperat de: <https://oauth.net/2/>
- [24] Vue.js. Vue.js documentation. [pàgina web]. [Consulta: 19 de març de 2025]. Recuperat de: <https://vuejs.org/>
- [25] Vuex. Vuex documentation. [pàgina web]. [Consulta: 23 d'abril de 2025]. Recuperat de: <https://vuex.vuejs.org/>
- [26] vuex-persistedstate. vuex-persistedstate GitHub repository. [pàgina web]. [Consulta: 23 d'abril de 2025]. Recuperat de: <https://github.com/robinvdvleuten/vuex-persistedstate>
- [27] Vue Router. Vue Router documentation. [pàgina web]. [Consulta: 3 d'abril de 2025]. Recuperat de: <https://router.vuejs.org/>
- [28] FullCalendar. FullCalendar documentation. [pàgina web]. [Consulta: 17 d'abril de 2025]. Recuperat de: <https://fullcalendar.io/>
- [29] Bootstrap. Bootstrap documentation. [pàgina web]. [Consulta: 28 de març de 2025]. Recuperat de: <https://getbootstrap.com/>
- [30] Axios. Axios GitHub repository. [pàgina web]. [Consulta: 18 d'abril de 2025]. Recuperat de: <https://github.com/axios/axios>
- [31] Vue Toasted. vue-toasted GitHub repository. [pàgina web]. [Consulta: 22 d'abril de 2025]. Recuperat de: <https://github.com/shakee93/vue-toasted>
- [32] html2canvas. html2canvas GitHub repository. [pàgina web]. [Consulta: 29 d'abril de 2025]. Recuperat de: <https://github.com/niklasvh/html2canvas>
- [33] ESLint. ESLint documentation. [pàgina web]. [Consulta: 18 d'abril de 2025]. Recuperat de: <https://eslint.org/>

- [34] pytest. pytest documentation. [pàgina web]. [Consulta: 12 de maig de 2025]. Recuperat de: <https://docs.pytest.org/>
- [35] Python Software Foundation. unittest.mock — mock object library. [pàgina web]. [Consulta: 10 de maig de 2025]. Recuperat de: <https://docs.python.org/3/library/unittest.mock.html>
- [36] TestClient. Testing with FastAPI. [pàgina web]. [Consulta: 12 de maig de 2025]. Recuperat de: <https://fastapi.tiangolo.com/advanced/testing/>
- [37] Playwright. Playwright documentation. [pàgina web]. [Consulta: 20 de maig de 2025]. Recuperat de: <https://playwright.dev/>
- [38] Hudge, S. Layers in software architecture. [pàgina web]. [Consulta: 21 de maig de 2025]. Recuperat de: <https://medium.com/@sagar.hudge/layers-in-software-architecture-c8cc16329ff6>
- [39] Reactive Programming. Estilos arquitectónicos: capas. [pàgina web]. [Consulta: 21 de maig de 2025]. Recuperat de: <https://reactiveprogramming.io/blog/es/estilos-arquitectonicos/capas>
- [40] Richards, M. Software architecture patterns. [pàgina web]. [Consulta: 21 de maig de 2025]. Recuperat de: <https://www.oreilly.com/library/view/software-architecture-patterns/9781491971437/ch01.html>
- [41] GeeksforGeeks. Component-based architecture in system design. [pàgina web]. [Consulta: 6 de juny de 2025]. Recuperat de: <https://www.geeksforgeeks.org/component-based-architecture-system-design/>

11. Annexos

Annex 1. Descripció dels casos d'ús

Descripció	UC1 - Registrar-se
Actors	Usuari
Precondicions	L'usuari no està registrat al sistema
Flux bàsic	1. L'usuari accedeix a la pàgina de registre. 2. El sistema mostra el formulari amb els camps obligatoris: nom, cognom, correu electrònic i contrasenya. 3. L'entrenador introdueix les dades requerides 4. El sistema valida en temps real a. Que cap camp estigui buit. b. Que el correu electrònic tingui un format vàlid. c. Que la contrasenya compleixi amb els requisits de seguretat: mínim 12 caràcters, una majúscula, una minúscula, un número i un caràcter especial. 5. El sistema habilita el botó "Crea el teu compte" 6. L'usuari prem el botó "Crea el teu compte" 7. El sistema comprova que el correu electrònic no estigui registrat. 8. El sistema crea el compte i redirigeix a l'usuari a la pàgina d'inici
Flux alternatiu 1	4. Algun dels camps està buit i el sistema mostra el missatge "El camp a emplenar és obligatori" 5. L'usuari ha de tornar al pas 3 del flux bàsic per introduir les dades correctament.
Flux alternatiu 2	4. El correu electrònic no té un format vàlid i el sistema mostra el missatge "Correu electrònic no vàlid" 5. L'usuari ha de tornar al pas 3 del flux bàsic i introduir un correu electrònic vàlid.
Flux alternatiu 3	4. La contrasenya no compleix amb els requisits de seguretat i el sistema mostra el missatge "Ha de tenir almenys 12 caràcters, una majúscula, una minúscula, un caràcter especial i un número" 5. L'usuari ha de tornar al pas 3 del flux bàsic i introduir una contrasenya vàlida
Flux alternatiu 4	7. El sistema detecta que el correu electrònic ja està registrat i mostra el missatge d'error "Ja existeix un usuari amb aquest correu" 8. L'usuari ha de tornar al pas 3 del flux bàsic i introduir un correu electrònic inexistent.
Postcondicions	L'usuari ha creat un compte i està autenticat al sistema

Taula 1.1. Descripció del cas d'ús UC1 – Registrar-se

Descripció	UC2 – Iniciar sessió
Actors	Usuari
Precondicions	- L'usuari està registrat al sistema - L'usuari disposa de les seves credencials d'accés
Flux bàsic	1. L'usuari accedeix a la pàgina d'inici de sessió. 2. El sistema mostra el formulari amb els camps obligatoris: correu electrònic i contrasenya. 3. L'usuari introdueix les credencials d'accés

	4. El sistema valida les dades d'accés: a. Que cap camp estigui buit. b. Que el correu electrònic tingui un format vàlid. 5. El sistema habilita el botó "Inicia sessió"
Flux alternatiu 1	4. Algun dels camps està buit i el sistema mostra el missatge "El camp a emplenar és obligatori"
Flux alternatiu 2	4. El correu electrònic no té un format vàlid i el sistema mostra el missatge "Correu electrònic no vàlid"
Flux alternatiu 3	7. El sistema detecta que el correu electrònic ja està registrat i mostra el missatge d'error "Correu o contrasenya incorrectes"
Postcondicions	L'usuari està autenticat al sistema

Taula 1.2. Descripció del cas d'ús UC2 – Iniciar sessió

Descripció	UC3 – Seleccionar equip
Actors	Usuari
Precondicions	- L'usuari està autenticat
Flux bàsic	1. L'usuari accedeix al sistema 2. El sistema mostra el llistat d'equips dirigits per l'usuari 3. L'usuari selecciona un equip concret de la llista. 4. El sistema redirigeix l'usuari a la pantalla de detall de l'equip seleccionat, mostrant tota la seva informació.
Flux alternatiu 1	2. Si no hi ha cap equip registrat, el sistema mostra el missatge: "No hi ha equips disponibles"
Postcondicions	L'usuari accedeix correctament al detall de l'equip seleccionat i pot gestionar-ne la informació

Taula 1.3. Descripció del cas d'ús UC3 – Seleccionar equip

Descripció	UC4 – Afegir equip
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari es troba a la pàgina principal
Flux bàsic	1. L'usuari prem el botó "Afegir equip". 2. El sistema mostra un formulari amb els camps: a. Nom de l'equip: buit b. Categoria: juvenil (per defecte) c. Lletra: A (per defecte) 3. L'usuari introdueix les dades en els camps del formulari 4. El sistema valida que tots els camps estiguin plens i habilita el botó "Afegir equip"

	5. L'usuari prem el botó "Afegir equip" 6. El sistema comprova que no hi hagi cap altre equip registrat amb el mateix nom, categoria i lletra. 7. El sistema crea l'equip i retorna l'usuari a la pantalla principal
Flux alternatiu 1	3. L'usuari deixa algun camp del formulari buit 4. El sistema manté el botó "Afegir equip" deshabilitat 5. L'usuari ha de tornar al pas 3 del flux bàsic per introduir les dades correctament.
Flux alternatiu 2	3. L'usuari prem el botó "Cancel·lar" 4. El sistema tanca el formulari i retorna a l'usuari a la pàgina principal
Flux alternatiu 3	6. El sistema detecta que ja existeix un equip amb el mateix nom, categoria i lletra i mostra el missatge d'error "L'equip ja existeix" 7. L'usuari ha de tornar al pas 3 del flux bàsic i modificar les dades
Postcondicions	L'equip ha estat creat correctament i queda registrat al sistema.

Taula 1.4. Descripció del cas d'ús UC4 – Afegir equip

Descripció	UC5 – Editar equip
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari es troba a la pàgina principal - L'usuari té almenys un equip registrat
Flux bàsic	1. L'usuari prem la icona d'editar associada a un equip 2. El sistema mostra un formulari amb el nom, la categoria i la lletra actuals de l'equip 3. L'usuari modifica els camps 4. El sistema valida que tots els camps estiguin plens i habilita el botó "Desar canvis" 5. L'usuari prem el botó "Desar canvis" 6. El sistema comprova que no existeix cap altre equip amb el mateix nom, categoria i lletra. 7. El sistema actualitza la informació de l'equip i retorna l'usuari a la pantalla principal
Flux alternatiu 1	3. L'usuari deixa algun camp del formulari buit 4. El sistema deshabilita el botó "Desar canvis" 5. L'usuari ha de tornar al pas 3 del flux bàsic per emplenar totes les dades
Flux alternatiu 2	3. L'usuari prem el botó "Cancel·lar" 4. El sistema tanca el formulari i retorna a l'usuari a la pàgina principal
Flux alternatiu 3	6. El sistema detecta que ja existeix un equip amb el mateix nom, categoria i lletra i mostra el missatge d'error "L'equip ja existeix" 7. L'usuari ha de tornar al pas 3 del flux bàsic i modificar les dades
Postcondicions	L'equip ha estat modificat correctament

Taula 1.5. Descripció del cas d'ús UC5 – Editar equip

Descripció	UC6 – Eliminar equip
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari es troba a la pàgina principal - L'usuari té almenys un equip registrat

Flux bàsic	1. L'usuari prem la icona d'eliminar del equip que vol suprimir 2. El sistema sol·licita la confirmació de l'acció: <i>Estàs segur que vols eliminar FC Barcelona?</i> 3. L'usuari prem el botó "Eliminar" 4. El sistema elimina l'equip del llistat
Flux alternatiu 1	3. L'usuari prem "Cancel·lar" 4. El sistema no realitza cap acció i retorna l'usuari a la pàgina principal
Postcondicions	L'equip ha estat eliminat correctament

Taula 1.6. Descripció del cas d'ús UC6 – Eliminar equip

Descripció	UC7 – Visualitzar plantilla
Actors	Usuari
Precondicions	- L'usuari està autènticat
Flux bàsic	1. L'usuari accedeix a la secció "Jugadors" d'un equip 2. El sistema mostra el llistat complet de jugadors amb les següents dades: a. Nom b. Posició c. Dorsal d. Gols e. Partits jugats f. Assistències g. Targetes grogues h. Targetes vermelles i. Percentatge de minuts jugats
Flux alternatiu 1	2. L'equip té jugadors registrats i el sistema mostra el missatge "L'equip no compta amb cap jugador"
Postcondicions	L'usuari pot visualitzar la plantilla completa amb les estadístiques dels jugadors

Taula 1.7. Descripció del cas d'ús UC7 – Visualitzar plantilla

Descripció	UC8 – Cercar jugador
Actors	Usuari
Precondicions	- L'usuari està autènticat - L'usuari està visualitzant la plantilla d'un equip
Flux bàsic	1. L'usuari escriu un text al cercador de jugadors 2. El sistema filtra i mostra només els jugadors que continguin aquest text al seu nom o al dorsal.
Flux alternatiu 1	1. L'usuari introdueix un text que no coincideix amb cap jugador 2. El sistema mostra el missatge "No s'han trobat jugadors amb aquests criteris de cerca"
Postcondicions	L'usuari pot consultar les dades dels jugadors que compleixen el criteri de cerca

Taula 1.8. Descripció del cas d'ús UC8 – Cercar jugador

Descripció	UC9 – Filtrar per posició
Actors	Usuari

Precondicions	- L'usuari està autenticat - L'usuari esta visualitzant la plantilla d'un equip
Flux bàsic	1. L'usuari selecciona una posició des d'un menú desplegable 2. El sistema mostra només els jugadors que tenen assignada aquella posició.
Flux alternatiu 1	1. L'usuari selecciona una posició per la qual no hi ha cap jugador registrat. 2. El sistema mostra el missatge "No s'han trobat jugadors amb aquests criteris de cerca"
Postcondicions	L'usuari pot veure els jugadors que ocupen una posició concreta

Taula 1.9. Descripció del cas d'ús UC9 – Filtrar per posició

Descripció	UC10 – Cercar i filtrar per posició
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari esta visualitzant la plantilla d'un equip
Flux bàsic	1. L'usuari introdueix un text de cerca i selecciona una posició al menú desplegable. 2. El sistema mostra només els jugadors que compleixen amb els dos criteris
Flux alternatiu 1	1. L'usuari introdueix un text de cerca i selecciona una posició al menú desplegable però cap jugador compleix ambdós criteris. 2. El sistema mostra el missatge "No s'han trobat jugadors amb aquests criteris de cerca"
Postcondicions	L'usuari pot consultar els jugadors que compleixen els criteris combinats de cerca i posició.

Taula 1.10. Descripció del cas d'ús UC10 – Cercar i filtrar per posició

Descripció	UC11 – Afegir jugador
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari esta visualitzant la plantilla
Flux bàsic	1. L'usuari prem el botó "Afegir jugador". 2. El sistema mostra un formulari amb els camps: nom dorsal i posició (amb "Defensa" preseleccionat) 3. L'usuari introdueix les dades 4. El sistema valida que tots els camps estiguin plens i habilita el botó "Afegir jugador" 5. L'usuari prem el botó "Afegir jugador". 6. El sistema valida la informació 7. El sistema crea el jugador i retorna l'usuari a la pantalla amb el llistat de jugadors
Flux alternatiu 1	3. L'usuari deixa algun camp buit 4. El sistema detecta que el formulari no és vàlid i manté el botó "Afegir jugador" deshabilitat 5. L'usuari torna al pas 3 del flux bàsic per emplenar els camps buits
Flux alternatiu 2	6. El sistema detecta un nom que ja està registrat i mostra el missatge "Ja existeix un jugador amb aquest nom a l'equip" 7. L'usuari torna al pas 3 del flux bàsic per a modificar les dades

Flux alternatiu 3	6. El sistema detecta un dorsal que ja està ocupat i mostra el missatge "Ja existeix un jugador amb aquest dorsal a l'equip" 7. L'usuari torna al pas 3 del flux bàsic per a modificar les dades
Flux alternatiu 4	3. L'usuari prem el botó "Cancel·lar". 4. El sistema tanca el formulari i no es desa cap informació.
Postcondicions	El jugador ha estat registrat al sistema

Taula 1.11. Descripció del cas d'ús UC11 – Afegir jugador

Descripció	UC12 – Editar jugador
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant la plantilla - L'equip compta amb almenys un jugador
Flux bàsic	1. L'usuari prem la icona d'editar d'un jugador 2. El sistema mostra un formulari amb les dades actuals: nom dorsal i posició 3. L'usuari modifica les dades del jugador. 4. El sistema valida que tots els camps estiguin plens i habilita el botó "Desar canvis" 5. L'usuari prem el botó "Desar canvis". 6. El sistema valida la informació 7. El sistema crea el jugador i retorna l'usuari a la pantalla amb el llistat de jugadors
Flux alternatiu 1	3. L'usuari deixa algun camp buit 4. El sistema detecta que el formulari no és vàlid i deshabilita el botó "Desar canvis" 5. L'usuari torna al pas 3 del flux bàsic per emplenar els camps buits
Flux alternatiu 2	6. El sistema detecta un nom que ja està registrat i mostra el missatge "Ja existeix un jugador amb aquest nom a l'equip" 7. L'usuari torna al pas 3 del flux bàsic per a modificar les dades
Flux alternatiu 3	6. El sistema detecta un dorsal que ja està ocupat i mostra el missatge "Ja existeix un jugador amb aquest dorsal a l'equip" 7. L'usuari torna al pas 3 del flux bàsic per a modificar les dades
Flux alternatiu 4	3. L'usuari prem el botó "Cancel·lar". 4. El sistema tanca el formulari i no es desa cap informació.
Postcondicions	El jugador ha estat modificat al sistema

Taula 1.12. Descripció del cas d'ús UC12 – Editar jugador

Descripció	UC13 – Eliminar jugador
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant la plantilla - L'equip compta amb almenys un jugador
Flux bàsic	1. L'usuari prem la icona d'eliminar del jugador que vol suprimir. 2. El sistema sol·licita la confirmació de l'acció: Estàs segur que vols eliminar Lamine Yamal? 3. L'usuari prem el botó "Eliminar" 4. El sistema elimina el jugador de la plantilla i retorna a l'usuari al llistat de jugadors

Flux alternatiu 1	3. L'usuari prem el botó "Cancel·lar" 4. El sistema manté el jugador a la plantilla i retorna a l'usuari al llistat de jugadors
Postcondicions	El jugador ha estat eliminat del sistema

Taula 1.13. Descripció del cas d'ús UC13 – Eliminar jugador

Descripció	UC14 – Visualitzar calendari
Actors	Usuari
Precondicions	- L'usuari està autenticat
Flux bàsic	1. L'usuari accedeix a la secció "Calendari" d'un equip 2. El sistema mostra el calendari en vista mensual amb els entrenaments i partits programats en les dates corresponents.
Flux alternatiu 1	2. Si no es troben entrenaments, el calendari es mostra buit
Postcondicions	L'usuari ha visualitzat els entrenaments i partits programats

Taula 1.14. Descripció del cas d'ús UC14 – Visualitzar calendari

Descripció	UC15 – Canviar vista del calendari
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip
Flux bàsic	1. L'usuari selecciona una nova vista: "Dia", "Setmana" o "Mes" 2. El sistema actualitza la visualització del calendari amb la nova vista i mostra els entrenaments i partits del calendari corresponents.
Flux alternatiu 1	1. L'usuari prem els botons "<" o ">" per avançar o retrocedir en el període corresponent a la vista activa (dia, setmana o mes). 2. El sistema actualitza la visualització mostrant els entrenaments i partits del període seleccionat.
Flux alternatiu 2	1. L'usuari prem el botó "Avui". 2. El sistema centra la visualització en el període actual segons la vista activa i mostra els entrenaments i partits corresponents.
Postcondicions	El calendari mostra correctament la vista i el període seleccionats, amb els entrenaments i partits corresponents

Taula 1.15. Descripció del cas d'ús UC15 – Canviar vista del calendari

Descripció	UC16 – Afegir entrenament
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari
Flux bàsic	1. L'usuari prem el botó "Afegir esdeveniment" 2. El sistema mostra un desplegable amb els tipus d'esdeveniment 3. L'usuari selecciona l'opció "Entrenament" 4. El sistema mostra un formulari amb els següents valors predefinits: a. Data i hora d'inici: actuals b. Hora de finalització: una hora i mitja després de l'inici c. Ubicació: "Camp Municipal" 5. L'usuari modifica les dades de l'entrenament. 6. El sistema valida que tots els camps estiguin plens, que l'hora d'inici sigui anterior a la de finalització i habilita el botó "Afegir"

	entrenament” 7. L'usuari prem el botó “Afegir entrenament”. 8. El sistema comprova que no hi hagi coincidència horària amb altres entrenaments o partits 9. El sistema crea l'entrenament i retorna l'usuari al calendari
Flux alternatiu 1	1. L'usuari prem sobre un dia entre setmana del calendari 2. El sistema obre el formulari amb la data i hora d'inici seleccionades, hora de fi predefinida i ubicació per defecte 3. L'usuari continua a partir del punt 5 del flux bàsic.
Flux alternatiu 2	5. L'usuari deixa buit un o més camps 6. El sistema detecta que hi ha camps buit i deshabilita el botó “Afegir entrenament” 7. L'usuari torna al punt 5 del flux principal per completar els camps
Flux alternatiu 3	5. L'usuari introdueix una hora de fi anterior a l'hora d'inici. 6. El sistema deshabilita el botó “Afegir entrenament” 7. L'usuari torna al punt 5 del flux bàsic per corregir l'horari
Flux alternatiu 4	8. El sistema detecta un solapament amb un entrenament programat i mostra el missatge d'error “Ja existeix un entrenament en aquest horari” 9. L'usuari torna al punt 5 del flux principal per corregir l'horari
Flux alternatiu 5	8. El sistema detecta un solapament amb un entrenament programat i mostra el missatge d'error “Ja existeix un entrenament en aquest horari” 9. L'usuari torna al punt 3 del flux principal per corregir l'horari
Flux alternatiu 6	5. L'usuari prem el botó “Cancel·lar”. 6. El sistema tanca el formulari retorna a l'usuari al calendari
Postcondicions	L'entrenament s'ha registrat al sistema

Taula 1.16. Descripció del cas d'ús UC16 – Afegir entrenament

Descripció	UC17 – Afegir partit
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari
Flux bàsic	1. L'usuari prem el botó “Afegir esdeveniment” 2. El sistema mostra un desplegable amb els tipus d'esdeveniment 3. L'usuari selecciona l'opció “Partit” 4. El sistema mostra un formulari amb els següents valors: a. Data i hora d'inici: actuals b. Hora de finalització: una hora i mitja després de l'inici c. Rival d. Ubicació: "Camp Municipal" i marcat com local 5. L'usuari modifica les dades del partit. 6. El sistema valida que tots els camps estiguin plens i habilita el botó “Afegir partit” 7. L'usuari prem el botó “Afegir partit”. 8. El sistema comprova que no hi hagi coincidència horària amb altres entrenaments o partits 9. El sistema crea el partit i retorna l'usuari al calendari
Flux alternatiu 1	1. L'usuari prem sobre un dia del cap de setmana del calendari 2. El sistema obre el formulari amb la data i hora d'inici seleccionada i

	ubicació per defecte 3. L'usuari continua a partir del punt 5 del flux bàsic.
Flux alternatiu 2	5. L'usuari deixa buit un o més camps 6. El sistema detecta que hi ha camps buit i deshabilita el botó "Afegir partit" 7. L'usuari torna al punt 5 del flux principal per completar el formulari
Flux alternatiu 3	8. El sistema detecta un encavalcament amb un entrenament programat i mostra el missatge d'error "Ja existeix un entrenament en aquest horari" 9. L'usuari torna al punt 5 del flux principal per corregir l'horari
Flux alternatiu 4	8. El sistema detecta un encavalcament amb un partit programat i mostra el missatge d'error "Ja existeix un entrenament en aquest horari" 9. L'usuari torna al punt 3 del flux principal per corregir l'horari
Flux alternatiu 5	5. L'usuari prem el botó "Cancel·lar". 6. El sistema tanca el formulari retorna a l'usuari al calendari
Postcondicions	El partit s'ha registrat al sistema

Taula 1.17. Descripció del cas d'ús UC17 – Afegir partit

Descripció	UC18 – Moure esdeveniment del calendari
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un esdeveniment
Flux bàsic	1. L'usuari arrossega un esdeveniment a una nova data i/o hora 2. El sistema comprova que no hi hagi solapament amb altres esdeveniments de l'equip i l'actualitza amb la nova data i hora
Flux alternatiu 1	2. El Sistema detecta solapament horari amb un entrenament de l'equip, mostra el missatge d'error "Ja existeix un entrenament en aquest horari" i retorna l'esdeveniment al seu horari original
Flux alternatiu 2	2. El Sistema detecta solapament horari amb un partit de l'equip, mostra el missatge d'error "Ja existeix un partit en aquest horari" i retorna l'esdeveniment al seu horari original
Postcondicions	El sistema ha actualitzat d'horari de l'esdeveniment

Taula 1.18. Descripció del cas d'ús UC18 – Moure esdeveniment del calendari

Descripció	UC 19 – Eliminar activitat del calendari
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un esdeveniment
Flux bàsic	1. L'usuari prem la icona d'eliminar corresponent al esdeveniment que vol suprimir. 2. El Sistema sol·licita confirmació de l'acció: Estàs segur que vols eliminar Partit (Diumenge, 25 Maig 00:00 - 02:00)? 3. L'usuari prem el botó "Eliminar". 4. El Sistema elimina l'entrenament de la base de dades. 5. El Sistema actualitza els esdeveniments de l'equip
Flux alternatiu	3. L'usuari prem el botó "Cancel·lar"

1	4. El Sistema manté l'entrenament al calendari i retorna a l'usuari a la pantalla del calendari
Postcondicions	El sistema ha eliminat l'esdeveniment

Taula 1.19. Descripció del cas d'ús UC19 – Moure activitat del calendari

Descripció	UC 20 – Visualitzar informació general de l'entrenament
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un entrenament
Flux bàsic	1. L'usuari selecciona un entrenament del calendari 2. El Sistema mostra el dia, l'horari, la ubicació i la duració de la sessió d'entrenament
Postcondicions	El sistema ha mostrat correctament la informació general de l'entrenament

Taula 1.20. Descripció del cas d'ús UC 20 – Visualitzar informació general de l'entrenament

Descripció	UC 21 – Visualitzar assistència a l'entrenament
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un entrenament
Flux bàsic	1. L'usuari selecciona un entrenament del calendari 2. El Sistema mostra el percentatge d'assistència i el nom i el dorsal dels jugadors que assisteixen a la sessió d'entrenament
Flux alternatiu 1	1. L'usuari selecciona un entrenament del calendari 2. El Sistema mostra el percentatge d'assistència i els jugadors que assisteixen a la sessió d'entrenament 3. L'usuari selecciona la pastilla "Absents" 4. El Sistema mostra el percentatge d'assistència i els jugadors que assisteixen a la sessió d'entrenament
Postcondicions	El sistema ha mostrat correctament la informació d'assistència

Taula 1.21. Descripció del cas d'ús UC 21 – Visualitzar assistència a l'entrenament

Descripció	UC 22 – Visualitzar exercicis de l'entrenament
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un entrenament
Flux bàsic	1. L'usuari selecciona un entrenament del calendari 2. El Sistema mostra els exercicis que es realitzen en aquesta sessió d'entrenament
Flux alternatiu 1	2. No hi ha exercicis a la sessió d'entrenament i el Sistema mostra el missatge "Encara no hi ha exercicis per aquesta sessió d'entrenament"
Postcondicions	El sistema ha mostrat correctament els exercicis a realitzar durant l'entrenament

Taula 1.22. Descripció del cas d'ús UC 22 – Visualitzar exercicis de l'entrenament

Descripció	UC23 – Editar entrenament
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari ha seleccionat un entrenament del calendari
Flux bàsic	1. L'usuari prem la icona d'editar de les dades generals de l'entrenament 2. El sistema mostra un formulari amb data, hora d'inici, de finalització i ubicació actuals de l'entrenament 3. L'usuari modifica les dades de l'entrenament. 4. El sistema valida que tots els camps estiguin plens, que l'hora d'inici sigui anterior a la de finalització i habilita el botó "Desar canvis" 5. L'usuari prem el botó "Desar canvis". 6. El sistema comprova que no hi hagi coincidència horària amb altres entrenaments o partits 7. El sistema modifica l'entrenament i retorna l'usuari als detalls de l'entrenament
Flux alternatiu 2	3. L'usuari deixa buit un o més camps 4. El sistema detecta que hi ha camps buit i deshabilita el botó "Desar canvis" 5. L'usuari torna al punt 3 del flux principal per completar els camps
Flux alternatiu 3	3. L'usuari introdueix una hora de fi anterior a l'hora d'inici. 4. El sistema deshabilita el botó "Desar canvis" 5. L'usuari torna al punt 3 del flux bàsic per corregir l'horari
Flux alternatiu 4	6. El sistema detecta un solapament amb un entrenament programat i mostra el missatge d'error "Ja existeix un entrenament en aquest horari" 7. L'usuari torna al punt 3 del flux principal per corregir l'horari
Flux alternatiu 5	6. El sistema detecta un solapament amb un partit programat i mostra el missatge d'error "Ja existeix un partit en aquest horari" 7. L'usuari torna al punt 3 del flux principal per corregir l'horari
Flux alternatiu 6	3. L'usuari prem el botó "Cancel·lar". 4. El sistema tanca el formulari retorna a l'usuari al calendari
Postcondicions	El sistema ha modificat l'esdeveniment

Taula 1.23. Descripció del cas d'ús UC23 – Editar entrenament

Descripció	UC 24 – Marcar un jugador com a absent
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant els jugadors presents a l'entrenament
Flux bàsic	1. L'usuari selecciona la icona d' absència del jugador que vol marcar com a absent 2. El sistema mou el jugador a la llista d'absents
Postcondicions	El jugador ha quedat registrat com a absent a l'entrenament

Taula 1.24. Descripció del cas d'ús UC 24 – Marcar un jugador com a absent

Descripció	UC 25 – Marcar un jugador com a present
Actors	Usuari
Precondicions	- L'usuari està autenticat

	- L'usuari està visualitzant els jugadors absents a l'entrenament
Flux bàsic	1. L'usuari selecciona la icona d' absència del jugador que vol marcar com a present 2. El sistema mou el jugador a la llista d'absents
Postcondicions	El jugador ha quedat registrat com a present a l'entrenament

Taula 1.25. Descripció del cas d'ús UC 25 – Marcar un jugador com a present

Descripció	UC 26 – Marcar tots els jugadors com a absents
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant l'assistència d'un entrenament
Flux bàsic	1. L'usuari selecciona el botó "Tots absents" 2. El sistema mou tots els jugador a la llista d'absents
Postcondicions	Tots els jugadors han quedat registrats com a absents a l'entrenament.

Taula 1.26. Descripció del cas d'ús UC 26 – Marcar tots els jugadors com a absents

Descripció	UC 27 – Marcar tots els jugadors com a presents
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant l'assistència d'un entrenament
Flux bàsic	1. L'usuari selecciona el botó "Tots presents" 2. El sistema mou tots els jugador a la llista de presents
Postcondicions	Tots els jugadors han quedat registrats com a presents a l'entrenament.

Taula 1.27. Descripció del cas d'ús UC 27 – Marcar tots els jugadors com a presents

Descripció	UC 28 – Afegir exercici a l'entrenament
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant els exercicis d'una sessió d'entrenament
Flux bàsic	1. L'usuari selecciona el botó "Afegir exercici" 2. El sistema mostra un formulari amb els camps nom, descripció i durada buits 3. L'usuari emplena els camps del formulari 4. El Sistema valida que tots els camps estiguin plens i habilita el botó "Afegir exercici" 5. L'usuari prem el botó "Afegir exercici". 6. El Sistema crea l'exercici i retorna l'usuari als detalls del entrenament
Flux alternatiu 1	3. L'usuari deixa buit un camp 4. El Sistema detecta que hi ha camps buit i deshabilita el botó "Afegir exercici" 5. L'usuari torna al punt 3 del flux principal per completar els camps obligatoris
Flux alternatiu 2	3. L'usuari prem el botó "Cancel·lar". 4. El sistema tanca el formulari i no es crea cap exercici
Postcondicions	L'exercici ha quedat registrat al sistema

Taula 1.28. Descripció del cas d'ús UC 28 – Afegir exercici a l'entrenament

Descripció	UC 29 – Editar exercici de l'entrenament
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant els exercicis d'una sessió d'entrenament
Flux bàsic	1. L'usuari selecciona la icona d'editar d'un exercici 2. El sistema mostra un formulari amb les dades actuals de l'exercici i el botó "Desar canvis" habilitat 3. L'usuari modifica els camps del formulari 4. El Sistema valida que tots els camps estiguin plens 5. L'usuari prem el botó "Desar canvis". 6. El Sistema valida la informació 7. El Sistema modifica l'exercici i retorna l'usuari als detalls del entrenament
Flux alternatiu 1	3. L'usuari deixa buit un camp 4. El Sistema detecta que hi ha camps buit i deshabilita el botó "Afegir exercici" 5. L'usuari torna al punt 3 del flux principal per completar els camps obligatoris
Flux alternatiu 2	3. L'usuari prem el botó "Cancel·lar". 4. El sistema tanca el formulari i no es modifica cap exercici
Postcondicions	El sistema ha modificat l'exercici

Taula 1.29. Descripció del cas d'ús UC 29 – Editar exercici de l'entrenament

Descripció	UC 30 – Afegir dibuix tàctic a l'exercici
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant els exercicis de l'entrenament - El calendari conté almenys un exercici
Flux bàsic	1. L'usuari prem "Afegeix dibuix tàctic" d'un exercici 2. El Sistema mostra un camp de futbol editable 3. L'usuari realitza un disseny representatiu de l'exercici 4. L'usuari prem el botó "Guardar". 5. El sistema desa el dibuix tàctic associat a l'exercici.
Flux alternatiu 1	4. L'usuari prem el botó "Cancel·lar" 5. El Sistema descarta els canvis i retorna a la llista d'exercicis
Postcondicions	El sistema ha desar el dibuix tàctic a l'exercici

Taula 1.30. Descripció del cas d'ús UC 30 – Afegir dibuix tàctic a l'exercici

Descripció	UC 31 – Editar dibuix tàctic de l'exercici
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un esdeveniment
Flux bàsic	1. L'usuari prem la icona d'editar del dibuix tàctic d'un exercici 2. El sistema mostra el dibuix tàctic existent sobre el camp de futbol 3. L'usuari modifica el dibuix tàctic. 4. L'usuari prem el botó "Guardar" 5. El Sistema desa les modificacions al dibuix tàctic

Flux alternatiu 1	4.L'usuari prem el botó "Cancel·lar" 5.El sistema descarta els canvis del dibuix tàctic i retorna al llistat d'exercicis
Postcondicions	El sistema ha actualitzat el dibuix tàctic

Taula 1.31. Descripció del cas d'ús UC 31 – Editar dibuix tàctic de l'exercici

Descripció	UC 32 – Eliminar dibuix tàctic de l'exercici
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un esdeveniment
Flux bàsic	1. L'usuari prem la icona d'eliminar del dibuix tàctic d'un exercici 2. El sistema sol·licita la confirmació de l'acció: Estàs segur que vols eliminar dibuix tàctic? 3. L'usuari prem el botó "Eliminar" 4. El sistema elimina el dibuix tàctic de l'exercici
Flux alternatiu 1	3. L'usuari prem el botó "Cancel·lar" 4. El sistema manté el dibuix tàctic i retorna a l'usuari al llistat d'exercicis
Postcondicions	El sistema ha eliminat el dibuix tàctic de l'exercici

Taula 1.32. Descripció del cas d'ús UC 32 – Eliminar dibuix tàctic de l'exercici

Descripció	UC 33 – Eliminar exercici
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un esdeveniment
Flux bàsic	1. L'usuari prem la icona d'eliminar d'un exercici 2. El sistema sol·licita la confirmació de l'acció: Estàs segur que vols eliminar Rondo? 3. L'usuari prem el botó "Eliminar" 4. El sistema elimina l'exercici
Flux alternatiu 1	3. L'usuari prem el botó "Cancel·lar" 4. El Sistema manté l'exercici i retorna a l'usuari al llistat d'exercicis
Postcondicions	El sistema ha eliminat l'exercici

Taula 1.33. Descripció del cas d'ús UC 33 – Eliminar exercici

Descripció	UC 34 – Visualitzar informació general del partit
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un partit
Flux bàsic	1. L'usuari selecciona un partit del calendari 2. El sistema mostra el dia i l'hora, la ubicació, l'equip rival i el resultat del partit, si aquest ja ha començat.
Postcondicions	El sistema ha mostrat correctament la informació general del partit seleccionat

Taula 1.34. Descripció del cas d'ús UC 34 – Visualitzar informació general del partit

Descripció	UC 35 – Visualitzar la cronologia del partit
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un entrenament
Flux bàsic	1. L'usuari selecciona un partit del calendari 2. El sistema mostra la cronologia del partit amb dos botons per afegir accions del partit
Flux alternatiu 1	2. Si no existeixen accions registrades, el sistema mostra el missatge: "No hi ha accions registrades" i els dos botons per afegir-ne.
Postcondicions	El sistema ha mostrat correctament la cronologia i la informació relacionada del partit

Taula 1.35. Descripció del cas d'ús UC 35 – Visualitzar la cronologia del partit

Descripció	UC 36 – Visualitzar l'alineació del partit
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant el calendari d'un equip - El calendari conté almenys un entrenament
Flux bàsic	1. L'usuari selecciona un partit del calendari 2. El sistema mostra la cronologia del partit 3. L'usuari prem el botó "Alineacions" 4. El sistema mostra els jugadors titulars, suplents i no convocats per a aquest partit
Postcondicions	El sistema ha mostrat correctament la informació d'alineació del partit.

Taula 1.36. Descripció del cas d'ús UC 36 – Visualitzar l'alineació del partit

Descripció	UC37 – Editar dades generals del partit
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari ha seleccionat un partit del calendari
Flux bàsic	1. L'usuari prem la icona d'editar de les dades generals del partit 2. El sistema mostra un formulari amb les dades actuals del partit 3. L'usuari modifica les dades de l'entrenament. 4. El sistema valida que tots els camps estiguin plens i habilita el botó "Desar canvis" 5. L'usuari prem el botó "Desar canvis". 6. El sistema comprova que no hi hagi coincidència horària amb altres entrenaments o partits 7. El sistema modifica l'entrenament i retorna l'usuari als detalls de l'entrenament
Flux alternatiu 2	3. L'usuari deixa buit un o més camps 4. El sistema detecta que hi ha camps buit i deshabilita el botó "Desar canvis" 5. L'usuari torna al punt 3 del flux principal per completar els camps
Flux alternatiu 3	6. El sistema detecta un solapament amb un entrenament programat i mostra el missatge d'error "Ja existeix un entrenament en aquest horari"

	7. L'usuari torna al punt 3 del flux principal per corregir l'horari
Flux alternatiu 4	6. El sistema detecta un solapament amb un partit programat i mostra el missatge d'error "Ja existeix un partit en aquest horari" 7. L'usuari torna al punt 3 del flux principal per corregir l'horari
Flux alternatiu 5	3. L'usuari prem el botó "Cancel·lar". 4. El sistema tanca el formulari retorna a l'usuari als detalls del partit
Postcondicions	El sistema ha modificat la informació general del partit

Taula 1.37. Descripció del cas d'ús UC37 – Editar dades generals del partit

Descripció	UC 38 – Moure jugador com a titular
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant l'alineació d'un partit
Flux bàsic	1. L'usuari arrossega un jugador a una posició buida de l'onze titular. 2. El sistema assigna el jugador a aquesta posició com a titular. 3. El jugador desapareix de la seva categoria anterior (suplents o no convocats). 4. El sistema mostra l'alineació actualitzada.
Flux alternatiu 1	1. L'usuari arrossega un jugador a una posició del onze titular ocupada per un altre jugador. 2. El sistema assigna el nou jugador a la posició com a titular. 3. El jugador que ocupava inicialment la posició és reassignat al rol del jugador entrant. 4. El sistema mostra l'alineació actualitzada amb el canvi de posicions.
Postcondicions	El jugador queda assignat correctament com a titular en una posició del camp.

Taula 1.38. Descripció del cas d'ús UC38 – Moure jugador com a titular

Descripció	UC 39 – Moure jugador com a suplent
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant l'alineació d'un partit
Flux bàsic	1. L'usuari arrossega un jugador a una posició buida de la banqueta 2. El sistema assigna el jugador a com a suplent. 3. 3. El jugador desapareix de la seva categoria anterior (titular o no convocats). 4. 4.El sistema mostra l'alineació actualitzada.
Flux alternatiu 1	1. L'usuari arrossega un jugador a una posició de la banqueta ocupada per un altre jugador. 2. El sistema assigna el nou jugador com a suplent 3. El jugador que ocupava inicialment la posició és reassignat al rol del jugador entrant 4. El sistema mostra l'alineació actualitzada amb el canvi de posicions
Postcondicions	El jugador queda assignat correctament com a suplent en una posició de la banqueta.

Taula 1.39. Descripció del cas d'ús UC39 – Moure jugador com a suplent

Descripció	UC 40 – Moure jugador com a no convocat
------------	---

Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant l'alineació del partit
Flux bàsic	1. L'usuari arrossega un jugador titular o suplent al llistat de jugadors no convocats 2. El sistema assigna el jugador a no convocat. 3. La posició que ocupava queda buida 4. El sistema mostra l'alineació actualitzada.
Postcondicions	El jugador queda assignat correctament com a no convocat

Taula 1.40. Descripció del cas d'ús UC40 – Moure jugador com a no convocat

Descripció	UC 41 – Canviar de formació
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant l'alineació d'un partit
Flux bàsic	1. L'usuari selecciona una nova formació 2. El sistema actualitza la disposició tàctica al camp
Flux alternatiu	2. Si ja hi ha jugadors alineats, el sistema els reubica en posicions el més similars possible a la formació anterior.
Postcondicions	El camp mostra la nova formació actualitzada amb els jugadors reubicats

Taula 1.41. Descripció del cas d'ús UC41 – Canviar de formació

Descripció	UC 42 – Afegir gol
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant la cronologia d'un partit
Flux bàsic	1. L'usuari selecciona la icona d'afegir de l'equip que ha marcat 2. El sistema obre un formulari amb "Gol" com a tipus d'acció 3. L'usuari introdueix el minut del gol, el jugador que ha marcat i, si escau, el jugador que ha fet l'assistència. 4. El sistema comprova que tots els camps obligatoris estiguin completats i habilita el botó "Afegir acció" 5. L'usuari prem el botó "Afegir acció" 6. El sistema registra el gol a la cronologia del partit.
Flux alternatiu 1	3. L'usuari deixa buit un camp obligatori 4. El Sistema detecta que hi ha camps buit i deshabilita el botó "Afegir acció" 5. L'usuari torna al punt 3 del flux principal per completar els camps obligatoris
Postcondicions	El sistema ha registrat correctament l'acció de gol a la cronologia del partit.

Taula 1.42. Descripció del cas d'ús UC42 – Afegir gol

Descripció	UC 43 – Afegir targeta
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant la cronologia d'un partit
Flux bàsic	1. L'usuari selecciona la icona d'afegir de l'equip que ha vist la targeta 2. El sistema obre un formulari amb "Gol" com a tipus d'acció 3. L'usuari selecciona "Targeta" com a tipus d'acció, el color de la

	targeta i el jugador i el minut en que l'ha rebut 4. El sistema comprova que tots els camps obligatoris estiguin completats i habilita el botó "Afegir acció" 5. L'usuari prem el botó "Afegir acció" 6. El sistema registra la targeta a la cronologia del partit.
Flux alternatiu 1	3. L'usuari deixa buit un camp obligatori 4. El Sistema detecta que hi ha camps buit i deshabilita el botó "Afegir acció" 5. L'usuari torna al punt 3 del flux principal per completar els camps obligatoris
Postcondicions	La targeta ha estat registrada correctament a la cronologia del partit

Taula 1.43. Descripció del cas d'ús UC43 – Afegir targeta

Descripció	UC 44 – Afegir canvi
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant la cronologia d'un partit
Flux bàsic	1. L'usuari selecciona la icona d'afegir de l'equip que ha vist la targeta 2. El sistema obre un formulari amb "Gol" com a tipus d'acció 3. L'usuari selecciona "Canvi" com a tipus d'esdeveniment, el minut en que es produeix i el jugador que entra i el que surt 4. El sistema comprova que tots els camps obligatoris estiguin completats i habilita el botó "Afegir acció" 5. L'usuari prem el botó "Afegir acció" 6. El sistema registra el canvi a la cronologia del partit.
Flux alternatiu 1	3. L'usuari deixa buit un camp obligatori 4. El Sistema detecta que hi ha camps buit i deshabilita el botó "Afegir acció" 5. L'usuari torna al punt 3 del flux principal per completar els camps obligatoris
Postcondicions	El canvi ha estat registrat correctament a la cronologia del partit

Taula 1.44. Descripció del cas d'ús UC44 – Afegir canvi

Descripció	UC 45 – Eliminar acció de partit
Actors	Usuari
Precondicions	- L'usuari està autenticat - L'usuari està visualitzant la cronologia d'un partit - La cronologia conté almenys una acció
Flux bàsic	1. L'usuari prem la icona d'eliminar d'una acció 2. El sistema sol·licita la confirmació de l'acció: Estàs segur que vols eliminar Gol de Lamine Yamal ? 3. L'usuari prem el botó "Eliminar" 4. El sistema elimina l'exercici
Flux alternatiu 1	3. L'usuari prem el botó "Cancel·lar" 4. El Sistema manté l'exercici i retorna a l'usuari a la cronologia
Postcondicions	El sistema ha eliminat l'acció de la cronologia

Taula 1.45. Descripció del cas d'ús UC45 – Eliminar acció de partit

Annex 2. Descripció de les històries d'usuari

US1 – Registrar-se

Com a usuari no registrat, vull poder crear un compte al sistema, per tal d'accedir a les funcionalitats reservades als usuaris autenticats

Escenari 1 – Validació de camps obligatoris

En cas que l'usuari accedeixi al formulari de registre

Quan ompli parcialment els camps

El sistema mostrarà un missatge d'error: "El camp buit és obligatori"

Escenari 2 – Format del correu electrònic

En cas que l'usuari escrigui un correu electrònic amb un format incorrecte

Quan ompli el camp correu electrònic

El sistema mostrarà un missatge d'error "Correu electrònic no vàlid"

Escenari 3 – Validació de contrasenya

En cas que l'usuari escrigui una contrasenya que no compleix els requisits de seguretat

Quan ompli el camp contrasenya

El sistema mostrarà el missatge d'error "Ha de tenir almenys 12 caràcters, una majúscula, una minúscula, un caràcter especial i un número"

Escenari 4 – Correu ja registrat

En cas que l'usuari introdueixi un correu electrònic ja existent

Quan premi el botó "Crea el teu compte"

El sistema mostra el missatge d'error "Ja existeix un usuari amb aquest correu"

Escenari 5 – Registre correcte

En cas que l'usuari introdueixi dades vàlides i un correu no existent

Quan premi el botó "Crea el teu compte"

El sistema crearà el compte i redirigirà a l'usuari a la pàgina d'inici

US2– Iniciar sessió

Com a usuari registrat, vull poder iniciar sessió amb les meves credencials, per tal d'accedir al meu compte i a les funcionalitats del sistema

Escenari 1 – Validació de camps obligatoris

En cas que l'usuari accedeixi al formulari de registre

Quan ompli parcialment els camps

El sistema mostrarà un missatge d'error "El camp buit és obligatori"

Escenari 2 – Format del correu electrònic

En cas que l'usuari escrigui un correu electrònic amb un format incorrecte

Quan ompli el camp correu electrònic

El sistema mostra el missatge "Correu electrònic no vàlid"

Escenari 3 – Credencials incorrectes

En cas que l'usuari introdueixi un correu inexistent o contrasenya errònia
Quan premi el botó "Inicia Sessió"
El sistema mostrarà el missatge d'error "Correu o contrasenya incorrectes"

Escenari 4 – Inici de sessió amb èxit

En cas que l'usuari introdueixi unes credencials vàlides
Quan premi el botó "Inicia Sessió"
El sistema autenticarà a l'usuari i el redirigirà a la pàgina principal

US3 – Seleccionar equip

Com a usuari autenticat, vull poder veure una llista d'equips disponibles i seleccionar-ne un, per tal de consultar o gestionar-ne la informació

Escenari 1 – Visualització de la llista d'equips

En cas que l'usuari accedeixi a la pàgina d'inici
Quan hi hagi equips registrats
El sistema mostrarà el llistat d'equips que dirigeix l'usuari

Escenari 2 – No hi ha equips disponibles

En cas que l'usuari accedeixi a la pàgina d'inici
Quan no hi hagi cap equip registrat
El sistema mostrarà el missatge "No hi ha equips disponibles"

Escenari 3 – Selecció d'un equip

En cas que l'usuari visualitzi la llista d'equips disponibles
Quan prem sobre un equip de la llista
El sistema redirigeix l'usuari a la pantalla de detall de l'equip i en mostra tota la informació

US4 – Afegir equip

Com a usuari autenticat, vull poder afegir nous equips al sistema, per tal de registrar-los i gestionar-los

Escenari 1 – Visualització del formulari

En cas que l'usuari visualitzi la llista d'equips disponibles
Quan premi el botó "Afegir equip"
El sistema mostrarà un formulari amb els camps nom, categoria i lletra

Escenari 2 – Validació de camps buits

En cas que l'usuari obri el formulari per afegir un equip
Quan algun camp estigui buit
El sistema mantindrà deshabilitat el botó "Afegir equip" fins que s'omplin tots els camps

Escenari 3 – Equip ja existent

En cas que l'usuari ompli el formulari amb un equip ja registrat (mateix nom, categoria i lletra)

Quan premi el botó “Afegir equip”
El sistema mostrarà el missatge d’error “L’equip ja existeix”

Escenari 4 – Cancel·lació de l’acció

En cas que l’usuari obri el formulari i decideixi no continuar
Quan premi el botó de “Cancel·lar”
El sistema tancarà el formulari i retornarà a l’usuari a la pàgina principal

Escenari 5 – Afegir equip correctament

En cas que l’usuari introdueixi un nom, categoria i lletra que no coincideixi amb cap altre equip existent
Quan premi el botó “Afegir equip”
El sistema crearà l’equip i retornarà a l’usuari a la pàgina principal

US5 – Editar equip

Com a usuari autenticat, vull poder editar la informació dels equips, per tal de mantenir les dades actualitzades i facilitar-ne la gestió

Escenari 1 – Visualització del formulari

En cas que l’usuari es trobi a la pàgina principal i tingui equips registrats
Quan premi la icona d’editar d’un equip
El sistema mostrarà un formulari amb el nom, la categoria i la lletra actual de l’equip

Escenari 2 – Validació de camps buits

En cas que l’usuari obri el formulari per editar un equip
Quan algun camp estigui buit
El sistema mantindrà deshabilitat el botó de “Desar canvis” fins que s’omplin tots els camps

Escenari 3 – Equip ja existent

En cas que l’usuari ompli el formulari amb un equip ja registrat (mateix nom, categoria i lletra)
Quan premi el botó “Desar canvis”
El sistema mostrarà el missatge d’error “L’equip ja existeix”

Escenari 4 – Cancel·lació de l’acció

En cas que l’usuari obri el formulari
Quan premi el botó de “Cancel·lar”
El sistema tancarà el formulari i retornarà a l’usuari a la pàgina principal

Escenari 5 – Editar equip correctament

En cas que l’usuari introdueixi un nom, categoria i lletra únics
Quan premi “Desar canvis”
El sistema modificarà l’equip i retorna a l’usuari a la pàgina principal

US6 – Eliminar equip

Com a usuari autenticat, vull poder eliminar equips del sistema, per tal de gestionar només els equips que són actuals

Escenari 1 – Visualització de la confirmació

En cas que l'usuari es trobi a la pàgina principal i tingui equips registrats

Quan premi la icona d'eliminar d'un equip

El sistema mostrarà una finestra de confirmació amb el missatge "Estàs segur que vols eliminar **FC Barcelona**?"

Escenari 2 – Acció cancel·lada

En cas que l'usuari iniciï el procés d'eliminació

Quan premi el botó "Cancel·lar"

El sistema mantindrà l'equip i retornarà a l'usuari a la pantalla principal

Escenari 3 – Eliminació confirmada

En cas que l'usuari iniciï el procés d'eliminació

Quan premi el botó "Eliminar"

El sistema eliminarà l'equip i retornarà a l'usuari a la pantalla principal

US7 – Visualitzar plantilla

Com a usuari autenticat, vull poder veure els jugadors de la plantilla, per tal de consultar-ne les dades i estadístiques

Escenari 1 – Visualització de la plantilla

En cas que l'usuari accedeixi a la secció "Jugadors"

Quan no apliqui cap filtre ni cap cerca

El sistema mostrarà el llistat de jugadors amb nom, posició, dorsal, gols, partits jugats, assistències, grogues, vermelles i percentatge de minuts jugats

Escenari 2 – No existeixen jugadors

En cas que l'usuari accedeixi a la secció "Jugadors"

Quan l'equip no tingui cap jugador

El sistema mostrarà el missatge "L'equip no té cap jugador"

US8 – Cerca de jugadors

Com a usuari autenticat, vull cercar un jugador pel seu nom o dorsal, per tal de trobar-lo ràpidament dins la plantilla.

Escenari 1 – Cerca de jugadors

En cas que l'usuari estigui visualitzant la plantilla,

Quan l'usuari introdueixi un nom o un dorsal

El sistema mostrarà els jugadors que coincideixen amb el criteri de cerca

Escenari 2 – Cap coincidència en la cerca

En cas que l'usuari estigui visualitzant la plantilla,

Quan l'usuari introdueixi un nom o un dorsal que no coincideix amb cap jugador,

El sistema mostrarà el missatge "No s'han trobat jugadors amb aquests criteris de cerca"

US9 – Filtrar per posició

Com a usuari autenticat, vull filtrar els jugadors per posició, per tal de veure només els d'una posició concreta

Escenari 1 – Filtrar per posició

En cas que l'usuari estigui visualitzant la plantilla,
Quan seleccioni una posició del desplegable
El sistema mostrarà només els jugadors d'aquella posició

Escenari 2 – Cap coincidència en la cerca

En cas que l'usuari estigui visualitzant la plantilla,
Quan seleccioni una posició de la qual no hi ha jugadors
El sistema mostrarà el missatge "No s'han trobat jugadors amb aquests criteris de cerca"

US10 – Cercar jugadors i filtrar per posició

Com a usuari autenticat, vull combinar la cerca i el filtre per posició, per tal de trobar jugadors específics dins un rol determinat

Escenari 1 – Cercar i filtrar per posició

En cas que l'usuari estigui visualitzant la plantilla,
Quan l'usuari seleccioni una posició i introdueixi un text al cercador
El sistema mostrarà els jugadors que compleixin ambdós criteris

Escenari 2 – Cap coincidència en la cerca

En cas que l'usuari estigui visualitzant la plantilla,
Quan l'usuari seleccioni una posició i introdueixi un text al cercador que no coincideixi amb cap jugador
El sistema mostrarà el missatge "No s'han trobat jugadors amb aquests criteris de cerca"

US11- Afegir jugador

Com a usuari autenticat, vull poder afegir nous jugadors a la plantilla, per tal de tenir actualitzada la plantilla

Escenari 1 – Visualització del formulari

En cas que l'usuari visualitzi la plantilla de l'equip
Quan premi "Afegir jugador"
El sistema mostrarà un formulari amb els camps nom, dorsal i posició

Escenari 2 – Validació de camps buits

En cas que l'usuari obri el formulari per afegir un jugador
Quan algun camp estigui buit
El sistema deshabilitarà el botó d' "Afegir jugador" fins que s'omplin tots els camps

Escenari 3 – Nom de jugador ja existent

En cas que l'usuari introdueixi un nom ja registrat a l'equip
Quan premi "Afegir jugador"

El sistema mostrarà el missatge d'error "Ja existeix un jugador amb aquest nom a l'equip"

Escenari 4 – Dorsal ja utilitzat

En cas que l'usuari introdueixi un dorsal ocupat a l'equip

Quan premi "Afegir jugador"

El sistema mostrarà el missatge d'error "Ja existeix un jugador amb aquest dorsal a l'equip"

Escenari 5 – Cancel·lació de l'acció

En cas que l'usuari obri el formulari

Quan premi el botó de "Cancel·lar"

El sistema tancarà el formulari i no afegirà cap jugador

Escenari 6 – Afegir equip correctament

En cas que l'usuari emplei les dades amb un nom inexistent i un dorsal no ocupat

Quan premi el botó "Afegir jugador"

El sistema crearà l'equip i retornarà a l'usuari a la pàgina de la plantilla

US12 - Editar jugador

Com a usuari autenticat, vull poder editar la informació d'un jugador existent, per tal de corregir errors o actualitzar la seva posició, nom o dorsal.

Escenari 1 – Visualització del formulari

En cas que l'usuari visualitzi la plantilla i tingui jugadors,

Quan premi la icona d'editar d'un jugador

El sistema mostrarà un formulari amb les dades actuals del jugador

Escenari 2 – Validació de camps buits

En cas que l'usuari obri el formulari per editar un jugador

Quan algun camp estigui buit

El sistema deshabilitarà el botó "Desar canvis" fins que s'omplin tots els camps

Escenari 3 – Nom de jugador ja existent

En cas que l'usuari ompli el formulari amb un nom ja registrat a l'equip

Quan premi "Desar canvis"

El sistema mostrarà el missatge d'error "Ja existeix un jugador amb aquest nom a l'equip"

Escenari 4 – Dorsal ja utilitzat

En cas que l'usuari ompli el formulari amb un dorsal ocupat a l'equip

Quan premi "Desar canvis"

El sistema mostrarà el missatge d'error "Ja existeix un jugador amb aquest dorsal a l'equip"

Escenari 5 – Cancel·lació de l'acció

En cas que l'usuari obri el formulari

Quan premi el botó de "Cancel·lar"

El sistema tancarà el formulari i no modificarà cap jugador

Escenari 6 –Editar jugador correctament

En cas que l'usuari emplei el formulari amb un nom inexistent i un dorsal lliure

Quan premi el botó “Desar canvis”

El sistema actualitzarà el jugador i retornarà a l'usuari a la pàgina de la plantilla

US13 – Eliminar jugador

Com a usuari autenticat, vull poder eliminar jugadors del sistema, per tal de mantenir la llista de jugadors actualitzada

Escenari 1 – Visualització de la confirmació

En cas que l'usuari visualitzi la plantilla de l'equip i hi hagi jugadors

Quan premi la icona d'eliminar d'un jugador

El sistema mostrarà una finestra de confirmació amb el missatge “Estàs segur que vols eliminar **Lamine Yamal**?”

Escenari 2 – Acció cancel·lada

En cas que l'usuari iniciï el procés d'eliminació

Quan premi el botó “Cancel·lar”

El sistema mantindrà el jugador i retornarà a l'usuari a la pantalla de la plantilla

Escenari 3 – Eliminació confirmada

En cas que l'usuari comenci el procés d'eliminació

Quan premi el botó “Eliminar”

El sistema eliminarà el jugador i retornarà a l'usuari a la pantalla de la plantilla

US14 - Visualitzar calendari

Com a usuari autenticat, vull poder veure el calendari del meu equip, per tal de plantificar el meu temps i consultar fàcilment entrenaments i partits

Escenari 1 – Visualització del calendari

En cas que l'usuari hagi accedit a un equip

Quan seleccioni la secció “Calendari”

El sistema mostrarà el calendari en vista mensual amb els entrenaments i partits programats

Escenari 2 – Consultar detall d'un esdeveniment

En cas que hi hagi esdeveniments programats

Quan l'usuari seleccioni un esdeveniment del calendari

El sistema el redirigirà a la pantalla de detall de l'esdeveniment

Escenari 3 – Calendari buit

En cas que l'equip no hi hagi esdeveniments programats

Quan l'usuari accedeixi a la secció “Calendari”

El sistema mostrarà el calendari buit, sense esdeveniments

US15 - Canviar vista del calendari

Com a usuari autenticat, vull poder canviar la vista entre dia, setmana i mes, per tal d'adaptar la visualització a les meves necessitats de planificació

Escenari 1 – Canvi de vista

En cas que l'usuari estigui visualitzant el calendari

Quan premi els botons "Dia", "Setmana" o "Mes"

El sistema actualitzarà la vista del calendari amb els esdeveniments corresponents al període seleccionat.

Escenari 2 – Navegació entre períodes

En cas que l'usuari estigui visualitzant el calendari

Quan seleccioni els botons "<" o ">" per avançar o retrocedir en el temps,

El sistema mostrarà els esdeveniments del període anterior o següent, segons la vista activa.

Escenari 3 – Tornar al dia actual

En cas que l'usuari visualitza un període diferent del dia actual

Quan premi el botó "Avui"

El sistema centrarà la vista en el període actual i mostra els esdeveniments corresponents

US16- Afegir entrenament

Com a usuari autenticat, vull poder afegir un entrenament al calendari, per tal de planificar l'activitat esportiva de l'equip

Escenari 1 – Obrir formulari via botó

En el cas que l'usuari premi el botó "Afegir esdeveniment"

Quan seleccioni l'opció "Entrenament"

El sistema mostrarà un formulari amb data i hora d'inici actuals, hora de fi automàtica (1 hora i 30 minuts després), i ubicació: "Camp Municipal"

Escenari 2 – Obrir formulari prement el calendari

En el cas que l'usuari estigui visualitzant el calendari

Quan l'usuari seleccioni un dia entre setmana del calendari,

El sistema obrirà el formulari d'entrenament amb la data seleccionada preomplerta.

Escenari 3 – Validació de camps buits

En cas que l'usuari obri el formulari per afegir un entrenament

Quan algun camp estigui buit

El sistema deshabilitarà el botó "Afegir entrenament" fins que s'omplin tots els camps

Escenari 4 – Hora de fi anterior a la d'inici

En cas que l'usuari obri el formulari per afegir un entrenament

Quan seleccioni una hora de fi anterior a la d'inici

El sistema deshabilitarà el botó "Afegir esdeveniment"

Escenari 5 – Encavalcament amb un entrenament

En el cas que l'entrenament coincideixi amb un entrenament o partit ja programat,
quan es premi "Afegir esdeveniment",
El sistema mostrarà l'error "Ja existeix un entrenament en aquest horari"

Escenari 6 – Encavalcament amb un partit

En el cas que l'entrenament coincideixi amb un entrenament o partit ja programat,
quan es premi "Afegir esdeveniment",
El sistema mostrarà l'error "Ja existeix un partit en aquest horari".

Escenari 7 – Cancel·lació de l'acció

En cas que l'usuari obri el formulari i decideixi no continuar
Quan premi el botó de "Cancel·lar"
El sistema tanca el formulari i retornarà l'usuari a la secció del calendari

Escenari 8 – Afegir entrenament correctament

En cas que l'usuari empleni les dades amb un horari disponible
Quan prem el botó "Afegir entrenament"
El sistema crearà l'entrenament i retornarà a l'usuari a la pàgina del calendari

US17- Afegir partit

Com a usuari autenticat, vull poder afegir un partit al calendari, per tal de planificar l'activitat esportiva de l'equip

Escenari 1 – Obrir formulari via botó

En el cas que l'usuari premi el botó "Afegir esdeveniment"
Quan seleccioni l'opció "Partit"
El sistema mostrarà un formulari amb data i hora d'inici actuals, ubicació: "Camp Municipal", localitat "local" i rival

Escenari 2 – Obrir formulari prement el calendari

En el cas que l'usuari estigui visualitzant el calendari
Quan l'usuari seleccioni un dia de cap de setmana del calendari,
El sistema obrirà directament el formulari de partit amb la data seleccionada preomplerta.

Escenari 3 – Validació de camps buits

En cas que l'usuari obri el formulari per afegir un partit
Quan algun camp estigui buit
El sistema deshabilitarà el botó "Afegir partit" fins que s'omplin tots els camps

Escenari 4 – Encavalcament amb un entrenament

En el cas que l'entrenament coincideixi amb un entrenament ja programat
Quan es premi "Afegir partit"
El sistema mostrarà l'error "Ja existeix un entrenament en aquest horari"

Escenari 5 – Encavalcament amb un partit

En el cas que l'entrenament coincideixi amb un partit ja programat

Quan es premi “Afegir partit”

El sistema mostrarà l’error “Ja existeix un partit en aquest horari”

Escenari 6 – Cancel·lació de l’acció

En cas que l’usuari obri el formulari i decideixi no continuar

Quan premi el botó “Cancel·lar”

El sistema tancarà el formulari i retornarà a l’usuari a la secció del calendari

Escenari 7 – Afegir partit correctament

En cas que l’usuari emplei les dades amb un horari disponible

Quan premi el botó “Afegir partit”

El sistema crearà el partit i retornarà a l’usuari a la pàgina del calendari

US18- Moure esdeveniment

Com a usuari autenticat, vull poder moure un esdeveniment a una nova data o hora, per tal de reprogramar-lo de manera àgil

Escenari 1 – Moure a una nova data i hora

En cas que l’usuari visualitzi el calendari amb esdeveniments

Quan arrossegui un esdeveniment a una nova data o hora

El sistema actualitzarà l’horari de l’esdeveniment

Escenari 2 – Encavalcament amb un entrenament

En el cas que el nou horari coincideixi amb un entrenament,

quan l’usuari alliberi l’esdeveniment en la nova franja,

El sistema mostrarà el missatge d’error “Ja existeix un entrenament en aquest horari” i retornarà l’esdeveniment al seu lloc original.

Escenari 3 – Encavalcament amb un partit

En el cas que el nou horari coincideixi amb un altre partit,

Quan l’usuari deixi anar l’esdeveniment en la nova franja,

El sistema mostrarà el missatge d’error “Ja existeix un partit en aquest horari” i retornarà l’esdeveniment al seu horari original.

US 19 – Eliminar esdeveniment del calendari

Com a usuari autenticat, vull poder eliminar un esdeveniment del calendari, per tal de gestionar correctament el calendari del meu equip.

Escenari 1 – Confirmació d’eliminació

En el cas que l’usuari visualitzi un calendari,

Quan seleccioni la icona d’eliminar d’un esdeveniment

El sistema mostrarà una finestra de confirmació abans de continuar amb el missatge “Estàs segur que vols eliminar **Partit (Diumenge, 25 Maig 00:00 - 02:00)?**”

Escenari 2 – Acció cancel·lada

En cas que l’usuari iniciï el procés d’eliminació

Quan premi el botó “Cancel·lar”

El sistema mantindrà el jugador i retornarà a l'usuari a la pantalla del calendari

Escenari 3 – Eliminació confirmada

En cas que l'usuari comenci el procés d'eliminació

Quan premi el botó “Eliminar”

El sistema eliminarà el jugador i retorna a l'usuari a la pantalla del calendari

US 20 - Visualitzar informació general de l'entrenament

Com a usuari autenticat, vull visualitzar la informació general d'un entrenament, per tal de conèixer ràpidament els detalls bàsics.

Escenari 1 – Visualització de detalls generals

En cas que l'usuari visualitzi el calendari,

Quan seleccioni un entrenament,

El sistema mostrarà la data, hora d'inici i fi, ubicació i durada.

US 21 – Visualitzar l'assistència a l'entrenament

Com a usuari autenticat, vull veure quins jugadors han assistit a un entrenament, per tal de controlar l'assistència de l'equip.

Escenari 1 – Veure jugadors presents

En cas que l'usuari visualitzi el calendari,

Quan seleccioni un entrenament,

El sistema mostrarà el percentatge d'assistència i la llista de jugadors assistents amb nom i dorsal.

Escenari 2 – Canviar vista a jugadors absents

En cas que l'usuari estigui visualitzant els detalls d'un entrenament,

Quan premi la pestanya “Absents”,

El sistema mostrarà la llista de jugadors absents.

US 22 – Visualitzar exercicis de l'entrenament

Com a usuari autenticat, vull veure els exercicis programats en una sessió, per tal de consultar la planificació de l'entrenament.

Escenari 1 – Entrenament amb exercicis

En cas que l'entrenament contingui exercicis,

Quan l'usuari accedeixi a la sessió,

El sistema mostrarà la llista d'exercicis programats.

Escenari 2 – Entrenament sense exercicis

En cas que l'entrenament no contingui exercicis,

Quan l'usuari accedeixi a la sessió,

El sistema mostrarà el missatge: "Encara no hi ha exercicis per aquesta sessió d'entrenament".

US23 - Editar entrenament

Com a usuari autenticat, vull editar les dades de l'entrenament, per tal de poder-los reprogramar

Escenari 1 – Visualització del formulari

En cas que l'usuari visualitzi els detalls de l'entrenament

Quan premi la icona d'editar de les dades generals de l'entrenament

El sistema mostrarà un formulari amb la data l'horari i la ubicació actuals de l'entrenament

Escenari 2 – Validació de camps buits

En cas que l'usuari obri el formulari per editar les dades d'un entrenament

Quan algun camp estigui buit

El sistema deshabilitarà el botó "Desar canvis" fins que s'omplin tots els camps

Escenari 3 – Hora de fi anterior a la d'inici

En cas que l'usuari obri el formulari per editar l'un entrenament

Quan seleccioni una hora de fi anterior a la d'inici

El sistema deshabilitarà el botó "Desar canvis"

Escenari 4 – Encavalcament amb un partit

En cas que l'horari de l'entrenament coincideixi amb un partit,

Quan l'usuari premi el botó "Desar canvis"

El sistema mostrarà el missatge d'error "Ja existeix un partit en aquest horari"

Escenari 5 – Encavalcament amb un entrenament

En cas que l'horari de l'entrenament coincideixi amb un partit,

Quan l'usuari premi el botó "Desar canvis"

El sistema mostrarà el missatge d'error "Ja existeix un entrenament en aquest horari"

Escenari 6 – Cancel·lació de l'acció

En cas que l'usuari obri el formulari i decideixi no continuar

Quan premi el botó de "Cancel·lar"

El sistema tancarà el formulari i no modificarà l'entrenament

Escenari 7 – Editar entrenament correctament

En cas que l'usuari emplei el formulari amb unes dades vàlides

Quan premi el botó "Desar canvis"

El sistema modificarà l'entrenament i retornarà a l'usuari a la pàgina detallada de l'entrenament

US 24 – Marcar un jugador com a absent

Com a usuari autenticat, vull poder marcar jugadors presents com a absents, per tal de poder actualitzar l'assistència a un entrenament

Escenari 1 – Marcar un jugador com a absent

En cas que l'usuari estigui visualitzant la llista de presents,
Quan premi la icona d'absència,
El sistema mourà el jugador a la llista d'absents.

UC 25 – Marcar un jugador com a present

Com a usuari autenticat, vull poder marcar jugadors absents com a presents, per tal de poder actualitzar l'assistència a un entrenament

Escenari 1 – Marcar un jugador com a present

En cas que l'usuari estigui visualitzant la llista d'absents,
Quan premi la icona de presència,
El sistema mourà el jugador a la llista de presents.

UC 26 – Marcar tots els jugadors com a absents

Com a usuari autenticat, vull poder marcar tots els jugadors presents com a absents, per tal de poder actualitzar massivament l'assistència a un entrenament

Escenari 1 – Marcar tots els jugadors com a absents

En cas que visualitzi l'assistència,
Quan premi el botó "Tots absents",
El sistema mourà tots els jugadors a la llista d'absents.

UC 27 – Marcar tots els jugadors com a presents

Com a usuari autenticat, vull poder marcar tots els jugadors presents com a absents, per tal de poder actualitzar massivament l'assistència a un entrenament

Escenari 1 - Marcar tots els jugadors com a presents

En cas que visualitzi l'assistència,
Quan premi el botó "Tots presents",
El sistema mourà tots els jugadors a la llista de presents.

UC 28 – Afegir exercici a l'entrenament

Com a usuari autenticat, vull poder afegir un exercici, per tal de poder organitzar els entrenaments

Escenari 1 – Visualització del formulari

En cas que l'usuari visualitzi el calendari
Quan premi el botó "Afegir exercici"
El sistema mostrarà un formulari amb els camps nom, descripció i durada buits

Escenari 2 – Validació de camps buits

En cas que l'usuari obri el formulari per afegir un exercici
Quan algun camp estigui buit

El sistema mantindrà deshabilitat el botó “Afegir exercici” fins que s’omplin tots els camps

Escenari 3 – Cancel·lació de l’acció

En cas que l’usuari obri el formulari i decideixi no continuar

Quan premi el botó “Cancel·lar”

El sistema tanca el formulari i no fa cap canvi

Escenari 4 – Afegir exercici correctament

En cas que l’usuari empleni totes les dades

Quan premi el botó “Afegir exercici”

El sistema crearà l’exercici i retornarà a l’usuari a la pàgina detallada de l’entrenament

UC 29 – Editar exercici de l’entrenament

Com a usuari autenticat, vull poder modificar un exercici, per tal de poder ajustar-lo segons les necessitats de l’entrenament

Escenari 1 – Visualització del formulari

En cas que l’usuari visualitzi els detalls de l’entrenament

Quan premi la icona d’editar d’un exercici

El sistema mostrarà un formulari amb les dades actuals de l’exercici

Escenari 2 – Validació de camps buits

En cas que l’usuari visualitzi el formulari per afegir un exercici

Quan algun camp estigui buit

El sistema deshabilitarà el botó “Desar canvis” fins que s’omplin tots els camps

Escenari 3 – Cancel·lació de l’acció

En cas que l’usuari obri el formulari i decideixi no continuar

Quan premi el botó “Cancel·lar”

El sistema tancarà el formulari i no modificarà l’exercici

Escenari 4 – Modificar exercici correctament

En cas que l’usuari empleni totes les dades

Quan premi el botó “Desar canvis”

El sistema modificarà l’exercici i retornarà a l’usuari a la pàgina detallada de l’entrenament

UC 30 – Afegir dibuix tàctic a l’exercici

Com a usuari autenticat, vull afegir un dibuix tàctic a un exercici, per tal de representar-lo visualment.

Escenari – 1 Mostrar el camp editable

En cas que l’usuari visualitzi un exercici

Quan seleccioni “Afegir dibuix tàctic”

El sistema mostrarà un camp de futbol editable

Escenari – 2: Desar dibuix

En cas que l'usuari visualitzi el camp de futbol editable

Quan realitzi un dibuix i premi el botó “Desa”

El sistema guardarà el dibuix a l'exercici

Escenari – 3: Cancel·lar dibuix

En cas que l'usuari visualitzi el camp de futbol editable

Quan l'usuari premi el botó “Cancel·la”

El sistema descartarà el dibuix i tornarà a mostrar els exercicis

UC 31 – Editar dibuix tàctic de l'exercici

Com a usuari autenticat, vull modificar un dibuix tàctic, per tal de modificar la representació.

Escenari 1 – Obrir pissarra tàctica

En cas que l'usuari visualitzi un exercici

Quan l'usuari seleccioni la icona d'editar del dibuix tàctic d'un exercici

El sistema mostrarà el camp de futbol editable amb el dibuix actual

Escenari 2 - Actualitzar dibuix

En cas que l'usuari visualitzi el camp de futbol editable

Quan modifiqui el dibuix i premi el botó “Desa”

El sistema actualitzarà el dibuix de l'exercici

Escenari 3 - Cancel·lar dibuix

En cas que l'usuari visualitzi el camp de futbol editable

Quan l'usuari premi el botó “Cancel·la”

El sistema descartarà el dibuix i torna a mostrar els exercicis

UC 32 – Eliminar dibuix tàctic de l'exercici

Com a usuari autenticat, vull eliminar un dibuix tàctic, per tal de d'eliminar la representació gràfica d'un exercici

Escenari 1 – Visualització de la confirmació

En cas que l'usuari visualitzi un entrenament i hi hagi un dibuix tàctic

Quan premi la icona d'eliminar d'un exercici

El sistema mostrarà una finestra de confirmació amb el missatge “Estàs segur que vols eliminar **dibuix tàctic**”?

Escenari 2 – Acció cancel·lada

En cas que l'usuari iniciï el procés d'eliminació

Quan premi “Cancel·lar”

El sistema mantindrà la imatge al i retorna a l'usuari als exercicis de l'entrenament

Escenari 3 – Eliminació confirmada

En cas que l'usuari comenci el procés d'eliminació

Quan premi el botó “Eliminar”

El sistema eliminarà la imatge de l'exercici i retornarà als exercicis de l'entrenament

UC 33 – Eliminar exercici

Com a usuari autenticat, vull poder eliminar un exercici, per tal que l'entrenament només inclogui els exercicis que es realitzaran en aquesta sessió

Escenari 1 – Visualització de la confirmació

En cas que l'usuari visualitzi un entrenament i hi hagi exercicis

Quan premi la icona d'eliminar d'un exercici

El sistema mostrarà una finestra de confirmació amb el missatge "Estàs segur que vols eliminar **Rondo**"?

Escenari 2 – Acció cancel·lada

En cas que l'usuari inici el procés d'eliminació

Quan premi "Cancel·lar"

El sistema mantindrà l'exercici i retornarà a l'usuari als detalls de l'entrenament

Escenari 3 – Eliminació confirmada

En cas que l'usuari comenci el procés d'eliminació

Quan premi el botó "Eliminar"

El sistema eliminarà l'exercici i retornarà a l'usuari a la pantalla dels detalls de l'entrenament

US34 - Visualitzar detalls del partit

Com a usuari autenticat, vull poder veure els detalls del partit, per tal de saber quan i on es juga el partit i quin és el marcador

Escenari 1 – Visualització dels detalls del partit

En cas que l'usuari estigui a la secció "calendari"

Quan seleccioni un partit

El sistema mostrarà l'horari, la data, el rival, i la ubicació del partit

Escenari 2 – Visualització del resultat del partit

En cas que el partit hagi s'hagi disputat

Quan l'usuari accedeixi als detalls d'un partit

El sistema mostrarà també el resultat del partit

US35 - Visualitzar cronologia del partit

Com a usuari autenticat, vull poder veure la cronologia del partit, per tal de saber com ha evolucionat el partit

Escenari 1 – Visualització de la cronologia

En cas que l'usuari estigui a la secció "calendari"

Quan seleccioni un partit

El sistema mostrarà els gols, les targetes i els canvis amb el minut en que ha succeït i els jugadors implicats

Escenari 2 – No hi ha accions de partit

En cas que no s'hagin produït cap acció destacada en el partit

Quan vegi la cronologia

El sistema mostrarà el missatge "No hi ha esdeveniments registrats"

US 36 - Veure alineació del partit

Com a usuari autenticat, vull poder veure la alineació, per tal de veure quins jugadors han participat en el partit

Escenari 1 – Visualització de l'alineació

En cas que l'usuari estigui a la pàgina detallada del partit

Quan seleccioni "Alineació"

El sistema mostrarà els jugadors que han sigut titulars, els que han sigut suplents i els no convocats

Escenari 2 – No hi ha alineació

En cas que no s'hagi completat l'alineació per al partit

Quan l'usuari vegi l'alineació

El sistema mostrarà tots els jugadors de la plantilla com a no convocats

US36 - Editar detalls del partit

Com a usuari autenticat, vull poder editar la data, l'horari, la ubicació o el rival d'un partit per tal de corregir o actualitzar la informació del partit quan sigui necessari.

Escenari 1 – Visualització del formulari

En cas que l'usuari visualitzi els detalls del partit

Quan premi la icona d'editar de les dades generals del partit

El sistema mostrarà un formulari amb la data, l'hora, la ubicació i el rival actuals del partit

Escenari 2 – Validació de camps buits

En cas que l'usuari obri el formulari per editar les dades un partit

Quan algun camp estigui buit

El sistema deshabilitarà el botó "Desar canvis" fins que s'omplin tots els camps

Escenari 3 – Encavalcament amb un partit

En cas que l'horari del partit coincideixi amb un altre partit

Quan l'usuari premi "Desar canvis"

El sistema mostrarà el missatge d'error "Ja existeix un partit en aquest horari"

Escenari 4 – Encavalcament amb un entrenament

En cas que l'horari del partit coincideixi amb entrenament

Quan l'usuari premi "Desar canvis"

El sistema mostrarà el missatge d'error "Ja existeix un entrenament en aquest horari"

Escenari 5 – Cancel·lació de l'acció

En cas que l'usuari obri el formulari i decideixi no continuar

Quan premi el botó "Cancel·lar"

El sistema tancarà el formulari i no modificarà el partit

Escenari 6 – Editar entrenament correctament

En cas que l'usuari empleni les dades correctament

Quan premi el botó "Desar canvis"

El sistema modificarà el partit i retorna a l'usuari a la pàgina detallada del partit

US37 - Moure jugador com a titular

Com a usuari autenticat, vull poder assignar un jugador suplent o no convocat com a titular, per tal de configurar l'onze inicial del partit

Escenari 1 – Assignació a una posició buida

En cas que l'usuari visualitzi l'alineació del partit

Quan arrossegui un jugador suplent o no convocat a una posició buida del camp

El sistema l'assignarà com a titular i l'eliminarà de la seva categoria anterior

Escenari 2 – Substitució de jugador titular

En cas que l'usuari visualitzi l'alineació del partit

Quan arrossegui un jugador suplent o no convocat a una posició ocupada del camp

El sistema intercanviarà els rols dels jugadors

US 38 - Moure jugador com a suplent

Com a usuari autenticat, vull poder assignar un jugador titular o no convocat com a suplent, per tal de configurar la banqueta del partit

Escenari 1 – Assignació a una posició buida

En cas que l'usuari visualitzi l'alineació del partit

Quan arrossegui un jugador titular o no convocat a una posició buida de la banqueta

El sistema l'assignarà com a suplent i l'eliminarà de la seva categoria anterior

Escenari 2 – Substitució de jugador suplent

En cas que l'usuari visualitzi l'alineació del partit

Quan arrossegui un jugador titular o no convocat a una posició ocupada de la banqueta

El sistema intercanviarà els rols dels jugadors

US 39- Moure jugador a no convocat

Com a usuari autenticat, vull poder assignar un jugador titular o suplent com a titular, per tal de configurar la convocatòria del partit

Escenari 1 – Desconvocar un jugador

En cas que l'usuari visualitzi l'alineació del partit

Quan arrossegui un jugador titular o suplent al llistat de jugadors no convocats

El sistema l'assignarà com a no convocat i deixarà buida la seva posició al camp

US 40- Canviar formació

Com a usuari autenticat, vull poder canviar la formació tàctica de l'equip , per tal de adaptar l'estructura de l'alineació a com juga l'equip

Escenari 1 – Seleccionar una nova formació

En cas que l'usuari visualitzi l'alineació del partit
Quan seleccioni una formació diferent
El sistema actualitzarà la disposició tàctica

Escenari 2 – Reubicació de jugadors

En cas que l'alineació compti amb jugadors titulars
Quan l'usuari modifiqui la formació
El sistema reubicarà els jugadors, assignant-los a posicions el més similars possible a les originals i garantint que continuïn com a titulars.

US41- Afegir gol

Com a usuari autenticat, vull poder afegir un gol a la cronologia del partit, per tal de registrar qui ha marcat i quan s'ha produït.

Escenari 1 – Visualització formulari

En cas que l'usuari visualitzi la cronologia del partit
Quan seleccioni el botó "Afegir esdeveniment"
El sistema mostrarà un formulari amb els camps per introduir el jugador que ha marcat, el minut i opcionalment el jugador que ha fet l'assistència

Escenari 2 – Validació de camps buits

En cas que l'usuari obri el formulari per afegir un gol
Quan algun camp estigui buit
El sistema deshabilitarà el botó "Afegir esdeveniment" fins que s'omplin tots els camps

Escenari 3 – Afegir gol

En cas que l'usuari ompli el formulari per afegir un gol
Quan seleccioni el botó "Afegir esdeveniment"
El sistema registrarà el gol a la cronologia del partit

US 42- Afegir targeta

Com a usuari autenticat, vull poder afegir una targeta a la cronologia del partit, per tal de registrar qui ha vist una targeta i en quin moment.

Escenari 1 – Visualització formulari

En cas que l'usuari visualitzi la cronologia del partit
Quan seleccioni el botó "Afegir esdeveniment" i seleccioni "Targeta" com a tipus d'esdeveniment
El sistema mostrarà un formulari amb els camps per introduir el jugador, el minut i el color de la targeta

Escenari 2 – Validació de camps buits

En cas que l'usuari obri el formulari per afegir una targeta

Quan algun camp estigui buit

El sistema deshabilitarà el botó "Afegir esdeveniment" fins que s'omplin tots els camps

Escenari 3 – Afegir targeta

En cas que l'usuari ompli el formulari per afegir una targeta

Quan seleccioni el botó "Afegir esdeveniment"

El sistema registrarà la targeta a la cronologia del partit

US 43- Afegir canvi

Com a usuari autenticat, vull poder afegir un canvi a la cronologia del partit, per tal de reflectir les substitucions de jugadors durant partit

Escenari 1 – Visualització formulari

En cas que l'usuari visualitzi la cronologia del partit

Quan seleccioni el botó "Afegir esdeveniment" i seleccioni "Canvi" com a tipus d'esdeveniment

El sistema mostrarà un formulari amb els camps per introduir el jugador que entra, el jugador que surt i el minut del canvi.

Escenari 2 – Validació de camps buits

En cas que l'usuari obri el formulari per afegir una targeta

Quan algun camp estigui buit

El sistema deshabilitarà el botó de "Afegir esdeveniment" fins que s'omplin tots els camps

Escenari 3 – Afegir canvi

En cas que l'usuari ompli el formulari per afegir una targeta

Quan seleccioni el botó "Afegir esdeveniment"

El sistema registrarà el canvi a la cronologia del partit

US 44- Eliminar acció de partit

Com a usuari autenticat, vull poder eliminar un esdeveniment de la cronologia, per tal de mantenir la cronologia del partit actualitzada

Escenari 1 – Visualització de la confirmació

En cas que l'usuari visualitzi la cronologia i hi hagi esdeveniments

Quan prem la icona d'eliminar d'un esdeveniment

El sistema mostra una finestra de confirmació abans de continuar amb el missatge "Estàs segur que vols eliminar **Gol de Lamine Yamal** ?"

Escenari 2 – Acció cancel·lada

En cas que l'usuari iniciï el procés d'eliminació

Quan premi el botó "Cancel·lar"

El sistema mantindrà l'esdeveniment a la cronologia i retornarà a l'usuari als detalls del partit

Escenari 3 – Eliminació confirmada

En cas que l'usuari comença el procés d'eliminació

Quan premi el botó "Eliminar"

El sistema elimina l'esdeveniment de la cronologia i retornarà a l'usuari als detalls del partit