

---

This is the **published version** of the bachelor thesis:

Alcover Neto, Daniel; Montenegro Ruiz, Victoria, dir. Creació d'una aplicació web per a la gestió en un consell d'esports de l'àmbit municipal, en relació amb l'assignació d'horari, persones destinades a l'arbitratge i taules de control. 2022. (Enginyeria Informàtica)

---

This version is available at <https://ddd.uab.cat/record/280718>

under the terms of the  license

---

This is the **published version** of the bachelor thesis:

Alcover Neto, Daniel; Montenegro Ruiz, Victoria, dir. Creació d'una aplicació web per a la gestió en un consell d'esports de l'àmbit municipal, en relació amb l'assignació d'horari, persones destinades a l'arbitratge i taules de control. 2022. (Enginyeria Informàtica)

---

This version is available at <https://ddd.uab.cat/record/280718>

under the terms of the  license

# Creació d'una aplicació web per a la gestió en un consell d'esports de l'àmbit municipal, en relació amb l'assignació d'horari, persones destinades a l'arbitratge i taules de control

Daniel Alcover Neto

## Resum

Aquest projecte consisteix en el desenvolupament d'una aplicació web per a la gestió d'un consell escolar d'esports. Els principals objectius sn desenvolupar una aplicació web funcional que serveixi per gestionar i controlar un consell escolar d'esports. També la centralització, és a dir, en una sola aplicació web ha d'haver tot el necessari per complir l'anterior objectiu, ja que, avui dia la majoria de consells escolars tenen dues aplicacions web: una per la visualització de les dades per part dels usuaris finals i una altra per a la gestió interna del consell i les dades que després seran mostrades a l'altra aplicació. Finalment l'últim objectiu és aplicar un sistema de rols pels usuaris registrats per tenir accés o no a les diferents funcionalitats de l'aplicació web.

La metodologia utilitzada ha estat Scrum, ja que gràcies al seu sistema de funcionament, permet organitzar fàcilment les diferents tasques en sprints i l'ordre d'aquests per dur a terme el projecte. S'ha utilitzat diferents tecnologies que estan a l'ordre del dia pel desenvolupament: Angular pel Frontend, Django pel Backend i les APIs i MySQL per a la base de dades.

En conclusió, s'ha desenvolupat una aplicació web funcional que compleix amb tots els objectius esmentats anteriorment.

## Paraules clau

Aplicació Web, Angular, Gestió de Consell Escolar, MySql, Lliga Interescolar, Django, Arbitratge

## Abstract

This project involves the development of a web application for managing a school sports council. The main objectives are to develop a functional web application that serves to manage and control a school sports council. Also, centralization, that is, all the necessary tools for achieving the previous objective must be within a single web application, as most school councils currently have two web applications: one for data visualization by end-users and another for internal council management and the data that will later be shown in the other application. The final objective is to apply a role system for registered users to have access or not to the different functionalities of the web application.

The methodology used was Scrum, as thanks to its operating system, it easily organizes the different tasks into sprints and the order of these to carry out the project. Various leading-edge technologies have been used for the development: Angular for the Frontend, Django for the Backend and APIs, and MySQL for the database.

In conclusion, a functional web application has been developed that meets all the objectives mentioned above.

## Index Terms

Web Application, Angular, School Board Management, MySql, School League, Django, Arbitration

---

## 1 INTRODUCCIÓ - CONTEXT DEL TREBALL

En l'actualitat, la promoció d'activitats esportives en l'àmbit escolar és fonamental per a fomentar hàbits saludables i valors com el treball en equip, la disciplina i el respecte entre els estudiants. Els consells escolars d'esports són responsables d'organitzar i gestionar esdeveniments esportius en l'entorn educatiu. Per a dur a terme aquesta tasca de manera eficient, és necessari comptar amb eines i sistemes de gestió adequats que facilitin la planificació, coordinació i seguiment d'aquestes activitats.

El desenvolupament d'aplicacions web ha experimentat un creixement significatiu en els últims anys, permetent la

creació de solucions a mesura per a abordar problemàtiques específiques en diversos àmbits. En el cas dels consells escolars d'esports, existeix una oportunitat per a desenvolupar una aplicació web que simplifiqui i millori la gestió de les seves activitats i esdeveniments esportius, brindant als responsables una plataforma fàcil d'usar i accessible des de qualsevol dispositiu amb connexió a Internet.

La justificació d'aquest projecte de final de grau en enginyeria informàtica radica en la necessitat de proveir als consells escolars d'esports d'una eina tecnològica que els permeti dur a terme les seves responsabilitats de manera eficient i efectiva. A més, el projecte contribuirà al camp de l'enginyeria informàtica, aplicant i combinant coneixements i tècniques de desenvolupament web, bases de

dades, disseny d'interfícies d'usuari i experiència d'usuari per a crear una solució innovadora i útil per al sector educatiu i esportiu.

En resum, el desenvolupament d'una aplicació web per a la gestió de consells escolars d'esports permetrà millorar l'organització i coordinació d'activitats i esdeveniments esportius, la qual cosa tindrà un impacte positiu en l'experiència dels estudiants i el personal involucrat en aquests programes. Al mateix temps, el projecte brindarà a l'estudiant l'oportunitat d'aplicar i demostrar les seves habilitats i coneixements adquirits en l'àmbit de l'enginyeria informàtica. També el permetrà aprendre noves tecnologies molt usades avui dia.

Aquest projecte té 3 objectius principals:

- Desenvolupar una aplicació web funcional que sigui capaç de portar la gestió i el control d'un consell escolar d'esports. És a dir, una aplicació que controli les diferents lligues simultànies, els grups d'aquestes, la classificació de cada un d'ells i els partits que el componen. A més de tota la informació que engloba un partit com la localització, la data o els equips involucrats.
- La centralització en quant a la idea que una sola aplicació web englobi totes les tasques necessàries per complir l'anterior objectiu. Això comporta el control dels partits i les seves dades relacionades com a punt principal; i també la gestió de les assignacions dels àrbitres. A més, que aquests puguin afegir els resultats de cada partit. Però l'aplicació web també ha de ser capaç fer gestions en l'àmbit de lliga com poder modificar les dades guardades (afegir o eliminar equips/àrbitres/grups) i inserir/modificar/eliminar partits/grups.
- Totes les funcionalitats dins de l'aplicació web han d'estar disponibles en la mateixa aplicació web, però segons els rols de l'usuari es tindrà accés a unes funcionalitats o a altres.

## 2 MARC TEÒRIC

El desenvolupament web es divideix en dos components principals: el frontend i el backend:

- Frontend es refereix a la part visible i interactiva de l'aplicació, que els usuaris experimenten a través dels seus navegadors web. Aquesta part del desenvolupament implica l'ús de llenguatges de marcat (HTML), fulles d'estil (CSS) i llenguatges de programació (JavaScript) per a dissenyar i construir interfícies d'usuari atractives i funcionals.
- Backend és la part del sistema que gestiona la lògica del negoci, processa les sol·licituds de l'usuari i es comunica amb les bases de dades i altres serveis. El desenvolupament backend implica la utilització de llenguatges de programació com Python, així com el disseny i la implementació de bases de dades utilitzant sistemes de gestió de bases de dades (DBMS) com MySQL.

Per aquest projecte, s'empraran tecnologies i pràctiques modernes de desenvolupament web, que permetin crear

una aplicació escalable, segura i fàcil de mantenir. En l'àmbit del frontend s'usarà un framework de JavaScript

anomenat Angular i pel backend un framework de Python anomenat Django. Gràcies a l'ús d'aquests frameworks, ja tenim garantida la compatibilitat i accessibilitat en diferents dispositius i navegadors (i les seves diferents versions) ja que Angular ja hi està preparat per això. També caldrà seguir les millors pràctiques de seguretat i protecció de dades.

## 3 ESTAT DE L'ART

Els consells escolars d'esports són entitats encarregades de coordinar i supervisar les activitats esportives dins de l'àmbit interescolar. Aquestes entitats exerceixen un paper crucial en l'activitat física entre els estudiants, contribuint al desenvolupament d'hàbits saludables, habilitats socials i valors fonamentals, com el treball en equip, el respecte i la responsabilitat.

Donat el paper central dels consells escolars d'esports en la gestió d'activitats esportives en l'àmbit escolar, és fonamental comptar amb eines eficients i efectives per a dur a terme les seves responsabilitats.

Avui dia i per norma general, els consells escolars funcionen amb dues aplicacions web diferents.

- Una primera on es mostren les dades de les lligues, grups, equips, partits, etc. No cal estar registrat i exclusivament s'usa per mostrar les dades/notícies, etc.
- Una segona que utilitzen per gestionar el consell escolar i les dades de les diferents lligues. És obligatori estar-hi registrat i és exclusiu pels treballadors del consell.

Solen ser aplicacions web subcontractades a empreses que es dediquen a mantenir aplicacions web complexes, per tant, el propi consell només paga una mensualitat i aquesta s'encarrega de tot.

Aquesta divisió en dues aplicacions té una sèrie d'avantatges i inconvenients: per exemple, com a avantatge el consell no es preocupa de res del sistema, ja que l'empresa contractada s'encarrega del manteniment. En canvi, ha de tenir les dues aplicacions funcionant sempre, per tant, ha de pagar pel manteniment de les dues. També, a nivell del desenvolupament d'un sistema amb dues aplicacions funcionant sobre les mateixes dades, resulta més complexe, per tant, el preu en el moment del desenvolupament és més alt.

L'aplicació web proposada en aquest projecte té com a objectiu millorar la gestió dels consells escolars d'esports, oferint una plataforma centralitzada per a l'organització i supervisió de les activitats esportives i la gestió de recursos. En facilitar aquests processos, l'aplicació contribuirà a l'activitat física entre els estudiants i donarà suport al treball dels consells escolars d'esports en la consecució dels seus objectius.

El fet de tenir una sola aplicació web, elimina bona part dels inconvenients esmentats anteriorment, ja que, tenint una sola aplicació, només cal pagar un manteniment i el seu desenvolupament pot ser més econòmic.

## 4 TECNOLOGIES UTILITZADES EN EL PROJECTE

En aquest projecte, s'han emprat diverses tecnologies i eines per a desenvolupar una aplicació web eficient i escalable que gestioni el consell escolar d'esports. A continuació, es descriuen les tecnologies clau utilitzades en el projecte i les raons darrere de la seva elecció:

- Base de dades: MySQL és un sistema de gestió de bases de dades relacional (RDBMS) de codi obert que utilitza SQL (Structured Query Language) per a gestionar i accedir a les dades. S'ha seleccionat MySQL com a tecnologia per a la base de dades a causa del seu rendiment, estabilitat i escalabilitat. A més, MySQL compta amb una àmplia comunitat de desenvolupadors i una sòlida documentació, la qual cosa garanteix la facilitat d'ús i el suport continu.
- Backend: Django és un framework de desenvolupament web d'alt nivell basat en Python que permet la creació ràpida d'aplicacions web robustes i escalables. S'ha triat Django a causa del seu enfocament en la reutilització de components i la seva arquitectura modular, la qual cosa permet una major productivitat i flexibilitat en el desenvolupament. A més, Django segueix el patró de disseny Model-Vista-Controlador (MVC) i proporciona una capa d'abstracció per a interactuar amb la base de dades, la qual cosa simplifica la implementació de la lògica del negoci i facilita la integració amb altres tecnologies. A més, té una fàcil adaptació per a usar MySQL.
- Frontend: Angular és un popular framework de desenvolupament web basat en TypeScript desenvolupat per Google, que facilita la construcció d'aplicacions web d'una sola pàgina (SPA) i ofereix una experiència d'usuari fluida i dinàmica. L'elecció d'Angular es deu al seu enfocament en la modularitat i la reutilització de components, la qual cosa facilita la creació d'interfícies d'usuari riques i escalables. A més, Angular compta amb una àmplia comunitat de desenvolupadors i una gran quantitat de recursos i biblioteques disponibles, la qual cosa agilitza el procés de desenvolupament i garanteix l'adopció de bones pràctiques.

La combinació d'aquestes tecnologies permet fer una aplicació web funcional i senzilla però complint tant els objectius com els estàndards bàsics d'una aplicació web.

Durant el projecte s'ha utilitzat una tecnologia més que és Docker. S'ha usat pura i exclusivament durant el desenvolupament del projecte:

- Docker és una plataforma de codi obert que permet automatitzar el desplegament, l'administració i l'execució d'aplicacions dins de contenidors. Els contenidors són entorns aïllats i lleugers que encapsulen una aplicació i totes les seves dependències, incloent-hi llibreries, frameworks i fitxers de configuració necessaris perquè funcioni correctament en qualsevol entorn. S'ha escollit Docker per tenir la base de dades en un contenidor durant el desenvolupament, ja que, un contenidor reproduceix

amb gaire certesa l'entorn on hi estaria després de desplegar l'aplicació. A més, per l'aïllament, ja que, tenint la base de dades en un contenidor certifiques que no pugui haver conflictes amb altres aplicacions o serveis.

## 5 METODOLOGIA

La metodologia té una importància crucial en el desenvolupament de projectes de software, incloent-hi les aplicacions web. Proporciona una estructura i un enfocament sistemàtic per planificar, executar i controlar totes les etapes del projecte. Per tant, és molt important tenir un marc de feina en quant a la metodologia adequat per al projecte i que tots els integrants de l'equip sàpiguen com funciona i el puguin seguir correctament.

El marc de feina escollit per dur a terme aquest projecte és Scrum. És una metodologia àgil de gestió de projectes que es basa en la col·laboració i l'adaptabilitat per aconseguir resultats òptims. S'enfoca en treballar de manera iterativa i incremental, organitzant el treball en cicles de desenvolupament anomenats "sprints".

Els sprints són iteracions fixes de temps (normalment de dues a quatre setmanes) que en el cas d'aquest projecte són de dues setmanes. En cada sprint, es fa una selecció d'una quantitat de tasques prioritzades i es planifiquen per a la implementació. Al final de cada sprint, es pot proporcionar una versió funcional del producte.

Com a resum, Scrum ajuda a que hi hagi un control continu sobre el desenvolupament, i se sàpiga en cada moment que s'ha de fer i el temps màxim que hi ha per fer-ho, ja que, s'ha marcat en la planificació inicial. També ajuda a dividir la feina de tot el projecte en "trossos" més petits i independents que fan més fàcil la seva comprensió.

L'eina escollida per portar tota la gestió del projecte és Trello. És una eina molt intuïtiva i fàcil d'utilitzar. Ajuda a reduir gaire el temps de gestió d'un projecte i, a més, ofereix diferents maneres visuals d'observar l'estat del projecte en un cert moment.

Una pissarra al Trello és una utilitat que permet inserir tasques i portar un control sobre aquestes. Per tant, dins un sol projecte pots tenir tantes pissarres com sprints. Això fa que es tingui el control de cada sprint individualitzat, a més, d'un control general de totes les pissarres.

### 5.1 Requeriments a desenvolupar

La llista de requeriments a desenvolupar durant aquest projecte és la següent:

1. Crear una base de dades relacional amb MySQL per guardar totes les dades dels usuaris, equips, àrbitres i partits.
2. Crear un backend senzill encarregat de la comunicació amb la base de dades i també de fer unes funcionalitats senzilles com podria ser l'acció de "login".
3. Crear les diferents APIs perquè es puguin comunicar el Frontend i el Backend.
4. Fer una aplicació web amb Angular que tingui les següents pantalles :
  - a. Pantalla principal.

- b. Llista de lligues i grups.
- c. Llista d'equips i camps/localitzacions.
- d. Classificació, equips i partits dels diferents grups de cada lliga.
- e. Pantalla per assignar àrbitres als partits per part d'un gestor.
- f. Pantalla per visualitzar partits assignats a un àrbitre.
- g. Pantalla per modificar resultats dels partits per part dels àrbitres.
- h. Pantalla de creació de nou equip, club, partit, lliga, grup i camp/localització.

La completació de tots aquests requeriments certifiquen tenir una aplicació web funcional preparada per gestionar lligues, grups i els partits que els componen.

## 6 ACTORS/ROLS

L'aplicació web té definits diferents actors que engloben tots els possibles usuaris que poden interactuar amb aquest. Es fa aquesta divisió en actors per marcar l'accés a les diferents funcionalitats:

- Administrador: és l'encarregat de mantenir la integritat de les dades i modificar-les segons pertoqui. També de crear usuaris específics. Té accés a totes les funcionalitats de l'aplicatiu.
- Gestor: és l'encarregat de la gestió de les lligues, és a dir, de crear els partits a l'inici de cada lliga i de les assignacions dels àrbitres poques setmanes abans de cada partit.
- Àrbitre: és l'encarregat d'afegir els resultats dels partits assignats a ell/a.
- Usuari no registrat: són totes aquelles persones que no treballen al consell, però que volen accedir a l'aplicatiu per consultar informació sobre les lligues. No cal que hi tinguin un usuari registrat i, per tant, no és necessari iniciar sessió.

## 7 PLANIFICACIÓ

Aquest projecte es divideix en 4 parts principals i 7 sprints.

Fases:

1. Definició.
  - a. Sprint 1:
    - i. Fer disseny i planificació inicial de tot el projecte.
2. Implementació.
  - a. Sprint 2:
    - i. Dissenyar i muntar la base de dades SQL.
  - b. Sprint 3:
    - i. Crear el projecte Backend, la connexió amb la base de dades i els casos d'ús necessaris.
  - c. Sprint 4:
    - i. Muntar APIs al Backend.
  - d. Sprint 5:

- i. Crear projecte Angular i definir el disseny visual de l'aplicació web.
- ii. Implementar cas d'ús de login i sistema d'acompliment de les restriccions per actors.
- iii. Crear les següents pantalles:
  - a) Pantalla principal.
  - b) Llistat de lligues.
  - c) Llistat d'equips, grups i camps/localitzacions.
  - d) Classificació, equips i partits d'un grup.

e. Sprint 6:

- i. Continuació amb la creació de pantalles.
  - a) Assignació dels àrbitres als partits.
  - b) Llista de partits assignats a un àrbitre.
  - c) Modificació del resultat d'un partit.
  - d) Afegir a les pantalles de llistes botons per afegir i eliminar noves dades.

3. Desplegament del projecte i testatge end-to-end.

a. Sprint 7:

- i. Fer desplegament de l'aplicació web.
- ii. Testatge final.

Aquesta planificació en sprints es pot dividir en dos grans parts: backend i frontend. Backend compren base de dades, el servidor i l'API. El frontend compren l'aplicació web que usa aquest backend.

Per poder desenvolupar el frontend, primer cal tenir el backend funcionant, per això s'ha escollit aquest ordre d'execució.

Un cop el backend està desenvolupat, testejat i preparat per rebre peticions, s'inicia el frontend. Primer es fixa un disseny general que tindrà tota la web. A continuació, es desenvolupen les diferents pantalles i peticions que aniran al backend per demanar dades o manar accions com assignar un àrbitre a un partit.

Finalment, quan ja es té el frontend desenvolupat, hi ha un últim sprint on es fa el desplegament de l'aplicatiu, el seu testatge final i comprovació del disseny per assegurar que tot funciona correctament.

A l'annex 1 es pot veure un diagrama de Gantt de les diferents fases.

## 8 DESENVOLUPAMENT

Tot el desenvolupament del projecte s'ha fet seguint la planificació explicada anteriorment.

### Sprint 1

#### Planificació

El primer pas d'aquest primer sprint és definir la planificació de tot el projecte.

### Definició de requeriments i Casos d'ús

El segon pas del desenvolupament del projecte és fer una anàlisi i el disseny. Aquesta tasca engloba la generació de la llista de requeriments tant funcionals com no funcionals.

Un cop s'ha generat aquesta llista de requeriments, s'extreuen els diferents casos d'ús de cadascun d'ells.

### Disseny de l'arquitectura del sistema

A continuació, es fa la definició del disseny de l'arquitectura del sistema. Com s'ha comentat anteriorment, hi ha tres parts principals, cadascuna amb la seva tecnologia escollida:

- Base de dades: MySQL.
- Backend: Django.
- Frontend: Angular.

Per tant, el disseny es basa en aquesta divisió en tres parts que es comuniquen entre si.

Aquest disseny facilita la separació de responsabilitats i la modularitat, la qual cosa permet que cada component es desenvolupi, provi i mantingui de manera independent. Això simplifica el procés de desenvolupament i facilita la incorporació de noves funcionalitats i millores en el futur.

A més, està dissenyat tenint en compte aspectes d'escalabilitat i rendiment. Es poden emprar tècniques com la implementació en múltiples servidors, la utilització de sistemes d'emmagatzematge en caixet i l'optimització de consultes a la base de dades per a garantir que l'aplicació pugui manejar una càrrega d'usuaris i dades en constant creixement.

### Sprint 2

Aquest sprint es centra exclusivament en la base de dades, el seu disseny i implementació.

### Disseny de la base de dades

La base de dades està dissenyada per a emmagatzemar i gestionar la informació relacionada amb els consells escolars d'esports, incloent-hi dades sobre instal·lacions, partits, lligues i grups. La base de dades es compon de les següents taules amb les seves relacions:

- User.
  - Role.
  - Profile.
    - 1 a 1 amb la taula "User".
    - N a N amb la taula "Role".
- League.
- Group.
  - 1 a N amb la taula "League".
- GroupTeam.
  - 1 a N amb la taula "Team".
  - 1 a N amb la taula "Group".
- Club.
- Team.
  - 1 a N amb la taula "Club".
- Location.
- Member.
  - 1 a N amb la taula "Team".
- Match.
  - 1 a N amb la taula "Group".
  - 1 a N amb la taula "Profile".

- 1 a N amb la taula "GroupTeam". Dos cops, una per cada equip que juga el partit.
- 1 a N amb la taula "Location".
  - MatchDay.

A l'annex 2 es pot observar un diagrama Entitat-relació de l'estructura de la base de dades i també, els atributs que té cada taula.

### Desenvolupament de la base de dades

Un cop ja es té el disseny de la base de dades, es comença la seva implementació. Primer es crea un contenidor Docker utilitzant la imatge de MySQL.

A continuació, cal connectar-se amb un client per crear dins el contenidor la nostra base de dades. En el nostre cas el client és el Django, ja que, aquest pot actuar com a tal i crear la base de dades.

### Sprints 3 i 4

Aquests dos sprints són un la continuació de l'altre. La feina a fer consisteix en la següent:

1. Crear el projecte backend de Django i aplicar la base de dades en models.
2. Crear les funcions d'accés a la base de dades per agafar/afegir/modificar/eliminar dades.
3. Crear les APIs necessàries per poder-se comunicar en un futur amb el frontend.
4. Crear els casos d'ús necessaris.

Primer es fa estudi exhaustiu de Django i les seves funcionalitats. Es comprova que Django té un sistema gaire senzill per crear bases de dades MySQL. Només cal crear els models (classes) que vols que siguin les taules i utilitzar un parell de comandes en el terminal. Automàticament, Django crea la bases de dades i afegeix les taules tal i com han estat definides als models. Però aquests models han d'estar organitzats d'una manera específica. Django funciona amb aplicacions, es crea una aplicació general que comprèn tot el projecte Backend. Dins es creen diferents aplicacions que representen cada mòdul funcional, com usuaris, partits, lligues, etc. Dins de cada aplicació es creen els models corresponents amb els seus atributs, les seves relacions. Amb això ja es té el punt 1.

Per desenvolupar els punts 2 i 3, s'ha d'utilitzar una llibreria anomenada Django Rest Framework. S'utilitza per crear APIs CRUD. Amb molt poques línies de codi es poden crear aquestes APIs però s'han d'afegir a les aplicacions que pertoqui. També a cada aplicació cal definir la URL que activarà cadascuna d'aquestes APIs. Per tant, en el moment que es creen les APIs per la connexió en un futur amb el Frontend, també s'està completant el punt 2.

Finalment, la majoria de casos d'ús que inclouen Backend ja estan completats quan s'ha desenvolupat els punts 2 i 3, menys un: el cas d'ús de login.

Aquest cas d'ús és diferent a tota la resta, per tant, aquest cal desenvolupar-lo per separat. Aquest fet fa que es trigui més temps en completar el punt. Pel seu desenvolupament no s'ha utilitzat Django Rest Framework en aquest cas, ja que, per fer login només cal un tipus de petició (no és necessari CRUD), per tant, amb Django bàsic es pot fer

fàcilment.

Com a extra, s'implementa en el Backend un sistema d'autenticació per les APIs. A cada petició que necessiti tenir un usuari iniciat, s'ha d'enviar un "token" generat pel Django que autentifica l'autoria d'aquella petició.

## Sprint 5

A l'inici d'aquest sprint estem en un punt que el backend i la base de dades ja estan completament desenvolupats.

Aquest sprint es centra ja en el Frontend, la feina estipulada per aquest sprint és la següent:

1. Crear projecte Angular i definir el disseny visual de l'aplicació web.
2. Implementar cas d'ús de login i sistema d'acompliment de les restriccions per actors/rols.
3. Crear les següents pantalles:
  - A. Pantalla principal.
  - B. Llistat de lligues, equips, grups i camps/localitzacions.
  - C. Classificació, equips i partits d'un grup d'una lliga.

Un cop es té clar la feina que cal desenvolupar, es fa altre cop un estudi exhaustiu. Es fa per tenir perfectament clar que cal desenvolupar, de quina forma volem fer-ho i recalcar l'estimació de temps del sprint per comprovar si és correcta.

En aquest estudi es troben certs inconvenients en la base de dades. Aquests canvis són com crear una nova taula intermèdia entre altres dues o eliminar una relació, ja que no és necessària. Un cop corregits aquests petits errors, ja podem passar al desenvolupament de la feina d'aquest sprint.

El procés de creació d'un projecte angular és gaire ràpid i senzill usant un parell de comandes en el terminal, així que el punt 1 ja està completat.

A continuació, cal definir el disseny visual de tota l'aplicació web. Enlloc de fer un disseny de zero personalitzat, Angular té disponibles centenars de llibreries de disseny amb un munt d'estils diferents. A més, té llibreries de components que també tenen el seu disseny personalitzat. Per aquest projecte s'ha escollit les següents llibreries pel disseny i els components:

- Bootstrap: llibreria específicament per disseny d'aplicacions web. Té un munt d'estils diferents agradables a la vista.
- Angular Material: és una llibreria de components (llistes, taules, pestanyes, etc.) molt útil. Facilita gaire fer pantalles complexes com llistes personalitzades o taules amb columnes que contenen dades amb diferents formats, etc. També té els seus estils únics agradables a la vista.

La unió d'aquestes 2 llibreries fa un disseny visual agradable a la vegada que senzill, a més, ajuden gaire en el desenvolupament de la mateixa aplicació facilitant i simplificant el codi.

Seguidament, es comença el procés de desenvolupament de les pantalles. Cada pantalla és un component d'angular amb una URL específica. La pantalla principal és el component pare. Aquest té una barra de navegació a la part

superior i tot l'espai restant és el que usaran la resta de components per mostrar la informació.

Aquesta barra de navegació té a l'esquerra els botons per anar a les pantalles de llistes i els botons per pantalles específiques desenvolupades en el següent sprint. A la dreta té els botons d'iniciar sessió o tancar sessió (apareix un o l'altre segons si hi ha usuari iniciat o no).

Després de finalitzar la pantalla principal, venen les pantalles de llistes. Aquestes pantalles són gaire semblants entre si. Usant els components de "Angular Material" es fa primer uns filtres per les dades que són diferents segons la informació que contingui a la llista. Just a sota dels filtres hi ha una llista personalitzada per mostrar les dades que pertocui en cada cas. En finalitzar aquestes pantalles ja es té els punts A i B del punt 3.

Pel punt C cal crear una nova pantalla. Aquesta pantalla depèn de dades específiques, del grup i lliga seleccionats. Per tant aquesta pantalla es mostra quan en la llista de lligues es fa click en una d'elles i llavors, ens mostra la pantalla anomenada "detall de lliga". Un cop estem a la pantalla, es pot observar un selector que té com a opcions els grups relacionats amb la lliga seleccionada. A sota del selector hi ha 3 pestanyes: classificació, llista de partits i llista d'equips; totes aquestes dades sobre el grup seleccionat. Amb aquesta pantalla ja es té completades les pantalles d'aquest sprint.

Per finalitzar la feina només falta el punt 2. En el frontend el login és gaire senzill, ja que, només cal implementar la petició enviant usuari i contrasenya i modelar un comportament segons la resposta rebuda. En cas que les credencials siguin correctes, es rep el "token" de validació necessari per a les peticions i també el tipus d'actor que és l'usuari. En cas que siguin incorrectes, es rep un missatge dient-ho.

Per la segona part del punt 2, que consisteix en les restriccions per actors/rols, cal mirar primer si hi ha usuari iniciat; en cas que no hi hagi, cap de les funcionalitats que necessiten autenticació hi apareixen; en cas que si hi hagi, s'ha de mirar el rol que tingui i, llavors mostrar les funcionalitats que pertocuin. Aquestes funcionalitats són les pantalles específiques que hi ha a la barra de navegació.

## Sprint 6

Aquest sprint és la continuació directa de l'anterior. La feina a fer és la següent:

1. Pantalla per assignar àrbitres als partits.
2. Llistat de partits assignats a l'àrbitre que hagi iniciat sessió a l'aplicació.
3. Modificació del resultat d'un partit per part de l'àrbitre.
4. Modificació de les dades de la base de dades per part d'un administrador.

Aquestes pantalles són aquelles per gestionar les lligues per part de treballadors del consell.

Primer hi ha el punt 1, la pantalla per assignar àrbitres als partits. És una pantalla on hi ha dos selectors, un per lliga i un per grup; apareixen els partits de les dades seleccionades en una llista, i aquests tenen un botó per modificar-los i així poder assignar àrbitre, a més de poder canviar

més dades com la data o la localització.

A continuació, hi ha els punts 2 i 3 que són el llistat de partits assignats a un àrbitre i la modificació de resultat del partit. En cas que l'usuari amb sessió iniciada tingui el rol d'àrbitre, hi ha una pantalla disponible que és una llista dels partits que té assignats i cada un té un botó per introduir el resultat.

Finalment, el punt 4, modificació de dades per part d'un administrador. En cas que l'usuari amb sessió iniciada tingui el rol d'administrador, a les pantalles de llistes de dades, apareix al costat dels filtres un botó verd per poder afegir noves dades segons la llista que sigui. També a la pròpia llista, cada fila que correspon a una dada, té un botó vermell que és per eliminar-la. Així un administrador pot editar dades des de l'aplicació web.

## Sprint 7

Aquest sprint és l'últim del desenvolupament. Arribats aquest sprint, la feina marcada en un inici tant per la base de dades com pel backend com pel frontend ja està finalitzada.

Aquest sprint consisteix en:

- Fer desplegament de l'aplicació web.
- Testatge final.

El desplegament s'ha fet en una xarxa local, s'ha usat un ordinador antic com a servidor, s'ha disposat en la xarxa local i, per tant, l'aplicació web ara és accessible des de qualsevol altre dispositiu que estigui connectat a la xarxa.

S'ha fet el desplegament així, ja que, és la representació més semblant en local que es pot fer a una situació real com seria per exemple utilitzar els serveis de Amazon Web Services o Google per tenir l'aplicació web desplegada.

Finalment, s'ha donat a diferents persones de diferents rangs d'edat accés a l'aplicació web i s'ha demanat que fessin certes accions.

Després, s'ha recollit el feedback rebut per part d'aquestes persones, s'ha estudiat i s'ha fet una petita llista de canvis visuals que ha semblat necessari fer. Canvis com canviar el color d'alguns filtres ja que no complia amb el disseny visual general, o a les pantalles de llistes moure una mica la llista i els filtres, ja que, estava massa junt i no era agradable a la vista, etc.

Finalment, s'han aplicat aquests canvis senzills i això finalitza el procés de desenvolupament del projecte.

## 9 RESULTATS

Després de la finalització del meu treball de final de grau, el resultat obtingut és una aplicació web integral per a la gestió d'un consell escolar d'esports. Aquesta aplicació proporciona una solució unificada que combina les funcionalitats necessàries per a la visualització de dades per part dels usuaris finals i la gestió interna del consell.

L'aplicació ha estat desenvolupada utilitzant MySQL per a la gestió de la base de dades, Django com a backend i Angular per al frontend. La elecció d'aquestes tecnologies no va ser casual. MySQL, amb la seva gran capacitat per manejar grans volums de dades, s'adapta perfectament a la necessitat d'emmagatzemar i recuperar la gran quantitat d'informació relacionada amb el consell escolar. Django, un

framework de Python, es va utilitzar per desenvolupar el backend, proporcionant una base sòlida per a la lògica de negoci i la interacció amb la base de dades. Finalment, Angular, amb el seu enfocament basat en components i el seu àmpli ecosistema, es va seleccionar per a la construcció d'un frontend interactiu i dinàmic.

La interfície d'usuari, construïda amb Angular, és intuïtiva i amigable per als usuaris. Les opcions de navegació permeten als usuaris explorar diferents àrees. Cadascuna d'aquestes seccions presenta dades rellevants d'una manera fàcil de llegir, permetent als usuaris obtenir la informació que necessiten ràpidament i sense complicacions.

Una característica important de l'aplicació és el seu sistema d'actors. Depenent de l'actor assignat a un usuari registrat, se li atorguen diferents nivells d'accés a les funcionalitats de l'aplicació. Aquesta característica permet una gestió eficient i segura de la informació.

Finalment, però no menys important, s'ha posat un gran èmfasi en l'optimització del rendiment i la seguretat de l'aplicació. Les consultes a la base de dades s'han optimitzat per garantir temps de càrrega ràpids, i s'han implementat mesures de seguretat per protegir la integritat de les dades i la privacitat dels usuaris.

En conclusió, el resultat d'aquest treball de final de grau és una aplicació web completa i funcional que proporciona una solució eficient i efectiva per a la gestió d'un consell escolar d'esports. Aquest projecte ha demostrat ser una oportunitat invaluable per aplicar i consolidar les meves habilitats en desenvolupament web utilitzant algunes de les tecnologies més populars i potents disponibles avui dia.

## 10 CONCLUSIONS

Els objectius d'aquest treball final de grau són:

1. Desenvolupar una aplicació web funcional per gestionar un consell escolar d'esports.
2. Centralització.
3. Aplicar rols d'usuari per a dividir les tasques segons el rol que es tingui.

Aquests tres objectius principals han estat la base per crear tota la planificació, per tant, en el moment que ja s'ha completat el desenvolupament seguint la planificació, es compleixen aquests objectius. Primer, s'ha desenvolupat una aplicació web funcional enfocada a les necessitats d'un consell escolar.

A continuació, s'ha creat tot en una sola aplicació. Aquest objectiu és molt important ja que avui en dia els consells escolars utilitzen una web senzilla per a visualitzar la informació de les lligues i partits, i una altra web separada per a la gestió d'aquestes dades. Amb la nostra aplicació web, aquesta necessitat de tenir dues aplicacions web separades desapareixeria.

Finalment, s'ha implantat un sistema de rols d'usuari que, segons el rol o rols assignats, permet l'accés a certes funcionalitats. En un inici del projecte es van establir uns requeriments i uns casos d'ús basats en aquests requeriments. Tots aquests requeriments s'han satisfet amb èxit. Cada requisit constava de dues parts per a la seva realització: una part de frontend i una part de backend.

Pel que fa al backend, utilitzant Django i MySQL, s'ha

pogut dissenyar i desenvolupar un software per gestionar la lògica de negoci i l'accés a la base de dades. Primerament, es va crear el diagrama de la base de dades i després es va implementar utilitzant els models de Django. A més, s'han creat APIs per a la comunicació externa amb aquest backend.

Pel que fa al frontend, s'ha dissenyat una interfície d'usuari senzilla però fàcil d'utilitzar, és a dir, una aplicació web usable i interactiva. S'han desenvolupat components personalitzats per a les pantalles i s'ha utilitzat el routing d'Angular per a una navegació fluida entre les pantalles.

Durant el procés de desenvolupament, s'han realitzat proves per validar els requeriments inicials. S'ha comprovat que el frontend i el backend s'intercomuniquen correctament, que les dades s'envien i es recuperen amb precisió i que les operacions es completen de manera eficient.

Una de les mesures de seguretat implementades en l'aplicació web és l'ús de tokens d'autenticació per a les sol·licituds que requereixen una sessió iniciada. Cada usuari creat té un token associat que és necessari per a realitzar aquestes sol·licituds. Si un usuari intenta fer una sol·licitud sense enviar el token d'autenticació corresponent, la sol·licitud serà rebutjada. A més, gràcies a Django, l'atac conegut com a "SQL Injection" no és possible ja que les dades introduïdes per l'usuari i enviades en la sol·licitud no s'envien directament a SQL i s'executen, sinó que primer passen per Django, que les controla.

Aquesta capa de seguretat basada en tokens d'autenticació protegeix les sol·licituds que requereixen una sessió iniciada, assegurant que només els usuaris autenticats i autoritzats puguin realitzar operacions crítiques. En canvi, les sol·licituds que no requereixen autenticació no tenen accés a informació sensible ni a operacions crítiques.

En conclusió, s'han utilitzat tecnologies noves en el frontend i el backend del projecte. Això m'ha permès aprendre molt sobre Angular i Django. També he recordat com funciona SQL, específicament MySQL, i com dissenyar eficientment una base de dades per a aquesta tecnologia.

En el frontend, utilitzant Angular, he après considerablement sobre la seva estructura de components i sobre diverses llibreries àmpliament utilitzades com Angular Material (components) i Bootstrap (disseny visual / CSS).

En el backend, amb Django, he après molt sobre el seu funcionament. Django és un framework ben pensat, ja que només definint els models i les seves relacions, automàticament crea la base de dades en la tecnologia compatible desitjada, en el meu cas, MySQL. Per tant, en cap moment ha estat necessari crear la base de dades a part i les seves funcions, ja que Django s'encarrega de les comunicacions amb la base de dades. També he après com crear APIs CRUD amb Django i com personalitzar-les segons les necessitats, cosa que és extremadament fàcil. Finalment, he après com implementar sistemes de seguretat com els tokens esmentats anteriorment.

Durant el procés de desenvolupament, s'han produït alguns conflictes, com incompatibilitats de llibreries necessàries o algunes estimacions de temps incorrectes per a certs requeriments, el que ha retardat el procés en relació amb la

planificació inicial. No obstant això, realitzant una bona tasca d'investigació i cerca en línia, s'han resolt les incompatibilitats trobant altres llibreries compatibles que realitzaven la mateixa funció. Quant a les estimacions, s'han solucionat redistribuint la càrrega de treball entre els sprints (sense afectar la data de finalització del projecte).

La metodologia utilitzada en aquest projecte ha estat Scrum. És un marc de treball àmpliament utilitzat arreu del món i fàcil d'implementar en un projecte de programari com aquest. Considero que ha estat la millor opció, ja que, a més de haver-la utilitzat abans i saber com funciona, és una metodologia molt còmoda i adaptable. Scrum m'ha ajudat a mantenir un enfocament iteratiu, gestionar eficaçment el temps, saber què cal desenvolupar en cada moment per complir amb la planificació inicial i adaptar-me als canvis que han estat necessaris durant el projecte, com ara inconvenients que feien perdre massa temps i alentien el desenvolupament o noves funcionalitats o pantalles del frontend que no s'havien comentat inicialment en la planificació.

Com s'ha comentat anteriorment, avui en dia les aplicacions web utilitzades en l'àmbit dels consells escolars d'esports tenen dues aplicacions web separades: una per als usuaris no registrats per veure els resultats dels partits i classificacions dels equips, i una altra per a la gestió d'aquestes dades i la introducció dels resultats, assignació d'àrbitres, etc. Aquest projecte ha dissenyat i desenvolupat una única aplicació que combina les funcionalitats de les dues aplicacions web utilitzades actualment. Per tant, els consells escolars només haurien de pagar pel manteniment d'una sola aplicació web, reduint així els costos de desenvolupament i manteniment a llarg termini.

Finalment, ha estat un projecte molt complet, educatiu i desafiant a nivell personal. S'han aconseguit complir tots els objectius establerts inicialment, superant els obstacles malgrat una data de finalització ajustada. Cada minut invertit en aquest projecte ha contribuït a la meua formació personal, des de realitzar un bon estudi del projecte i una planificació adequada, fins a treballar en diferents aspectes del desenvolupament, com el disseny de la base de dades o l'aprenentatge de l'ús d'Angular i Django. També m'ha ajudat a afrontar canvis i problemes, trobant les millors solucions perquè no afectin la planificació global del projecte.

En definitiva, aquest projecte m'ha beneficiat en tots els àmbits possibles: com a desenvolupador, com a enginyer de programari per a una bona planificació i com a "Project Manager" o, en aquest cas, "Scrum Master" per gestionar de manera adequada un projecte, el temps i els recursos disponibles.

## BIBLIOGRAFIA

- [1] *Consell Sabadell*. Obtenido de <http://www.consellsabadell.cat/>
- [2] *Consells Esportius de Catalunya*. Obtenido de <https://ceeb.cat/es/>
- [3] *Consell Escolar Terrassa*. Obtenido de <https://ceterrassa.cat/>
- [4] *Frontend i Backend*. Obtenido de <https://platzi.com/blog/que-es-frontend-y-backend/>
- [5] *HTML*. Obtenido de <https://developer.mozilla.org/es/docs/Web/HTML>
- [6] *CSS*. Obtenido de <https://developer.mozilla.org/es/docs/Web/CSS>
- [7] *JavaScript*. Obtenido de <https://developer.mozilla.org/es/docs/Web/JavaScript>
- [8] *Python*. Obtenido de <https://es.wikipedia.org/wiki/Python>
- [9] *MySQL*. Obtenido de <https://es.wikipedia.org/wiki/MySQL>
- [10] *Django*. Obtenido de <https://www.djangoproject.com/>
- [11] *Angular*. Obtenido de <https://angular.io/>
- [12] *BDs relacionals*. Obtenido de <https://cloud.google.com/learn/what-is-a-relational-database?hl=es-419>
- [13] *SQL*. Obtenido de <https://es.wikipedia.org/wiki/SQL>
- [14] *MVC*. Obtenido de <https://developer.mozilla.org/es/docs/Glossary/MVC>
- [15] *TypeScript*. Obtenido de <https://www.typescriptlang.org/>
- [16] *SPA*. Obtenido de <https://digital55.com/blog/que-son-single-page-application-spa-desarrollo-elegido-por-gmail-linkedin/>
- [17] *Stakeholders*. Obtenido de <https://rockcontent.com/es/blog/que-es-un-stakeholder/>
- [18] *UI-UX*. Obtenido de <https://rockcontent.com/es/blog/ui-ux/>
- [19] *API RestFul*. Obtenido de <https://aws.amazon.com/es/what-is/restful-api/>
- [20] *Diagrama E-R*. Obtenido de <https://www.lucidchart.com/pages/es/que-es-un-diagrama-entidad-relacion>
- [21] *Docker*. Obtenido de <https://aws.amazon.com/es/docker/>
- [22] *Casos d'ús*. Obtenido de <https://www.ibm.com/docs/es/elms/elm/6.0.3?topic=requirements-defining-use-cases>
- [23] *Django Rest Framework*. Obtenido de <https://www.django-rest-framework.org/>
- [24] *GIT*. Obtenido de <https://git-scm.com/>
- [25] *CRUD*. Obtenido de <https://es.wikipedia.org/wiki/CRUD>

- 
- E-mail de contacte: 1562889@uab.cat
  - Menció realitzada: Enginyeria del Software
  - Treball tutoritzat per: Victoria Montenegro Ruíz (departament)
  - Curs 2022/23

# ANNEX

1.

	27 de Febrer	6 de Març	13 de Març	20 de Març	27 de Març	3 d'Abril	10 d'Abril	17 d'Abril	24 d'Abril	1 de Maig	8 de Maig	15 de Maig	22 de Maig	29 de Maig
Fase de definició														
Fase d'implementació														
Fases de desplegament del projecte														
Fases de testatge end-to-end														

La data és la del primer dia de la setmana.

2.

