



DOCUMENTO FUNCIONAL DEL PROYECTO

ALUMNO/GRUPO: Daniel Pariente Garcia y Alex Rubio Rosa

2n SMX C

1. Introducción y contexto

Objectiu: SafeGoalStats és una aplicació web centrada en el futbol, especialment en la Lliga Espanyola, amb l'objectiu de proporcionar estadístiques detallades i segures a aficionats i analistes.

- **Problema o necessitat:** Els usuaris actualment han de buscar informació dispersa sobre partits, jugadors i estadístiques. No hi ha una eina centralitzada que ofereixi dades fiables i segures.
- **Usuari final:** Aficionats al futbol, analistes esportius i usuaris interessats en estadístiques de la Lliga Espanyola. També estudiants de SMX que vulguin aprendre com es fa una aplicació web completa.
- **Solució proposada:** Desenvolupar una aplicació web que mostri estadístiques actualitzades de LaLiga (classificació en temps real i màxims golejadors) mitjançant una API externa (API-Football), permeti consultar dades de manera segura. Tota la gestió d'usuaris (registre i login) serà segura, amb contrasenyes xifrades (bcrypt) i autenticació mitjançant JWT.

2. Análisis de requisitos

2.1. Requisitos funcionales (RF)

Código	Descripción del requisito funcional
RF1	Permetre registrar nous usuaris amb email i contrasenya.
RF2	Iniciar sessió amb credencials (email i contrasenya) i gestionar el perfil d'usuari (tancar sessió).

RF3	Consultar la classificació de LaLiga en temps real (posició, equip, punts, partits jugats, victòries, empats, derrotes, gols a favor, gols en contra, diferència).
RF4	Consultar els màxims golejadors de LaLiga (nom del jugador, equip, nombre de gols).
RF5	Tancar sessió (logout).

2.2. Requisitos no funcionales (RNF)

Código	Descripción del requisito no funcional
RNF1	La interfície serà accessible des de dispositius mòbils i ordinadors (disseny responsive).
RNF2	Les pàgines es carregaran en menys de 3 segons per a contingut estàndard.
RNF3	Les dades dels usuaris estaran protegides amb contrasenyes xifrades (bcrypt).
RNF4	Compatible amb navegadors moderns (Chrome, Firefox, Edge).
RNF5	La connexió amb l'API externa tindrà un sistema de cache per no superar el límit de 100 peticions/dia del pla gratuït.
RNF6	Còpies de seguretat periòdiques de la base de dades (manuals, cada setmana).

2.3. Restricciones

- **Tecnologies obligatòries:** Frontend amb HTML, CSS i JS; Backend amb Node js + Express; Base de dades SQLite, API Externa: API-Football
- **Recursos disponibles:** Temps: 8 mesos; Equip: 2 alumnes; Material: ordinadors i accés a internet.
- **Limitacions tècniques:** L'API-Football té un límit de 100 peticions per dia en el pla gratuït, la qual cosa requereix implementar un sistema de cache.
 - La documentació de l'API està en anglès i és tècnicament complexa, la qual cosa va dificultar la integració inicial.

- Node.js és asíncron per defecte, la qual cosa va requerir aprendre a gestionar promeses i async/await.
- SQLite no és tan escalable com MySQL, però per a un projecte educatiu és suficient.

3. Análisis de usuarios y roles

Objetivo: identificar quién usará el sistema y qué podrá hacer.

Describe los distintos tipos de usuario, sus necesidades y sus permisos.

Rol	Descripción	Permisos principales
Administrador	Gestiona usuaris i dades de l'aplicació	Alta, baixa i modificació de dades
Usuari registrat	Usuari que ha creat un compte i ha iniciat sessió	Consultar classificació i golejadors, tancar sessió
Visitant	Usuari que no s'ha registrat o no ha iniciat sessió	Sense permisos

4. Casos de uso / Escenarios de uso

Objetivo: mostrar cómo interactúan los usuarios con el sistema.

Selecciona de tres a cinco casos principales y descríbelos brevemente.

Codi	Nom del cas	Actor principal	Descripció	Resultat esperat
CU1	Registrar usuari	Visitant	L'usuari introdueix email i contrasenya en un modal i es crea un compte.	Usuari registrat correctament a SQLite (contrasenya xifrada)
CU2	Iniciar sessió	Usuari registrat	Introduir credencials (email, contrasenya) i accedir a l'aplicació.	Accés concedit, es genera un JWT i es guarda a localStorage
CU3	Consultar classificació de LaLiga	Usuari registrat	Accedir a la secció de classificació i veure la taula amb posicions, equips, punts, etc.	Classificació mostrada correctament amb dades reals de l'API

CU4	Consultar màxims golejadors	Usuari registrat	Accedir a la secció de golejadors i veure la llista de jugadors amb els seus gols.	Golejadors mostrats correctament amb dades reals de l'API
CU5	Tancar sessió	Usuari registrat	Fer clic a "Tancar sessió" al menú.	El JWT s'elimina de localStorage i l'usuari torna a ser visitant

5. Modelo de datos o estructura de la información

Entitats principals:

- **Usuari:** email, contrasenya (xifrada), data de creació
- **Opinió:** text de l'opinió, ID de l'usuari que l'escriu, data de creació
- **Classificació:** no es guarda a la base de dades, es consumeix de l'API externa
- **Golejadors:** no es guarden a la base de dades, es consumeixen de l'API externa

6. Diseño de la interfaz

Pantalla	Funcionalitat principal	Casos d'ús relacionats
Pantalla principal	Presentació del projecte, menú de navegació	-
Modal de registre	Registrar nou usuari	CU1
Modal de login	Iniciar sessió (finestra emergent)	CU2
Secció de classificació	Veure classificació de LaLiga en temps real	CU3
Secció de golejadors	Veure màxims golejadors de LaLiga	CU4

Observacions sobre el disseny:

- S'ha utilitzat CSS amb Flexbox i Grid per a la maquetació.
 - S'han implementat media queries per a la responsivitat (es veu bé a mòbil i ordinador).
 - Els formularis de registre i login són modals (finestres emergents) per no carregar pàgines noves.
 - El menú de navegació està sempre visible a la part superior.
-

7. Planificación técnica

- **Llenguatges i frameworks:** Frontend: HTML, CSS, JavaScript (pur), Backend: Node.js amb Express
 - **Base de dades:** SQLite
 - **Eines de disseny:**
 - **Disseny de wireframes:** Figma
 - **Gestió de tasques:** Trello
 - **Control de versions:** GitHub
 - **Editor de codi:** Visual Studio Code
 - **Repartiment de tasques:**
 - **Alex:** Backend (Node.js, Express, SQLite, bcrypt, JWT), connexió amb API-Football (classificació i golejadors), rutes d'opinions.
 - **Daniel:** Frontend (HTML, CSS, JavaScript, modals, fetch), disseny a Figma, maquetació, estils responsius, menú de navegació
 - **Cronograma:** Inclou fases: investigació, disseny, desenvolupament, memòria, defensa.
-

8. Análisis de riesgos

L'objectiu d'aquesta secció és identificar els possibles problemes que poden aparèixer durant el desenvolupament del projecte SafeGoalStats i establir estratègies per minimitzar-los.

8.1. Identificación de riesgos

- **Falta de temps o mala planificació:** Riscos derivats de no distribuir correctament les tasques entre els membres del grup.

- **Problemes tècnics o incompatibilitats:** Possibles errors amb Node js + Express, integració amb la base de dades o APIs externes.
- **Pèrdua de dades:** Fallades de la base de dades o errors humans que poden fer perdre informació important.
- **Canvis en els requisits:** Modificacions en les necessitats o objectius que poden afectar el desenvolupament.

8.2. Valoración y respuesta

Clasifica cada riesgo según su probabilidad e impacto, e indica cómo se mitigará.

Risc	Probabilitat	Impacte	Pla de prevenció o contingència
Falta de temps	Alta	Alta	Dividir tasques i establir lliuraments intermedis; revisió setmanal del progrés.
Problemes tècnics	Mitjana	Mitjà	Provar tecnologies abans de començar; documentar errors i solucions.
Pèrdua de dades	Baixa	Alta	Fer còpies de seguretat setmanals del repositori i base de dades; utilitzar Git.

9. Validación y criterios de éxito

- **Criteris d'acceptació:**
 - Registre i login funcionals amb contrasenyes xifrades (bcrypt) i generació de JWT.
 - Classificació de LaLiga mostrada correctament amb dades reals de l'API-Football.
 - Màxims golejadors de LaLiga mostrats correctament amb dades reals de l'API-Football.
 - Tancament de sessió funcional..
- **Proves previstes:**
 - Proves d'autenticació (registre, login, accés a rutes protegides, tancament de sessió).
 - Proves de connexió amb l'API externa (classificació i golejadors).
 - Proves de seguretat (SQL injection, contrasenyes xifrades, validació de formularis).
 - Proves de compatibilitat amb navegadors (Chrome, Firefox, Edge).
 - Proves de responsivitat (Chrome DevTools mode mòbil).

- **Indicadors de qualitat:**
 - Temps de resposta de l'aplicació < 3 segons (amb cache de l'API).
 - Cap error crític durant les proves d'autenticació.
 - Contrasenyes xifrades correctament a SQLite (no es veuen en clar).
 - Interfície responsive (es veu bé a mòbil i ordinador).
-

10. Conclusión

SafeGoalStats és un projecte que compleix els objectius marcats a l'inici: permet centralitzar estadístiques de la Lliga Espanyola (classificació i màxims golejadors) mitjançant una API externa real, gestiona usuaris de manera segura (registre, login, contrasenyes xifrades, JWT), i inclou un sistema d'opinions que fomenta la participació dels usuaris.

Les tecnologies utilitzades (Node.js, Express, SQLite, HTML, CSS, JavaScript) han estat adequades per al nostre nivell de SMX, tot i que inicialment havíem considerat Rust. La integració amb l'API-Football va ser la part més difícil del projecte, però després de moltes proves i errors, vam aconseguir que funcionés correctament.

El projecte aporta valor als aficionats a la LaLiga (amb dades fiables i actualitzades) i també als estudiants de SMX (com a exemple d'aplicació web completa amb backend, frontend, base de dades, seguretat i API externa).

Pròxims passos:

- Connectar més endpoints de l'API (resultats de partits en temps real, calendari).
- Millorar el disseny visual (més atractiu, amb imatges).
- Pujar la web a un servidor públic (Render, Vercel o similar).
- Afegir autenticació amb Google (OAuth2).
- Implementar un sistema de valoració (puntuació d'1 a 5 estrelles) a les opinions..

"L'anàlisi funcional proporciona una visió clara i realista del projecte. A partir d'aquí, comença la fase de desenvolupament."