

# Report di TWEB

Giuseppe Marino

29 giugno 2024

## 1 Introduzione

Questo documento presenta tutte le decisioni implementative e tutti gli aspetti tecnici distintivi del progetto "The Pundit Pub", un sito web dedicato al calcio. Esso offre aggiornamenti e news in tempo reale, dati approfonditi su partite, giocatori e competizioni e si propone come un punto di riferimento per gli appassionati del pallone.

## 2 Soluzione del Frontend:

Per quanto concerne il lato frontend, esso è stato sviluppato utilizzando React, una libreria open-source, front-end e JavaScript per la creazione di interfacce utente di proprietà di META, in quanto permette una facile implementazione e gestione del codice nonché una elevata personalizzazione stilistica del sito. Per quanto riguarda lo sviluppo del sito, ho deciso di suddividere le pagine in più livelli, implementando una struttura modulare, organizzando il codice seguendo una gerarchia di componenti, composti da atomi e card. Tale metodologia ha permesso di mantenere il codice ordinato e ottimamente gestibile e, inoltre, cercando di garantire la massima pulizia e coerenza dello stesso, ho definito una directory dedicata solo ed esclusivamente alla gestione delle richieste verso i vari server, assicurando così una chiara separazione fra le parti del progetto e una sua maggiore manutenibilità.

### 2.1 Autenticazione e Gestione della Sessione:

L'autenticazione, fondamentale nelle sezioni dell'applicazione che richiedono l'accesso a dati personali o sensibili, è stata resa in modo tale da garantire la sicurezza dei dati trasmessi.

### 2.2 Implementazione del Login:

Ho sviluppato un sistema di registrazione e autenticazione che non solo facilita l'interazione con la chat, ma offre anche un'esperienza personalizzata agli utenti registrati. Prevediamo di arricchire ulteriormente questa personalizzazione con aggiornamenti automatici e vantaggi esclusivi per gli utenti registrati in futuro. Ciò nonostante, il sito rimane accessibile anche agli utenti non registrati, permettendo loro di consultare le ultime notizie e informazioni sulle partite senza la necessità di un account.

### 2.3 Design e Personalizzazione:

L'implementazione del design visivo è stata fatta mediante l'uso di CSS e componenti personalizzate utilizzando anche le strategie per arricchire l'esperienza utente.

### 2.4 Gestione del Loading:

Approfondimento sull'uso di useMemo per ottimizzare il caricamento delle pagine e garantire un'interfaccia reattiva, inoltre è stato implementato un meccanismo di loading che evita che la pagina sia visualizzata prima che sia effettivamente montata.

## **2.5 Chat e Interazione:**

La pagina è stata progettata con tecnologia WebSocket per fornire un'esperienza di messaggistica in tempo reale. Sono disponibili tre stanze di chat distinte, ognuna dedicata a temi di discussione specifici, per facilitare conversazioni organizzate e mirate. L'accesso alla chat è riservato agli utenti autenticati ed è stato implementato tramite un'icona di chat, facilmente accessibile da qualsiasi parte del sito

## **2.6 Sviluppo del sito:**

Nella homepage del sito sono presenti link alle tre pagine "Games", "Teams" e "Competitions" oltre che due sezioni contenenti le news in evidenza e altre news (vedi punto 5). Per quanto riguarda invece le altre tre pagine e in particolare per "Teams" e "Competitions" ho sviluppato due "sotto pagine", di nome "Single-Team" e "Single-Competition" riguardanti la singola squadra o la singola competizione scelta nelle pagine precedenti. Single-Team presenta oltre al box informativo anche due sottosezioni di nome "players" e "games" che mostrano i giocatori appartenenti alla squadra e le partite della stessa sotto forma di popup cliccabile che porta alla pagina del giocatore o della partita scelta. Per poter visualizzare tutte le informazioni dei giocatori in modo ordinato e senza riempire le pagine delle squadre, ho creato una pagina dedicata di nome "Players" che mostra tutte le informazioni del giocatore e le partite disputate ordinate per data. Cliccando su una qualsiasi partita si apre la pagina "single-game", raggiungibile anche dalla sezione delle partite giocate dalla squadra selezionata, che mostra tutte le informazioni della singola partita, tra cui azioni di gioco e formazioni divise in due sezioni. Ogni singolo nome di ogni singolo giocatore presente in ogni sezione porta, tramite click, alla pagina del giocatore. Le partite sono anche raggiungibili tramite la pagina "Games", la quale presenta un calendario navigabile sul quale, in presenza di partite nel giorno scelto (indicato da un pallino sotto al numero del giorno), è possibile cliccare per vedere tutte le partite giocate in quel giorno.

# **3 Soluzione Persistenza dei Dati e Architettura Server**

## **3.1 Persistenza dei dati:**

Nella progettazione dell'architettura server, ho adottato un approccio bifocale per la persistenza dei dati, utilizzando MongoDB e PostgreSQL per soddisfare le diverse esigenze del sistema. PostgreSQL, con le sue capacità transazionali e il modello di dati relazionale, è stato scelto per gestire dati statici e fortemente strutturati come competizioni, club e informazioni sui giocatori; infatti, la sua affidabilità e integrità referenziale lo rendono ideale per assicurare la consistenza di dati che richiedono relazioni definite e operazioni CRUD rigorose.

D'altra parte, ho utilizzato la flessibilità e la scalabilità di MongoDB per memorizzare dati dinamici e in rapida evoluzione, quali i dettagli delle partite, gli eventi di gioco e le prestazioni dei giocatori nelle partite. La natura schema-less di MongoDB ha permesso di adattarmi facilmente a strutture di dati complesse e in continua evoluzione, ottimizzando le operazioni di lettura/scrittura per dati che presentano variazioni frequenti e che beneficiano di una modellazione meno rigida.

## **3.2 Architettura Server:**

Per orchestrare l'interazione tra il client e i sistemi di gestione dei database, ho implementato una costellazione di server in cui il "Main Server" funge da intermediario, ricevendo le richieste del client e instradandole ai server dedicati. Questi ultimi, configurati specificamente per interagire con MongoDB e PostgreSQL, gestiscono efficientemente le operazioni sui rispettivi database. Questa separazione dei compiti non solo chiarifica la struttura del back-end ma garantisce anche che le operazioni sui dati siano eseguite in modo ottimale, con la possibilità di bilanciare il carico e di scalare orizzontalmente i servizi in base alla domanda.

Ho inoltre definito l'analisi delle strutture server e del loro ruolo nella gestione dei dati persistenti, con particolare enfasi sulle richieste eseguite al database Postgres e al MongoServer.

Ho improntato tutta l'architettura dei server sulla base della modularizzazione del Codice e della segmentazione in moduli, per massimizzare l'efficienza e la manutenibilità, inoltre grazie a queste caratteristiche potrebbe garantire una scalabilità efficiente in caso di ampio traffico di utenti. Per quanto riguarda Springboot ho definito, per ogni tabella del database alla quale ci stiamo riferendo, services, repositories e controllers in modo separato e modulare, per evitare errori e per migliorarne la leggibilità e la possibile manutenibilità in futuro.

Infine, per MongoDB, ho deciso di suddividere in routes i gruppi di dati simili e utilizzare i modelli che si riferiscono ai dati nel database.

## **4 Suddivisione dei compiti e Documentazione**

### **4.1 Suddivisione dei compiti:**

Ho sviluppato in solitaria ogni parte del progetto con l'aiuto, per il debugging e il controllo di alcuni commenti, dell'IA (vedi 6), in particolare ho utilizzato Google Gemini ( ex Bard ) e Chat-GPT 4 versione "o" di OpenAI.

### **4.2 Documentazione:**

Riguardo alla documentazione, mi sono assicurato che il codice fosse accuratamente documentato, descrivendo ogni endpoint, i parametri richiesti e i valori restituiti.

## **5 Extra information:**

Per migliorare l'esperienza dell'utente ho inserito una richiesta ad una API esterna, di proprietà di Google News, che si occupa di news calcistiche. All'interno del progetto viene fornita la chiave, la quale ha un limite giornaliero di 30-40 utilizzi, che permette di ottenere tali news.

## **6 Bibliografica:**

Per quanto riguarda la documentazione del progetto, mi sono avvalso di un'ampia gamma di fonti autorevoli, inclusi siti educativi di riferimento come W3Schools, forum per sviluppatori come Stack Overflow, la documentazione di Socket.io e React nonché la robusta libreria di componenti MUI di Material-UI. In aggiunta, ho utilizzato l'intelligenza artificiale, in particolare Google Gemini e Chat-GPT 4° di OpenAI per il debugging del codice (vedi 4.1), migliorando significativamente la precisione, e Dall-E 3 di Microsoft Copilot per il prompt di alcune immagini presenti nel sito, onde evitare l'utilizzo di altre con diritto d'autore o licensing.

## **7 Conclusione:**

Questo sito web è l'ideale per gli appassionati di calcio. Offre funzionalità come aggiornamenti in tempo reale delle partite, le ultime notizie e spazi di discussione per gli utenti. Ci impegniamo a rendere la nostra applicazione un punto di riferimento quotidiano per gli utenti, garantendo un'esperienza utente coinvolgente e soddisfacente.