

# *Università degli studi di Salerno*

## *Dipartimento di Informatica*

### *Corso di laurea in Informatica*

#### **Ingegneria del software**

# SoccerHub

<b>Docenti:</b>	<b>Nome</b>	<b>Studenti:</b>	<b>Matricola</b>
prof: Carmine Gravino	Gaetano Pascarella		0512117837
prof: Giammaria Giordano	Fabian Andres Scalera Gabriele Armando Scialla Domenico Rago		0512118647 0512116940 0512118479

*Anno accademico: 2024/25*

**Versione 1.0**

## **1. Scopo del sistema - Introduzione**

In un ambiente sportivo sempre più competitivo, avere accesso a informazioni aggiornate e gestire i dati in modo efficiente è essenziale. SoccerHub non solo automatizza molte delle attività amministrative, ma fornisce anche strumenti analitici per monitorare i progressi della squadra, facilitando decisioni più informate. In più, fornisce uno strumento completo e facile da usare che consente ad allenatori, manager e dirigenti sportivi di avere un controllo completo sulla gestione della squadra. Con un'interfaccia intuitiva e integrata con notifiche automatiche, il sistema supporta una gestione coordinata e senza interruzioni delle operazioni sportive, migliorando così le prestazioni complessive della squadra. SoccerHub è una piattaforma software progettata per semplificare e ottimizzare la gestione operativa e tecnica di una squadra professionale. La gestione di una squadra sportiva comporta molteplici attività, dalla pianificazione degli allenamenti alla gestione delle partite, fino al monitoraggio delle prestazioni dei giocatori e alla comunicazione tra lo staff tecnico e i membri del team. L'efficienza con cui queste attività vengono svolte può avere un impatto diretto sulle prestazioni della squadra e sul successo complessivo delle sue operazioni.

## **2. Scopo del sistema - Obiettivi**

SoccerHub intende fornire semplici strumenti di gestione e visualizzazione della squadra, facilitando la comunicazione interna e utilizzando dati analitici per decisioni strategiche. Inoltre, permette ai tifosi di essere a conoscenza delle scelte tecniche e delle novità della squadra, garantisce un incremento della partnership e promuove le innovazioni tecnologiche utilizzate, con l'obiettivo di aumentare la visibilità e il valore del marchio della squadra.

### **3. Scopo del sistema - Funzionalità Principali**

#### **Gestione del profilo giocatori:**

- Creazione e aggiornamento dei profili dei giocatori con informazioni personali, statistiche di prestazione, infortuni e progressi individuali.

#### **Pianificazione di allenamenti e partite:**

- Organizzazione del calendario per allenamenti, partite e altri eventi sportivi.
- Assegnazione di ruoli e convocazioni ai giocatori per ogni evento.

#### **Monitoraggio delle prestazioni:**

- Registrazione delle statistiche di prestazione durante partite e allenamenti, con report dettagli per ogni giocatore.
- Analisi delle prestazioni individuali e di squadra nel tempo.

#### **Dashboard di controllo:**

- Panoramica dello stato dei giocatori, incluse presenze, gli allenamenti pianificati e le prestazioni complessive.

#### **Gestione degli infortuni:**

- Monitoraggio dello stato di salute dei giocatori, con gestione degli infortuni e dei tempi di recupero.

### **4. Data di inizio e di Fine**

Inizio: Ottobre 2024

Fine: prima decade di Febbraio 2025

### **5. Deliverables**

RAD, SDD, ODD, Matrice di Tracciabilità, Test Plan, Test Case Specification,

Test incident Report, Test Summary Report, Manuale D'Uso, Manuale Installazione e ogni altro documento richiesto per lo sviluppo del sistema.

## 6. Vincoli collaborativi e comunicativi

- Uso di sistemi di versioning - GitHub
- Utilizzo di tool per la suddivisione dei task e attività (Trello, ClickUp)
- Utilizzo di tool di comunicazione tracciabile (Slack, Discord)

## 7. Vincoli tecnici

### Analisi e specifica dei requisiti

- Specifica di **minimo 2** e **massimo 4** scenari per ogni membro del team.
- Diagramma degli use.
- Un sequence diagram ogni due membri del team.
- Specifica dei requisiti funzionali e non funzionali relativi a uno use case selezionato.
- Specifica degli oggetti boundary, control e entity per gli use case specificati.
- Un sequence diagram ogni due membri del team
- Specifica di un class diagram per team
- Almeno un activity diagram per sistema esistente o sistema esistente proposto e almeno uno statechart diagram ogni due membri del team.

### System Design

- Specifica dei design goal e analisi trade-off relativi ad almeno due coppie di design goal.
- Definizione dell'architettura del sistema.
- Individuazione di dove poter utilizzare due design pattern , indicandone l'obiettivo e come sarebbero implementati.

### Object Design

- Uso di UML.

### Testing

- Ogni studente dovrà effettuare il testing di unità, tramite category partition, dei casi d'uso specificati.

## **8.      *Criteri di Accettazione/ Acceptance Criteria***

- Utilizzo appropriato di GitHub.
- Adeguato utilizzo del pull-based development.
- Adeguato utilizzo di un tool di comunicazione.
- Adeguato utilizzo di un tool per la gestione dei task.
- Documentazione adeguata.
- Appropriato testing dei casi d'uso

## **9.      *Criteri di premialità***

- Uso adeguato di sistemi di build.
- Uso adeguato di un processo di continuous integration tramite Travis.
- Adozione di processi di code review.
- Uso adeguato di tool avanzati di testing (Mockito, Cobertura, ecc.)