



TeamManager

CORSO DI LAUREA MAGISTRALE IN INGEGNERIA INFORMATICA
2024/2025

INGEGNERIA DEL SOFTWARE

AUTORI :

Antonio Nicolò Scarvaglieri

Giuseppe Ravesi

Vincenzo Venezia

TeamManager	1
1.0 IDEAZIONE E ANALISI DEI REQUISITI	4
1.1 Introduzione	4
1.2 Requisiti	4
1.3 Obiettivi e casi d'uso	6
1.4 Modello dei casi d'uso	8
UC1: GESTISCI ROSA	9
UC2: GESTICI EVENTO, CRUD	10
UC3: GESTISCI DISPONIBILITA, CRUD	12
UC4: ANALIZZA ANDAMENTO ROSA	14
UC5: INVIO NOTIFICHE	15
UC6: GESTISCI PROGRESSI GIOCATORE, CRUD	16
UC7: CREA GIOCATORE, CRUD	17
UC8: SUGGERISCI FORMAZIONE	18
UC9: SUGGERISCI SESSIONE MIRATA	20
UC10: CONFRONTA GIOCATORI	22
1.5 Documento di visione	23
1.6 Regole di dominio	23
1.7 Specifiche Suppletive	25
USABILITA	25
AFFIDABILITA	25
VINCOLI HARDWARE E SOFTWARE	25
ASPETTI LEGALI	26
1.8 Glossario	26
2.0 ANALISI ORIENTATA AGLI OGGETTI	27
2.1 Introduzione	27
ITERAZIONE 1	28
ITERAZIONE 2	28
ITERAZIONE 3	29
ITERAZIONE 4	29

2.2 Modello di dominio	32
2.3 Diagrammi di Sequenza di Sistema e Contratti delle Operazioni	33
3.0 PROGETTAZIONE.....	34
3.1 Diagrammi delle classi	34
3.2 Diagrammi di sequenza	35
4.0 TESTING.....	35
4.1 Introduzione	35
4.2 Definizione delle classi e metodi da testare.....	36
ITERAZIONE 1.....	36
TeamManager	36
Amichevole e Allenamento	36
Calendario	36
Disponibilità	36
Rosa	37
ITERAZIONE 2.....	37
Calendario	37
ITERAZIONE 3.....	37
Calendario	37
ITERAZIONE 4.....	38
5.0 CONCLUSIONI E IMPLEMENTAZIONI FUTURE	38
5.1 Introduzione alla fase finale.....	38
5.2 Test di sistema	39
5.3 Implementazioni future.....	39

1.0 IDEAZIONE E ANALISI DEI REQUISITI

1.1 Introduzione

La fase di ideazione è fondamentale per porre le basi di un progetto solido e ben strutturato. L'obiettivo principale è sviluppare una visione chiara e complessiva dell'idea da realizzare, valutandone la fattibilità in termini di tempi, risorse e complessità tecnica. Questa fase aiuta a prevenire criticità future, garantendo una pianificazione efficace sin dall'inizio.

Per raggiungere una comprensione approfondita dei requisiti e delle esigenze, è stato fondamentale analizzare diversi documenti chiave, tra cui il Modello dei Casi d'Uso, il Documento di Visione, le Specifiche Supplementari e il Glossario. Questi strumenti hanno permesso di definire con precisione gli obiettivi del progetto, delineando le linee guida da seguire e fornendo un quadro strutturato su cui basare le fasi successive dello sviluppo.

1.2 Requisiti

TeamManager nasce dall'esigenza di fornire a una moderna società calcistica un sistema gestionale integrato. La piattaforma centralizza le operazioni fondamentali per la gestione della rosa, la pianificazione delle attività, la comunicazione delle disponibilità in remoto e offre inoltre strumenti per la raccolta di statistiche essenziali sui singoli giocatori essenziali per fornire report avanzati al fine di supportare le decisioni tecniche tattiche dell'allenatore. L'obiettivo è offrire ad allenatori e giocatori uno strumento per organizzare le attività quotidiane in modo semplice ed efficace, riducendo la dipendenza da comunicazioni frammentate e ottimizzando l'uso delle risorse. Sulla base di queste esigenze, il sistema dovrà soddisfare i seguenti requisiti, raggruppati per area funzionale:

1. Gestione della Rosa e dei Giocatori

- **Creazione e gestione dei giocatori:** L'allenatore deve poter inserire nuovi giocatori nel sistema, associando a ciascuno informazioni anagrafiche e tecniche.
- **Amministrazione della rosa:** L'allenatore deve poter aggiungere giocatori alla rosa, modificarne i dati (es. ruolo attuale, numero di maglia, stato) o rimuoverli.

2. Gestione Eventi e Disponibilità

- **Pianificazione eventi:** L'allenatore deve poter creare, modificare e cancellare eventi nel calendario (allenamenti o partite amichevoli), specificandone i dettagli come data, orario, luogo ecc.
- **Gestione delle presenze:** I giocatori devono poter visualizzare gli eventi a cui sono convocati e comunicare la propria disponibilità, confermando la presenza notificando un'assenza motivata.

3. Analisi delle Performance e Supporto Decisionale

- **Raccolta statistiche :** Il sistema deve raccogliere dati individuale dei giocatori durante i vari eventi.
- **Sviluppo individuale:** Il sistema deve analizzare i dati di un giocatore per suggerire all'allenatore sessioni di allenamento mirate a colmarne le lacune.
- **Confronto giocatori:** Il sistema deve permettere di confrontare le statistiche di due diversi giocatori per visualizzare le differenze di prestazione.
- **Suggerimento di formazioni:** Grazie alle statistiche raccolte il sistema sarà in grado di suggerire delle formazioni all'allenatore.
- **Andamento della rosa:** L'allenatore potrà visualizzare l'andamento generale della rosa

4. Comunicazione e Notifiche

- **Notifiche automatiche:** Il sistema deve inviare notifiche ai giocatori riguardo a nuovi eventi, modifiche al calendario o aggiornamenti della rosa che li riguardano.

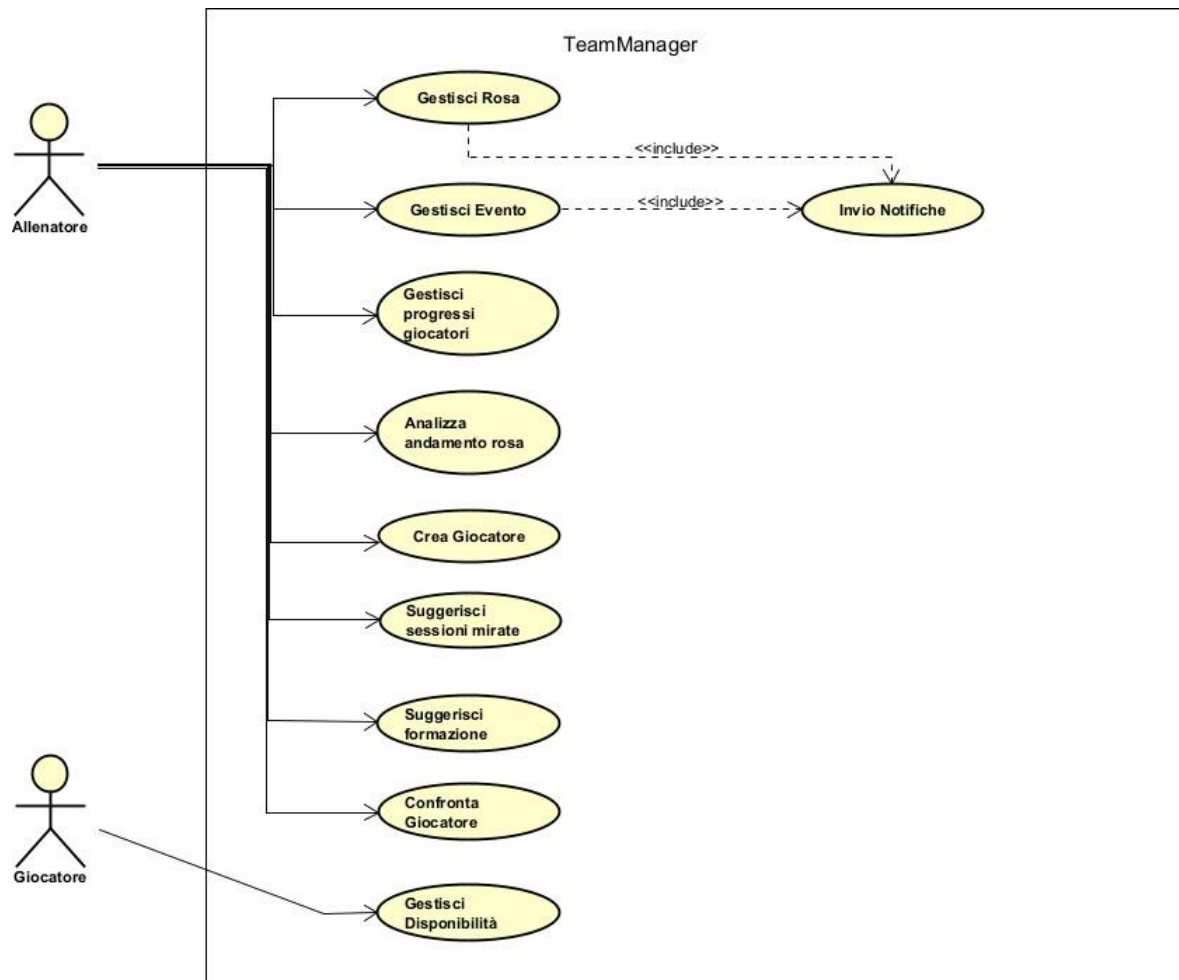
1.3 Obiettivi e casi d'uso

Seguendo i requisiti descritti nel paragrafo precedente, sono stati identificati gli attori coinvolti nell'applicazione e gli obiettivi da raggiungere. Sulla base di tali informazioni, sono stati delineati i relativi casi d'uso.

Attore	Obiettivo	Caso D'uso
Allenatore	Gestire l'aggiunta, la modifica e la rimozione dei giocatori dalla rosa, aggiornando informazioni come ruolo, numero di maglia e stato fisico.	UC1: Gestisci Rosa
Allenatore	Creare, modificare e cancellare eventi nel calendario della rosa, definendo orari e luoghi per ogni allenamento o amichevole.	UC2: Gestisci Evento (CRUD)
Giocatore	Visualizza i vari eventi pianificati sul calendario dell'applicazione, avendo la possibilità di inserire la presenza o comunicare un'assenza con motivazione.	UC3: Gestisci disponibilità (CRUD)
Allenatore	Genera un report con statistiche aggregate per valutare l'andamento della rosa (media goal segnati, media presenze, etc.).	UC4: Analizza andamento Rosa
Sistema	Inviare notifiche automatiche relative agli eventi programmati e alle situazioni critiche come modifiche alla formazione o infortuni.	UC5: Invio Notifiche
Allenatore	L'allenatore registra parametri relativi a un giocatore dopo un evento (gol, assist, gol aspettati, intercettazioni riuscite, passaggi riusciti, palloni	UC6: Gestisci progressi giocatori (CRUD)

	recuperati, palloni persi, velocità media, durata allenamento, etc.)	
Allenatore	L'allenatore crea nuovi giocatori per poi poter essere inseriti nella rosa	UC7: Crea giocatore (CRUD)
Allenatore	Il sistema elabora automaticamente una formazione consigliata (11 titolari) basandosi sulle statistiche registrate nelle recenti amichevoli	UC8: Suggerisci formazione
Allenatore	Il sistema propone allenamenti personalizzati in base ai punti deboli rilevati grazie alle statistiche dei singoli giocatori (es. → allenamento di scatto per colmare una lacuna relativa alla velocità massima raggiunta dal calciatore che non soddisfa l'allenatore)	UC9: Suggerisci sessioni mirate
Allenatore	Il sistema permette di effettuare un confronto tra due diversi giocatori, relativamente alle loro statistiche(goal fatti, ecc.).	UC10: Confronta giocatore

I casi d'uso possono essere rappresentati con questo diagramma UML:



1.4 Modello dei casi d'uso

Tra i casi d'uso individuati, è stata fornita una descrizione dettagliata **per tutti i casi d'uso integrati in questa prima versione** mentre gli altri sono stati descritti in modo breve e informale.

UC1: GESTISCI ROSA

Nome del caso d'uso	UC1: Gestisci Rosa
Portata	Applicazione TeamManager
Livello	Obiettivo giocatore
Attore primario	Allenatore
Parti interessate e interessi	<ul style="list-style-type: none"> • Allenatore: Vuole gestire i giocatori all'interno della rosa ed i dati relativi a ciascun giocatore • Giocatore: vuole conoscere il suo status in tempo reale
Precondizioni	<ul style="list-style-type: none"> • L'allenatore ha effettuato l'accesso al sistema; • L'allenatore può accedere alla Rosa; • L'allenatore può visualizzare la lista dei giocatori presenti in Rosa; • L'allenatore conosce ogni singolo giocatore della rosa.
Garanzie di successo	Le modifiche effettuate dall'allenatore avvengono con successo e sono disponibili in tempo reale ai giocatori
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'allenatore effettua l'accesso a TeamManager 2. L'allenatore accede alla sezione "Rosa". 3. Il sistema mostra tutti i giocatori presenti nella Rosa all'allenatore; 4. L'allenatore può effettuare diverse operazioni all'interno della Rosa; Il passo 4 può essere eseguito più volte; 5. L'allenatore termina salvando le modifiche; 6. Il giocatore viene aggiornato in tempo reale della modifica.
Estensioni	<p>*a Il sistema, in caso di crash o arresto improvviso, può:</p> <ol style="list-style-type: none"> 1. Effettuare delle operazioni di ripristino; 2. Effettuare un backup all'ultima modifica effettuata correttamente. <p>3a Il sistema fornisce all'allenatore:</p> <ol style="list-style-type: none"> 1. La lista di tutti i giocatori presenti nella rosa; 2. il numero di giocatori inseriti nella rosa; <p>3b Anche il giocatore dalla sua sessione accedendo alla sezione Rosa potrà visualizzare la lista dei giocatori presenti nella rosa;</p> <p>4a L'allenatore può inserire un nuovo giocatore all'interno della rosa attraverso l'opzione <i>aggiungi giocatore (+)</i>:</p>

	<ol style="list-style-type: none"> 1. L'allenatore seleziona il giocatore all'interno della lista dei giocatori presenti nel sistema; 2. L'allenatore inserisce il giocatore all'interno della rosa associando ad esso ulteriori parametri di rosa. In particolare, al giocatore verrà assegnato un ruolo specifico (portiere, difensore, centrocampista o attaccante) e uno dei seguenti status: <ul style="list-style-type: none"> • Disponibile: il giocatore è disponibile per i vari eventi; • Infortunato: il giocatore risulta non disponibile per i vari eventi; • Sospeso: il giocatore viene temporaneamente sospeso dalla rosa 3. Possono essere inseriti un massimo di 22 giocatori all'interno della Rosa, il sistema genererà un messaggio di errore qualora si provasse ad aggiungere un ulteriore nuovo giocatore. <p>4b L'allenatore può effettuare delle modifiche sui dati di un giocatore presente in rosa attraverso l'opzione <i>modifica</i>:</p> <ol style="list-style-type: none"> 1. L'allenatore seleziona il giocatore all'interno della rosa; 2. L'allenatore potrà effettuare delle modifiche sul giocatore (status, ruolo e numero di maglia). <p>4c L'allenatore può eliminare un giocatore dalla rosa attraverso l'opzione <i>Elimina(X)</i></p> <ol style="list-style-type: none"> 1. L'allenatore seleziona il giocatore all'interno della rosa; 2. Il giocatore viene rimosso dalla rosa; <p>6a Il sistema fornisce al giocatore una notifica che lo aggiorna relativamente alle nuove operazioni.</p>
Requisiti speciali	Supporto di più lingue
Elenco delle varianti tecnologiche e dei dati	Non specificate
Frequenza di ripetizione	Variabile e dipende dalla volontà dell'allenatore
Varie	Non specificate

UC2: GESTICI EVENTO, CRUD

Nome del caso d'uso	UC2: Gestisci evento
Portata	Applicazione TeamManager
Livello	Obiettivo giocatore

Attore primario	Allenatore
Parti interessate e interessi	<ul style="list-style-type: none"> • Allenatore: Vuole gestire e pianificare i vari eventi; • Giocatore: Vuole conoscere data durata e luogo dei vari eventi.
Precondizioni	<ul style="list-style-type: none"> • L'allenatore ha effettuato l'accesso al sistema; • L'allenatore può accedere al Calendario; • L'allenatore può visualizzare la lista degli eventi già pianificati.
Garanzie di successo	<ul style="list-style-type: none"> • Le pianificazioni effettuate dall'allenatore avvengono con successo e sono disponibili in tempo reale ai giocatori • Si garantisce la non sovrapposizione di più eventi.
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'allenatore effettua l'accesso a TeamManager; 2. L'allenatore accede alla sezione Calendario; 3. L'allenatore visualizza il Calendario offerto dal sistema; 4. L'allenatore può effettuare diverse operazioni; Il passo 4 può essere iterato più volte 5. le operazioni vengono salvate all'interno del calendario.
Estensioni	<p>*a Il sistema, in caso di crash o arresto improvviso, può:</p> <ol style="list-style-type: none"> 1. Effettuare delle operazioni di ripristino; 2. Effettuare un backup all'ultima modifica effettuata correttamente. <p>3a Il sistema fornisce all'allenatore la lista degli eventi già pianificati.</p> <p>3b Anche il giocatore dalla sua sessione accedendo alla sezione Calendario potrà visualizzare la lista degli eventi già pianificati;</p> <p>4a L'allenatore può pianificare un nuovo evento all'interno del Calendario attraverso l'opzione <i>aggiungi evento(+)</i>:</p> <ol style="list-style-type: none"> 1. L'allenatore crea l'evento all'interno del calendario specificando durata, luogo, orario e tipologia. Si indentificano due tipologie di evento: <ul style="list-style-type: none"> • sessione di allenamento (4a.1); • partita amichevole con altre squadre(4a.2). 2. L'evento sarà ora visibile all'interno del calendario; 3. L'allenatore non può pianificare eventi nello stesso luogo, nella Stessa data e nello stesso orario(non è consentita la sovrapposizione degli eventi). Il sistema eventualmente genererà un messaggio di errore. <p>4b L'allenatore può modificare un evento all'interno del Calendario attraverso l'opzione <i>modifica</i>:</p> <ol style="list-style-type: none"> 1. L'allenatore può selezionare l'evento da modificare(durata, luogo, ecc,).

	<p>2. L'evento viene modificato;</p> <p>4c L'allenatore può eliminare un evento all'interno del Calendario attraverso l'opzione <i>elimina evento(X)</i>:</p> <ol style="list-style-type: none"> 1. L'allenatore può selezionare un evento da eliminare; 2. L'evento viene rimosso dal calendario. <p>4d L'allenatore può visualizzare i dettagli di un evento all'interno del Calendario attraverso l'opzione <i>+dettagli</i>, stessa cosa anche per il giocatore dalla sua sessione.</p> <p>5a Il sistema fornisce al giocatore una notifica che lo aggiorna di eventuali modifiche.</p>
Requisiti speciali	Supporto di più lingue;
Elenco delle varianti tecnologiche e dei dati	Non specificate
Frequenza di ripetizione	Variabile e associato alla volontà dell'allenatore
Varie	Non specificate

UC3: GESTISCI DISPONIBILITA, CRUD

Nome del caso d'uso	UC3: Gestisci disponibilità
Portata	Applicazione TeamManager
Livello	Obiettivo Allenatore
Attore primario	Giocatore
Parti interessate e interessi	<ul style="list-style-type: none"> • Giocatore: può comunicare la propria disponibilità ai vari eventi; • Allenatore: Vuole conoscere la disponibilità di ciascun giocatore relativa ai vari eventi.
Precondizioni	<ul style="list-style-type: none"> • Il giocatore della rosa ha effettuato l'accesso al sistema; • Il giocatore può accedere alla sezione "Calendario"; • Il giocatore può visualizzare la lista degli eventi già pianificati.
Garanzie di successo	<ul style="list-style-type: none"> • Il giocatore può comunicare la propria disponibilità ai vari eventi;

Scenario principale di successo	<ol style="list-style-type: none"> 1. Il giocatore effettua l'accesso a TeamManager; 2. Il giocatore accede alla sezione calendario; 3. Il giocatore visualizza il calendario offerto dal sistema; 4. Il giocatore può comunicare la propria disponibilità all'evento selezionato; Il passo 4 può essere iterato più volte 5. le operazioni vengono salvate all'interno del calendario;
Estensioni	<p>*a Il sistema, in caso di crash o arresto improvviso, può:</p> <ol style="list-style-type: none"> 1. Effettuare delle operazioni di ripristino; 2. Effettuare un backup all'ultima modifica effettuata correttamente. <p>3a Il sistema fornisce al giocatore la lista degli eventi già pianificati.</p> <p>4a Il giocatore all'interno del Calendario può comunicare la propria disponibilità attraverso l'opzione <i>+Disponibilità</i>:</p> <ol style="list-style-type: none"> 1. Il giocatore può selezionare un evento all'interno del calendario; 2. Il giocatore comunica la propria disponibilità all'evento in questione. In particolare, il giocatore potrà selezionare 2 opzioni: <ul style="list-style-type: none"> • Presente; • Assente: in tal caso il giocatore deve comunicare anche l'eventuale motivazione; 3. Il giocatore potrà visualizzare grazie alla colonna <i>adesione fornita</i> se ha fornito o meno la propria disponibilità ad uno specifico evento. 4. Il giocatore può comunicare la propria disponibilità solo per eventi futuri. <p>4b L'allenatore dalla sua sessione visualizzerà sui dettagli evento la lista dei giocatori disponibili a quello specifico evento</p> <p>5a l'allenatore verrà notificato ogni qual volta un giocatore comunica la propria disponibilità.</p>
Requisiti speciali	Supporto di più lingue;
Elenco delle varianti tecnologiche e dei dati	Non specificate
Frequenza di ripetizione	Variabile e associato al Giocatore.
Varie	Non specificate

UC4: ANALIZZA ANDAMENTO ROSA

Nome del caso d'uso	UC4: Andamento Rosa
Portata	Applicazione TeamManager
Livello	Obiettivo allenatore
Attore primario	Allenatore
Parti interessate e interessi	<ul style="list-style-type: none"> • Allenatore: Vuole analizzare l'andamento generale della rosa; • Sistema: recupera i dati necessari che, una volta elaborati, conducono alla generazione della pagina e dei dati.
Precondizioni	<ul style="list-style-type: none"> • L'allenatore ha effettuato l'accesso al sistema; • L'allenatore può accedere alla Rosa; • La rosa deve essere al completo; • Il sistema è in grado di recuperare una molteplicità di dati.
Garanzie di successo	L'allenatore visualizzerà correttamente la pagina sull'andamento della rosa.
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'allenatore effettua l'accesso a TeamManager; 2. L'allenatore accede alla sezione di "Rosa"; 3. L'allenatore potrà selezionare l'opzione "andamento Rosa" 4. Il sistema recupera i vari dati necessari per l'elaborazione; 5. Il sistema elabora e analizza i dati ricevuti utilizzando una logica appropriata; 6. I dati vengono utilizzati per creare una pagina relativa all'andamento generale della rosa; 7. L'allenatore visualizzerà correttamente il contenuto della pagina.
Estensioni	<p>*a Il sistema, in caso di crash o arresto improvviso, può:</p> <ol style="list-style-type: none"> 1. Effettuare delle operazioni di ripristino; 2. Effettuare un backup all'ultima modifica effettuata correttamente <p>2a Il sistema fornisce all'allenatore la lista di tutti i giocatori presenti nella rosa;</p> <p>4a Il sistema, per una corretta generazione dell'andamento si baserà su 4 fattori fondamentali:</p> <ul style="list-style-type: none"> • conteggio dei giocatori per status: viene mostrato i numeri di giocatori appartenenti ad uno specifico status;

	<ul style="list-style-type: none"> • conteggio degli eventi pianificati ma da svolgere: viene mostrato il numero di allenamenti che devono ancora essere svolti; • media di presenze e assenze relative agli eventi del mese corrente: vengono mostrate le percentuali di assenze e presenze generali relative agli eventi del mese corrente; • media su statistiche relative ai giocatori: vengono mostrate delle statiche amichevoli relative all'intera Rosa (Una media). In particolare verranno mostrati: <ul style="list-style-type: none"> • Goal; • Assist; • Cartellini; • Falli commessi. <p>verranno considerati per la generazione i dati relativi a tutte le amichevoli a cui i giocatori hanno partecipato facendone una media.</p> <p>5a: il sistema procede alla generazione della pagina con i dati disponibili che possiede;</p>
Requisiti speciali	<ol style="list-style-type: none"> 1. Il sistema deve garantire tempi di risposta rapidi per il recupero dati. 2. La pagina deve essere presentata in modo chiaro e comprensibile.
Elenco delle varianti tecnologiche e dei dati	Non specificate
Frequenza di ripetizione	Variabile, in base alla necessità dell'allenatore di valutare l'andamento della Rosa
Varie	Non specificate

UC5: INVIO NOTIFICHE

1. Il sistema implementa un meccanismo di notifiche;
2. Il sistema manda una notifica al giocatore ogni qual volta avviene un'operazione al proprio status in rosa;
3. Il sistema manda una notifica al giocatore alla creazione o modifica di un evento;
4. Il sistema manda una notifica all'allenatore sulla disponibilità di un giocatore.

UC6: GESTISCI PROGRESSI GIOCATORE, CRUD

Nome del caso d'uso	UC6: Gestisci progressi giocatore
Portata	Applicazione TeamManager
Livello	Obiettivo Allenatore
Attore primario	Allenatore
Parti interessate e interessi	<ul style="list-style-type: none"> • Allenatore: Vuole registrare le statistiche ai singoli giocatori per uno specifico evento;
Precondizioni	<ul style="list-style-type: none"> • L'allenatore ha effettuato l'accesso al sistema. • L'allenatore visualizza correttamente la lista dei giocatori presenti in Rosa; • L'allenatore visualizza la lista di disponibilità relative al giocatore ad uno specifico evento;
Garanzie di successo	<ul style="list-style-type: none"> • Le statistiche vengono registrate correttamente e sono associate al giocatore per uno specifico evento.
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'allenatore effettua l'accesso a TeamManager; 2. L'allenatore accede alla sezione "Gestisci progressi giocatore"; 3. L'allenatore visualizza la Rosa; 4. L'allenatore può selezionare un giocatore all'interno della rosa; 5. L'allenatore cliccando sull'opzione "GO" visualizzerà gli eventi a cui esso ha partecipato; 6. L'allenatore può effettuare diverse operazioni <p>I passi 4, 5 e 6 può essere iterati più volte</p> <ol style="list-style-type: none"> 7 le modifiche vengono salvate correttamente.
Estensioni	<p>a Il sistema, in caso di crash o arresto improvviso, può:</p> <ol style="list-style-type: none"> 1. Effettuare delle operazioni di ripristino; 2. Effettuare un backup all'ultima modifica effettuata correttamente <p>3a Il sistema fornisce all'allenatore:</p> <ol style="list-style-type: none"> 1. La lista di tutti i giocatori presenti nella rosa; 2. il numero di giocatori inseriti nella rosa. <p>5a Il sistema fornisce all'allenatore la lista delle disponibilità del giocatore selezionato.</p> <p>6a L'allenatore può registrare una nuova statistica relativa ad un evento attraverso l'opzione "+Statistica"</p>

	<ol style="list-style-type: none"> 1. L'allenatore può selezionare un evento a cui il giocatore ha partecipato; 2. Le statistiche si differenzieranno in base alla tipologia di evento in: <ul style="list-style-type: none"> • Statistica amichevole: relativa ad eventi di tipo amichevole; • Statistica allenamento: relativa ad eventi di tipo allenamento; Ognuna con attributi specifici. 3. L'Allenatore inserisce tutti i dati relativa alla tipologia di statistica. 4. La Statistica viene registrata correttamente <p>6b L'allenatore può eliminare una statistica già registrata e relativa ad un evento attraverso l'opzione "<i>Xstatistica</i>"</p> <ol style="list-style-type: none"> 1. L'allenatore può selezionare un evento in cui il giocatore ha partecipato e dove la statistica è già stata registrata. 2. La statistica viene eliminata correttamente. <p>6c L'allenatore può visualizzare una statistica già registrata e relativa ad un evento attraverso l'opzione "<i>info Stats</i>"</p> <ol style="list-style-type: none"> 1. l'allenatore può selezionare un evento a cui il giocatore ha partecipato; 2. L'allenatore potrà visualizzare la statistica solo per eventi in cui è già stata registrata.
Requisiti speciali	Supporto di più lingue;
Elenco delle varianti tecnologiche e dei dati	Non specificate
Frequenza di ripetizione	Variabile e associato alla volontà dell'allenatore
Varie	Non specificate

UC7: CREA GIOCATORE, CRUD

Nome del caso d'uso	UC7: Crea Giocatore
Portata	Applicazione TeamManager
Livello	Obiettivo giocatore
Attore primario	Allenatore

Parti interessate e interessi	<ul style="list-style-type: none"> • Allenatore: Vuole creare un nuovo giocatore all'interno del sistema
Precondizioni	L'allenatore effettua l'accesso al sistema.
Garanzie di successo	Il giocatore viene creato e inserito nella lista dei giocatori all'interno del sistema;
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'allenatore effettua l'accesso a TeamManager; 2. L'allenatore accede alla sezione "Crea Giocatore"; 3. L'allenatore inserisce i campi relativi al nuovo giocatore(nome, cognome, data di nascita, e-mail, ecc.); 4. L'allenatore conferma la creazione del giocatore; I passi 3 e 4 possono essere iterati più volte 5. Il giocatore viene creato correttamente;
Estensioni	<p>*a Il sistema, in caso di crash o arresto improvviso, può:</p> <ol style="list-style-type: none"> 1. Effettuare delle operazioni di ripristino; 2. Effettuare un backup all'ultima modifica effettuata correttamente <p>3a non possono essere creati giocatori con lo stesso indirizzo e-mail(non sono ammessi duplicati)</p> <ol style="list-style-type: none"> 1. Il sistema genererà a un messaggio di errore qualora si provasse a creare un giocatore con l'indirizzo e-mail già utilizzato e presente nel sistema <p>5a Il giocatore viene inserito all'interno della lista dei giocatori presente nel sistema.</p>
Requisiti speciali	Supporto di più lingue;
Elenco delle varianti tecnologiche e dei dati	Non specificate
Frequenza di ripetizione	Variabile e associato alla volontà dell'allenatore
Varie	Non specificate

UC8: SUGGERISCI FORMAZIONE

Nome del caso d'uso	UC8: Suggerisci Formazione
Portata	Applicazione TeamManager
Livello	Obiettivo Allenatore

Attore primario	Allenatore
Parti interessate e interessi	<ul style="list-style-type: none"> • Sistema: recupera i dati necessari che, una volta elaborati, conducono alla generazione della formazione; • Allenatore: vuole visualizzare la formazione suggerita dal sistema;
Precondizioni	<ul style="list-style-type: none"> • L'allenatore ha effettuato l'accesso al sistema. • La rosa risulta essere al completo. • Vi sono statistiche sufficienti per la generazione della formazione;
Garanzie di successo	Il sistema suggerisce una formazione e viene visualizzata correttamente dall'allenatore.
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'allenatore effettua l'accesso a TeamManager; 2. L'allenatore accede alla sezione "Suggerisci Formazione" 3. L'allenatore deve selezionare un modulo; 4. Il sistema recupera i dati necessari per la generazione; 5. I dati vengono elaborati per recuperare i giocatori più in forma; 6. La formazione viene generata; 7. L'allenatore visualizza correttamente la formazione suggerita
Estensioni	<p>*a Il sistema, in caso di crash o arresto improvviso, può:</p> <ol style="list-style-type: none"> 1. Effettuare delle operazioni di ripristino; 2. Effettuare un backup all'ultima modifica effettuata correttamente. <p>3a L'allenatore potrà selezionare uno dei seguenti moduli:</p> <ul style="list-style-type: none"> • 4-3-3 • 3-5-2 • 3-4-3 <p>4a: Per la generazione della formazione verranno utilizzate le Statistiche raccolte nelle varie amichevoli;</p> <p>4b verranno considerati per la generazione i dati relativi a tutte le Amichevoli a cui il giocatore ha partecipato facendone una media.</p> <p>4c La formazione verrà suggerita se e solo se la rosa risulta completa</p> <p>4d La formazione verrà suggerita se e solo se all'interno della rosa ci sono i giocatori richiesti per ruolo altrimenti il sistema genera un messaggio di errore</p> <p>5a Per il recupero dei giocatori verranno utilizzati campi specifici per ruolo. In particolare:</p>

	<ul style="list-style-type: none"> • Parate: per i portieri; • intercettiRiusciti: per i difensori; • passaggiChiave: per i centrocampisti; • Goal: per gli attaccanti; <p>In base a questi parametri verranno effettuati dei confronti per recuperare i giocatori più in forma per ruolo</p>
Requisiti speciali	Supporto di più lingue
Elenco delle varianti tecnologiche e dei dati	Non specificate
Frequenza di ripetizione	Variabile e associato alla volontà dell'allenatore
Varie	Non specificate

UC9: SUGGERISCI SESSIONE MIRATA

Nome del caso d'uso	UC9: Suggerisci Sessione Mirata
Portata	Applicazione TeamManager
Livello	Obiettivo Allenatore
Attore primario	Allenatore
Parti interessate e interessi	<ul style="list-style-type: none"> • Sistema: il sistema recupera i dati necessari che, una volta elaborati, conducono alla generazione di specifiche sessioni mirate; • Allenatore: vuole visualizzare la sessione mirata suggerita per uno specifico giocatore in caso di lacune;
Precondizioni	<ul style="list-style-type: none"> • L'allenatore ha effettuato l'accesso al sistema. • L'allenatore visualizza la lista dei giocatori presenti nella rosa. • Vi sono statistiche per la generazione della Sessione mirata;
Garanzie di successo	Il sistema genera una sessione mirata che viene visualizzata correttamente dall'allenatore
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'allenatore effettua l'accesso a TeamManager; 2. L'allenatore accede alla sezione "Sessione Mirata"; 3. L'allenatore potrà selezionare un giocatore della rosa; 4. Il sistema recupera i dati relativi al giocatore; 5. I dati vengono elaborati per individuare eventuali lacune; 6. Il sistema genera una sessione mirata specifica per quel giocatore;

	7. L'allenatore visualizza correttamente la sessione suggerita.
Estensioni	<p>a Il sistema, in caso di crash o arresto improvviso, può:</p> <ol style="list-style-type: none"> 1. Effettuare delle operazioni di ripristino; 2. Effettuare un backup all'ultima modifica effettuata correttamente. <p>2a Il sistema fornisce all'allenatore la lista di tutti i giocatori presenti nella rosa;</p> <p>4a Per la generazione delle sessioni mirate verranno utilizzate le Statistiche raccolte nelle varie sessioni di allenamento;</p> <p>4b Verranno automaticamente esclusi giocatori senza alcun parametro Statistico.</p> <p>4c verranno considerati per la generazione i dati relativi a tutte le Sessione di allenamento a cui il giocatore ha partecipato facendone una media.</p> <p>5a i campi specifici relativi ad una sessione di allenamento individueranno una sessione mirata ben precisa. In particolare avremo:</p> <ul style="list-style-type: none"> • allenamento sullo scatto: considerano la velocità massima; • allenamento sulla resistenza: considerando la frequenza cardiaca media; • allenamento sul tiro: considerando la forza del tiro; • allenamento sui pesi: considerano la valutazione fisica media; • allenamento aerobico: considerando la velocità media. <p>5c La sessione mirata viene suggerita solo se un giocatore non rispetta le soglie definite dall'allenatore per un campo specifico.</p> <p>6a viene suggerita una sessione mirata per ogni lacuna individuata.</p> <p>6b se il giocatore non presenta lacune il sistema notifica l'allenatore che non vi è bisogno di alcuna sessione mirata.</p>
Requisiti speciali	Supporto di più lingue;
Elenco delle varianti tecnologiche e dei dati	Non specificate
Frequenza di ripetizione	Variabile e associato alla volontà dell'allenatore
Varie	Non specificate

UC10: CONFRONTA GIOCATORI

Nome del caso d'uso	UC8: Confronta Giocatori
Portata	Applicazione TeamManager
Livello	Obiettivo Allenatore
Attore primario	Allenatore
Parti interessate e interessi	<ul style="list-style-type: none"> • Sistema: il sistema recupera i dati necessari per il confronto; • Allenatore: vuole visualizzare il confronto tra due giocatori.
Precondizioni	<ul style="list-style-type: none"> • L'allenatore ha effettuato l'accesso al sistema. • L'allenatore visualizza la lista dei giocatori presenti nella rosa.
Garanzie di successo	Il sistema genererà un confronto tra due giocatori che viene visualizzato correttamente dall'allenatore
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'allenatore effettua l'accesso a TeamManager; 2. L'allenatore accede alla sezione "Confronta Giocatori"; 3. L'allenatore potrà selezionare due giocatori della rosa; 4. Il sistema recupera i dati relativi ai due giocatori; 5. I dati vengono elaborati dal sistema; 6. Il sistema genera il confronto tra i due giocatori; 7. L'allenatore visualizza correttamente il confronto.
Estensioni	<p>a Il sistema, in caso di crash o arresto improvviso, può:</p> <ol style="list-style-type: none"> 1. Effettuare delle operazioni di ripristino; 2. Effettuare un backup all'ultima modifica effettuata correttamente. <p>2a Il sistema fornisce all'allenatore la lista di tutti i giocatori presenti nella rosa;</p> <p>4a Per il confronto verranno utilizzate le Statistiche raccolte nelle varie amichevoli;</p> <p>4b verranno considerati per il confronto i dati relativi a tutte le Amichevoli a cui il giocatore ha partecipato facendone una media.</p>

	6a gli attributi migliori saranno evidenziati di verde, i peggiori di rosso
Requisiti speciali	Supporto di più lingue;
Elenco delle varianti tecnologiche e dei dati	Non specificate
Frequenza di ripetizione	Variabile e associato alla volontà dell'allenatore
Varie	Non specificate

1.5 Documento di visione

Durante la stesura di questo capitolo, è stata elaborata la versione definitiva del documento di Visione. Per la consultazione andare nella cartella *APPENDICE A- documento di visione*

1.6 Regole di dominio

ID	Regola	Modificabilità	Sorgente
R1	Ogni giocatore registrato non può essere rimosso dall'allenatore senza una motivazione valida (es. infortunio prolungato, uscita dalla squadra, comportamento inadeguato, etc.).	Media, potrebbe essere necessario prevedere la gestione di ex-giocatori.	Regola interna della squadra.
R2	Il giocatore deve avere un identificativo univoco, non sono consentiti duplicati e non può essere inserito più volte all'interno della stessa rosa.	Bassa, strettamente legata alla gestione della rosa.	Regola interna del sistema.
R3	La rosa può contenere un massimo di 22 giocatori , qualora la rosa fosse piena nessun altro giocatore potrà essere inserito al suo interno	Bassa, strettamente legata alla gestione della rosa.	Regola interna della squadra.

R4	Un giocatore può avere uno e un solo stato tra: Disponibile, Infortunato o Sospeso .	Media, nuove categorie potrebbero essere aggiunte in futuro.	Regola interna del sistema.
R5	Non ci possono essere due giocatori con lo stesso numero di maglia all'interno della rosa	Bassa, strettamente legata alla gestione della rosa.	Regola interna della squadra.
R6	Ogni giocatore inserito nella rosa deve avere un ruolo assegnato dall'enumerazione predefinita (PORTIERE, DIFENSORE, ecc.).	Media, nuove categorie potrebbero essere aggiunte in futuro.	Regola interna del sistema.
R7	Più eventi possono essere pianificati nello stesso giorno, ma non possono sovrapporsi .	Bassa, modificabile solo con una revisione del modello degli eventi.	Regola interna del sistema per evitare conflitti.
R8	La sessione mirata può essere generata solo su giocatori esistenti in rosa e a cui sono associate statistiche amichevoli.	Bassa, strettamente legata alla gestione della rosa.	Regola tecnica interna.
R9	Le statistiche di performance relative a un evento possono essere registrate solo per i giocatori che hanno confermato la loro presenza e vi hanno effettivamente partecipato.	bassa, strettamente legata alla partecipazione dei giocatori al suddetto evento	Regola tecnica interna.
R10	Il sistema può suggerire una formazione solo se vengono soddisfatti il numero minimo di giocatori per ruolo in base alla tipologia di modulo e se la rosa è al completo	Bassa, strettamente legata alla generazione dei dati	Regola interna del sistema.
R11	Il sistema fornisce all'allenatore un numero limitato di formazioni preesistenti(esempio 4 3 3), ai fini della formazione suggerita.	Media, nuove formazioni potrebbero essere aggiunte in futuro.	Regola interna del sistema.
R12	Non è permesso comunicare una disponibilità da parte di un giocatore per un evento passato.	Bassa, strettamente legata alla generazione partecipazione dei giocatori agli eventi	Regola interna del sistema.

1.7 Specifiche Supplementari

USABILITA

- L'interfaccia grafica deve essere di facile utilizzo anche per un utente non esperto;
- L'interfaccia utente deve essere coerente in tutte le sue parti. Controlli, menu e azioni comuni devono mantenere la stessa posizione e lo stesso stile per facilitare l'apprendimento;
- Il sistema deve fornire un riscontro visivo immediato in risposta alle azioni dell'utente (es. messaggi di errore);

AFFIDABILITA

- Il software prodotto deve poter mantenere i dati anche in caso di anomalie come guasti hardware, attacchi informatici;
- I dati che i vari utenti salvano sono consultabili in sicurezza attraverso autenticazione con credenziali;
- Il sistema deve garantire la coerenza e la correttezza dei dati attraverso validazioni in fase di inserimento, rispettando le regole di dominio (es. un giocatore può avere un solo ruolo)
- Il sistema deve prevedere una funzionalità di backup dei dati, possibilmente automatica, per consentire un ripristino all'ultimo stato funzionante in caso di problemi;

VINCOLI HARDWARE E SOFTWARE

- L'esecuzione del software richiedere una connessione stabile a rete internet.
- L'applicazione necessita solamente, nella macchina in cui si intende eseguire, la Java Virtual Machine.
- Le operazioni comuni (es. caricamento della rosa, visualizzazione del calendario) devono essere eseguite in tempi rapidi, idealmente entro 2-3 secondi. La generazione di report complessi non deve bloccare l'interfaccia utente.

- Il software è sviluppato interamente utilizzando come linguaggio di programmazione Java con l'estensione Java swing per la creazione dell'interfaccia.
- L'applicazione deve avere un consumo di memoria (RAM) e CPU ragionevole per non impattare negativamente sulle altre operazioni del computer dell'utente.

ASPETTI LEGALI

- Il software realizzato è di tipo freeware, in quanto si è deciso di mantenere il codice sorgente privato.
- I dati inseriti nel sistema (informazioni sui giocatori, statistiche, eventi) sono di esclusiva proprietà e responsabilità dell'utente o della squadra che utilizza il software.
- Un giocatore non può visualizzare o modificare le informazioni di altri giocatori in virtù delle Normative GDPR e best practices di sicurezza.

1.8 Glossario

- **Rosa:** Insieme di tutti i giocatori che appartengono alla squadra gestita dall'applicazione;
- **Evento:** Qualsiasi attività pianificata all'interno dell'applicazione, come un allenamento o una partita amichevole;
- **Amichevole:** tipologia di evento che simula una partita contro una squadra avversaria;
- **Sessione di allenamento:** tipologia di evento, attività fisica svolta dai giocatori
- **Notifica:** Messaggio inviato dal sistema per aggiornamenti e comunicazioni;
- **Disponibilità:** Stato di partecipazione di un giocatore a un determinato Evento.
- **Status:** Condizione del giocatore, che può essere disponibile, infortunato o sospeso;

- **Calendario:** Funzionalità del sistema che permette di visualizzare e pianificare gli eventi della squadra in un arco di tempo temporale;
- **Modulo:** Lo schema tattico di una formazione (es. 4-4-2) che può essere usato dal sistema come base per suggerire gli undici titolari o generare report di squadra;
- **Statistica:** Insieme dei dati statistici raccolti per un giocatore durante un evento (es. gol, assist, minuti giocati).
- **Statistica amichevole:** Insieme dei dati statistici raccolti per un giocatore durante un amichevole;
- **Statistica allenamento:** Insieme dei dati statistici raccolti per un giocatore durante una sessione di allenamento;
- **Lacuna:** Carenza di un giocatore stabilità sulla media di tutte le sue sessioni di allenamento;
- **Sessione Mirata:** Una tipologia di allenamento personalizzato che il sistema suggerisce per un giocatore, basandosi sull'analisi delle sue Prestazioni per colmarne specifiche lacune.

2.0 ANALISI ORIENTATA AGLI OGGETTI

2.1 Introduzione

Seguendo l'approccio iterativo ed evolutivo suggerito dal Unified Process (UP), la realizzazione dell'applicazione **TeamManager** è stata articolata su 4 iterazioni. In questo modo è stato possibile implementare in maniera incrementale il nucleo dell'architettura del software, sono state risolte le problematiche relative ai rischi maggiori ed è stata affrontata un'analisi dei requisiti graduale, in modo da limitare al minimo il danno causato da eventuali errori di progettazione e implementazione.

ITERAZIONE 1

- Implementazione scenario di successo **UC7: Crea Giocatore** per la creazione di un nuovo giocatore all'interno del sistema.
- Implementazione scenario principale e scenari alternativi di successo **UC1: Gestisci Rosa** per l'aggiunta di un nuovo giocatore in Rosa, la modifica di un giocatore della rosa e l'eliminazione di un giocatore dalla rosa.
- Implementazione scenario principale e scenari alternativi di successo **UC2: Gestisci Evento** per la creazione di un nuovo evento su calendario, la modifica di un evento esistente e l'eliminazione di un evento dal calendario.
- Implementazione scenario di successo **UC3: Gestisci disponibilità** per comunicare la disponibilità di un giocatore ad uno specifico evento
- Implementate classi: **TeamManager, Giocatore, GiocatoreInRosa, Rosa, Evento, Calendario, disponibilità**.
- Pattern utilizzati: Implementato pattern **GoF singleton** per TeamManager e **Facade controller** per TeamMangaer, **Creator** per la creazione delle varie entità, **information expert** per la gestione delle responsabilità. Per una visione più dettagliata andare nella cartella *elaborazione I file- iterazione 1* *paragrafo 4.0*
- Definiti ed eseguiti test unitari con **JUnit** sui metodi creati

ITERAZIONE 2

- Implementazione scenario principale e scenari alternativi di successo **UC6: Gestisci progressi giocatore** per la creazione di una statistica relativa ad uno specifico evento a cui il giocatore ha partecipato, eliminazione di una statistica esistente associata ad un giocatore e visualizzazione di una statistica già registrata per un giocatore ad uno specifico evento
- Implementazione scenario di successo **UC9: Crea Sessione Mirata** per la creazione di una sessione mirata per colmare una lacuna relativa ad un giocatore in merito alle proprie Statistiche allenamento globali

- Implementazione scenario di successo **UC10: Confronta Giocatori** per visualizzare il confronto tra due giocatori in merito alle loro statistiche amichevoli globali
- Introdotta la nuova classe **Statistica**
- Sostituzione di alcuni attributi String con nuovi attributi Enum, eliminazione attributi superflui e spostamento dell'attributo numMaglia da Giocatore a GiocatoreInRosa (per una visione più dettagliata consultare il file – *iterazione 2*
- Raffinamento regole di dominio e implementazione
- Eseguiti nuovi test unitari con **JUnit** sui nuovi metodi creati

ITERAZIONE 3

- Implementazione scenario di successo **UC4: Analizza andamento rosa** per dare ad un allenatore una visione generale sull'andamento della rosa
- Implementazione scenario di successo **UC8: suggerisci formazione** per suggerire una formazione all'allenatore composta da tutti i giocatori più in forma per ruolo
- Ulteriore raffinamento regole di dominio.
- Eseguiti nuovi ulteriori test unitari con **JUnit** sui nuovi metodi creati

ITERAZIONE 4

- Implementazione interfaccia grafica GUI col progetto;
- Stesura finale regole di dominio;
- Utilizzo della persistenza;
- Implementato pattern Pure Fabrication per la persistenza. Per una visione più dettagliata consultare la cartella *elaborazione 4* file *iterazione 4* paragrafo 6.0.

Di seguito si riportano alcune delle principali schermate dell'interfaccia, per una visione più dettagliata consultare la cartella *Elaborazione 4* file - *iterazione 4* paragrafo 5.1.

Login - Allenatore

The screenshot shows a web application window titled "TeamManager v1.0". Below the title bar is a menu bar with "Options". The main content area has a light gray background with the text "TeamManager v1.0" centered at the top. In the center is a white login box with a black border. Inside this box, there are two labels: "e-Mail" and "Password". The "e-Mail" field contains the text "allenatore@team.com". The "Password" field contains the text "admin". Below the password field is a radio button labeled "Show password", which is currently selected. At the bottom right of the login box is a blue button labeled "Login".

Sessione- Allenatore

The screenshot shows the same "TeamManager v1.0" window. The main content area now displays "Benvenuto Allenatore!" (Welcome Coach!) in bold. Below this greeting is a large white rectangular area containing six gray buttons arranged in a 3x2 grid. The buttons are labeled: "Rosa", "Gestisci Progressi", "Suggerisci Formazione", "Calendario", "Sessione Mirata", "Crea giocatore", and "Confronta Giocatore".

Rosa – Allenatore

TeamManager v1.0

Options

Rosa Attuale - Incompleta 21 / 22

Mario	Rossi	#19	ATTACCANTE	DISPONIBILE
Cristiano	Ronaldo	#7	ATTACCANTE	DISPONIBILE
Lionel	Messi	#90	ATTACCANTE	DISPONIBILE
Kylian	Mbappé	#11	ATTACCANTE	DISPONIBILE
Robert	Lewandowski	#9	ATTACCANTE	INFORTUNATO
Kevin	De Bruyne	#17	CENTROCAMPISTA	DISPONIBILE
Virgil	van Dijk	#4	DIFENSORE	DISPONIBILE
Manuel	Neuer	#1	PORTIERE	DISPONIBILE
Luka	Modrić	#10	CENTROCAMPISTA	DISPONIBILE
Gianluigi	Buffon	#12	ATTACCANTE	DISPONIBILE
Frenkie	de Jong	#21	CENTROCAMPISTA	DISPONIBILE
Joshua	Kimmich	#6	CENTROCAMPISTA	DISPONIBILE
Ruben	Dias	#3	DIFENSORE	DISPONIBILE
Vinicius	Junior	#20	ATTACCANTE	DISPONIBILE
Jude	Bellingham	#22	CENTROCAMPISTA	INFORTUNATO
Alphonso	Davies	#19	DIFENSORE	DISPONIBILE
Lautaro	Martinez	#23	ATTACCANTE	DISPONIBILE
Mike	Maignan	#16	CENTROCAMPISTA	DISPONIBILE
Victor	Osimhen	#18	ATTACCANTE	DISPONIBILE
Bukayo	Saka	#7	ATTACCANTE	DISPONIBILE
Joško	Gvardiol	#24	DIFENSORE	DISPONIBILE

+

Modifica

X

Andamento Rosa

Back

Calendario– Allenatore

TeamManager v1.0

Options

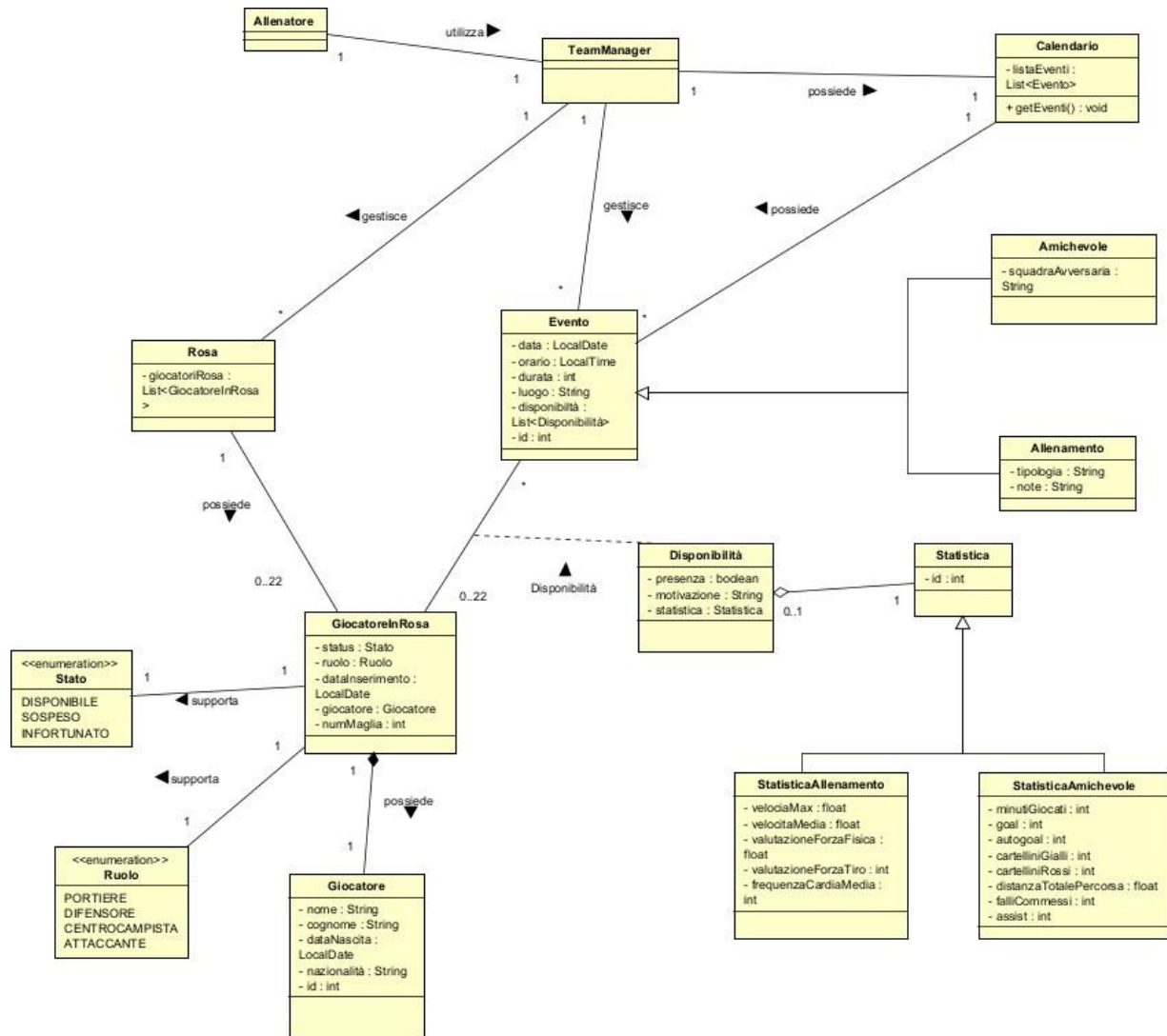
Back **Calendario Eventi** **+** **X** *Modifica*

ID	Data	Orario	Durata	Luogo
1	2023-04-05	10:00	90	Campo Principale
2	2023-04-12	15:30	90	Stadio Comunale
3	2023-04-19	09:30	120	Campo Secondario
4	2023-05-03	20:45	90	Stadio Olimpico
5	2023-05-10	11:00	90	Campo Principale
6	2023-06-07	18:00	90	Stadio Amichevole
7	2023-06-14	10:30	105	Campo Principale
8	2023-06-21	19:00	90	Stadio Comunale
9	2023-07-05	09:00	120	Campo Secondario
10	2023-07-12	17:30	90	Stadio Estivo
11	2023-07-19	10:00	90	Campo Principale
12	2023-07-22	18:30	90	Stadio Comunale
13	2023-07-25	09:30	120	Campo Secondario
14	2023-07-29	20:45	90	Stadio Olimpico
15	2023-08-02	11:00	90	Campo Principale
16	2023-08-05	17:00	90	Stadio Amichevole

2.2 Modello di dominio

la stesura del Modello di Dominio rappresenta un passo fondamentale dell'approccio iterativo UP. Questo modello rappresenta una visione concettuale del sistema, rappresentante entità, attributi e associazioni con cui il software dovrà interagire o che dovrà rappresentare. Dopo le quattro iterazioni si è arrivato al modello di dominio finale. Le principali **classi concettuali** coinvolte nei casi d'uso previsti nelle 4 iterazioni verranno trasformate in componenti software all'interno del modello di dominio. Di seguito sono elencate le classi concettuali emerse dall'analisi di tutte le iterazioni, accompagnate da una breve descrizione

Classe	Descrizione
Giocatore	Rappresenta un individuo iscrivibile alla squadra. Contiene informazioni anagrafiche (nome, cognome, e-mail), tecniche (ruolo).
GiocatoreInRosa	Rappresenta il giocatore pronto ad essere inserito all'interno della rosa a cui vengono associati parametri esclusivi.
Rosa	Contiene i giocatori attualmente attivi nella squadra. È gestita dall'allenatore ed è derivata dall'archivio generale.
Evento	Rappresenta una sessione di squadra (allenamento o partita amichevole), con informazioni su data, orario, luogo e obiettivo.
Disponibilità	Associa un giocatore a un evento specifico, indicandone la disponibilità (presente/assente) e una possibile motivazione.
Allenatore	Figura responsabile della gestione della rosa, della creazione degli eventi e della consultazione delle disponibilità.
Statistica	Raccoglie i parametri individuali del giocatore in seguito a una sua partecipazione ad un evento.
TeamManager	Rappresenta il sistema TeamManager



Per una migliore consultazione del digramma andare nella cartella *APPENDICE A* file – *Modello di dominio*

2.3 Diagrammi di Sequenza di Sistema e Contratti delle Operazioni

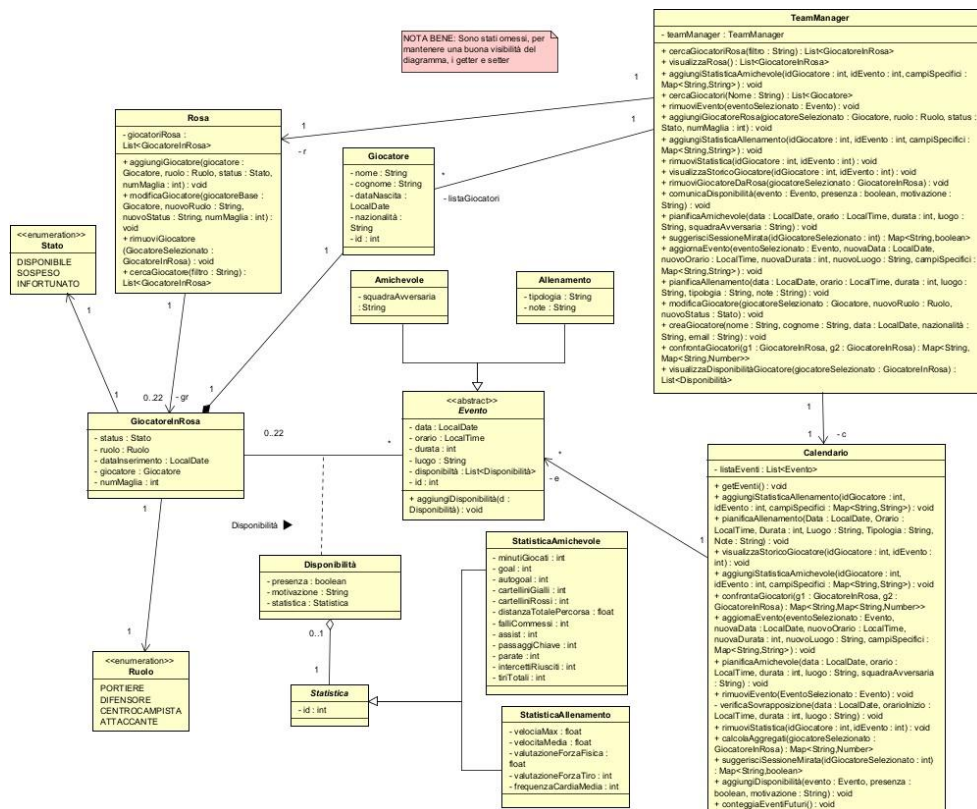
Passo successivo dell'analisi orientata agli oggetti è la creazione dei diagrammi di sequenza del sistema SSD che rappresentano e illustrano il corso degli eventi di input e output all'interno del sistema tra l'utente e il sistema TeamManager per tutti i casi d'uso esaminati in ciascuna iterazione. Le principali operazioni di sistema dei vari SSD verranno descritte mediante l'utilizzo dei contratti. I vari diagrammi possono essere consultati nella cartella *Elaborazione 1* file – *iterazione 1* paragrafo 2.1 per l'iterazione

1, nella cartella *Elaborazione 2 file – iterazione 2 paragrafo 2.1* per l'iterazione 2 e nella cartella *Elaborazione 3 file – iterazione 3 paragrafo 2.1* per l'iterazione 3. Nelle varie cartelle di *Elaborazione* sono disponibili nella sezione *schemi* tutti i diagrammi nel formato Astah.

3.0 PROGETTAZIONE

3.1 Diagrammi delle classi

L'ultimo passo dell'approccio iterativo è la stesura del modello di progetto ovvero dei diagrammi che descrivono la progettazione logica. Il diagramma delle classi rappresenta la struttura statica del sistema TeamManager, evidenziando le principali entità coinvolte in tutte le iterazioni.



Per una migliore consultazione del diagramma andare nella cartella *Appendice A file – Diagramma delle classi*.

3.2 Diagrammi di sequenza

La stesura del modello di progetto comprende anche i diagrammi di sequenze delle varie operazioni che rappresentano la progettazione logica dal punto di vista dinamico. I vari diagrammi possono essere consultati nella cartella *Elaborazione 1 file – iterazione 1* per l’iterazione 1, *paragrafo 3.2*, nella cartella *Elaborazione 2 file – iterazione 2 paragrafo 3.2* per l’iterazione 2 e nella cartella *Elaborazione 3 file – iterazione 3 paragrafo 3.2* per l’iterazione 3. Nelle varie cartelle di *Elaborazione* sono disponibili nella sezione *schemi* tutti i diagrammi nel formato astah.

4.0 TESTING

4.1 Introduzione

Dopo aver progettato il codice, si è proceduto con la verifica della correttezza delle classi e dei metodi implementati attraverso la pratica del **Testing**; utilizzando un approccio con accesso “**White-Box**”, è stato possibile testare direttamente il codice senza ausilio di un’interfaccia grafica, utilizzando il framework **JUnit4**. Tramite JUnit4, è stato possibile creare classi di test associate a quelle implementate, le quali hanno permesso di testare il codice utilizzando parametri “fittizi” e svolgere test di tipo **unitario** per le iterazioni 1, 2 e 3; I test di tipo unitario sono test relativi alla singola unità del programma, nel caso in esame le classi. Lo scopo del testing è stato quello di individuare “bug” all’interno del programma, ovvero comportamenti riscontrati che non soddisfano le aspettative e, inoltre, verificare la robustezza e l’affidabilità del codice, cercando di evidenziare falle in punti critici di alcuni metodi. Più un programma viene Testato minore è la probabilità di incorrere a bug e malfunzionamenti. Nell’ultima iterazione(4) si è passato anche allo svolgimento di test di tipo **funzionali**; Questi test permettono di verificare che il software funzioni completamente in ogni sua forma trattandolo come una scatola nera alla quale per ogni input ci aspettiamo un determinato output.

4.2 Definizione delle classi e metodi da testare

Nella fase iniziale antecedente al testing in JUnit4, è stato oggetto di interesse definire le principali classi e i metodi da testare, omettendo nell'implementazione del testing metodi semplici come *getter* e *setter*; effettuata tale premessa, vediamo i vari test effettuati per iterazione una volta definiti le seguenti classi e i metodi di test:

ITERAZIONE 1

Test unitari

TeamManager

- testSingletonInstance();
- testCreaGiocatore();
- testEliminaGiocatore();
- testCercaGiocatori();

Amichevole e Allenamento

- testAggiungiDisponibilità();

Calendario

- testPianificaAllenamento();
- testPianificaAmichevole();
- testAggiornaEvento();
- testRimuoviEvento();
- testGetEventi();
- testAggiungiDisponibilità();

Disponibilità

- testMetodoCostruttore();

Rosa

- testAggiungiGiocatore();
- testRimuoviGiocatore();
- testModificaGiocatore();
- testCercaGiocatori();
- testGetGiocatori();

I vari Test possono essere consultati in maniera più dettagliata nella cartella *Elaborazione 1 file - iterazione 1 paragrafo 5.2*

ITERAZIONE 2

Test unitari

Calendario

- testAggiungiStatisticaAllenamento();
- testAggiungiStatisticaAmichevole();
- testRimuoviStatistica();
- testVisualizzaStorico();
- testSuggerisciSessioneMirata():
- testConfrontaGiocatoriCompleto():

I vari Test possono essere consultati in maniera più dettagliata nella cartella *Elaborazione 2 file - iterazione 2 paragrafo 5.1*

ITERAZIONE 3

Test unitari

Calendario

- testSuggerisciFormazione;
- testConteggiaEventiFuturi;
- testCalcolaMediaPresenzeAssenzeMeseCorrente;
- testCalcolaMediaStatisticaRosa;

I vari Test possono essere consultati in maniera più dettagliata nella cartella *Elaborazione 3 file - iterazione 3 paragrafo 5.1.*

ITERAZIONE 4

Test funzionali

Sono stati eseguiti i test seguendo un approccio iterativo utilizzando l'approccio black box svolgendo il testing dell'applicazione senza interagire con il codice, accedendo direttamente alle funzioni del programma tramite l'interfaccia grafica (GUI) per valutare passo dopo passo la corretta implementazione dei casi d'uso con i vari scenari di successo principali e alternativi, la corretta implementazione delle eccezioni e delle regole di dominio tramite l'interfaccia GUI.

I vari Test possono essere consultati in maniera più dettagliata nella cartella *Elaborazione 4 file - iterazione 4 paragrafo 5.1.*

5.0 CONCLUSIONI E IMPLEMENTAZIONI FUTURE

5.1 Introduzione alla fase finale

Il processo di verifica del software ha seguito l'approccio iterativo dello sviluppo. Al termine di ogni iterazione, le nuove funzionalità implementate sono state sottoposte a **test di integrazione** per garantirne il corretto funzionamento con il resto del sistema. Parallelamente, sono stati eseguiti **test unitari** sulle classi e sui metodi critici per

validarne la logica interna. Al completamento di tutte le iterazioni, si è proceduto con la fase finale di **Test di Sistema**, descritta di seguito, per validare il prodotto nella sua interezza

5.2 Test di sistema

La fase di test di sistema ha avuto l'obiettivo di verificare la correttezza globale dell'applicazione e la sua aderenza ai requisiti definiti. Gli sviluppatori hanno eseguito test di tipo **black-box**, interagendo con l'applicazione esclusivamente attraverso l'interfaccia grafica (GUI) per simulare l'esperienza di un utente finale. Per condurre le prove, sono stati creati **dati fittizi**, come giocatori, eventi e statistiche di performance, al fine di costruire scenari di test realistici. L'obiettivo era verificare tutti i casi d'uso sviluppati durante le iterazioni, assicurandosi che il flusso di dati e le interazioni tra le diverse componenti funzionassero come previsto.

5.3 Implementazioni future

Dopo aver gettato le basi con una solida modellazione del dominio, l'applicazione strategica dei design pattern e la definizione rigorosa delle logiche di controllo (come le guardie), ci troviamo di fronte a un sistema con una struttura chiara e un comportamento prevedibile. Tuttavia, un software davvero utile non si limita a elaborare dati in memoria; deve anche **memorizzare queste informazioni in modo permanente** e renderle disponibili nel tempo e tra diverse sessioni di utilizzo. Questo ci porta al cruciale concetto di **persistenza dei dati** e all'utilizzo dei database. Le future implementazioni, quindi, si concentreranno sull'estensione delle funzionalità e sull'introduzione di nuove capacità, sempre tenendo a mente la manutenibilità e la scalabilità, con la persistenza come pilastro fondamentale. Verrà anche implementato il Caso d'uso UC5 invia notifiche per permettere l'invio di notifiche in tempo reale,

ampiamente discusso nei vari casi d'uso già trattati, rendendo l'applicazione maggiormente versatile e improntata verso il futuro.