|  |
| --- |
| Progetto di Ingegneria del Software 2015/16  U_F_L.jpg  **U.F.L.**  University Fanta League   * **System Design Document -** |
| http://www.fisica.unisa.it/CentroQualita/images/logoUnisa.gif |

**Coordinatore del progetto:**

|  |  |
| --- | --- |
| Nome | Matricola |
| Prof. De Lucia Andrea |  |

**Partecipanti:**

|  |  |
| --- | --- |
| Nome | Matricola |
| Foresta Stefano | 0512100504 |
| Franzese Gennaro | 0512100270 |
| Paglialonga Giuseppe | 0510200478 |

|  |  |
| --- | --- |
| **Scritto da:** | Stefano Foresta, Giuseppe Paglialonga, Gennaro Franzese |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versione | Descrizione | Autore |
| 28/11/2015 |  | Prima stesura, impostazione template | Stefano Foresta |
| 29/11/2015 |  | Architettura del sistema proposto | Gennaro Franzese |
| 02/12/2015 |  | Servizi sottosistemi, architettura del sistema e revisione | Giuseppe Paglialonga |

*Indice*

# Introduzione

## Scopo del sistema

## Design goals

### Criteri di prestazione

### Criteri di affidabilità

### Criteri di manutenzione

## Overview

# Architettura del sistema

# Architettura del sistema proposto

## Overview

## Decomposizione in sottosistemi

## Hardware/Software mapping

## Gestione dei dati persistenti

## Controllo degli accessi e sicurezza

### Controllo degli accessi

## Gestione del controllo globale

## Boundary conditions

### Configuration

### Exception handling

# Servizi sottosistemi

1. **Introduzione**
   1. **Scopo del sistema**

Qualche settimana prima dell’inizio del campionato, per iniziare il fantacalcio, solitamente ci si incontra per creare una fanta squadra formata dai giocatori di Serie A, le quali in base alle prestazioni durante le partite del campionato permetteranno di stabile quale formazione di giocatore avrà ottenuto la migliore valutazione. In ambito universitario può accadere, soprattutto tra studenti fuori sede, che è difficile incontrarsi e quindi difficile gestire l’inizio del gioco. Questa web application eliminirà il problema della distanza, che rallenta la costituzione di una lega, fornendo all’utilizzatore uno strumento che gli consente di partecipare anche a distanza e al gestore di lega di non dover perdere tempo facendo chiamate per organizzare l’incontro che serve a costituire le squadre e quindi la stessa lega.

* 1. **Design goals**

Il sistema sarà progettato con un’ interfaccia user-friendly in modo da agevolare l’utilizzo a persone che non hanno molta dimestichezza con web-application di questo genere ma più in generale con il computer. Il software sarà ideato in modo da gestire al meglio le risorse del sistema evitando fastidiosi “crash”. Sarà un prodotto utile e semplice nell’interazione che permetterà ai propri utenti di trovarsi davanti un software facile da utilizzare.

* + 1. **Criteri di prestazione**

|  |  |
| --- | --- |
| **Criteri di design** | **Definizione** |
| Tempo di risposta | Il sistema deve essere efficiente, deve garantire una risposta in breve tempo, massimo 15 secondi senza che si verifichino errori. |
| Throughput | Il sistema riesce a soddisfare massimo 20 task contemporaneamente in un tempo prefissato di un minuto. |
| Memoria | La gestione della memoria verrà gestita dal sistema il quale oltre a gestirne la memoria fisica (RAM), gestisce anche la memoria secondaria (Hard Disk). |

* + 1. **Criteri di affidabilità**

|  |  |
| --- | --- |
| **Criteri di design** | **Definizione** |
| Robustezza | Il sistema deve prevedere e gestire eventuali input errati senza bloccare il funzionamento del sistema. Il sistema ha un layer-application che serve ad elaborare i dati inviati dalle varie interfacce. |
| Sicurezza | Per sicurezza, oltre a intendere la privacy del cliente, si intende l’inalterabilità dei dati dovuta da modifiche sbagliate, quindi in caso di modiche, il sistema tramite interfaccia GUI, manderà un messaggio di conferma a colui che utilizza il software. |
| Tolleranza crash | Questo è un aspetto fondamentale, il sistema non può operare in situazioni di errore, pertanto al verificarsi di un crash verrà salvato il lavoro in corso ed il sistema software terminerà la sua esecuzione. E inoltre, salverà i dati in file temporanei per ogni operazione effettuata. |
| Affidabilità | Il sistema oltre alla gestione degli errori deve garantire il funzionamento 24h su 24h tranne nei tempi relativi alla manutenzione ordinaria, che solitamente va fatta durante la pausa di campionato, e straordinaria. |

* + 1. **Criteri di manutenzione**

|  |  |
| --- | --- |
| **Criteri di design** | **Definizione** |
| Estendibilità | Ai fini di uno sviluppo futuro deve essere possibile aggiungere, in un secondo momento, altre funzionalità al sistema senza dover stravolgere il design del sistema. |
| Leggibilità | Il codice deve essere chiaro, quindi i nomi delle variabili e dei metodi che verranno utilizzati devono essere coerenti al contesto d’uso, dove è strutturato, per rendere più veloce una modifica o una estensione del sistema anche da parte di terzi che si avvicinano al progetto in un secondo momento. |

* 1. **Overview**

Il presente documento SDD(System Design Document) si occupa di: Definire gli obiettivi di design del progetto(identificando quali caratteristiche di qualità dovrebbero essere ottimizzate).

Decomporre il sistema in sottosistemi più piccoli che possono essere realizzati da team individuali utilizzando stili architetturali standard. Selezionare le strategie per costruire il sistema quali:

* Strategie hardware/software
* Strategie che si riferiscono alla gestione dei dati persistenti
* Il flusso di controllo globale
* Le politiche di controllo degli accessi
* La gestione delle condizioni limite

L’output del system design è un modello del sistema che include la decomposizione del sistema in sottosistemi e una chiara descrizione di ognuna delle strategie.

1. **Architettura del sistema**

L’architettura attualmente proposta non va a sostituire nessuna architettura esistente, il sistema è alla sua prima versione.

1. **Architettura del sistema proposto**
   1. **Overview**

Il software è scomposto secondo lo schema client/server:

* **Client:** il browser preferito del cliente.
* **Server:** le pagine PHP che implementano l'intero sistema.

Sono possibili tre configurazioni:

* **Single-Tier:** tutto il sistema è distribuito ed eseguito su un’unica macchina, che fungerà sia da client che da server.
* **Two-Tier:** il sistema è distribuito ed eseguito su due macchine differenti, una eseguirà sia il server sia il database, mentre la seconda macchina eseguirà il client.
* **Three-Tier:** a differenza della configurazione precedente, il database verrà eseguito da una macchina separata.

La configurazione scelta è la seconda (Two-Tier). Essa è basata su tre layer:

1. **Interface Layer:** oggetti con cui l’utente interagisce, quali form, pulsanti, finestre, ecc.
2. **Logic Application Layer:** livello che si occupa di gestire e contenere le operazioni che gli utenti richiedono al sistema.
3. **Storage Layer:** livello che gestisce la memorizzazione dei dati persistenti.
   1. **Decomposizione in SottoSistemi**

Ogni layer contiene al suo interno diversi moduli, di seguito si presenta una loro breve descrizione:

**Interface layer**

**Logic application layer**

**Storage layer**

Di seguito sono specificate, per ogni modulo, le componenti da cui è formato, che rappresentano nel dettaglio le operazioni di cui si occupa e i dati su cui operano. Tutti i moduli specificati fanno parte dello strato “Logic application layer”:

**Interface Layer**

|  |  |
| --- | --- |
| MODULO | DESCRIZIONE |
| **Interfaccia web** | Modulo che rappresenta l’interfaccia grafica web con cui l’utente interagirà con il sistema inoltrando le richieste. |

**Logic application layer**

|  |  |
| --- | --- |
| **MODULO** | **DESCRIZIONE** |
| **Gestione Utente** | Modulo che si occupa della creazione, modifica e  eliminazione dell’account utente. |
| **Gestione Leghe** | Modulo che si occupa della gestione delle leghe,inserimento squadre, aggiornamento classifica |
| **Gestione Giornata** | Modulo che si occupa delle formazioni dei giocatori dell’utente, dell’inserimento dei voti da parte dell’amministratore e del calcolo e aggiornamento delle classifiche. |
| **Gestione squadra** | Modulo che permette all’utente di realizzare e aggiornare la rosa dei giocatori della squadra |
| **Gestione mercato** | Modulo che permette la gestione delle modifiche alle squadre, consente o meno la modifica delle squadre da parte degli utenti, aggiunge giocatori al database, modifica i valori dei giocatori |
| **Informazioni** | Modulo che permette agli utenti di visualizzare informazioni sugli utenti iscritti , le squadre e le classifiche |
| **Calendario** | Modulo che permette la visualizzazione del campionato in merito agli incontri che sono stati calendarizzati in precedenza |
| **Storage** | Modulo che si occupa della comunicazione con il database |
|  | |

**Storage application**

|  |  |
| --- | --- |
| **Modulo** | **Descrizione** |
| **Database** | Modulo che inoltra le richieste di recupero dati direttamente al database fisico e ne recupera i dati, per poi restituirli allo strato delle applicazioni logiche. |

****

* 1. **Hardware/Software mapping**

Il software utilizza un’architettura Two-Tier in configurazione Client-Server, per Client si intende la macchina utilizzata dall’utente dove gira il browser da lui utilizzato. Il lato Server contiene gli script PHP per la logica dell’applicazione e il database per la gestione dei dati persistenti. Il software è suddivisibile in strati logici o layers:

* **Interface layer:** è responsabile della visualizzazione del contenuto richiesto dall’utente e della gestione dell’input utente;
* **Logic application layer:** è responsabile della logica di controllo di sistema, si occupa inoltre di elaborare i dati in base alle funzionalità richieste dall’utente. Infine, fa da tramite tra l’interfaccia utente (quindi le richieste dell’utente e i suoi input) e il livello di storage (quindi il database che contiene tutti i dati relativi alla web application)
* **Storage application:** è il responsabile dell’interfacciamento con il database.
  1. **Gestione dei Dati Persistenti**

La scelta dei dati persistenti è stata fatta analizzando tutti gli oggetti rilevanti che in caso di chiusura controllata/incontrollata dovranno essere conservati.

**Lega:** Memorizza l’anno del campionato corrente, le informazioni sulle Leghe di Facoltà, e la classifica dell’ateneo.

**Calendario:** memorizza le date in cui saranno disputate le giornate di lega.

**Giornata:** memorizza i dettagli relativi alla giornata che compone il calendario e il riferimento alle pagelle della stessa giornata, se presenti.

**LegaFacoltà:** memorizza le classifiche relative alla lega scelta.

**ClassificaFacoltà:** memorizza le classifiche relative alle leghe-facoltà.

**ClassificaAteneo:** memorizza la classifica d’ateneo.

**Formazione:** memorizza l’elenco di giocatori che disputerà la giornata di lega.

**FantaSquadra:** memorizza il nome e l’elenco di giocatori che compongono la squadra dell’utente.

**FantaGiocatori:** memorizza informazione relative al fantagiocatore che comporrà la rosa.

**ArchivioGiocatori:** memorizza tutti i giocatori che possono concorrere nella lega.

**PagellaGiocatori:** memorizza il voto di giornata dei giocatori che concorrono nella lega.

**FantagiocatoreVoto:** memorizza le informazioni relativo al voto di giornate di lega di un fantagiocatore.

**Fantamercato:** memorizza lo stato del fantacalciomercato in cui gli utenti possono acquistare giocatori.

**Utente:** memorizza le informazioni degli utenti iscritti.

**Admin:** memorizza la password dell’amministratore.

**Lega**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| LegheFalcoltà | Chiave esterna, rappresenta le leghe presenti nell’ateneo, di riferimento all’attributo “id\_nome” della tabella “LegaFacoltà” |
| Classifica\_ateneo | Chiave esterna che si riferisce a “id\_classifica” nella tabella ClassificaAteneo |
| Calendario | Chiave esterna che si riferisce all’attributo “id\_calendario” della tabella Calendario |
| fantacalciomercato | Chiave esterna che si riferisce all’attributo “id\_fantamercato” della tabella “FantaMercato” |
| utenti |  |

**Tabella Calendario**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| id\_calendario | Chiave Primaria di Calendario |
| Giornate | Chiave esterna che rappresenta le giornate di lega, di riferimento all’attributo “num\_giornata” della tabella “Giornata” |

**Tabella Giornata**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| num\_Giornata | Numero della giornata di lega e chiave primaria di “Giornata” |
| Data\_giornata | Data della giornata di lega |
| pagelleGiornata | Chiave esterno di riferimento all’attributo “id\_pagelle\_giocatori” della tabella “PagellaGiocatori” |

**Tabella LegaFacoltà**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| id\_nome | Chiave primaria e nome della lega. |
| classifica | Chiave esterna di riferimento all’attributo “id\_classifica\_lega” della tabella ClassificaLega |
| Formazioni | Chiave esterna di riferimento all’attributo “id\_formazione” della tabella “Formazione” |

**Tabella Formazione**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| idFormazione | Chiave Primaria di FORMAZIONE |
| dataora\_di\_consegna | Data e ora di consegna della formazione |
| modulo | Modulo della formazione consegnata |
| Giocatori | Giocatori che comporranno la fantasquadra a seconda dello schema selezionato, riferisce all’attributo “id\_fantagiocatore” della tabella “FantaGiocatore” |
|  |  |

**Tabella ClassificaFacoltà**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| Id\_classifica\_lega | Chiave primaria e nome della lega. |
| fantasquadre | Chiave esterna che si riferisce a “id\_fantasquadra” nella tabella Fantasquadra |

**Tabella ClassificaAteneo**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| Id\_classifica | Chiave primaria della classificaAteneo |
| fantasquadre | Chiave esterna che si riferisce alle squadre che, essendo in prima posizione nella classifica della loro facoltà, concorrono con gli stessi punti alla classifica ateneo. Si riferisce a “id\_fantasquadra” nella tabella Fantasquadra |

**Tabella Fantasquadra**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| id\_Fantasquadra | Chiave primaria della fantasquadra |
| giocatori | Chiave esterna, rappresente i giocatori che compongono la fantasquadra e si riferisce a “id\_giocatore” nella tabella “FantaGiocatore” |
| punti | Punti collezionati dalla fantasquadra e che servono per la classifica |

**Tabella FantaGiocatore**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| idFantagiocatore | Chiave primaria del fantagiocatore |
| ruolo | Ruolo del fantagiocatore |
| nome | Nome del fantagiocatore. |
| prezzo\_iniziale | Costo precampionato del fantagiocatore |
| prezzo\_attuale | Costo attuale del fantagiocatore |
| voto\_ultima\_giornata | Fantavoto del fantagiocatore |

**Tabella ArchivioGiocatori**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| idArchivio | Chiave esterna di riferimento alle leghe e alle relative facoltà |
| giocatori | Chiave esterna di riferimento all’attributo “idGiocatore” della tabella “Giocatore” |
| num\_giocatori | Numero di giocatori presenti nell’archivio |
| data\_ultimo\_upload | Data dell’ultimo upload |

**Tabella Giocatore**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| idGiocatore | Chiave primaria per Giocatore. |
| ruolo | Ruolo del fantagiocatore |
| nome | Nome del fantagiocatore. |
| prezzo\_iniziale | Costo precampionato del fantagiocatore |
| prezzo\_attuale | Costo attuale del fantagiocatore |
| squadra | Squadra del giocatore |

**Tabella PagellaGiocatori**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| id\_pagella\_giocatori | Chiave primaria della pagella giocatori. |
| Giocatori | Chiave esterna di riferimento all’attributo “id\_fantagiocatore\_voto” della tabella FantagiocatoreVoto |
| giornata\_num | Chiave esterna di riferimento all’attributo “num\_giornata” della tabella “Giornata” |
| prezzo\_iniziale | Costo precampionato del fantagiocatore |
| prezzo\_attuale | Costo attuale del fantagiocatore |

**Tabella FantagiocatoreVoto**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| id\_fantagiocatore\_voto | Chiave primaria della fantagiocatore\_voto. |
| id\_fantagiocatore | Chiave esterna di riferimento all’attributo “id\_fantagiocatore” della tabella Fantagiocatore |
| voto | Fantavoto relativo al giocatore. |
| punti\_assist | Numero di punti da assist |
| punti\_ammonizione\_espulsione | Numero di punti sottratti per ammonizione o espulsione |
| punti\_goal\_subiti | Numero di punti sottratti per goal subiti |
| punti\_rigori\_parati | Numero di punti ricavati per rigori parati |

**Tabella Fantamercato**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| Id\_fantamercato | Chiave primaria per Fantamercato. |
| stato | Rappresenta se attivo che i giocatori possono acquistare o vendere fantagiocatori. |

**Tabella Admin**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| password | Password dell’amministratore |
| email | Email dell’amministrore |

**Tabella Utente**:

|  |  |
| --- | --- |
| **Campo** | **Ruolo** |
| id\_utente | Chiave primaria di “Utente” |
| Nome | Nome dell’utente |
| Cognome | Cognome dell’utente |
| username | Nickname scelto dall’utente in fase di registrazione |
| matricola | Matricola dell’utente |
| facoltà | Facoltà di appartenenza dell’utente |
| email | Email dell’utente |
| fantasquadra | Chiave esterna che rappresenta la fantasquadra dell’utente, di riferimento all’attributo “id\_fantasquadra” della tabella “FantaSquadra” |

**3.5 Controllo degli accessi e sicurezza**

**3.5.1 Controllo degli accessi**

Il sistema prevede tre tipi di utenti:

**Admin**: è l’amministratore del sistema, ha accesso a tutte le sue componenti ed è unico. L’account dell’Admin non può essere modificato da nessuno tranne che da se stesso, inoltre non può essere cancellato in nessun modo. Admin può modificare la password e l’email.

**Utente**: ha accesso a visualizzare le informazione generali della lega, quali classifiche e formazioni con i vari punti ottenuti dai concorrenti. In più può modificare la sua fantasquadra solo quando il fantacalciomercato è attivo.

Gli oggetti presenti nel sistema, che vengono gestiti direttamente da un attore:

**Lega**

**Calendario**

**Giornata**

**LegaFacoltà**

**ClassificaFacoltà**

**ClassificaAteneo**

**Formazione**

**FantaSquadra**

**FantaGiocatori**

**ArchivioGiocatori**

**PagellaGiocatori**

**FantagiocatoreVoto**

**Fantamercato**

**Admin**

**Utente**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Oggetto\Attore** | **Lega** | **Calendario** | **Giornata** | **LegaFacoltà** | **Giornata** | **ClassificaFacoltà** |
| **Amministratore** | Crea  Modifica  Visualizza  Elimina | Crea  Modifica  Visualizza  Elimina | Crea  Modifica  Visualizza  Elimina | Crea  Modifica  Visualizza  Elimina | Crea  Modifica  Visualizza  Elimina | Crea  Modifica  Visualizza  Elimina |
| **Utente** | -  -  -  - | -  -  -  - | -  -  -  - | -  -  -  - | -  -  -  - | -  -  Visualizza  - |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Oggetto\Attore** | **ClassificaAteneo** | **Formazione** | **FantaSquadra** | **FantaGiocatori** | **ArchivioGiocatori** |
| **Amministratore** | Crea  Modifica  Visualizza  Elimina | Crea  Modifica  Visualizza  Elimina | Crea  Modifica  Visualizza  Elimina | Crea  Modifica  Visualizza  Elimina | Crea  Modifica  Visualizza  Elimina |
| **Utente** | -  -  Visualizza  - | Crea  Modifica  Visualizza  - | Crea  -  Visualizza  - | -  -  -  - | -  -  -  - |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Oggetto\Attore** | **PagellaGiocatori** | **Fantagiocatore\_voto** | **Fantamercato** | **Admin** | **Utente** |
| **Amministratore** | Crea  Modifica  Visualizza  Elimina | Crea  Modifica  Visualizza  Elimina | -  Modifica  -  - | -  Modifica  Visualizza  - | Crea  Modifica  Visualizza  Elimina |
| **Utente** | -  -  -  - | -  -  -  - | -  -  -  - | -  -  -  - | Crea  -  Visualizza  - |

**Sicurezza**

Il software assicura la protezione dei dati tramite un controllo degli accessi al sistema. All’utente del sistema viene assegnata username ed una password.

Il software garantisce che non sia possibile che un utente non registrato possa accedere al sistema. Per accedere al sistema, l’utente deve possedere un account e solo gli addetti hanno una visione globale dei dati impedendo ai clienti la visione dei dati di altri clienti. Il sistema, ad ogni sessione chiede all'utente l'inserimento dei suddetti dati e ne verifica la correttezza. In caso di inserimento dati non corretti, il sistema mostrerà nuovamente a video la schermata di login. Una volta verificati i dati, verrà mostrata all'utente la schermata relativa alle funzionalità a cui può accedere. La sicurezza dell'accesso dipende fortemente dalla lunghezza e dal tipo della password scelta dall’utente.

UserID e password dell'account devono rispettare i canoni riportati di seguito:

• UserID: deve avere una lunghezza compresa fra un minimo di 7 e un massimo di 12 caratteri. Può contenere caratteri alfanumerici, ma non caratteri speciali.

• Password: deve avere una lunghezza compresa fra un minimo di 7 e un massimo di 12 caratteri alfanumerici. Come per UserID, non deve contenere caratteri speciali e inoltre deve essere diversa dalla UserID.

**3.6 Gestione del controllo globale**

Il flusso di controllo del sistema software in questione è gestito da un “event driven control”. Il flusso principale attende il verificarsi di eventi che attivano funzionalità specifiche. Al verificarsi di un evento, il più delle volte generato dall’ utente tramite l'interfaccia grafica, il controllo viene trasferito al relativo oggetto che lo gestisce. Il suddetto oggetto ricevuto il controllo manda in esecuzione un sottosistema.

La motivazione di questa scelta risiede nel fatto che questo meccanismo di flusso è particolarmente flessibile ed adatto alle interfacce utenti, e sfrutta appieno le potenzialità del linguaggio di programmazione usato per sviluppare questo software.

**3.7 Boundary conditions**

**3.7.1 Configuration**

Qui di seguito vengono definite le fasi in cui gli oggetti vengono creati, resi persistenti (archiviati nel database) o distrutti:

**Lega:** Viene inizializzata annualmente da admin e viene archiviata ogni fine lega insieme a tutto il suo contenuto dal sistema.

**Calendario:** Viene creato da admin nella fase pre-campionato e viene memorizzato a fine lega dal sistema.

**Giornata:** Si riferisce a una giornata che compone il calendario.

**LegaFacoltà:** viene avviata dall’amministratore e viene archiviata una settimana dopo l’ultima giornata di lega;

**ClassificaFacoltà:** viene inizializzata dopo la prima giornata di lega dal sistema a seconda degli iscritti e viene aggiornata di giornata in giornata. Anche gli Utenti possono visualizzare le classifiche, comprese quelle degli anni precedenti. Viene archiviata quanto si comincia una nuova lega, l’anno successivo

**ClassificaAteneo:** viene creata dopo la prima giornata di lega e viene aggiornata con i primi classificati delle leghe facoltà, di giornata in giornata. I punti per la classifica ateneo verranno sommati ogni qualvoltà si finirà una giornata di lega facoltà in prima posizione.

**Formazione:** viene creata dall’utente selezionando i fantagiocatori, da una FantaSquadra.

**FantaSquadra:** viene creata dall’utente acquistando i Giocatori, nella fase pre-campionato. Le modifiche possono essere effettuate fino alla prima giornata di lega e in qualunque momento l’amministratore attiva il Fantamercato. Viene archiviata all’inizio di una nuova lega.

**FantaGiocatori:** vengono acquistati nella fase di Fantamercato e compongono la fantasquadra.

**ArchivioGiocatori:** archivioGiocatori contiene l’elenco di tutti i giocatori reali e che gli utenti potranno acquistare tramite il fantamercato.

**PagellaGiocatori:** viene caricata tramite un file dall’amministratore dopo ogni giornata di lega e aggiorna i punteggi di tutte le fantasquadre nel sistema, aggiornando di conseguenza anche i punti delle classifiche. Rimangono consultabili sia dall’utente che dall’amministratore. Vengono archiviati a inizio nuova lega.

**FantagiocatoreVoto:** è il voto che viene caricato attraverso PagellaGiocatori e che viene assegnato a ogni Fantagiocatore per quella giornata. Vengono archiviati a inizio nuova lega.

**Fantamercato:** è la oggetto che se attivato consente agli utenti di modificare le proprie fantasquadre.

**Admin:** rappresenta l’amministratore del sistema e serve a conservare l’email dell’amministratore e la password. Non viene mai archiviato in quanto è l’utente indispensabile per il funzionamento.

**Utente:** rappresenta l’Utente e tutti i suoi dati, da questi confermati in fase di registrazione.  
Può essere bannato dall’amminstratore o cancellato oppure viene eliminato se in archivio da più di 3 campionati ma non partecipe alle leghe annuali.

**3.7.2Exception handling**

1. **Connessione al database fallita**: Errore del sistema durante la connessione al database. Impossibilità di accedere ai dati e impossibilità di inserimento dati.

2. **Interruzione inaspettata erogazione di corrente**: Mancanza erogazione corrente, crash del sistema e conseguente perdita di dati.

**4. Servizi sottosistemi**

Descriviamo ora i servizi forniti da ognuno dei sottosistemi in termini di operazioni

|  |  |
| --- | --- |
| **Modulo** | **Gestione Utente** |
| **Descrizione** | Sottosistema che gestisce le operazione  sull’account |
| **Servizi Offerti** | |
| **Servizio** | **Descrizione** |
| Login | Il servizio permette all’utente di accedere al sistema |
| Logout | Il servizio permette all’utente di disconnettersi dal sistema e permette all’utente di loggarsi nuovamente |
| Registrazione | Il servizio permette all’utente di essere inserito nel database del sistema. |
| Modifica | Il servizio permette di modificare la password dell’utente |
| Recupera Password | Il servizio permette di recuperare la password dell’utente |
|  | |
| **Gestione Leghe** | Sottosistema che gestisce le operazioni  Sulle leghe |
| **Servizi Offerti** | |
| **Servizio** | **Descrizione** |
| Avvia lega | Il servizio permette all’amministratore di avviare le leghe |
| Inserimento calendario | Il servizio permette all’amministratore di caricare le date in cui si terranno gli incontri sel campionato |
|  | |

|  |  |
| --- | --- |
| **Gestione Giornata** | Sottosistema che gestisce le operazioni per gli eventi del campionato |
| **Servizi Offerti** | |
| Inserisci Formazione | Il servizio permette all’utente di stilare la lista dei calciatori da utilizzare durante la giornata in corso |
| Inserisci Voti | Il servizio permette all’amministratore di inserire i voti assegnati ai singoli calciatori |
| Calcola classifiche | Il servizio permette al sistema di calcolare i voti ottenuti dalle squadre in base ai voti dei singoli giocatori e di aggiornare le classifiche di lega |

|  |  |
| --- | --- |
| **Gestione Squadra** | Sottosistema che gestisce la realizzazione delle squadre da parte degli utenti |
| **Servizi Offerti** | |
| Realizza rosa | Il servizio permette all’utente di stilare la lista completa dei calciatori da poter utilizzare durante il fantacalcio |
| Rimuovi Giocatori | l servizio permette durante il calciomercato all’utente di rimuovere dei giocatori dalla sua lista |
| Aggiungi Giocatori | l servizio permette durante il calciomercato all’utente di aggiungere dei giocatori alla sua lista |

|  |  |
| --- | --- |
| **Gestione Mercato** | Sottosistema che gestisce le modifiche delle squadre e del valore dei giocatori |
| **Servizi Offerti** | |
| Apri Mercato | Il servizio permette all’amministratore di permettere che gli utenti realizzino le loro rose e possano modificare i giocatori delle loro squadre |
| Carica giocatori | Il servizio permette all’amministratore di caricare la lista dei giocatori con i rispettivi valori nel database |
| Chiudi Mercato | Il servizio permette all’amministratore di impedire che gli utenti realizzino le loro rose e possano modificare i giocatori delle loro squadre |
| Aggiungi giocatori | Il servizio permette all’amministratore di aggiungere giocatori la lista dei giocatori acquistabili |

|  |  |
| --- | --- |
| **Gestione Informazioni** | Sottosistema che permette agli utenti di visualizzare informazioni sugli utenti iscritti , le squadre e le classifiche |
| **Servizi Offerti** | | |
| Visualizza classifica | Il servizio permette agli utente di visualizzare le classifiche delle varie leghe | |
| Visualizza utenti | Il servizio permette agli utente di visualizzare le squadre degli utenti iscritti alle leghe | |
|  |  | |