**Relazione finale**

**PROGRAMMA GESTIONALE DEL FANTACALCIO**

**Analisi dei requisiti**

**La situazione iniziale**

Viene richiesta la creazione di un programma per computer desktop che permetta la gestione dei tornei di Fantacalcio.

**Tempi**

Per l’esecuzione del progetto sono stati definiti dei tempi precisi di consegna:

* Consegna delle analisi dei requisiti, funzionale e tecnica entro il 22/10/2021;
* Consegna del codice sorgente del programma e del file eseguibile entro il 04/11/2021.

**Scopo e funzionamento del gioco**

Il **Fantacalcio** è un gioco che simula partite e tornei utilizzando le gesta dei veri giocatori del campionato. A questo gioco possono giocare un minimo di due giocatori (che vengono chiamati “**fanta-allenatori**”), fino ad arrivare ad un massimo di 10.  
Lo scopo è quello di formare una “**fanta-squadra**”, cioè una squadra immaginaria, formata dai giocatori di un torneo reale, come la **Serie A**, la **Champions League** oppure la **Premier League inglese**, per poi accumulare punteggi in base alle prestazioni dei calciatori che sono stati scelti.  
Il primo momento che sancisce l'inizio del **Fantacalcio** è l’asta, durante la quale i partecipanti del gioco formano le proprie squadre. L’asta prevede diversi fondamentali passaggi:

1. Prima di tutto è necessaria una lista contenente l'elenco di tutti i giocatori del campionato di riferimento. In queste tabelle ad ogni calciatore è associato un nome, un cognome, una squadra di appartenenza, un numero di maglia, un ruolo (che può essere attaccante, difensore, portiere e centrocampista) e un numero, detto quotazione iniziale, che ne rappresenta il valore "**fantacalcistico**".
2. Ogni **fanta-allenatore** parte con un budget stabilito e uguale per tutti (per esempio 260 Fantamilioni). Tale somma serve per acquistare i giocatori e formare la squadra. Un giocatore può giocare in una sola squadra.
3. Viene dato inizio alla fase più importante dell’asta: i partecipanti si sfidano a colpi di offerte, che non hanno limiti nei rilanci, per aggiudicarsi i giocatori e chi offre di più per un determinato calciatore, se lo aggiudica in esclusiva.
4. Alla fine dell'asta ogni partecipante deve avere a disposizione una “**rosa**” di **25 giocatori**: **3 portieri**, **8 difensori**, **8 centrocampisti** e **6 attaccanti**.

Completate le squadre, inizia il gioco, che durerà per tutta la stagione calcistica.  
Ogni giornata di campionato reale, ovvero quello giocato negli stadi, i **fanta-allenatori** schierano una squadra composta da 11 calciatori scelti dalle **rose** a loro disposizione, più le **riserve** (al massimo sette) che potranno sostituire eventuali calciatori che non scenderanno in campo o che per i motivi più disparati non prenderanno voto. Le prestazioni dei calciatori scelti determineranno l'andamento della squadra.  
I punteggi dei giocatori sono dati dai voti delle pagelle individuali che siti e quotidiani sportivi stilano al termine delle partite. A questi voti poi si aggiungono dei **bonus** o dei **malus** che aumentano o diminuiscono il punteggio totale della prestazione del calciatore:

* +3 punti per ogni **gol** segnato;
* +3 punti per ogni **rigore parato** (portiere);
* +2 punti per ogni **rigore segnato**;
* +1 punto per ogni **assist** effettuato;
* -0,5 punti per ogni **ammonizione**;
* +1 **portiere** **imbattuto**;
* -1 punto per ogni **gol** **subito** dal portiere;
* -1 punto per ogni **espulsione**;
* -2 punti per ogni **autorete**;
* -3 punti per un **rigore sbagliato**.

Alla fine del campionato reale, vince chi avrà accumulato più punti nella classifica generale.

**Scopo del programma**

Lo scopo del programma non è quello di fornire ai fanta-allenatori il gioco vero e proprio del Fantacalcio, bensì di permettere la registrazione dei dati di gioco, assumendo la forma di programma gestionale: molte delle parti del gioco, come l’asta, devono essere gestite autonomamente dai fanta-allenatori e poi inserite nel programma, cercando quindi di migliorare notevolmente la gestione dei dati per loro.

**Chi deve usare il programma?**

Il programma può essere utilizzato da un solo utente di qualunque tipo che abbia compreso appieno il funzionamento del gioco, illustrati all’interno dell’analisi funzionale.

**Requisiti software**

Il programma è un’applicazione basata su .NET Core 3.1, compatibile con Windows, MacOS e Linux.

**Requisiti hardware necessari**

Il programma, essendo estremamente semplice, necessita, ovviamente, di un PC e di una tastiera per l’inserimento dei comandi da tastiera.

**Analisi funzionale**

**Premessa**

Per realizzare il programma gestionale del Fantacalcio non si utilizzerà una GUI (Graphical User Interface), ma un’interfaccia di tipo CLI (Command Line Interface, di tipo console), offerta da .NET Core. Di seguito verrà elencato il funzionamento previsto per il programma, anche se non è una versione definitiva, quindi ci potranno essere dei cambiamenti più o meno importanti nel corso dello svolgimento del progetto.   
Successivamente, quando si dirà “il programma richiede l’inserimento”, si intende che l’utente debba inserire un comando da tastiera, composto da una o più lettere, e poi deve premere invio per confermare l’inserimento.  
Per capire il funzionamento del Fantacalcio oppure i requisiti hardware e software del programma, fare riferimento all’analisi dei requisiti. Per vedere quali sono le classi, i metodi e ogni parte tecnica, consultare l’analisi tecnica.

**Schermata iniziale**

All’avvio del software viene presentata una schermata iniziale, nella quale vengono fornite all’utente alcune indicazioni di base su come funziona il programma. In base al momento in cui ci si trova e alle richieste del software in questa schermata è possibile distinguere due situazioni differenti.

* Se è il primo avvio, cioè nessun campionato di Fantacalcio è stato ancora impostato, il programma richiede all’utente l’inserimento, innanzitutto, del numero di fanta-allenatori che partecipano al gioco. Successivamente chiede l’inserimento del torneo calcistico reale scelto dai partecipanti. Una volta compiuto quest’ultimo passaggio, la console viene pulita e si passa alla schermata di inserimento delle rose dei giocatori.
* Se è un avvio comune, invece, in cui sono già stati configurati i dati di gioco, viene visualizzato un menu di scelta tra diverse opzioni:
  + Visualizzare la lista dei fantagiocatori inseriti, insieme alle loro caratteristiche (inserendo il comando **statistiche**);
  + Visualizzare lo schieramento oppure schierare in campo i giocatori per una partita (con il comando **schieramento**);
  + Aggiornare le statistiche dei giocatori (con il comando **aggiorna**);
  + Visualizzare la classifica parziale (con il comando **classifica**);
  + Cancellare i dati del torneo di Fantacalcio (con il comando **cancella**);
  + Visualizzare la schermata dei comandi (attivabile inserendo il carattere “**?**”);
  + Uscire dal programma (con il comando **esci**).

**Schermata di inserimento delle rose dei giocatori**

La schermata di inserimento delle rose dei giocatori non è attivabile direttamente dalla schermata iniziale con un comando in caso di avvio comune, ma solamente nel caso di un primo avvio. Supponendo che i fanta-allenatori si siano procurati manualmente le liste dei fantagiocatori e che abbiano già provveduto a fare l'asta, il programma chiederà prima l’inserimento del budget rimasto a ogni fanta-allenatore, poi delle rose dei giocatori: da parte del software verrà effettuata una suddivisione automatica in base ai ruoli (portiere, difensore, centrocampista e attaccante), quindi l'utente dovrà inserire manualmente il nome, il cognome, la squadra di appartenenza, il numero di maglia e la quotazione iniziale. Per evitare che più giocatori inseriscano lo stesso giocatore, verranno eseguiti dei controlli informatici che chiederanno nuovamente l'inserimento nel caso in cui si verifichi questa situazione.   
Alla fine, quando ogni fanta-allenatore avrà inserito tutti i suoi fantagiocatori, le rose verrano riepilogate. Se i fanta-allenatori notano errori di inserimento, allora possono riavviare l’inserimento con il comando **reinserisci**, altrimenti inserendo il comando **salva** si procede al salvataggio della configurazione inserita su file e il programma ritorna automaticamente alla schermata iniziale.

**Schermata di ricerca/visualizzazione dei fantagiocatori**

Una volta che tutti i passaggi iniziali sono stati completati, tramite la schermata iniziale è possibile accedere alla schermata dei fantagiocatori. Qui è possibile eseguire una ricerca dei fantagiocatori inserendo, come filtri, le caratteristiche dei giocatori: nome, cognome, squadra, numero di maglia, ruolo e quotazione.  
I filtri si possono saltare non inserendo nulla e premendo direttamente **invio**. Successivamente viene visualizzata la lista con tutti i fantagiocatori che corrispondo ai filtri, insieme alle loro caratteristiche e il nome del fanta-allenatore che li possiede. Una volta finita la consultazione si può tornare alla schermata iniziale con il comando **menu**.

**Schermata di schieramento in campo**

Sempre dopo aver completato l’inserimento dei dati iniziali, è possibile accedere alla schermata di schieramento in campo. Come prevede il regolamento del fantacalcio, è necessario che ogni fanta-allenatore schieri in campo undici giocatori ogni qualvolta che avvenga una partita reale, più eventuali riserve, che vengono scelti dalle proprie rose. Anche qui le situazioni che si possono presentare sono due.

* Se gli schieramenti non sono ancora stati impostati, allora il programma visualizza la rosa del primo fanta-allenatore, ordinata numericamente: il fanta-allenatore inserisce così i numeri che corrispondono a ogni giocatore che preferisce divisi da un trattino (del tipo 1-3-8-9-ecc…). Il procedimento viene ripetuto per ogni fanta-allenatore. Anche qui verranno eseguiti i controlli informatici per evitare che venga inserito lo stesso giocatore più volte.
* Se lo schieramento è stato già impostato, invece, il programma lo visualizza e chiede all’utente se vuole cambiarlo (comando **cambia**) oppure se vuole tornare alla schermata iniziale (comando **menu**): nel primo caso, chiede conferma all’utente se vuole eliminare o meno gli schieramenti (l’utente può rispondere con **si** o **no**) e, in caso affermativo, avvia la procedura già esplicata nella prima situazione, ovvero quella in cui nessun schieramento è ancora stato impostato.

**Schermata di aggiornamento delle statistiche**

La schermata dei aggiornamento delle statistiche permette di aggiornare i punteggi di ogni fantagiocatore al termine di ogni partita reale del torneo scelto. Attraverso la funzione di ricerca già spiegata prima, sì cerca un fantagiocatore, sì inserisce il suo numero, e, in base ai bonus/malus scelti, sì aumenta (inserendo +*punteggio*) o se ne diminuisce il punteggio (inserendo -*punteggio*). Una volta inserito il nuovo punteggio, il programma chiede se si vuole salvare la nuova statistica; in caso sia affermativo che negativo, chiede poi se si vogliono cambiare le statistiche si altri giocatori: con il comando **cambia** si ritorna alla ricerca del fanta-giocatore, mentre con il comando **menu** si torna alla schermata iniziale.

**Schermata di visualizzazione delle classifiche parziali**

Questa schermata permette di visualizzare la classica parziale ordinata in modo decrescente in base al punteggio ottenuto da ogni fantacalciatore nel torneo di Fantacalcio. Dopo aver completato la lettura, si può tornare alla schermata iniziale con il comando **menu**.

**Schermata di cancellazione dei dati**

La schermata di cancellazione dei dati permette di far ritornare il programma allo stato iniziale, cancellando ogni dato inserito sul torneo di Fantacalcio: questo tipo di operazione può rendersi necessario nel caso in cui il torneo sia terminato, oppure se i fanta-allenatori scelgono di terminare il torneo prima della sua naturale scadenza. Il programma chiederà la conferma all’utente se vuole proseguire l’operazione di eliminazione (l’utente può rispondere con **si** o **no**): in caso affermativo cancella i dati e riavvia l’applicazione, che visualizzerà una schermata iniziale da primo avvio.

**Schermata dei comandi**

Questa schermata non fa altro che elencare i comandi necessari per il funzionamento del programma nel caso in cui un utente ne abbia bisogno. Dopo aver completato la lettura, si può tornare alla schermata iniziale con il comando **menu**.

**Uscire dal programma**

Il software può essere chiuso dalla schermata iniziale con il comando **esci**, oppure con l’icona presenta nella finestra: se si esce dal programma in quest’ultimo modo mentre sono in esecuzione operazioni di salvataggio, oppure nel caso in cui l’utente non abbia terminato un inserimento, allora si rischia di perdere i dati, visto che il salvataggio non viene effettuato per ogni inserimento.

**Premessa**  
Il programma avrà un’interfaccia grafica di tipo CLI (Command Line Interface), realizzata con .NET Core 3.1, attraverso l’uso del linguaggio di programmazione C#. Per realizzare il programma si userà l’ambiente di sviluppo Microsoft Visual Studio 2019. Di seguito verrà spiegato il funzionamento tecnico del programma, anche se questa non è una versione definitiva, quindi ci potranno essere dei cambiamenti più o meno importanti nel corso dello svolgimento del progetto.   
Per capire il funzionamento del Fantacalcio oppure i requisiti hardware e software del programma, fare riferimento all’analisi dei requisiti. Per vedere come funziona il software a livello utente, consultare l’analisi funzionale.

**Analisi tecnica**

**Direttive *using***

Le principali direttive *using* che verranno utilizzate saranno molto probabilmente le seguenti:

* *using* **System**, direttiva che permette l’uso dei metodi più comuni;
* *using* **System.Threading.Tasks**, utilizzata per l’uso di metodi che posticipano le istruzioni;
* *using* **System.Linq**, che permette l’uso di specifici metodi per il confronto di stringhe;
* *using* **System.IO**, fondamentale per le operazioni sui file;
* *using* **System.Collections.Generic**, necessaria per la creazione di liste.

**Paradigma di programmazione**

Il codice verrà redatto seguendo la **OOP**, ovvero programmazione orientata agli oggetti, che prevede la presenza di:

* ***classi***, che hanno il compito di definire le caratteristiche di un insieme di oggetti che hanno proprietà e compiono azioni uguali;
* ***attributi***, cioè delle variabili e/o costanti che definiscono le caratteristiche o proprietà degli oggetti instanziabili invocando la classe; i valori inizializzati degli attributi sono ottenuti attraverso il cosiddetto costruttore;
* ***metodi***, ossia procedure che operano sugli attributi.

Gli oggetti, quindi, sono istanze delle classi.

**Classi**

Le classi che compongono il codice del programma, insieme ai loro metodi e attributi, sono le seguenti.

* Classe **Programma**: questa classe viene eseguita automaticamente all’avvio del programma.

Il principale attributo della classe è:

* + **comandoInserito** (string), che contiene il comando inserito dall’utente in quei casi in cui non si stiano inserendo dei particolari dati necessari per l’impostazione del torneo di Fantacalcio.

I metodi della classe, invece, sono:

* **Main**, il metodo a cui il programma fa per primo riferimento in seguito al suo avvio, che contiene fondamentalmente la visualizzazione a schermo dei due casi possibili della schermata iniziale;
* **SchermataInserimentoRose**, che contiene la parte di visualizzazione a schermo della schermata di inserimento delle rose di fanta-giocatori;
* **SchermataVisualizzazioneFantaCalciatori**, che contiene la parte di visualizzazione a schermo della schermata di ricerca dei fanta-giocatori;
* **SchermataSchieramentoCampoFantaCalciatori**, che contiene la parte di visualizzazione a schermo di schieramento in campo dei fanta-giocatori;
* **SchermataAggiornamentoStatisticheFantaCalciatori**, che contiene la parte di visualizzazione a schermo della schermata di aggiornamento delle statistiche dei fanta-giocatori;
* **SchermataVisualizzazioneClassifiche**, che contiene la parte di visualizzazione a schermo della schermata delle classifiche parziali dei fanta-giocatori;
* **SchermataCancellazioneDati**, che contiene la parte di visualizzazione a schermo della schermata di cancellazione dei dati;
* **SchermataComandi**, che contiene la parte di visualizzazione a schermo della schermata dei comandi;
* **UscitaProgramma**, che contiene i metodi per uscire dal programma, tra cui **Environment.Exit()**;
* Classe **Files**: questa classe contiene tutti i metodi necessari per eseguire la lettura e scrittura dei files.   
  Gli attributi della classe sono:
  + **inputOutput** (string), che contiene informazioni su ciò che deve essere letto (input) oppure stampato (output);
  + **percorsoIO** (string), che contiene il percorso assoluto di un file che deve essere letto (input) oppure scritto (output).

I metodi della classe, invece, sono:

* Metodo **costruttore**, che permette l’effettivo passaggio del valore degli attributi di altre classi agli attributi della classe corrente;
* **VerificaEsistenzaFile**, metodo che permette di verificare se esiste o meno il percorso del file che gli è stato passato attraverso **File.Exists()**;
* **CreaCartella**, metodo che viene richiamato quando né i file né la cartella di salvataggio sono presenti; contiene il metodo **Directory.CreateDirectory()**, le cui eccezioni verranno gestite attraverso un **blocco try-catch**;
* **CreaFile**, metodo richiamato successivamente a **CreaCartella**, che permette di creare i file attraverso il metodo **File.Create()**, le cui eccezioni verranno gestite attraverso un **blocco try-catch**;
* **LeggiFile**, metodo che permette, attraverso **File.ReadAllLines()**, la lettura e l’inserimento all’interno delle liste di fanta-calciatori, così come delle rose e degli schieramenti, di ciò che si trova nei files di salvataggio;
* **ScriviFile**, metodo che permette di scrivere su di un file di testo attraverso **File.WriteAllLines()**, le cui eccezioni verranno gestite attraverso un **blocco try-catch**;
* **GestisciErrori**, metodo richiamato nel caso di eccezioni verificatasi nei metodi precedentemente spiegati, che contiene i codici di errore e quale tipo di messaggi da visualizzare nelle varie situazioni.
* Classe **FantaAllenatore**: questa classe permette di accedere ai metodi che riguardano i fanta-allenatori.   
  Gli attributi della classe sono:
  + **torneo** (string), ossia il tipo di torneo reale di riferimento per il Fantacalcio scelto dai fanta-allenatori;
  + **nome** (string), ovvero il nome del fanta-allenatore;
  + **codiceRosa** (string), che indica il numero di rosa che appartiene al fanta-allenatore;
  + **budgetDisponibile** (integer), che indica il numero di Fantamilioni che sono rimasti al fanta-allenatore;
  + **FantaCalciatori** (lista string), che contiene le caratteristiche di ogni fanta-calciatore;
  + **RoseFantaCalciatori** (vettore integer), che contiene gli indici di posizione dei giocatori che si trovano all’interno della lista **FantaCalciatori** e che fanno parte della rosa del fanta-allenatore;
  + **SchieramentiFantaCalciatori** (vettore integer), che contiene gli indici di posizione dei giocatori che si trovano all’interno della lista **FantaCalciatori** e che fanno parte dello schieramento del fanta-allenatore.

I metodi della classe, invece, sono:

* Metodo **costruttore**, che permette l’effettivo passaggio del valore degli attributi di altre classi agli attributi della classe corrente;
* **AggiungiFantaAllenatore**, che permette di aggiungere un fanta-allenatore;
* **AggiungiRosa**, che permette ad ogni fanta-allenatore di creare la propria rosa di fanta-giocatori;
* **AggiungiSchieramento**, che permette ad ogni fanta-allenatore di creare il proprio schieramento giornaliero di fanta-giocatori;
* Classe **FantaCalciatore**: questa classe permette di accedere ai metodi che riguardano i fanta-calciatori. Gli attributi della classe sono:
  + **nome** (string), ovvero il nome del fanta-calciatore;
  + **cognome** (string), ovvero il cognome del fanta-calciatore;
  + **squadra** (string), ovvero la squadra di appartenenza del fanta-calciatore;
  + **ruolo** (string) , ovvero il ruolo fanta-calciatore all’interno della squadra;
  + **numeroMaglia** (integer), ovvero il numero di maglia del fanta-calciatore;
  + **quotazioneIniziale** (integer), ovvero la quotazione iniziale del fanta-calciatore;
  + quotazioneAttuale (integer), ovvero la quotazione attuale del fanta-calciatore;
  + **punteggioClassifica** (integer), ovvero il punteggio del fanta-calciatore nella classifica parziale;
  + **FantaCalciatori** (lista string), che contiene le caratteristiche di ogni fanta-calciatore.

I metodi della classe, invece, sono:

* Metodo **costruttore**, che permette l’effettivo passaggio del valore degli attributi di altre classi agli attributi della classe corrente;
* **AggiungiFantaCalciatore**, metodo che permette di aggiungere un giocatore alle liste di fanta-calciatori, utilizzando specifici metodi di inserimento degli oggetti **List**;
* **RicercaFantaCalciatore**, che permette di ricercare uno o più fanta-giocatori in base ai filtri inseriti dall’utente, utilizzando specifici metodi di ricerca degli oggetti **List**;
* **AggiornaStatisticheFantaCalciatore**, che aggiorna le quotazioni e i punteggi di ogni fanta-calciatore, utilizzando specifici metodi di inserimento degli oggetti **List**;
* **OrdinaFantaCalciatori**, che ordina in modo decrescente i giocatori in base al loro punteggio, utilizzando specifici metodi di ordinamento degli oggetti **List**.

Per concludere, in generale:

* ogni classe può avere degli attributi in più, o in meno, oppure possono essere leggermente differenti rispetto a quelli elencati fino ad ora;
* ogni classe, tranne Programma, contiene il metodo di tipo ovverride (un metodo cioè che ne sovrascrive un altro già esistente) **ToString**, e che ritorna il valore di una stringa costruita da più stringhe, pronta per essere visualizzata a schermo.
* ogni metodo, presenta delle variabili interne necessarie ai fini del funzionamento del programma, per esempio indici o contatori, che non possono essere completamente previsti e, di conseguenza, elencati all’interno di questa analisi.

**Fonti della relazione**

* [https://www.focusjunior.it/news/news-e-curiosita/Fantacalcio-cose-e-come-si-gioca/](https://www.focusjunior.it/news/news-e-curiosita/fantacalcio-cose-e-come-si-gioca/)
* [https://www.Fantacalcio.it/consigli-Fantacalcio/14\_10\_2020/consigli-Fantacalcio-come-dividere-il-budget-all-asta-391828](https://www.fantacalcio.it/consigli-fantacalcio/14_10_2020/consigli-fantacalcio-come-dividere-il-budget-all-asta-391828)
* <https://it.wikipedia.org/wiki/Fantacalcio>
* <https://dotnet.microsoft.com/download/dotnet/3.1>