Requirements Analysis Document

Sommario

[1. Introduzione 4](#_Toc157244864)

[1.1 Scopo del sistema 4](#_Toc157244865)

[1.2 Ambito del sistema 4](#_Toc157244866)

[1.3 Obiettivi e criteri di successo del progetto 5](#_Toc157244867)

[2 Sistema proposto 5](#_Toc157244868)

[2.1 Panoramica 5](#_Toc157244869)

[2.1.1 Funzionalità Guest 5](#_Toc157244870)

[2.1.2 Funzionalità Giocatore 6](#_Toc157244871)

[2.1.3 Funzionalità Organizzatore 6](#_Toc157244872)

[2.1.4 Funzionalità Moderatore 6](#_Toc157244873)

[2.2 Requisiti funzionali 6](#_Toc157244874)

[2.2.a Gestione Guest: 6](#_Toc157244875)

[2.2.b Gestione utente: 6](#_Toc157244876)

[2.2.c Gestione giocatore: 7](#_Toc157244877)

[2.2.d Gestione torneo: 7](#_Toc157244878)

[2.2.e Gestione moderatore: 7](#_Toc157244879)

[2.3 Requisiti non funzionali 7](#_Toc157244880)

[2.3.1 RNF\_Usabilità: 7](#_Toc157244881)

[2.3.2 RNF\_Affidabilità: 7](#_Toc157244882)

[2.3.3 RNF\_Prestazioni: 7](#_Toc157244883)

[2.3.4 RNF\_Supportabilità: 7](#_Toc157244884)

[2.3.5 RNF\_Vincoli Implementazione: 7](#_Toc157244885)

[2.3.6 RNF\_Vincoli Interfaccia: 7](#_Toc157244886)

[2.3.7 RNF\_Vincoli Packaging: 8](#_Toc157244887)

[2.3.8 RNF\_Aspetti legali: 8](#_Toc157244888)

[2.4 Modelli di sistema 8](#_Toc157244889)

[2.4.1 Scenari 8](#_Toc157244890)

[2.4.1.1 Utente Guest che vuole registrarsi sia come organizzatore che come giocatore. 8](#_Toc157244891)

[2.4.1.2 Iscrizione di un giocatore al torneo 9](#_Toc157244892)

[2.4.1.3 Il giocatore inserisce la propria squadra 9](#_Toc157244893)

[2.4.1.4 Un organizzatore di tornei che vuole creare un torneo 10](#_Toc157244894)

[2.4.1.5 Gestione dei Tornei 10](#_Toc157244895)

[2.4.2 Use cases model 10](#_Toc157244896)

[2.4.2.1 Use case model guest 10](#_Toc157244897)

[2.4.2.2 Use case model giocatore 10](#_Toc157244898)

[2.4.2.3 Use case model organizzatore 10](#_Toc157244899)

[2.4.2.3 Use case model moderatore 10](#_Toc157244900)

[2.4.3 Use cases 11](#_Toc157244901)

[UC 1: Autenticazione 11](#_Toc157244902)

[UC 1.1: Credenziali sbagliate o non presenti 12](#_Toc157244903)

[UC 1.2: Account Bannato 13](#_Toc157244904)

[UC 1.3 Account ancora non accettato 13](#_Toc157244905)

[UC 2: Registrazione giocatore 14](#_Toc157244906)

[UC 2.1 Password non corrispondono 15](#_Toc157244907)

[UC 2.2 Email non valida 16](#_Toc157244908)

[UC 2.3 Username già esistente 16](#_Toc157244909)

[UC 3: Registrazione organizzatore 17](#_Toc157244910)

[UC 3.1 Password non corrispondono 18](#_Toc157244911)

[UC 3.2 Email non valida 18](#_Toc157244912)

[UC 3.3 Username già esistente 19](#_Toc157244913)

[UC 4: Accettare/Rifiutare un organizzatore 19](#_Toc157244914)

[UC 5: Cercare torneo 20](#_Toc157244915)

[UC 6: Logout 21](#_Toc157244916)

[UC 7: Bannare/Sbannare un giocatore/organizzatore 21](#_Toc157244917)

[UC 8: Crea torneo 22](#_Toc157244918)

[UC 8.1 Torneo già esistente 23](#_Toc157244919)

[UC 8.2 Cancella creazione Torneo 23](#_Toc157244920)

[UC 9: Iscrizione Torneo 24](#_Toc157244921)

[UC 9.1: Iscrizione piene 25](#_Toc157244922)

[UC 9.2: Iscrizione torneo rifiutato 25](#_Toc157244923)

[UC 10: Seguire un Organizzatore 26](#_Toc157244924)

[UC 11: Modificare profilo 26](#_Toc157244925)

[UC 11.1 Password non corrispondono 27](#_Toc157244926)

[UC 11.2 Email non valida 28](#_Toc157244927)

[UC 11.3 Username già esistente 28](#_Toc157244928)

[UC 12: Sostituire membro Team 29](#_Toc157244929)

[UC 13: Iniziare torneo 29](#_Toc157244930)

[UC 14: Terminare torneo 30](#_Toc157244931)

[UC 15: Togliere partecipanti 30](#_Toc157244932)

[UC 16: Visualizzare Profilo Giocatore 31](#_Toc157244933)

[UC 17: Aggiungere Risultato 31](#_Toc157244934)

[UC 18: Partecipa Torneo 32](#_Toc157244935)

[2.4.4 Object model 33](#_Toc157244936)

[2.4.4.1 Class diagram 33](#_Toc157244937)

[2.4.5 Dynamic model 33](#_Toc157244938)

[2.4.5.1 Sequence diagram 33](#_Toc157244939)

[2.4.5.2 Statechart diagram 33](#_Toc157244940)

[2.4.5.3 Activity diagram 33](#_Toc157244941)

[2.4.6 User interface 33](#_Toc157244942)

[2.4.6.1 Diagrammi navigazionali 33](#_Toc157244943)

[2.4.6.2 Screen mock-up 33](#_Toc157244944)

# Introduzione

## 1.1 Scopo del sistema

Il nostro sistema è progettato per fornire un supporto completo agli appassionati di Pokémon che desiderano organizzare e partecipare a tornei Pokémon. Si tratta di un sito web dedicato esclusivamente alla gestione degli aspetti organizzativi e partecipativi dei tornei, offrendo agli utenti uno strumento centralizzato per gestire giocatori e tornei in modo efficiente.

Per i Partecipanti:

Gli appassionati di Pokémon possono utilizzare il sistema per iscriversi ai tornei, costruire il proprio player online e connettersi con altri giocatori. L'interfaccia intuitiva consente agli utenti di visualizzare e iscriversi a tornei in corso, gestire il proprio profilo, e tenere traccia delle partite passate. Gli scontri tra i giocatori possono avvenire attraverso: l'utilizzo di software esterni online oppure utilizzando qualsiasi genere di console trovandosi nello stesso luogo, il nostro sistema facilita la gestione di iscrizioni, regole del torneo e risultati.

Per gli Organizzatori:

Il sistema fornisce agli organizzatori uno strumento potente per la creazione e la gestione di tornei. Gli strumenti di amministrazione consentono di definire regole del torneo, gestire iscrizioni, monitorare l'avanzamento delle competizioni e pubblicare risultati. Gli organizzatori possono concentrarsi sulla logistica e sulle dinamiche del torneo.

## 1.2 Ambito del sistema

Il nostro sistema è progettato per la gestione online completa di giocatori, organizzatori e moderatori di tornei Pokémon. Attraverso una piattaforma web dedicata, forniamo strumenti avanzati per la registrazione, l'organizzazione e la partecipazione ai tornei. L'aggiunta della figura del moderatore rafforza il nostro impegno a garantire un ambiente di gioco sicuro e regolamentato, contribuendo a creare una comunità Pokémon online dinamica e ben moderata.

## 1.3 Obiettivi e criteri di successo del progetto

Il progetto mira a risolvere le attuali lacune nella gestione dei tornei Pokémon, fornendo una piattaforma dedicata. Gli obiettivi principali sono:

Registrazione e Profilo Utente:

* Consentire agli utenti di registrarsi e creare un profilo personale.
* Possibilità di inserire informazioni dettagliate sul proprio team Pokémon e livello di esperienza.

Creazione di Tornei:

* Gli utenti abilitati alla creazione di tornei potranno:
* Creare nuovi tornei.
* Definire regole e formato del torneo.

Partecipazione a Tornei:

* Gli utenti saranno in grado di:
* Cercare tornei in base alle proprie preferenze e abilità.
* Partecipare ai tornei, con gestione di iscrizioni, generazione di bracket e notifiche sugli incontri imminenti.

Matchmaking Equilibrato:

* Implementazione di un sistema di matchmaking che considera il livello di abilità dei partecipanti.
* Creazione di match equilibrati per migliorare l'esperienza di gioco.

Tracciamento e Riconoscimenti:

* Monitorare e tenere traccia dei risultati dei tornei, vittorie e sconfitte.
* Assegnare riconoscimenti agli utenti più talentuosi in base alle loro performance.

# 2 Sistema proposto

## 2.1 Panoramica

Il sistema sarà accessibile tramite il web e vi saranno diversi utenti: Guest, Giocatore, Organizzatore e Moderatore.

### 2.1.1 Funzionalità Guest

* Può visualizzare l'homepage.
* Può cercare i tornei.

### 2.1.2 Funzionalità Giocatore

* Può registrarsi e fare il login.
* Può visualizzare le proprie statistiche e dati personali di profilo.
* Può modificare i dati personali di profilo.
* Può partecipare ai tornei.
* Può seguire gli organizzatori.
* Può visualizzare i tornei.

### 2.1.3 Funzionalità Organizzatore

* Può registrarsi è attendere l'accettazione dal moderatore.
* Può fare il login al proprio account.
* Può creare tornei e gestirli.
* Può visualizzare i profili dei giocatori iscritti al suo torneo.
* Può rimuovere i giocatori dai suoi tornei.
* Può inserire i risultati delle partite.

### 2.1.4 Funzionalità Moderatore

* Può visualizzare le notifiche.
* Può accettare o rifiutare i giocatori.
* Può visualizzare tutti i giocatori e organizzatori.
* Può bannare o sbannare un giocatore o un organizzatore.

## Requisiti funzionali

### 2.2.a Gestione Guest:

RF2\_Registrazione giocatore

RF2.1\_Password non corrispondono

RF2.2\_Email non valida

RF2.3\_Username già esistente

RF3\_Registrazione organizzatore

RF3.1\_Password non corrispondono

RF3.2\_Email non valida

RF3.3\_Username già esistente

### 2.2.b Gestione utente:

RF1\_Autenticazione

RF1.1\_Credenziali sbagliate o non presenti

RF1.2\_Account bannato

RF1.3\_Account ancora non accettato RF6\_Logout

RF11\_Modificare profilo

RF11.1\_Password non corrispondono

RF11.2\_Email non valida

RF11.3\_Username già esistente

### 2.2.c Gestione giocatore:

RF12\_Sostituire membro Team

### 2.2.d Gestione torneo:

RF5\_Cercare torneo

RF8\_Crea torneo

RF8.1\_Torneo già esistente

RF8.2\_Cancella creazione Torneo

RF9\_Iscrizione Torneo

RF9.1\_Iscrizione piene

RF9.2\_Iscrizione torneo rifiutato

RF10\_Seguire un Organizzatore

RF13\_Iniziare torneo

RF14\_Terminare torneo

RF15\_Togliere partecipanti

RF16\_Visualizzare Profilo Giocatore

RF17\_Aggiungere Risultato

RF18\_Partecipa torneo

### 2.2.e Gestione moderatore:

RF4\_Accettare/rifiutare un organizzatore

RF7\_Bannare/sbannare un giocatore/organizzatore

## 2.3 Requisiti non funzionali

### 2.3.1 RNF\_Usabilità:

Il sistema dovrà offrire un'interfaccia intuitiva e user-friendly per garantire un'esperienza di utilizzo agevole sia per i giocatori che per gli organizzatori, non necessitando di documentazione di supporto.

### 2.3.2 RNF\_Affidabilità:

La piattaforma mira a garantire una gestione affidabile, minimizzando possibili interruzioni e assicurando che i dati siano conservati in modo sicuro per preservarne l'integrità.

### 2.3.3 RNF\_Prestazioni:

Il sistema deve garantire prestazioni ottimali, gestendo simultaneamente un elevato numero di utenti connessi , garantendo tempi di risposta rapidi per un'esperienza ottimale.

### 2.3.4 RNF\_Supportabilità:

Deve essere previsto un solido sistema di supporto per affrontare tempestivamente eventuali problemi tecnici o richieste degli utenti, assicurando un'assistenza continua e tempestiva.

Inoltre sarà possibile estendere il sistema o sostituire piattaforma hardware.

### 2.3.5 RNF\_Vincoli Implementazione:

Il sistema deve essere compatibile con le specifiche della piattaforma hardware indicata e deve superare con successo i test definiti dal team di testing, rispettando le relative restrizioni.

### 2.3.6 RNF\_Vincoli Interfaccia:

Il sistema deve essere progettato per interagire senza problemi con i sistemi esistenti e il processo di esportazione/importazione dei dati nel sistema deve seguire procedure standard

stabilite

### 2.3.7 RNF\_Vincoli Packaging:

Non ci sono vincoli di packaging.

### 2.3.8 RNF\_Aspetti legali:

Il sistema deve aderire rigorosamente a tutte le normative legali e norme in vigore, garantendo la sicurezza dei dati e rispettando la privacy degli utenti, al fine di evitare eventuali questioni legali.

## 2.4 Modelli di sistema

### 2.4.1 Scenari

#### 2.4.1.1 Utente Guest che vuole registrarsi sia come organizzatore che come giocatore.

Pippo(Utente Guest) è un grande appassionato di Pokemon sin da bambino, e spesso si riunisce con dei suoi amici che condividono la stessa passione. Nell'ultimo incontro il suo amico Paolo gli ha parlato di un nuovo formidabile sito web, di nome GottaBattleEmAll, nel quale è possibile ricercare dei tornei Pokemon. Pippo incuriosito da questa informazione decide di cercare tale sito. Pippo naviga sulla homepage del sito e vede dei tornei ma non appena clicca su quest'ultimi gli si chiede la Registrazione/Login.

Pippo per usufruire dei servizi oﬀerti dal sito sceglie di registrarsi siccome non può loggare non possedendo un account. Quando clicca su register gli compare una schermata in cui può inserire i propri dati nei campi come in Figura1. Pippo vuole registrarsi sia come organizzatore, per aver la possibilità di creare tornei sia come giocatore, per avere la possibilità di partecipare ai tornei. Prima sceglie di registrarsi come giocatore, eﬀettua la registrazione con successo e eﬀettua il login, dopodiché decide di sloggare e registrarsi come organizzatore, in questo caso deve aspettare l'accettazione, viene quindi inviata una notiﬁca a Franco (Moderatore) il quale logga tramite la sua pagina apposita in cui ha inserito la sua email e password. Franco si trova nella sua dashboard in Figura 2 . Franco vede la notiﬁca di Pippo e visualizza il suo stato trovandolo "pending". Per poter confermare o riﬁutare la richiesta si reca nell'opposita sezione di sinistra andando sulla schermata delle notiﬁche , in questa pagina visualizza nuovamente la richiesta e decide di accettare Pippo(Figura 3). Franco avvisa Pippo che adesso puo loggare. Pippo logga e commette un azione andando contro il regolamento. Franco decide di bannarlo,l ogga nella sua apposita pagina, dalla dashboard si reca nella sezione "elenco utenti" (Figura 4) qui trova Pippo, visualizza il suo account e lo banna. Pippo si ritrova sloggato ritenta il login ma senza successo.

#### 2.4.1.2 Iscrizione di un giocatore al torneo

Pippo è un utente registrato vuole iscriversi al torneo entra sulla pagina di login tramite il pulsante in alto a destra , seleziona login e si trova su Figura 5 . Pippo sbaglia ad inserire i dati per il login riprova ed il login ha successo. Si presenta la pagina principale con il login eﬀettuato e una sezione in alto a destra con l'accesso al proﬁlo e il logout.

Pippo appena loggato come utente giocatore vuole cercare un torneo a cui partecipare, inizialmente sfoglia i tornei presenti nella homepage , ma siccome nessuno di questi attira la sua attenzione decide di cercarlo tramite la sezione apposita digitando "TorneoPaldea". Pippo visualizza il risultato della ricerca in un'altra schermata. Pippo seleziona il primo risultato ma si accorge che non è presente la funzionalità di iscrizione ma bensì il messaggio : "Torneo già iniziato", allora Pippo decide di cercare un altro torneo e digita "TorneoKanto". Seleziona un nuovo torneo dove questa volta è presente la funzionalità di iscrizione, prova a iscriversi ma compare il messaggio: "Iscrizioni piene". Pippo prova a cercare un altro torneo digitando "TorneoUnisa" e riesce a trovarlo ed a iscriversi. Una volta iscritto visualizzando la lista dei partecipanti nota l'aggiunta del suo nome. Bro (organizzatore) decide di rimuovere Pippo , Pippo riprova a iscriversi al torneo dal quale è stato rimosso ma gli compare il messaggio: "Sei stato rimosso dal torneo non puoi iscriverti". Il torneo inizia ma Pippo riesce solo a visualizzare l'andamento del torneo. Pippo decide di andare sulla pagina dell'organizzatore Bro è seguirlo per avere delle notiﬁche per quando un prossimo torneo verrà organizzato.

#### 2.4.1.3 Il giocatore inserisce la propria squadra

Pippo è loggato vuole inserire la sua squadra nel suo proﬁlo.

Da loggato entrare nel suo proﬁlo tramite la sezione in alto a destra, è visualizzare le informazioni del proprio account, le statistiche e la possibilità di modiﬁcare la propria squadra.

(mock up)

Decide prima di visualizzare le proprie informazioni del proﬁlo attraverso una schermata su cui sono visibili Nome, Cognome, Email e la possibilità di cambiare le proprie informazioni.

(mock up)

Successivamente vuole visualizzare le proprie statistiche è attraverso una schermata visualizza tutte le statistiche da giocatore(Partite vinte, perse ecc.).

(mock up)

In ﬁne decide di modiﬁcare la propria squadra e lo fa attraverso una schermata in cui visualizza i Pokemon che già di default formano il suo team . Vuole sostituire il pokemon "Ditto" con "Charizard", tramite un apposita schermata ricerca "Charizard" e lo seleziona.

Contento della sua squadra di 6 pokemon appena modiﬁcata, eﬀettua il logout siccome deve andare a dormire.

#### 2.4.1.4 Un organizzatore di tornei che vuole creare un torneo

Bro è registrato come organizzatore dei tornei e vuole creare un torneo, clicca su login e accede con l'email [bro@brosky.com](mailto:bro@brosky.com) e password \*\*\*\*\*\*\*\*. I dati sono errati gli compare un messaggio di errore con su scritto " email o password errati". Reinserisce i dati e riesce a eﬀettuare il login.

Il Bro loggato vede la pagina principale in cui si trova un pulsante organizza torneo, si apre una schermata in cui compila i campi per la creazione del torneo Figura 6.

#### 2.4.1.5 Gestione dei Tornei

Bro ha creato diversi tornei e vuole gestirli. Una volta eﬀettuato l'accesso e giunto alla schermata principale, Bro vede un elenco dei tornei da lui creati tra i quali: "TorneoSinnoh", "TorneoUnisa" e "TorneoUnima" come in Figura 7

Seleziona "TorneoSinnoh" che ancora non è iniziato, per vedere chi sono i partecipanti e valutare se rimuovere qualcuno di essi. Nota che il numero di iscrizioni è al completo è decide di farlo iniziare.

Successivamente seleziona il "TorneoUnisa" che è in corso e vuole inserire un risultato per il match che si è appena concluso tra Nakura e Luis. Visualizza il progresso globale del torneo, dopodiché seleziona il bracket dei giocatori è inserisce il risultato di "1-0" (mock up) per Nakura. Curioso della squadra pokemon del vincitore, seleziona il suo nome per visualizzare il suo proﬁlo utente con la relativa squadra pokemon.

Una volta inseriti i risultati verrà distribuito l'elo e i risultati non potranno più essere modiﬁcati.

In ultimo seleziona il torneo "TorneoUnima" e siccome tutti le partite sono state disputate , Bro decide di visualizzare per l'ultima volta i bracket e di concludere il torneo.

### 2.4.2 Use cases model

#### 2.4.2.1 Use case model guest

#### 2.4.2.2 Use case model giocatore

#### 2.4.2.3 Use case model organizzatore

#### 2.4.2.3 Use case model moderatore

### 2.4.3 Use cases

#### UC 1: Autenticazione

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 1: Autenticazione | |
| Attori Partecipanti: | Utente:  Giocatore, Organizzatore e Moderatore | |
| Entry: | L’utente si trova sulla home page dell'Utente specifico con la schermata di autenticazione | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L’utente inserisce username e password e selezione se loggarsi come organizzatore o giocatore, invece nel caso di moderatore si troverà nella sua apposita pagina di login 2. L’utente invia i dati al sistema | 1. Il sistema controlla le credenziali e il controllo ha esito positivo 2. Il sistema reindirizza l’utente alla sua home page |
| Exit Condition: | L’utente è autenticato e si trova sulla sua home page | |
| Eccezioni: | Alternativa 1.1  Se al punto 3 il sistema rileva credenziali non corrette/assenti, verrà eseguito il caso d’uso UC 1.1 Credenziali sbagliate o non presenti. | |

|  |  |
| --- | --- |
|  | Alternativa 1.2  Se al punto 3 il sistema rileva credenziali relative ad un account bannato, verrà eseguito il caso d’uso UC 1.2 Account Bannato  Alternativa 1.3  Se al punto 3 il sistema rileva credenziali relativa ad un account Organizzatore non ancora accettato, verrà eseguito il caso d'uso UC  1.3 Account ancora non accettato |

#### UC 1.1: Credenziali sbagliate o non presenti

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 1.1: Credenziali sbagliate o non presenti | |
| Attori Partecipanti: | Utente:  Giocatore, Organizzatore e Moderatore | |
| Entry: | L’utente ha provato ad autenticarsi e le credenziali non erano corrette/assenti | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “username o password non corrette/assenti” 2. il sistema ripresenta all’utente la schermata di autenticazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di autenticazione | |

#### UC 1.2: Account Bannato

|  |  |
| --- | --- |
| Titolo: | UC 1.2: Account Bannato |
| Attori Partecipanti: | Utente:  Giocatore, Organizzatore |

|  |  |  |
| --- | --- | --- |
| Entry: | L’utente ha provato ad autenticarsi e le credenziali erano relative ad un account bannato | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “account bannato” 2. il sistema ripresenta all’utente la schermata di autenticazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di autenticazione | |

#### UC 1.3 Account ancora non accettato

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 1.3 Account ancora non accettato | |
| Attori Partecipanti: | Utente: Organizzatore | |
| Entry: | L’utente ha provato ad autenticarsi e le credenziali erano relative ad un account organizzatore ancora non accettata dal moderatore | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “account ancora non accettato” 2. il sistema ripresenta all’utente la schermata di autenticazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di autenticazione | |

#### UC 2: Registrazione giocatore

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 2: Registrazione giocatore | |
| Attori Partecipanti: | Utente: Guest | |
| Entry: | L’utente si trova sulla home page con la schermata di registrazione | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L’utente inserisce username, nome, cognome, email, password e conferma password e seleziona la registrazione come giocatore. 2. L’utente invia i dati al sistema | 1. Il sistema controlla le credenziali e il controllo ha esito positivo 2. Il sistema reindirizza l’utente nella pagina di login |
| Exit Condition: | L’utente è registrato e si trova sulla pagina di login | |
| Eccezioni: | Alternativa 2.1  Se al punto 3 il sistema rileva che i due campi password non combaciano, verrà eseguito il caso d’uso UC 2.1 Password non corrispondono.  Alternativa 2.2  Se al punto 3 il sistema rileva che l'email non è valida, verrà eseguito il caso d’uso UC 2.2 Email non valida.  Alternativa 2.3  Se al punto 3 il sistema rileva un username già esistente, verrà eseguito il caso d'uso UC 2.3 Username già esistente. | |

#### UC 2.1 Password non corrispondono

|  |  |
| --- | --- |
| Titolo: | UC 2.1 Password non corrispondono |
| Attori Partecipanti: | Utente: Guest |

|  |  |  |
| --- | --- | --- |
|  |  | |
| Entry: | L’utente ha provato ad registrarsi e le password non corrispondono | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “password non corrispondono” 2. il sistema ripresenta all’utente la schermata di registrazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di registrazione. | |

#### UC 2.2 Email non valida

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 2.2 Email non valida | |
| Attori Partecipanti: | Utente: Guest | |
| Entry: | L’utente ha provato ad registrarsi e l'email non è valida | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “email non valida” 2. il sistema ripresenta all’utente la schermata di registrazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di registrazione. | |

#### UC 2.3 Username già esistente

|  |  |
| --- | --- |
| Titolo: | UC 2.3 Username già esistente |
| Attori Partecipanti: | Utente: Guest |

|  |  |  |
| --- | --- | --- |
| Entry: | L’utente ha provato ad registrarsi e l'username è già esistente | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “Username già esistente” 2. il sistema ripresenta all’utente la schermata di registrazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di registrazione. | |

#### UC 3: Registrazione organizzatore

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 3: Registrazione organizzatore | |
| Attori Partecipanti: | Utente: Guest | |
| Entry: | L’utente si trova sulla home page con la schermata di registrazione | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L’utente inserisce username, nome, cognome, email, password e conferma password e seleziona la registrazione come organizzatore. 2. L’utente invia i dati al sistema | 1. Il sistema controlla le credenziali e il controllo ha esito positivo 2. Il sistema invierà un messaggio di avvenuta richiesta di registrazione 3. Il sistema invierà una notifica al moderatore per permettere di accettare l'organizzatore 4. Il sistema reindirizza l'utente sulla pagina di login |
| Exit Condition: | L’utente si trova sulla pagina di login in attesa dell'avvenuta registrazione | |

|  |  |
| --- | --- |
| Eccezioni: | Alternativa 3.1  Se al punto 3 il sistema rileva che i due campi password non combaciano, verrà eseguito il caso d’uso UC 2.1 Password non corrispondono.  Alternativa 3.2  Se al punto 3 il sistema rileva che l'email non è valida, verrà eseguito il caso d’uso UC 2.2 Email non valida.  Alternativa 3.3  Se al punto 3 il sistema rileva un username già esistente, verrà eseguito il caso d'uso UC 2.3 Username già esistente. |

#### UC 3.1 Password non corrispondono

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 3.1 Password non corrispondono | |
| Attori Partecipanti: | Utente: Guest | |
| Entry: | L’utente ha provato ad registrarsi e le password non corrispondono | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “password non corrispondono” 2. il sistema ripresenta all’utente la schermata di registrazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di registrazione. | |

#### UC 3.2 Email non valida

|  |  |
| --- | --- |
| Titolo: | UC 3.2 Email non valida |
| Attori Partecipanti: | Utente: Guest |

|  |  |  |
| --- | --- | --- |
|  |  | |
| Entry: | L’utente ha provato ad registrarsi e l'email non è valida | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “email non valida” 2. il sistema ripresenta all’utente la schermata di registrazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di registrazione. | |

#### UC 3.3 Username già esistente

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 3.3 Username già esistente | |
| Attori Partecipanti: | Utente: Guest | |
| Entry: | L’utente ha provato ad registrarsi e l'username è già esistente | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “Username già esistente” 2. il sistema ripresenta all’utente la schermata di registrazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di registrazione. | |

#### UC 4: Accettare/Rifiutare un organizzatore

|  |  |
| --- | --- |
| Titolo: | UC 4: Accettare/Rifiutare un organizzatore |
| Attori Partecipanti: | Moderatore |
| Entry: | Il moderatore si trova sulla home page |

|  |  |  |
| --- | --- | --- |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. Il moderatore clicca "notifiche" dove visualizza l'elenco delle richieste degli organizzatori 2. Il moderatore cerca tra le varie notifiche l'organizzatore che vuole accettare/rifiutare e clicca sulla specifica scelta 3. Il moderatore invia i dati al sistema | 1. Il sistema accetterà/rifiuterà in base alla decisione del moderatore 2. Il sistema reindirizza il moderatore nella sezione "notifiche" |
| Exit Condition: | Il moderatore non visualizza piu l'organizzatore accettato/rifiutato | |
| Eccezioni: |  | |

#### UC 5: Cercare torneo

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 5: Cercare torneo | |
| Attori Partecipanti: | Utente: Giocatore | |
| Entry: | Il giocatore si trova sulla home page ed è gia autenticato | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. Il giocatore inserisce il nome del torneo che vuole ricercare nell'apposita sezione. 2. Il giocatore invia i dati al sistema | 1. Il sistema ricerca per nome torneo 2. Il sistema reindirizza nella pagina con il torneo con il nome inserito, se esistente |

|  |  |
| --- | --- |
| Exit Condition: | Il giocatore si trova nella pagina dove visualizza il torneo che ha ricercato, se esistente |
| Eccezioni: |  |

#### UC 6: Logout

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 6: Logout | |
| Attori Partecipanti: | Utente:  Giocatore, Organizzatore e Moderatore | |
| Entry: | L’utente si trova su qualsiasi pagina | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L’utente selezione la funzionalità di logout | 2. Il sistema reindirizza l’utente alla sua home page |
| Exit Condition: | L’utente non è autenticato e si trova sulla home page | |

#### UC 7: Bannare/Sbannare un giocatore/organizzatore

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 7: Bannare/Sbannare un giocatore/organizzatore | |
| Attori Partecipanti: | Moderatore | |
| Entry: | Il moderatore si trova nella sezione "elenco utenti" dove visualizza tutti i giocatori/organizzatori | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. Il moderatore cerca un giocatore/organizzatore che vuole bannare/sbannare e clicca sulla specifica scelta 2. Il moderatore invia i dati al sistema | 3. Il sistema bannerà/sbannerà in base alla decisione del moderatore |

|  |  |  |
| --- | --- | --- |
|  |  | 4. Il sistema reindirizza il moderatore nella sezione "elenco utenti" |
| Exit Condition: | Il moderatore visualizza lo stato del giocatore/organizzatore cambiato | |
| Eccezioni: |  | |

#### UC 8: Crea torneo

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 8: Crea torneo | |
| Attori Partecipanti: | Organizzatore | |
| Entry: | L'organizzatore è loggato e si trova sulla sua home page | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L'organizzatore seleziona la funzionalità di creazione del torneo.   1. L'Organizzatore inserisce:    * Nome Torneo    * Data    * Ora    * Regole    * Premi    * Organizzazione    * Capienza 2. L'organizzatore invia i dati al sistema. | 2. Il sistema reindirizza l'organizzatore nell'apposita pagina di creazione del torneo.  5. Il sistema controlla la correttezza dei dati ed ha esito positivo.  6. Il sistema reindirizza l'organizzatore sulla pagina del |

|  |  |  |
| --- | --- | --- |
|  |  | torneo con un messaggio di avvenuta creazione. |
| Exit Condition: | L'organizzatore si trova sulla pagina del torneo. | |
| Eccezioni: | Alternativa 8.1:  Se al punto 5 il sistema rileva un torneo gia esistente con quel nome, verrà eseguito il caso d'uso UC 7.1 Torneo già esistente.  Alternativa 8.2:  Se al punto 3 l'organizzatore clicca sull'opzione di cancellazione del torneo, verrà eseguito il caso d'uso UC 7.2 Cancella creazione Torneo | |

#### UC 8.1 Torneo già esistente

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 8.1 Torneo già esistente | |
| Attori Partecipanti: | Organizzatore | |
| Entry: | L'organizzatore ha provato a creare il torneo, ma ha utilizzato un nome torneo gia esistente. | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. Il sistema mostra il messaggio di errore “nome torneo già esistente” 2. Il sistema ripresenta all’organizzatore la stessa pagina di creazione torno con i campi già compilati tranne per il nome |
| Exit Condition: | L'organizzatore si trova nella pagina di creazione del torneo. | |

#### UC 8.2 Cancella creazione Torneo

|  |  |
| --- | --- |
| Titolo: | UC 8.2 Cancella creazione Torneo |
| Attori Partecipanti: | Organizzatore |

|  |  |  |
| --- | --- | --- |
| Entry: | L'organizzatore ha deciso di non continuare con la creazione del torneo | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. Il sistema ripresenta all’organizzatore la schermata di home page |
| Exit Condition: | L’organizzatore si trova sulla home page. | |

#### UC 9: Iscrizione Torneo

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 9: Iscrizione Torneo | |
| Attori Partecipanti: | Giocatore | |
| Entry: | Il giocatore si trova sulla pagina di un torneo (Non ancora iniziato) | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. Il giocatore seleziona l'apposita funzionalità di iscrizione. | 1. Il sistema controlla se si puo iscrivere. 2. Il sistema iscrive il giocatore al torneo. 3. Il sistema reindirizza il giocatore nella pagina del torneo. |
| Exit Condition: | Il giocatore è iscritto al torneo | |
| Eccezioni: | Alternativa 9.1:  Se al punto 2 il sistema rileva che il torneo è pieno, verrà eseguito il caso d'uso UC 9.1 Iscrizione piene.  Alternativa 9.2:  Se al punto 2 , il sistema rileva che il giocatore è stato rimosso precedentemente, verrà eseguito il caso d'uso 9.2 Iscrizione torneo rifiutato. | |

#### UC 9.1: Iscrizione piene

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 9.1: Iscrizione piene | |
| Attori Partecipanti: | Giocatore | |
| Entry: | Il giocatore ha provato ad iscriversi, ma il torneo gia era pieno. | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. Il sistema mostra il messaggio di errore "Iscrizioni piene". |
| Exit Condition: | Il giocatore si trova sulla pagina del torneo, al quale non può iscriversi | |
| Eccezioni: |  | |

#### UC 9.2: Iscrizione torneo rifiutato

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 9.2: Iscrizione torneo rifiutato | |
| Attori Partecipanti: | Giocatore | |
| Entry: | Il giocatore ha provato ad iscriversi, ma era stato precedentemente rimosso dal torneo | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. Il sistema mostra il messaggio di errore "Sei stato rimosso dal torneo non puoi iscriverti". |
| Exit Condition: | Il giocatore si trova sulla pagina del torneo, al quale non può iscriversi | |
| Eccezioni: |  | |

#### UC 10: Seguire un Organizzatore

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 10: Seguire un Organizzatore | |
| Attori Partecipanti: | Giocatore | |
| Entry: | Sulla pagina di un Torneo | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. Il giocatore seleziona il nome dell'organizzatore  3. Il giocatore seleziona la funzionalità segui | 2. Il sistema reindirizza sulla pagina dell'organizzatore |
| Exit Condition: | Il giocatore si trova sulla pagina dell'organizzatore e lo segue | |
| Eccezioni: |  | |

#### UC 11: Modificare profilo

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 11: Modificare profilo | |
| Attori Partecipanti: | Utente:  Giocatore e Organizzatore | |
| Entry: | L'utente si trova nella pagina di impostazioni del profilo | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L'utente modifica i dati che gli interessato tra:    * Username    * Nome    * Cognome    * Email    * Password    * Conferma Password 2. L’utente invia i dati al sistema | 3. Il sistema controlla le modifiche e il controllo ha esito positivo. |

|  |  |  |
| --- | --- | --- |
|  |  | 4. Il sistema reindirizza l’utente nella sua pagina di profilo |
| Exit Condition: | L'utente si trova sulla sua pagina di profilo | |
| Eccezioni: | Alternativa 11.1  Se al punto 3 il sistema rileva che i due campi password non corrispondono, verrà eseguito il caso d’uso UC 10.1 Password non combaciano.  Alternativa 11.2  Se al punto 3 il sistema rileva che l'email non è valida, verrà eseguito il caso d’uso UC 10.2 Email non valida.  Alternativa 11.3  Se al punto 3 il sistema rileva un username già esistente, verrà eseguito il caso d'uso UC 10.3 Username già esistente. | |

#### UC 11.1 Password non corrispondono

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 11.1 Password non corrispondono | |
| Attori Partecipanti: | Utente:  Giocatore e Organizzatore | |
| Entry: | L’utente ha provato a cambiare password ma le password non corrispondono | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “password non corrispondono” 2. il sistema ripresenta all’utente la schermata di registrazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di registrazione. | |

#### UC 11.2 Email non valida

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 11.2 Email non valida | |
| Attori Partecipanti: | Utente:  Giocatore e Organizzatore | |
| Entry: | L’utente ha provato a cambiare l'email ma non è valida | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “email non valida” 2. il sistema ripresenta all’utente la schermata di registrazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di registrazione. | |

#### UC 11.3 Username già esistente

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 11.3 Username già esistente | |
| Attori Partecipanti: | Utente:  Giocatore e Organizzatore | |
| Entry: | L’utente ha provato a cambiare l'username ma già è esistente | |
| Flow of events: | Azioni utente | Azioni sistema |
|  |  | 1. il sistema mostra il messaggio di errore “Username già esistente” 2. il sistema ripresenta all’utente la schermata di registrazione |
| Exit Condition: | L’utente si trova sulla home page specifica dell'Utente con la schermata di registrazione. | |

#### UC 12: Sostituire membro Team

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 12: Sostituire membro Team | |
| Attori Partecipanti: | Giocatore | |
| Entry: | Giocatore si trova sulla pagina del suo profilo | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. Il giocatore seleziona la funzionalità di sostituzione di un membro della squadra |  |
| 2. Il giocatore seleziona il pokemon che vuole sostituire | 3. Il sistema propone al giocatore i pokemon disponibili |
| 4. Il giocatore selezione il nuovo pokemon | 4. il sistema sostituisce il pokemon alla squadra  5. Il sistema reindirizza il giocatore sulla pagina di profilo con la squadra modificata |
| Exit Condition: | Il giocatore si trova nella pagina di profilo con la squadra modificata | |
| Eccezioni: |  | |

#### UC 13: Iniziare torneo

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 13: Iniziare torneo | |
| Attori Partecipanti: | Organizzatore | |
| Entry: | L'organizzatore si trova sulla sua home page | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L'organizzatore seleziona il torneo | 2. Il sistema reindirizza sulla pagina del torneo |

|  |  |  |
| --- | --- | --- |
|  | che ha intenzione di far iniziare  3. L'organizzatore seleziona la funzionalità di inizio | 5. Il sistema inizializza il torneo  6. Il sistema reindirizza sulla pagina del torneo iniziato |
| Exit Condition: | L'organizzatore si trova sulla pagina del torneo iniziato | |
| Eccezioni: |  | |

#### UC 14: Terminare torneo

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 14: Terminare torneo | |
| Attori Partecipanti: | Organizzatore | |
| Entry: | L'organizzatore si trova sulla sua home page | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L'organizzatore seleziona il torneo in corso  3. L'organizzatore seleziona la funzionalita di terminazione | 2. Il sistema reindirizza sulla pagina del torneo  4. Il sistema termina il torneo e reindirizza sulla pagina del torneo finito |
| Exit Condition: | l'organizzatore si trova sulla pagina del torneo | |
| Eccezioni: |  | |

#### UC 15: Togliere partecipanti

|  |  |
| --- | --- |
| Titolo: | UC 15: Togliere partecipanti |
| Attori Partecipanti: | Organizzatore |

|  |  |  |
| --- | --- | --- |
| Entry: | L'organizzatore si trova sulla pagina del torneo non iniziato | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L'organizzatore seleziona il giocatore da eliminare | 1. Il sistema rimuove il giocatore selezionato dal torneo 2. Il sistema reindirizza sulla pagina del torneo |
| Exit Condition: | L'organizzatore si trova sulla pagina del torneo senza il giocatore eliminato | |
| Eccezioni: |  | |

#### UC 16: Visualizzare Profilo Giocatore

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 16: Visualizzare Profilo Giocatore | |
| Attori Partecipanti: | Organizzatore | |
| Entry: | L'organizzatore si trova sulla pagina del torneo iniziato | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L'organizzatore seleziona il profilo del giocatore che vuole visualizzare | 2. Il sistema reindirizza sulla pagina del profilo del giocatore |
| Exit Condition: | L'organizzatore si trova sulla pagina del profilo del giocatore | |
| Eccezioni: |  | |

#### UC 17: Aggiungere Risultato

|  |  |
| --- | --- |
| Titolo: | UC 17: Aggiungere Risultato |
| Attori Partecipanti: | Organizzatore |

|  |  |  |
| --- | --- | --- |
| Entry: | L'organizzatore si trova sulla pagina del torneo | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. L'organizzatore seleziona la partita da convalidare 2. L'organizzatore selezione il giocatore che ha vinto 3. L'organizzatore invia i dati al sistema | 1. Il sistema salva il risultato 2. il sistema reindirizza alla pagina di partita con risultato selezionato |
| Exit Condition: | L'organizzatore si trova sulla pagina della partita con risultato selezionato | |
| Eccezioni: |  | |

#### UC 18: Partecipa Torneo

|  |  |  |
| --- | --- | --- |
| Titolo: | UC 18: Partecipa Torneo | |
| Attori Partecipanti: | Utente: Giocatore | |
| Entry: | Il giocatore si trova sulla home page ed è gia autenticato | |
| Flow of events: | Azioni utente | Azioni sistema |
|  | 1. Il giocatore seleziona la funzionalità di "Tornei".  3. Il giocatore seleziona il torneo che vuole visualizzare. | 2. Il sistema ricerca tutti i tornei a cui è già iscritto ed è iniziato.  4. Il sistema reindirizza nella pagina del torneo selezionato. |
| Exit Condition: | Il giocatore si trova nella pagina dove visualizza il torneo che ha selezionato. | |
| Eccezioni: |  | |

### 2.4.4 Object model

#### 2.4.4.1 Class diagram

### 2.4.5 Dynamic model

#### 2.4.5.1 Sequence diagram

#### 2.4.5.2 Statechart diagram

#### 2.4.5.3 Activity diagram

### 2.4.6 User interface

#### 2.4.6.1 Diagrammi navigazionali

#### 2.4.6.2 Screen mock-up