



Giovanni - Celia - 215522

Marco - De Cicco - 184841

Tommaso - Froio - 180101

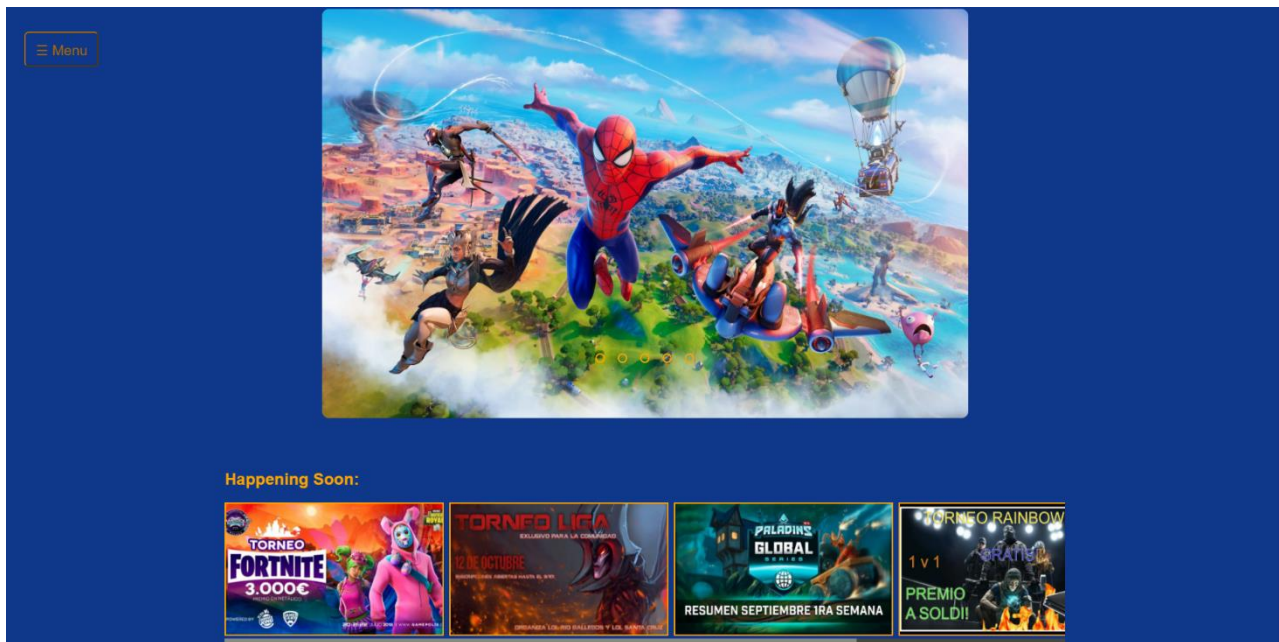
Samuele - Chiarella - 186544

# Indice

<b>Analisi Dei Requisiti Utente</b> .....	<b>2</b>
<b>Analisi requisiti di Sistema</b> .....	<b>4</b>
<b>Casi d'uso</b> .....	<b>5</b>
CDU1 – Gestire Torneo.....	5
Diagramma di sequenza di sistema .....	7
CDU2 – Iscrivere ad un torneo .....	8
Diagramma di sequenza di sistema .....	10
CDU3 – Avviare e monitorare l'andamento del torneo .....	11
Diagramma di sequenza di sistema .....	13
CDU4 – Gestire Team .....	14
Diagramma di sequenza di sistema .....	16
CDU5 – Gestire profilo organizzatore .....	17
Diagramma di sequenza di sistema .....	19
<b>Modello di Dominio</b> .....	<b>20</b>
<b>Contratti</b> .....	<b>21</b>
<b>Architettura logica del sistema</b> .....	<b>26</b>

# Analisi dei Requisiti Utente:

L'applicazione web permette di gestire, creare e partecipare a tornei di videogiochi online.

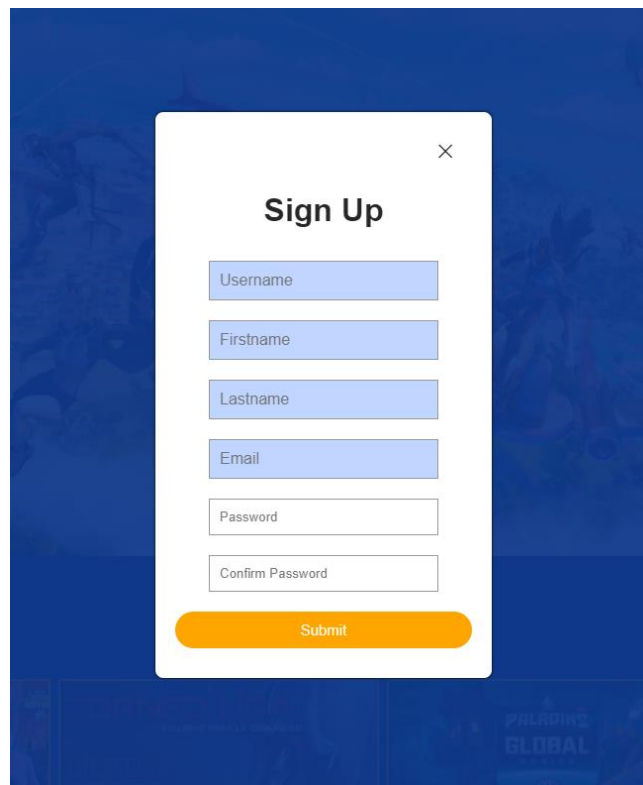


Sarà inoltre possibile assistere al torneo in live streaming o controllare l'andamento del torneo tramite apposite funzionalità e richiedere supporto allo staff tramite un'apposita chat.

## Funzionalità comuni:

- Esiste già un utente amministratore nel database;

- Chiunque può registrarsi (tranne i francesi KEKW);

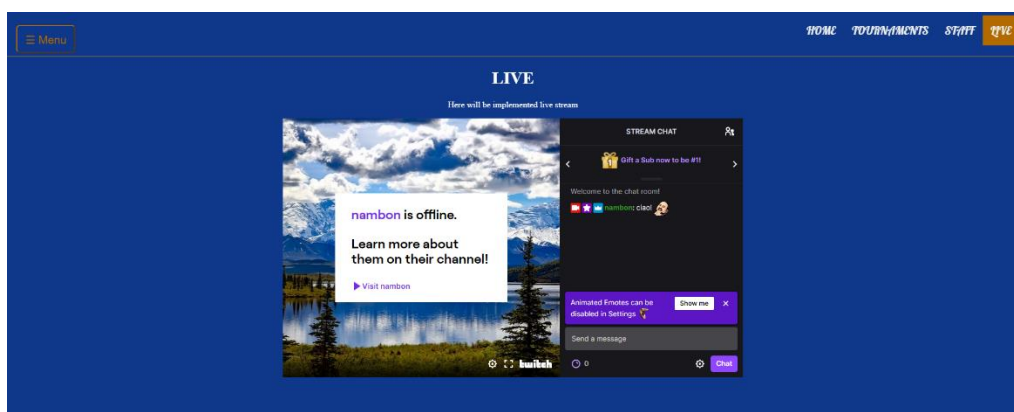
A white rectangular sign-up form with a close button (X) in the top right corner. The form is titled "Sign Up" and contains input fields for Username, Firstname, Lastname, Email, Password, and Confirm Password. A yellow "Submit" button is at the bottom. The form is centered on a dark blue background with a faint, repeating pattern of the word "PILGRIMS" and "GLOBAL".

- In ogni pagina torneo saranno presenti:

- Descrizione torneo;
- Bracket torneo;
- Team partecipanti.

- In ogni pagina torneo potrà essere presente:

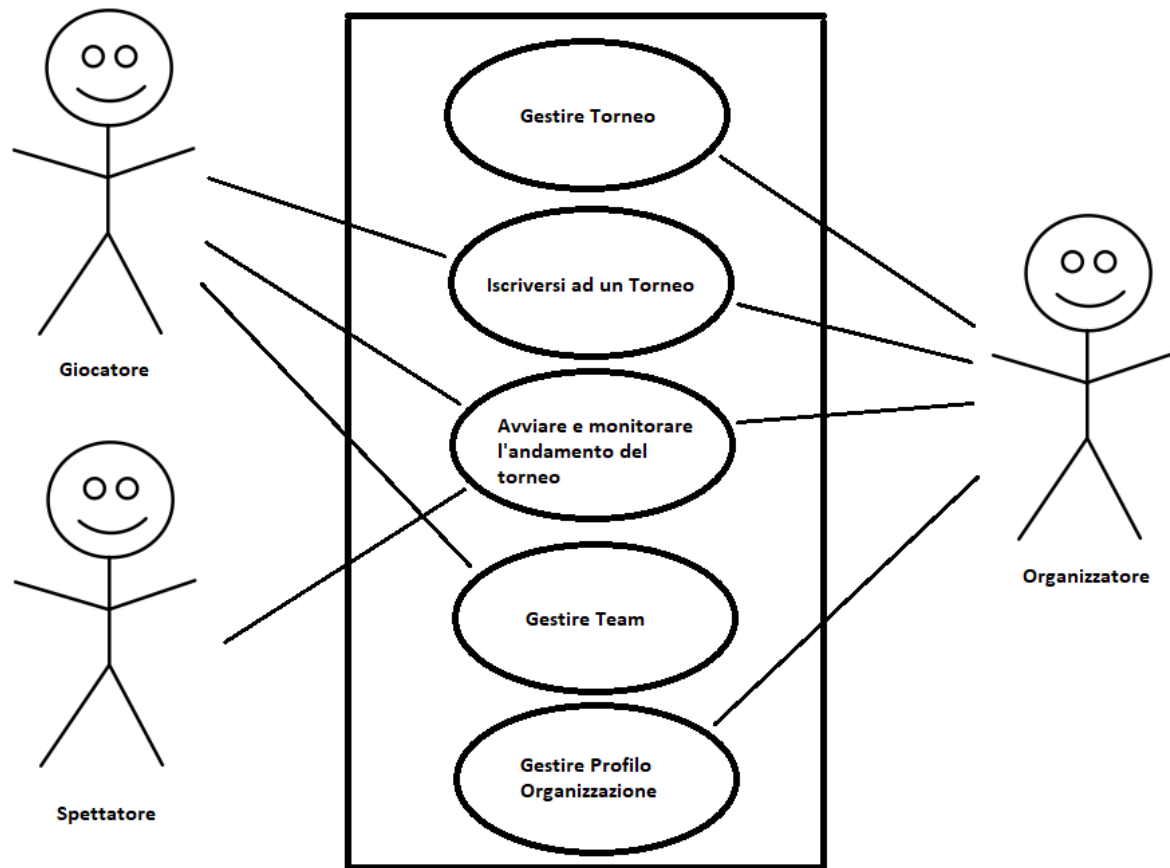
- Live stream su Twitch.tv qualora fosse disponibile;



- Banner di un eventuale sponsor.

# Analisi dei requisiti di sistema

Diagramma dei casi d'uso:



Il Sistema offre al Giocatore la possibilità di:

- Iscriversi ad un Torneo;
- Avviare e monitorare l'andamento del torneo;
- Gestire Team.

Il Sistema offre all'Organizzatore la possibilità di:

- Gestire Torneo;
- Iscriversi ad un Torneo;
- Avviare e monitorare l'andamento del torneo;
- Gestire Profilo Organizzazione.

Il Sistema offre allo Spettatore la possibilità di:

- Avviare e monitorare l'andamento del torneo.

# Casi d'uso

**Nome caso d'uso:** Gestire Torneo.

**Portata:** Gestore torneo di videogiochi.

**Livello:** Obiettivo utente.

**Attore primario:** Organizzatore.

**Parti interessate ed interessi:**

- Organizzatore: Avere la possibilità di creare un torneo con determinate specifiche, che sia visibile e facilmente accessibile ai giocatori interessati a quella categoria di tornei.
- Team: Avere la possibilità di cercare ed eventualmente partecipare al torneo.
- Spettatore: Avere la possibilità di assistere ad una eventuale live stream oppure controllare l'andamento del torneo.
- Sponsor: Avere la possibilità di pubblicizzare il proprio marchio sulla pagina del torneo.

**Pre-Condizioni:**

L'Organizzatore è identificato ed autenticato. L'organizzazione è stata scelta.

**Garanzia di successo:**

Il torneo viene creato secondo le specifiche e sarà visibile e consultabile a tutti gli utenti registrati e non.

L'eventuale marchio dello sponsor è visibile sulla pagina del torneo.

**Scenario principale di successo:**

- 1) L'Organizzatore vuole creare un torneo.
- 2) Il Sistema avvia la creazione del torneo.
- 3) L'Organizzatore inserisce le specifiche del torneo.

4) Il Sistema controlla il corretto inserimento e crea il torneo.

#### Estensioni:

1. L' Organizzatore decide di modificare un torneo:

1. Il Sistema avvia la modifica del torneo.
2. L'Organizzatore inserisce le specifiche del torneo (alcuni campi non sono modificabili).
3. Il Sistema controlla il corretto inserimento e modifica il torneo.

3. L' Organizzatore decide di ricercare una categoria particolare di giochi:

1. L' Organizzatore inserisce filtro desiderato.
2. Il Sistema restituisce i risultati della ricerca.
  - a. La ricerca non ha prodotto risultati:
    1. Il Sistema notifica l'assenza di risultati in base ai criteri di ricerca e si torna al punto 2).
3. Si prosegue con il punto 4).

4. Il Sistema trova degli errori nei campi:

1. Il Sistema notifica gli errori all'Organizzatore.
2. Ritorna al punto 3) dello scenario principale.

#### Requisiti speciali:

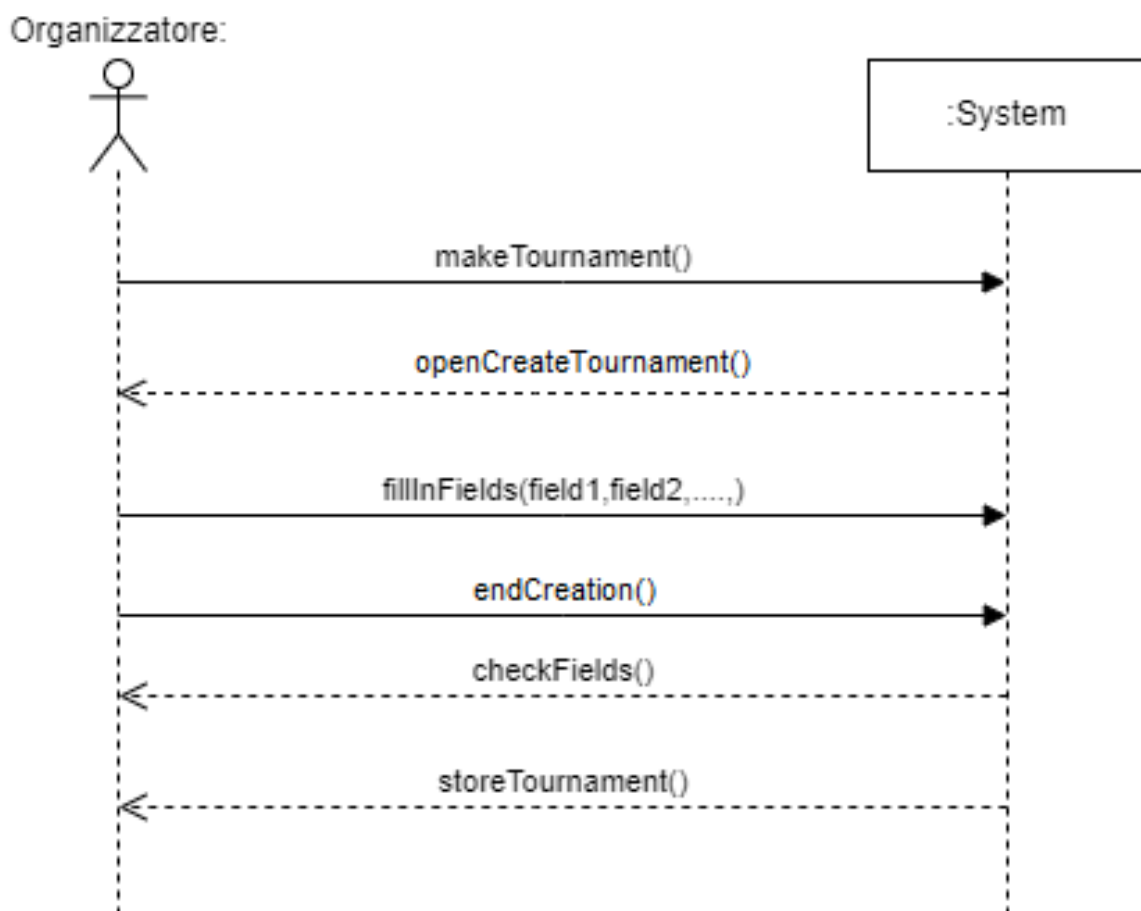
- Latenza nella risposta delle pagine web non superiore a 5 secondi.

**Frequenza di ripetizione:** Illimitate volte.

### Problemi aperti:

- Sarebbe meglio fare in modo che tutti i campi (compresa la data del torneo) siano modificabili?
- Ci può essere più di uno sponsor per torneo?
- Possibilità differenti di sponsorizzare un marchio oltre ai banner?

### Diagramma della Sequenza di Sistema





**Nome del caso d'uso:** Iscrivere ad un Torneo.

**Portata:** Gestore torneo di videogiochi.

**Attore primario:** Team.

**Parti interessate e interessi:**

- **Giocatore:** Avere la possibilità di cercare e selezionare e iscriversi ad un torneo.

**Pre-Condizioni:**

Il Giocatore è identificato ed autenticato. Il Giocatore è il creatore di un team.

**Garanzia di successo:**

L'iscrizione va a buon fine ed il Team risulta correttamente iscritto al torneo.

**Scenario principale di successo:**

- 1) Il Giocatore vuole iscriversi ad un torneo.
- 2) Il Sistema visualizza i tornei disponibili.
- 3) Il Giocatore sceglie e seleziona il torneo.
- 5) Il Sistema visualizza il torneo.
- 6) Il Giocatore vuole iscrivere il proprio team al torneo.
- 7) Il Sistema avvia l'iscrizione al torneo.
- 8) Il Giocatore sceglie, conferma il team con cui iscriversi e prosegue con la registrazione al torneo.
- 9) Il Sistema iscrive il team al torneo e aggiorna i partecipanti.

### Estensioni:

3a. Il Giocatore decide di modificare i criteri di ricerca:

1. Il Giocatore inserisce il filtro desiderato.
2. Il Sistema restituisce i risultati della ricerca.

2a. La ricerca non ha prodotto risultati:

1. Il Sistema notifica l'assenza di risultati in base ai criteri di ricerca e si torna al punto 2).
3. Si prosegue con il punto 4).

8a. Il Giocatore non fa parte di nessun team o vuole partecipare con un team non ancora creato:

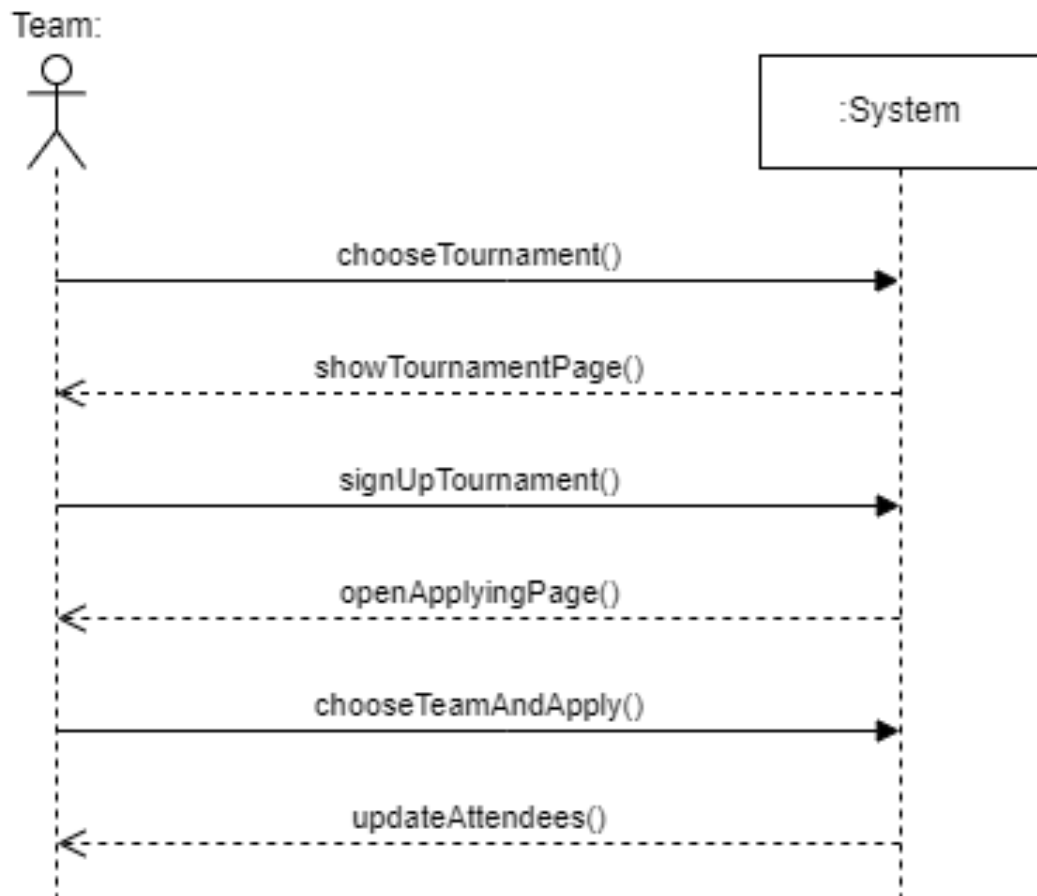
1. Il Giocatore avvia la creazione di un team.
2. Il Giocatore iscrive al torneo il team creato.

### Requisiti speciali:

- Evitare conflitti dovuti alla concorrenza nella registrazione ai tornei.
- Latenza nella risposta delle pagine web non superiore a 5 secondi.

**Frequenza di ripetizione:** Illimitate volte.

## Diagramma della Sequenza di Sistema



**Nome caso d'uso:** Avviare e monitorare l'andamento del torneo.

**Portata:** Gestore torneo di videogiochi.

**Livello:** Obiettivo utente.

**Attore primario:** Team.

**Parti interessate ed interessi:**

- Organizzazione: Avere la possibilità di monitorare il torneo.
- Team: Avere la possibilità giocare il proprio match e controllare l'andamento del torneo.
- Spettatore: Avere la possibilità di assistere ad una eventuale live stream oppure controllare l'andamento del torneo.

**Pre-Condizioni:**

- La Data d'inizio deve essere tale.
- Il numero minimo dei team deve essere soddisfatto.

**Garanzia di successo:**

- Il torneo viene avviato. i team si scontrano. Il bracket viene aggiornato. Si arriva ad un vincitore.

**Scenario Principale di Successo:**

- 1) Il Sistema avvia il torneo.
- 2) Il Sistema sorteggia.
- 3) Il Sistema controlla il check-in e aggiorna il bracket.
- 4) Il Sistema rimane in attesa del risultato delle partite.
- 5) Il Team vincente comunica il risultato.
- 6) Il Team perdente conferma il risultato.

- 7) Il Sistema controlla il risultato e aggiorna il bracket.
- 8) Il Sistema proclama il vincitore.

### Estensioni:

- a. In qualsiasi momento il Team può aprire un protest:
  1. Il Team leader contatta l'Organizzazione spiegando la motivazione del protest.
  2. L'Organizzazione prende e comunica la decisione al Team.
  3. Il Sistema chiude il protest.
3. Il Team non ha effettuato il check-in.
  1. Il Team avversario passa al turno successivo.
  2. Il Sistema aggiorna il bracket.
6. Ci sono difformità nel risultato.
  1. Il Team apre un protest e invia uno screenshot delle partite.
  2. L' Organizzazione controlla e aggiorna il risultato.
  3. Il Sistema chiude il protest.
8. Il torneo non è concluso.
  1. Ritorna al punto 3.

### Requisiti Speciali:

- Interfaccia utente con bracket e live streaming se prevista.
- Latenza nella risposta delle pagine web non superiore a 5 secondi.
- Il Team deve aver fatto il Check-in se previsto dall'Organizzatore

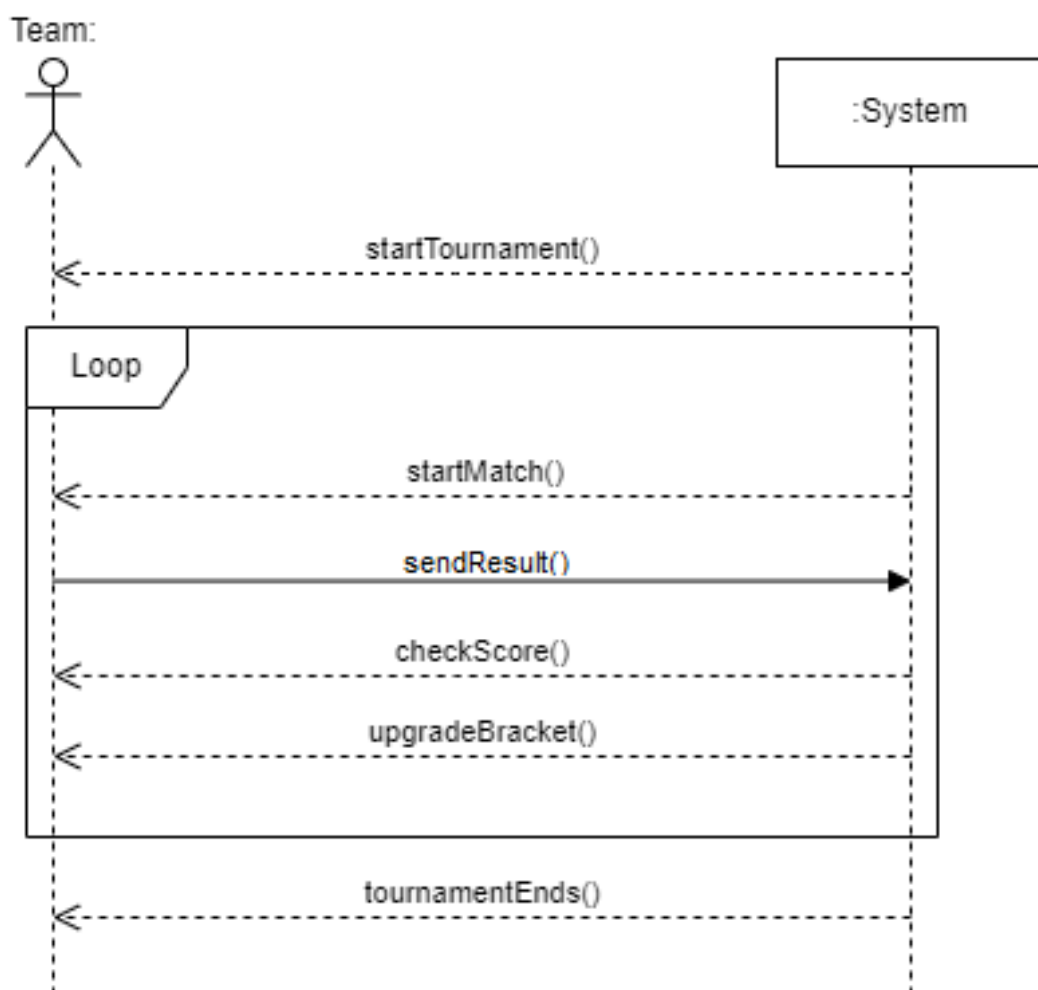
entro 10 minuti prima di ogni partita.

**Frequenza di ripetizione:** Illimitata.

### Problemi Aperti:

- L'Organizzatore può cambiare modalità di torneo in corso?
- Che bisogna fare quando un Team gioca con un player non presente nel Team?

### Diagramma della Sequenza di Sistema



**Nome caso d'uso:** Gestire Team.

**Portata:** Gestore torneo di videogiochi.

**Livello:** Obiettivo utente.

**Attore primario:** Giocatore.

**Parti interessate ed interessi:**

- Giocatore: Ha la possibilità di creare e gestire un team.

**Pre-Condizioni:**

Il Giocatore deve avere effettuato il login sulla piattaforma.

**Garanzia di successo:**

Il Giocatore riesce a creare un team e modificarne le sue specifiche.

**Scenario principale di successo:**

- 1) Il Giocatore vuole creare un team.
- 2) Il Sistema avvia la creazione del team.
- 3) Il Giocatore sceglie le specifiche di base.
- 4) Il Sistema controlla le specifiche, crea il team e avvia la modifica del team.
- 5) Il Giocatore inserisce le specifiche del team.
- 6) Il sistema convalida e aggiorna il team.

**Estensioni:**

4. Il Sistema trova degli errori nei campi:

1. Il Sistema notifica gli errori al Giocatore.

2. Ritorna al punto 3) dello scenario principale.
5. Il Giocatore vuole aggiungere dei membri al team:
  1. Il Giocatore inserisce un utente.
  2. Il Sistema convalida l'utente ed aggiorna i membri.

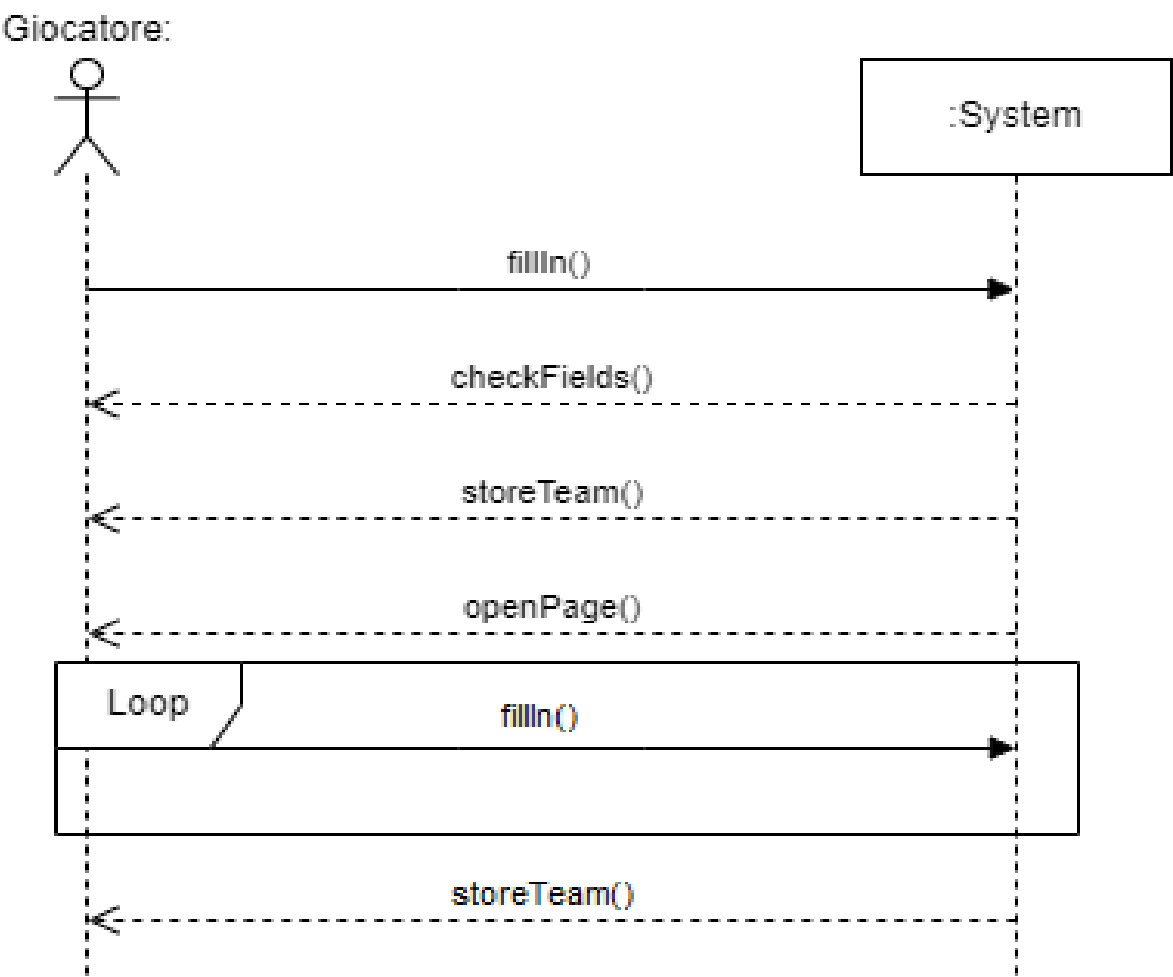
#### Requisiti Speciali:

- Latenza nella risposta delle pagine web non superiore a 5 secondi.
- Le finestre di segnalazione degli eventuali errori sono facilmente individuabili dall'utente.

**Frequenza di ripetizione:** illimitato.



# Diagramma della Sequenza di Sistema



**Nome del caso d'uso:** Gestire Profilo Organizzazione.

**Portata:** Gestore Tornei di videogiochi.

**Livello:** Obiettivo utente.

**Attore Primario:** Organizzatore.

**Parti Interessati ed Interessi:**

- Organizzatore: Vuole poter creare la propria organizzazione, modificare i dati di quest'ultima ed essere in grado di aggiungere altri eventuali membri dello staff.

**Pre-condizioni:** Tutti gli utenti interessati sono autenticati come organizzatori.

**Garanzia di Successo:**

L'Organizzazione viene creata. L'Organizzatore è in grado di modificare le info di quest'ultima a suo piacimento.

L'Organizzatore ha la possibilità di aggiungere i membri dello staff.

**Scenario principale di successo:**

1. L' Organizzatore vuole creare un'organizzazione.
2. Il Sistema avvia la creazione dell'organizzazione.
3. L' Organizzatore inserisce gli elementi di base dell'organizzazione.
4. Il sistema convalida, crea e avvia la modifica dell'organizzazione.
5. L' Organizzatore inserisce le specifiche dell'organizzazione.
6. Il sistema convalida e aggiorna l'organizzazione.

### Estensioni:

4. Il Sistema trova degli errori nei campi:

1. Il Sistema notifica gli errori all'Organizzatore.
2. Ritorna al punto 3) dello scenario principale.

5. L' Organizzatore vuole aggiungere dei membri allo staff dell'organizzazione:

1. L' Organizzatore aggiunge un utente.
2. Il Sistema convalida l'utente ed aggiorna i membri.

6. Il Sistema notifica errori nei campi:

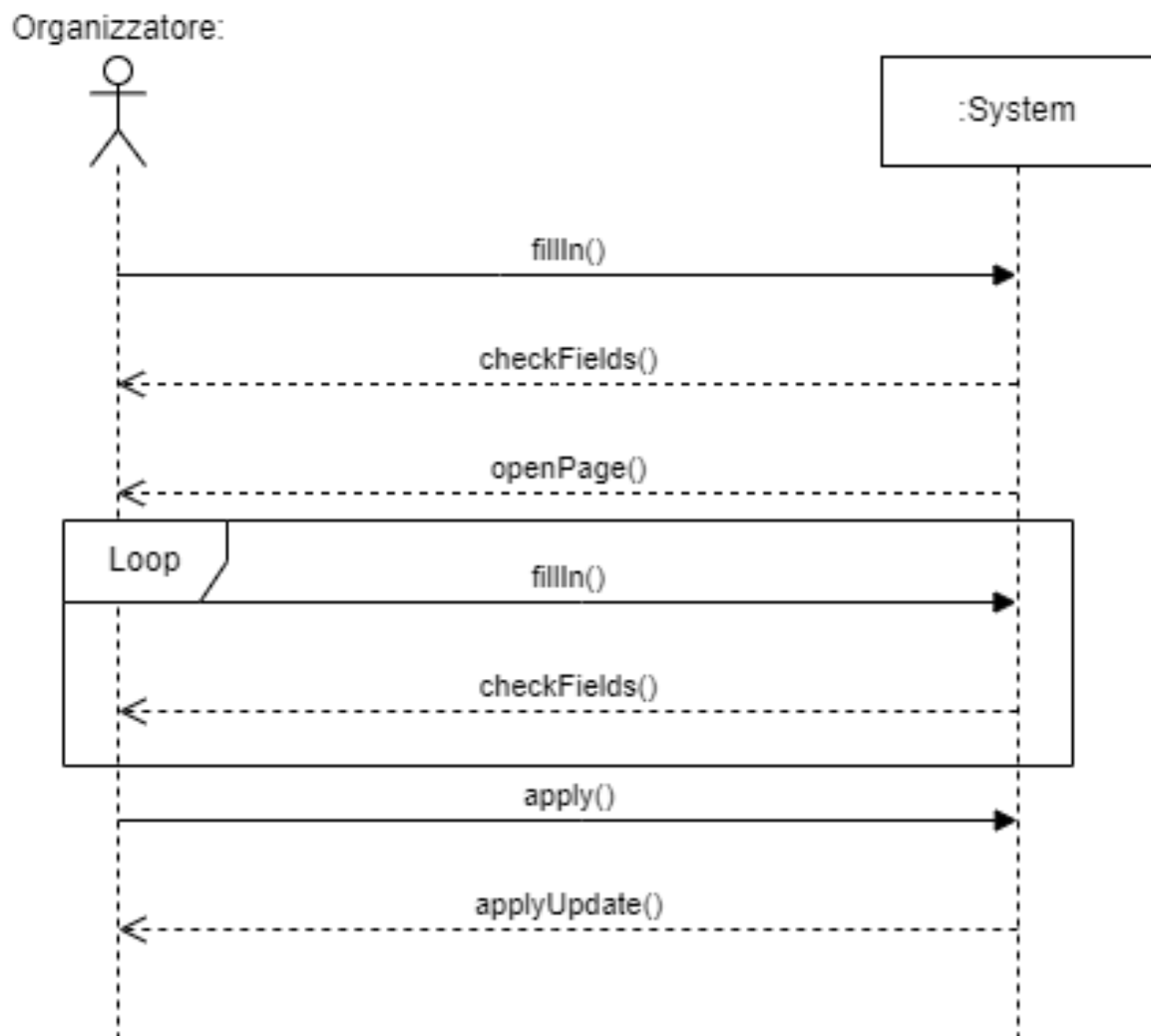
1. L'Organizzatore modifica i campi.
2. Torna al punto 6) dello scenario principale.

### Requisiti Speciali:

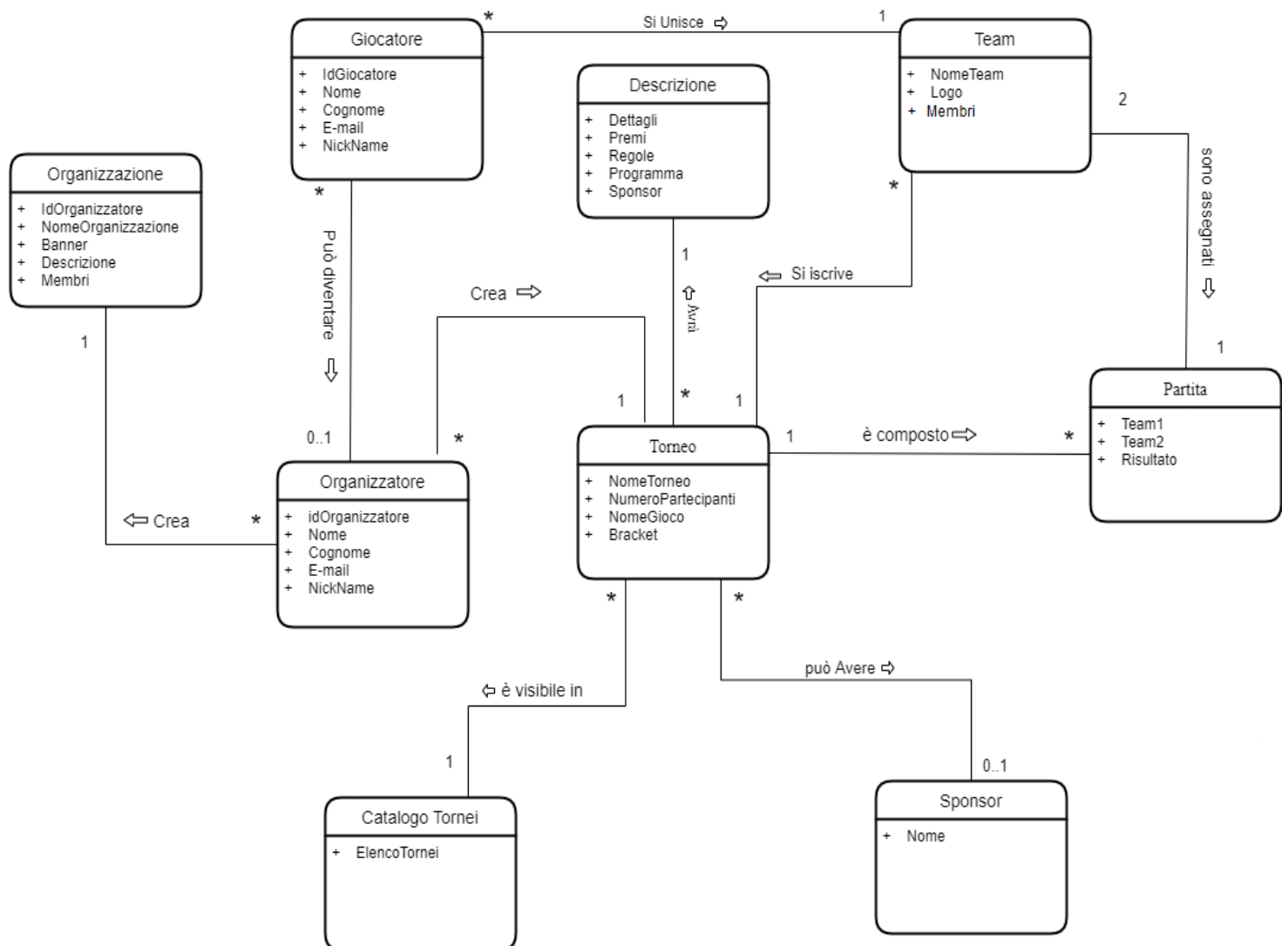
- Latenza nella risposta delle pagine web non superiore a 5 secondi.
- Le finestre di segnalazione degli eventuali errori sono facilmente individuabili dall'utente.
- 

**Frequenza di ripetizione:** Ogni qualvolta l'Organizzatore decide di creare una nuova Organizzazione.

## Diagramma della Sequenza di Sistema



# MODELLO DI DOMINIO



## Entità:

- **Giocatore**
- **Team**
- **Organizzatore**
- **Organizzazione**
- **Torneo**
- **Descrizione**

- Catalogo Tornei
- Sponsor

## Descrizione Modello di Dominio:

Un Giocatore viene creato, si registra come Organizzatore e crea la sua Organizzazione. L'Organizzatore sceglie l'Organizzazione e crea un Torneo che alla creazione sarà aggiunto al catalogo tornei. Quest'ultimo avrà una Descrizione e potrà avere uno Sponsor. Un Giocatore crea un Team o ne entra a far parte. Il creatore del Team iscrive il proprio Team ad un Torneo. Quando il Torneo ha inizio vengono create le Partite alle quali i Team sono assegnati.

## CONTRATTI

### Contratto CO1: makeTournament

**Operazione:** makeTournament()

**Riferimenti:** caso d'uso: Gestire Torneo

**Pre-condizione:**

- Un'istanza per Organizzatore è stata creata.
- Un'istanza per Organizzazione è stata creata.

**Post-condizioni:** L'organizzazione è stata confermata.

### Contratto CO2: fillIn

**Operazione:** fillIn (tournamentName, tournamentAttendeesNumber, gameName, description)

**Riferimenti:** caso d'uso: Gestire Torneo

**Pre-condizione:** nessuna.

**Post-condizioni:** Vengono cambiati gli attributi del torneo: nome, numero di partecipanti, gioco e descrizione.

### **Contratto C03:** endCreation

**Operazione:** endCreation()

**Riferimenti:** caso d'uso: Gestire Torneo

**Pre-condizione:** il Torneo ha assegnati: nome, numero di partecipanti, gioco e descrizione.

**Post-condizioni:** Viene creata un'istanza di Torneo.

### **Contratto C04:** sendResult

**Operazione:** sendResult()

**Riferimenti:** caso d'uso: Avviare e Monitorare l'andamento del torneo

**Pre-condizione:**

- Un'istanza di Torneo è stata creata
- Più istanze di Team sono state create.
- Un'istanza di Partita è stata creata.

**Post-condizioni:** l'istanza Partita cambia l'attributo Risultato.

## Contratto C05: fillIn

**Operazione:** fillIn(organizationName, banner)

**Riferimenti:** caso d'uso: Gestire profilo organizzazione.

**Pre-condizione:**

- Un'istanza di Organizzazione è stata creata.
- Un'istanza di Organizzatore è stata creata.

**Post-condizioni:** Vengono assegnati gli attributi dell'organizzazione: idOrganizzazione, nome, banner.

## Contratto C06: fillIn

**Operazione:** fillIn (description, members)

**Riferimenti:** caso d'uso: Gestire profilo organizzazione.

**Pre-condizione:** Un'istanza di Organizzazione è stata creata.

**Post-condizioni:** Vengono assegnati gli attributi dell'organizzazione: descrizione, membri.

## Contratto C07: fillIn

**Operazione:** fillIn (teamName, logo)

**Riferimenti:** caso d'uso: Gestire team.

**Pre-condizione:** nessuna.

**Post-condizioni:** Vengono assegnati gli attributi del team: nomeTeam, logo.



## Contratto C08: fillIn

**Operazione:** fillIn (description, members)

**Riferimenti:** caso d'uso: Gestire team.

**Pre-condizione:** Un'istanza di Team è stata creata.

**Post-condizioni:** Vengono assegnati gli attributi del team: descrizione, membri

## Contratto C09: chooseTournament

**Operazione:** chooseTournament()

**Riferimenti:** caso d'uso: Iscrizione ad un Torneo.

**Pre-condizione:** nessuna.

**Post-condizioni:** Il Giocatore seleziona il torneo di interesse e lo invia al Sistema.

## Contratto C10: signUpTournament

**Operazione:** signUpTournament()

**Riferimenti:** caso d'uso: Iscrizione ad un Torneo.

**Pre-condizione:** Un'istanza del Torneo scelto è stata creata.

**Post-condizioni:** Il Giocatore richiede l'iscrizione al torneo e lo invia al Sistema.

**Contratto C11:** chooseTeamAndApply

**Operazione:** chooseTeamAndApply()

**Riferimenti:** caso d'uso: Iscrizione ad un Torneo.

**Pre-condizione:** Un'istanza di Team è stata creata.

**Post-condizioni:** Il Giocatore richiede l'iscrizione al torneo invia al Sistema.

## ARCHITETTURA LOGICA DEL SISTEMA

