Tabella dei contenuti

[1. Introduzione 3](#_Toc113530645)

[1.1 Obiettivo 3](#_Toc113530646)

[1.2 Scopo 3](#_Toc113530647)

[1.3 Definizioni, acronimi e abbreviazioni 3](#_Toc113530648)

[1.4 Fonti 3](#_Toc113530649)

[1.5 Struttura del documento 4](#_Toc113530650)

[2. Descrizione generale 4](#_Toc113530651)

[2.1 Inquadramento 4](#_Toc113530652)

[2.2 Macro funzionalità del sistema 4](#_Toc113530653)

[2.3 Caratteristiche utente 5](#_Toc113530654)

[2.4 Vincoli 5](#_Toc113530655)

[2.5 Assunzioni e dipendenze 5](#_Toc113530656)

[2.6 Requisiti da analizzare in futuro 5](#_Toc113530657)

[3. Specifica dei requisiti 5](#_Toc113530658)

[3.1 Requisiti di interfaccia esterna 5](#_Toc113530659)

[3.2 Requisiti funzionali 6](#_Toc113530660)

[3.2.1 Creazione partita 6](#_Toc113530661)

[3.2.2 Ingresso in partita 7](#_Toc113530662)

[3.2.3 Eliminazione partita 7](#_Toc113530663)

[3.2.4 Avvio partita 8](#_Toc113530664)

[3.2.5 Giocata 8](#_Toc113530665)

[3.2.6 Registrazione giocata 9](#_Toc113530666)

[3.2.7 Calcolo del vincitore della mano 9](#_Toc113530667)

[3.2.8 Pescare una carta 10](#_Toc113530668)

[3.2.9 Calcolo del vincitore della partita 10](#_Toc113530669)

[3.3 Requisiti non funzionali 11](#_Toc113530670)

[3.3.1 Requisiti prestazionali 11](#_Toc113530671)

[3.3.2 Database 11](#_Toc113530672)

[3.3.3 Vincoli di progetto 11](#_Toc113530673)

# 1. Introduzione

## 1.1 Obiettivo

L’obiettivo di questo documento è quello di presentare una descrizione dettagliata della struttura e delle funzionalità offerte dal software “Briscola Multiplayer”.

Il documento si rivolge agli utenti del sistema.

## 1.2 Scopo

“Briscola multiplayer” è un videogioco che dà agli utenti la possibilità di connettersi fra di loro, da remoto, a coppie e di giocare a Briscola.

Il sistema gestisce la connessione e la comunicazione tra gli utenti e fornisce una grafica semplice e intuitiva per garantire un’esperienza d’uso piacevole.

## 1.3 Definizioni, acronimi e abbreviazioni

|  |  |
| --- | --- |
| **Termine** | **Descrizione** |
| Briscola | La briscola è un popolare gioco di carte.  Si gioca con un mazzo di 40 carte con i valori asso, 2, 3, 4, 5, 6, 7, fante, cavallo, re di semi italiani o francesi.  Si può giocare in due, in quattro a coppie di due, in tre eliminando il 2 di coppe o fiori, secondo il mazzo usato oppure in sei, tre contro tre, eliminando tutti e quattro i 2. |
| Mazzo di gioco | Mazzo di carte virtuale generato dal server. |
| UUID | Universally Unique Identifier: identificativo generato in modo pseudocasuale.  Non è garantita la loro univocità ma la probabilità che vengano creati duplicati è abbastanza vicina a zero da essere trascurabile. |

## 1.4 Fonti

Nessuna.

## 1.5 Struttura del documento

Il documento prosegue con i seguenti capitoli:

* Capitolo 2: fornisce una panoramica generale del sistema “Briscola Multiplayer” e delle figure d’utente che lo utilizzeranno.
* Capitolo 3: dedicato alla descrizione dei requisiti di interfaccia e dei requisiti funzionali e non funzionali del sistema “Briscola Multiplayer”.

# 2. Descrizione generale

## 2.1 Inquadramento

Il sistema software “Briscola Multiplayer” è indipendente e autonomo.

## 2.2 Macro funzionalità del sistema

Il sistema è costituito da due componenti: il server, e il client.

Funzionalità del server:

* Accetta connessioni da parte di istanze client
* Permette alle istanze client connesse di creare una nuova partita
* Permette alle istanze client connesse di unirsi ad una partita creata da un’altra istanza client
* Assegna un identificatore ad ogni istanza client che crea o si unisce ad una partita
* Crea e invia il mazzo di gioco quando richiesto da un’istanza client
* Gestisce lo scambio di informazioni tra le istanze client connesse alla stessa partita: carte giocate e carte pescate

Funzionalità del client:

* Fornisce un’interfaccia grafica all’utente
* Permette all’utente di scegliere se creare una nuova partita o se unirsi ad una partita esistente e comunica la scelta al server
* Avvia la partita quando riceve il mazzo di gioco dal server
* Gestisce la logica di gioco durante la partita e invia al server le carte giocate dall’utente

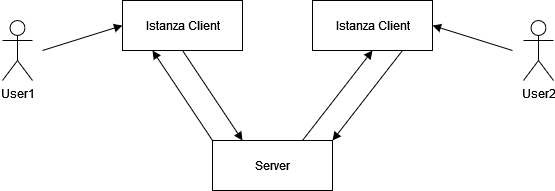


Figura - Interazione Client-Server

## 2.3 Caratteristiche utente

“Briscola Multiplayer” si rivolge ad un utenza con dimestichezza con il terminale, in grado di eseguire su di esso comandi indicati in una guida, fornita dallo sviluppatore.

## 2.4 Vincoli

Devono essere gestiti i casi in cui il server non è in funzione o in caso di disconnessione di quest’ultimo o degli utenti.

## 2.5 Assunzioni e dipendenze

Il sistema richiede che sia installato python3.8 o superiore.

## 2.6 Requisiti da analizzare in futuro

Il sistema dovrà prevedere l’installazione e l’esecuzione tramite eseguibili, per la parte client.

# 3. Specifica dei requisiti

## 3.1 Requisiti di interfaccia esterna

Il client dovrà presentare all’utente una finestra, la quale dovrà avere quattro diversi tipi di schermate:

* Schermata home: da mostrare quando l’utente avvia il programma client, Conterrà due bottoni per permettere all’utente di scegliere se creare una nuova partita o se unirsi ad una partita esistente
* Schermata di caricamento: da mostrare quando l’utente sceglie di creare una nuova partita. Conterrà un messaggio che indica il fatto che sia in corso la ricerca di un avversario e un pulsante per dare la possibilità all’utente di annullare l’operazione e tornare alla schermata home
* Schermata di gioco: da mostrare quando, dopo che l’utente ha effettuato la scelta nella schermata home, un avversario si collega alla partita. Conterrà il mazzo di carte, la carta briscola, le carte in mano ad entrambi i giocatori, i mazzi delle carte vinte per entrambi i giocatori e le carte giocate da entrambi i giocatori.
* Schermata di errore: da mostrare quando il server o l’avversario si disconnettono. Conterrà un messaggio di errore e un bottone per permettere di tornare alla schermata home.

In ogni schermata dovrà essere presente un ulteriore bottone che permette all’utente di chiudere il programma client.

Il server dovrà permettere ad istanze client di connettersi ad esso e di comunicare con esso durante lo svolgimento delle partite.

Gli utenti non dovranno avere la possibilità di interagire direttamente con il server.

## 3.2 Requisiti funzionali

### 3.2.1 Creazione partita

|  |  |
| --- | --- |
| **Introduzione** | **Attori coinvolti**  Utente |
| **Descrizione generale della funzione**  Consente ad un utente di creare una nuova partita. |
| **Input** | **Descrizione generica dei dati**  Nessuno |
| **Descrizione**  **(Processo)** | **Sequenza di operazioni**  L’utente, dalla schermata home del client, sceglie di creare una nuova partita.  Successivamente attende che un avversario si colleghi alla partita appena creata oppure annulla l’operazione causando l’eliminazione della partita.  L’istanza client comunica le scelte dell’utente al server, il quale procede all’assegnazione di un identificatore al client (UUID) e alla creazione/eliminazione della partita. |
| **Risposta ed eventuali anomalie**  Messaggio di errore nel caso in cui il server non sia raggiungibile.  Eliminazione della partita appena creata nel caso in cui l’utente decidesse di annullare l’operazione. |
| **Parametri che impattano sull’output**  Nessuno |
| **Output** | Identificatore dell’istanza client e identificatori utili all’istanza client a comunicare con l’istanza della partita creata dal server. |

### 3.2.2 Ingresso in partita

|  |  |
| --- | --- |
| **Introduzione** | **Attori coinvolti**  Utenti |
| **Descrizione generale della funzione**  Consente ad un utente di unirsi ad una partita creata da un altro utente. |
| **Input** | **Descrizione generica dei dati**  Nessuno |
| **Descrizione**  **(Processo)** | **Sequenza di operazioni**  L’utente, dalla schermata home del client, sceglie di unirsi ad una partita esistente.  L’istanza client comunica la scelta al server.  Il server controlla che siano presenti partite in attesa di un avversario, se è presente almeno una partita in attesa associa il client richiedente alla partita creata meno di recente e procede all’assegnazione di un identificatore all’istanza client (UUID). |
| **Risposta ed eventuali anomalie**  Messaggio di errore nel caso non ci siano partite in attesa.  Si dà la possibilità all’utente di tornare alla schermata home. |
| **Parametri che impattano sull’output**  La presenza o meno di partite in attesa. |
| **Output** | Nel caso in cui l’ingresso in partita vada a buon fine: identificatore dell’istanza client e identificatori utili all’istanza client a comunicare con l’istanza della partita creata dal server.  Nel caso in cui l’ingresso in partita non vada a buon fine: valore che indichi questo fatto. |

### 3.2.3 Eliminazione partita

|  |  |
| --- | --- |
| **Introduzione** | **Attori coinvolti**  Utente |
| **Descrizione generale della funzione**  Eliminazione di una partita. |
| **Input** | **Descrizione generica dei dati**  Identificatore dell’istanza della partita (che può corrispondere con l’identificatore dell’utente creatore). |
| **Descrizione**  **(Processo)** | **Sequenza di operazioni**  Eliminazione di una partita nel caso in cui un utente decida di annullare l’operazione di creazione oppure quando uno degli utenti partecipanti si disconnette.  L’operazione avviene sul server ed è attivata da un’istanza client. |
| **Risposta ed eventuali anomalie**  Nessuna |
| **Parametri che impattano sull’output**  Nessuno |
| **Output** | L’istanza della partita corrispondente all’identificatore fornito non sarà più accessibile. |

### 3.2.4 Avvio partita

|  |  |
| --- | --- |
| **Introduzione** | **Attori coinvolti**  Utenti connessi alla stessa partita |
| **Descrizione generale della funzione**  Avvio della partita per gli utenti connessi ad essa.  Questo avviene quando un secondo utente si unisce ad una partita creata da un altro utente. |
| **Input** | **Descrizione generica dei dati**  Nessuno |
| **Descrizione**  **(Processo)** | **Sequenza di operazioni**  Il server procede alla creazione del mazzo di gioco, alla scelta della carta briscola e alla distribuzione della carte agli utenti.  Viene poi comunicato il tutto agli utenti connessi alla partita. |
| **Risposta ed eventuali anomalie**  Messaggio di errore nel caso in cui uno degli utenti si disconnettesse.  Si dà la possibilità all’utente di tornare alla schermata home. |
| **Parametri che impattano sull’output**  Nessuno |
| **Output** | Mazzo di gioco, carta briscola e carte in mano ai giocatori. |

### 3.2.5 Giocata

|  |  |
| --- | --- |
| **Introduzione** | **Attori coinvolti**  Utenti connessi alla stessa partita |
| **Descrizione generale della funzione**  Registrazione della carta giocata dall’utente e comunicazione della stessa all’avversario. |
| **Input** | **Descrizione generica dei dati**  Carta giocata dall’utente |
| **Descrizione**  **(Processo)** | **Sequenza di operazioni**  L’istanza client, se è il turno dell’utente a cui è associata, attende che l’utente clicchi una carta fra quelle che ha in mano.  Quando una delle carte in mano all’utente viene cliccata, essa viene registrata come giocata e comunicata all’avversario tramite il server.  Le istanze client provvedono anche ad aggiornare la schermata di gioco spostando la carta giocata al centro. |
| **Risposta ed eventuali anomalie**  Nessuna |
| **Parametri che impattano sull’output**  Nessuno |
| **Output** | Carta giocata dall’utente. |

### 3.2.6 Registrazione giocata

|  |  |
| --- | --- |
| **Introduzione** | **Attori coinvolti**  Utente |
| **Descrizione generale della funzione**  Aggiornamento dello stato di gioco in seguito alla giocata di un utente. |
| **Input** | **Descrizione generica dei dati**  Carta giocata dall’utente |
| **Descrizione**  **(Processo)** | **Sequenza di operazioni**  L’istanza client, dopo aver registrato la giocata dell’utente, oppure dopo aver ricevuto la carta giocata dall’avversario, aggiorna:   * Il turno di gioco * Nel caso in cui il mazzo sia finito (indicazione del fatto che la partita è finita), modifica lo stato di gioco di conseguenza. |
| **Risposta ed eventuali anomalie**  Nessuna |
| **Parametri che impattano sull’output**  Nessuno |
| **Output** | Modifiche allo stato di gioco. |

### 3.2.7 Calcolo del vincitore della mano

|  |  |
| --- | --- |
| **Introduzione** | **Attori coinvolti**  Utenti connessi alla stessa partita |
| **Descrizione generale della funzione**  Calcolo del vincitore della mano in base alle carte giocate. |
| **Input** | **Descrizione generica dei dati**  Carte giocate da entrambi gli utenti. |
| **Descrizione**  **(Processo)** | **Sequenza di operazioni**  Selezione del vincitore in base alle regole della Briscola.  L’utente vincitore vince le carte giocate da entrambi gli utenti e gli/le viene assegnato il primo turno della prossima mano.  Operazione che avviene nei client. |
| **Risposta ed eventuali anomalie**  Nessuna |
| **Parametri che impattano sull’output**  Nessuno |
| **Output** | Vincitore della mano. |

### 3.2.8 Pescare una carta

|  |  |
| --- | --- |
| **Introduzione** | **Attori coinvolti**  Utenti connessi alla stessa partita |
| **Descrizione generale della funzione**  Permette all’utente di pescare una carta dal mazzo. |
| **Input** | **Descrizione generica dei dati**  Mazzo e carta briscola |
| **Descrizione**  **(Processo)** | **Sequenza di operazioni**  Quando il vincitore della mano è stato calcolato, entrambi i giocatori connessi alla partita pescheranno una carta dal mazzo (il primo a pescare sarà il vincitore), richiedendola al server.  Gli utenti dovranno poter richiedere al server la carta pescata dall’avversario.  Le carte pescate andranno a sostituire la carta precedentemente giocata da ciascuno degli utenti, aggiornando le loro carte in mano. |
| **Risposta ed eventuali anomalie**  Nessuna |
| **Parametri che impattano sull’output**  Nessuno |
| **Output** | Carte in mano aggiornate |

### 3.2.9 Calcolo del vincitore della partita

|  |  |
| --- | --- |
| **Introduzione** | **Attori coinvolti**  Utenti connessi alla stessa partita |
| **Descrizione generale della funzione**  Calcolo del vincitore della partita in base alle carte vinte. |
| **Input** | **Descrizione generica dei dati**  Carte vinte dagli utenti durante la partita. |
| **Descrizione**  **(Processo)** | **Sequenza di operazioni**  Per ogni utente si sommano i punteggi associati ad ogni carta vinta.  Si valuta quale giocatore ha totalizzato più punti, il quale sarà il vincitore della partita, oppure, nel caso in cui i punteggi siano identici, si sarà verificato un pareggio.  Operazione eseguita nel client. |
| **Risposta ed eventuali anomalie**  Nessuna |
| **Parametri che impattano sull’output**  Nessuno |
| **Output** | Vincitore/pareggio |

## 3.3 Requisiti non funzionali

### 3.3.1 Requisiti prestazionali

Il sistema deve avere tempi di risposta ragionevolmente contenuti, in modo da rendere piacevole l’esperienza d’uso.

### 3.3.2 Database

Non necessario.

### 3.3.3 Vincoli di progetto

Nessuno