****

**DOCUMENTAZIONE DEI REQUISITI**

**Tombola**

**Lotti Giovanni 447123**

**Montebugnoli Leonardo 447525**

**Oprea Ancuta Andreea 435670**

**Pasotti Alessandro 446419**

**Poirè Matteo 440193**

**Rossi Alessandro 449396**

SOMMARIO

**1.INTRODUZIONE ……………………………………………………3**

1.1 Scopo del documento …………………………………………………………..3

1.2 Scopo del prodotto ……………………………………………………………..3

1.3 Glossario ………………………………………………………………………..3

1.4 Regole del gioco ………………………………………………………………...4

1.5 Descrizione del documento …………………………………………………..4

**2.DESCRIZIONE GENERALE ………………………………………5**

2.1 Prospettiva del Prodotto ……………………………………………………...5

2.2 Funzioni del Prodotto …………………………………………………………5

2.3 Caratteristiche Utente ………………………………………………………..5

2.4 Vincoli Generali ………………………………………………………………..5

**3.CASI D’USO …………………………………………………………..6**

**4.DEFINIZIONE DEI REQUISITI FUNZIONALI ……………….7**

4.1 Tecnologie Usate ………………………………………………………………7

4.2 Server ……………………………………………………………………………7

4.3 Classi Json ……………………………………………………………………...8

4.4 Funzionamento del Programma …………………………………………….8

**5.REQUISITI NON FUNZIONALI ………………………………….9**

5.1 Requisiti di Prodotto ………………………………………………………….9

5.2 Requisiti di Processo ………………………………………………………….9

5.3 Requisiti Esterni ………………………………………………………………9

**6.REQUISITI DI SISTEM …………………………………………..10**

**1.INTRODUZIONE**

**1.1 Scopo Del Documento**

Il fine di questo documento è quello di fornire una descrizione esaustiva dell’applicazione, delineandone i vari casi d’uso, il comportamento in determinate situazioni, i vincoli che la caratterizzano e gli sviluppi futuri.

**1.2 Scopo Del Prodotto**

Una piattaforma per il gioco della “Tombola”. Un numero variabile di giocatori deve potersi iscrivere ad una sessione del gioco in qualunque momento e le partite dovranno susseguirsi una dopo l’altra.

**1.3 Glossario**

* **User :** Utente che partecipa alla sessione della tombola.
* **Client :** Applicativo locale in funzione su ogni dispositivo usato dagli users.
* **Server :** Applicativo remoto che fornisce servizi ai Clients.
* **Bot :** Unità di intelligenza artificiale, in gioco.
* **PC :** Personal computer.
* **Username :**  Nome utente.
* **Api :** Acronimo di "application programming interfacce" indica ogni insieme di procedure disponibili al programmatore, di solito raggruppate a formare un set di strumenti specifici per l’espletamento di un determinato compito all’interno di un certo programma.
* **Menu :** Rappresenta una lista di comandi che l’utente ha a disposizione per istruzioni all’applicazione.

**1.4 Regole Del Gioco**

l gioco della tombola può essere giocato da un numero qualunque di giocatori .   
I giocatori dispongono di una o più cartelle, composte ognuna da tre righe, su ciascuna delle quali sono riportati cinque numeri da 1 a 90. Ogni volta che il numero estratto è presente su una o più delle sue cartelle, il giocatore copre la casella corrispondente.   
Lo scopo del gioco è quello di realizzare la combinazione della tombola, ovvero arrivare per primi a coprire tutti i numeri presenti su una delle proprie cartelle. Normalmente vengono anche assegnati premi minori per combinazioni intermedie, come l'ambo (vinto dal primo giocatore che copre due numeri presenti sulla stessa riga di una cartella), il terno (tre numeri sulla stessa riga), la quaterna (quattro numeri sulla stessa riga) e la cinquina (tutti e cinque i numeri della riga).

**1.5 Descrizione Del Documento**

Il seguente documento è diviso in 5 sezioni:

**Glossario:** Raccolta dei vocaboli più specifici presenti nel documento, accompagnati dalla relativa spiegazione.

**Modelli del Sistema:** Analisi dei requisiti del sistema, fatta partendo dai diversi scenari di utilizzo dell’applicazione: l’elenco dei casi d’uso sarà accompagnato da appositi diagrammi UML per facilitare la comprensione delle funzionalità del sistema.

**Definizione dei Requisiti Funzionali:** Esposizione dei servizi offerti dall’applicazione a seconda dei vari contesti di utilizzo.

**Definizione dei Requisiti Non Funzionali:** vincoli del sistema e del processo di sviluppo.

**Specifica dei Requisiti:** Descrizione dettagliata dei requisiti funzionali.

**2.DESCRIZIONE GENERALE**

**2.1 Prospettiva del Prodotto**

Il sistema software “Tombola” non è parte e non deve integrarsi in alcun sistema software, essendo un sistema completamente autonomo.

**2.2 Funzioni del Prodotto**

La funzione principale del prodotto è consentire all’utente finale di testare la sua fortuna e giocare a tombola.

**2.3 Caratteristiche Utente**

Il sistema software “Tombola” è rivolto ad una utenza con discreta conoscenza del dominio applicativo ma senza particolari conoscenze informatiche. All’utente è richiesta solamente una conoscenza informatica di base (alfabetizzazione informatica).

**2.4 Vincoli Generali**

Per poter utilizzare il software è necessario un PC con connessione ad Internet.

**3.CASI D’USO**

**Gioco Tombola**

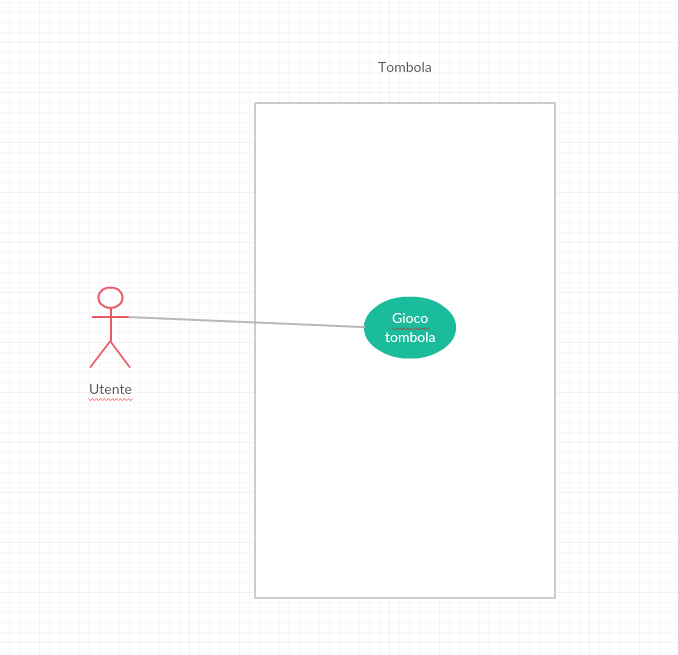
**Nome:** Gioco Tombola.

**Goal:** Giocare a tombola online.

**Livello:** Utente.

**Attori:** Utente.

**Descrizione**: *L'UTENTE* sul suo PC potra' utilizzare il software per giocare al gioco della tombola online. *L'UTENTE* inserira' il suo username e potra' scegliere di unirsi a una sessione del gioco. Una volta trovata la sessione desiderata provera' ad unirsi e in caso di successo potra' scegliere tra un numero di cartelle variabile. In caso di insuccesso ritornera' alla schermata principale e potra' scegliere tra altre sessioni e si ripete fino a quando non e' in gioco. Una volta in gioco *l'UTENTE* vedra' la sua cartella con i suoi numeri e il procedimento di gioco e' lo stesso del gioco fisico.



**4.DEFINIZIONE DEI REQUISITI FUNZIONALI**

In questa sezione verranno esposti i principali requisiti funzionali dell’applicazione Tombola, ossia le funzionalità che devono essere implementare per soddisfare le specifiche dell’applicazione**.**

L’applicazione consiste in due parti: un client e un server. L’utente potrà connettersi tramite un client al server.

Il client è un applicativo software su PC. L’utente una volta entrato in una sessione della tombola potrà scegliere tra 1 o più cartelle, al massimo 6.

**4.1 Tecnologie Usate**

Nel nostro progetto le tecnologie utilizzate sono:

* Jetty server come server che gestisce il gioco centrale
* http come protocollo di comunicazione
* json come tipologia di messaggio di risposta
* jsonriter come libreria di supporto per i json
* servlets come oggetto su server per comunicazione

**4.2 Server**

Il server ha principalmente 3 servlet:

1. /aggiungiPlayer: servlet preposta per aggiungere un player.Restituisce un messaggio formattato json con le cartelle usalìte e l’username usato.
2. /estrazioni: servlet che risponde con un messaggio formattato json con i numeri attualmente estratti
3. /controlloCartelle: servlet usata per controllare le cartelle.

Lato server vengono svolte le attività che svolgerebbe un vero tomboliere, ovvero di occupa di distribuire le cartelle ai vari client, memorizzando a sua volta i player “logici” con le cartelle corrispondenti.

Se riceve una richiesta di aggiunta giocatore, deve creare e memorizzare un’istanza della classe Player con le rispettive N cartelle richieste. Infine, formatta un json e lo restituisce al client con le informazioni richieste.

Se riceve una richiesta delle estrazioni, semplicemente formatta un json con le estrazioni e le restituisce al client.

Se riceve una richiesta di controllo della cartella, preleva l’indice della cartella, dal player con l’username specificato, esegue il controllo e infine formatta la risposta in json e la restituisce al client.

**4.3 Classi Json**

Per gestire la serializzazione dei Json sono state usate due classi principali:

* JsonWriter: classe che serializza in json gli oggetti
* JsonReader: classe che traduce un json in oggetti logici.

**4.4 Funzionamento del Programma**

A seguito della pressione del tasto PLAY, il client si connette e richiede a /*aggiuntaPlayer* un player con l’username specificato nella gui e il numero di cartelle voluto. Il server riceve la richiesta, crea un nuovo player e impacchetta un json con le informazioni. Il client riceve la risposta e spacchetta il json.

Se l’utente chiama un ambo, invia la richiesta a /*controlloCartelle* segnalando il proprio username e l’indice della cartella da controllare. Il server riceve la richiesta, ottiene il player specificato e controlla la cartella segnalata. Così anche per le altre chiamate: terno, quaterno, cinquina e tombola.

**5.REQUISITI NON FUNZIONALI**

I vincoli cui l’applicazione deve conformarsi nell’eseguire le operazioni. Tali requisiti si dividono in tre macro tipologie:

* Di prodotto
* Di processo
* Esterni

**5.1 Requisiti di Prodotto**

**Velocità di esecuzione:** Una volta premuto sull’icona dell’applicazione, essa si deve avviare in un tempo inferiore a 2 secondi.

**Annullamento di operazioni:** Garantisce all’utente di annullare un’operazione di una sequenza in qualunque punto.

**5.2 Requisiti di Processo**

**Compatibilità:** L’applicazione deve essere compatibile con sistemi operativi Windows, Mac e Linux.

**Input:** L’input dei dati necessari al funzionamento dell’applicazione verranno inseriti tramite una tastiera che comparirà sullo schermo del device o tramite tastiera fisica se il device ne permette il collegamento. Un altro tipo di input sarà quello generato dal dispositivo gps integrato nel device.

**5.3 Requisiti Esterni**

**Privacy:** L’applicazione non dovrà memorizzare dati utente al di fuori del device.

**6.REQUISITI DI SISTEMA**

Sia Client sia Server sono applicativi sviluppati mediante il linguaggio Java. Lo sviluppo dell’interfaccia grafica è svolto mediante la libreria JavaFX, in particolare usando lo strumento “Java Scene Builder” che permette una facile implementazione dell'interfaccia grafica mediante la tecnica drag and drop. Inoltre permette di esportare l'intefaccia in formato FXML che e' importabile in java.

Dal punto di vista implementativo il sistema dovra' svolgere, lato server, la funzione di tomboliere e lato client dovra' fornire un ambiente di gioco il piu' possibile simile alla realta'.