

Tank Battle - Documento di Analisi

Khanej Olgerti – Matr. 564455

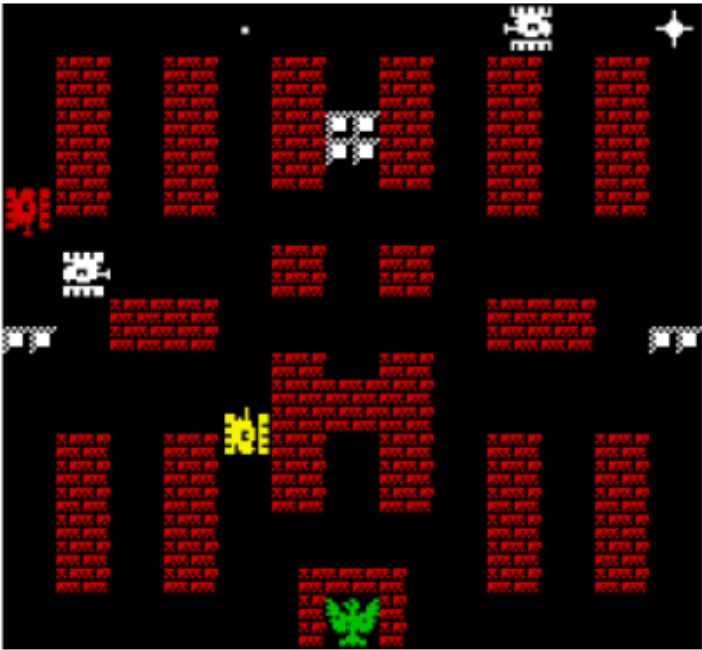
Ingegneria Informatica – Corso di Programmazione Avanzata 2019/2020

VISTA STATICA

Tank Battle

TANK BATTLE

Player Score: 5



RANKING

| # | Username | Score |
|---|--------------|-------|
| 1 | picci85 | 102 |
| 2 | gerti98 | 22 |
| 3 | Jack67 | 17 |
| 4 | aixel402 | 15 |
| 5 | dark_warrior | 12 |

REGISTRAZIONE

Username

dark_warrior

Play

VISTA DINAMICA

Caso d'uso 1 (Inserimento dati giocatore e nuovo gioco)

1. **FOREACH** Partita archiviata
 - 1.1. Il Sistema visualizza nella tabella dei ranking: #, Username e il relativo Score, ordinati per Score decrescente.
2. L' Utente inserisce Username
3. **IF** Utente preme tasto PLAY
 - 3.1. Il Sistema azzera il campo Player Score
 - 3.2. Il Sistema fa iniziare la partita visualizzando il tankGiocatore.

- 3.3. **IF** Il giocatore preme i tasti configurati per il movimento (W,A,S,D)
 - 3.3.1. Il Sistema sposta il tankGiocatore se possibile nella relativa direzione
- 3.4. **IF** Il giocatore preme il tasto configurato per sparare (J) e nessun altro proiettile del Tank giocatore è attivo
 - 3.4.1. Il Sistema fa sparare il tankGiocatore nella direzione in cui esso è rivolto
- 3.5. **IF** il proiettile del giocatore colpisce un tankNemico
 - 3.5.1. Il tankNemico ed il proiettile del giocatore vengono eliminati dal gioco
 - 3.5.2. Il Sistema aumenta il Player Score di 1
- 3.6. **IF** un proiettile colpisce un BloccoDiGioco
 - 3.6.1. Il proiettile viene eliminato dal gioco
- 3.7. **IF** il tankGiocatore viene colpito dal tankNemico o dal proiettileNemico
 - 3.7.1. Game Over
- 3.8. **IF** un tankNemico colpisce, con un proiettile, la base o tocca la base
 - 3.8.1. Game Over
- 3.9. **IF** Game Over
 - 3.9.1. Il Sistema aggiorna la tabella dei ranking con l'ultimo punteggio.
 - 3.9.2. Il Sistema blocca il gioco.
 - 3.9.3. Torna al Punto 1.

FILE DI CONFIGURAZIONE LOCALE IN XML

Il sistema all'avvio legge da file di configurazione locale i seguenti dati:

- Il numero di campi da visualizzare nella tabella dei Ranking
- L'età massima (in giorni) della partita per poter essere visualizzata nella tabella dei Ranking
- L'indirizzo IP del client, l'indirizzo IP e la porta del server di log
- User e Password del Database

CACHE LOCALE DEGLI INPUT

Alla chiusura il Sistema salva su file binario: Username, il Player Score, la posizione e la direzione del tankGiocatore e dei tankNemici (se la partita è in corso).

All'avvio il Sistema carica da file binario i suddetti dati.

ARCHIVIO

Il sistema archivia i seguenti dati:

- Username, punteggio, data della partita e identificatore partita relativa al punteggio per ogni partita.

FILE DI LOG REMOTO IN XML

Il sistema invia una riga di log al server di log per ogni evento specificato di seguito:

- Avvio dell'applicazione ("AVVIO")
- Pressione del pulsante "PLAY"

- Termine dell'applicazione ("TERMINE")

La riga di log contiene: nome dell'applicazione, indirizzo IP del client, data corrente, ora corrente, l'etichetta associata all'evento.

Tank Battle - Documento di Progetto

Khanej Olgerti – Matr. 564455

Ingegneria Informatica – Corso di Programmazione Avanzata 2019/2020

RESPONSABILITA' DELLE CLASSI

AreaRegistrazione: Inizializza il campo della GUI relativa all'inserimento dell'username; gestisce avvio della partita mediante relativo pulsante; controlla se prima all'avvio della partita non vi è alcun username inserito.

BloccoDiGioco: Memorizza tutte le informazioni che descrivono un blocco all'interno del gioco Tank Battle; gestisce posizionamento dell'elemento di gioco tramite specifici metodi setter; gestisce l'eventuale controllo di collisione tra un blocco ed un altro elemento di gioco (tank o proiettile).

CacheStatoTankBattle: Memorizza su file binario lo stato del gioco (posizioni e direzioni dei vari tank), il campo username ed il punteggio corrente della partita; reimmette tali dati nelle relative locazioni all'avvio dell'applicativo.

DatabaseRanking: restituisce una classifica dei punteggi delle precedenti partite dei vari giocatori.

EventoDiNavigazioneGUI: (client) contiene le informazioni degli eventi di Log quali avvio e chiusura dell'applicativo, tasto play premuto; si serializza in XML; invia l'evento di LOG in XML alla classe LogDiNavigazioneGUI.

LogDiNavigazioneGUI: (server) riceve un evento di log XML; invoca la validazione dell'evento tramite XML schema, aggiunge la riga XML al file di log incrementalmente e la stampa in output.

ParametriDiConfigurazione: legge il file di configurazione XML; invoca la validazione del file di configurazione XML; deserializza il contenuto XML come oggetto Java; mette a disposizione di qualsiasi classe i parametri username e password di accesso al database, ipClient, ipServer e numero di porta del server per la comunicazione tra il server di log e l'applicativo, i numeri di campi del ranking da visualizzare in tabella visuale.

Partita: classe *bean* che memorizza le informazioni di una partita; mette a disposizione alle classi *DatabaseRanking* e *TabellaVisualeRanking* tali informazioni.

ProiettileDiGioco: Memorizza tutte le informazioni che descrivono un proiettile (sparato dai vari tank) all'interno del gioco Tank Battle; gestisce posizionamento dell'elemento di gioco tramite specifici metodi setter e getter.

ServerLog: (server) riceve un evento di log XML; invoca la validazione dell'evento tramite XML schema, aggiunge la riga XML al file di log incrementalmente e la stampa in output.

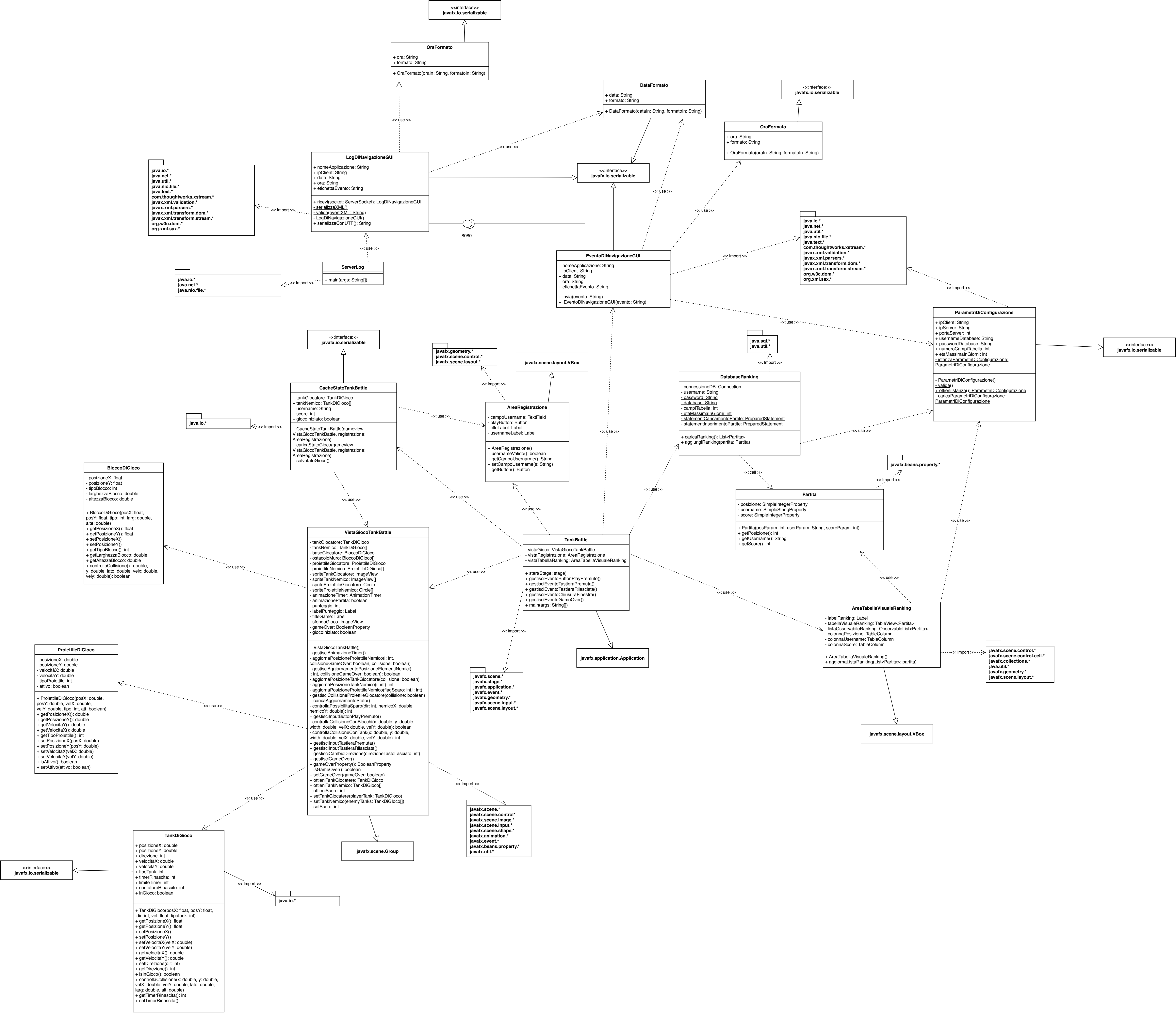
AreaTabellaVisualeRanking: inizializza la parte della GUI relativa alla visualizzazione della tabella del ranking.

TankBattle: inizializza la GUI: in particolare inizializza le varie aree *TabellaVisualeRanking*, *AreaRegistrazione*, *VistaGiocoBattle*; effettua il demultiplexing dei vari eventi generabili dai vari

elementi della GUI e dalla tastiera nell'area responsabile alla gestione; coordina memorizzazione e caricamento dei dati in *CacheStatoTankBattle*; coordina invio eventi tramite *EventoDiNavigazioneGUI*, e coordina aggiornamento *AreaTabellaVisualeRanking* tramite apposito event handler di fine partita.

TankDiGioco: Memorizza tutte le informazioni che descrivono un Tank all'interno del gioco Tank Battle; gestisce posizionamento dell'elemento di gioco tramite specifici metodi setter e getter. gestisce l'eventuale controllo di collisione tra un tank ed un altro elemento di gioco (proiettile).

VistaGiocoBattle: Area della GUI che inizializza l'area di gioco; gestisce le animazioni dei vari elementi di gioco e determina la loro presenza durante la partita gestendo le collisioni; gestisce gli input da tastiera per coordinare il comportamento del tank del giocatore; determina eventi di gameover.



Tank Battle – Manuale Utente

Khanej Olgerti – Matr. 564455

Ingegneria Informatica – Corso di Programmazione Avanzata 2019/2020

INTERFACCIA APPLICATIVO

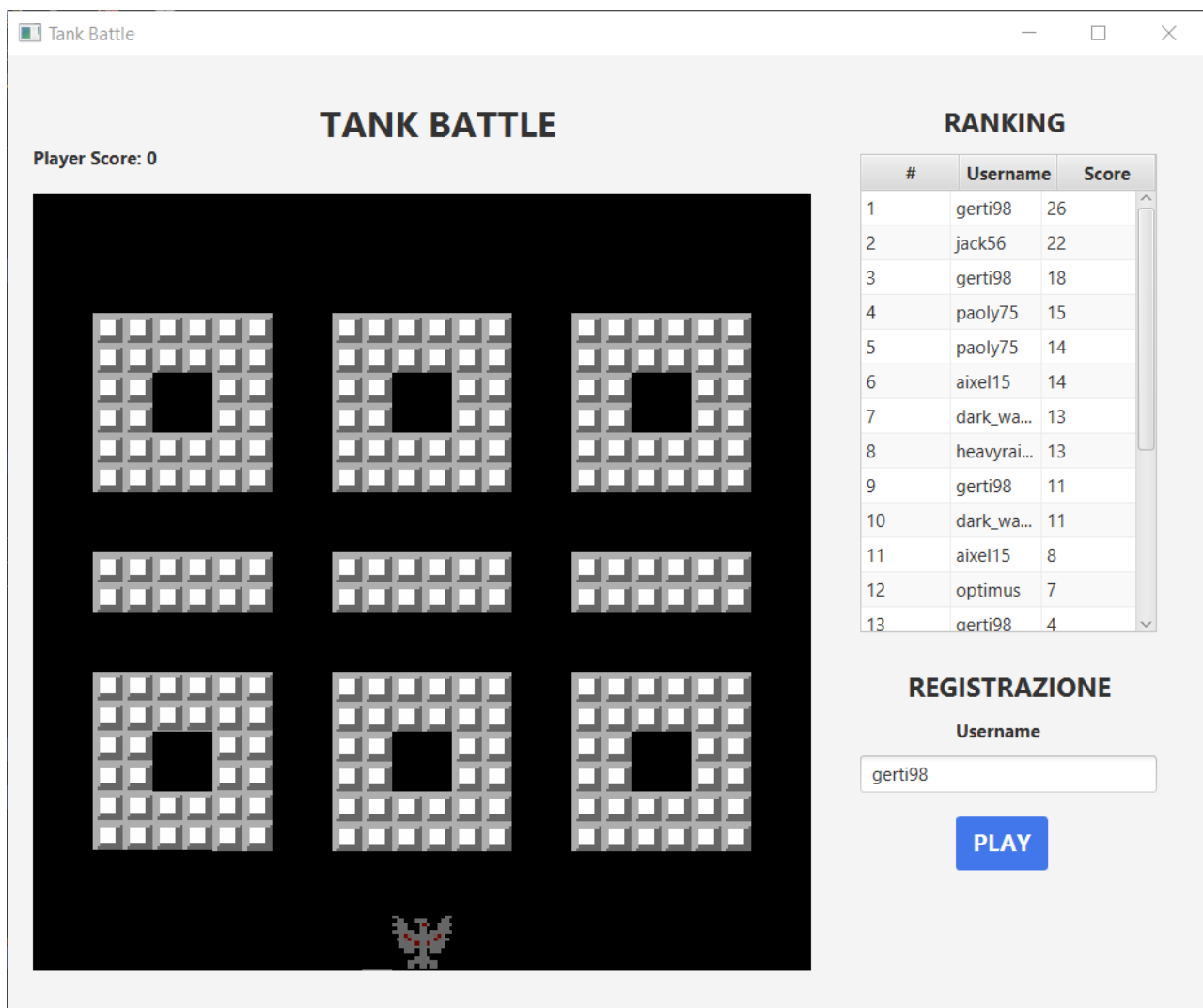


Fig.1 Interfaccia di Tank Battle

L'interfaccia di gioco si presenta suddivisa in tre aree: l'area Registrazione, l'area Ranking e l'area di gioco Tank Battle.

L'area Registrazione permette di **iniziare la partita** ed **inserire i relativi dati del giocatore**.

L'area Rankink mostra in tabella i dati relativi alle partite dei giocatori, in ordine decrescente di punteggio. E' possibile trascinare la scrollbar per visualizzare tutti i campi della tabella visuale desiderati. La tabella visuale contiene i seguenti campi:

- #: indica la posizione in classifica del giocatore con il punteggio della corrispondente partita.
- Username: indica il nome del giocatore che ha effettuato il punteggio corrispondente.
- Score: indica il punteggio, in termini di tank nemici abbattuti, ottenuto dal giocatore.

All'avvio dell'applicativo verranno mostrati in tabella Ranking i risultati delle ultime partite (vedi Configurazione XML).

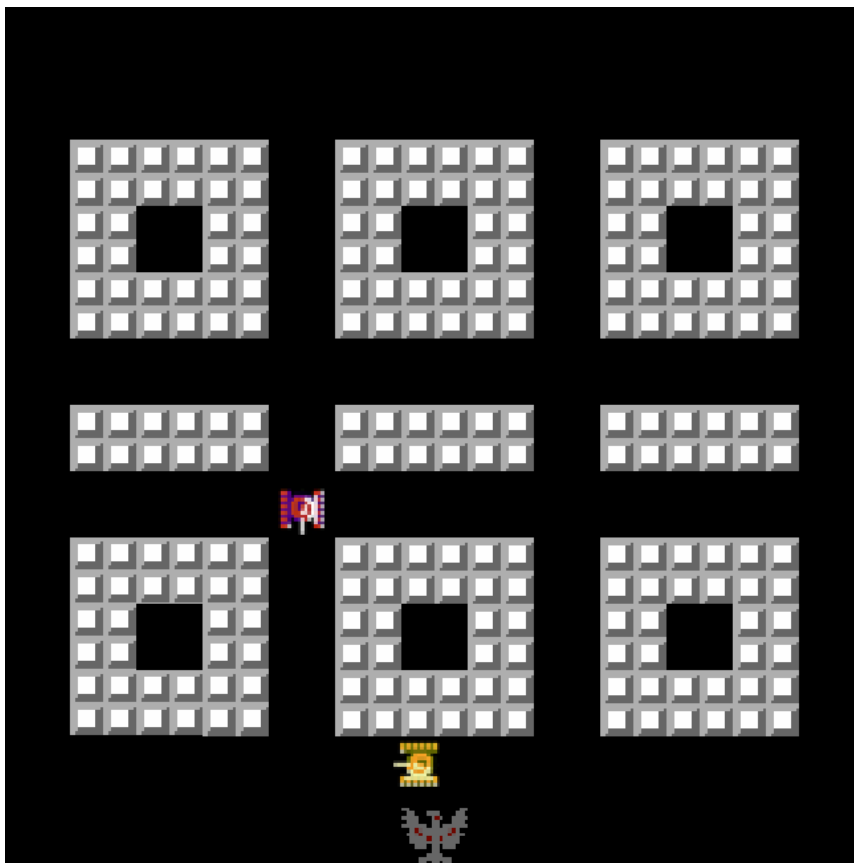


Fig. 3: Interfaccia di Gioco Tank Battle

| RANKING | | |
|---------|-------------|-------|
| # | Username | Score |
| 1 | gerti98 | 26 |
| 2 | jack56 | 22 |
| 3 | gerti98 | 18 |
| 4 | paoly75 | 15 |
| 5 | paoly75 | 14 |
| 6 | axel15 | 14 |
| 7 | dark_wa... | 13 |
| 8 | heavyrai... | 13 |
| 9 | gerti98 | 11 |
| 10 | dark_wa... | 11 |
| 11 | axel15 | 8 |
| 12 | optimus | 7 |
| 13 | gerti98 | 4 |

| REGISTRAZIONE | |
|---------------|---------|
| Username | |
| | gerti98 |
| PLAY | |

*Fig 2: Interfaccia di Tank Battle:
Area Ranking ed Area Registrazione*

INSERIMENTO GIOCATORE E NUOVO GIOCO

Per avviare una partita è necessario inserire prima di tutto i dati del giocatore: deve essere inserito l'username del giocatore nel corrispondente campo di inserimento dell'area Registrazione.

Dopodichè deve essere premuto il bottone PLAY e la partita avrà inizio.

Una volta avviata la partita, l'utente può controllare il movimento del tankGiocatore di colore **giallo** tramite i tasti **W, A, S, D** per muoversi rispettivamente in alto, a sinistra, in basso e a destra. L'utente

può **sparare** un proiettile, tramite il **tasto J**, nella direzione di puntamento del tankGiocatore regolata dal movimento effettuato.

Se il proiettile del giocatore colpisce un tankNemico, il punteggio corrente viene incrementato di 1.

L'utente deve proteggere la propria **base**, posizionata in basso al tankGiocatore nella **fig. 3**, dall'assalto dei tankNemici i quali arriveranno dalle quattro corsie verticali dell'interfaccia di gioco, a partire dall'alto.

Sono presenti inoltre dei **blocchi** indistruttibili all'interno dell'area di gioco (di colore bianco), i quali limiteranno i movimenti del tankGiocatore, dei tankNemici e dei proiettili sparati. In particolare, il tankGiocatore potrà sparare solamente un proiettile alla volta, pertanto è richiesta particolare abilità nel coordinare i movimenti del tank e dello sparo.

E' possibile effettuare **GameOver** in tre modi diversi: se la **base** viene colpita da un **proiettile** (di qualsiasi tank) (**fig. 4**) oppure se la base viene **toccata** da un **tank nemico**, oppure se il tankGiocatore viene colpito da un **proiettile nemico** o **toccato** da un **tank nemico**.

In caso di **GameOver** il gioco si bloccherà rendendo impossibile all'utente interagire con il tankGiocatore come fatto in precedenza. Inoltre, verranno aggiornati i campi dell'area Ranking con i dati della partita appena effettuata.

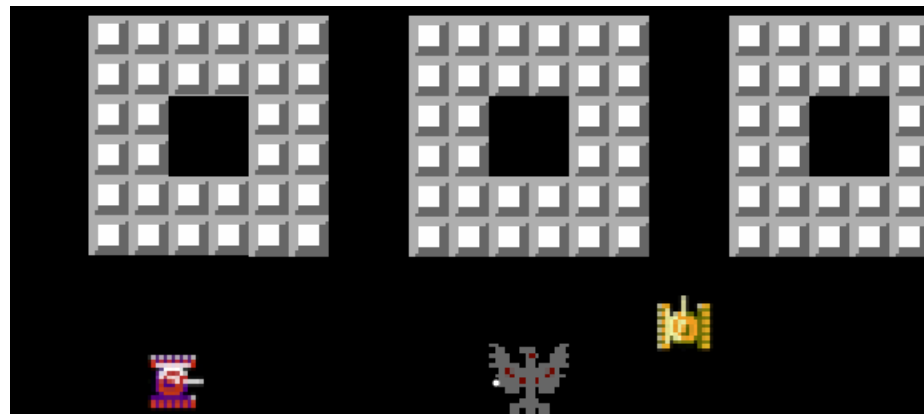


Fig. 4: GameOver per proiettile a base

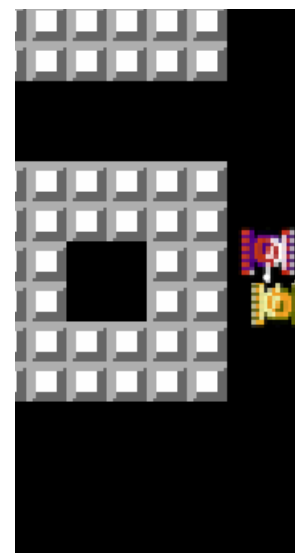


Fig. 5: GameOver per contatto tra tank nemico e tank giocatore

ARCHIVIO

Database: tankbattle. Tabella: Partita.

Vengono archiviati i seguenti dati al termine di ogni partita effettuata (vedi **fig.5**):

- idPartita: codice univoco di una partita
- username: username inserito da utente prima della registrazione
- score: punteggio relativo alla partita
- dataPartita: data della partita effettuata in formato yyyy-mm-dd

| idPartita | username | score | dataPartita |
|-----------|--------------|-------|-------------|
| 60 | dark_warrior | 11 | 2020-01-30 |
| 61 | axiel15 | 8 | 2020-01-30 |
| 62 | axiel15 | 3 | 2020-01-30 |
| 63 | gerti98 | 18 | 2020-01-30 |
| 64 | jack56 | 22 | 2020-01-30 |
| 65 | jack56 | 1 | 2020-01-30 |
| 66 | dark_warrior | 2 | 2020-01-30 |

Fig. 5: Archivio su Base di Dati. Tabella Partita

FILE DI CONFIGURAZIONE XML

E' possibile modificare il file di configurazione locale, chiamato *configurazione.xml*, esso contiene i seguenti campi (vedi **fig.6**):

- ipClient: contiene l'indirizzo IP del client (l'applicativo TankBattle)
- ipServer: contiene l'indirizzo IP del server di log
- portaServer: contiene il numero di porta del server di log
- username: username di accesso per il database
- password: password di accesso per il database
- numeroCampiTabella: numero dei campi da visualizzare nella tabella visuale
- etaMassimaInGiorni: età massima (in giorni) di una partita per poter essere visualizzata nella Tabella ranking

```
<?xml version="1.0" encoding="UTF-8"?>
<ParametriDiConfigurazione>
  <ipClient>127.0.0.1</ipClient>
  <ipServer>127.0.0.1</ipServer>
  <portaServer>8080</portaServer>
  <usernameDatabase>root</usernameDatabase>
  <passwordDatabase></passwordDatabase>
  <numeroCampiTabella>12</numeroCampiTabella>
  <etaMassimaInGiorni>30</etaMassimaInGiorni>
</ParametriDiConfigurazione>
```

Fig. 6: Contenuto file configurazione.xml

SERVER DI LOG

Vi è un server di log che registra, tramite stampa su terminale e salvataggio su file XML chiamato *eventoXML.xml* (vedi **fig. 7**) i seguenti eventi: **avvio** dell'applicativo, **terminazione** dell'applicativo e pressione del bottone **play**. Per ogni evento registrato viene inoltre memorizzato il **nomeApplicativo** che ha generato tale evento (TankBattle), l'indirizzo ip del Client (l'applicativo TankBattle), data ed ora di verifica dell'evento con relativo formato come attributo. Inoltre, per ogni evento il ServerLog stamperà su terminale i vari eventi registrati (oltre al salvataggio su file) come si può vedere in **Fig.7**.

```

<?xml version="1.0" encoding="UTF-8"?>
<EventoDiNavigazioneGUI>
  <nomeApplicazione>TankBattle</nomeApplicazione>
  <ipClient>127.0.0.1</ipClient>
  <etichettaEvento>AVVIO</etichettaEvento>
  <DataFormato formato="yyyy-MM-dd">
    <data>2020-01-31</data>
  </DataFormato>
  <OraFormato formato="HH:mm:ss">
    <ora>00:14:01</ora>
  </OraFormato>
</EventoDiNavigazioneGUI>
<!--#####-->

```

Fig. 6: Contenuto file eventoXML.xml

```

*****
Nome Applicazione: TankBattle
Ip Client: 127.0.0.1
Data (yyyy-MM-dd) : 2020-01-31
Ora (HH:mm:ss): 00:17:02
Evento: AVVIO
*****
Nome Applicazione: TankBattle
Ip Client: 127.0.0.1
Data (yyyy-MM-dd) : 2020-01-31
Ora (HH:mm:ss): 00:17:04
Evento: PLAY
*****
Nome Applicazione: TankBattle
Ip Client: 127.0.0.1
Data (yyyy-MM-dd) : 2020-01-31
Ora (HH:mm:ss): 00:17:52
Evento: TERMINE
*****
Nome Applicazione: TankBattle
Ip Client: 127.0.0.1
Data (yyyy-MM-dd) : 2020-01-31
Ora (HH:mm:ss): 00:18:11
Evento: TERMINE
*****

```

Fig. 7: Terminale del Server di Log