THE LAST HOPE

Indice:

1. Introduzione al gioco

Luglio 1990. L'Ucraina dichiara di voler diventare un Paese indipendente con la Dichiarazione di sovranità dell'Ucraina.
Febbraio 2014. Ucraina. Eserciti russi giungono in Crimea. I rapporti diplomatici tra i due Paesi sono ormai insanabili.
Marzo 2021. 100.000 soldati russi sono schierati al confine ucraino, per impedire che il Paese entri definitivamente a far parte della NATO.
Febbraio 2022. Il presidente Vladimir Putin annuncia l'inizio delle operazioni militari russe che porteranno all'invasione dell'Ucraina e alla definitiva rottura tra Oriente e Occidente.
La Russia ha intenzione di porre fine a questa storia durata fin troppo e utilizzare l'armata nucleare per vincere questa guerra.
Sarà il giocatore colui che impedirà che questo accada.

Per mancanza di tempo si è deciso di limitarsi all'ambiente ucraino, costruendo così una demo avanzata del gioco.
L'enigma finale del gioco è risolvibile solo eseguendo il server sulla stessa macchina.

2. Ambiente del gioco

2.1 Zone Ucraine

- **Zona1**: Garage;
- **Zona2**: Primo piano;
- **Zona3**: Piano terra;
- **Zona4**: Seminterrato.
- **Zona5**: Rifugio segreto.

2.2 Zone Russe

- **Zona 1**: Magazzino;
- **Zona 2**: Alloggio1;
- **Zona 3**: Alloggio2;
- **Zona 4**: Alloggio3;
- **Zona 5**: Alloggio4;
- **Zona 6**: Camera di sorveglianza;
- **Zona 7**: Stanza dell'ufficiale;

- **Zona 8**: Sala operativa.

3. Personaggi del gioco

- **Protagonista**: soldato ucraino dallo spirito combattente;
- **Alleato**: uomo ai livelli alti del comando ucraino, imprigionato dai russi perché a conoscenza di informazioni rischiose per loro;
- **Informatica**: uno stretto contatto informatico del precedente, già stata nella base militare e che conosce il suo sistema informatico per il quale ha progettato un virus in grado di disabilitarlo;
- **Nemico**: soldati militari russi in Ucraina;
- **Nemico:** soldati militari russi nella base militare.

4. Enigmi del gioco

- Illudere le guardie per salvare il comandante prigioniero facendo scattare l'allarme incendio. (**Sol**: appiccare incendio al primo piano);
- Accedere alla stanza segreta in Ucraina. (**Sol**: dopo aver salvato il comandante, scegliere di restare e sbloccare la porta a est del seminterrato rompendo il lucchetto con il martello);
- Sbloccare la password finale da codice. (**Sol**: decriptare la scritta esadecimale online mediante la chiave data e convertire il risultato) (**Sol password**: N0t_S0_E4sy_L0l).

Se non si riesce a risolvere l'enigma si potrà abbandonare il gioco uscendo dalla porta principale che si trova a ovest del Ground Floor.

- Illudere la guardia sorvegliante per recuperare la tessera gialla per entrare nella stanza dell'ufficiale. (**Sol**: portargli del caffè con del sonnifero dentro);
- Sbloccare una combinazione da quattro numeri per recuperare la tessera rossa dalla cassaforte. (**Sol**: mediante tre note con scritti i numeri (Es --x-) ed una con scritto un indizio sul quarto numero ('L'unione fa la forza') (somma dei precedenti con modulo 10)) (**Sol combinazione**: 1731);

5. Oggetti del gioco

5.1 Duraturi

- Oggetto Jammer per disabilitare telecamere;

- Tessera rossa per accedere alla sala operativa;
- Tessera gialla per accedere alla stanza dell'ufficiale;
- Chiavetta USB
- Pinze;
- Martello;

5.2 Non duraturi

- Tanica di benzina;
- Fiammiferi;
- Caffè;
- Pillole;
- Prima nota per aprire cassaforte;
- Seconda nota per aprire cassaforte;
- Terza nota per aprire cassaforte;
- Nota con indizio sul quarto numero per aprire cassaforte;

6. Argomenti trattati

- Database:
  Usato per memorizzare risorse testuali quali Cutscene (Intro-Outro) e Dialoghi per ogni personaggio a seconda del contesto in cui si trova quest'ultimo.
- File:
  Usato per salvare la partita in corso e potervi poi accedere successivamente. Salvataggio effettuato in maniera automatica dopo 5minuti, oppure manualmente premendo la combinazione ctrl-S.
- Thread:
  Usato per operazioni sul salvataggio automatico.
  Usato, inoltre, per operazioni di implementazione del suono.
- Grafica:
  Usata all'interno del gioco per rendere l'esperienza più accattivante.
- Espressioni Lambda:
  Usate per rendere più compatte e generiche le operazioni di ricerca e per implementare le tante interfacce funzionali che il pacchetto Swing offre.
- Client/Server
  Usato per implemementare il minigioco a cui si accede nella stanza segreta.

7. Organizzazione packages e relative classi

7.1 character: Package che contiene le informazioni riguardo le operazioni legate ai personaggi del gioco.

Classi presenti:

Character, rappresenta un personaggio.

Character_Type, contiene l'enum coi tipi possibili di un personaggio.

7.2 crypto: Package che contiene le informazioni riguardo le operazioni legate alla criptazione della password finale.

Classi presenti:

XorCrypter, il suo unico metodo crittografa una stringa con l'operazione di xor mediante una chiave di un byte.

7.3 database: Package che contiene le informazioni riguardo le operazioni legate al database utilizzato.

Classi presenti:

DbInitAdventure, crea (se non esiste) e inizializza un database.

H2Utility, implementa i metodi presenti in DbUtility, utilizzando il database H2.

Classi astratte:

DbUtility, contiene i metodi necessari per l'utilizzo di un database.

7.4 error: Package che contiene le informazioni riguardo le operazioni legate agli errori da restituire in output a seconda del contesto.

Classi presenti:

Fatal, contiene il messaggio da visualizzare in caso di errori fatali.

7.5 gameEnvironment: Package che contiene le informazioni riguardo le operazioni legate all'ambiente del gioco.

Classi presenti:

AdventureWorld, classe engine del gioco.

MapGame, rappresenta una mappa.

Position, necessaria per tener traccia di una posizione. (del personaggio e/o dell'oggetto)

RulesPosition, verifica che la combinazione stanza-mappa sia corretta.

WorldState, contiene l'enum coi tipi possibili di uno stato del gioco.

7.6 gameSound: Package che contiene le informazioni riguardo le operazioni legate al suono.

Classi presenti:
Sound, contiene i metodi necessari per l'implementazione del suono.
SoundThread, crea un thread il cui compito è riprodurre un suono.

7.7 graphic: Package che contiene le informazioni riguardo le operazioni legate alla grafica implementata.

Classi presenti:
ChoiceFrame, estende Jframe e crea una finestra standard adatta alle scelte.
DirectionalButton, permette di spostarsi fra stanze.
InitialFrame, estende Jframe e costituisce la parte grafica del gioco.
MyActionListeners, estende ActionListener ed è l'handler dei labels.
MyKeyListeners, estende KeyListener ed è un handler per gli eventi di input da tastiera.
MyChoiceFrame, estende ChoiceFrame.
CombineActionListener, DirectionalButtonActionListener, ItemsInBagActionListener, ItemsInRoomActionListener, MyActionListenerOnObject, MyActionListenerOnObjectDescribe, MyActionListeners reagiscono agli eventi di determinati componenti.
MyListCellRenderer, estende Jlabel e aggiunge un titolo alle combobox.
MyWindowAdapter, estende WindowAdapter e manipola gli eventi che subisce una finestra.

7.8 items: Package che contiene le informazioni riguardo le operazioni legate agli oggetti utilizzati nel gioco.

Classi presenti:
MapConteiner, rappresenta un contenitore di oggetti.
Item_Type, contiene l'enum coi tipi possibili di un oggetto.
Classi astratte:
Item, rappresenta un oggetto.
UsableItem, rappresenta un oggetto consumabile.
Generalizzazioni presenti:
RedCard, YellowCard Coffee, Fire, GasolineCan, Jammer, Hammer, Matches, NoteOne, NoteTwo, NoteThree, NoteFour, Pills, Pliers, SleepingCoffee, Usb **che estendono** Item.

7.9 net: Package che contiene le informazioni riguardo le operazioni legate alla connessione client-server.

Interfacce presenti:

Client, estende communicable e tratta le generiche operazioni di un client.
Communicable, implementata da ogni oggetto che è capace di scambiare oggetti con un altro.
Server, estende communicable e tratta le generiche operazioni di un server.
Classi presenti:
MyClient, implementa Client.
MyServer, implementa Server.

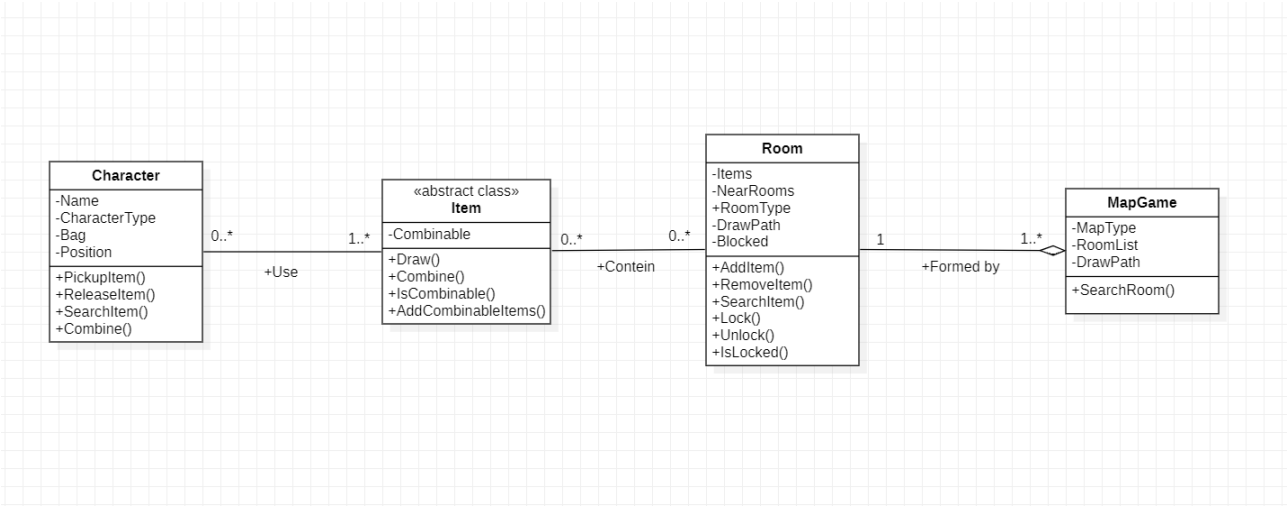
7.10 rooms: Package che contiene le informazioni riguardo le operazioni legate alle stanze presenti nelle mappe di gioco.

Classi presenti:
Room, rappresenta una stanza.
AdjacentRooms, rappresenta le stanze adiacenti ad una stanza.
Room_Type, contiene l'enum coi tipi possibili di una stanza.

7.11 savingGame: Package che contiene le informazioni riguardo le operazioni legate al salvataggio della partita in corso.

Classi presenti:
FileSaverTask, schedula il salvataggio di una certa struttura dati mediante un timer.
MyTimerTask, effettua un salvataggio dei dati con un intervallo di tempo regolare.
WrapperFileSerializable, wrapper per le operazioni su file.
Interfacce utilizzate:
Savageable, implementata da qualunque oggetto debba memorizzare i propri dati.

8. Diagramma delle classi (prospettiva concettuale)



9. Specifica algebrica (Struttura dati scelta: Character)

9.1 Sintattica:

Sorts: Boolean, String, Character, Position, CHARACTER_TYPE, Map<Item, Integer>, Item

Operation:

- Character (String, Position, CHARACTER_TYPE) -> Character
- Character (String, Position, CHARACTER_TYPE, Map<Item, Integer>) -> Character
- getName () -> String
- setName (String) -> Character
- getPosition () -> Position
- setPosition (Position) -> Character
- getType () -> CHARACTER_TYPE
- setType (CHARACTER_TYPE) -> Character
- getBag () -> Map<Item, Integer>

- pickupItem (Item) -> Character
- releaseItem (Item) -> Character
- searchItem (Predicate) -> Boolean
- combine (Item, Item) -> Item

9.2 Semantica:

Declare name: String; position: Position; type: CHARACTER_TYPE; voidBag: Map<Item, Integer>; item: Item; itemToCombine: Item; combinableItem: Item; bag: Map<Item, Integer>; x: Item; pred: Predicate<Map.Entry<Item,Integer>>, pred1:Predicate<ITEM_TYPE>.

- getName (Character (name, position, type)) -> name
- setName (Character (name, position, type), nameRec) -> Character (nameRec, position, type)
- getPosition (Character (name, position, type)) -> position
- setPosition (Character (name, position, type), positionRec) -> Character (name, positionRec, type)
- getType (Character (name, position, type)) -> type
- setType (Character (name, position, typeRec)) -> Character (name, position, typeRec)
- getBag (Character (name, position, type)) -> voidBag
- pickupItem (Character (name, position, type), item) -> Character (name, position, type, voidbag.add(item))
- searchItem (Character (name, position, type), pred) -> false

- getName (Character (name, position, type, bag)) -> name
- setName (Character (name, position, type, bag), nameRec) -> Character (nameRec, position, type, bag)
- getPosition (Character (name, position, type, bag)) -> position
- setPosition (Character (name, position, type, bag), positionRec) -> Character (name, positionRec, type, bag)

- getType (Character (name, position, type, bag)) -> type
- setType (Character (name, position, typeRec, bag)) -> Character (name, position, typeRec, bag)
- getBag (Character (name, position, type, bag)) -> bag
- pickupItem (Character (name, position, type, bag), item) -> Character (name, position, type, bag.add (item))
- releaseItem (Character (name, position, type, bag), item) -> Character (name, position, type, bag.remove (item))
- searchItem (Character (name, position, type, bag), pred) -> if
- combine (Character (name, position, type, bag), itemToCombine, combinableItem) -> if searchItem(pred) && itemToCombine.isCombinable(pred1) -> itemToCombine.getEntry(combinableItem).getValue ()

  else -> null

9.3 Restrittiva:

- releaseItem (Character (name, position, type), item) -> error
- combine (Character (name, position, type), itemToCombine, combinableItem) -> error

10. Bugs Noti

10.1 Nel ciclo di vita di un oggetto di tipo USB, viene registrato un fallimento nella gestione del caret relativo alla JTextArea. Un caret gestisce eventi legati al movimento del cursore nella JTextArea. Per motivi ancora ignoti, il caret non viene registrato dal manipolatore degli eventi, referenziando un puntatore nullo. Una possibile causa potrebbe essere la mancanza di una JScrollablePane a circondare la zona di testo editabile.