**DETECTIVE CONAD**

**Immagine che contiene testo

Descrizione generata automaticamenteMembri del TEAM BALLO**

* Michele Riviello, matricola 724802
* Vincenzo Parrulli, matricola 724933
* Vito Musco, matricola 724569

**Descrizione generale del gioco**

Questa avventura testuale, sviluppata in linguaggio Java, offre un’interfaccia utente con la quale interagire e supporta il salvataggio e caricamento della partita.

Il giocatore vestirà i panni di un detective e dovrà cercare di risolvere un caso assegnatogli, trovando tutte le prove necessarie a smascherare i colpevoli.

La struttura del gioco si basa sulla creazione e gestione di una mappa, composta da varie stanze, gestite come classi, che contengono gli oggetti specifici per la locazione che esse rappresentano assieme ai metodi che, a loro volta, rappresenteranno le interazioni possibili all’interno della data stanza.

All’avvio della partita viene data all’utente la possibilità di visionare o meno un filmato introduttivo, dopodiché dovrà attivare sull’interfaccia utente le varie interfacce grafiche che gli consentiranno di poter iniziare la partita.

**Mappa di gioco**

Per muoversi da una stanza all’altra, il giocatore dovrà cliccare sulla freccia corrispondente alla direzione in cui vuole andare, dal tastierino per i comandi di spostamento.

Gli oggetti di gioco saranno contenuti in una lista propria di ogni stanza



**Stile architetturale ECB**

Lo stile architetturale utilizzato per la suddivisione in pacchetti del progetto è l’ECB (Entity Control Boundary). Questo modello viene utilizzato in caso d'uso guidato di progettazione del software orientato agli oggetti, che struttura le classi componendo un software in base alle loro responsabilità nella realizzazione del caso d'uso. La Boundary class è responsabile della comunicazione con gli attori(utenti), e rappresenta un'interfaccia con l'ambiente esterno; Interfacce utente: finestre, dialoghi, menu Interfacce di comunicazione con altri sistemi Interfacce verso dispositivi esterni. La Control class è responsabile del flusso di controllo della user story (può essere modellato da un diagramma comportamentale) e contiene la logica dell'applicazione. L' Entity class è un'astrazione chiave del sistema ed è responsabile della persistenza dei dati incapsulati dall'astrazione.

**Salvataggio e Caricamento**

Durante la partita viene data la possibilità al giocatore di salvare i progressi ottenuti per poterli caricare successivamente.

Per poter salvare e caricare la partita su database abbiamo utilizzato dei metodi appositi.

**Utilizzo del DataBase**

Il database viene utilizzato per salvare i dati di una partita all'interno delle tabelle che vengono create.

Le tabelle contenute nel database sono: Azioni, Partita, Oggetti. Inoltre, ci permette di caricare i dati di una partita già salvata in precedenza per poterla continuare.

**Utilizzo delle Lambda Expression**

Le lambda expression sono state utilizzate per attribuire determinate funzionalità ad alcuni pulsanti dell'interfaccia grafica, consentendo all’utente di poter interagire con essi.

**Utilizzo dei socket**

I socket vengono utilizzati per stabilire una connessione client server. Dopo aver stabilito la connessione l'utente può accedere al menu di gioco, ed iniziare così una partita. Quando l’utente abbandonerà il gioco il server riporterà il tempo trascorso dall’inizio alla fine della connessione.

**Utilizzo dei Thread**

I thread vengono utilizzati nel server per conteggiare il tempo di gioco di un client in millisecondi.

**Utilizzo dei file**

I file sono stati utilizzati per prelevare alcuni dei dialoghi del gioco, per inserirli opportunamente quando richiesto in base alla situazione.

**Interfacce grafiche**

Nel gioco sono state inserite alcune interfacce grafiche con cui il giocatore può interagire:

* il collegamento cellulare, che contiene sotto forma di “applicazioni” tutte le azioni possibili in gioco;
* l’editor di testo, che mostra il testo dell’avventura;
* il tastierino di spostamento, che consente di spostarsi nelle direzioni indicate dalle frecce.

Questi sono stati inseriti per cercare di rendere l’avventura più coinvolgente possibile, in modo da far sentire il giocatore più stimolato visivamente.

**PRINCIPALI CLASSI**

**Database**

La classe Database permette di creare le tabelle necessarie per il salvataggio e il caricamento delle partite. Per fare ciò vengono utilizzati appositi metodi, che in caso di caricamento di una partita, effettuano controlli sulle azioni eseguite, sugli oggetti acquisiti dall’inizio della partita e sulla posizione del giocatore all’interno della mappa.

**Dialoghi**

La classe Dialoghi preleva alcuni dei dialoghi presenti nel gioco da un file chiamato dialoghi.txt, permettendone quindi l’utilizzo nelle varie situazioni che si vengono a creare nel corso di una partita.

**Eseguibile**

La classe Eseguibile contiene il main del programma.

**LogicaGioco**

La classe LogicaGioco gestisce la logica del programma: permette all’utente di eseguire le mosse durante una partita; assegna le opportune funzionalità a ciascun pulsante dell’interfaccia grafica; in base alle mosse compiute dal giocatore, permette di proseguire nella partita e, in definitiva di sbloccare un finale alternativo qualora la soluzione non fosse corretta.

**Interfaccia**

La classe Interfaccia gestisce la parte grafica del programma. All’avvio dell’applicazione l’interfaccia è composta da una schermata iniziale di menu: tramite la classe Interfaccia si ha la possibilità di scegliere tra tre voci (Indaga, Crediti, Esci). Nel caso in cui il giocatore dovesse scegliere la voce ‘Indaga’, viene visualizzata un’ulteriore finestra in cui viene chiesto se si vuole continuare una partita, se ce ne dovesse essere una, altrimenti di avviarne una nuova, inserendo il nome del salvataggio.

Successivamente viene visualizzata una schermata con tre pulsanti che consentono di aprire tre interfacce grafiche: una per l’editor di testo, una per il cellulare e una che controlli gli spostamenti.

Attraverso questa classe vengono gestiti inoltre tutti i pulsanti implementati nel programma.

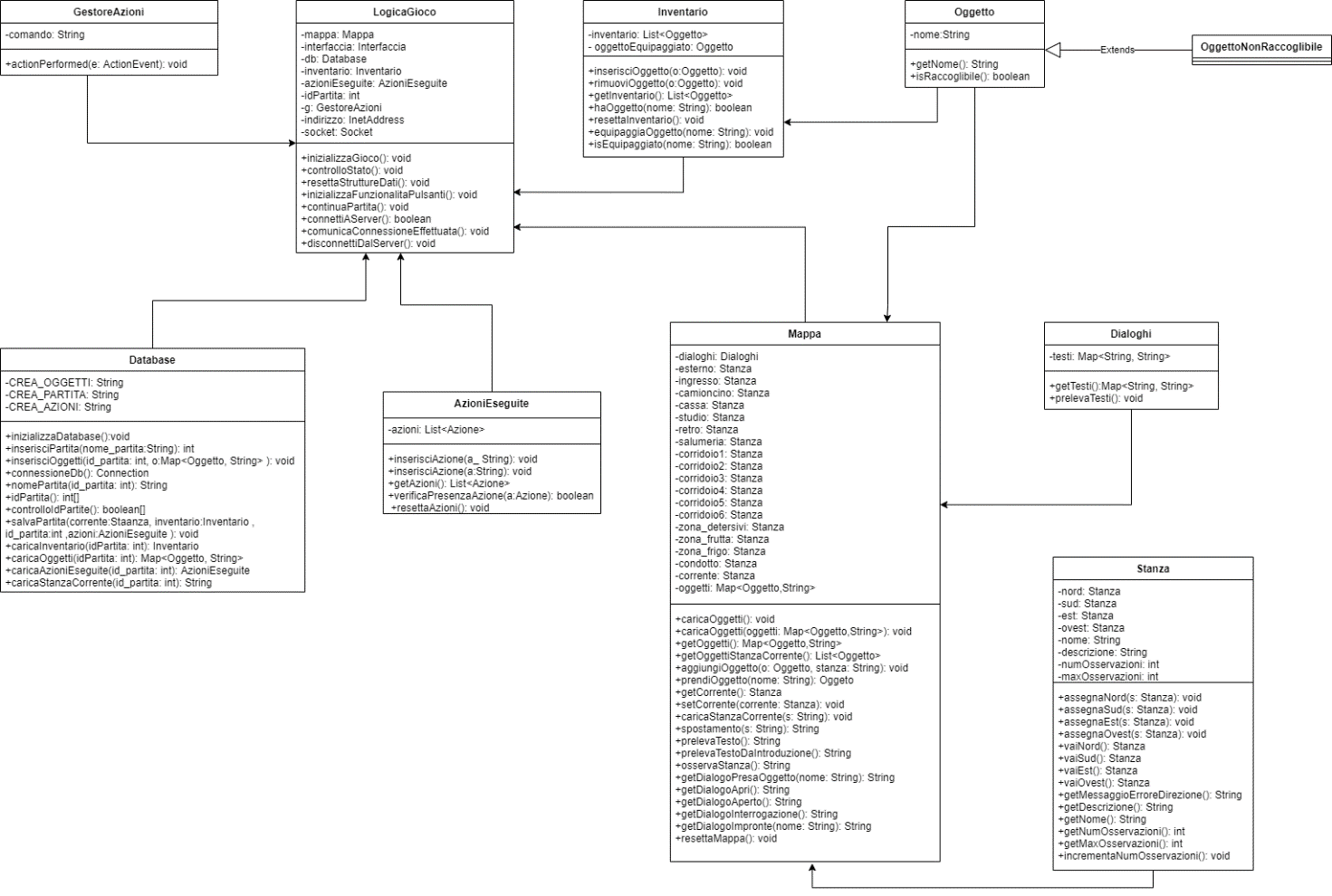
**Mappa**

La classe Mappa crea la mappa del gioco inserendo tutte le stanze nelle varie posizioni.

Permette di caricare gli oggetti all’interno di una determinata stanza, e di compiere le azioni possibili con essi.

Comunica con la classe Dialoghi per stampare a video determinati testi, riguardanti la stanza o gli oggetti che si interessa osservare.

**Diagramma delle classi:**

****

**SPECIFICA ALGEBRICA – LISTA**

**SPECIFICA SINTATTICA**  
OPERATORI:  
creaLista() --> lista  
inserisciOggetto(oggetto)--> lista  
rimuoviOggetto(oggetto) --> lista  
getInventario() --> lista  
haOggetto(nome) --> boolean  
resettaInventario() --> lista

**SPECIFICA SEMANTICA**  
Tipi:  
-Lista: insieme delle sequenze I=<o1, o2,...,on> con n>=0, di elementi di tipo  
oggetto dove l'i-esimo elemento ha valore o(i) e posizione pos(i).  
-boolean: insieme dei valori di verità  
OPERATORI:  
-creaLista()--> l';  
PRE:\  
POST:l'= <> (lista vuota);  
-inserisciOggetto(o)--> l';  
PRE:\  
POST:l'= <o1, ...,on,o> con n>=0;  
-rimuoviOggetto(o)--> l';  
PRE: l=<o1,...,on, o> (la lista deve contenere o);  
POST: l'= <o1,...,on> | per ogni i da 0 a n oi != o;  
-getInventario()--> l;  
PRE:\ POST: l= <o1,...,on> con n>=0;  
-haOggetto(n)--> b;  
PRE: l != <>;  
POST: b=true se l = <o1,....,on> per ogni i da 0 a n esiste almeno un oi = n;  
altrimenti b=false;  
-resettaInventario()--> l';  
PRE: l != <>;  
POST: l'=<>;

**COSTRUTTORI E OSSERVAZIONI**

|  |  |  |
| --- | --- | --- |
| **OSSERVAZIONI** | **COSTRUTTORE DI l’** | |
|  | **creaLista()** | **aggiungiOggetto(o)** |
| **rimuoviOggetto(o’)** | error | error |
| **getInventario()** | error | getInventario() |
| **haOggetto(n)** | false | If(o=n) then true else false |
| **resettaInventario()** | error | rimuoviOggetto(aggiungiOggetto(o)) |

**ORGANIZZAZIONE**

In seguito alla pandemia di Covid-19 siamo stati costretti a dover collaborare a distanza. Ciò comportava l’utilizzo di piattaforme come ad esempio Microsoft Teams, o in alcuni casi Discord, per poter comunicare in videochiamata e visionare passo per passo il lavoro in contemporanea.

Abbiamo anche utilizzato GitHub che si è dimostrato un mezzo molto utile per condividere il codice tra noi, riuscendo a mantenere gli aggiornamenti senza ricorrere ad un continuo scambio di file.

Gli incontri erano sempre prefissati, e il tutto era coordinato tramite un gruppo su Whatsapp, che ci ha permesso anche di rafforzare il nostro legame al di fuori degli incontri “lavorativi”.

**SOLUZIONE DEL GIOCO**

Il gioco inizia con il Detective Conad che si trova all’esterno del supermercato dove è avvenuto il crimine. Prima di poter entrare dovrà recuperare i guanti dal suo camioncino e indossarli. Dopodiché trovandosi all’ingresso del supermercato, per poter proseguire dovrà fotografare la piantina del supermercato.

Successivamente dovrà spostarsi nell’ufficio del direttore e osservare tutte le volte necessarie, affinché non trovi una grata, da aprire. Una volta preso il cacciavite dal suo camioncino lo dovrà utilizzare per aprire la grata, entrando nel condotto e ricorrendo all’utilizzo del flash al cellulare per poter proseguire all’interno di esso. Alla fine del condotto si ritroverà nella cella frigorifera del supermercato, dove osservando scoprirà il corpo del direttore e troverà la chiave per aprire la porta della salumeria che conduce alla cella frigorifera. Dopo aver recuperato la torca UV dal suo camioncino, dovrà, per mezzo di questa, rintracciare le impronte appartenenti ai colpevoli in: salumeria, ufficio del direttore e cella frigorifera.

Dopo aver trovato le impronte dovrà recarsi all’ingresso del supermercato, dove ci sono i dipendenti, per confrontare le impronte trovate con le loro.

Una volta equipaggiate le impronte dovrà premere “interroga” per confrontarle con quelle già in possesso.

Infine, dovrà formulare le sue ipotesi e cercare di incastrare i colpevoli, trovandosi dinanzi a due scenari:

1. Finale corretto: tutti e tre i dipendenti vengono incastrati;
2. Finale alternativo: non vengono incastrati tutti e tre i dipendenti.

Per poter terminare la partita, l’utente dovrà premere su “chiudi il caso”.