

THE ONE IN THE DARK

Autrice: Angela Mileti

GitHub: [Lola-bunny-owo/THE_ONE_IN_THE_DARK](https://github.com/Lola-bunny-owo/THE_ONE_IN_THE_DARK)

Sezione 1 - Spiegazione dell'avventura testuale realizzata

[Sezione 1.1 - Spiegazione]

L'avventura testuale realizzata si basa sulla scoperta di indizi e segreti che portano il giocatore a risolvere un grande mistero. Il giocatore s'identifica nel protagonista della storia (Spike), il quale scopre cose orribili sul suo passato. La storia ha uno sviluppo completo solo nell'ambiente "**casa**", ovvero l'unico ambiente nel quale (al momento) ha senso giocare.

Sono stati volutamente sviluppati anche altri ambienti, ai quali il giocatore può in ogni caso spostarsi ed eseguire esclusivamente comandi base di orientamento; la motivazione dietro questa scelta è la seguente: l'avventura testuale continuerà ad essere sviluppata per diventare più grande! :)

Precisamente, gli ambienti in cui il giocatore può spostarsi sono la casa, la spiaggia vicino alla casa, la biblioteca, la stazione di polizia, il diner e la foresta.

Descrizione degli ambienti

Casa: la casa del protagonista è formata da due piani, i quali contengono le seguenti stanze.

- Primo piano [4 stanze]: ingresso, salone, cucina, veranda (o secondo ingresso);
- Secondo piano [6 stanze]: corridoio largo, corridoio stretto, stanza da letto del protagonista, stanza da letto del custode, bagno, stanza segreta.

Una descrizione più ampia della casa del protagonista la si trova nella Sezione 1.2., essendo un ambiente completo.

Spiaggia: la spiaggia sulla quale si appoggia la casa si distende per diversi km. Anche se non contiene delle stanze vere e proprie, si individuano diverse "zone".

- Di fronte casa;
- Sentiero che porta alla foresta;
- Sentiero che porta al diner;
- Sentiero che porta alla biblioteca;
- Sentiero che porta alla stazione di polizia;

Questo ambiente si può definire "di transito", perché spostandosi tra le varie zone si ha accesso agli altri ambienti sopracitati.

Biblioteca: la biblioteca è il cuore della città perché è una miniera d'informazioni e conoscenza. Contiene libri e accoglie cittadini, ed è formata da diverse stanze.

- Ingresso
 - Sala lettura #1
 - Sala lettura #2
 - Info Point
 - Bagno
 - Sentiero che porta alla spiaggia
-

Stazione di polizia: la stazione di polizia è un ambiente molto utile, al quale dare una particolare attenzione.. È formato da diverse stanze.

- Ingresso
 - Ufficio #1
 - Ufficio #2
 - Area denunce
 - Stanza interrogatorio
 - Celle di custodia
 - Sentiero che porta alla spiaggia
-

Diner "Lola's Diner": il diner è un ambiente affollato e riscaldato dai caldi raggi del sole, dove clienti di qualsiasi genere sono i benvenuti. È inutile dire che è il luogo più visitato della città, nel quale circolano sempre nuove voci e nuove storie diverse. La bellissima locandiera Fanny, sogno di qualsiasi essere vivente, è la principale attrazione. Le stanze e zone individuabili sono le seguenti.

- Bancone
 - Bagno
 - Tabella città
 - Tavolini
 - Magazzino
 - Sentiero che porta alla spiaggia
-

Foresta: la foresta è un ambiente inquietante (come lo è, del resto, l'intera città), esplorata dalle poche anime coraggiose della cittadina. È scura, piena di cipressi calvi, faggi e sequoie. Le zone in cui ci si può spostare sono le seguenti.

- Ingresso
 - Zona spinosa
 - Sentiero che porta al diner
 - Sentiero che porta alla spiaggia
 - Sentiero sconosciuto
-

Comandi utili

All'interno dell'avventura testuale il giocatore, oltre a potersi spostare tra le stanze e gli ambienti, ha la possibilità di:

- **Ispezionare oggetti**, per conoscerne nome e descrizione

- **Raccogliere oggetti**, che può o meno inserire nell'inventario [l'inventario ha una capacità massima - 10 oggetti]
- **Scartare oggetti**, quando non gli sono più utili
- **Usare/aprire oggetti**, quando ha senso usarli/aprirli
- **Consultare la mappa**, per potersi orientare
- **Parlare con i personaggi**, per raccogliere informazioni o per ascoltare storie

Ulteriormente, il giocatore ha la possibilità di consultare un *menù di help*, all'interno del quale sono presenti sia i comandi sopracitati sia altri tipi di comandi utili.

[Sezione 1.2 - **Storia**]

È il 1995 e la storia si svolge nella misteriosa cittadina di *Awhell*, lontana dal resto del mondo. La cittadina è molto rinomata per alcune orribili vicende accadute nel 1975, e il giocatore (Spike) è il nuovo arrivato nella città, trasferitosi nella casa sulla spiaggia degli zii deceduti.

La casa a due piani ospita un signore barbuto dall'aspetto cupo, chiamato **il Grigio**, ovvero l'attuale custode della casa. Si gode la vecchiaia. Non molto si riesce a ricavare sulla storia di costui, giustificato dal fatto che la sua memoria non funziona molto bene.

La dimora è grande e le pareti di un bianco sporco, piena di librerie e scaffali con cianfrusaglie, tanti cavalletti da pittura e qualche manichino, macchine fotografiche, pennelli, strofinacci e tanti colori a tempera. Il protagonista nel corso della storia può notare diversi particolari che gli fanno scoprire che la casa in cui vive è stata la scena di crimini mai risolti.

La casa è infatti stato il luogo del suicidio dei suoi zii, Sig.ra **Felicia Thompson** e Sig. **Alan Thompson** nel 1975. In quell'anno però, il suicidio di due dei cittadini più affidabili e gentili non fu la cosa più spaventosa: diverse ragazzine scomparvero, e non furono mai ritrovati i loro corpi. Le ricerche sul colpevole o colpevoli non hanno mai portato a nulla, lasciando il caso irrisolto e facendo sprofondare nella depressione la cittadina.

Spike, spinto da curiosità e da pura voglia di conoscere la storia dei suoi zii, si ritrova a risolvere i misteri che avvolgono la città nella malinconia. Un amico si unisce a lui all'inizio della storia: un tenero cagnolino del colore del sole, Ein. Il suo dolcetto preferito sono delle gomme da masticare al caffè.

Sezione 2 - **Tecniche utilizzate**

Le tecniche utilizzate in questa avventura testuale sono diverse:

- **Ereditarietà tra classi**
- **I/O**
 - **Lettura da file di testo**
 - **Lettura di input da terminale**
 - **Lettura di file audio**
- **Thread**
- **Java Collections**
- **Java Swing**
- **Database (JDBC)**

- **API REST**

Ereditarietà tra classi

Nell'avventura testuale è molto presente il concetto di **ereditarietà tra classi**, in quanto sin da subito sono stati individuati "concetti condivisi" e "concetti specifici". Ad esempio, tutti gli ambienti dell'avventura hanno in comune qualche attributo e metodo, ma per ogni ambiente ci sono degli elementi non condivisi. Questo ha portato alla necessità di separare elementi condivisi e non.

I/O

Alla base dell'interazione giocatore-avventura c'è l'**I/O**, che è il cuore dello sviluppo; si può suddividere l'utilizzo in tre categorie:

1. Utilizzo dell'I/O per lettura da file di testo
2. Utilizzo dell'I/O per lettura di input da terminale
3. Utilizzo dell'I/O per lettura da file audio

La prima categoria permette al giocatore di leggere dei file di testo, come nel caso di utilizzo del comando [/help](#). A tal fine, ho utilizzato:

- le classi "Scanner", "File", "FileReader" e "BufferedReader".

La seconda categoria permette al giocatore di inserire degli input di vario genere nel terminale, e la macchina si adegua in base all'input. I comandi sono stati suddivisi in tipologia, e per questo non hanno un unico punto "di focus" - si possono individuare per ogni ambiente dei comandi di orientamento, e poi dei comandi generali che hanno lo stesso senso ovunque. A tal fine, ho utilizzato:

- le classi "Scanner" e "InputStreamReader".

La terza ed ultima categoria permette al giocatore di riprodurre della musica. Precisamente, una volta che il giocatore si è spostato nella sua stanza, può usare l'oggetto **Chitarra** e suonarla. A tal fine, ho utilizzato:

- le classi "File", "AudioInputStream" e "AudioSystem"
- l'interfaccia "Clip"

Thread

I Thread sono stati utilizzati in maniera molto semplice - dato che il giocatore nel corso dell'avventura testuale legge una quantità non indifferente di testi stampati sullo schermo del terminale, è risultato opportuno implementare un metodo di delay, il quale ad ogni chiamata permette di far "addormentare" il programma per un certo tempo (specificato in millisecondi). Così facendo, il giocatore ha modo di leggere il testo in tutta tranquillità.

Java Collections

Le Java Collections hanno avuto la loro parte nel programma, e la struttura dati (interfaccia) utilizzata è la **SortedSet** con una implementazione di tipo **TreeSet**; questa è servita per salvare il numero identificativo (Integer) delle stanze di ogni ambiente. Definita nella classe "Ambiente", ha trovato il suo scopo nelle sottoclassi attraverso alcuni metodi. Nello specifico, il metodo più in risalto è quello di aggiunta di un elemento al SortedSet (**add**).

Java Swing

Le Java Swing sono state utilizzate per rispondere al comando `/mappa` in modo adeguato. La loro implementazione permette di creare, come risposta al comando, una finestra contenente la mappa dell'ambiente in cui il giocatore è in quel momento. Le mappe sono state ideate e disegnate dalla sottoscritta.

Database (JDBC)

Grazie al protocollo **JDBC** è stato possibile creare delle connessioni a database in maniera molto semplice, dove il Database Engine utilizzato è **H2**. Ho così definito una classe "Database" (anche qui concetto di ereditarietà) e, dato che lo sviluppo dell'avventura è presente per l'ambiente casa, ho definito una classe "DatabaseCasa" nella quale sono state eseguite le varie operazioni di popolamento della tabella.

Ho anche definito una classe "DatabasePersonaggi", grazie alla quale l'utente ha la possibilità di interagire con i personaggi della storia secondo certi parametri. Nello specifico, questa classe potrà essere utilizzata per fornire al giocatore delle informazioni su tutti i personaggi conosciuti fino a quel momento (questa funzionalità sarà implementata più avanti).

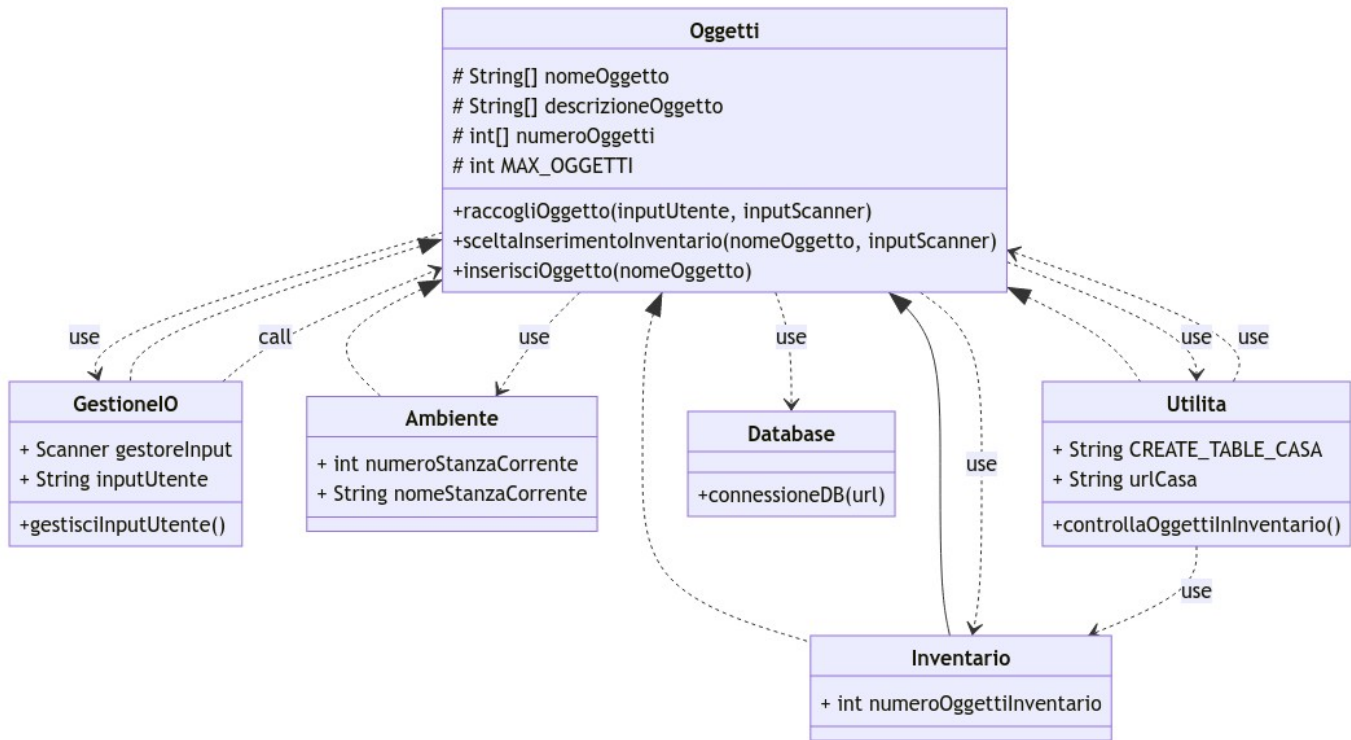
API REST

Ho utilizzato un piccolo escamotage per inserire all'interno dell'avventura le **API REST** - esaminando gli oggetti nella Casa, si può trovare l'oggetto **Orologio a pendolo** che se viene utilizzato restituisce l'ora. A tal fine, ho definito le classi:

- "RestOrologio": la classe nella quale si ha la vera e propria implementazione;
- "RestOrologioResponse": la classe che si occupa di definire gli attributi utili che sono restituiti dalla risposta di tipo Response.

Sezione 3 - Diagramma delle classi

Ho scelto di sottolineare l'importanza delle tre funzioni che sono utili al giocatore per poter raccogliere un oggetto (e quindi inserirlo nell'inventario). Le classi che partecipano allo sviluppo della funzionalità possono essere descritte attraverso il seguente diagramma delle classi.



Sezione 4 - Specifica algebrica

La specifica algebrica della struttura dati utilizzata nell'avventura testuale (SortedSet) è stata sviluppata nel modo seguente:

- **Specifica sintattica**

SORTS: SortedSet, int, element, boolean

OPERATIONS:

NewSortedSet() → SortedSet

Add(SortedSet, element) → SortedSet

Remove(SortedSet, element) → SortedSet

IsEmpty(SortedSet) → boolean

Size(SortedSet) → int

Contains(SortedSet, element) → boolean

- **Specifica semantica**

DECLARE: NumeroStanze: SortedSet, e: Element,

IsEmpty(NewSortedSet())= True

IsEmpty(Add(NumeroStanze, e))= False

Remove(NewSortedSet(), e')= error

```

Remove(Add(NumeroStanze, e), e')= NumeroStanze
Size(NewSortedSet())= i = 0
Size(Add(NumeroStanze, e), e')= Size(NumeroStanze+1)
Contains(NewSortedSet(), e')= error
Contains(Add(NumeroStanze, e), e')= if e = e' then true else false

```

- Specifica di restrizione

```

RESTRICTIONS:
Remove(NewSortedSet(), e')= error
Contains(NewSortedSet(), e')= error

```

Sezione 5 - Soluzione del gioco

Il giocatore inizia a giocare dalla stanza "Ingresso" dell'ambiente Casa. Ha molta libertà, può spostarsi ovunque egli desideri tranne che nella stanza segreta - in questa ci può entrare solo quando ha letto (attraverso il comando `/usa nomeOggetto`) due giornali presenti nella "Stanza del custode". La stanza del custode, come la stanza segreta, è al secondo piano della casa e **ci si può accedere solo ed esclusivamente** con un comando del tipo `/vai in stanza numeroStanza`.

Letti i due giornali, il giocatore può eseguire una sequenza di comandi tra i seguenti:

- vai avanti

oppure

- /usa Muro
- vai avanti

La differenza tra le due sequenze di comandi è che la prima fa spostare il giocatore direttamente nella stanza segreta, mentre la seconda dà un contesto al giocatore.

Una volta che il giocatore si è spostato nella stanza segreta, può scoprire diverse cose. Utilizzando gli oggetti **macchina fotografica n#1** e **macchina fotografica n#2**, il giocatore si ritrova in una situazione difficile che porta alla "fine" del primo capitolo dell'avventura testuale - in base agli oggetti che egli ha nell'inventario, lo sviluppo cambia forma.

Il risultato finale è lo stesso: la stanza segreta diventa inaccessibile. Da quel punto in poi, il giocatore può ancora spostarsi nelle altre stanze e ambienti come allo stesso modo può eseguire qualsiasi comando egli voglia; tuttavia, la trama del gioco si ferma nella stanza segreta - questo perché, come già specificato, l'avventura testuale è stata sviluppata al momento per il singolo ambiente "Casa" e più avanti continuerà ad essere sviluppata!

[Per aggiornamenti, basta mettere una stellina alla mia repository GitHub. :)]



Buon divertimento e spero che il mio progetto vi piaccia!