

ALMA MATER STUDIORUM · UNIVERSITÀ DI  
BOLOGNA

---

SCUOLA DI SCIENZE  
Corso di Laurea in Informatica Magistrale

# La Pianificazione per la Riabilitazione delle Funzioni Esecutive

Relatore:  
Chiar.mo Prof.  
Mauro Gaspari

Presentata da:  
Bartolomeo Lombardi

Sessione III  
Anno Accademico 2016/2017

*Alla mia famiglia, ai miei amici.  
A coloro che mi hanno sostenuto ogni volta  
che ho avuto bisogno di voltare pagina.*

# Introduzione

Nel corso dell'ultimo ventennio è cresciuto enormemente l'interesse verso le metodiche riabilitative in grado di rallentare ed alleviare il decadimento cognitivo di cui purtroppo è affetto, un numero sempre crescente di individui. In Italia la percentuale di pazienti che sviluppano deficit su differenti domini cognitivi tra cui, attenzione, memoria e funzioni esecutive è destinata a crescere con il progressivo invecchiamento della popolazione, dovuto principalmente ad un'alta incidenza di malattie degenerative come l'Alzheimer, la demenza, il Parkinson e la sclerosi multipla [7]. I soggetti affetti da deficit a carico delle funzioni esecutive ne risentono un forte impatto sulla vita quotidiana, riscontrando gravi conseguenze anche sulla qualità della vita [27]. Purtroppo, allo stato attuale non esistono cure farmacologiche in grado di annientare completamente queste patologie e la riabilitazione cognitiva è l'unica metodologia in grado di contenere e limitare il deterioramento cognitivo.

I pazienti effettuano sessioni di riabilitazione, solitamente condotte in ospedale, con l'aiuto di personale specializzato come logopedisti, psicologi ed altri operatori sanitari. Nella maggior parte dei casi ai pazienti vengono sottoposti esercizi da svolgere con carta e penna, di modo che l'operatore di turno possa seguire lo svolgimento dell'esercizio, intervenendo, se necessario, per consigliare o variare alcuni dettagli delle scelte del paziente. In questi anni è cresciuto l'interesse per le problematiche legate alle metodologie in uso per la riabilitazione cognitiva ed è in questa direzione che l'informatica si è mossa, intervenendo e proponendo strumenti automatizzati.

Molti esercizi cartacei sono stati, infatti, tradotti in software riabilitativi computerizzati con un significativo aumento della partecipazione da parte dei pazienti ed un grado di soddisfazione maggiore nel loro utilizzo [6, 28].

Considerata la mancanza, ad oggi, di software riabilitativi specifici per il trattamento della sclerosi multipla, il gruppo di ricerca del Prof. Mauro Gaspari del Dipartimento di Informatica - Scienza e Ingegneria in collaborazione con i neurologi e gli psicologi dell'ospedale Bellaria di Bologna guidati dal Dott. Sergio Stecchi, ha realizzato una piattaforma chiamata MS-rehab con il chiaro intento di fornire alle strutture sanitarie un programma completo e semplice per la riabilitazione cognitiva dei pazienti affetti da sclerosi multipla [11].

MS-rehab a differenza degli altri software riabilitativi, si configura come un'applicazione web di facile utilizzo che offre esercizi per ogni funzione cognitiva da riabilitare (attenzione, memoria e funzioni esecutive), permette la memorizzazione delle cartelle cliniche dei pazienti e la gestione delle valutazioni neuropsicologiche specifiche per la sclerosi multipla. La piattaforma presenta un'interfaccia minimale ed intuitiva, volta a stimolare nel paziente il desiderio di svolgere gli esercizi e continuare il processo riabilitativo anche da casa. Tra gli esercizi proposti nella versione attuale ve ne sono solo due sulle funzioni esecutive. L'obiettivo di questa tesi è stato quindi la realizzazione di un esercizio destinato alla riabilitazione delle funzioni esecutive partendo dal test "mappa dello zoo". Trattandosi di un test utilizzato nella fase di valutazione (fa parte infatti della batteria di test BADS, per la valutazione comportamentale dei deficit delle funzioni esecutive [31]), non è opportuno impiegarlo anche nella fase riabilitativa del paziente.

In questo elaborato si descrive quindi la progettazione e lo sviluppo di una versione alternativa, basata su pianificazione automatica, da utilizzare come esercizio riabilitativo mantenendo la struttura concettuale di base, creando un'interfaccia grafica minimale che permetta al paziente di compiere spostamenti essenziali per risolvere il problema. L'intelligenza artificiale definisce la pianificazione automatica come la generazione in maniera dinamica

di un piano di azioni per raggiungere dati obiettivi partendo da una condizione iniziale. Gli elementi necessari alla pianificazione sono la descrizione dell'obiettivo, la descrizione dello stato iniziale del mondo e la descrizione di tutte le operazioni possibili che possono essere attuate per modificare lo stato del mondo. Dati questi elementi, un pianificatore automatico effettua una ricerca nello spazio degli stati al fine di ricavare una sequenza di azioni che, se eseguite, provocano il raggiungimento dell'obiettivo [22, 26].

La possibilità di randomizzare gli obiettivi da raggiungere e la posizione degli stessi sulla mappa è stata realizzata, quindi, grazie all'interazione con il pianificatore che ha permesso di intuire se la configurazione generata permettesse soluzioni. La caratteristica principale dell'esercizio è il costante apporto di nuovi scenari in modo che sia virtualmente impossibile che si debba risolvere due volte lo stesso. Inoltre l'esercizio si adatta al paziente, poiché in base alla sua abilità il sistema sceglie la difficoltà di quest'ultimo, attraverso un modello dinamico che si prefigge di far riabilitare il paziente in base al suo livello soglia.

L'esercizio oggetto di questa tesi verrà sottoposto ad un gruppo di pazienti, affetti da sclerosi multipla, nella prossima sperimentazione multi-centro del sistema MS-rehab.

Nel capitolo 1 verrà descritto lo stato dell'arte della riabilitazione cognitiva, le patologie che influenzano i deficit cognitivi e il sistema MS-rehab. Nel capitolo 2 verrà descritto con maggior dettaglio il dominio cognitivo delle funzioni esecutive, le tecniche generiche per la riabilitazione e alcuni esercizi utilizzati in contesti clinici con un maggior interesse al test mappa dello zoo. Nel capitolo 3 vi sarà la definizione formale di planning, il concetto di pianificazione classica e il linguaggio PDDL. Nel capitolo 4 la progettazione di tutte le componenti per la realizzazione dell'esercizio, nello specifico l'analisi della conoscenza nel nuovo scenario implementato, il modello creato per la generazione del problema e la tecnica utilizzata per la realizzazione del comparto grafico. Il capitolo 5 viene descritto l'integrazione dell'esercizio nel sistema MS-rehab e la rivisitazione del modello matematico per il calcolo

delle performance. Infine vi è spazio alle conclusioni.

# Indice

<b>Introduzione</b>	<b>i</b>
<b>1 Background</b>	<b>1</b>
1.1 Stato dell'arte nella riabilitazione . . . . .	1
1.1.1 Ambiti applicativi . . . . .	1
1.1.2 Domini cognitivi . . . . .	5
1.1.3 Sistemi per la riabilitazione cognitiva . . . . .	7
1.1.4 Limiti dei sistemi esistenti . . . . .	10
1.2 La piattaforma MS-rehab . . . . .	10
1.2.1 La riabilitazione cognitiva nella SM . . . . .	12
1.2.2 Sezione per gli operatori . . . . .	14
1.2.3 Sezione per il paziente . . . . .	16
1.2.4 Esercizi . . . . .	16
1.2.5 Architettura . . . . .	16
<b>2 Riabilitazione delle funzioni esecutive</b>	<b>19</b>
2.1 Sistema cognitivo . . . . .	19
2.1.1 Funzione esecutiva: la pianificazione . . . . .	21
2.2 Aspetti della riabilitazione ecologica . . . . .	21
2.3 Esercizi cognitivi-funzionali . . . . .	22
2.4 Mappa dello zoo . . . . .	23
<b>3 Pianificazione</b>	<b>27</b>
3.1 Modello concettuale . . . . .	27

---

3.1.1	Sistema a transizione di stati . . . . .	28
3.1.2	Pianificatore . . . . .	29
3.1.3	Controllore . . . . .	30
3.2	Dominio indipendente . . . . .	30
3.3	Rappresentazione del problema . . . . .	32
3.3.1	Modellare con PDDL . . . . .	33
3.3.2	Modellare con PDDL: il mondo dei blocchi . . . . .	34
<b>4</b>	<b>Realizzazione</b>	<b>37</b>
4.1	Lo zoosafari . . . . .	37
4.2	Analisi della conoscenza nello zoosafari . . . . .	40
4.3	Utilizzo della pianificazione . . . . .	42
4.4	Formalizzazione in PDDL . . . . .	43
4.5	La generazione del problema . . . . .	45
4.6	La grafica . . . . .	47
4.7	Confronto con la mattina di impegni . . . . .	51
<b>5</b>	<b>Integrazione in MS-rehab</b>	<b>53</b>
5.1	Casi d'uso . . . . .	53
5.2	Diagramma di sequenza . . . . .	56
5.3	Struttura dell'integrazione . . . . .	57
5.4	Valutazione delle performance . . . . .	62
	<b>Conclusioni</b>	<b>65</b>
	<b>Bibliografia</b>	<b>69</b>



# Elenco delle figure

1.1	Assone sano e assone affetto da sclerosi multipla . . . . .	13
1.2	Architettura della piattaforma MS-rehab . . . . .	18
2.1	Test BADS: mappa dello zoo (dodici target) . . . . .	25
2.2	Test BADS: mappa dello zoo (sei target) . . . . .	26
3.1	Schema del problema: il mondo dei blocchi . . . . .	34
4.1	Struttura della mappa zoosafari . . . . .	38
4.2	Grafo rappresentante lo zoosafari . . . . .	46
4.3	Tileset utilizzata per il rendering grafico della mappa . . . . .	48
4.4	Rendering grafico della matrice HardMap. . . . .	50
4.5	Tileset utilizzata per il rendering grafico del visitatore. . . . .	51
5.1	Diagramma dei casi d'uso . . . . .	54
5.2	Diagramma di sequenza . . . . .	57
5.3	Processo di creazione scenario . . . . .	58
5.4	Diagramma delle classi del sistema . . . . .	60



# Elenco delle tabelle

1.1	Confronto dei software per la riabilitazione cognitiva. . . . .	11
5.1	Caso d'uso: visualizza descrizione esercizio. . . . .	54
5.2	Caso d'uso: visualizza esercizio. . . . .	55
5.3	Caso d'uso: esegue azioni. . . . .	55
5.4	Caso d'uso: interrompe esercizio. . . . .	56



# Capitolo 1

## Background

Data la cifra in crescita dei soggetti con deficit cognitivi, è maturata considerevolmente l'attenzione verso metodiche riabilitative in grado di rallentare e di alleviare il decadimento cognitivo, a causa dell'aumento del numero di individui che sono affetti da deficit cognitivi. Lo scopo degli interventi di riabilitazione è quello di migliorare ed instaurare un nuovo equilibrio personale, familiare e sociale del paziente [23]. L'interesse per le problematiche legate alle malattie che colpiscono le funzioni esecutive è andato al di là delle categorie professionali implicate originariamente, infatti l'informatica ha dato il suo contributo dovuto alla necessità di avere strumenti automatizzati e adeguati alla complessità dei deficit.

In questo capitolo verrà presentato lo stato dell'arte della riabilitazione cognitiva, le malattie che influenzano i deficit cognitivi, il sistema computerizzato per la riabilitazione cognitiva MS-rehab e le sue funzionalità, descrivendo nel dettaglio gli ambiti applicativi e le sue potenzialità.

### 1.1 Stato dell'arte nella riabilitazione

#### 1.1.1 Ambiti applicativi

Quando si parla di funzioni esecutive si intende l'insieme dei moduli funzionali della mente responsabili dei processi di pianificazione, controllo e coor-

dinazione a livello del sistema cognitivo. Si tratta delle funzioni che regolano l'attivazione e il dinamismo degli schemi coinvolti nei processi cognitivi.

Le funzioni esecutive sono compromesse da molteplici disturbi come ad esempio disturbo d'ansia, disturbo depressivo maggiore, disturbo dello spettro bipolare, disturbo da deficit dell'attenzione e iperattività; inoltre si riscontrano tali deficit anche nell'autismo e nella dislessia [13]. I *deficit esecutivi* sono stati rilevati nella Sindrome da deficit di attenzione e iperattività (ADHD), nella schizofrenia e nel disturbo della condotta. Oltre a questi disturbi i deficit esecutivi sono presenti in molte malattie degenerative, ad esempio l'Alzheimer, demenza vascolare, Parkinson, sclerosi multipla, ecc. La presenza di deficit cognitivi nei soggetti affetti da sclerosi multipla è un fatto noto e negli ultimi 20-30 anni si è iniziato uno studio sistematico dei disturbi cognitivi che ne derivano [5].

Tutte le patologie appena citate non sono facilmente inquadrabili dal punto di vista dei disturbi cognitivi che determinano, pertanto non esistono trattamenti specifici per le singole malattie, e risulta sempre necessario uno studio preventivo e una pianificazione individuale del processo di riabilitazione per ciascun paziente. L'efficacia di un intervento riabilitativo prescinde da una accurata valutazione del danno attraverso l'esecuzione di test neuropsicologici validati che danno indicazioni indispensabili sulla gravità dei deficit cognitivi e sulle aree cognitive compromesse del paziente. La fase di valutazione è una fase importante poiché indirizza le scelte terapeutiche nonché riabilitative da parte degli operatori.

### **Fase di valutazione cognitiva**

La fase di valutazione cognitiva comprende una valutazione dei principali domini cognitivi come ad esempio la memoria, l'attenzione e le funzioni esecutive, attraverso appositi sistemi validati. Questa valutazione si prefigura come indispensabile e necessaria alla definizione delle scelte terapeutiche sottendenti l'intervento riabilitativo. Solitamente, vengono impiegate batterie di test, ovvero insiemi di test neuropsicologici diversi fra loro, raggruppati e

somministrati in sequenza, con il chiaro scopo di ricoprire la valutazione della maggior parte delle dimensioni del comportamento cognitivo [21]. Sono disponibili differenti test per ciascuna delle funzioni cognitive, che permettono di ottenere un profilo generale dello stato cognitivo del paziente.

### **Fase di riabilitazione cognitiva**

La riabilitazione cognitiva è un efficace trattamento non farmacologico che comprende un ampio ventaglio prestazionale a seconda della natura della compromissione e del caso clinico. Consiste quindi nell'apprendimento di strategie compensatorie e nello sfruttamento delle abilità residue, al fine di contrastare il decorso degenerativo della malattia in atto. Allo stato dell'arte esistono diverse aree di intervento, tra cui:

- Programmi di stimolazione cognitiva per persone affette da demenza tipo Alzheimer;
- Programmi di stimolazione cognitiva per persone affette da malattia di Parkinson;
- Programmi di riabilitazione cognitiva per persone affette da sclerosi multipla;
- Programmi di riabilitazione cognitiva per persone con danno cerebrale acquisito;
- Programmi di riabilitazione cognitiva per persone colpite da ictus.

La riabilitazione cognitiva è l'unica metodologia in grado di contrastare il deterioramento cognitivo, seppur la maggior parte delle patologie precedentemente elencate sono degenerative.

L'obiettivo degli interventi riabilitativi è mantenere il cervello attivo e stimolare le capacità cognitive superiori per contrastare il declino cognitivo del paziente durante gli anni. La riabilitazione cognitiva è efficace perché si basa sul fenomeno della plasticità neuronale all'interno del nostro cervello, grazie

al quale il sistema nervoso è in grado di modificare la propria struttura e le proprie funzionalità a seconda dell'attività dei propri neuroni, come reazione alla diversità dell'ambiente [10]. Aumentando le dimensioni di alcune regioni del cervello in seguito al loro utilizzo ripetuto, le cellule neuronali hanno maggiore attività e di conseguenza formano più sinapsi tra di loro durante l'apprendimento; questo permette al cervello di ristabilirsi da disturbi e lesioni, contrastando e rallentando gli effetti di alterazioni strutturali prodotte da patologie come la sclerosi multipla, il Parkinson, il morbo di Alzheimer, ecc. Un altro fattore fondamentale è la riserva cognitiva, questa descrive la capacità del cervello di resistere al danno cerebrale causato dalla patologia in atto. Potrebbe essere definita come la capacità di ottimizzare o massimizzare le prestazioni attraverso il reclutamento di reti cerebrali non corrotte e strategie cognitive alternative. La riserva cognitiva è comunemente usata in letteratura per riferirsi sia alla riserva cerebrale, che identifica la dimensione del cervello sia a quella cognitiva che è la capacità di compensare attivamente e di rendere più efficace ed efficiente l'uso delle reti cerebrali. Questa "riserva" consente al cervello di mantenere la funzione, riorganizzandosi per compensare l'atrofia cerebrale e la perdita di cellule nervose e fibre nervose. Il cervello lo fa reindirizzando i percorsi di comunicazione per evitare aree danneggiate. Può anche adattare le aree non danneggiate ad assumere funzioni che una volta venivano svolte da aree ora danneggiate dalla malattia. Da diversi anni, la ricerca ha evidenziato come la riserva cognitiva, che viene costruita da ciascuno di noi negli anni con l'istruzione, l'abitudine alla lettura, il tipo di professione e le attività anche fisiche e sportive svolte nel tempo libero, abbia un ruolo nel contrastare l'insorgenza o il peggioramento di problematiche cognitive nei pazienti. A parità di altre condizioni, le persone con sclerosi multipla che hanno un'elevata riserva cognitiva perdono meno funzioni cognitive rispetto a quelle con minore riserva cognitiva, per la stessa quantità di lesioni cerebrali e atrofie [29].



### 1.1.2 Domini cognitivi

Prima di presentare lo stato dell'arte dei sistemi riabilitativi in contesto clinico è necessario introdurre quali sono le entità del dominio cognitivo comuni possibili da riabilitare.

#### **Attenzione**

L'attenzione è il processo cognitivo che consente alla mente di focalizzarsi su un determinato stimolo, annullando tutti quelli che potrebbero verificarsi nell'ambiente circostante; è di primaria rilevanza in quanto permette ai pazienti sani di sfruttare in modo corretto le risorse cognitive per analizzare ed elaborare gli stimoli percepiti attraverso i sensi. L'attenzione è di fondamentale importanza per l'apprendimento, in quanto un deficit in questa area rende difficile elaborare informazioni, eseguire istruzioni, memorizzare e concentrarsi [32].

Secondo il modello proposto da Sohlberg e Mateer [28], l'attenzione è suddivisa seguendo lo schema sottostante:

**Attenzione focalizzata** è la capacità di rispondere ad uno stimolo di tipo visivo, uditivo o tattile;

**Attenzione sostenuta** è la capacità di mantenere un buon livello di concentrazione agli stimoli mentre si sta svolgendo un compito differente o comunque continuativo nel tempo;

**Attenzione selettiva** è la capacità di sopprimere volontariamente gli stimoli in entrata (distrattori) per mantenere la concentrazione su un determinato target;

**Attenzione alternata** permette di direzionare il focus attentivo allo svolgimento di compiti che richiedono risorse cognitive differenti come sentire un suono o guardare immagini;

**Attenzione divisa** è la capacità di prestare attenzione a più stimoli contemporaneamente.

## Memoria

La letteratura scientifica descrive la memoria come la funzione cognitiva che codifica, immagazzina e recupera le informazioni. La fase di codifica si riferisce allo step iniziale in cui l'informazione si trasforma in un codice che la memoria riconosce (viene influenzato da fattori emotivi - cognitivi - motivazionali del soggetto), viene poi consolidato e immagazzinato nel tempo; invece la fase di recupero consente il prelevamento delle informazioni conservate in un tempo successivo.

Secondo il famoso modello proposto da Atkinson e Shiffrin [2], il processo di memorizzazione è in funzione alla durata di permanenza dell'informazione, si parla quindi di:

**Magazzino sensoriale** in grado di catturare l'informazione in entrata proveniente dai sensi e trattenerla per brevissimo tempo;

**Memoria a breve termine (MBT)** predisposta a immagazzinare un numero limitato di informazioni dal magazzino sensoriale (un adulto può memorizzare dalle due alle cinque informazioni) per un periodo di tempo molto breve, circa una decina di secondi;

**Memoria a lungo termine (MLT)** è l'area in cui sono conservate tutte le esperienze e le conoscenze acquisite nel corso della vita, infatti ha capacità di immagazzinamento illimitato e per lunghi periodi.

Nel loro modello Atkinson e Shiffrin fanno riferimento a componenti strutturali (ovvero i tre magazzini di memoria) e a processi di controllo, che si riferiscono a tutte le operazioni che vengono svolte consapevolmente al fine di immagazzinare produttivamente un'informazione, favorendone il passaggio dalla MBT alla MLT.

Alla fine degli anni '80 il panorama scientifico relativo agli studi sui processi di memoria si è ulteriormente arricchito di nuovi contributi: sono gli anni in cui Baddeley e Hitch introducono il concetto di memoria di lavoro

(MDL), descritta come un sistema in cui vengono mantenute temporaneamente delle informazioni mentre si svolgono altri compiti. Baddely propone un modello, alternativo a quello dei magazzini a breve termine formulato da Atkinson e Shiffrin, di memoria a due dimensioni definite come processo articolatorio, coinvolto nell'immagazzinamento e nell'elaborazione del materiale verbale, e magazzino visuospatiale, con analoga funzione rispetto al materiale visuospatiale [3]. Queste due componenti sono supervisionate da ciò che viene definita centrale esecutiva, che sovrintende ai compiti di ragionamento e decisione.

### **Funzioni esecutive**

Le funzioni esecutive sono un insieme di processi cognitivi superiori che permettono di risolvere correttamente dei problemi, pianificare, iniziare e portare a termine comportamenti diretti a uno scopo attraverso azioni coordinate e strategiche, utilizzare strategie flessibili, mantenere informazioni nella memoria di lavoro, porre attenzione, controllare e filtrare dati rilevanti e svolgere più compiti in parallelo. Siccome sarà oggetto di questa tesi, una analisi più approfondita delle funzioni esecutive verrà presentata nel capitolo 2.

#### **1.1.3 Sistemi per la riabilitazione cognitiva**

La riabilitazione tradizionale è uno degli approcci metodologici alla riabilitazione cognitiva, attualmente il più diffuso in contesto clinico, viene condotto in ospedale con l'aiuto di personale specializzato come logopedisti e psicologi; ai pazienti vengono sottoposti esercizi in forma cartacea e l'operatore di turno segue lo svolgimento dell'esercizio occasionalmente, intervenendo per consigliare o variare alcuni dettagli sulle scelte del paziente. Purtroppo questa tecnica riabilitativa oltre ad avere un vantaggio in quanto offre agli operatori un massimo controllo sul paziente e sullo stato riabilitativo dello stesso, presenta alcune difficoltà; con tale tecnica risulta difficile realizzare interventi continuativi, infatti possono passare settimane o addirittura mesi

tra una seduta e l'altra, inoltre il costo di questo tipo di riabilitazione è decisamente elevato poiché richiede la presenza stabile di personale specialistico dedicato.

Nell'ultimo ventennio sono state percorse tecniche riabilitative più moderne, con l'utilizzo di software riabilitativi computerizzati, il paziente in ambito ospedaliero effettua la terapia attraverso un computer, eventualmente con alcune periferiche dedicate. Questa tecnica è nata per contrastare le problematiche precedentemente descritte riguardo la riabilitazione tradizionale, in questo specifico caso la sessione riabilitativa risulta più interattiva, videosorvegliata sempre da un operatore, ma l'efficacia principale di questi sistemi è la possibilità di adattare automaticamente gli esercizi in base alla storia degli esercizi precedentemente svolti dal paziente. Secondo diversi studi presenti in letteratura si è valutato positivamente l'approccio computerizzato alla riabilitazione cognitiva, poiché risulta un valido strumento a disposizione degli operatori e può seriamente aiutare i pazienti nel processo riabilitativo [6].

I programmi per la riabilitazione cognitiva forniscono un ampio ventaglio di esercizi focalizzati sulle singole entità del dominio cognitivo (attenzione, memoria, funzioni esecutive) offrendo evidenti vantaggi rispetto la riabilitazione tradizionale, come ad esempio la possibilità di monitorare e memorizzare il percorso riabilitativo contenente i risultati dei pazienti attraverso tabelle e grafici, l'incremento della difficoltà automatico e ulteriori features utili nella riabilitazione. Oltre che a programmi specifici sono stati sperimentati anche giochi tradizionali ad esempio, un team di ricercatori Italiani ha testato alcuni giochi per il computer tagliati specificatamente per favorire la sessione di riabilitazione cognitiva [25].

In letteratura esistono alcune sperimentazioni circa la riabilitazione attraverso dispositivi mobili, volendo citare un gruppo di ricerca di Genova che ha sviluppato un applicazione mobile (COGNI-TRAcK) per la riabilitazione neurocognitiva ottenendo una partecipazione del circa 84% dei pazienti coinvolti e l'81% tra loro è risultato motivato a continuare ad usare l'ap-

plicazione anche dopo la fine della sperimentazione [30]. Questi dati sono incoraggianti riguardo l'utilizzo di applicazioni mobili per la riabilitazione cognitiva, tentando di aprire una strada verso un nuovo stile riabilitativo, più vicino al paziente che possa inserirsi nella sua quotidianità avendo un maggiore impatto nella vita di tutti i giorni.

Di seguito verranno elencati alcuni dei software per la riabilitazione cognitiva più usati in Italia [11]:

**Brainer** è un software basato sul web per la riabilitazione cognitiva sviluppato da Brainer srl (<https://www.brainer.it>). Brainer è stato utilizzato in diversi studi clinici, soprattutto su pazienti affetti da schizofrenia, ma potrebbe essere utilizzato anche da pazienti affetti da sclerosi multipla.

**Rehacom** è un software general purpose avanzato per la riabilitazione cognitiva, sfruttato specificamente per la riabilitazione della sclerosi multipla, distribuito da Hasomed (<https://www.rehacom.com>). Rehacom funziona su computer e richiede uno schermo di 19 pollici, inoltre può essere integrato con una tastiera specifica. Rehacom fornisce molti esercizi basati su un contesto realistico tridimensionale supportando diverse lingue.

**Erica** è un software general purpose per la riabilitazione cognitiva sviluppato da Giunti O.S. (<http://www.ERICA.giuntios.it>). Funziona su qualsiasi computer ed è usato per trattare anche pazienti affetti da sclerosi multipla.

**CogniPlus** è uno strumento nato in ambito scientifico per il trattamento dei disturbi delle funzioni cognitive sviluppato da SCHUHFRIED (<http://schuhfried.com>). CogniPlus è disponibile in 16 lingue, ed è stato utilizzato in molti studi per trattare la sclerosi multipla.

### 1.1.4 Limiti dei sistemi esistenti

Tutti i sistemi sopra citati sono general purpose e sono utilizzati per tutti gli ambiti applicativi descritti precedentemente. Nonostante Brainer offra più esercizi rispetto agli altri sistemi, questi non si basano su un contesto realistico e inoltre non implementa un adattamento della difficoltà degli stessi durante la sessione di riabilitazione. L'operatore può variare manualmente la difficoltà di soli tre gradi (facile, medio, difficile), una caratteristica che rende il software poco fruibile da pazienti autonomi. Riguardo ad Erica invece un limite è che non supporta l'allenamento a casa, quindi tutte le sessioni riabilitative devono avvenire in ospedale. Software come Brainer sfruttano un approccio basato sul web mentre Rehacom e CogniPlus forniscono il prodotto in chiavette USB per la riabilitazione a domicilio.

In Tabella 1.1 viene presentato un confronto tra i principali software utilizzati per la riabilitazione cognitiva [11], alcuni tra questi mostrano l'assenza di funzioni per la riabilitazione di gruppo, training domestico, controllo remoto e la presenza di esercizi non ambientati in un contesto realistico; capiremo meglio il valore aggiunto di fornire esercizi ambientati in un contesto realistico nel capitolo seguente. Tutte le mancanze appena citate sono un freno per la diffusione di queste piattaforme.

## 1.2 La piattaforma MS-rehab

MS-rehab è il frutto di uno studio sorto dalla necessità di sviluppare un software riabilitativo specifico per il trattamento della sclerosi multipla che potesse fornire tutte le features presenti nei sistemi di riabilitazione cognitiva allo stato dell'arte (tabella 1.1) e che offrisse un pacchetto riabilitativo completo multi piattaforma per i pazienti e per gli operatori. Una collaborazione nata tra i neurologi e gli psicologi dell'ospedale Bellaria di Bologna guidati dal Dott. Sergio Stecchi e il gruppo di ricerca del Prof. Mauro Gaspari del Dipartimento di Informatica - Scienza e Ingegneria, i quali hanno individuato quelli che erano i limiti dei sistemi esistenti, progettando e realizzando una

	Brainer	Rehacom	Erica	CogniPlus
Numero di esercizi	77	20+	35	15
Contesto realistico	no	si	no	si
Configurazione rapida	si	no	no	no
Esercitazioni a casa	si	si	no	si
Adattamento difficoltà	no	si	si	si
Supporto gruppi	no	no	no	no
Controllo remoto	no	no	no	no
Supporto al tablet	si	no	no	no
Piattaforma web	si	no	no	no

Tabella 1.1: Confronto dei software per la riabilitazione cognitiva.

nuova piattaforma per la riabilitazione cognitiva che vuole essere una nuova frontiera per il processo riabilitativo domestico ed ospedaliero. MS-rehab a differenza degli altri software riabilitativi si configura come una applicazione web di facile utilizzo che offre esercizi per ogni funzione cognitiva da riabilitare, permette la memorizzazione delle cartelle cliniche dei pazienti e la gestione delle valutazioni neuropsicologiche specifiche per la sclerosi multipla. La piattaforma presenta un'interfaccia minimale e intuitiva, volta a stimolare al paziente il desiderio di svolgere gli esercizi e continuare il processo riabilitativo anche da casa. Ogni esercizio contenuto nella piattaforma viene presentato dalla descrizione dell'obiettivo ed eventuali esempi, per assicurarsi che al paziente sia evidente il compito che deve svolgere per portarlo a termine. Inoltre al paziente non viene mai richiesto di risolvere lo stesso esercizio per più di una volta, poiché in base alla sua abilità il sistema adatta la difficoltà dell'esercizio che gli è stato assegnato, attraverso un modello dinamico che si prefigge di far riabilitare il paziente in base al suo livello soglia; con questo modo di operare il paziente trova sempre impegnativa la sfida riabilitativa stimolando così il recupero neuropsicologico.

### 1.2.1 La riabilitazione cognitiva nella SM

Prima di continuare la trattazione, occorre introdurre cosa si intende per sclerosi multipla (SM). Quest'ultima può essere descritta come un disturbo degenerativo demielinizzante che colpisce il sistema nervoso centrale. Alla base della SM dunque vi è un processo di demielinizzazione che determina danni o perdita della mielina, una sostanza lipidica che riveste le fibre nervose e che permette la trasmissione degli impulsi nervosi; in Figura 1.1 sono mostrate le lesioni (scientificamente chiamate "placche") in cui assumono caratteristiche simili a cicatrici, da cui deriva il termine sclerosi. Le placche che vengono a formarsi rallentano o interrompono la propagazione del segnale negli assoni. Se in un individuo sano la conduzione degli impulsi nervosi è di 100 m/s, un individuo affetto da sclerosi multipla può vedere ridotta la velocità di trasmissione di più del 90% con una velocità di propagazione di 5 m/s.

Nel mondo si contano dai 2,5 ai 3 milioni di persone con SM, di cui 600.000 in Europa e circa 114.000 in Italia (circa un italiano su 1000). La Sardegna è la regione più colpita con una media di 1 abitante su 700 a contrarre la malattia. La malattia può esordire a qualsiasi età della vita, ma è diagnosticata per lo più tra i 20 e i 40 anni, si sono verificati anche casi di bambini affetti da questa patologia [8], inoltre le donne risultano colpite in numero doppio rispetto agli uomini. Le cause scatenanti sono ad oggi ancora sconosciute, anche se è probabile che vi sia una combinazione di fattori genetici, ambientali e infettivi.

Gli effetti della malattia possono essere vari e numerosi, tra i principali possiamo ricordare, ad esempio disturbi dell'attività e della coordinazione motoria, parestesie, disturbi della vista, disturbi della parola, disturbi vescicali e/o intestinali, disturbi dell'equilibrio, disturbi sessuali, dolore, depressione, disturbi cognitivi. Il deterioramento cognitivo subito dai malati di SM può colpire capacità come l'attenzione, la memoria, le funzioni esecutive e tutto ciò ha impatto, ad esempio, sull'apprendimento di nuove capacità o concetti, ma anche su azioni giornaliere non trascurabili. I pazienti affetti da



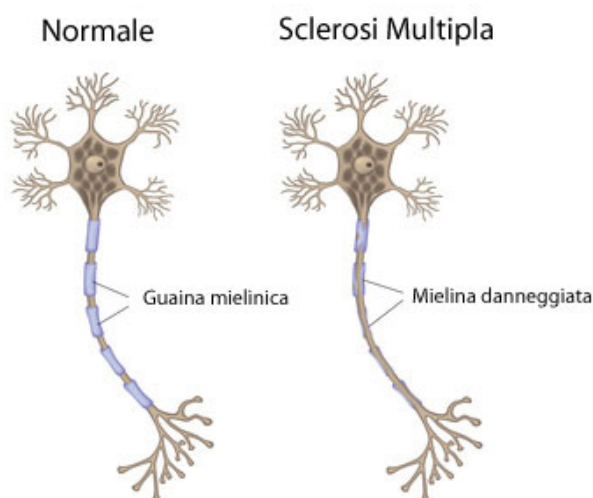


Figura 1.1: A sinistra un esempio di assone sano, mentre a destra quello affetto da sclerosi multipla.

questa malattia, infatti, presentano (con una percentuale che si aggira tra il 43% ed il 70%) deficit di natura cognitiva [11].

La riabilitazione cognitiva in tale contesto risulta un iter complesso in quanto il paziente deve attraversare una fase di valutazione in tutte le sue specificità per costruire un programma terapeutico efficace, che comprenda un'attenta valutazione diagnostica, la creazione di piani di cura individuali, trattamenti farmacologici-riabilitativi e visite follow-up. Durante la fase valutativa, uno psicologo eseguirà a sua volta una valutazione multidimensionale utilizzando vari strumenti e l'output risultante di tale fase determinerà il profilo cognitivo del paziente in base ai deficit identificati durante la valutazione. Dal profilo risultante, un'équipe di medici stabilisce i veri e propri obiettivi legati all'iter di riabilitazione; tutto ciò ha come fine ultimo il maggiore adattamento, sia motivazionale che psicologico, al trattamento da parte della persona affetta da SM e della famiglia.

L'approccio riabilitativo tipico consiste quindi nella stimolazione cognitiva effettuata sia in ospedale che nella propria abitazione, prediligendo tutti

quegli esercizi che forzano il paziente a produrre una codifica propria dell'informazione, come ad esempio gli esercizi di completamento di parole o di frasi.

### 1.2.2 Sezione per gli operatori

La piattaforma dispone di due sezioni, una per il personale sanitario (psicologici, neurologi e fisioterapisti) attraverso il quale gestiscono l'iter riabilitativo da un pratico pannello di controllo, mentre l'altra è adempita per i pazienti. MS-rehab dispone di un profilo per ogni paziente, che integra i dati anagrafici, clinici e neuropsicologici, contenente i risultati dei test a cui si è sottoposto e eventuali referti medici. Il profilo è suddiviso in sezioni specifiche che consente all'operatore un rapido accesso alle informazioni che desidera, nell'elenco sottostante sono riportati i vari segmenti accessibili dal personale sanitario.

**Profilo anagrafico** sono memorizzati i dettagli del paziente, come ad esempio l'età della scolarizzazione, lo stato civile, il numero dei figli e l'impiego lavorativo.

**Profilo clinico** contiene tutti i dettagli riguardo l'anno di diagnosi della SM, tipologia, terapie in atto e referti delle risonanze.

**Profilo neuropsicologico** in questa sezione sono memorizzate le informazioni riguardo lo stato neuropsicologico del paziente, nel dettaglio tutti i dati inerenti ai test somministrati al paziente durante le diverse valutazioni cognitive a cui viene sottoposto, come ad esempio i risultati della Rao e del White test.

MS-rehab offre inoltre la possibilità di effettuare cicli di riabilitazione di gruppo, per i pazienti che manifestano deficit simili, tale approccio si è rivelato molto vantaggioso presso il Bellaria dove sono soliti riabilitare dalle 4 alle 8 persone in sessioni settimanali. Attraverso il pannello è possibile, quindi gestire i gruppi e assegnare gli esercizi riabilitativi e la rispettiva

difficoltà (facile, medio, difficile) sia per il singolo paziente sia per l'intero gruppo. Questo tipo di configurazione se automatica rende la piattaforma più fruibile e maneggevole, in quanto si è riusciti ad estrapolare un modello che permette agli operatori di configurare un esercizio in pochi click, settando in automatico la maggior parte dei parametri, a differenza della maggior parte dei sistemi riabilitativi dove la configurazione iniziale richiedere un importante quantitativo di tempo, per definire, stimoli, target, colori, tempi, ecc.

Un ulteriore aspetto del sistema è la possibilità da parte degli operatori di controllare lo svolgimento degli esercizi assegnati al paziente e di modificare la difficoltà in tempo reale da un altro dispositivo, in questo modo al completamento dell'esercizio il sistema propone un nuovo esercizio della difficoltà decisa dall'operatore oppure in caso contrario una difficoltà avanzata impostata dal sistema. Inoltre all'operatore è data la possibilità di osservare l'andamento dei progressi del paziente, mediante un indice che descrive il miglioramento o il peggioramento rispetto gli esercizi svolti in precedenza.

Infine l'operatore ha la possibilità di monitorare i progressi di ciascun paziente nelle diverse aree riabilite, riguardo gli esercizi assegnati, ad esempio quanti esercizi effettivamente sono stati svolti dal paziente e i relative punteggi ottenuti. MS-rehab permette la visualizzazione di grafici aggregati secondo queste tipologie:

**Panoramica generale** grafico contenente le curve di miglioramento delle performance di ciascuna funzione riabilitata (attenzione, memoria e funzioni esecutive).

**Aggregati per funzione riabilitata** grafico che mostra l'andamento delle performance relativo ad una funzione cognitiva specifica.

**Esercizio** è una tipologia di grafico più complessa rispetto quelle appena citate, in quanto mostra l'andamento della riabilitazione relativo ad un esercizio in modo da poter capire le specifiche difficoltà di un paziente e prendere provvedimenti adeguati; infatti è possibile visualizzare ul-

teriori grafici che mostrano il dettaglio dei successi/fallimenti, risposte giuste/sbagliate/omesse e i tempi di gioco.

### 1.2.3 Sezione per il paziente

Nella sezione relativa al paziente viene presentata una visuale chiara ed immediata sugli esercizi che gli sono stati assegnati per la sessione riabilitativa che sia personale o di gruppo; quando il paziente si accinge a svolgere un esercizio può richiedere al sistema una sessione di allenamento nel caso non si ricordi come svolgerlo in modo corretto. Nella sessione di allenamento il paziente non viene valutato e inoltre vengono presentate istruzioni più dettagliate su come eseguire l'esercizio in modo corretto.

### 1.2.4 Esercizi

Attualmente la piattaforma offre un ricco ventaglio di esercizi, circa 23, che differiscono tra i diversi ambiti riabilitativi (attenzione, memoria e funzioni esecutive). Gli esercizi di attenzione (selettiva 1, selettiva 2, alternata, divisa) e di memoria (riconoscimento, visuo-spaziale e di lavoro) possono essere svolti, con diversi tipi di stimoli, su figure, volti o figure orientate. Infine vi sono gli esercizi legati alle funzioni esecutive, che verranno meglio descritti, nel capitolo seguente. La piattaforma offre un comparto grafico che si adatta facilmente a funzionare su tablet e computer con una particolare attenzione verso il mondo del mobile, poiché si vuole mettere a disposizione uno strumento utile agli ospedali e ai pazienti per la riabilitazione domestica, basata su hardware non dedicato allo scopo, favorendo così una maggiore diffusione del prodotto senza obbligare costi aggiuntivi per le periferiche.

### 1.2.5 Architettura

MS-rehab è una applicazione web realizzata con tecnologie Java, utilizzando il framework Spring MVC che implementa il modello Model-View-Controller. La piattaforma è sostenuta da Apache Tomcat che implementa

delle Java Servlet e delle Java Server Pages (JSP) da specifiche di Oracle Corporation, realizzate dal gruppo di ricerca del DISI. In figura 1.2 è mostrata l'architettura e le tecnologie da esso sfruttate; un punto di forza del sistema è l'adattabilità, in quanto può essere utilizzato sia da tablet che da computer. Questo è permesso grazie a Bootstrap, un famoso framework per sviluppare applicazioni responsive basato su tecnologie HTML, CSS e Javascript. MS-rehab può essere fruita su una vasta gamma di dispositivi eterogenei mantenendo compatibilità e coerenza, in quanto le pagine vengono adattate alle dimensioni dello schermo sul quale si opera.

L'applicazione viene ospitata da una macchina virtuale su un server del dipartimento di Scienze Informatiche dell'università di Bologna raggiungibile al sito <http://rehab.cs.unibo.it/MS-rehab-website/>; il server web si basa su tecnologie open source, infatti monta Ubuntu 16.04 il quale fornisce supporto a Java 8 e Apache Tomcat 8.0.39. Allo stesso livello è contenuto un database SQLite che si occupa della persistenza dei dati riguardo i pazienti e delle performance riabilitative ottenute.

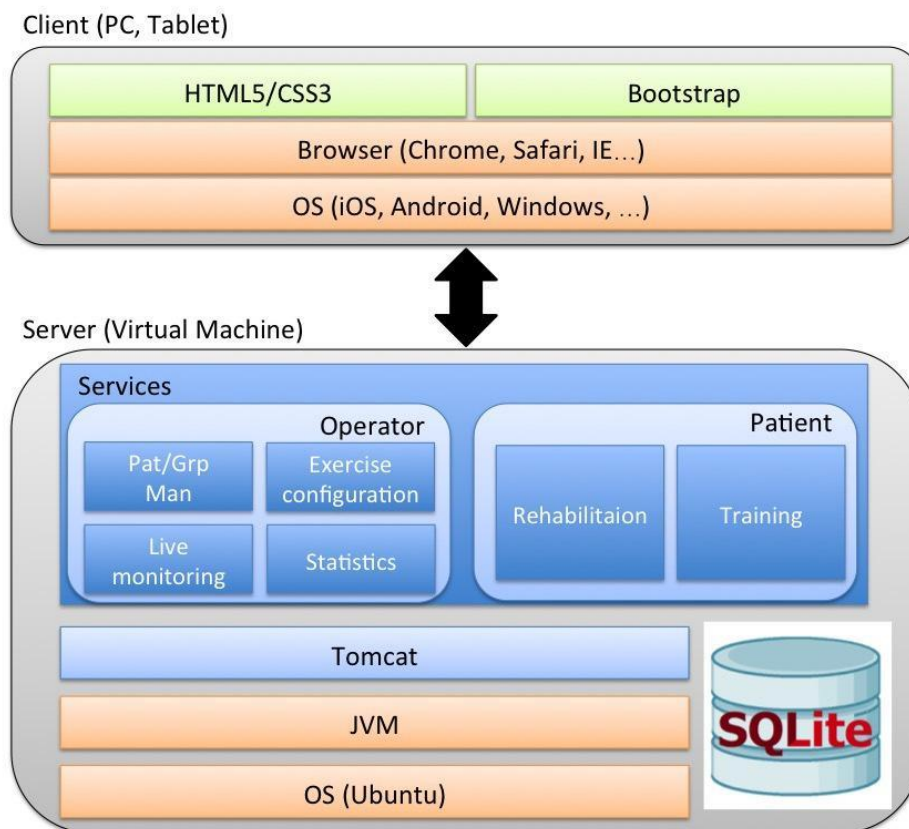


Figura 1.2: Architettura della piattaforma MS-rehab.

# Capitolo 2

## Riabilitazione delle funzioni esecutive

In neuropsicologia e psicologia cognitiva, le funzioni esecutive fanno riferimento a tutti quei processi che permettono ad un individuo di pianificare, attuare dei piani finalizzati al raggiungimento di un obiettivo, monitorare e modificare il proprio comportamento in base alle condizioni contestuali. Una rassegna degli studi neuropsicologici presenti in letteratura, mostra come deficit nelle funzioni esecutive portano ad un rendimento significativamente più basso in termini di anni trascorsi nell'istruzione, nei domini sociali e in altri domini della vita reale [14]. In questo capitolo verranno presentati gli aspetti cognitivi che caratterizzano le funzioni esecutive, tecniche generiche per la riabilitazione ed infine alcuni esercizi utilizzati in contesti clinici con un maggior interesse al test: mappa dello zoo.

### 2.1 Sistema cognitivo

In letteratura sono state fornite diverse definizioni per descrivere il concetto di funzioni esecutive, poiché tale termine non si riferisce ad una singola entità bensì ad un insieme di diversi sottoprocessi necessari allo svolgimento di un determinato compito. Ad esempio una definizione è quella di [Stauss,

1992] dove afferma che le funzioni esecutive sono *"una serie di abilità che permettono alle persone di creare obiettivi, conservarli in memoria, controllare le azioni, prevedere gli ostacoli al fine di raggiungere degli obiettivi"*. Più in dettaglio le funzioni esecutive sono *funzioni corticali superiori* che permettono alla persona di pianificare ed eseguire un piano per il raggiungimento di un obiettivo prefissato, inoltre garantiscono il corretto adattamento comportamentale in base al contesto, soprattutto nella gestione di situazioni nuove e non familiari; sono indispensabili nella vita quotidiana, dal momento che alcune attività richiedono problem-solving, come per esempio pianificare la propria giornata o semplicemente una azione che potrebbe essere quella di prepararsi il pranzo. In sintesi questo complesso sistema cognitivo è un processo fondamentale per il raggiungimento dei nostri obiettivi e si può scomporre in diverse fasi: una fase di analisi, pianificare per raggiungere l'obiettivo, sviluppare un piano per arrivare alla soluzione, considerare una linea temporale, tornare indietro se necessario per modificare il piano ed infine completarlo.

Per attuare con successo tali processi sono necessarie diverse capacità cognitive:

**pianificazione** capacità di formulare un piano e delle azioni da intraprendere in una sequenza gerarchica;

**inibizione** capacità di focalizzarsi sul piano ignorando stimoli motori ed emotivi non adeguati per il raggiungimento dell'obiettivo prefissato;

**memoria di lavoro** capacità di creare e tenere attivo nella mente il piano di lavoro che si è scelto;

**fluenza** capacità di generare soluzioni nuove e diverse rispetto al problema;

**attenzione selettiva** capacità di attenzione prolungata sul compito che si sta svolgendo per un lungo periodo di tempo;

**flessibilità** capacità di passare ad un ulteriore insieme di stimoli in base alle informazioni provenienti dal contesto.



**linguaggio** capacità di esprimersi in maniera adeguata in base al contesto.

Alcune di queste capacità richiedono il supporto di altri domini cognitivi, come ad esempio la memoria e l'attenzione, inoltre sono coinvolti anche meccanismi che regolano le emozioni, comportamenti e motivazioni.

### 2.1.1 Funzione esecutiva: la pianificazione

Prima di introdurre gli aspetti della riabilitazione ecologica e l'esercizio riabilitativo, studiato e automatizzato in questo lavoro di tesi, conosciuto in letteratura come: mappa dello zoo è indispensabile definire cosa si intende per pianificazione; tale concetto verrà inoltre formalizzato in contesto informatico nel capitolo successivo. Si definisce *"la pianificazione come un insieme di attività appartenenti al dominio cognitivo che anticipano e regolano il comportamento, consentendo di eseguire una sequenza di azioni al fine di raggiungere un dato obiettivo"* [17]. Per questa capacità è necessario che un individuo indirizzi l'attenzione sul problema per periodi relativamente lunghi, in modo da avere una capacità di ricezione e di elaborazione degli stimoli. La pianificazione è legata inoltre ad altri processi cognitivi superiori come il problem-solving; il processo in questione è definito da numerose variabili che condizionano la tipologia di strategia adottata, la prontezza e l'efficacia.

## 2.2 Aspetti della riabilitazione ecologica

Nel lobo frontale della neocorteccia risiedono tutti i meccanismi che regolano lo svolgimento delle funzioni esecutive, nello specifico è la corteccia prefrontale ad essere coinvolta nella formulazione e nell'esecuzione di schemi di azione. Nonostante non esista una diagnosi specifica di deficit delle funzioni esecutive, sono numerosi i quadri clinici nei quali è palese una difficoltà di pianificazione, attenzione selettiva, controllo comportamentale o flessibilità nell'adattarsi a situazioni nuove. Un problema a livello generale delle

funzioni esecutive è conosciuto come *sindrome disesecutiva* (DES) i cui effetti potrebbero compromettere gli individui nelle abitudini quotidiane, nel campo scolastico, sociale e lavorativo; tale problema non è necessariamente associata ad una lesione del lobo frontale ma anche ad una serie di deficit nel processamento, nella pianificazione, nell'inibizione e nella flessibilità cognitiva.

Nell'ultimo periodo si sono intensificati gli studi atti a descrivere i diversi processi frontali in modo sistematico, dato che le funzioni esecutive sono suddivise in vari processi maggiormente specifici che possono essere danneggiati singolarmente. Molte tecniche riabilitative per i deficit nella risoluzione di un problema, tendono a far allenare il paziente su tutte le fasi che costituiscono il procedimento che va dall'identificazione del problema alla valutazione della soluzione creata [16]. Il trattamento dei deficit delle funzioni esecutive è possibile mediante la terapia cognitiva e quella farmacologica, ma in questa tesi tratteremo solo gli aspetti riabilitativi costituiti da esercizi cognitivi-funzionali finalizzati al potenziamento delle abilità neuropsicologiche coinvolte, nello specifico esercizi che riguardano in modo particolare le componenti di problem-solving e di pianificazione attraverso compiti complessi con alta validità *ecologica* e buone probabilità di generalizzazione nei diversi contesti di vita. La riabilitazione ecologica non è altro che un insieme di attività poste al paziente con disabilità intellettiva su compiti tipici della vita quotidiana; attraverso esercizi realistici inseriti in un contesto credibile riescono ad impattare sulla qualità della vita del paziente in modo positivo.

### 2.3 Esercizi cognitivi-funzionali

Uno degli esercizi più noti per la riabilitazione delle funzioni esecutive è chiamato *mattina di impegni*, conosciuto in letteratura con l'acronimo PAD (Plan-A-Day), proposto da Funke, J. & Krüger, T. (1993) [9]. Il paziente deve svolgere una serie di compiti che si presentano nella vita di tutti i giorni, come ad esempio uscire di casa, andare in banca a prelevare il denaro,

accompagnare la figlia in piscina e degli orari ai quali svolgere tali impegni. Quindi il compito del paziente è quello di pianificare una sequenza di azioni per completare tutti gli obiettivi nei tempi stabiliti. Inoltre l'esercizio richiede di gestire ulteriori sottoprocessi, come ad esempio se viene richiesto il compito di fare la spesa, il baule dell'automobile deve essere vuoto e quindi svuotato nel caso fosse pieno per far fronte al volume richiesto dalle cose che si devono acquistare. Si tratta di passaggi da effettuare che denotano un livello piuttosto elevato di comprensione della realtà e richiedono alcune strategie di azione molto stimolanti per pazienti affetti da disfunzioni esecutive.

Un altro esempio di test riabilitativo è la *Mappa di Firenze* che sulla falsa riga dell'esercizio precedentemente descritto, induce il paziente a pianificare un percorso per visitare una serie di monumenti nella città di Firenze, come ad esempio il Duomo, la Galleria degli Uffizi, il Museo di S.Marco ed altri ancora con l'obiettivo di recarsi in tutti i posti in un tempo fissato, prestando attenzione che in alcuni musei è richiesta una visita di alcune ore.

## 2.4 Mappa dello zoo

L'esercizio che è stato automatizzato in questa tesi è *mappa dello zoo*, proposto da Wilson et al., 1998, uno degli esercizi più noti in letteratura appartenente alla categoria *Behavioural Assessment of the Dysexecutive Syndrome* (BADS) [31], una batteria di test per la valutazione comportamentale, della DES, ovvero dei deficit delle funzioni esecutive. Nello specifico, misura le abilità coinvolte nella vita quotidiana come ad esempio la pianificazione, l'organizzazione, l'attenzione e il problem-solving, la cui alterazione può impedire il recupero funzionale e l'abilità di rispondere ai programmi riabilitativi. BADS è composto da alcuni test, ciascuno dei quali presenta situazioni e problematiche facilmente riscontrabili nella vita di tutti i giorni. Il paziente, in ogni test è lasciato libero di decidere come organizzare la soluzione del compito e gestire il tempo a disposizione.

Il sistema è composto da sei esercizi:

**Cambio regola delle carte** valuta l'abilità di cambiare uno schema stabilito di risposta allo stimolo usando oggetti familiari;

**Programma di azione** è una prova di problem-solving, per la cui risoluzione è necessario combinare l'uso di alcuni strumenti forniti in una precisa sequenza;

**Ricerca di chiavi** il soggetto deve dimostrare come cercherebbe in un ambiente un mazzo di chiavi e la sua strategia viene valutata in base alla funzionalità;

**Stime cognitive** attraverso quattro domande si misura la capacità di fare stime sulla durata di alcuni eventi;

**Mappa dello zoo** il soggetto deve programmare un percorso in uno zoo scegliendo tra una serie di tappe proposte;

**Sei elementi** valuta la capacità di organizzare sei compiti da svolgere nell'arco di dieci minuti.

Esistono diverse aziende sul mercato che producono e vendono il kit BADS contenente il manuale, gli esercizi appena descritti ed eventuale materiale per effettuare i test ad un prezzo di seicento sterline.

In Figura 2.1 è mostrata un'istanza dell'esercizio mappa dello zoo, adatto a misurare le capacità di pianificazione e di problem-solving attraverso l'abilità nella messa in atto di un comportamento sequenziale.

I pazienti sono tenuti a formulare e attuare un piano per visitare una serie di luoghi designati sulla mappa di uno zoo. Tuttavia, quando si pianifica il percorso, alcune regole devono essere rispettate. La mappa e le regole sono state costruite in modo che ci siano solo quattro variazioni su una rotta che possono essere seguite in modo che nessuna delle regole venga violata. Le regole come ad esempio il vincolo dell'ordine con il quale visitare i luoghi, i vincoli sulle strade che possono essere transitabili una volta sola o in una sola direzione e vincoli di inizio e fine percorso, rendono più complesse le soluzioni che il paziente deve produrre per l'obiettivo.

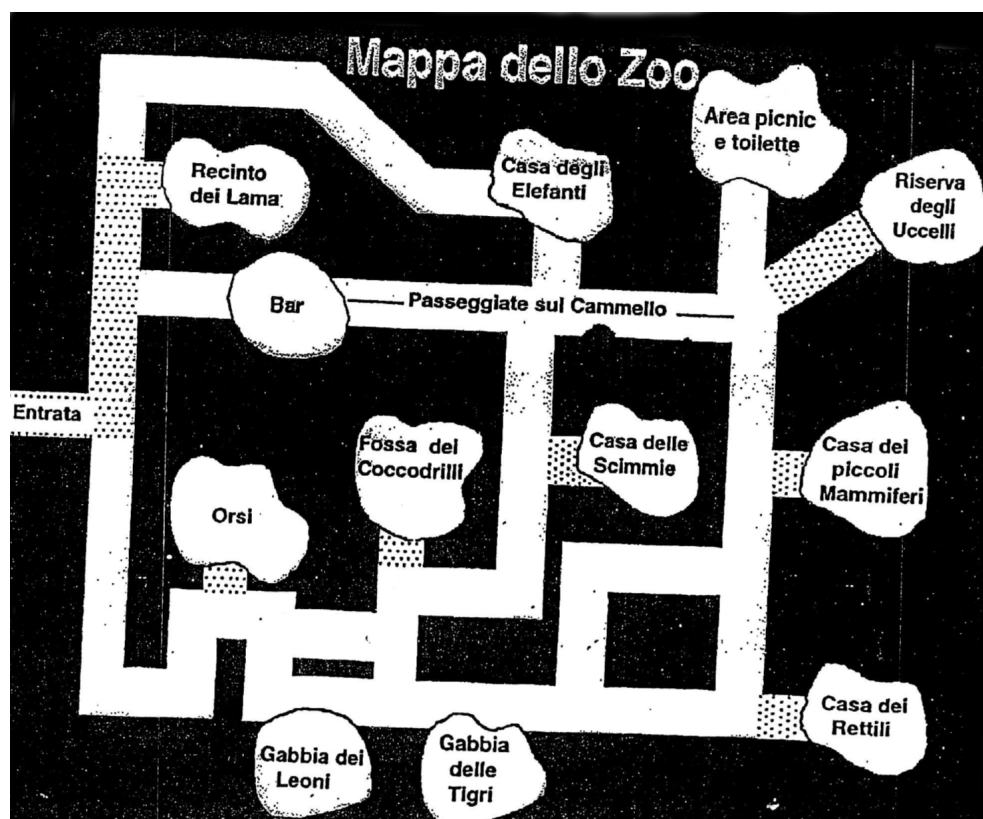


Figura 2.1: Test BADS: mappa dello zoo (dodici target)

Generalmente l'esercizio viene adattato al paziente, in Figura 2.2 è possibile notare come alcuni posti visitabili sulla mappa siano scomparsi rispetto alla Figura 2.1, questo vuol dire diminuire il numero degli obiettivi sulla mappa e modulare la difficoltà incrementandola eventualmente quando si verifica una stabilizzazione del punteggio ottenuto.

Le mappe di tutti gli esercizi appena descritti sono stereotipate, rappresentate con un modello esemplificato, in cui le strade risultano essere connettori ai target principali, tra i quali ci sono i target necessari a rappresentare posizioni utili per la navigazione del paziente.

In un contesto clinico, gli esercizi vengono proposti al paziente su supporto cartaceo grazie al quale lo stesso può manifestare le sue intenzioni circa gli spostamenti che intende compiere e che l'operatore può seguire durante lo

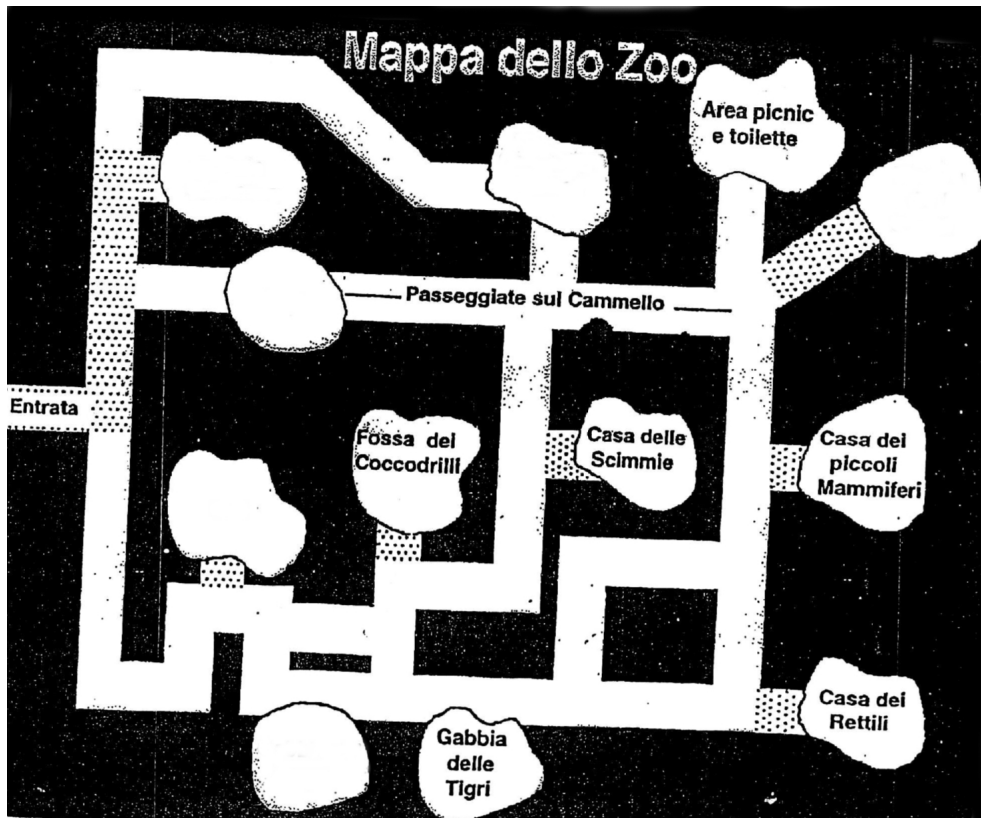


Figura 2.2: Test BADS: mappa dello zoo (sei target)

svolgimento dell'esercitazione.

Ad oggi, non esiste a nostra conoscenza un'implementazione automatizzata di mappa dello zoo nei vari sistemi informatici di riabilitazione in uso, a differenza degli esercizi descritti in sezione 2.3.

# Capitolo 3

## Pianificazione

L'intelligenza artificiale definisce la pianificazione come la progettazione di un piano di azioni per raggiungere dati obiettivi, ossia come un agente può sfruttare la struttura di un problema per costruire complessi piani d'azione. In questo capitolo verranno definiti il concetto formale di *planning*, la pianificazione classica e il linguaggio PDDL per la rappresentazione dei problemi.

### 3.1 Modello concettuale

Quotidianamente, ogni persona pianifica la propria giornata in base alle proprie attività valutando gli impegni che deve svolgere, tenendo in considerazione distanze geografiche, tempi di percorrenza ed altri fattori. Nel lessico comune, la parola *piano* può assumere diversi significati in base al contesto, ma nello specifico rappresenta un insieme di azioni con vincoli che uno o più agenti devono eseguire per raggiungere un dato obiettivo. Per definire concettualmente cosa sia il *planning* si è scelto di fare riferimento ad un modello teorico [22] caratterizzato da tre componenti principali: sistema a transizione di stati, controllore e pianificatore.

### 3.1.1 Sistema a transizione di stati

**Definizione.** *Un sistema a transizione di stati è un modello formale che definisce lo spazio degli stati e transizioni per rappresentare tutte le possibili situazioni che si possono verificare in una certa situazione del mondo reale.*

*Può essere definito come una quadrupla  $\Sigma = (S, A, E, \gamma)$ , dove:*

- $S = \{s_0, s_1, s_2, \dots\}$ , insieme degli stati;
- $A = \{a_0, a_1, a_2, \dots\}$ , insieme delle azioni;
- $E = \{e_0, e_1, e_2, \dots\}$ , insieme di eventi esogeni;
- $\gamma : S \times (A \cup E) \rightarrow 2^S$ , funzione di transizione.

I sistemi a transizione di stati possono essere rappresentati come grafi orientati, dove i nodi sono gli stati appartenenti all'insieme  $S$  e gli archi sono rappresentati dagli elementi  $A \cup E$ . Nello specifico se  $s' \in \gamma(s, a)$ , dove  $a \in A \cup E$  è un'azione o un evento, allora il grafo conterrà un arco che rappresenta la transizione dallo stato  $s$  allo stato  $s'$  etichettata con l'azione o l'evento  $a$ .

E' opportuno fare due considerazioni:

- Se  $a$  è un'azione e l'output della funzione  $\gamma(s, a)$  è un insieme non vuoto, allora vuol dire che l'azione  $a$  è applicabile allo stato  $s$ . Quindi se viene eseguita l'azione  $a$  nello stato  $s$ , allora il sistema transiterà in qualche stato appartenente all'insieme restituito dalla funzione di transizione  $\gamma(s, a)$ .
- Se  $e$  è un evento e  $\gamma(s, e)$  non è vuoto, allora l'evento  $e$  potrebbe verificarsi quando il sistema è nello stato  $s$ . Questi possono essere eventi dinamici del sistema, non controllabili dall'esecutore del piano. Come per l'azione anche in questo caso, all'avvenire di un evento in uno stato  $s$ , il sistema transiterà in qualche stato appartenente a  $\gamma(s, e)$ .



Lo scopo del pianificatore è quello di trovare una qualsiasi sequenza di azioni da applicare nel sistema per transitare dallo stato iniziale allo stato obiettivo. Lo stato goal  $s_g$  potrebbe essere un elemento appartenente all'insieme degli stati  $S$ , oppure un insieme definito come  $S_g = \{s_3, s_4, s_5\}$ , in questo caso l'obiettivo viene raggiunto quando il sistema raggiunge uno degli stati in  $S_g$ .

### 3.1.2 Pianificatore

Le informazioni che vengono definite prima di iniziare la pianificazione sono: il sistema  $\Sigma$ , gli obiettivi e una situazione iniziale. Tali componenti formano un problema  $P$  nel dominio del mondo reale che si è voluto rappresentare. Un pianificatore prende in input un problema  $P$  e restituisce un piano di azioni, nel caso in cui esistesse una sequenza di azioni che permette al sistema di transitare dallo stato iniziale allo stato obiettivo.

Un *piano* quindi è una sequenza di azioni che permette di raggiungere un obiettivo, per esempio:  $\langle \text{Muovi}, \text{MuoviSulTavolo} \rangle$  che il sistema può eseguire per arrivare allo stato goal. Queste azioni potrebbero descrivere uno dei più famosi domini di pianificazione, ossia il mondo dei blocchi [26] che consiste nella creazione di una o più pile di blocchi su un tavolo, attraverso un braccio robotico. Il braccio può tenere un blocco per volta e di conseguenza non può afferrare un blocco se ne ha sopra un altro, quindi può effettuare lo spostamento di un blocco in un'altra posizione, o sul tavolo attraverso l'azione "MuoviSulTavolo" o su un secondo blocco mediante l'azione "Muovi". Una *policy* invece è una funzione parziale dagli stati alle azioni, un esempio applicato al problema dei blocchi, potrebbe essere  $\{(s_0, \text{MuoviSulTavolo}), (s_1, \text{Muovi}), (s_3, \text{Muovi}), (s_5, \text{MuoviSulTavolo})\}$ , questo esempio verrà descritto successivamente nella sezione 3.3.2.

Il pianificatore combina il piano e la policy per risolvere il problema  $P$  definito, in modo tale che il sistema  $\Sigma$  attraverserà una sequenza di stati partendo dallo stato iniziale  $s_0$  per raggiungere lo stato o uno degli stati obiettivo.

### 3.1.3 Controllore

L'ultimo componente è il controllore situato logicamente tra il sistema a transizione di stati e il pianificatore a cui è assegnato il compito di elaborare l'azione da far eseguire al sistema, prendendo in input il piano di azioni elaborato dal pianificatore e le osservazioni riguardo lo stato attuale del mondo reale. Le *osservazioni* sono collezioni di dati che possono per esempio essere ottenute da sensori con il fine di comprendere lo stato corrente del sistema  $\Sigma$ . La modellazione delle osservazioni può avvenire attraverso la funzione  $\eta : S \rightarrow O$ , la quale mappa  $S$  in un insieme discreto di possibili osservazioni, quindi in input il controllore valuterà l'osservazione dello stato corrente  $s$ , ovvero  $o = \eta(s)$ . Possiamo dedurre che l'osservazione dipenda dal tipo di ambiente con cui si interagisce, quindi:

**Informazione completa** si ottiene se la funzione  $\eta$  è iniettiva, per cui ogni osservazione  $o$  corrisponde ad uno stato del sistema;

**Informazione incompleta** si ottiene se la funzione  $\eta$  non è iniettiva, quindi per ogni osservazione  $o$  il sistema è in uno stato dell'insieme  $\eta^{-1}(o) \subseteq S$ .

A questo punto è importante rivelare che per la realizzazione di questa tesi abbiamo preso in esame un ambiente completamente osservabile, deterministico, stocastico con un singolo agente.

## 3.2 Dominio indipendente

Allo stato dell'arte esistono diversi tipi di pianificatori, che possono essere configurati per funzionare in diversi domini di pianificazione; per la realizzazione di questa tesi si è utilizzato un pianificatore a dominio indipendente, che mira a risolvere il problema di pianificazione dato tramite un appropriato algoritmo di ricerca indipendente dal dominio. L'algoritmo di ricerca incorporato nel pianificatore non è specifico del problema da risolvere, si tratta di un algoritmo di ricerca indipendente dal dominio e utilizzato indipendentemente dal problema dato in input. L'efficienza della pianificazione si

basa sulla capacità del pianificatore di estrarre informazioni dalla struttura del problema e dai percorsi promettenti alle soluzioni, utilizzando l'euristica della ricerca informata. A differenza degli approcci dominio dipendenti che si basano su algoritmi specificamente concepiti per ogni tipo di problema, i pianificatori sono basati su un linguaggio dichiarativo (non procedurale e di livello molto alto) che specifica cosa deve essere fatto piuttosto che come farlo. La tipologia di configurazione a dominio indipendente viene utilizzata nella *pianificazione classica* e lavora con domini che hanno le seguenti restrizioni:

- il sistema  $\Sigma$  è composto da un insieme finito di stati;
- si ha totale conoscenza dello stato in cui si trova il sistema (completamente osservabile),  $\eta$  è una funzione identità;
- il sistema è deterministico, quindi per ogni stato  $s$  e azione  $a$ , si ha che  $|\gamma(s, a)| \leq 1$  questo significa che se l'azione  $a$  è applicabile allo stato  $s$  allora il sistema transiterà nello stato risultante della funzione di transizione  $\gamma(s, a)$ ;
- il sistema  $\Sigma$  è statico, quindi non possono verificarsi eventi dinamici interni ( $E = \emptyset$ );
- gli stati goal o lo stato goal sono elementi appartenenti all'insieme degli stati  $S$ , l'obiettivo viene raggiunto quando si è trovata una sequenza di azioni che porta il sistema nello stato goal o negli stati goals;
- la soluzione è una sequenza finita di azioni;
- le azioni non hanno durata, quindi sono istantanee;
- gli stati iniziali e gli stati obiettivo sono determinati a prescindere dalle dinamiche che possono verificarsi all'interno del sistema  $\Sigma$ .

Riassumendo, la pianificazione classica riduce la definizione formale di problema in: dato  $\Sigma = (S, A, \gamma)$ , lo stato iniziale  $s_0$  e un sottoinsieme di stati

goal  $S_g$ , trovare la sequenza di azioni corrispondenti alla sequenza di transizioni  $(s_0, s_1, \dots, s_n)$  tali che  $s_1 \in \gamma(s_0, a_1)$ ,  $s_2 \in \gamma(s_1, a_2)$ , ...,  $s_n \in \gamma(s_{n-1}, a_n)$  e  $s_n \in S_g$ .

### 3.3 Rappresentazione del problema

Abbiamo appena descritto formalmente i quattro elementi utili a definire un problema di pianificazione: lo stato iniziale, le azioni, il risultato dell'applicazione di un'azione e gli stati goal. Nel 1998, Drew McDermott e i suoi colleghi svilupparono PDDL (Planning Domain Definition Language) un linguaggio di pianificazione basato su STRIPS, che ha permesso anche la formalizzazione di precondizioni e obiettivi con letterali negativi. Questo linguaggio negli anni successivi è diventato lo standard per la rappresentazione fattorizzata di problemi di pianificazione, in cui ogni stato del mondo è rappresentato da una collezione di variabili. Il linguaggio, attraverso tale formalismo, consente di descrivere il dominio e la relativa descrizione del problema favorendo un ampio riutilizzo e consentendo un confronto più diretto agli approcci, che si traducono quindi in un rapido sviluppo della ricerca in questo ambito. Infatti, gli aspetti comuni ad una classe di problemi, sono formalizzati nella descrizione del dominio: stati e azioni. Nel problema, invece, sono specificati lo stato iniziale, gli stati obiettivo ed eventuali criteri per aiutare il risolutore a cercare la soluzione [19].

In PDDL ogni *stato* è rappresentato come una congiunzione di atomi, inoltre viene utilizzata l'ipotesi del mondo chiuso detta anche *semantica dei database*, ciò significa che tutto quello che non viene menzionato è falso e ogni variabile è distinta. Le *azioni* sono definite da uno o più *schemi di azioni* costituiti da:

**nome dell'azione** il nome con cui si sceglie di definire l'azione;

**elenco di variabili** necessarie affinché l'azione possa essere compiuta;

**precondizione** lo stato in cui tale azione può essere eseguita;

**effetto** lo stato in cui il sistema si troverà dopo l'esecuzione di tale azione.

Si ribadisce che, data un'azione  $a$  ed uno stato  $s$  allora  $a$  è *applicabile* in  $s$  se le precondizioni sono soddisfatte in  $s$ . Il risultato dell'esecuzione di  $a$  in  $s$  è la transazione del sistema dallo stato  $s$  ad  $s'$ .

### 3.3.1 Modellare con PDDL

Prima di presentare un esempio completo, si è scelto di mostrare lo schema principale di PDDL, poiché per modellare un dominio occorre definire alcuni aspetti fondamentali per la realizzazione [19]. Di seguito verrà mostrato l'elenco delle istruzioni contenute nel file `.pddl` utili a formalizzare un dominio:

**domain** nome del dominio;

**requirement** aspetto del modello che il sistema deve sfruttare come ad esempio termini atomici, uguaglianze, tipizzazione, ecc;

**object-type hierarchy** definisce una relazione tra gli elementi, o meglio una gerarchia di entità, ad esempio macchina appartiene alla classe dei veicoli;

**constants** elementi costanti a tutta la classe di problemi associata al dominio;

**predicates** insieme di fatti logici deducibili;

**actions** nome dell'azione;

**parameters** parametri rappresentati dalle entità del modello;

**preconditions** le condizioni in cui deve trovarsi il modello affinché possa essere intrapresa tale azione;

**effects** gli effetti dell'azione sul sistema al termine dell'azione svolta.

Le azioni possono presentare anche operazioni condizionali **when-effects**.

Per modellare un problema, invece è necessario definire i parametri fondamentali necessari al raggiungimento della soluzione:

**problem** nome del problema;

**domain** nome del dominio relativo al problema;

**objects** elementi presenti nel mondo;

**init** insieme di stati che rappresentano lo stato iniziale del sistema;

**goal** insieme di stati obiettivo.

### 3.3.2 Modellare con PDDL: il mondo dei blocchi

Per illustrare bene il linguaggio PDDL è opportuno presentare un esempio, utilizzando uno dei più famosi domini di pianificazione dell'intelligenza artificiale: il mondo dei blocchi [26]. Immaginiamo dei blocchi della stessa forma di qualsiasi materiale posti su un tavolo, ognuno identificato attraverso una lettera dell'alfabeto maiuscola come possiamo notare dalla figura 3.1; un braccio robotico può afferrare un solo blocco per volta e spostarlo sul tavolo o impilarlo su un altro, ma non può spostare un blocco che ne ha sopra un altro. L'obiettivo è quello di costruire una pila di blocchi ordinata con un dato criterio come nella figura 3.1, dove lo stato goal è ottenere una pila di blocchi in ordine alfabetico (A su B e B su C).

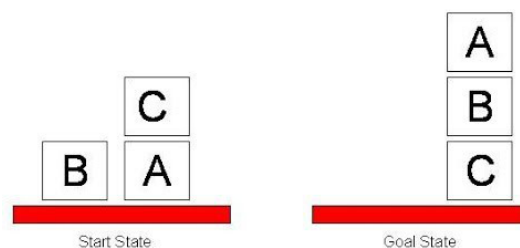


Figura 3.1: Schema del problema: il mondo dei blocchi; a sinistra della figura è rappresentato lo stato iniziale del mondo, mentre a destra lo stato obiettivo.

```
(define (domain mondoDeiBlocchi)
  (:requirements :strips)

  (:predicates
    (on ?x ?y) (libero ?x) (blocco ?x))

  (:action Muovi
    :parameters
      (?b ?x ?y)
    :precondition
      (and (blocco ?b) (libero ?b) (on ?b ?x) (blocco ?y)
        (libero ?y))
    :effect
      (and (not (on ?b ?x)) (libero ?x) (not (libero ?y))
        (on ?b ?y)))

  (:action MuoviSulTavolo
    :parameters
      (?b ?x)
    :precondition
      (and (blocco ?b) (on ?b ?x) (libero ?b))
    :effect
      (and (not (on ?b ?x)) (libero ?x) (on ?b tavolo)))
  )
```

Listing 3.1: Dominio: il mondo dei blocchi.

Attraverso l'azione *MuoviSulTavolo*, il braccio robotico può prendere un blocco libero (non ha nessun blocco sopra) e spostarlo in un'altra posizione, o sul tavolo; mentre con l'azione *Muovi* può spostare il blocco esclusivamente su un altro blocco. Nella parte di codice PDDL 3.1, è definito il dominio contenente l'implementazione delle azioni appena descritte. L'azione *Muovi*(*?b ?x ?y*) permette di muovere il blocco *b* da sopra *x* a sopra *y* se sia *b* che *y*

sono liberi, poiché una delle precondizioni per muovere  $b$  è che non ci sia un blocco su di esso. Alla fine dell'esecuzione dell'azione, il blocco  $b$  sarà libero, ma  $y$  no. Il predicato  $(on\ ?x\ ?y)$  viene usato per indicare che il blocco  $y$  si trova su  $x$ , dove  $x$  può essere un altro blocco o il tavolo. Dato che il linguaggio PDDL non consente quantificatori, si introducono ulteriori predicati;  $(libero\ ?x)$  che è vero quando c'è uno spazio libero sopra  $x$  per appoggiarvi un blocco, però in questo caso  $libero(tavolo)$  sarà sempre vero quindi il predicato  $(blocco\ ?x)$  viene introdotto come precondizione dell'azione  $Muovi(b,x,tavolo)$  per impedire al pianificatore di usare  $MuoviSulTavolo(b,x)$ .

```
(define (problem problemaMondoDeiBlocchi)
  (:domain mondoDeiBlocchi)

  (:objects a b c tavolo)

  (:init
    (on b tavolo) (on a tavolo) (on c a)
    (libero b) (libero c) (libero tavolo)
    (blocco a) (blocco b) (blocco c)
  )

  (:goal
    (and (on a b) (on b c) (on c tavolo))
  )
)
```

Listing 3.2: Problema: il mondo dei blocchi.

Nell'esempio presentato in Listing 3.2 si mette in luce tutti gli aspetti tipici di un problema: come la definizione del goal con predicati uniti da una congiunzione in `and`; gli stati iniziali: ovvero i predicati noti al problema. Il risultato del pianificatore sarà un piano avente una lista di azioni che farà transitare il sistema dallo stato iniziale allo stato obiettivo (Figura 3.1).



# Capitolo 4

## Realizzazione

La mappa dello zoo fa parte di una batteria di test impiegati per la valutazione comportamentale della DES, ovvero dei deficit delle funzioni esecutive [31]. Trattandosi di un test utilizzato per la fase di valutazione, non è opportuno impiegarlo anche nella fase riabilitativa del paziente. E' stata quindi progettata una versione alternativa da utilizzare come esercizio riabilitativo mantenendo la struttura concettuale di base ma cambiando la mappa a cui è stato aggiunto un nuovo tema, lo zoosafari, un nuovo comparto grafico e la possibilità di randomizzare gli obiettivi da raggiungere e la posizione degli stessi nella mappa. In questo modo ci si discosta dal test della batteria di valutazione generando problemi differenti e sempre diversi rendendo l'esercizio adatto per la riabilitazione. In questo capitolo verranno descritti nel dettaglio l'analisi della conoscenza nello scenario dello zoosafari, il modello utilizzato per la generazione del problema ed infine la tecnica utilizzata per la realizzazione della grafica.

### 4.1 Lo zoosafari

Lo zoosafari richiede di pianificare un percorso attraverso una mappa che rappresenta l'ambiente da visitare, con l'obiettivo di recarsi in un sottoinsieme di animali e finire il percorso in un dato luogo, rispettando alcune regole.

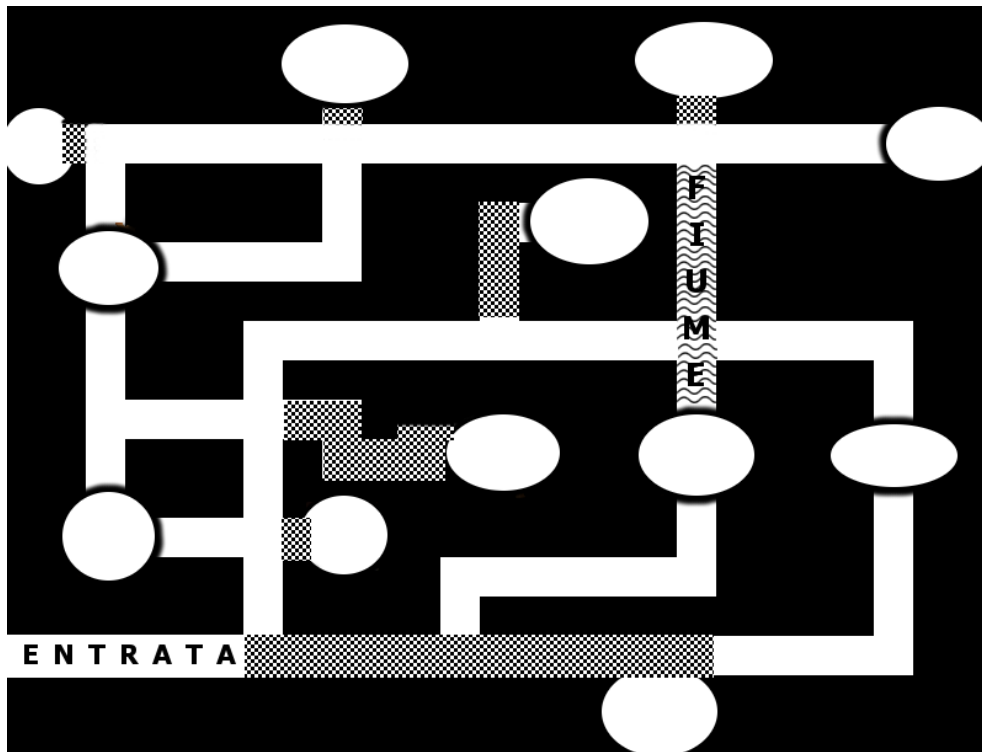


Figura 4.1: Struttura della mappa zoosafari.

Come per la mappa dello zoo la mappa è costituita da due tipologie di sentieri, viene denominato "bianco" il sentiero attraversabile una sola volta e "ombreggiato" il sentiero in cui il passaggio è consentito un numero illimitato di volte. Inoltre l'esercizio presenta delle regole, ereditate dal test mappa dello zoo come i vincoli sulle strade che possono essere transitabili una volta sola, la possibilità di fare una volta sola l'attraversamento del fiume e vincoli di inizio e fine percorso. Durante la fase di analisi è stato progettato lo schema mostrato in Figura 4.1 in cui è possibile notare le tipologie differenti di sentiero appena citate, l'attraversamento del fiume, i punti in cui verranno predisposti gli animali e l'entrata principale. E' opportuno esplicitare che tale schema è stato ottenuto effettuando delle rotazioni e riflessioni grafiche della mappa dello zoo (Figura 2.1) poiché l'obiettivo principale era quello di generare una nuova mappa senza effettuare modifiche alla struttura originale

del test.

Inoltre è stato disegnato ed infine implementato un comparto grafico, totalmente rivisitato rispetto al test cartaceo della mappa dello zoo mostrato in Figura 2.1, con una serie di animali disponibili, applicando una diversificazione della posizione delle strade e sostituendo la passeggiata sul cammello con l'attraversamento del fiume.

Ogni esercizio presente nel sistema MS-rehab dispone di tre classi di difficoltà principali (facile, medio e difficile) e ulteriori livelli di difficoltà intermedi, definiti da un intervallo che va da uno (più facile) a dieci (più difficile) compresi. Ciò vuol dire che dal livello uno al livello tre si tratta di difficoltà facile, dal quattro al sette invece della classe di difficoltà medio e dal livello otto in poi l'esercizio viene considerato difficile. Per rendere zoosafari congruente al ventaglio di esercizi disponibili in MS-rehab e consoni all'architettura, in fase di progettazione è stato necessario introdurre ulteriori dettagli per garantire la piena compatibilità con il modello che si occupa della gestione della difficoltà e la successiva valutazione delle performance; questi aspetti verranno approfonditi nel capitolo seguente che riguarda l'integrazione di zoosafari nel sistema riabilitativo MS-rehab.

Per la mappa di difficoltà facile si è scelto di rendere tutti i sentieri di tipologia ombreggiati, in questo modo il paziente è libero di spostarsi nella mappa senza dover preoccuparsi di pianificare un percorso valido, tale scelta è stata elaborata con lo scopo di far acquisire maggiore familiarità con l'esercizio. Lo zoosafari in difficoltà media invece mantiene la struttura (tipologia di sentieri) mostrata in Figura 4.1, ma offre un feedback grafico al paziente riguardo i sentieri di tipologia bianchi che sono stati già percorsi, quindi il visitatore sarà consapevole degli spostamenti che potrà effettuare senza rischiare di trovarsi in un vicolo cieco. Infine per l'esercizio in modalità difficile il paziente è costretto a pianificare a priori il percorso e ricordare tutti i sentieri che ha attraversato se non vuole rischiare di perdere la partita. L'intero che identifica il livello per ogni classe di difficoltà corrisponde ad un numero di animali da visitare, infatti ad ogni incremento di quest'ultimo corrisponde

ad un aumento degli obiettivi da visitare (fino ad un massimo di sette) sulla mappa per completare il gioco.

Con tutti gli accorgimenti appena descritti è stato possibile implementare una versione alternativa chiamata zoosafari che si differisce completamente dal test della batteria di valutazione rendendo l'esercizio adatto per la riabilitazione.

## 4.2 Analisi della conoscenza nello zoosafari

Con lo scopo di modellare il problema nel migliore dei modi, sono state estrapolate le azioni atomiche fondamentali per risolvere l'esercizio:

- camminare su un sentiero "bianco" al fine di raggiungere un luogo;
- camminare su un sentiero "ombreggiato";
- salire sul gommone;
- navigare in gommone;
- scendere dal gommone.

Inoltre sono stati considerati i seguenti vincoli:

- iniziare da un luogo specifico;
- finire il percorso in un dato luogo;
- attraversare una sola volta i sentieri "bianchi";
- passaggio sempre consentito sui sentieri "ombreggiati";
- effettuare una volta sola l'attraversamento del fiume.

Dopo una lunga analisi, sono stati individuati una serie di predicati che permettono di realizzare questi vincoli, da utilizzare al fine di modellare il

dominio. Inoltre tutte le azioni sono state formalizzate in azioni comprensibili dal pianificatore, le quali verranno successivamente presentate le singole implementazioni in linguaggio PDDL.

L'obiettivo è quello di muoversi nella mappa, in cui gli stati goals rappresentano la visita di alcuni luoghi/animali (tra orso, toilette, lama, scimmia, leone, bar, serpente, elefante, tigre, uccello, ippopotamo, e scoiattolo), considerando alcune regole che devono essere rispettate e finendo il percorso in un dato luogo. Le regole come ad esempio i vincoli sulle strade che possono essere transitabili una volta sola o in una sola direzione e vincoli di inizio e fine percorso, rendono più complesse le soluzioni che il pianificatore deve generare per raggiungere l'obiettivo. La soluzione che genera il pianificatore è una concatenazione di azioni che porta l'agente a transitare dallo stato iniziale negli stati goals, ovvero muoversi dal punto di inizio della mappa, visitare gli stati obiettivo e finire in un luogo specifico. Il problema dello zoosafari ha sempre almeno una soluzione, in quanto lo scenario è per costruzione risolvibile in un numero finito di passi, questa caratteristica è stata ereditata dalla mappa dello zoo.

Il fallimento può avvenire ad ogni difficoltà, per la mappa di classe facile in cui si è scelto di rendere tutti i sentieri di tipologia ombreggiati si è considerato un tempo massimo in cui è possibile visitare tutti gli obiettivi, se il visitatore supera il tempo stabilito senza completare gli obiettivi del gioco, allora viene considerato fallimento. Invece per le classi di difficoltà media e difficile in cui la mappa è composta principalmente da sentieri di tipologia bianchi è facile trovarsi in un vicolo cieco senza la possibilità di tornare più indietro e quindi si raggiunge una situazione di fallimento. Uno dei principali requisiti di zoosafari è la presenza di contenuti dinamici all'interno della mappa (stato iniziale e obiettivi), infatti ad ogni istanza dell'esercizio gli animali vengono collocati in maniera random sulla mappa (le possibili posizioni sono mostrate in Figura 4.1), allo stesso modo per gli obiettivi da visitare e il luogo in cui finire il percorso. Questa caratteristica permette il costante apporto di nuovi scenari in modo che sia virtualmente impossibile

che si debba risolvere due volte lo stesso.

### 4.3 Utilizzo della pianificazione

La realizzazione di zoosafari è avvenuta con un approccio orientato alla pianificazione, realizzando un componente essenziale senza il quale non sarebbe stato possibile generare un piano di azioni per raggiungere gli obiettivi definiti nel problema. Nel caso specifico, il pianificatore entra in gioco quando, una volta definita la struttura della mappa, le posizioni degli animali/luoghi (del tutto casuali sulla mappa) e il luogo in cui finire il percorso, si effettua il controllo se esista un cammino tale per cui l'agente sia in grado di visitare tutti gli obiettivi e concludere il percorso nel luogo dato. Se il pianificatore non determina un piano di azione, allora viene generata un'altra istanza di zoosafari fino a quando non venga individuata una soluzione. Da come si può intuire, quindi, il pianificatore viene interrogato in una sola fase, ossia quella di controllo della mappa con la configurazione generata degli animali, prima che quest'ultima venga inviata all'utente.

Un altro approccio percorribile per una realizzazione alternativa è quello ottenuto attraverso la programmazione a vincoli, in cui per ogni stato sarebbe necessario introdurre una serie di variabili, in questo caso un problema risulta risolto quando ogni variabile assume un valore che soddisfi tutti i vincoli su di essa. Un problema descritto in questo modo è chiamato "problema di soddisfacimento di vincoli" o CSP (*Constraint Satisfaction Problem*). Tuttavia si è ritenuto opportuno utilizzare la tecnica della pianificazione, capace di generare la sequenza di azioni dato un problema, queste possono risultare attinenti alle azioni che deve svolgere il paziente per risolvere l'esercizio.

Ad oggi esistono alcuni software riabilitativi che fanno uso della pianificazione ad esempio, un gruppo di ricerca canadese ha realizzato uno strumento capace di monitorare l'esecuzione di un piano da parte del paziente per generare promemoria riguardo le attività imminenti, allarmi per guasti anticipati, guida e rimedio nell'attività in corso. Lo strumento dovrebbe ridurre il ca-

rico di lavoro del personale sanitario e allo stesso tempo fornire al paziente un'assistenza più flessibile e personalizzata [15].

## 4.4 Formalizzazione in PDDL

Una volta conclusa l'analisi della conoscenza si è posta l'attenzione sulla definizione e sull'implementazione della sequenza di azioni applicabili per risolvere il problema zoosafari dato e descritto attraverso il linguaggio PDDL. A tal proposito si sono formalizzate le azioni individuate in precedenza secondo la sintassi del solver.

```
(:action Walk
  :parameters (?visitor ?from ?to)
  :precondition
    (and (visitor ?visitor)
          (at ?visitor ?from)
          (mobile ?visitor)
          (road ?from ?to)
          (not (= ?from ?to))
          (able-to-walk ?from ?to)
        )
  :effect
    (and (at ?visitor ?to)
          (not (to-discover ?to))
          (not (at ?visitor ?from))
          (not (able-to-walk ?to ?from))
          (not (able-to-walk ?from ?to))
        )
  )
)
```

Listing 4.1: Azione di attraversamento su sentiero bianco.

La parte di codice PDDL in Listing 4.1 mostra l'implementazione dell'a-

zione "camminare" che il visitatore può effettuare su un sentiero di tipologia "bianco". Per poter essere sollevata l'azione  $Walk(?visitor\ ?from\ ?to)$  è necessario che il visitatore si trovi in  $?from$ , che quest'ultimo sia libero (non sul gommone), che esista un sentiero di tipologia "bianco" tra  $?from$  e  $?to$  e che non sia stato mai attraversato precedentemente. Se tutte le precondizioni appena descritte sono soddisfatte, allora il visitatore dopo l'esecuzione dell'azione si troverà in  $?to$  ed il luogo risulterà visitato, il tragitto non sarà più attraversabile in entrambi i sensi, dovuto dalla negazione dei seguenti predicati:  $(not\ (able-to-walk\ ?to\ ?from))$  e  $(not\ (able-to-walk\ ?from\ ?to))$ .

```
(:action Raft-Ride
  :parameters (?visitor ?raft ?from ?to)
  :precondition
    (and (river-road ?from ?to)
         (at ?raft ?from)
         (visitor ?visitor)
         (mobile ?raft)
         (riding ?visitor ?raft)
         (able-to-ride ?from ?to)
         (not (= ?from ?to))
        )
  :effect
    (and (at ?raft ?to)
         (not (to-discover ?to))
         (not (at ?raft ?from))
         (not (able-to-ride ?from ?to))
         (not (able-to-ride ?to ?from))
        )
  )
)
```

Listing 4.2: Azione del giro in gommone.

L'azione  $Shadow-Walk(?visitor\ ?from\ ?to)$  viene attivata quando il visitatore deve attraversare un sentiero "ombreggiato"; l'implementazione segue



la stessa logica dell'azione  $Walk(?visitor\ ?from\ ?to)$  mostrata in Listing 4.1, eccetto che per la preconditione ( $shadow\ ?from\ ?to$ ), per la quale deve esistere un sentiero di tipologia "ombreggiato" tra  $?from$  e  $?to$  e come conseguenza la non negazione del predicato ( $able-to-walk\ ?from\ ?to$ ) e ( $able-to-walk\ ?to\ ?from$ ) rendendo dunque il tragitto percorribile un numero illimitato di volte in entrambi i sensi.

Per formalizzare il vincolo "effettuare una volta sola l'attraversamento del fiume", è stato necessario introdurre nel dominio una tipologia di sentiero percorribile solo da entità di tipo gommone; quest'ultimo può essere usato dal visitatore per raggiungere i luoghi collegati. In ogni punto di entrata del sentiero il visitatore può salire sul gommone attraverso l'azione  $Get-on-Raft(?visitor\ ?place\ ?raft)$ , navigare con  $Raft-Ride(?visitor\ ?raft\ ?from\ ?to)$  e infine scendere attraverso l'azione  $Get-off-Raft(?visitor\ ?place\ ?raft)$ . In questo senso possiamo notare che tra le preconditioni dell'azione  $Raft-Ride(?visitor\ ?raft\ ?from\ ?to)$  in Listing 4.2, il visitatore deve essere fisicamente sul gommone ( $riding\ ?visitor\ ?raft$ ), oltre ad essere nello stesso luogo ovvero trovarsi in  $?from$ . Dopo l'esecuzione di questa azione il gommone e il visitatore si troveranno in  $?to$ , il luogo risulterà visitato e il tragitto non sarà più attraversabile in entrambi i sensi. Il visitatore può concludere la navigazione del fiume scendendo dal gommone attraverso l'azione  $Get-off-Raft(?visitor\ ?place\ ?raft)$  che logicamente separa l'entità  $?raft$  da  $?visitor$ .

## 4.5 La generazione del problema

Una volta formalizzato il modello è stato necessario sviluppare una componente essenziale che permettesse di generare il problema per il dominio descritto nel paragrafo precedente. Va infatti considerato che uno dei principali requisiti di questo sistema è il costante apporto di nuovi problemi da presentare al paziente in modo che sia virtualmente impossibile che un paziente risolva due volte lo stesso.

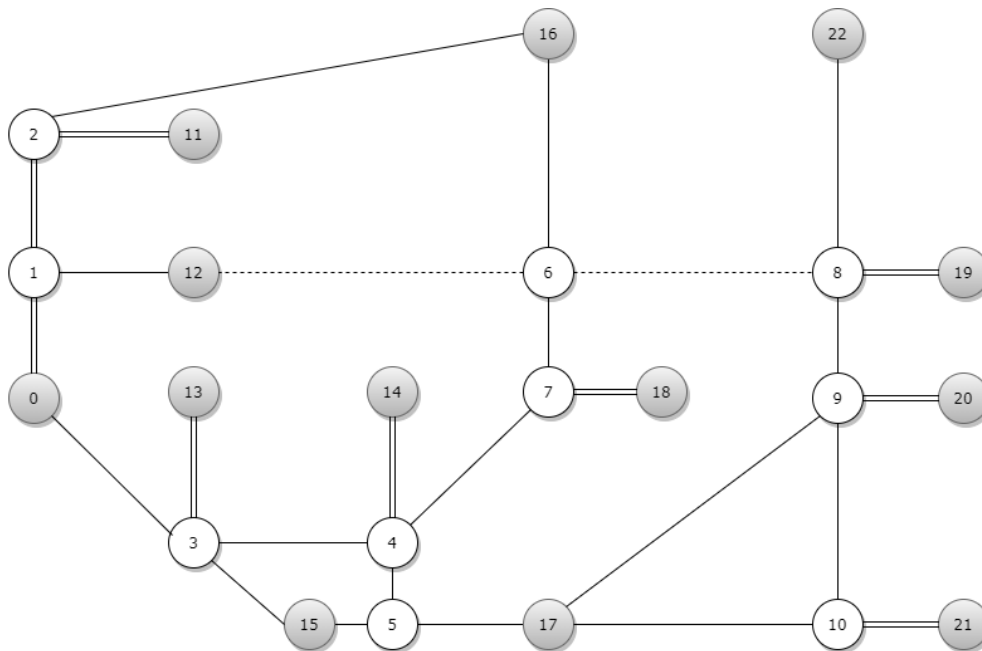


Figura 4.2: Grafo rappresentante lo zoosafari.

Per generare una possibile istanza del problema si è fatto uso di tecnologie Java, con l'aiuto di PDDL4J, una libreria open source con licenza LGPL il cui scopo è quello di facilitare lo sviluppo di strumenti per la pianificazione automatizzata basata sul linguaggio PDDL [24].

In Figura 4.2 viene mostrato il grafo dello zoosafari, attraverso il quale è stato possibile schematizzare e quindi generare il problema con un approccio casuale. Nel progettare lo zoosafari abbiamo utilizzato un grafo congruente alla struttura della mappa dello zoo (stessi nodi, archi e collegamenti), pertanto cambiando il comparto grafico e seguendo le specifiche date in sezione 2.4 si può realizzare la mappa dello zoo originale, come è stato fatto in altra istanza. I nodi colorati sono le possibili posizioni delle gabbie degli animali, mentre i restanti sono semplicemente incroci o curve. Gli archi che collegano i nodi sono rappresentati da tre tipi di linea, tante quante le tipologie di sentiero: la linea singola rappresenta il sentiero "bianco", la doppia linea quello "ombreggiato" ed infine la linea tratteggiata indica l'attraversamento

in fiume.

Il generatore definisce una istanza del problema in base alla difficoltà scelta (facile, medio e difficile) definendo: la struttura della mappa (nodi e archi), gli stati iniziali (disponendo in maniera del tutto random le gabbie degli animali, l'entrata e l'uscita nei nodi colorati) e gli stati goal (generalmente un sottoinsieme tra quelli posizionati). Una volta generato il problema viene analizzato dal solver (PDDL4J) che determina se esiste una sequenza di azioni tale per cui tutti gli stati goal risultino visitati, nel caso in cui non esista una soluzione allora viene generato un nuovo problema fino a quando quest'ultimo non presenti una soluzione. Questo approccio assicura la massima casualità nella posizione delle gabbie senza però rinunciare alla coerenza.

Inoltre, i problemi possono essere generati in 3 categorie (facile, medio, difficile) per complessità sulla base del numero di obiettivi che l'utente deve completare, con un piccolo incremento correlato al numero di animali visibili sulla mappa e alla presenza di una percentuale minore di sentieri "ombreggiati".

## 4.6 La grafica

Per il comparto grafico si è scelto di utilizzare una tecnica conosciuta con il termine *tile-based*, largamente utilizzata nello sviluppo di giochi 2D; quest'ultima consiste nel costruire il mondo di gioco partendo da piccole immagini di forma regolare chiamate *tiles* [1]. Questo tipo di tecnica permette un guadagno in termini di memoria utilizzata e performance, in quanto non è necessario caricare immagini di grandi dimensioni contenenti l'intera mappa di gioco, ma essa viene generata mediante l'affiancamento di tiles in una struttura regolare a griglia chiamata *tilemaps*. Essendo il sistema MS-rehab una web application è stato necessario sviluppare tale tecnica con JavaScript, Canvas e HTML5 per sfruttare al meglio le differenti tecnologie necessarie a progettare un software con una così preponderante componente grafica. Un altro motivo per cui si è scelta tale tecnica ricade nella possibilità di map-

pare l'intera area di gioco su una griglia logica che, può essere manipolata all'interno del codice, gestendo ad esempio le collisioni tra gli oggetti, l'individuazione del percorso effettuato e gli obiettivi completati. Sono numerosi i giochi che sono stati sviluppati utilizzando questa tecnica, possiamo citare a scopo esemplificativo Super Mario Bros, Pacman, Zelda: Link's Awakening, Starcraft, Sim City 2000, ecc.



Figura 4.3: Tilesheet utilizzata per il rendering grafico della mappa.

La Figura 4.3 è denominata *spritesheet* e contiene tutte le tiles (in questo caso sono 6x3 tiles di dimensione 64x64 pixel) utili a disegnare l'intera mappa di gioco con il tema dello zoosafari. Uno dei vantaggi legati all'utilizzo di una spritesheet risiede nella capacità di riuscire a mappare ogni tiles attraverso un indice, ottenendo così un identificatore. Ad esempio assegnando 0 alla prima tiles (erba), 1 alla seconda (sentiero ombreggiato), 2 alla terza tiles (sentiero bianco) e così via, avremo che ogni tiles è identificata da un numero. In questo modo possiamo renderizzare la mappa di gioco attraverso una matrice regolare, avente in ogni posizione un numero compreso tra 0 e 17.

Inoltre la matrice consente di definire l'area di gioco anche da un punto di vista formale; le interagibilità tra le zone vengono circoscritte mediante una mappa in cui la chiave è rappresentata da un indice e il valore da una proprietà.

Queste ultime possono essere di tre tipi:

- area non calpestabile (erba): 0;
- area calpestabile (sentiero ombreggiato): 1;
- area transitabile una sola volta (sentiero bianco e fiume): 2, 5.

Di seguito mostriamo un'istanza di matrice generata a difficoltà difficile:

$$\text{HardMap} = \begin{bmatrix} 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 7 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 8 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 1 & 2 & 2 & 2 & 2 & 9 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 1 & 1 & 10 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 11 & 2 & 2 & 2 & 2 & 2 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 2 & 2 & 2 & 2 & 2 & 2 & 1 & 2 & 2 & 2 & 2 & 5 & 2 & 2 & 2 & 2 & 2 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 2 & 2 & 2 & 2 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 & 1 & 1 & 1 & 0 & 12 & 0 & 0 & 0 & 0 & 13 & 0 & 0 & 0 & 0 & 14 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 15 & 2 & 2 & 2 & 1 & 1 & 16 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 17 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Tale matrice viene istanziata dal generatore di problemi descritto nel paragrafo precedente, che si occupa di inserire gli animali in coordinate stabilite in base alla soluzione trovata dal pianificatore. In Figura 4.4 abbiamo una semplice rappresentazione della nostra area di gioco, o meglio il rendering grafico delle tiles definite nella matrice `HardMap` appena descritta.

Si è scelto di rendere la mappa statica, ovvero di disegnare completamente l'intera area di gioco (di dimensioni `800x608`) attraverso un ciclo annidato che scorre su colonne e righe della matrice, un caso molto comune nei giochi arcade come ad esempio Pacman, Arkanoid o Sokoban [18]; in questo modo non è stato necessario sviluppare un componente capace di scorrere per visualizzare parti di gioco. Le tilemaps a scorrimento mostrano solo una piccola parte del mondo alla volta. Possono seguire un personaggio come nei giochi platform o RPG o permettere al giocatore di controllare la telecamera, come nei giochi di strategia o di simulazione.

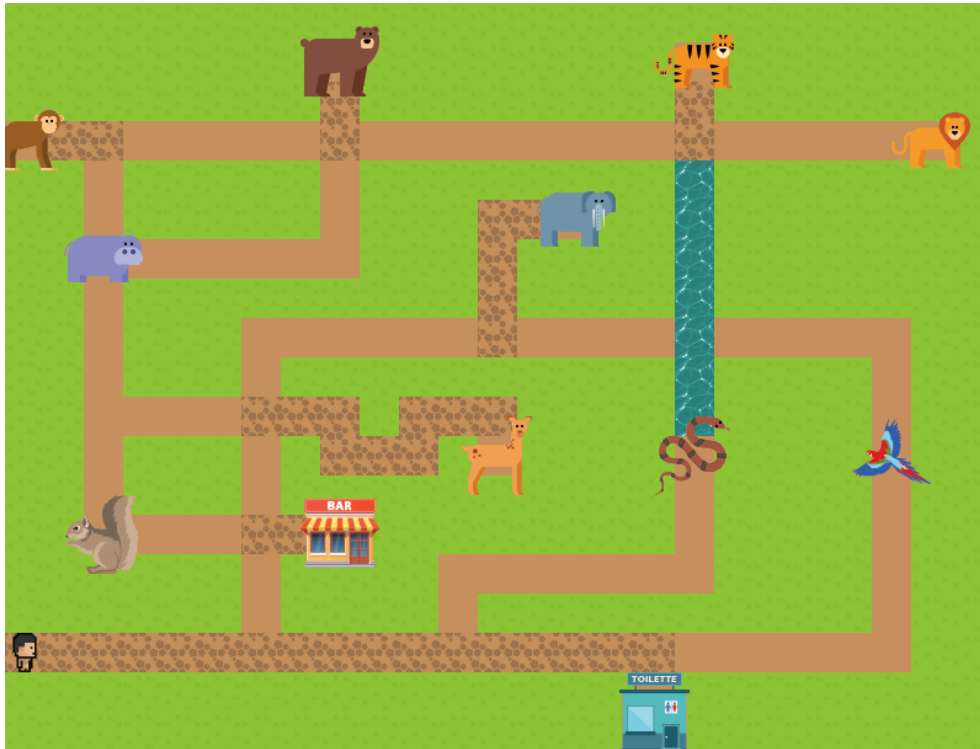


Figura 4.4: Rendering grafico della matrice HardMap.

L'interazione con la mappa avviene attraverso l'implementazione di un agente definito come *visitatore* capace di muoversi nei sentieri circoscritti mostrati nella figura precedente. Come per la mappa, anche per il visitatore è stato gestito il rendering grafico e la collisione con i vari oggetti, infatti quest'ultimo non può attraversare l'area di gioco definita dall'erba.

In Figura 4.5 sono rappresentate le tiles utilizzate per disegnare il viaggiatore sulla mappa, la prima tile viene utilizzata quando il viaggiatore prende una direzione verso l'alto, la seconda quando attraversa un sentiero verso il basso, la terza viene disegnata quando il viaggiatore si muove verso sinistra e la quarta quando si sposta verso destra. Le restanti due tiles sono utilizzate per rappresentare il viaggiatore che attraversa il fiume, la prima quando naviga il fiume verso sud e l'altra quando lo attraversa verso nord.



Figura 4.5: Tileset utilizzata per il rendering grafico del visitatore.

## 4.7 Confronto con la mattina di impegni

Volendo confrontare la metodologia utilizzata in zoosafari con la mattina di impegni [4], esercizio sviluppato da un mio collega per il sistema MS-rehab; in cui viene sfruttato il pianificatore per generare sia le mappe (ambiente credibile ma casuale con edifici, rotonde e oggetti) che il problema (data una mappa, genera un problema risolvibile in quella mappa), si possono fare le seguenti considerazioni.

In mattina di impegni si chiede al paziente di svolgere una serie di compiti, come ad esempio passare dal supermercato alle 15:30, comprare la carne, recarsi in piscina alle 16:30. Per valutare la correttezza di ogni singola mossa del paziente, ad ogni azione viene effettuata una chiamata al pianificatore con la descrizione del mondo attuale, in modo che se il generatore trova ancora una soluzione dopo l'esecuzione della mossa allora il paziente è libero di continuare l'esercizio altrimenti viene segnalata la sconfitta. Nel caso di zoosafari non si è voluto adottare la stessa metodologia di approccio al pianificatore come in mattina di impegni, ritenendo che questo potesse rendere l'esercizio lento e fastidioso.

Per quanto concerne il comparto grafico dell'esercizio mattina di impegni, i luoghi sono disegnati per mezzo di icone che esprimono ciascun elemento presente nella mappa (strada, rotonda, casa, banca, piscina, supermercato, farmacia, gelateria, stazione e discarica), la scarsa qualità di queste ultime rende la grafica semplicistica e riduttiva a differenza di zoosafari che ha seguito le linee guida per realizzare una grafica utilizzata tutt'oggi dai videogiochi in 2D.





# Capitolo 5

## Integrazione in MS-rehab

Dopo aver realizzato le componenti principali atte all'implementazione dell'esercizio, si è proceduto all'integrazione di quest'ultimo nel sistema riabilitativo MS-rehab. In questo capitolo descriveremo con un approccio ingegneristico le problematiche affrontate riguardo l'analisi dei requisiti rilevati dai casi d'uso, i diagrammi di sequenza, le classi introdotte per la realizzazione e la valutazione delle performance.

### 5.1 Casi d'uso

Per chiara identificazione dell'obiettivo, durante la fase di analisi è stato necessario la stesura di un diagramma dei casi d'uso per interpretare le interazioni tra l'attore ed il sistema MS-rehab; finalizzato a precisare meglio i requisiti espressi ed a chiarire quali componenti sarebbero stati necessari per lo sviluppo.

In Figura 5.1 sono mostrati i casi d'uso relativi all'attore paziente, questi verranno descritti nel dettaglio nel paragrafo sottostante.

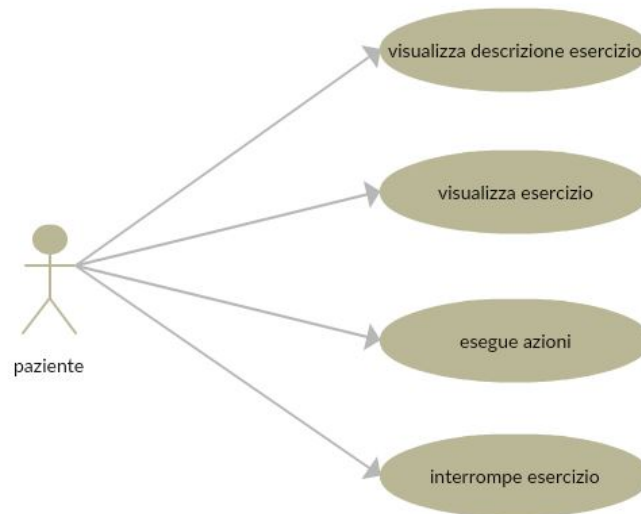


Figura 5.1: Diagramma dei casi d'uso.

---

**CASO D'USO:**

*Visualizza descrizione esercizio*

---

**Descrizione:**

*Consente al paziente autenticato di visualizzare la descrizione di un esercizio.*

**Attori:**

*Il paziente autenticato.*

**Precondizioni:**

*Nessuna.*

**Flusso:**

*Il paziente visualizza un nuovo esercizio con i vincoli e può procedere verso lo svolgimento.*

**Postcondizioni:**

*Nessuna.*

---

Tabella 5.1: Caso d'uso: visualizza descrizione esercizio.

---

<b>CASO D'USO:</b>	<i>Visualizza esercizio</i>
<b>Descrizione:</b>	<i>Il paziente autenticato visualizza l'ambiente dell'esercizio con il quale può interagire.</i>
<b>Attori:</b>	<i>Il paziente autenticato.</i>
<b>Precondizioni:</b>	<i>Il paziente deve aver visualizzato la descrizione dell'esercizio.</i>
<b>Flusso:</b>	<i>Il paziente visualizza il nuovo esercizio e può interagirvi.</i>
<b>Postcondizioni:</b>	<i>Nessuna.</i>

---

Tabella 5.2: Caso d'uso: visualizza esercizio.

---

<b>CASO D'USO:</b>	<i>Esegue azioni</i>
<b>Descrizione:</b>	<i>Il paziente autenticato esegue un'azione nello scenario proposto.</i>
<b>Attori:</b>	<i>Il paziente autenticato.</i>
<b>Precondizioni:</b>	<i>Il paziente deve aver visualizzato la descrizione dell'esercizio.</i>
<b>Flusso:</b>	<i>Il paziente muove con tastiera il visitatore sullo schermo per compiere una azione di movimento con l'intento di visitare i luoghi dati come obiettivi.</i>
<b>Postcondizioni:</b>	<i>Nessuna.</i>

---

Tabella 5.3: Caso d'uso: esegue azioni.

---

<b>CASO D'USO:</b>	<i>Interrompe esercizio</i>
--------------------	-----------------------------

---

<b>Descrizione:</b>	<i>Il paziente autenticato interrompe l'esercizio.</i>
<b>Attori:</b>	<i>Il paziente autenticato.</i>
<b>Precondizioni:</b>	<i>Il paziente deve aver avviato l'esercizio.</i>
<b>Flusso:</b>	<i>Il paziente autenticato può interrompere l'esercizio in qualsiasi momento senza penalità o al termine di quest'ultimo.</i>
<b>Postcondizioni:</b>	<i>Il sistema salva la performance del paziente nel caso l'esercizio sia stato completato o si sia raggiunta una situazione per cui non esiste più un percorso valido.</i>

---

Tabella 5.4: Caso d'uso: interrompe esercizio.

## 5.2 Diagramma di sequenza

Una volta formalizzate le funzioni che l'attore paziente deve poter effettuare sul sistema si è passati alla fase successiva della progettazione; ovvero la comprensione della sequenza di azioni degli attori coinvolti nell'approccio alla riabilitazione cognitiva. In figura 5.2 viene presentato una semplificazione dello schema di come è stato strutturato il sistema.

Come è possibile notare dallo schema, la sequenza parte dall'operatore sanitario che assegna un esercizio riabilitativo ad un gruppo o ad un singolo paziente. Quando il paziente richiede quindi l'esercizio che gli è stato assegnato scatena nel server una serie di operazioni che coinvolgono direttamente il generatore di problemi. L'esercizio viene inviato dal server, il paziente quindi si troverà a compiere azioni per concludere con successo o con fallimento l'esercitazione, una volta raggiunto l'esito il sistema salverà il risultato ottenuto nel database. Inoltre dopo aver completato l'esercizio riabilitativo al paziente sarà mostrato un esercizio di difficoltà successiva consono al risultato ottenuto.

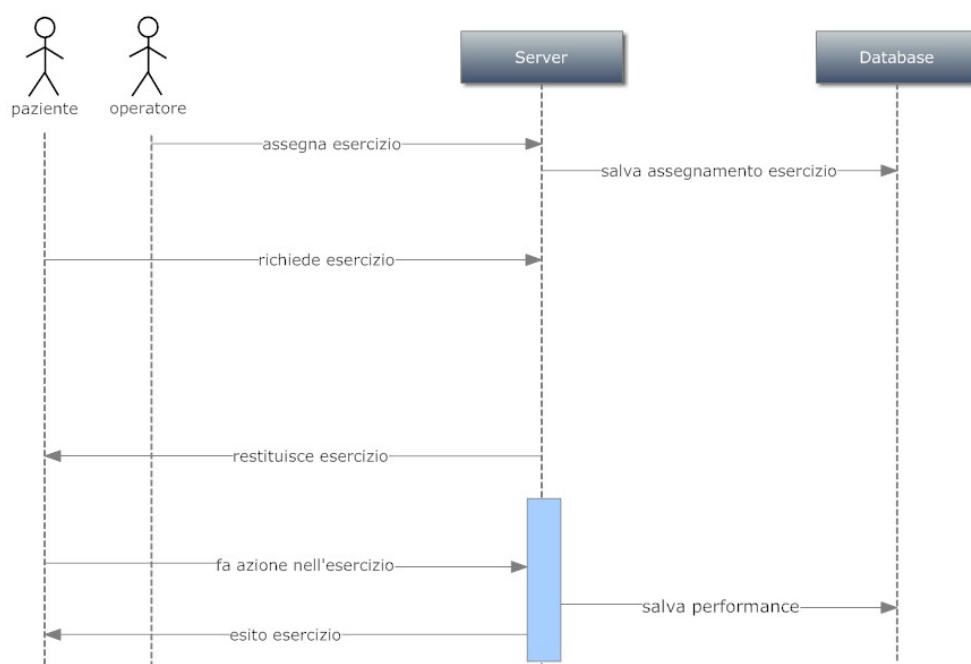


Figura 5.2: Diagramma di sequenza.

### 5.3 Struttura dell'integrazione

Zoosafari è stato realizzato seguendo le linee guida per produrre un software di qualità, infatti per garantire una facile manutenzione del codice è stato strutturato in modo da facilitare la ricerca degli errori, l'aggiunta di nuove funzionalità al sistema e l'adattamento ai cambiamenti del dominio applicativo. Inoltre l'integrazione dimostra che il progetto zoosafari è interoperabile poiché abile a cooperare con altri sistemi, nel caso specifico MS-rehab. Il comparto grafico sviluppato in HTML5 e Canvas ha reso il software portabile, ovvero in grado di funzionare su più piattaforme hardware/software. Infine quest'ultimo è riusabile, perché può essere usato, in tutto o in parte, per costruire un nuovo sistema.

Prima di entrare nei dettagli implementativi di zoosafari lato server, sarebbe opportuno descrivere il processo di generazione sviluppato per creare uno scenario conforme all'analisi della conoscenza, tutte le componenti atte

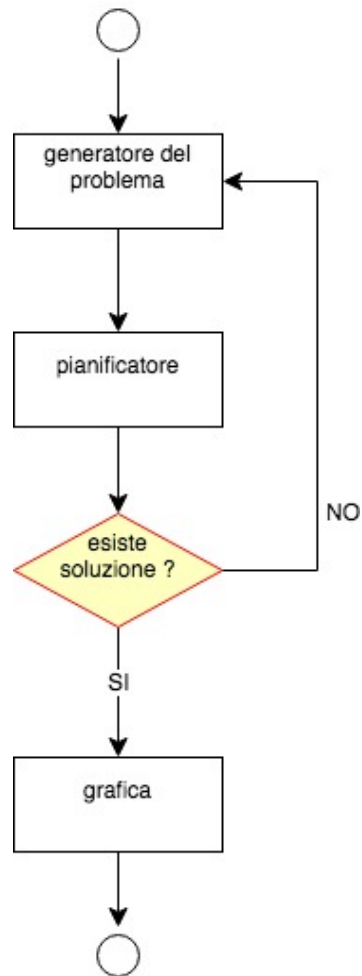


Figura 5.3: Processo di creazione scenario.

alla realizzazione di zoosafari sono descritte nel dettaglio nel capitolo precedente. In Figura 5.3, viene mostrato il diagramma UML con le principali componenti realizzate al fine di generare una istanza di zoosafari congruente agli obiettivi da visitare e ai vincoli da rispettare. Il processo parte con la generazione del problema in cui viene definita un'istanza di zoosafari in base alla difficoltà richiesta dal sistema MS-rehab, questa ha il compito di definire la struttura della mappa, gli stati iniziali (le gabbie degli animali e l'uscita vengono disposte in maniera casuale) e gli stati goals (gli animali che devono

essere visitati al fine di superare l'esercizio). Una volta che si è conclusa la fase di generazione del problema è compito del pianificatore effettuare il controllo sulla mappa generata, ossia verificare se esiste un cammino tale per cui il visitatore sia in grado di visitare tutti gli obiettivi e concludere il percorso nel luogo stabilito. Nel caso non esista una soluzione allora il processo riparte dal generatore di problemi, generando una nuova configurazione di mappa per poi essere testata nuovamente dal pianificatore, fino a quando non si trova una configurazione avente almeno una soluzione. Dopo aver superato il controllo del pianificatore il processo passa al comparto grafico, il componente che si occupa di renderizzare l'intera area di gioco il quale permette l'interazione tra il software e il paziente con lo scopo per quest'ultimo di compiere azioni al fine di superare l'esercizio. Oltre a graficare gli animali, i sentieri e il viaggiatore, questo componente gestisce anche la logica di gioco come ad esempio le collisioni tra gli oggetti, gli obiettivi completati e i vincoli riguardo: l'attraversamento dei sentieri "bianchi" e la singola navigazione del fiume. Quando il paziente termina l'esercizio con un punteggio superiore all'80% vengono aggiornate delle variabili d'ambiente usate nel generatore del problema per aumentare la difficoltà dell'esercizio che gli è stato assegnato, in modo che risulti sempre impegnativa la sfida riabilitativa stimolando così il recupero neuropsicologico.

La struttura di zoosafari lato server può essere agilmente descritta da un diagramma delle classi (Figura 5.4) che ne mette in luce punti di forza e principali limiti. Il sistema può essere considerato un'entità separata dal resto del programma MS-rehab in quanto è sufficiente importare la classe *PDDLGeneratorZooMap* per avere tutte le funzionalità del sistema. La classe principale espone tre metodi statici ognuno dei quali si occupa di risolvere, con l'aiuto di una istanza del solver (PDDL4J), tre differenti problemi tante quante sono le diverse tipologie di mappa che possono essere richieste dal sistema in base alla classe di difficoltà (facile, medio e difficile). Ciascun metodo statico della classe *PDDLGeneratorZooMap* istanzia un oggetto di tipo *ZooMap* che va a dichiarare una serie di parametri utili a definire il

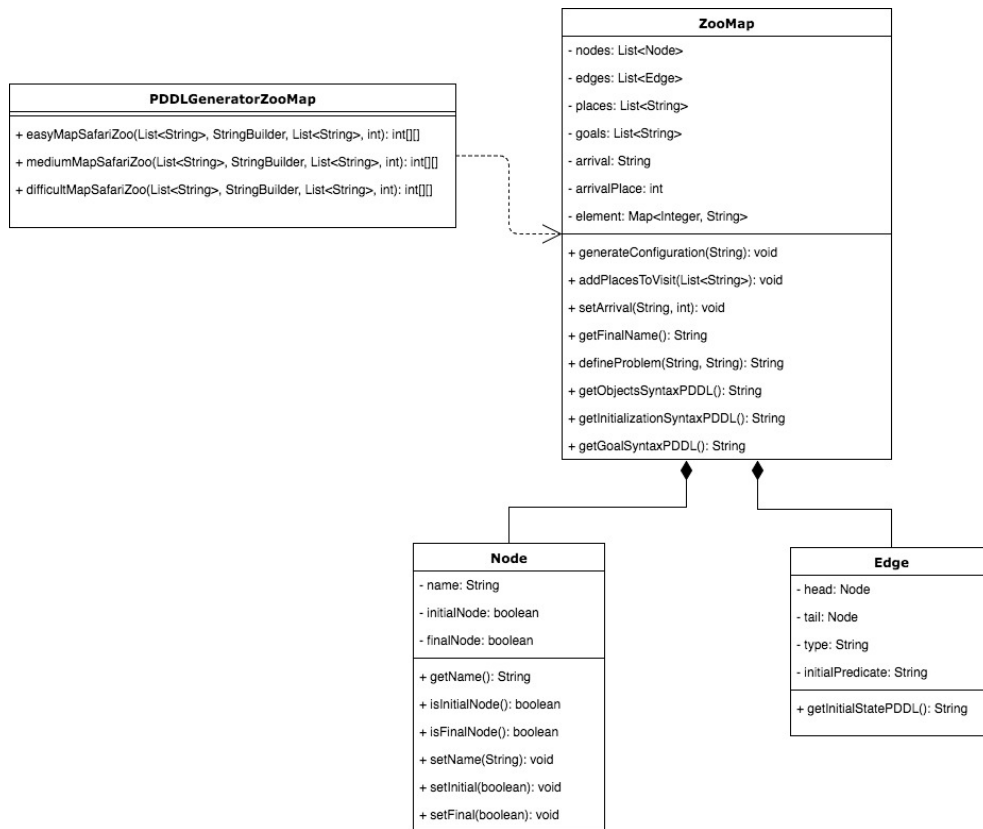


Figura 5.4: Diagramma delle classi del sistema.

problema (la struttura della mappa, gli stati iniziali e gli stati obiettivo) in linguaggio PDDL.

La classe *ZooMap* funge da generatore del problema (definito nel paragrafo 4.5 del capitolo precedente) ed è composto da una serie di oggetti descritti nell'elenco sottostante.

**nodes** una lista di oggetti di tipo nodo;

**edges** una lista di archi utili a collegare i nodi;

**places** una lista di stringhe contenenti i nomi degli animali che si vogliono posizionare sul grafo;



**arrival** il nome dell'animale in cui il giocatore è obbligato a visitare come ultima tappa per completare il gioco;

**arrivalPlace** l'identificativo del nodo in cui si vuole far posizione l'oggetto *arrival*;

**element** un dizionario in cui la chiave identifica il nodo e il valore corrisponde all'animale posizionato.

```
do {  
...  
    String PROBLEM_FILE = "";  
  
    ZooMap map = new ZooMap();  
  
    map.setArrival(inFinalPlace.toString(), inFinalIndex);  
    map.addPlacesToVisit(listGoals);  
    map.generateConfiguration("difficult");  
  
    PROBLEM_FILE += map.defineProblem("zoosafari",  
        PROBLEM_NAME);  
    PROBLEM_FILE += map.getObjectsSyntaxPDDL();  
    PROBLEM_FILE += map.getInitializationSyntaxPDDL();  
    PROBLEM_FILE += map.getGoalSyntaxPDDL();  
...  
} while (!isPlanned);
```

Listing 5.1: Generazione del problema attraverso l'istanza di ZooMap.

Nella parte di codice mostrata in Listing 5.1 viene istanziato un oggetto di classe *ZooMap*, quest'ultimo attraverso i metodi messi a disposizione va a settare il luogo in cui si vuole far terminare l'esercizio, aggiunge gli obiettivi da visitare sulla mappa e con il metodo *generateConfiguration("difficult")* va a congelare la struttura del grafo(nodi e archi) dettati dalla configurazione

difficile di zoosafari, posizionando gli animali in maniera random sui nodi prestabiliti (Figura 4.4). I restanti metodi sono utili a tradurre le informazioni in linguaggio PDDL comprensibili al solver; una volta generata l'intera stringa *PROBLEM\_FILE* viene data in input al solver e se quest'ultimo non riesce a trovare un cammino nell'insieme degli stati attraverso il quale l'utente riesca a visitare gli stati obiettivo, allora si ripete l'iter andando a generare una nuova configurazione delle posizioni degli animali sulla mappa.

## 5.4 Valutazione delle performance

Una volta che il paziente ha terminato l'esercizio, il sistema assegna un punteggio in base alla performance raggiunta durante lo svolgimento, con lo scopo di valutare i progressi o le ricadute delle prestazioni del soggetto. MS-rehab dispone di una sezione che fornisce grafici riguardo l'andamento delle prestazioni dei pazienti utili all'équipe di dottori, mirata a tenere monitorata la terapia in corso e valutare eventuali decisioni riguardo il quantitativo di esercizi assegnati. Non è semplice decidere una metrica di valutazione in un esercizio di pianificazione, in quanto la soluzione è arbitrariamente complessa e inoltre possono coesistere differenti soluzioni ma tutte isomorfe nel numero dei passi che portano al raggiungimento degli obiettivi dati. MS-rehab propone un modello formale di calcolo per gli esercizi di planning che valuta cinque componenti fondamentali:

**pC** numero di azioni corrette fatte dal paziente;

**pW** numero di azioni errate eseguite durante l'esercizio;

**pM** numero di azioni mancanti per completare l'esercizio;

**pT** il tempo in secondi impiegato per risolvere l'esercizio;

**maxT** il tempo massimo consentito per svolgere l'esercizio.

Definiamo  $O$  l'insieme delle azioni generate dal solver, viene considerata come la soluzione ottima che permette di soddisfare tutti gli obiettivi dell'esercizio preso in considerazione e  $A$  invece l'insieme delle azioni eseguite dal paziente durante lo svolgimento dell'esercizio. Se il paziente ha completato l'esercizio allora vuol dire che ha eseguito una sequenza di azioni che ha permesso di visitare tutti gli obiettivi, il numero di mosse potrebbero essere uguali al numero di azioni generate dal solver e ciò vuol dire che il paziente ha eseguito una soluzione ottima (se  $|A| = |O|$ ) e allora  $pC = |A|$ ,  $pW = 0$  e  $pM = 0$ . Per tutti gli altri casi invece assegno alle componenti atte alla valutazione della performance come segue:

$$\begin{aligned} pC &= |O| \\ pW &= |A| - |O| \\ pM &= 0 \end{aligned}$$

Caso in cui il paziente non è stato in grado di superare l'esercizio i coefficienti vengono inizializzati a zero e poi aggiornati come segue:

$$\begin{aligned} \forall a \in A \mid a \in O &\implies pC = pC + 1 \\ \forall a \in A \mid a \notin O &\implies pW = pW + 1 \\ \forall o \in O \mid o \notin A &\implies pM = pM + 1 \end{aligned}$$

Una volta valorizzate tutte le variabili vengono date in input ad una F-measure utile a misurare l'accuratezza che il paziente ha ottenuto nell'esercizio, tenendo in considerazione precisione (5.1), recupero (5.2) e il tempo (5.3) impiegato a terminare quest'ultimo; il risultato di tale funzione è la performance che ha ottenuto il paziente. Per chiarire il modello usato in MS-rehab che si occupa del calcolo delle performance, definiamo le componenti utili a calcolare la F-measure [20]:

$$p = \frac{pC}{pC + pW} \quad p \in [0, 1] \quad (5.1)$$

$$r = \frac{pC}{pC + pM} \quad r \in [0, 1] \quad (5.2)$$

$$t = \frac{\max T - pT}{\max T} \quad \max T > 0, \quad t \in [0, 1] \quad (5.3)$$

Il sistema tiene in considerazione di una soglia che indica la performance minima che il paziente deve ottenere per superare l'esercizio preso in considerazione e viene indicata con  $thr$ . Viene definita la variabile ausiliaria  $t'$  in modo da rendere più chiara la formula finale.

$$t' = (1 - thr) * t + thr \quad thr \in [0, 1], \quad t' \in [thr, 1]$$

Ora che abbiamo tutte le componenti necessarie possiamo definire la performance come:

$$\text{performance} = \left( (1 + \beta^2) * \left( \frac{p * r}{\beta^2 * p + r} \right) * t' \right) \quad \beta > 0, \quad \text{performance} \in [0, 1]$$

Come si può banalmente intuire dall'equazione appena definita, se un paziente completa l'esercizio senza commettere errori entro il tempo massimo consentito allora ottiene una performance di almeno  $thr$ . I parametri  $thr$  e  $\beta$  sono utili a rendere più o meno semplice il superamento dell'esercizio, nel nostro caso è stato impostato  $thr$  a 0,8 e  $\beta$  uguale a 1 per dare la stessa importanza alla precisione e al recupero.

La formula della performance viene poi normalizzata con un valore che riporta il punteggio massimo in base alla classe di difficoltà, per esempio un esercizio svolto in difficoltà facile può ottenere un punteggio che va da 0 a 33, per quello medio da 0 a 66 e per un esercizio svolto in modalità difficile da 0 a 100. Come si è mostrato la formula prende in esame le componenti fondamentali per una corretta rappresentazione delle performance, il tempo infatti gioca un ruolo chiave nelle disfunzioni esecutive e un miglioramento nei tempi di risposta rappresenta un grande miglioramento per la vita del paziente; inoltre permette di disegnare grafici precisi sullo stato riabilitativo del paziente e monitorare nel corso del tempo le sue performance.

# Conclusioni

L'obiettivo di questa tesi è stato la progettazione e quindi la realizzazione di un esercizio destinato alla riabilitazione delle funzioni esecutive dei pazienti affetti da sclerosi multipla, da inserire nel sistema riabilitativo MS-rehab.

Le funzioni esecutive sono una componente essenziale da riabilitare nei pazienti affetti da sclerosi multipla, in quanto un deficit in questa area compromette gli individui nelle attività quotidiane, nel campo scolastico, sociale e lavorativo. Vari studi presenti in letteratura dimostrano che è possibile riabilitare i deficit delle funzioni esecutive mediante la terapia cognitiva, attraverso esercizi cognitivi-funzionali finalizzati al potenziamento delle abilità neuropsicologiche coinvolte [25, 30]. Nel caso specifico, si tratta di esercizi che riguardano le componenti di problem-solving e di pianificazione attraverso compiti complessi, con buone probabilità di generalizzazione nei diversi contesti di vita.

Il sistema MS-rehab prevedeva soltanto due esercizi per la riabilitazione delle funzioni esecutive, un esercizio sul controllo dell'inibizione e un esercizio ecologico "mattina di impegni" su aspetti di pianificazione [4] dove viene richiesto al paziente di svolgere una serie di compiti come ad esempio uscire di casa, recarsi in banca per prelevare del denaro, andare a fare la spesa indicando degli orari ai quali svolgere tali attività. Si è scelto di arricchire il sistema con un nuovo esercizio ecologico di pianificazione denominato visita allo zoosafari, una generalizzazione di un esercizio denominato la mappa dello zoo che fa parte della batteria di test di valutazione sulle funzioni esecutive denominata BADS [31].

In questo nuovo esercizio si chiede al paziente di pianificare un percorso attraverso una mappa che rappresenta uno zoo safari, con l'obiettivo di visitare una serie di animali, posti in punti specifici, rispettando alcuni vincoli come ad esempio sentieri che possono essere transitati una sola volta, il luogo in cui terminare la visita e la possibilità di effettuare una singola volta la navigazione del fiume sul gommone.

La principale caratteristica dell'esercizio realizzato è la presenza di contenuti dinamici all'interno della mappa, infatti ad ogni istanza dell'esercizio gli animali vengono collocati nelle posizioni stabilite in maniera del tutto random sulla mappa, allo stesso modo per gli obiettivi da visitare e il luogo in cui finire il percorso. Questa caratteristica permette il costante apporto di nuovi scenari in modo che sia virtualmente impossibile che si debba risolvere due volte lo stesso. Inoltre zoosafari si adatta al paziente, poiché in base alla sua abilità il sistema sceglie la difficoltà di quest'ultimo (facile, medio o difficile), attraverso un modello dinamico che si prefigge di far riabilitare il paziente in base al suo livello soglia. La realizzazione è stata possibile mediante l'uso della pianificazione automatica, con il quale è stato possibile effettuare il controllo dell'esistenza di una soluzione per raggiungere gli obiettivi selezionati casualmente (gli animali da visitare e il luogo in cui concludere il percorso) dal sistema. Il pianificatore viene utilizzato, quindi, solo nella fase di controllo della mappa con la configurazione generata a priori degli animali, prima che quest'ultima passi al comparto grafico. Relativamente a "mattina di impegni" in cui per valutare la correttezza di ogni singola mossa del paziente viene effettuata una chiamata al pianificatore per controllare se esiste ancora una soluzione.

Secondo l'analisi che abbiamo effettuato circa i software disponibili per la riabilitazione delle funzioni esecutive sulla tematica della pianificazione [12] risultano forniti soltanto alcuni scenari predefiniti e a nostra conoscenza nessuno di questi sistemi ha utilizzato tecniche di pianificazione.

La soluzione proposta è stata sottoposta ad una psicologa esperta in riabilitazione cognitiva e si è dimostrata adeguata agli scopi riabilitativi trat-

tando uno scenario reale, con vincoli utili di difficoltà crescente. L'esercizio sviluppato si presenta come un prodotto completo ed è stato integrato in MS-rehab per essere utilizzato con i pazienti in un imminente studio premarket multi-centro progettato per verificare l'efficacia di MS-rehab e quindi anche dell'esercizio stesso. Questo anche con il fine di ottenere la marcatura CE.

Per quanto riguarda gli sviluppi futuri, si aspettano i primi dati dell'imminente esperimento al fine di verificarne l'efficacia sui pazienti e di realizzare alcuni eventuali aggiustamenti. Inoltre si può pensare di migliorare l'esercizio introducendo funzionalità aggiuntive come ad esempio: un suggeritore che presenti, al paziente in difficoltà, il prossimo tratto da percorrere per avvicinarsi alla soluzione.

Un'esenzione possibile del progetto è l'utilizzo di un generatore di mappe in cui si sfrutta la pianificazione automatica per creare un ambiente consono, con la possibilità di gestire una maggiore quantità di scenari grafici come ad esempio zoopark, zoo acquatico, zoofarm ed altri affini.





# Bibliografia

- [1] Ashar, Florian Scholz, Teoli, Chris Mills, Belén Albeza. Tiles and tilemaps overview. Resources for developers by Mozilla. <https://developer.mozilla.org/en-US/docs/Games/Techniques/Tilemaps>
- [2] Atkinson, R.C., Shiffrin, R. M. (1968), Human Memory: A Proposed System and its Control Processes, in K.W. Spence, J.T. Spence (Eds), The Psychology of learning and motivation, Vol. 2, New York, Academic Press.
- [3] Baddeley, A. (1992). Working memory. *Science*, 255(5044), 556-559.
- [4] D. Baschieri (2017). Riabilitazione cognitiva delle funzioni esecutive nella Sclerosi Multipla: un approccio basato sul planning. Tesi LM, Università di Bologna, Scuola di Scienze, Corso di laurea in Informatica.
- [5] R. H. Benedict, D. Cookfair, R. Gavett, M. Gunther, F. Munschauer, N. Garg, and B. Weinstock-Guttman. Validity of the minimal assessment of cognitive function in multiple sclerosis (MACFIMS). *J Int Neuropsychol Soc*, 12(4):549-558, Jul 2006.
- [6] S. Bonavita, R. Sacco, M. Della Corte, S. Esposito, M. Sparaco, A. d'Ambrosio, R. Docimo, A. Biseco, L. Lavorgna, D. Corbo, S. Cirillo, A. Gallo, F. Esposito, and G. Tedeschi. Computer-aided cognitive rehabilitation improves cognitive performances and induces brain functional

- connectivity changes in relapsing remitting multiple sclerosis patients: an exploratory study. *J. Neurol.*, 262(1):91-100, Jan 2015.
- [7] C. Caltagirone, Neurodegenerazione, la morte dei neuroni, le malattie neurodegenerative, *Il Libro dell'Anno 2005*, Istituto dell'Enciclopedia italiana Treccani.
- [8] A. Compston and A. Coles. Multiple sclerosis. *Lancet*, 372(9648):1502-1517, Oct 2008.
- [9] J. "Funke and T." Krüger. "Plan-A-Day": Konzeption eines modifizierbaren Instruments zur Führungskräfte-Auswahl sowie erste empirische Befunde. "J. Funke & A. Fritz (Eds.) *Neue Konzepte und Instrumente zur Planungsdiagnostik*", pages 97-120, 1995.
- [10] Garg, M. K., Sharma, P., & Kirmani, M. N. (2016). Neuroplasticity of the Brain : Neurophysiological Perspective, 3(4).
- [11] Mauro Gaspari, Floriano Zini, Debora Castellano, Federica Pinardi, and Sergio Stecchi. An advanced system to support cognitive rehabilitation in multiple sclerosis. In *Proceeding of IEEE RTSI 2017, 3° International Forum on Research and Technologies for Society and Industry*. IEEE Press & IEEE Xplore digital library, 2017.
- [12] Hasomed. (2012). Cognitive therapy and brain training: Plan A Day.
- [13] Hosenbocus, S., & Chahal, R. (2012). A Review of Executive Function Deficits and Pharmacological Management in Children and Adolescents. *Journal of the Canadian Academy of Child and Adolescent Psychiatry* (Vol. 21).
- [14] Kroll, J., Karolis, V., Brittain, P. J., Tseng, C. J., Froudish-Walsh, S., Murray, R. M., & Nosarti, C. (2017). Real-life impact of executive function impairments in adults who were born very preterm. *Journal of the International Neuropsychological Society*, 23(5), 381-389.

- 
- [15] La Placa, M., Pigot, H., & Kabanza, F. (2009). Assistive planning for people with cognitive impairments. In Proc. of Workshop on Intelligent Systems for Assisted Cognition hosted by Int'l Joint Conference on Artificial Intelligence (IJCAI).
- [16] E. Làdavas. La riabilitazione neuropsicologica. 2012.
- [17] Lezak, M. D. (2004). Neuropsychological assessment. Oxford University Press, USA.
- [18] Chris Mills, Belén Albeza. Square tilemaps implementation: Static maps. Resources for developers by Mozilla. [https://developer.mozilla.org/en-US/docs/Games/Techniques/Tilemaps/Square\\_tilemaps\\_implementation:\\_Static\\_maps](https://developer.mozilla.org/en-US/docs/Games/Techniques/Tilemaps/Square_tilemaps_implementation:_Static_maps).
- [19] McDermott, D., Ghallab, M., Howe, A., & C. (1998). PDDL - The Planning Domain Definition Language. The AIPS-98 Planning Competition Committee, (December 2012), 27.
- [20] Manning, C. D., Raghavan, P., & Schütze, H. (2008). Introduction to Information Retrieval. Journal of the American Society for Information Science and Technology, 1, 496.
- [21] Mondini S., Mapelli D., Arcara G. La valutazione neuropsicologica. Carrocci Faber (2009).
- [22] Nau, D. S. (2007). Current Trends in Automated planning. AI Magazine, 28(4), 43.
- [23] Passafiume, D., & Di Giacomo, D. (2006). La demenza di Alzheimer. Guida all'intervento di stimolazione cognitiva e comportamentale. Franco Angeli.
- [24] Pellier, D., & Fiorino, H. (2018). PDDL4J: a planning domain description library for java. Journal of Experimental & Theoretical Artificial Intelligence, 30(1), 143-176.

- 
- [25] Prosperini L, De Giglio L, Pozzilli C, Pantano P, Bianchi V, Borriello G et al. A Low-Cost Cognitive Rehabilitation With a Commercial Video Game Improves Sustained Attention and Executive Functions in Multiple Sclerosis: A Pilot Study. *Neurorehabilitation and Neural Repair*. 2014.
- [26] Stuart J. Russel e Peter Norving. Capitolo 10: Pianificazione Classica, "Intelligenza artificiale: Un approccio moderno, 3<sup>rd</sup> edizione", Pearson, 2010.
- [27] Shallice, T., & Burgess, P. W. (1991). Deficits in strategy application following frontal lobe damage in man. *Brain*, 114(2), 727-741.
- [28] Sohlberg, McKay Moore, and Catherine A. Mateer. *Cognitive rehabilitation: An integrative neuropsychological approach*. Guilford Publications, 2017.
- [29] Sumowski, J. F., Rocca, M. A., Leavitt, V. M., Dackovic, J., Mesaros, S., Drulovic, J., Filippi, M. (2014). Brain reserve and cognitive reserve protect against cognitive decline over 4.5 years in MS. *Neurology*, 82(20), 1776-1783.
- [30] A. Tacchino, L. Pedulla, L. Bonzano, C. Vassallo, M. A. Battaglia, G. Mancardi, M. Bove, and G. Bricchetto. A New App for At-Home Cognitive Training: Description and Pilot Testing on Patients with Multiple Sclerosis. *JMIR Mhealth Uhealth*, 3(3):e85, Aug 2015.
- [31] Wilson, B. A., Evans, J. J., Alderman, N., Burgess, P. W., & Emalie, H. (1997). Behavioural assessment of the dysexecutive syndrome. *Methodology of frontal and executive function*, 239-250.
- [32] U.S. Department of Health and Human Services. *Attention Deficit Hyperactivity Disorder (ADHD)*. Health & Outreach, revisited 2016 edition, 2009.

# Ringraziamenti

Arrivato a questo punto conclusivo della mia carriera universitaria sento il desiderio di ringraziare tutti coloro che mi hanno aiutato e sostenuto durante questo percorso.

Un primo ringraziamento va al mio relatore Prof. Mauro Gaspari, per avermi concesso l'opportunità di lavorare al suo fianco su un progetto di questo calibro, per avermi considerato e appoggiato su tutte le mie idee riguardo il lavoro di tesi, per avermi fatto appassionare alla IA e alla Health informatics.

Un grande ringraziamento va alla mia famiglia: Vito, Teresa, Mariagrazia e Giusy, che con il loro instancabile sostegno morale mi hanno permesso di arrivare fin qui contribuendo alla mia formazione personale.

Grazie alla mia amica nonché collega Cristina Lelli per i numerosi consigli di carattere informatico durante il periodo di ricerca, per aver contribuito con le sue doti grafiche e per avermi sopportato a lavoro rispondendo a miei dilemmi esistenziali.

Grazie alle mie amiche Claudia Geraci e Stefania Lombardi per tutte le volte che mi hanno consigliato riguardo la stesura di questo elaborato.

Ringrazio tutti i miei colleghi del servizio Sistemi Informativi dell'Agenzia sanitaria e sociale regionale della Regione Emilia-Romagna: Alessandro M., Chiara B., Giuliana G., Martino F., Marco M., Sauro S. e il mio capo Stefano A.

Ho desiderio di ringraziare con affetto anche tutti i miei compagni di corso: Andrea S., Amerigo M., Davide A., Enrico V., Gabriele G., Gianluca

M., Lorenzo L., Margherita D., Martin C., Matteo C., Maxim G., Simone P., Simone S., Stefano T. e in generale tutti gli amici conosciuti durante questo periodo per avermi permesso di condividere un'esperienza unica e indimenticabile.

Infine vorrei ringraziare i miei ex coinquilini del Pilastro tra cui Antonio C. M. che ha portato la luce nelle mie giornate buie, Giuseppe L. e le ragazze dell'appartamento 10: Irene M., Giada V., Giovanna B., Lorena M., Sabrina L., Stefania L.