

UNIVERSITÀ DEGLI STUDI DI CATANIA
DIPARTIMENTO DI MATEMATICA E INFORMATICA
CORSO DI LAUREA TRIENNALE IN INFORMATICA

Marco Longo

Set di applicazioni per il potenziamento di individui con disturbi disortografici

RELAZIONE PROGETTO FINALE

Relatore:
Chiar.mo Prof. Filippo Stanco

Correlatore:
Prof. Dario Allegra
Dott. Luigi Claudio Viagrande
Dott.ssa Maria Rosaria Conte

Indice

Introduzione.....	4
1. Disturbi Specifici dell’Apprendimento.....	5
1.1. Introduzione ai DSA.....	5
1.2. Cause e fattori associati.....	6
1.3. L’apprendimento della lettura e della scrittura.....	7
1.4. La disortografia.....	7
2. Serious Games.....	10
2.1. Cos’è un Serious Game.....	10
2.2. Storia dei Serious Games.....	10
2.3. Modalità di apprendimento ed applicabilità.....	12
2.4. Approccio alle modalità di apprendimento.....	13
2.5. Perché usare i Serious Games.....	14
2.6. Sviluppo di un Serious Game.....	16
3. Unity3D.....	17
3.1. Introduzione al game engine.....	17
3.2. Caratteristiche principali.....	18
3.3. Interfaccia dell’editor.....	18
3.4. Scripting.....	22
3.5. User Interface Tool.....	23
4. DSApp.....	24
4.1. Panoramica sull’applicazione.....	25
4.2. Nuovi esercizi.....	29
4.3. Miglioramenti generali.....	33
5. Conclusione.....	34
Riferimenti.....	35

Introduzione

L’obiettivo di questo lavoro di tesi è quello di espandere il Serious Game “DSApp”, nato per supportare l’intervento sui bambini con Disturbi Specifici dell’Apprendimento (DSA).

DSApp è un gioco per personal computer sviluppato in Unity3D e pensato per il potenziamento e trattamento di individui in età scolare con disortografia. L’idea è nata grazie alla collaborazione con il Centro di Psicologia Scolastica *Katane Lab* [20], ed è tutt’ora portata avanti dallo stesso.

Il primo capitolo di questa tesi presenta una breve introduzione ai Disturbi Specifici dell’Apprendimento, incentrata principalmente sulle problematiche relative alla disortografia ed ai processi di scrittura.

Nel secondo capitolo viene trattato l’utilizzo dei Serious Games e la loro applicabilità nel mondo della formazione scolastica e professionale.

Il terzo capitolo introduce il game engine Unity3D, la principale tecnologia adoperata per la realizzazione del progetto.

Nel quarto ed ultimo capitolo vengono mostrate le novità aggiunte all’applicazione e descritti i principali miglioramenti effettuati.

1. Disturbi Specifici dell'Apprendimento

1.1. Introduzione ai DSA

I Disturbi Specifici dell'Apprendimento (DSA) sono manifestazioni patologiche che colpiscono le capacità di apprendimento dei bambini. I DSA riconosciuti dalla legge 170/2010 sono quattro: Dislessia (Disturbo Specifico della Lettura), Disortografia (Disturbo Specifico della Scrittura nella componente ortografica), Disgrafia (Disturbo Specifico della Scrittura nella realizzazione grafica), Discalculia (Disturbo Specifico del Calcolo). La legge 170 dell'8 Ottobre 2010, riconosce la definizione dei DSA e stabilisce una serie di indicazioni volte a garantire il diritto all'istruzione dei bambini che presentano tali disturbi. Tale legge riconosce la dislessia, la disgrafia, la disortografia e la discalculia quali disturbi specifici di apprendimento, che si manifestano in presenza di capacità cognitive adeguate, in assenza di patologie neurologiche e di deficit sensoriali. In particolare, si intende per disortografia un disturbo specifico di scrittura che si manifesta attraverso difficoltà nei processi linguistici di transcodifica. I disturbi citati fanno riferimento ad abilità scolastiche, come la lettura, la scrittura e il calcolo, o a funzioni necessarie alla prestazione scolastica, cioè la capacità di ragionamento, comprensione ed espressione. Una diagnosi di DSA è quindi formulabile solo ed esclusivamente in età scolare.



si chiamano **DISLESSIA, DISGRAFIA, DISORTOGRAFIA e DISCALCULIA**

Nonostante la definizione non consenta di indicare come DSA le situazioni che possono essere associate a deficit sensoriali e motori o all'appartenenza ad un ambiente socioculturale svantaggiato, ciò non significa che bambini con le suddette caratteristiche non possano presentare tali disturbi.

Sebbene i disordini possano manifestarsi in concomitanza con altre disabilità, essi non sono il risultato di tali condizioni o delle influenze di queste.

1.2. Cause e fattori associati

Si presume che i DSA siano dovuti a disfunzioni del sistema nervoso centrale e che possano manifestarsi durante tutto l'arco della vita. Il presentarsi o meno di una particolare difficoltà dell'apprendimento dipenderebbe dall'interazione fra specifiche caratteristiche genetiche. In questa prospettiva, alcuni fattori genetici, comuni ad altri disturbi, agendo in concerto con altri fattori di rischio genetico e ambientale, potrebbero essere responsabili dello sviluppo di deficit cognitivi, talvolta in concomitanza con altri. Lo svantaggio determinato dall'ambiente è rilevante per i disturbi di apprendimento, non perché può costituirne la causa, ma per il suo effetto aggravante e la sua azione ostacolante il recupero funzionale. Problemi nell'autoregolazione dei comportamenti, nella percezione sociale e nelle interazioni sociali possono coesistere con i disordini di apprendimento.

1.3. L'apprendimento della lettura e della scrittura

I disturbi che colpiscono la lettura e la scrittura sono i più studiati in letteratura. Esistono diversi modelli che illustrano l'apprendimento della lettura e della scrittura individuandone diversi stadi evolutivi. Secondo quanto riportato dallo psicologo statunitense Albert Ellis, i bambini tra i 4 e i 5 anni riconoscono le parole scritte come se fossero un oggetto, una figura. Essi basano tale capacità sul patrimonio verbale acquisito fino a quel momento e sono in grado di riconoscere parole familiari in base a specifiche caratteristiche visive. Verso i 6-7 anni, comincia ad essere appresa la corrispondenza fonema-grafema (cioè la capacità di associare suoni a simboli).

Solo così il bambino è in grado di leggere parole che non conosce. Verso i 7-8 anni comincia a consolidarsi un vocabolario interno e, successivamente, le regole che governano la complessità della corrispondenza fra grafema e fonema: il bambino diventa un lettore esperto man mano che apprende nuovi vocaboli e i corrispondenti significati. I successivi sviluppi riguardano l'efficienza e la velocità dei processi che conducono all'automatizzazione della lettura e della scrittura.

La scrittura è un atto compositivo complesso che richiede la specializzazione di differenti processi, la cui integrazione dà luogo a elaborazioni di notevole complessità. Il processo d'apprendimento della scrittura non è quindi semplice e necessita di tempi lunghi in cui le abilità si perfezionano gradualmente. Spesso, nel corso dello sviluppo, si presentano problemi che ritardano o bloccano il consolidamento delle suddette abilità [1].

1.4. La disortografia

Con il termine disortografia si intende il disturbo specifico della scrittura, che riguarda lo scarso controllo ortografico. Il disturbo della scrittura può essere suddiviso in due componenti: una di natura linguistica (deficit nei processi di cifratura) e una di natura motoria (deficit nei processi di realizzazione grafica). Il termine disortografia riguarda soltanto l'aspetto linguistico, ossia i deficit nei processi di cifratura [2].

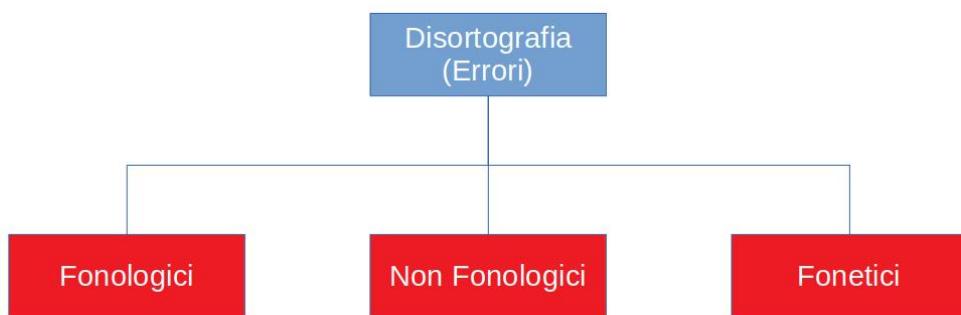
Quando si parla di abilità di scrittura non ci si riferisce ad un processo unitario, bensì ad un insieme di più componenti diverse tra loro:

- Capacità di espressione scritta: comprende tutti i processi che consentono la produzione di un testo scritto, a partire dalla generazione delle idee e della loro organizzazione in una forma coerente;
- Competenza ortografica: comprende i processi di conversione dei singoli fonemi nei corrispondenti grafemi ed i processi di recupero della forma ortografica dell'intera parola, necessari per scrivere correttamente tutte le parole;

- Competenza grafo motoria: comprende le abilità che consentono la riproduzione dei singoli segni grafici, ed il recupero dei pattern motori che consentono la realizzazione di ogni grafema;
- Capacità di produzione scritta: capacità di esprimersi oralmente e di organizzare un testo scritto con contenuti grammaticalmente e sintatticamente corretti.

Per valutare la gravità del disturbo è possibile identificare le tipologie di errori suddividendoli in tre gruppi:

- Errori Fonologici: errori in cui non è rispettato il rapporto tra fonemi e grafemi;
- Errori Non Fonologici: errori nella rappresentazione ortografica (visiva) delle parole, senza errori nel rapporto tra fonemi e grafemi.
- Errori Fonetici: errori legati all'uso di accenti e doppie.



Nel primo caso si possono verificare errori quali omissione, aggiunta o inversione di lettere o sillabe, scambio di grafemi ed errori legati all'uso di diagrammi e trigrammi. Nel secondo caso invece vengono fatti errori come la separazione o la fusione illegale di parole, scambio di grafemi omofoni, uso non corretto della lettera 'h' nel verbo avere ed uso non corretto del grafema 'i' nei gruppi ce/cie, sce/scie, ge/gie. Nell'ultimo caso si verificano errori come l'omissione o l'aggiunta di lettere doppie o accenti.

Ogni alunno con DSA ha caratteristiche proprie che richiedono strumenti personalizzati adeguati. “*Le linee guida per il diritto allo studio degli alunni e degli studenti con disturbi specifici di apprendimento*” specificano che gli *strumenti compensativi* sono strumenti didattici e tecnologici che sostituiscono o facilitano la prestazione richiesta nell’abilità deficitaria e che le *misure dispensative* sono invece interventi che consentono all’alunno o allo studente di non svolgere alcune prestazioni che, a causa del disturbo, risultano particolarmente difficoltose e che non migliorano l’apprendimento [3].

2. Serious Games

2.1. Cos'è un Serious Game

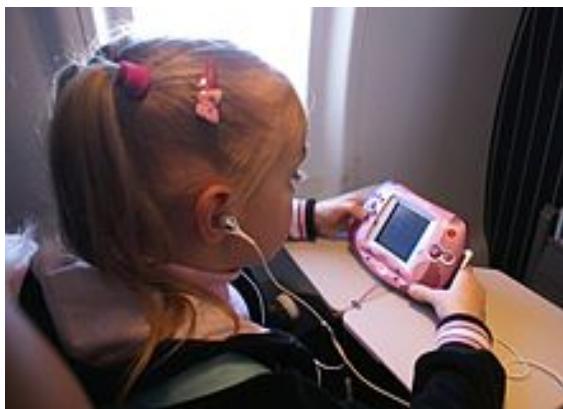
Un serious game, noto anche come applied game, è un gioco che non ha come unico intento quello di intrattenere. I serious games sono infatti strumenti formativi in cui gli aspetti seri e quelli ludici stanno idealmente in equilibrio. L'aggettivo “serious” viene in genere anteposto quando ci si vuole riferire ai videogiochi utilizzati in ambito educativo, scientifico, politico, ingegneristico e sanitario. L’idea che vi sta alla base è quella di enfatizzare esplicitamente il valore pedagogico aggiunto del divertimento e della competizione.

2.2. Storia dei Serious Games

L’origine dell’uso dei giochi nel campo dell’educazione risale alla seconda metà del ventesimo secolo. Negli anni ‘70, erano molto diffusi giochi cartacei a scopo educativo, i quali avevano come scopo principale quello di stimolare gli studenti verso la lettura, la scrittura e il calcolo. Il termine serious game venne probabilmente usato la prima volta da Clark Abt, sviluppatore di giochi militari su computer, nel 1971. Nel suo libro *Serious Games*, il ricercatore statunitense fornisce anche una definizione dei giochi educativi, a prescindere dal supporto (digitale o real-world). Li descrive come giochi con un esplicito e ben strutturato scopo educativo, non pensati primariamente per il divertimento, senza però escluderlo.

Durante i primi anni del 2000, nacquero diverse tipologie di giochi educativi progettati per un pubblico molto giovane. Molti di questi giochi non erano digitali, ma prendevano parecchia ispirazione dai videogiochi per console e per dispositivi mobili. Basandosi sulla popolarità dei *Game Boy*, prodotti da Nintendo a partire dal 1989, la compagnia LeapFrog Enterprises [5] creò, nel 2003, il cosiddetto *Leapster*, una console portatile a scopo educativo ideata per i bambini dai 4 ai 10 anni, contenente una serie di mini-giochi in stile arcade avente carattere formativo.

Nel 2010, i serious games avevano raggiunto un livello evolutivo superiore, andando ad incorporare anche sistemi economici reali. Nel 2015, venne lanciato sul mercato *Project Discovery*, un serious game creato dai genetisti dell'Università di Ginevra. I giocatori, nel ruolo di “citizen scientists” (cioè di semplici cittadini che partecipano ad una ricerca scientifica), per mezzo di mini-giochi interattivi, catalogano ed esaminano campioni genetici reali, producendo dati che vengono in seguito analizzati ulteriormente dai ricercatori [4].



Una bambina con un Leapster



Scena di gioco in Project Discovery

2.3. Modalità di apprendimento ed applicabilità

Non esiste una teoria uniforme sulla pedagogia dei serious games. Se in passato i giochi erano basati sulla psicologia comportamentale, come ad esempio nel gioco *Mathblaster*, oggi gli sviluppatori di videogiochi educativi attingono da vari modelli pedagogici. Particolare importanza ha assunto l'apprendimento esperienziale che parte dal presupposto che le informazioni e le sensazioni vissute rimangono fortemente impresse e permettono in questo modo al giocatore di affinare percezione, attenzione e memoria, favorendo modifiche comportamentali attraverso il learning by doing (imparare facendo). Il giocatore di serious games ha il vantaggio di agire in un ambito protetto. Interiorizzare qualcosa che si è fatto di persona, quindi attivamente, risulta molto più semplice rispetto all'apprendimento di contenuti veicolati durante lezioni frontali, apprendimento cosiddetto passivo. Rendendo la simulazione molto vicina alla realtà, si riduce la paura del nuovo aumentando la fiducia dell'utente nel mettersi in gioco.

L'elemento ludico aumenta il coinvolgimento e permette di agire più spontaneamente, senza sentirsi giudicati. La possibilità di ripetere l'esercizio innumerevoli volte permette di giungere alla totale padronanza della dinamica "esplorata" aumentando la serenità con la quale si utilizza lo strumento e con la quale si affronterà la situazione reale.

I serious games oggi trovano applicazioni in numerosi contesti. L'ambito civile, la formazione aziendale, le campagne di educazione e di sensibilizzazione, nonché varie attività promozionali e campagne sociali, sono tutti campi di applicazione del serious gaming. In contesti commerciali i serious games vengono solitamente utilizzati per simulare situazioni di vendita faccia a faccia e telefonica, interazioni in occasione di colloqui o di riunioni e, in generale, tutte quelle situazioni in cui si renda necessaria un'esperienza diretta per assimilare contenuti e comportamenti. Per ciò che attiene agli usi militari, il serious game trova applicazione sia nell'addestramento del personale tecnico (quali operatori radar, personale di macchina a bordo delle navi, piloti di aerei ed elicotteri) che nella simulazione di operazioni militari complesse come i cosiddetti war games. Molti videogiochi, anche se non concepiti originariamente come serious games, possono trovare uso in ambito educativo. Negli ultimi anni il serious game è diventato uno degli strumenti più apprezzati nel campo dell'Agile Coaching, in quanto capace di stimolare la capacità di collaborazione, il team building, la creatività e l'innovazione. Una delle tecniche di serious game più conosciuta e meglio codificata è quella del *LEGO Serious Play*, un metodo finalizzato a sviluppare il pensiero, la comunicazione e la risoluzione di problemi complessi di gestione aziendale attraverso l'impiego del gioco di costruzioni Lego [6].



Svolgimento di attività di LEGO serious play

2.4. Approccio alle modalità di apprendimento

Giocando ad un serious game si possono allenare diverse abilità, come la lettura, la scrittura e il calcolo, ma è anche possibile imparare qualcosa di nuovo. Negli ultimi anni si sta sviluppando un nuovo approccio nella costruzione di giochi a scopo educativo, il quale coinvolge i seguenti elementi:

- Universi di gioco: il giocatore deve immergersi nell'universo di gioco e deve pensare e percepire l'esperienza di apprendimento, piuttosto che semplicemente allenare una particolare abilità. Le azioni del gioco devono trovare un punto in comune con ciò che si dovrebbe imparare;
- Mantenimento della qualità di gioco: le caratteristiche tipiche dei videogiochi tradizionali, come la motivazione, l'interesse e l'interattività devono essere sempre mantenute;
- Uniformità tra gioco e apprendimento: gli obiettivi, l'azione e le sfide proposte dal gioco devono sovrapporsi all'esperienza di apprendimento, la quale viene a sua volta facilitata dall'esperienza di gioco;
- Sviluppo di qualità: la progettazione del gioco deve avvenire in collaborazione con esperti del settore e la realizzazione va fatta attraverso tecnologie di sviluppo avanguardistiche, adoperate nell'ormai consolidata industria del videogioco.

L'obiettivo principale che si intende raggiungere è quello di creare serious games che possano competere sul mercato con i tanto amati videogiochi tradizionali. Questi ultimi sono infatti più coinvolgenti, stimolanti e accattivanti, in virtù della loro interattività, dei ricchi universi che contengono e delle sfide che propongono.

I serious games attuali, in generale, usano la loro parte ludica come semplice premio per l'aver appreso qualcosa, piuttosto che renderla parte integrante di ciò che si impara. Il nuovo approccio che si intende seguire prevede dunque un ampliamento delle caratteristiche già presenti nei videogiochi di successo, attraverso missioni educative che siano rilevanti nell'universo di gioco [7].

2.5. Perché usare i Serious Games

Qualunque tipologia di gioco, che sia da tavolo, sociale o digitale, richiede che il giocatore impari qualcosa. Il minimo requisito è infatti l'apprendimento delle regole del gioco. I giochi possono inoltre stimolare emozioni come la soddisfazione, la felicità, la rabbia e la frustrazione, le quali aprono un canale diretto con la nostra mente, consentendo una comprensione più rapida e profonda delle nuove conoscenze, dei nuovi concetti e delle nuove abilità, favorendo anche una più lunga conservazione nella nostra memoria.

Per riuscire a progredire nel gioco, i giocatori, in genere, devono sviluppare una serie di strategie e di abilità propedeutiche alle fasi successive. Sotto questo punto di vista, si può asserire che tutti i giochi siano educativi, in quanto richiedono al giocatore un apprendimento costante, richiamando frequentemente regole e meccaniche di gioco.

Moderne teorie riguardanti l'allenamento, l'apprendimento e l'educazione, affermano che è più semplice imparare quando lo si fa in modo attivo, ricevendo feedback in maniera più immediata. I serious games rappresentano una tecnologia che si presta perfettamente all'apprendimento attivo, permettendo ai giocatori di imparare con delle azioni piuttosto che ascoltando o leggendo. Viene sfruttato inoltre il fattore "piacere" del gioco, il quale intensifica le esperienze emozionali e di apprendimento.

Si possono elencare diversi risultati dell'apprendimento ottenuti grazie ai serious games, come ad esempio l'acquisizione e la comprensione del contenuto, le abilità percettive e cognitive (riconoscimento del contesto, allenamento della memoria, attitudine al problem-solving), cambiamento di comportamento e atteggiamento (acquisizione di autostima e miglioramento abilità sociali come abilità di leadership) e competenze sociali (lavoro di squadra e abilità di comunicazione).

Un altro ben noto vantaggio nell'uso dei serious games è quello di consentire ai giocatori di sperimentare situazioni che sono impossibili nel mondo reale per diverse ragioni, tra cui la sicurezza, i costi o il tempo [8] [9].

Questa tipologia di giochi rappresenta uno strumento alternativo, sicuro ed economico per l’allenamento in situazioni pericolose che richiederebbero più risorse nel mondo reale (ad esempio i simulatori di volo).

In sintesi, è possibile specificare tre ragioni principali per l’uso crescente dei serious games nell’istruzione scolastica e professionale:

- usano le azioni piuttosto che le spiegazioni e creano motivazione personale e soddisfazione;
- ospitano più stili e abilità di apprendimento;
- promuovono attività decisionali e di problem-solving in un ambiente virtuale.

Tutti questi elementi dimostrano che l’uso di questi giochi può essere un mezzo efficace per comunicare e diffondere la conoscenza anche al di là del pubblico accademico, contribuendo così a rafforzare i legami tra scienza e società [\[10\]](#) [\[11\]](#).

2.6. Sviluppo di un Serious Game

Un sostanziale impedimento alla diffusione dei serious games come strumento per l’istruzione e la formazione sembra essere rappresentato dagli sforzi solitamente ingenti coinvolti nel loro processo di sviluppo, il quale si dimostra spesso complesso e dispendioso in termini di tempo e costi. Infatti, gli approcci e gli strumenti esistenti per lo sviluppo dei videogiochi tradizionali, non possono essere semplicemente utilizzati per l’istruzione.

La complessità dello sviluppo dei serious games è da attribuire, almeno in parte, all’elevato numero di diverse figure professionali coinvolte: dal cliente che specifica i suoi bisogni, agli specialisti cognitivi, gli esperti pedagogici e gli sviluppatori di software. I frameworks esistenti per la progettazione di serious games, dimostrano la presenza di questa ampia diversità di discipline coinvolte attraverso la varietà di fonti da cui attingono. Infatti, è possibile trovare al loro interno elementi relativi ad ambiti quali l’istruzione, l’educazione, la sociologia e la psicologia [\[12\]](#) [\[13\]](#) [\[14\]](#).

Ad esempio, il framework MDA (Mechanics, Dynamics, and Aesthetics) rappresenta un tentativo di colmare il divario tra progettazione e sviluppo del gioco e la ricerca tecnica. Nonostante questo framework fosse stato creato inizialmente per lo sviluppo di videogiochi da intrattenimento, esso è andato incontro ad una serie di modifiche, mirate al suo adattamento per l'uso educativo e formativo. Tali miglioramenti hanno portato alla creazione di un nuovo framework, detto DPE (design-play-experience), il quale permette di definire approcci di progettazione di giochi digitali a scopo di apprendimento. Secondo DPE, nel processo di progettazione del gioco sono principalmente coinvolte tre componenti [\[15\]](#):

- lo sviluppo del gioco da parte del designer (Design);
- la giocabilità dell'applicazione per il giocatore (Play);
- l'esperienza che il giocatore trae dal gioco (Experience).

Successivamente a DPE, sono nati nuovi framework, i quali hanno dato vita ad alcuni modelli fondamentali come quello delle “Sei ‘I’ dello sviluppo di un serious games” (Identity, Immersion, Interactivity, Increasing complexity, Informed teaching, Instructional), che descrivono un modello molto avanzato con il quale i game designer possono progettare giochi educativi molto efficaci.

Tuttavia, i framework esistenti di solito forniscono solo una piccola direzione progettuale, spesso mancano di una prospettiva pedagogica, e non riescono a suggerire come affrontare la complessità del gioco. La progettazione di un gioco rimane un processo creativo, unico per ogni situazione. Pertanto, non esiste una ricetta unica per il design che garantisca il successo [\[16\]](#).

3. Unity3D

3.1. Introduzione al game engine

Unity3D è un game engine cross-platform sviluppato da Unity Technologies e utilizzato per lo sviluppo di applicazioni interattive e videogiochi 2D e 3D.

La storia di Unity inizia nei primi anni del 2000, quando tre giovani programmati, David Helgason, Joachim Ante e Nicholas Francis, si unirono per iniziare a programmare ciò che poi sarebbe diventato uno dei più famosi sistemi software dell'industria del videogame. La prima versione dell'engine venne lanciata sul mercato nel 2005 come esclusiva Apple e ottenne subito un enorme successo. La popolarità di Unity crebbe negli anni, portando alla sua espansione su diverse piattaforme commerciali. Ad oggi Unity è supportato sia da Windows (a partire dalla versione XP) che da Linux (sperimentalmente dal 2015 con la versione 4.0).

Unity permette di realizzare prodotti per ben 27 piattaforme diverse, tra cui PlayStation 4, Xbox One, Android, Nintendo Switch, Smart TV e sistemi di realtà virtuale. La crescita del game engine continua ancora oggi, con moltissimi progetti di estensione del supporto per ulteriori piattaforme [\[17\]](#).

3.2. Caratteristiche principali

L'editor di Unity fornisce un ambiente di sviluppo per la creazione di videogiochi, basato su un sistema drag and drop. Helgason stesso sostiene che sia possibile creare un gioco in Unity senza scrivere alcuna riga di codice, anche se la maggior parte dei progetti richiedono una buona parte di programmazione. Unity supporta i seguenti linguaggi di programmazione: C# (consigliato), JavaScript e Boo, un linguaggio che utilizza una sintassi in stile Python. Nonostante lo sviluppo all'interno dell'engine venga effettuato tramite questi linguaggi, Unity è un programma scritto in C++. L'ambiente di sviluppo viene eseguito su Mono, una versione open source del framework .NET.

Unity è un engine all-in-one, in quanto dispone di una grande varietà di tools interni che permettono la realizzazione di interi progetti, senza l'ausilio di ulteriori software. Tali tools consentono la creazione di animazioni, il lavoro sul comparto grafico, il mixing e l'editing di clip audio, l'emulazione di fenomeni fisici attraverso motori dedicati sia al 2D che al 3D e il profiling relativo alle prestazioni.

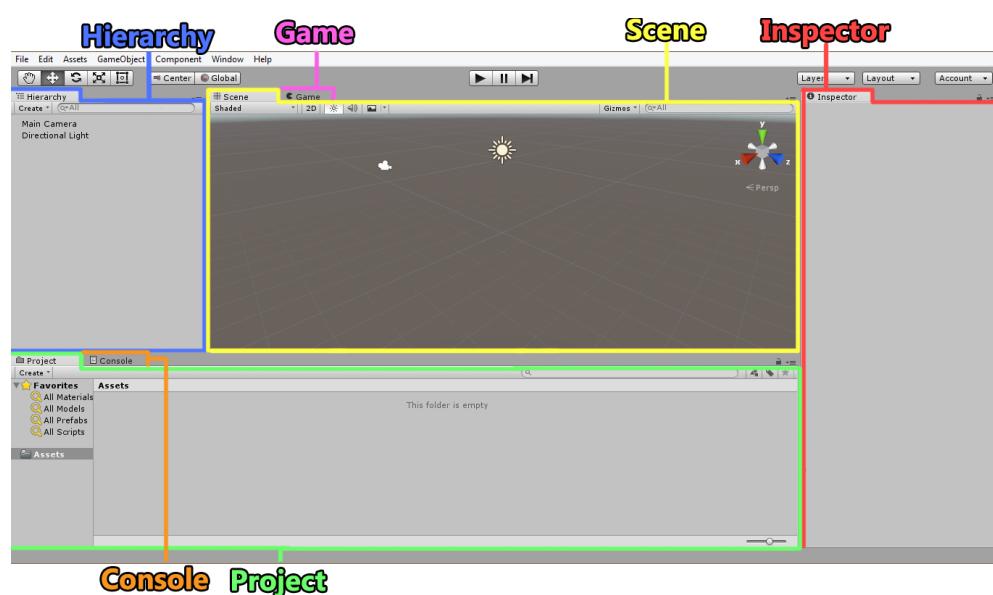
3.3. Interfaccia dell'editor

Un concetto fondamentale, che sta alla base del funzionamento dell'intero engine, è quello di Game Object: tutti gli elementi presenti all'interno della scena di Unity sono dei Game Objects; questi sono, di base, dei contenitori vuoti, ai quali vengono applicati vari componenti che vanno a definire la natura degli oggetti.

Tali componenti sono detti appunto Components e sono in grado di specializzare ogni singolo Game Object, affidandogli una particolare funzionalità.

L'editor fornisce inoltre il cosiddetto sistema dei Prefabs, il quale consente di creare, configurare ed archiviare un Game Object completo di tutti i suoi Components, in modo da rendere possibile il loro riutilizzo. Un Prefab funge quindi da modello, a partire dal quale è possibile creare sulla scena nuove istanze dello stesso oggetto.

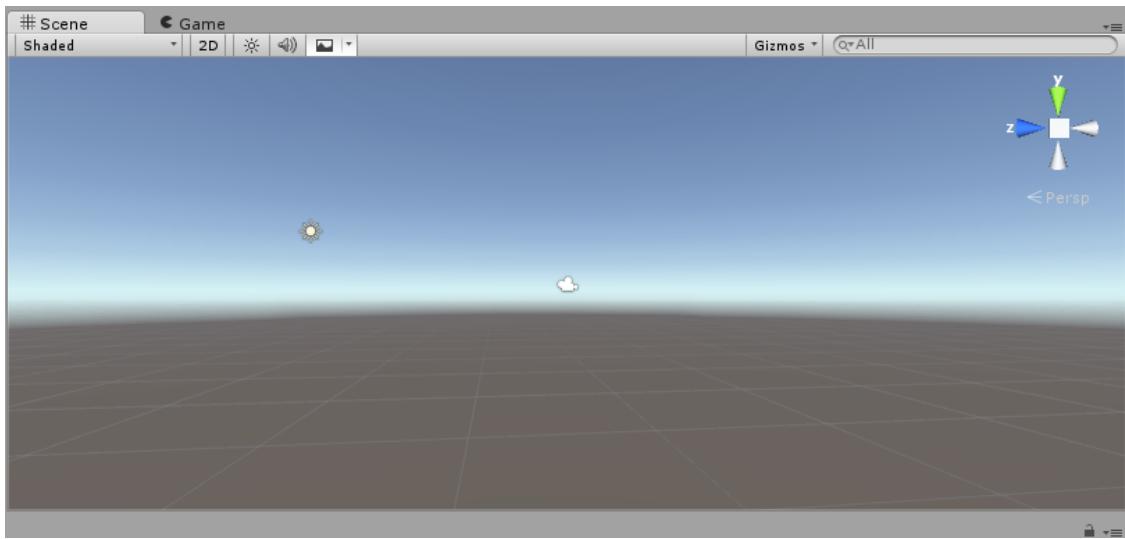
L'interfaccia di Unity è composta da 6 pannelli base:



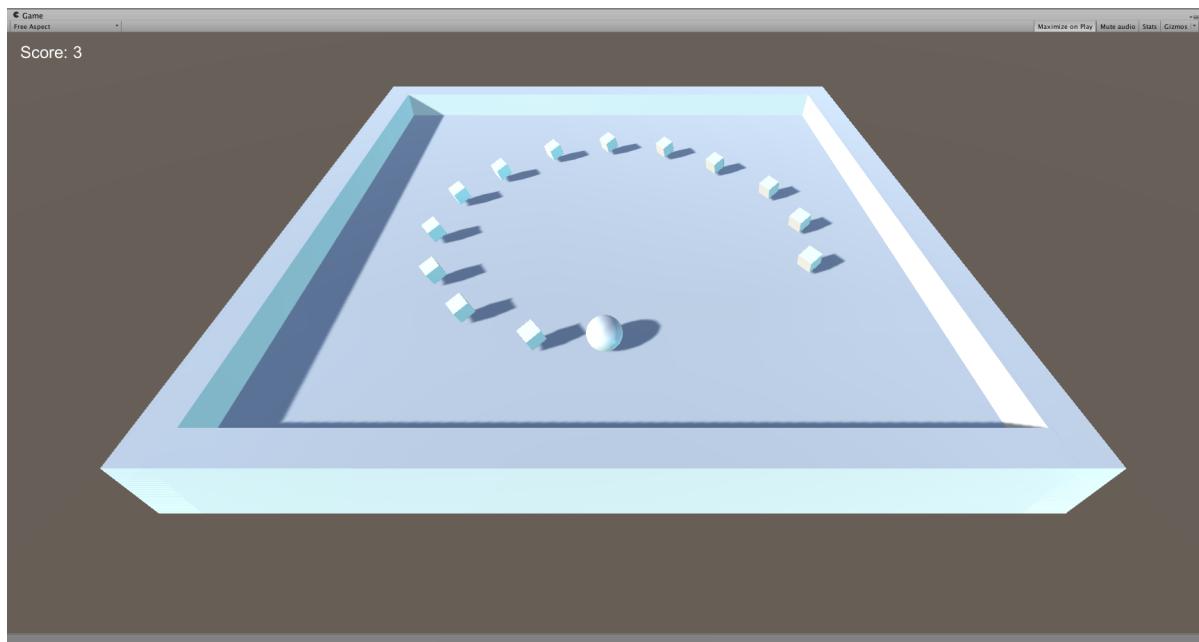
- Hierarchy: questa scheda contiene un albero gerarchico dei Game Objects presenti sulla scena corrente. Alla creazione di ogni nuovo oggetto e alla cancellazione di uno degli oggetti già presenti, l'albero si aggiorna automaticamente. Gli oggetti vengono sistemati in maniera automatica nell'ordine in cui vengono creati, ma vi è la possibilità di riorganizzarli secondo le proprie esigenze. La gerarchia degli oggetti viene definita tramite il meccanismo di “parenting”, il quale permette l’organizzazione dei Game Objects in gruppi. Ogni gruppo ha come capo un “parent object”, al quale fanno riferimento uno o più oggetti figli, i quali a loro volta possono essere dei “parent object” per altri oggetti.



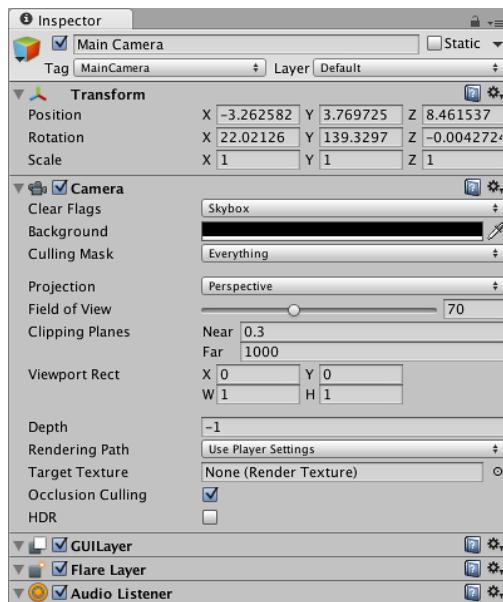
- Scene: la scena di gioco è il contenitore per i vari Game Objects. La Scene Tab propone all’utente una vista interattiva, attraverso la quale è possibile posizionare, spostare, ruotare e ridimensionare i Game Objects, gestire la telecamera e regolare l’illuminazione. Lo spazio sulla scena può essere reso sia tridimensionale che bidimensionale. La presenza o meno dell’asse cartesiano “z” permette di discernere le due modalità. Nello sviluppo del progetto viene utilizzata la visualizzazione 2D.



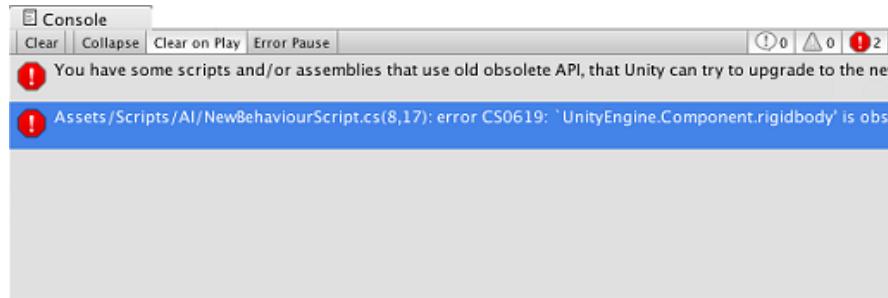
- Game: questa scheda dà un'anteprima rappresentativa del prodotto finale. La Game Tab mostra la scena inquadrata dalla telecamera principale e fornisce la possibilità di testare le funzionalità implementate.



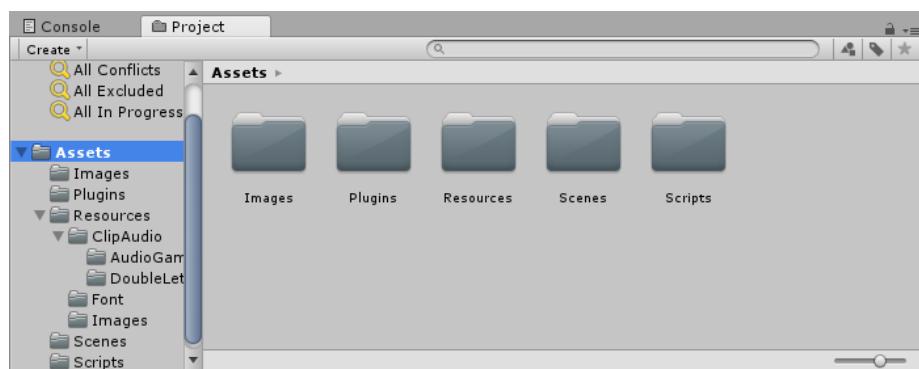
- Inspector: è la scheda che consente di visualizzare le informazioni dei vari Game Objects presenti sulla scena. In particolare, vengono mostrati i relativi Components e le loro proprietà, e viene data la possibilità di modificarli direttamente. Nel caso in cui l'oggetto contenga degli script, sarà inoltre possibile istanziarne le variabili pubbliche.



- Console: in questa scheda vengono mostrati tutti i messaggi di errore ed i warnings generati da Unity. Attraverso la console è inoltre possibile ricevere messaggi di debug, i quali vengono generati da specifiche funzioni inserite negli script.



- Project: questa scheda fornisce una visualizzazione d'insieme degli Assets che sono stati importati nel progetto. È composta da due pannelli affiancati, i quali permettono rispettivamente di navigare all'interno delle cartelle del progetto e di visualizzarne i contenuti.



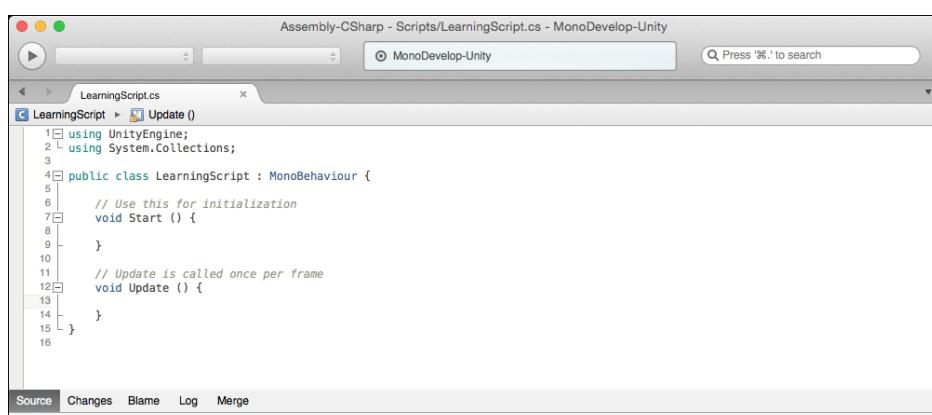
3.4. Scripting

Al fine di affidare ad ogni oggetto uno o più specifici comportamenti, è necessario creare degli script, cioè dei frammenti di codice che caratterizzino i Game Objects ai quali vengono uniti. Uno script in Unity non è altro che una sequenza di istruzioni, scritte nel linguaggio di programmazione C#, che descrivono le funzionalità di un oggetto. Tutte le classi implementate negli script Unity ereditano dalla classe MonoBehaviour, la quale fornisce un'ampia interfaccia contenente molteplici funzioni ed eventi.

Tra le funzioni principali ricordiamo:

- Awake: funzione richiamata nel momento in cui l'oggetto associato allo script viene caricato sulla scena di gioco. Viene generalmente usata per inizializzare le variabili che rappresentano lo stato iniziale dell'oggetto.
- Start: funzione richiamata quando l'oggetto associato allo script viene reso attivo sulla scena. Viene sempre eseguita dopo l'Awake e prima del primo Update.
- Update: funzione richiamata ad ogni frame dell'esecuzione. È utilizzata per implementare qualunque tipologia di cambiamento dinamico nel gioco.

Lo scripting è un ingrediente essenziale nello sviluppo di un gioco, in quanto consente di generare risposte agli input del giocatore e di organizzare l'esecuzione dei vari eventi. Oltre a questo, gli script consentono di creare effetti grafici e sonori, gestire fenomeni fisici e persino implementare sistemi di Intelligenza Artificiale (IA) personalizzati.



The screenshot shows the MonoDevelop-Unity interface with the title bar "Assembly-CSharp - Scripts/LearningScript.cs - MonoDevelop-Unity". The main window displays the code for "LearningScript.cs":

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class LearningScript : MonoBehaviour {
5
6     // Use this for initialization
7     void Start () {
8
9     }
10
11    // Update is called once per frame
12    void Update () {
13
14    }
15
16 }
```

At the bottom of the interface, there are tabs for "Source", "Changes", "Blame", "Log", and "Merge".

3.5. User Interface Tool

L'UI Tool di Unity è uno strumento che consente di creare interfacce grafiche utilizzando un'ampia gamma di elementi grafici, i quali vengono inseriti in un'unica area di lavoro denominata canvas. All'interno della gerarchia dei Game Objects, il canvas rappresenta l'oggetto padre, mentre tutti gli elementi che compongono l'interfaccia saranno suoi figli. Sulla scena, l'oggetto canvas viene rappresentato da un semplice rettangolo, all'interno del quale è possibile inserire elementi grafici quali testo, sliders, bottoni, immagini ed effetti vari. Dato che ogni elemento dell'interfaccia è un Game Object, è possibile associarvi scripts e Components, così da definire i loro comportamenti.

Il gioco DSApp è interamente realizzato sfruttando le funzionalità offerte dall'User Interface Tool di Unity [\[18\]](#) [\[19\]](#).

4. DSApp

Il progetto di tesi consiste nell'estensione dell'applicazione DSApp attraverso una serie di nuovi esercizi e di miglioramenti relativi all'esperienza utente.

Il gioco si rivolge principalmente a bambini in età scolare, tra i 6 e 13 anni, che risentono di Disturbi Specifici dell'Apprendimento, e si pone come obiettivo il potenziamento e il trattamento dell'individuo attraverso il gioco. L'applicazione presuppone che l'utilizzo sia supervisionato da parte di un tutor, il quale potrà verificare e gestire l'andamento dei vari giocatori.

4.1. Panoramica sull'applicazione

Il gioco prevede due sezioni principali: una sezione di gioco, contenente l'indice relativo agli esercizi interattivi, e una sezione dedicata al tutor, la quale contiene dati relativi agli utenti registrati sul sistema.

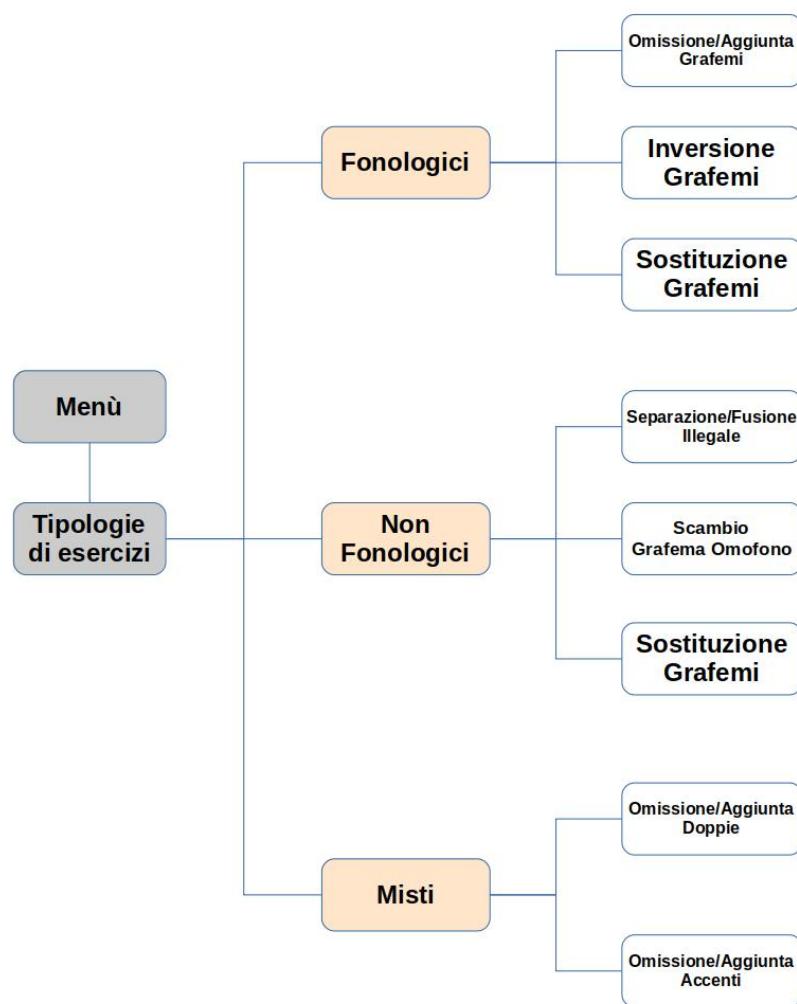


All'interno della sezione di gioco è possibile navigare tra le diverse categorie di esercizi ed è inoltre possibile scegliere tra le due modalità di gioco esistenti:

- Gioco libero: questa modalità permette di esercitarsi sui giochi senza effettuare la raccolta dei dati. È stata pensata principalmente per l'allenamento del bambino in assenza della supervisione del tutor.

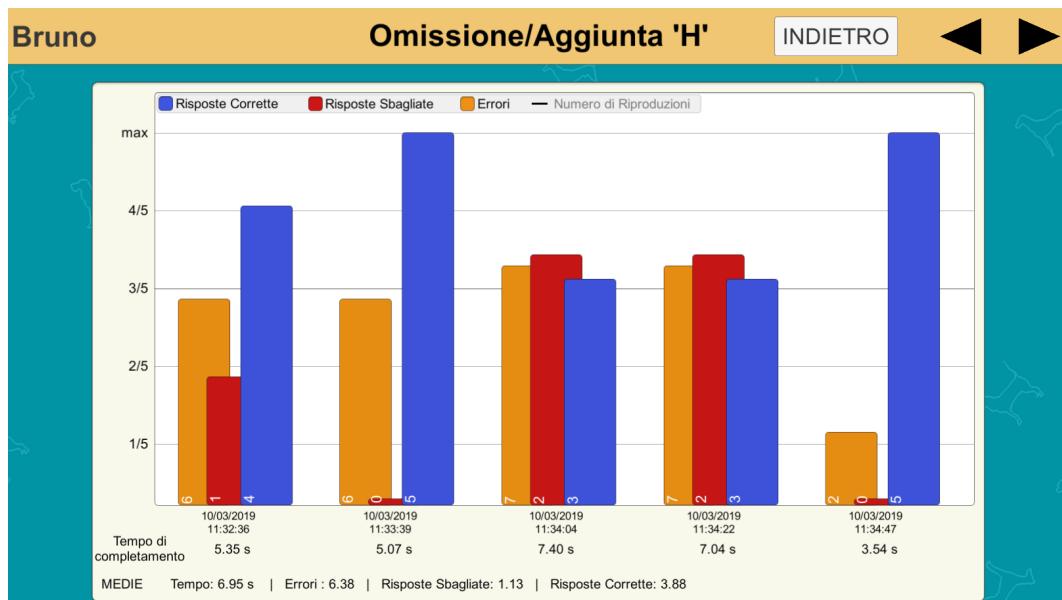
- Test: questa modalità è protetta da una password ed è accessibile solo in presenza del tutor. Gli esercizi proposti sono quelli presenti nella modalità “Gioco Libero”, ma il tempo di gioco, gli errori e le risposte sbagliate vengono conteggiati e poi registrati.

I giochi all’interno dell’applicazione sono suddivisi in tre macro categorie, corrispondenti alle tre tipologie di errori comuni negli individui affetti da disturbi disortografici. Ogni macro categoria contiene delle sotto categorie specifiche per il genere di errore relativo.



Una volta selezionata la categoria desiderata, l’utente può scegliere tra i vari esercizi che vengono proposti dal sistema. Ogni sessione di gioco prevede l’estrazione casuale di 5 domande da un database, realizzato in *SQLite*, le quali vengono proposte all’utente secondo la metodologia prevista dall’esercizio.

Nella sezione dedicata al tutor, accessibile solo mediante password, è possibile inserire i nominativi dei vari giocatori e consultare le statistiche relative ad ognuno di essi.



Per ogni bambino e per ogni esercizio viene riportato uno storico dei test effettuati, con i relativi risultati: tempo di completamento, errori, risposte errate e risposte esatte. I valori rappresentati dalle barre sono normalizzati, in maniera tale da permettere un'analisi immediata del progresso del bambino.

4.2. Nuovi esercizi

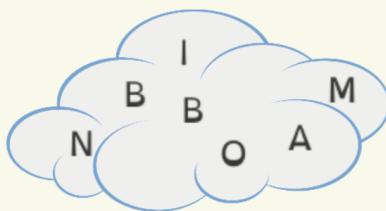
L'aggiunta di nuovi esercizi ha allargato la varietà di attività proposte all'utente, garantendo così più opzioni per ogni singola macro categoria.

L'implementazione di ogni nuovo esercizio richiede la creazione di una relativa tabella nel database delle domande, la quale contiene sia i quesiti, che le relative risposte, codificate in maniera opportuna. Il menù principale deve essere inoltre arricchito in modo da comprendere un'opzione per l'accesso al nuovo gioco e la specifica logica dell'esercizio va implementata tramite script opportuni. Tutti gli esercizi fanno capo ad un unico metodo che si occupa di individuare la tabella dalla quale estrarre le domande, scegliere casualmente uno dei vari record presenti e richiamare il metodo specifico per l'esercizio in questione.

Di seguito si riporta un elenco dei nuovi giochi aggiunti, associati alle rispettive sotto categorie:

- Omissione o aggiunta di grafemi: il gioco consiste nel comporre delle parole di senso compiuto usando le lettere presenti all'interno della nuvola. La generazione delle lettere all'interno della nuvola è casuale ed eventuali soluzioni multiple vengono accettate dal sistema.

Scrivi le parole utilizzando le lettere che trovi all'interno della nuvola



Inserisci la risposta

Invia

Torna al Menù

- Inversione di grafemi: il bambino deve trovare all'interno della tabella di sillabe, tutte quelle che corrispondono al modello posto in alto. L'esercizio termina quando si trovano tutte le corrispondenze esatte.

Ricerca la sequenza di lettere uguale al modello

IL

LI	LE	GI	IL	IL	LA	IL	AS
LI	IF	LI	UL	WI	LO	FI	ND
LU	LI	AD	WE	IL	OL	LO	LI
LI	AR	LI	II	IT	IL	AT	MU
IL	IL	LI	LI	AR	LI	LA	LE

Torna al Menù

- Scambio di grafemi: l'esercizio contiene una serie di immagini rappresentanti oggetti e animali, le quali vanno associate al suono richiesto dalla domanda. In particolare, viene richiesto di selezionare tutte quelle figure che contengono o iniziano con il suono indicato. L'utente apprende immediatamente l'esito della sua scelta vedendo apparire sotto l'immagine appena selezionata il nome del soggetto rappresentato.

Seleziona le figure che **iniziano** con il suono **F**

Fiammifero ✓

Pallone ✗

Scatola ✗

Quadrifoglio ✗

Fata ✓

[Torna al Menù](#)

- Separazioni e fusioni illegali: l'utente, data una frase completa priva di spazi, deve conteggiare il numero di parole che essa contiene e scrivere il valore corretto nella casella apposita. Se si danno più di due risposte sbagliate il sistema mostra automaticamente la risposta corretta.

Quante parole ci sono?

Il mio migliore amico si chiama Carlo

7

Successiva

Corretto!

[Torna al Menù](#)

- Scambio grafema omofono: viene presentata un'interfaccia in cui sono presenti 4 tasti contenenti quattro grafemi omofoni (“CU”, “CQU”, “QU” e “CCU”) e una frase mancante di uno solo di questi grafemi. Anche in questo caso, dopo il terzo errore, il sistema mostra la risposta corretta

Inserisci il riquadro corretto

La nonna mi ha regalato un ...adro

CU	QU
CCU	CQU

[Torna al Menù](#)

- Omissione o aggiunta di ‘h’: il bambino deve scegliere quale tra le due frasi proposte è quella corretta. Le due frasi presentano lo stesso contenuto a meno della parola contenente la lettera ‘h’. La posizione delle frasi è scelta in maniera casuale durante l'esecuzione.

Clicca sulla frase corretta

Posso venire oggi a casa tua?

Posso venire oggi ha casa tua?

[Torna al Menù](#)

- Omissione o aggiunta di doppie: il gioco consiste nell'individuare quale tra le due opzioni è quella corretta. I pulsanti relativi contengono la stessa parola ripetuta in un caso senza doppia consonante e nell'altro con doppia consonante. Anche qui la posizione dei due bottoni è casuale.

Completa le frasi con la parola corretta

Quando piove, esco con l' _

ombrelo **ombrelllo**

[Torna al Menù](#)

- Omissione o aggiunta di accento: l'esercizio richiede che vengano selezionate tra quelle proposte, le parole che dovrebbero contenere l'accento. Il gioco termina non appena l'utente seleziona tutte le opzioni esatte.

Clicca le parole che contengono l'accento

Sabato **Lunedì**

Perche **Giovedì**

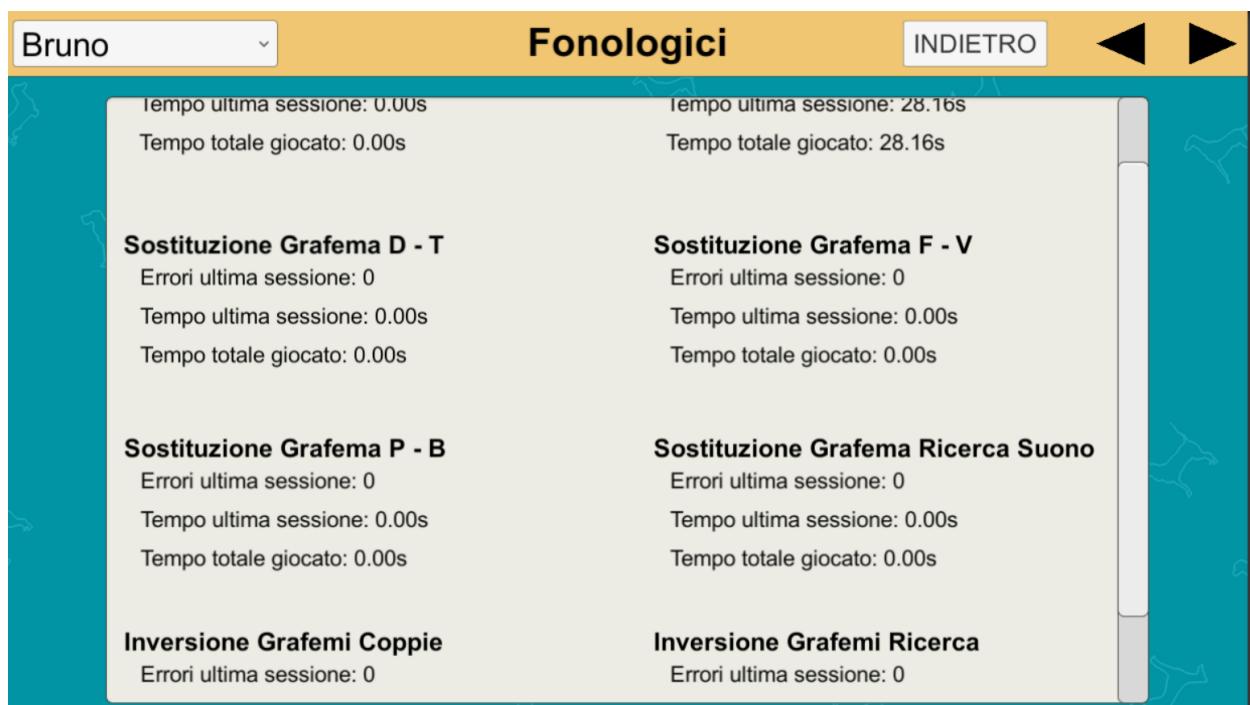
[Torna al Menù](#)

4.3. Miglioramenti generali

Ogni gioco è stato implementato in parallelo con la sua sezione nel menù delle statistiche. I dati sono suddivisi in base alle tre macro categorie ed è possibile passare dall'una all'altra attraverso appositi pulsanti sull'interfaccia.

Dato l'elevato numero di esercizi presenti, è risultato indispensabile aggiungere un meccanismo di scorrimento. È stata quindi aggiunta una scrollbar verticale che permette la navigazione attraverso tutti gli esercizi per la macro categoria correntemente attiva. Al passaggio da una macro categoria all'altra, la barra mantiene la sua posizione, a meno che la categoria di destinazione contenga un numero inferiore di esercizi: in tal caso la barra si riposiziona all'inizio e lo scorrimento viene disabilitato, così da impedire l'occultamento delle sezioni.

È stata inoltre implementata una nuova funzionalità, non presente nella precedente versione del gioco, la quale permette di ricordare la macro categoria attiva al momento della selezione di una particolare sezione. Una volta completata la consultazione delle relative statistiche, l'utente viene reindirizzato sulla macro categoria che stava esaminando, anziché sulla schermata iniziale.



5. Conclusione

I serious games rappresentano ad oggi una soluzione concreta per il trattamento di bambini con Disturbi Specifici dell’Apprendimento, in quanto permettono di affrontare tali disordini in una maniera giocosa e divertente. Le tecnologie che vanno via via sviluppandosi favoriscono la creazione di applicazioni sempre migliori, e la collaborazione con le diverse figure professionali coinvolte rende i prodotti sempre più efficienti.

È fondamentale che il processo di potenziamento venga seguito attentamente da un tutor, il quale deve monitorare i progressi dei bambini. Per tali motivi, il progetto DSApp risulta essere piuttosto adatto alla sperimentazione medica.

È inoltre prevista l’implementazione di DSApp per i disturbi non trattati in questo lavoro di tesi. Si potrà così formare una suite di applicazioni per il trattamento di tutti i disturbi più comuni.

Riferimenti

1. Mirella Zanobini, Maria Carmen Usai, *Psicologia della disabilità e dei disturbi dello sviluppo*, Settima Edizione, 2019.
2. Consiglio Nazionale Ordine degli Psicologi, *I DSA e gli altri BES*, 2016.
3. C. Cornoldi, *La produzione del testo scritto: generare e pianificare le idee*, «<https://www.labda-spinoff.it>», 2013.
4. K. Katsaliaki, N. Mustafee, *A survey of serious games on sustainable development*, 2012.
5. J. H. Gray, J. Bulat, C. Jaynes, A. Cunningham, "LeapFrog learning". *Mobile Technology for Children: Designing for Interaction and Learning*, 2009.
6. LEGO, *Serious Play The Method*, «<https://www.lego.com>» , 2019.
7. Simon Egenfeldt-Nielsen, *The basic learning approach behind Serious Games*, 2005.
8. D. Michael, S. Chen, *Serious Games: Games That Educate, Train, and Inform*, Boston, 2006.
9. Abt, Clarck C., *Serious Games*, 1971.
10. H. Jenkins, B. Camper, A. Chisholm, N. Grigsby, E. Klopfer, S. Osterweil, J. Perry e P. Tan, *From serious games to serious gaming*, 2009.
11. M. Ulicsak, M. Wright, Games in education: *Serious games*, Slough, U.K.: Future-Lab, 2010.
12. H. Astleitner, D. Leutner, *Designing instructional technology from an emotional perspective*, 2000.
13. T. Mettler, R. Pinto, *Serious Games as a Means for Scientific Knowledge Transfer - A Case From Engineering Management Education*, 2015.

14. V. Guillén-Nieto, M. Aleson-Carbonell, *Serious games and learning effectiveness: The case of it's a deal!*, 2012.
15. B. M. Winn, *The design, play, and experience framework*, 2009.
16. L. Annetta, *The I's have it: A framework for serious educational game design. Review of General Psychology*, 2010.
17. J. Brodkin, *How Unity3D Became a Game-Development Beast*, «<https://insights.dice.com/2013/06/03/how-unity3d-become-a-game-development-beast/>», 2003.
18. Unity3D, «<https://unity3d.com/unity/editor>».
19. Unity3D, *Unity User Manual*, «<https://docs.unity3d.com/Manual/index.html>».
20. Katane Lab, «<https://www.katanelab.it>».