

DOCUMENTAZIONE PROGETTO VITAS

A CURA DEGLI STUDENTI

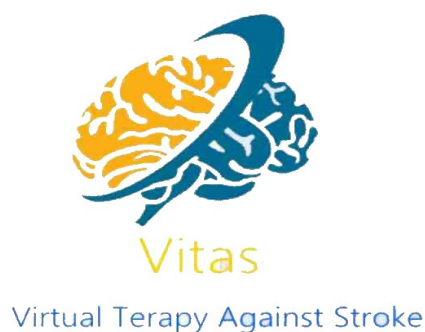
GIOVANNI MENEGOZZO

ENRICO MAGNABOSCO

CON LA SUPERVISIONE

PROF. PAOLO FIORINI

DOTT. DIEGO DALL'ALBA



Introduzione:

Il progetto Vitas nasce da un tirocinio universitario presso l'università di Verona. Vitas è composto da un insieme di applicazioni software basate su reali test fisioterapici e da alcuni Serious Game.

Questa relazione ha l'intento di spiegare in dettaglio il codice sorgente che troverete su <https://bitbucket.org/altairlab/vitas>.

Hardware e software utilizzato:

Il Progetto è stato interamente sviluppato in unity 3D nella versione 5.6. L'utilizzo di versioni successive ne può compromettere il funzionamento.

Il sistema operativo utilizzato è Windows 8.1;

Si necessitano inoltre del dispositivo hardware Leap Motion.

Il sistema funziona tramite molteplici sistemi di input (touchscreen e leap motion) e ogni applicazione al suo interno è configurato in maniera automatica.

Il sistema è compatibile con macOS e Windows ed è strutturato per essere espandibile ad altri sistemi di output.

Requisiti di sistema:

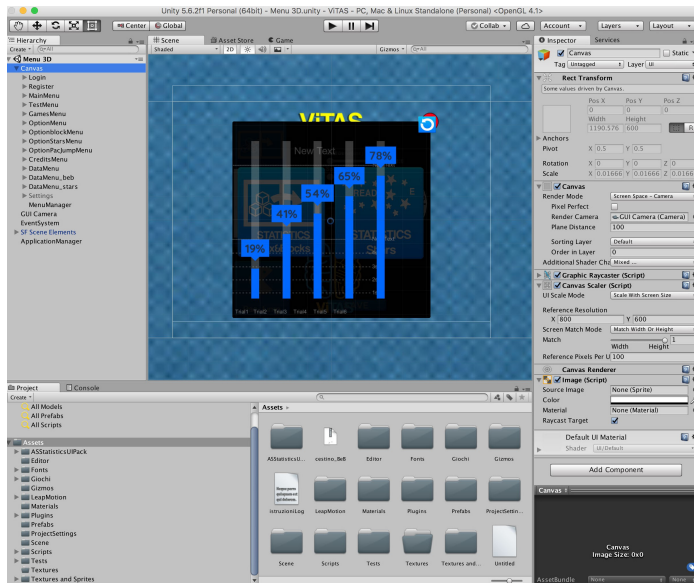
Il progetto funziona sia su piattaforma macOS che Windows.

Requisiti Consigliati:

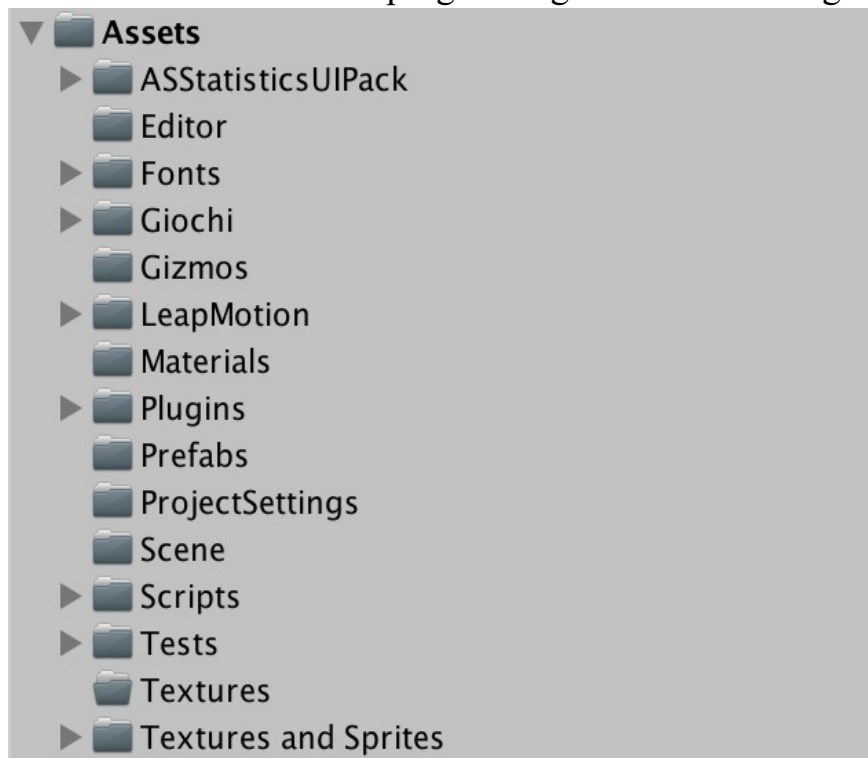
- Windows 7 o superiore, macOS Sierra
- 8GB di memoria Ram
- Processore i5 2.0Ghz
- Scheda grafica dedicata da almeno 1.5Gb
- Spazio libero su disco 2Gb
- [Unity](#) versione 5.0 o superiore.
- [Leap Motion](#) driver con ultima versione disponibile.

Organizzazione Cartelle:

Scaricare dalla [Repository](#) la cartella del progetto ViTAS e scompattarla. Avviare Unity e selezionare 'Apri Progetto' con destinazione la cartella ViTAS. Una volta avviato, Unity compilerà tutte le librerie mancanti. Il Progetto si presenta nel seguente modo:



All'interno della sezione Project si possono trovare tutti i file necessari a far funzionare correttamente il progetto organizzate come segue nella cartella Assets:

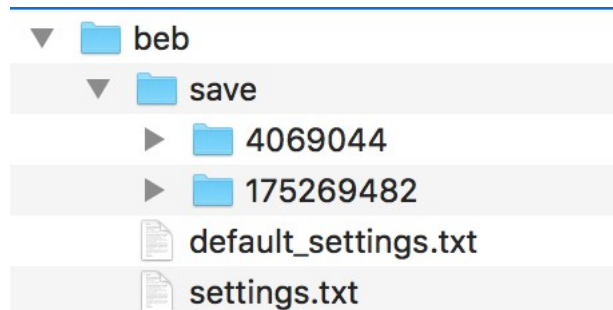


- ASStatisticsUIPack: contenente i file per la stampa degli istogrammi per la visualizzazione dei dati
- Giochi: file riguardanti i giochi presente nell'applicazione(FlappyCloud, LaneRunner, PacJump)
- LeapMotion: contiene le librerie per poter importare la Leap Motion all'interno del progetto
- Scene: scene utilizzate all'interno dell'applicazione (partenza: 'Menu 3D')
- Scripts: file di script in c# utilizzati nel menu, giochi e test
- Tests: file che riguardano i test presenti nell'applicazione(BlocksAndBox, Stelle_Test)
- Altre Cartelle: Contengono file di supporto per le vari scene come Textures, materiali o Prefabs.

Le cartelle principali sono 'Giochi', 'Tests' e 'Scripts'. Tutti i file di script del progetto sono scritti in C#.

I dati dei pazienti dopo essere stati criptati attraverso un codice identificativo sono salvati in base al nome del test/gioco e in base al codice identificativo.

Tutte le cartelle degli applicativi con salvataggi hanno un file 'default_settings.txt' che contiene le opzioni di default che si vogliono tenere all'avvio dell'applicazione, ed un file 'setting.txt' che contiene i settaggi in uso del paziente corrente.



La cartella 'Save' contiene i dati di ogni utente organizzati in cartelle con ID univoco.

Script Rilevanti:

Leap Reader: Si occupa di salvare i dati della mano (vedi documentazione in allegato Leap Log file).

Leap Motion Stats: Stampa le statistiche del leap reader a video.

Leap Motion Star Stats: Calcola le statistiche (posizione velocità e jerk) dello star.

Leap Motion Box Stats: Calcola le statistiche (posizione velocità e jerk) del BeB.

Star Stat: Calcola le statistiche (punteggi e tempi) del BeB.

Statistiche_beb: Calcola le statistiche (punteggi e tempi posizione) del BeB.

Menu: Si occupa della gestione del canvas dell'avvio degli script e del caricamento delle scene

Grabbable Object: è lo script assegnato a cubo che insieme allo script Grabbing hand assegnato alla mano facilita la presa degli oggetti. Questo script funziona tramite un ray-cast dalla mano al primo oggetto incontrato. Il Grabbing hand riconosce la posizione della mano controllando l'avvicinarsi di pollice ed indice. La funzione radius settata all'inizio del gioco setta la distanza del raggio minima affinché avvenga la presa dell'oggetto tra mano e oggetto più vicino individuato tramite ray-cast.

Collaboratori

Giovanni Menegozzo, Enrico Magnabosco, Diego Dall'alba