



**UNIVERSITÀ DI PISA**

DIPARTIMENTO DI

FILOLOGIA, LETTERATURA E LINGUISTICA

CORSO DI LAUREA MAGISTRALE IN INFORMATICA UMANISTICA

**TESI DI LAUREA**

Studio ed implementazione di un prototipo immersivo per la fruizione virtuale della  
letteratura: La Selva Oscura di Dante

CANDIDATO

Giovanni Giuliadori

RELATORE

Prof. Marcello Carrozzino

CORRELATORE

Ing. Franco Tecchia

ANNO ACCADEMICO 2020/2021

# Sommario

Introduzione.....	4
Capitolo 1, lo stato dell'arte.....	7
Breve storia della realtà virtuale.....	7
Varie applicazioni della virtualità.....	9
La <i>virtual reality</i> .....	9
Videogiochi ed educazione .....	10
Il libro e la virtualità .....	12
Dante nella Cultura Digitale.....	16
Capitolo 2, il prototipo per una letteratura immersiva .....	19
Descrizione task .....	19
Classificazione e definizioni utili .....	21
Utilizzo del medium videoludico.....	23
Capitolo 3, descrizione tecnica .....	27
Piattaforma Hardware .....	28
Descrizione Software utilizzati .....	28
<i>REAPER</i> e la gestione delle tracce sonore dei versi .....	29
Fuse .....	29
Blender.....	31
Mixamo .....	36
Unreal Engine 4.....	38
Programmazione in UE4 .....	43
Descrizione elementi unici.....	44
Descrizione elementi variabili e Mappe.....	61

<i>Screenshot</i> dell'applicazione.....	88
Conclusioni .....	93
Valutazioni esterne .....	93
Il questionario di valutazione e i risultati.....	93
Riflessioni sulle valutazioni ricevute .....	101
Possibili sviluppi e considerazioni finali .....	104
Bibliografia e sitografia .....	107

## Introduzione

*I giovani capiscono istintivamente l'ambiente presente, il dramma elettrico. Vivono miticamente e in profondità. Questa è la ragione della grande alienazione fra le generazioni.*

Marshall McLuhan,  
*Il medium è il messaggio*

L'idea di questo progetto nasce qualche anno fa, era il 2018 e il candidato stava affrontando l'esperienza triennale in Lettere Moderne; in questo contesto, durante lo studio per un esame riguardante la *Divina Commedia*, il suddetto è rimasto affascinato dalla potenza immaginifica del poema: quelle parole che, con tanta veemenza descrittiva, prendevano nella mente forma di immagini, scorci, personaggi ed azioni.

Un amore, quello per la letteratura, che non ha potuto fare a meno di confrontarsi, scontrarsi e quindi fondersi con un'altra viscerale passione, quella per i *videogame*. In quei giorni ci si pose una domanda, anche in seno a degli studi sull'argomento autonomamente effettuati in merito: «Ma è possibile che risulti così difficile accendere l'interesse delle nuove generazioni nei confronti di tanta potenza espressiva? Perché i giovani sembrano aver difficoltà ad interagire con i contenuti testuali e ad assorbirli, mentre non ne hanno quando questi "messaggi" (nell'accezione mcLuhiana) assumono forme multimediali?»

Nel tempo l'idea si è depositata lentamente assumendo una forma e, grazie agli studi specialistici effettuati, anche sostanza, per quanto virtuale; fino ad arrivare ad oggi con la creazione di un software per un'esperienza in realtà virtuale in merito al *I Canto* della *Divina Commedia* di Dante Alighieri. Questo è ciò che si intende presentare attraverso la discussione odierna, non un prodotto perfetto o sconvolgente alla vista, ma un germoglio concreto nato da quell'idea: le materie studiate sono state numerose, complesse e diramate (dalla sociologia, alla critica letteraria, fino ad arrivare alla programmazione, alla modellazione 3D e manuali specifici per utilizzare i diversi software); alcuni strumenti utilizzati necessitano di anni di esperienza e studi per poter essere correttamente utilizzati, tempo di cui, purtroppo, non si disponeva. Ma non di meno, non hanno vacillato la passione per gli argomenti, la disponibilità del professore e dei suoi assistenti ed un po' di testardaggine, che hanno comunque permesso lo sviluppo di un prototipo, per quanto amatoriale, comunque completo e

funzionante. Il tutto nella volontà di trovare una qualche modalità per trasmettere e comunicare ai cosiddetti *nativi digitali*, sfruttando il loro medium preferenziale, quei contenuti poetici, quei valori umani e quelle suggestioni con cui il *Sommo Poeta* arricchisce gli immaginari dei suoi lettori da settecento anni.

Il progetto, come accennato, vuole presentarsi come il prototipo per una *letteratura immersiva*, quindi non una semplice applicazione virtuale con contenuti letterari, ma qualcosa di più insito al senso stesso della materia umanistica; qualcosa che ne rispetti i momenti, le prassi, le definizioni ed i ritmi che autori e critici, nel corso di secoli, hanno elaborato. Lo scopo finale si basa sull'antefatto appena posto ed è quello di dare un nuovo respiro ad argomenti tanto necessari per la crescita degli individui e delle società, quali sono quelli della letteratura, consentendo ad essi di accedere ai nuovi canali comunicativi che formano la complessa trama degli universi virtuali; non attraverso una semplice *rimediazione*<sup>1</sup> della materia, ma applicando ad essa un ripensamento strutturale che possa farla adattare meglio alle necessità educative ed antropologiche delle nuove generazioni; andando, in tal modo, a creare un'esperienza che intende avvicinarsi il più possibile a quella concreta della "lettura di un testo", ma attraverso il linguaggio elettronico e multimediale delle tecnologie immersive.

Per dare una corretta contestualizzazione al lavoro in questione, si è pensato di strutturare lo scritto nel modo seguente:

- Stato dell'arte:

Con un primo capitolo si andranno a conoscere le tecnologie e le tecniche utili per comprendere lo stato di sviluppo attuale di vari argomenti inerenti alla virtualità. Ad una breve storia in merito ai dispositivi immersivi seguiranno esempi di applicazioni di tali mezzi su vari campi del sapere; successivamente, si analizzerà il mondo del *videogame* adattato all'educazione, portando il discorso a sviscerare i motivi della scelta e del successo di tale medium per trasmettere competenze, conoscenze o abilità; quindi, si analizzeranno le modalità con cui il libro è entrato nell'universo digitale e vari esempi di digitalizzazione di

---

<sup>1</sup> Cfr: Bolter e Grusin 2000

contenuti testuali; infine, in chiusura di questo primo capitolo, si valuteranno alcune soluzioni con cui, nello specifico, la *Divina Commedia* è stata convertita in contenuto virtuale.

- Prototipo per una letteratura immersiva:

Nel secondo capitolo si entrerà nel progetto vero e proprio; si spiegheranno quali erano le finalità che ci si era posti ed in che modo è stato definito il concetto di *letteratura immersiva*. Lo si farà andando ad analizzare il rapporto esistente tra libro e contenuto digitale al fine di trovare un connubio olistico tra i due nel rispetto e la valorizzazione del primo attraverso i canali del secondo. Si daranno definizioni e spiegazioni in merito a termini utilizzati e a scelte etimologiche; per giungere, quindi, a spiegare i motivi psicologici ed antropologici per cui il candidato ha deciso di utilizzare il medium del videogioco per veicolare un contenuto letterario.

- Descrizione tecnica:

Nel terzo capitolo verrà trattata la descrizione tecnica del progetto. Si passeranno al vaglio la struttura hardware e le applicazioni software che hanno reso possibile ultimare e caratterizzare la *Selva Oscura* virtuale. Quindi si giungerà alla spiegazione dettagliata sulla programmazione che permette il funzionamento del progetto in Unreal Engine 4; di questo verranno descritte tutte le componenti, in che modo funzionano, come comunicano, il ruolo che svolgono ed altri tecnicismi, quali: l'IA dei personaggi, l'interattività e varie altre questioni riguardanti la struttura del software.

- Conclusioni:

Questi ultimi paragrafi riguarderanno le valutazioni esterne che l'applicazione ha ricevuto da parte di alcuni volontari a cui è stata fatta provare l'applicazione ed è poi stato chiesto di compilare un questionario; da qui si analizzeranno le descrizioni in merito ai punti di forza e alle problematiche del progetto, andando a riflettere su possibili miglioramenti ed ottimizzazioni. In chiusura, si articoleranno diverse idee sugli sviluppi futuri per la *letteratura immersiva* e le varie modalità di implementazione; fino a giungere alle conclusioni finali sul lavoro effettuato.

## Capitolo 1, lo stato dell'arte

*Il libro è un'estensione dell'occhio...*

*I vestiti un'estensione della pelle...*

*I circuiti elettrici un'estensione del sistema nervoso centrale.*

*I mezzi di comunicazione, alterando l'ambiente, evocano in noi sintesi uniche di percezioni sensoriali. L'estensione di un qualsiasi senso altera il modo in cui pensiamo ed agiamo, il modo in cui percepiamo il mondo.*

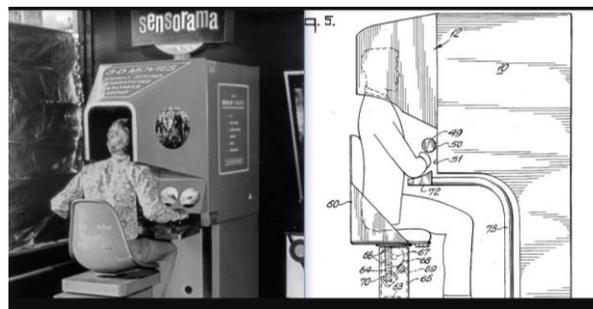
*Quando questi rapporti cambiano, cambiano gli uomini.*

Marshall McLuhan,

*Il medium è il messaggio*

### *Breve storia della realtà virtuale*

Il precursore di quei dispositivi che oggi sono conosciuti come *sistemi di realtà virtuale* risale al 1962 quando Morton Heilig sviluppò il *Sensorama*, un simulatore di corsa in motocicletta con dei sistemi in grado di stimolare contemporaneamente quattro canali sensoriali: la vista attraverso uno schermo, l'udito mediante altoparlanti stereofonici, l'olfatto con l'ausilio di un propagatore di aromi, nonché il tatto utilizzando dispositivi che simulavano le vibrazioni del mezzo, i cambi di temperatura durante la corsa ed il vento contrario. A questo ingegnoso, benché primordiale, esperimento mancava ancora un tassello fondamentale rispetto ai moderni sistemi virtuali, cioè l'*interattività*, intesa come la possibilità di cambiare le caratteristiche dell'ambiente simulato in funzione degli *input* dell'utente.



**Figura: *Sensorama* di Morton Heilig nel 1956<sup>2</sup>**

---

<sup>2</sup> Immagine presa da: <https://pbs.twimg.com/media/DFruE00XYAk0vqq.jpg>

Un altro passaggio importante per lo sviluppo dei sistemi virtuali è avvenuto nel 1965 con l'invenzione di *The Ultimate Display*, da parte di Ivan Sutherland, artefatto che può considerarsi il primo visore di realtà virtuale: un casco *head-mounted display* a visione stereoscopica, con *head tracking* ed uso di immagini generate al pc. Definita dal proprio inventore come «Una finestra su un mondo virtuale», l'idea non ebbe purtroppo successo a causa del peso e dell'ingombro del sistema tecnologico che doveva esser calato dal soffitto sull'utente e mantenuto da imbragature.



Figura: *The Ultimate Display* di Sutherland<sup>3</sup>

Negli anni '80 vi furono ulteriori progressi con l'invenzione del guanto virtuale, il *Data glove*, di Tom Zimmerman nel 1981, un dispositivo aptico utile ad interagire con gli ambienti virtuali in modo più naturale ed intuitivo; e, la *Virtual suit* nel 1989 sviluppata sempre da Zimmerman in collaborazione a Jaron Lanier, una tuta sensorizzata che consentiva di interagire con l'intero corpo all'interno della simulazione; nasce in questo modo il termine di *immersione* con il significato di "percezione di trovarsi in un mondo virtuale". Ma era ancora troppo presto per una proliferazione commerciale di queste tecnologie a causa dei costi di sviluppo proibitivi, i pochi contenuti e la carente qualità percettiva.

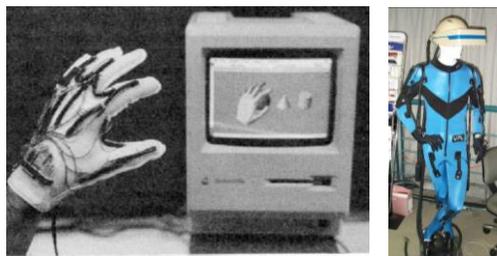


Figure: *Data Gloves* e *Virtual Suit*<sup>4</sup>

---

<sup>3</sup> Immagine presa da: <https://www.roadtovr.com/wp-content/uploads/2016/05/ultimate-display.jpg>

<sup>4</sup> Immagini prese da: 1) *Data Gloves*: <https://www.semanticscholar.org/paper/A-hand-gesture-interface-device-Zimmerman-Lanier/acd0214bdf7ca8afd5e6c3a5140bda13dde378d>

I tempi maturarono solo nel 2014 quando Facebook acquistò la società Oculus:

*In pratica, Facebook ha colto nella realtà virtuale il cuore di un nuovo medium esperienziale, in grado di sostituire il piccolo schermo come mezzo di comunicazione dominante dal punto di vista pubblicitario [...].<sup>5</sup>*

Tale iniziativa ha preso atto parallelamente allo sviluppo della tecnologia *smartphone* che ha consentito la produzione di strumenti a buone prestazioni con un costo abbordabile ed in schermi ristretti; proprio da ciò nasce l'idea casco immersivo stereoscopico commerciale, dall'intuizione di Palmer Luckey: un sistema con due schermi separati e giroscopio, con una qualità percettiva maggiore rispetto alle soluzioni precedenti ed al prezzo di poche centinaia di euro.<sup>6</sup>

## *Varie applicazioni della virtualità*

### *La virtual reality*

Si andranno ora ad esaminare quali sono le applicazioni acquisite dalle tecnologie e dai medium immersivi in vari ambiti, in modo da poter dare un'idea generale sugli sviluppi che hanno avuto tali mezzi nel corso del tempo. Per quel che riguarda le prime applicazioni della realtà virtuale, durante gli anni '70, esse nascono in ambito militare ed aerospaziale per la simulazione di situazioni pericolose per l'addestramento di piloti ed astronauti; ma oggi i settori dove si utilizza la *virtual reality*, sono molto più numerosi<sup>7</sup>:

- Design: Il VR viene sfruttato sia in modellazione che in esposizione di prodotti. Con il divenire tridimensionale dello spazio virtuale risulta più agevole la stima delle proporzioni, la valutazione degli spazi e la visualizzazione dell'artefatto finale. I prodotti possono essere rappresentati realisticamente e con scale appropriate riducendo in tal modo altresì i costi di sviluppo e produzione per la creazione di prototipi; in particolar modo, tali tecnologie sono

---

2) *Virtual Suit*: [https://upload.wikimedia.org/wikipedia/commons/a/a8/VPL\\_DataSuit\\_1.jpg](https://upload.wikimedia.org/wikipedia/commons/a/a8/VPL_DataSuit_1.jpg)

<sup>5</sup> Riva e Gaggioli, *Realtà Virtuali* 2019, pp. 14 - 15

<sup>6</sup> *Ivi* pp. 11-24

<sup>7</sup> Cfr.: *Ivi* pp. 37-61

utilizzate: nell'industria automobilistica, in architettura, in ambito manifatturiero e, negli ultimi anni, altresì nei processi produttivi attraverso la cosiddetta *manutenzione remota*.

- Educazione ed addestramento: utili per aiutare gli studenti nella comprensione di concetti, in particolare nelle discipline scientifiche, riuscendo a potenziare il processo di apprendimento esperienziale attraverso: simulazioni realistiche e controllate, uso del codice analogico per modelli astratti, verifica dell'apprendimento, coinvolgimento emotivo ed apprendimento sensorimotorio.
- Ricerca scientifica: utilizzata come strumento di visualizzazione spaziale di modelli e dati complessi, tale tecnologia sia permette un'iterazione diretta ed intuitiva con essi, sia di fungere da ambiente per simulare esperimenti scientifici. Altri ambiti di ricerca riguardano la psicologia e le neuroscienze, sempre per la gestione di simulazioni ambientali favorevoli e riproducibili per la ricerca sperimentale.
- Medicina e salute: La realtà virtuale qui si esprime in numerose tecniche avanzate utilizzate nei seguenti campi: l'addestramento chirurgico, la diagnosi, la riabilitazione neuromotoria, la terapia del dolore e la psicoterapia.
- Intrattenimento: Per l'industria dell'intrattenimento la VR ha costituito una vera e propria rivoluzione grazie a strumenti sempre più sofisticati in grado di garantire sempre una maggiore immersione e la produzione di contenuti di qualità sempre più alta; in particolar modo, i settori maggiormente impattati sono stati: i videogiochi, il cinema immersivo ed i parchi tematici.
- Marketing e commercio: Secondo alcune teorie economiche si sta sviluppando un mercato basato sull'esperienza come cardine dell'offerta, più che sui prodotti; l'obiettivo diventa coinvolgere i consumatori progettando esperienze a cui associare brand e prodotti; in questo l'immersività consentita dalla realtà virtuale gioca un ruolo importante in tali metodologie di vendita.

## Videogiochi ed educazione

Un altro ambito utile da investigare per le discussioni che verranno riguarda le modalità attraverso cui i videogiochi si avvicinano all'educazione e alla didattica. Numerosi sono gli scritti ed i dibattiti in merito all'efficacia dell'attività ludica, non solo virtuale, sull'apprendimento; da queste ricerche

emerge come il gioco nei bambini, negli adulti, come anche negli animali, mette in atto una serie di fenomeni psicologici utili ad apprendere al meglio una mansione o un concetto. L'attività ludica, in quanto elemento distaccato dalla "vita reale", agisce da *sandbox* per una sperimentazione libera e senza rischi di un'attività:

*[Nel gioco] si osserva una prevalenza dei mezzi sui fini: nella valutazione complessiva dell'attività ludica, il procedimento risulta più importante del risultato. [...] Altra caratteristica dell'attività ludica che può essere opportunamente sfruttata dal punto di vista dell'apprendimento è la riduzione del rischio di insuccesso. Come attività volontaria, inserita nel corso delle normali attività delle persone, ma ben distinta dalla vita cosiddetta seria, il gioco mette tra parentesi la sanzione negativa determinata da un insuccesso. [...] Il gioco poi funziona come straordinario motivatore offrendo una moratoria temporanea della frustrazione che consente ai partecipanti di applicarsi con convinzione e trasporto. Il divertimento collegato alle attività ludiche fa affrontare le situazioni di gioco con entusiasmo: anche compiti complessi che necessitano di un alto investimenti in termini di energie vengono affrontati con maggiore leggerezza o almeno con un atteggiamento che sposta in avanti la soglia del decadimento dell'attenzione e dell'impegno.<sup>8</sup>*

Da questi spunti si può intuire il valore del fenomeno della *gamification*<sup>9</sup> che consiste nello sfruttare queste componenti mentali insite nel gioco in ambiti non ludici per migliorare la *performance* nell'apprendimento o nel raggiungimento di determinati obiettivi mediante elementi quali: interattività, punti da accumulare, livelli da superare e ricompense da ottenere.

Nell'ambito della virtualità questo concetto prende forma attraverso la definizione di *serious game* come definita da Bergenson:

*Applicazioni interattive, generalmente mediate da computer, in assenza o in presenza di una componente hardware significativa, che hanno un obiettivo sfidante, sono divertenti e coinvolgenti da giocare, incorporano una qualche modalità di scoring e trasferiscono all'utente almeno un'abilità, una conoscenza o una disposizione che può essere concretizzata anche nel mondo reale.<sup>10</sup>*

---

<sup>8</sup> Mulargia 2016, pp. 141-142-143

<sup>9</sup> Deterding, et al. 2001

<sup>10</sup> Bergenson 2006

Queste modalità di insegnamento vengono ad oggi sfruttate in molti settori disciplinari, assumendo diverse forme<sup>11</sup>:

- Salute: Proposte improntate a promuovere il benessere fisico e psicologico dell'utente; finalizzate, ad esempio, a contenere l'ansia prima di un intervento, a sensibilizzare verso comportamenti salutari nella vita quotidiana, nella gestione di malattie croniche o come terapia del dolore. Oppure, altri utilizzi volti all'addestramento del personale medico nella simulazione di urgenze cliniche.
- *Advergaming*: Riguardano le soluzioni ludiche in ambito pubblicitario e di marketing, cioè giochi sviluppati, ad un qualche livello, per persuadere l'utente a determinati acquisti o comportamenti.
- Allenamento ed addestramento: videogiochi il cui obiettivo principale consiste nel miglioramento o all'acquisizione di qualche *skill* attraverso simulazioni pratiche.
- Educazione: Prodotti nei quali, durante lo sviluppo, si è posta principale attenzione sull'obiettivo educativo; utilizzati sia in pedagogia che in ambienti scolastici più maturi quali scuole superiori ed università.
- Scienza e ricerca: Improntati a favorire la raccolta, la visualizzazione e l'elaborazione dei dati.
- Produzione e lavoro: Si impiega la realtà videoludica a supporto del processo di realizzazione di un artefatto, o per ottenere una profittabilità personale.

## Il libro e la virtualità

La naturale trasformazione del libro nel contesto digitale consiste nell' *ebook*, utilizzabile attraverso: sistemi hardware di visualizzazione quali pc, tablet o smartphone; un sistema software di lettura ed un testo in *e-text* (un formato digitale comprensibile per la lettura umana). Lo scopo che si prefigge è quello di replicare il più fedelmente possibile l'esperienza di lettura di un libro cartaceo, consentendo azioni naturali quali lo scorrere le pagine e l'inserimento di segnalibri; ed altresì arricchire il contenuto con elementi tipici dei medium digitali attraverso link esterni, immagini digitalizzate ed elementi multimediali. Partendo da ciò si può sfociare nell'oceano della digitalizzazione di contenuti testuali: non solo libri narrativi e poetici, ma raccolte epistolari, saggi, commenti, ed antichi manoscritti.

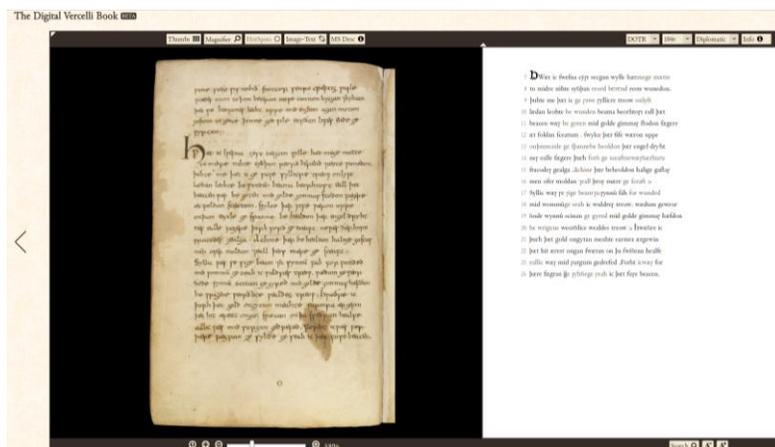
---

<sup>11</sup> Cfr.: Triberti e Argenton 2019, pp. 123-133

Attraverso tale tecnica un testo può essere reso disponibile su web per favorirne sia la facilità di fruizione e consultazione che, al contempo, la conservazione dell'originale: si pensi ad esempio agli antichi manoscritti medioevali; attraverso la loro digitalizzazione è possibile renderli disponibili per la ricerca velocemente e globalmente mediante immagini ad alta risoluzione, nonché si protegge il testo cartaceo da delicati spostamenti e manovre che rischiano di danneggiarlo. Alcuni esempi di testi digitalizzati sono:

- *Vercelli Book Digitale*: progetto che nasce nel 2003 su iniziativa del professor Roberto Rosselli per la digitalizzazione del *Codex Vercillensis*, una miscellanea di opere religiose risalente al X secolo. Si cita:

*Grazie a un'edizione digitale è finalmente possibile studiare un codice senza essere obbligati a spostarsi presso l'istituzione che lo custodisce, e che spesso pone notevoli restrizioni proprio riguardo la consultazione. Per certi aspetti, inoltre, un'edizione digitale di un manoscritto fornisce strumenti di indagine superiori rispetto alla tradizionale consultazione diretta: evidenziazioni di dettagli con illuminazioni particolari, ingrandimenti a piacere, filtri grafici per analizzare le immagini e la possibilità di effettuare ricerche testuali complesse costituiscono un indubbio vantaggio per ogni studioso.*<sup>12</sup>



**Figura: interfaccia di lettura del manoscritto digitalizzato: a sinistra un'immagine dell'originale, a destra la sua trascrizione in formato digitale. Sono disponibili varie opzioni per facilitarne la fruizione come link, zoom e note meta testuali**

<sup>12</sup> Rosselli Del Turco 2021

- *Progetto Mubil*: portato avanti dal laboratorio PERCRO dell'Istituto TeCIP della Scuola Superiore Sant'Anna di Pisa e la Norwegian University of Science and Technology e si occupa della progettazione e lo sviluppo di strumenti educativi digitali, con una visualizzazione tridimensionale interattiva, per la fruizione immersiva di libri storici.



**Figura: Augmented book<sup>13</sup>**

**(sinistra) ed esperienza in *cave* per una visita virtuale di un libro (destra), dal progetto Mubil**

*All'interno del progetto sono stati perciò sviluppati diversi concept per la consultazione virtuale di opere scritte. Tra questi l'Augmented Book è stato applicato al Kreuterbuch di Adam Lonitzer (1587), un trattato sulla distillazione medicinale, e consente di sfogliare virtualmente l'opera con uno schermo 3D e di fruire di contenuti quali immagini, filmati, animazioni 3D o narrazioni vocali, direttamente sulle pagine.<sup>14</sup>*

- *Codex Atlanticus*: Applicazione dedicata alla digitalizzazione del *Codice Atlantico* di Leonardo da Vinci; basa la sua efficienza sulla visualizzazione dell'informazione e permette di andare a cercare, filtrare, selezionare in base a vari parametri quali tematiche delle pagine o data di scrittura.<sup>15</sup>

---

<sup>13</sup> Percroart 2013:

*This concept, which we have defined augmented book preserves the physical shape of the book as such. The 3D environment is basically an ancient desk on top of which is placed the 3D model of the book which can be browsed with natural gestures (different interaction devices, ranging from traditional mouse/joystick to touch-screens or Kinect are supported). On each page a set of relevant hotspots, commonly spatially referenced in correspondence to the illustrations, are placed and can be interactively selected. Upon selection, a 3D model of the related object pops-up from the book and floats above it, with an accompanying audio narration explaining the tool and its functionalities.*

<sup>14</sup> Pipitone 2013

<sup>15</sup> The Visual Agency 2017

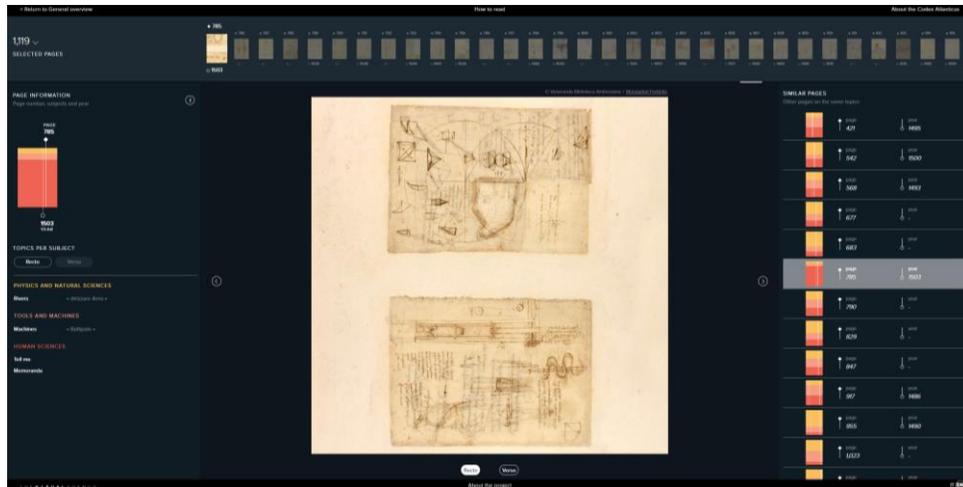


Figura: Esempio della visualizzazione di una pagina nel sito del *Codex Atlanticus*

- *Torch AR*: applicazione in *augmented reality* sviluppata dalla Torch che consente di aggiungere modelli 3d a libri reali.

*Attach augmented reality models to an art book. Provide an enhanced and expanded view of three-dimensional sculptures by expanding the model in space. With image tracking and an easy visual animation engine, Torch helps you quickly build AR apps that further enrich the experience of lush printed materials.* <sup>16</sup>

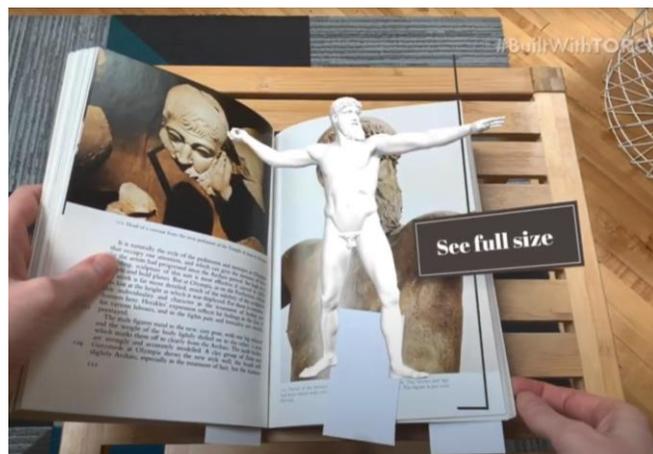


Figura: applicazione in AR della Torch che permette di visualizzare un modello 3D associato ad un libro d'arte

<sup>16</sup> Torch 2019

## Dante nella Cultura Digitale

Si andranno ora ad analizzare varie soluzioni e proposte per la fruizione di contenuti inerenti alla *Divina Commedia* di Dante Alighieri. Al di là dei progetti di digitalizzazione di opere e manoscritti resi fruibili sul web, le proposte per un Dante in 3D, con *rendering* non in *real-time*, sono presenti da anni. Un esempio è *I-Dante*, un progetto sviluppato da CarraroLab nel 2010 come *app mobile*:

*Il testo del poema di Dante, in italiano e inglese, si arricchisce con centinaia di immagini, consultabili mediante l'interazione tattile in una inedita miniatura digitale. Il viaggio di Dante viene ricostruito in 3d, grazie ai modelli virtuali dell'Inferno, del Purgatorio e del Paradiso e alla ricostruzione a 360 gradi di singoli ambienti, al video in motion graphics che rende tridimensionali le immagini di Dorè.<sup>17</sup>*



**Figura: *I-Dante* della CarraroLab**

Oppure il progetto *LaCom3Dia* della BricksLab<sup>18</sup>, con intenti più didattici che divulgativi rispetto l'esempio precedente, ha creato un'applicazione immersiva e multiplatforma con diversi ambienti virtuali, ciascuno per un canto dell'opera.

---

<sup>17</sup> Carraro Lab 2010

<sup>18</sup> BricksLab s.d.



**Figura: LaCom3Dia della BriksLab**

Anche *La Divina Commedia VR*<sup>19</sup> del gruppo ETT, un cortometraggio audiovisivo di 7 minuti in VR dedicato alla rappresentazione immersiva di alcuni momenti esteticamente significativi dell'Inferno.



**Figura: La Divina Commedia VR<sup>20</sup> del gruppo ETT**

Si possono trovare soluzioni per Dante anche in ambito videoludico, una risale al 1984, uscita per *Commodore64* con il titolo *Dante's Inferno*<sup>21</sup>, un *top-down adventure game* ambientato nel regno infernale e che vedrà il protagonista vagare nei vari gironi che compongono tale ambiente con l'intento di raccogliere oggetti per risolvere enigmi ed uscire incolume dagli intricati labirinti. O il

---

<sup>19</sup> Digital Strategy & Design s.d.

<sup>20</sup> Digital Strategy & Design s.d.

<sup>21</sup> Sterling 2008

meno datato lavoro omonimo pubblicato dalla Electronic Arts nel 2010 *Dante's Inferno*; un *hack and slash horror* che prende spunto dal personaggio di Dante e dal suo allegorico viaggio negli inferi proponendo una riscrittura storica (che vede Dante come soldato durante la Terza Crociata) e narrativa riprendendo dall'opera originaria le ambientazioni ed i personaggi.



Figure: *Dante's Inferno*, l'uscita di per Commodore64 (a sinistra) e la copertina del gioco della EA uscito nel 2010

## Capitolo 2, il prototipo per una letteratura immersiva

*Il passato andava a modo suo. Di fronte a una situazione completamente nuova, tendiamo sempre ad aggrapparci agli oggetti e ai sapori del passato più recente. Guardiamo il presente in uno specchietto retrovisore. Marciamo all'indietro verso il futuro.*

Marshall McLuhan,  
*Il medium è il messaggio*

### *Descrizione task*

Si vuole ora andare a descrivere, rispetto a quanto detto finora, dove il progetto qui presentato intende porsi. Il fondamento di partenza non giace nel mondo digitale, bensì nel libro; e la finalità che ci si è posti consiste proprio nel trasformare e riadattare questo medium in qualcosa di nuovo che possa rispecchiare e rispondere alle necessità dell'uomo contemporaneo. Riassumendo, si può affermare che il virtuale non è stato il punto di partenza, bensì il punto di arrivo. Importante da aggiungere, per delineare ulteriormente la forma di questo lavoro, è il fatto che si sono sì utilizzati software e strumenti specifici per lo sviluppo di contenuti in *real-time*, ma con l'intento di rielaborare il funzionamento di tali mezzi, non per dare al libro un assetto videoludico, bensì il contrario, cioè per adattare il medium virtuale all'opera scritta: in altre parole, cercando di sfruttare il videogioco come medium, non come contenuto. Attorno a questi presupposti si è articolato il lavoro: il voler rendere un'applicazione virtuale ed immersiva che fosse il più vicino possibile all'esperienza di "leggere" la *Divina Commedia*; proponendo immagini, suscitando emozioni ed inserendo note meta-testuali di contestualizzazione ed approfondimento.

In base a quanto detto, entrando nel merito, il cardine del lavoro è stato il testo della Selva Oscura, le sue rime, le metafore ed i ritmi narrativi; su questo è stata costruita l'infrastruttura virtuale. Letterariamente parlando, la *Divina Commedia* è un poema in terzine di endecasillabi in cui Dante, come narratore esterno, racconta del suo viaggio in prima persona al passato; questo assetto narrativo è stato virtualizzato proprio per mantenere coerenza quanto più stretta possibile con l'opera: una voce narrante esterna "legge" i versi mentre lo spettatore si ritrova a muoversi in un ambiente in cui avviene una scena con dei personaggi che si muovono, interagiscono e danno atto

visivo alla narrazione coerentemente con quanto il narratore sta dicendo. Inoltre, un altro elemento testuale aggiuntivo che si è voluto mantenere, necessario altresì per godere al meglio dell'opera nella sua complessità e profondità, sono le note ed i commenti meta-testuali; elementi presenti in quasi tutte le moderne edizioni del poema utili a esplicitare le metafore, le allegorie, le parole più desuete e le informazioni di carattere storico, linguistico o esegetico. Per la virtualizzazione di questi elementi si è fatto ricorso all'interattività attraverso delle "sfere informative": il giocatore, interagendo con tali elementi, può vederne visualizzato il contenuto in un apposito menù virtuale nel suo braccio sinistro, come verrà spiegato in seguito.



Da quanto finora emerso è possibile notare un'altra caratteristica inerente a questo progetto, cioè la sua natura di linguaggio *ibrido*. Infatti, con l'obiettivo di ricreare le scene quanto più strettamente coerenti con il testo, si è fatto ricorso a tre medium: non solo al libro, come contenuto e linguaggio, ed al videogioco, per l'interattività e l'immersività che dispone per il giocatore, ma anche al film per la regia della scena. Si hanno infatti dei modelli dei personaggi con animazioni che, attraverso istruzioni di programmazione, "recitano" la parte coerentemente con quanto viene narrato: sono state associate specifiche animazioni a determinati passi nel tentativo di incrementare il coinvolgimento emotivo dello spettatore trasmettendo, attraverso tali azioni "recitate" dalla IA, lo stato emotivo dei protagonisti (paura, sgomento, rasserenamento, inquietudine, e via dicendo).

Questo è stato l'intento nel suo complesso, il tentare di creare un linguaggio per la letteratura che possa, andando a prendere idee e spunti da altri medium, coinvolgere e far appassionare lo spettatore contemporaneo al contenuto letterario della *Divina Commedia* attraverso un codice visivo, immersivo ed interattivo al fine di esaltare i contenuti immaginifici insiti nell'opera dantesca e renderli al contempo comprensibili, apprezzabili e trasmissibili ai *nativi digitali*: più predisposti ad assimilare

contenuti sottoforma di codice multimediale (film, fumetti, serie tv e videogiochi) che non come testi scritti.

### *Classificazione e definizioni utili*

Innanzitutto, bisogna partire con qualche precisazione terminologica, e lo si fa iniziando con il dire che una difficoltà preliminare riscontrata è stata quella di trovare un'etichetta adatta al software sviluppato. L'applicazione potrebbe provare a rientrare nella definizione di *gioco* come intesa dallo storico olandese Huizinga:

*Considerato per la forma si può dunque, riassumendo, chiamare gioco un'azione libera: conscia di non essere presa "sul serio" e situata al di fuori della vita consueta, che nondimeno può impossessarsi totalmente del giocatore; un'azione a cui in sé non è congiunto un interesse materiale. Da cui non proviene vantaggio, che si compie entro un tempo ed uno spazio definiti di proposito, che si svolge con ordine secondo date regole, e suscita rapporti sociali che facilmente si circondano di mistero o accentuano mediante travestimento la loro diversità dal mondo solito.<sup>22</sup>*

Il *gioco*, quindi, come elemento libero ed esterno alla vita ordinaria che diventa momento di cultura e crescita per l'individuo sembra calzare con l'idea con cui si è sviluppata l'applicazione; ed effettivamente esiste, nel vasto cosmo dei generi videoludici, una definizione che sembrerebbe ben descrivere il lavoro qui proposto, cioè quella di *serious game*: soluzioni ludiche, come precedentemente definito, sfruttate al fine di veicolare un messaggio, impartire una lezione o garantire un'esperienza formativa o di sensibilizzazione. Ma, anche attraverso questa designazione, non si riesce a far rientrare pienamente l'esperienza in questione nella nozione di *gioco* espressa da Huizinga, a causa della mancanza di una componente essenziale, quella di *gara*:

*Che cos'è vincere? Che cos'è "vinto"? Vincere è "risultare superiore" nell'esito d'un gioco. [...] Si gareggia o si gioca "per qualcosa". È in prima e ultima istanza la vittoria stessa per la quale si lotta o si gioca, ma unite a quella vittoria vi sono tante maniere in cui essa è celebrata.<sup>23</sup>*

---

<sup>22</sup> Huizinga 2002, p. 17

<sup>23</sup> *Ivi*, p. 61

Si è infatti deciso di mantenere, a favore dell'aspetto letterario ed esegetico, l'autoreferenzialità dell'azione di "leggere un libro"; un gesto, appunto, che non si compie per competizione, sfida o vittoria, ma per se stessa. Quindi, seguendo questo filo logico, leggere non è un gioco (così come non lo è guardare un film) e lo stesso si è deciso, in fase di sviluppo, per quest'esperienza nel tentativo di avvicinare il più possibile l'utente alla predisposizione mentale che avrebbe se volesse "leggere" la *Divina Commedia*, non per trionfare su qualcosa o qualcuno, ma per passione mossa da libera volontà.

Proprio per questo, si potrebbe optare, come nomenclatura più contestuale al lavoro, non il termine di videogioco bensì di *simulazione* virtuale:

*In order to identify the essential attributes of simulation, we focused on writers who used simulation to address learning objectives. Four attributes were identified: a model of reality defined as a system; a dynamic model; a simplified model; and a model that has fidelity, accuracy and validity. A fifth attribute can be added to this list: the simulation should address directly the learning objectives. [...] Upon examination of their essential attributes, it is clear that games and simulations are distinctive concepts. A game is a fictitious, whimsical or artificial situation in which players are put in a position of conflict. [...] On the contrary, simulation is a simplified, dynamic and precise representation of reality defined as a system.<sup>24</sup>*

Con la precisazione necessaria che qui con "realtà" non si intende la realtà effettiva ed esperibile, ma la realtà letteraria estrapolata dal libro; quindi, la simulazione di una realtà narrativa. Nel fare tali considerazioni si vuole inoltre far emergere un altro concetto e cioè quello della soggettività di tale realtà narrativa: il candidato, andando a leggere ed immaginare la *Divina Commedia* per quest'applicazione ne ha ricavato delle immagini puramente soggettive attraverso l'interpretazione del contenuto testuale. Per questo risulta poco congruo anche il termine di *simulazione*, proprio perché non c'è oggettività scientifica sui simboli mentali evocati dalla lettura di un passo, non c'è una realtà univoca; si è pensato di poter parlare quindi di *interpretazione virtuale* per la classificazione dell'elaborato in questione ed ulteriori considerazioni in merito verranno trattate nel capitolo riguardo le valutazioni esterne.

---

<sup>24</sup> Sauv , Renaud e Marquis 2007

## *Utilizzo del medium videoludico*

Nonostante quanto appena detto in merito alla definizione, rimane comunque molto saldo il collegamento con il videogioco nelle sue prassi, nei suoi strumenti e nei processi psicologici che instaura nell'utente; il quale, nel corso di questo lavoro, verrà spesso definito con il termine di *giocatore* per comodità terminologica. Infatti, come si è già anticipato, si è voluto utilizzare il videogioco solo come medium per veicolare il contenuto narrativo, e lo si è fatto per i motivi che si andranno ora ad analizzare.

Innanzitutto, perché il pubblico a cui questo lavoro vuole fare riferimento e con cui vorrebbe cercare di comunicare sono i cosiddetti *nativi digitali* secondo la definizione di Prensky:

*Today's students are no longer the people our educational system was designed to teach. [...] Today's students represent the first generations to grow up with this new technology. They have spent their entire lives surrounded by and using computers, videogames, digital music players, video cams, cell phones, and all the other toys and tools of the digital age. [...] It is now clear that as a result of this ubiquitous environment and the sheer volume of their interaction with it, today's students think and process information fundamentally differently from their predecessors.[...] But the most useful designation I have found for them is Digital Natives. Our students today are all "native speakers" of the digital language of computers, video games and the Internet.*<sup>25</sup>

Quindi dei soggetti che si sono adattati a quegli strumenti tecnologici con cui si sono continuamente interfacciati e che, come è sempre avvenuto con l'avvento di un nuovo medium (sia anche la scrittura), sono stati cambiati nei comportamenti, nelle modalità di pensiero e nei linguaggi; si andrà a spiegare in che modo partendo dalla comprensione del medium in questione.

Le tecnologie digitali si basano su una serie di fattori quali:

- 1) La variabilità: le informazioni digitalizzate, essendo composte di numeri ed operazioni algebriche possono essere facilmente replicate, modificate e compresse.
- 2) La multimedialità: Nei cosiddetti nuovi media vengono generalmente coinvolti ed integrati più canali sensoriali.

---

<sup>25</sup> Prensky, *Digital Natives, Digital Immigrants* 2001

- 3) Interattività: La fruizione delle informazioni non è né rigida né lineare e dipende dalla volontà dell'utente.
- 4) Automazione: Molte operazioni necessarie per attingere alle informazioni sono gestite automaticamente da apparati software e hardware; componenti con le quali un utente può comunicare attraverso interfaccia.<sup>26</sup>

Queste caratteristiche hanno generato quella che viene definita “società della rete”<sup>27</sup> che si distingue dall'assetto sociale precedente per la trasformazione nella percezione spaziale e temporale dei suoi individui; in particolare una *frammentazione temporale*, dove ogni contenuto può essere acceduto in qualsiasi momento senza necessità di una fruizione lineare, ed una conseguente discretizzazione degli spazi o *spazio dei flussi*, caratterizzato dalla fruizione dell'informazione in pacchetti discreti, sempre disponibili ed autonomi (indipendenti dagli altri per essere compresi). Da ciò consegue un cambiamento delle modalità di fruizione della conoscenza che ha messo in crisi il testo scritto: costituito da elementi puramente visivi e simbolici, nonché disposti linearmente nel tempo e nello spazio, contrasta con le caratteristiche precedentemente elencate dei canali comunicativi della *società di rete* che, al contrario, privilegiano informazioni multimediali, interattive e discretizzate.

Ed è qui che, secondo alcuni autori, possono intervenire i videogiochi:

*Questo «spazio dei flussi» e questo «tempo senza temporalità» costituiscono il contesto degli universi immateriali e delle realtà virtuali in cui interagiamo, lavoriamo, comunichiamo, trascorriamo il nostro tempo libero. I codici di tale contesto penetrano nella società e forgianno la nostra cultura in molteplici modi. Tra queste modalità di socializzazione ai nuovi codici, hanno svolto un ruolo fondamentale i videogiochi e gli universi tecno-ludici trasmessi dai loro programmatori. Tali universi stanno in un certo senso re-incantando il mondo, proponendo risposte assai efficaci ad alcune tra le cui incombenti necessità dell'uomo contemporaneo, al suo bisogno di ridurre le distanze dal mondo circostante e dagli altri, di frenare alcune delle più drammatiche conseguenze del processo di individualizzazione.<sup>28</sup>*

---

<sup>26</sup> Cfr: Riva, *Nativi Digitali - Crescere ed apprendere nel mondo dei nuovi media* 2019, pp. 24-25

<sup>27</sup> Castells 2002

<sup>28</sup> Pecchinenda 2010, pp. 45-46

Queste “risposte” conseguono da alcune specifiche proprie del medium videoludico che impattano nella psiche del fruitore. Innanzitutto, bisogna definire il fenomeno di *immersione*; quella situazione percettiva che permette, a chi si interfaccia con un ambiente virtuale, di “sentirsi lì”. Il rapporto tra un giocatore ed il suo *avatar* (affascinante termine di derivazione aramaica e traducibile con “discesa” o “incarnazione”) è tanto profondo in questo medium che, successivamente alla sconfitta del proprio personaggio durante un’esperienza videoludica, il giocatore esclama: “Sono morto!”. Questo avviene per numerosi motivi che ruotano, appunto, attorno al concetto di *immersione*, che avviene quando un videogioco funziona e coinvolge: le interfacce tecnologiche (che siano *joystick* o schermi) sembrano scomparire o, meglio, entrare in simbiosi come fossero un’estensione naturale del corpo umano, che, in questo modo, «si fa tecnologia»<sup>29</sup>. Per approfondire, si possono distinguere due tipi di interazioni con strumenti, intesi in senso generico, quella definita *azione mediata diretta*, in cui si sfrutta il movimento del corpo per compiere un’azione mediante lo strumento (come utilizzare la racchetta da tennis per colpire una pallina); e l’*azione mediata indiretta*, in cui attraverso il corpo si controlla uno strumento che a sua volta agisce su un oggetto terzo (è il caso del *joystick* attraverso cui si modifica il mondo virtuale in un videogioco). Entrambe queste azioni comportano un’incorporazione dello strumento all’interno del modello mentale dell’utilizzatore:

*In questo caso l’apprendimento produce l’incorporazione dell’artefatto all’interno dei campi dello spazio peri-personale del soggetto alterandone i confini. Per esempio, una serie di studi ha dimostrato che imparando a giocare a tennis e riuscendo ad usare intuitivamente la racchetta per prendere la pallina, il campo visivo dei nostri neuroni bimodali dell’area premotoria F4 si allungano di una lunghezza corrispondente alla racchetta.<sup>30</sup>*

Ma, a oltre questa reazione, le *azioni mediate indirette* ne comportano un’altra:

*Il soggetto muove il proprio joystick (artefatto prossimale) per guidare il proprio avatar (artefatto distale) in modo da raggiungere il tesoro e completare la missione. In questo caso, l’apprendimento, oltre a produrre l’incorporazione del joystick all’interno del campo dello spazio peri-personale del soggetto, genera un nuovo campo centrato intorno all’avatar. [...] In altre parole, la costruzione di modelli simulativi che consentono di controllare un artefatto distale (avatar) attraverso la manipolazione intuitiva dell’artefatto prossimale (joystick), oltre*

---

<sup>29</sup> *Ivi*, pp. 56-57

<sup>30</sup> *Ivi*, p. 87

*all'incorporazione del joystick, produce anche l'incarnazione nello spazio in cui l'avatar si trova: il soggetto, per la durata dell'azione mediata, è presente nello spazio virtuale in cui si trova l'artefatto distale.<sup>31</sup>*

Da qui scaturisce il concetto di *telepresenza*, letteralmente “presenza a distanza”: se l'esperienza virtuale che si sta vivendo ha delle regole coerenti (che siano logiche e comprensibili, per quanto immaginarie) e se l'utente le accetta temporaneamente come vere in sostituzione a quelle reali che governano il mondo (*sospensione dell'incredulità*); il corpo umano diventa un tutt'uno con la tecnologia e risulta possibile trasferire la propria presenza all'interno dell'*avatar*, muovendosi ed interagendo con il mondo virtuale come se si fosse, a tutti gli effetti, in una realtà a sé stante.

Queste sono le modalità e le funzioni psicologiche che agiscono nel videogiocatore durante l'esperienza del *gaming* e proprio queste si sono volute sfruttare nell'ideare quest'esperienza. Tutto ciò nel tentativo di veicolare contenuti letterari attraverso il medium videoludico andando ad attivare tali fenomeni di coinvolgimento ed immersione con la finalità di riuscire ad instaurare un canale comunicativo consono alle trasformazioni antropologiche dell'era elettronica e quindi ad accendere nello spettatore la fascinazione per quelle parole, quei messaggi e quelle immagini.

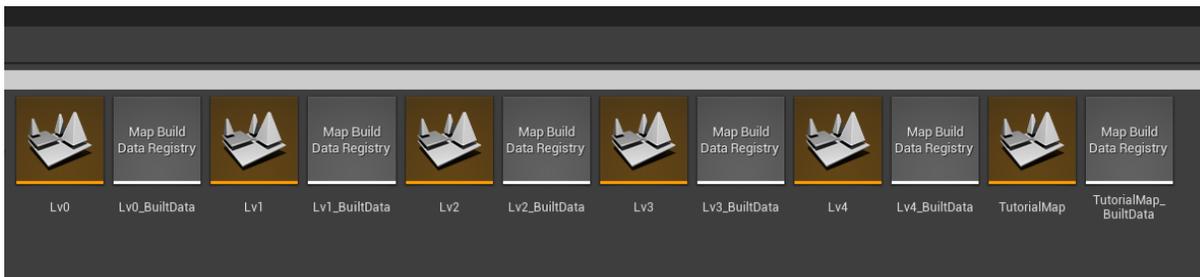
---

<sup>31</sup> *Ibidem*

## Capitolo 3, descrizione tecnica

Si andrà ora ad analizzare come si è elaborato e strutturato il *software* in questione; seguirà una descrizione generale dell'applicazione e poi si scenderà nella sua analisi dettagliata nel descrivere i programmi utilizzati, i metodi e le funzioni sviluppate.

Il problema principale che si è dovuto risolvere ha riguardato le modalità in cui strutturare la linea narrativa del *Canto I* del poema dantesco. Andando ad analizzare la cantica suddetta si è notato che ci sono delle discontinuità spaziali e temporali nella struttura della fabula, che possono considerarsi nuclei espositivi a sé stanti; si è quindi deciso di mantenere e sfruttare questi separatori “naturali” del testo per suddividere il progetto in diverse parti, dette “Livelli” o “Mappe” (attenendosi alla terminologia utilizzata da Unreal Engine 4, il motore grafico utilizzato). In particolare, le mappe utili alla narrazione sono quattro, a queste se ne aggiungono una di *tutorial* per spiegare i comandi necessari ad interagire con il mondo virtuale, ed un “Livello 0”, che funge da *hub* centrale per raggiungere le altre sezioni o chiudere il software.



I versi del *Primo Canto*, in base alla struttura narrativa estratta, sono stati suddivisi in questo modo:

- “Lv1”, versi da 1 a 9: Inizia il poema, Dante si ritrova a vagare all’interno della Selva Oscura.
- “Lv2”, versi da 10 a 27: dopo aver girovagato per la Selva tutta la notte, arriva l’alba ed il protagonista giunge ai piedi del *Diletto Colle* dietro il quale sta sorgendo il sole che gli ridà coraggio.
- “Lv3”, versi da 28 a 60: Nel tentativo di scalare il monte Dante viene bloccato dalle tre fiere, in successione: una lonza, un leone ed un lupo. La vista è tanto terribile da spingere il Sommo Poeta a rifuggire nella Selva Oscura.
- “Lv4”, versi da 61 a 136: Nel suo capitolare Dante incontra l’anima di Virgilio che lo soccorre e gli dà spiegazioni sul viaggio che dovrà affrontare per espiare i suoi peccati e poter giungere al

cospetto di Dio. Finisce il primo canto con i due poeti che partono insieme verso il regno degli inferi.

## *Piattaforma Hardware*

Si è deciso di sviluppare il software per Oculus Quest 2<sup>32</sup>, un casco per la realtà virtuale prodotto dalla Facebook Technologies, una divisione del gruppo Meta di Mark Zuckerberg; lo si è scelto in quanto prodotto commerciale di buona fascia.

Oculus Quest 2	
	
Oculus Quest 2 headset and controllers	
<b>Developer</b>	Facebook Technologies, LLC
<b>Type</b>	Virtual reality headset
<b>Release date</b>	October 13, 2020
<b>Lifespan</b>	2020-present
<b>Introductory price</b>	US\$299 (128 GB) US\$399 (256 GB)
<b>Operating system</b>	Oculus Quest system software, based on Android source code. (Android 10) <sup>[1]</sup>
<b>System on a chip</b>	Qualcomm Snapdragon XR2
<b>Memory</b>	6 GB
<b>Storage</b>	64 GB, 128 GB, 256 GB
<b>Display</b>	LCD 1832 x 1920 per eye @ 120 Hz <sup>[2]</sup>
<b>Graphics</b>	Adreno 650
<b>Input</b>	6DOF inside-out tracking through 4 built-in cameras
<b>Camera</b>	4 cameras
<b>Connectivity</b>	USB-C Bluetooth Wi-Fi
<b>Online services</b>	Oculus Store
<b>Mass</b>	503 g (17.7 oz)
<b>Related articles</b>	Oculus Rift S, Oculus Quest
<b>Website</b>	Official website <a href="#">[3]</a>

**Figura: Caratteristiche tecniche dell'Oculus Quest 2<sup>33</sup>**

Il casco ha un sistema operativo basato su Android, quindi, per poter far funzionare l'applicazione è stato necessario aggiungere *plug-in*, ambienti di sviluppo ed appositi *SDK* per la comunicazione tra il linguaggio detto e quello Windows del pc che si è utilizzato.

## *Descrizione Software utilizzati*

Si sono altresì utilizzati diversi strumenti software, che verranno ora descritti brevemente nelle loro funzionalità e nell'utilizzo che se n'è fatto all'interno del progetto in questione.

---

<sup>32</sup> Facebook Technologies 2020

<sup>33</sup> Immagine presa da: [//en.wikipedia.org/wiki/Oculus\\_Quest\\_2](https://en.wikipedia.org/wiki/Oculus_Quest_2)

## REAPER e la gestione delle tracce sonore dei versi

Come voce narrante per la *Divina Commedia* si è scelta l'interpretazione recitativa di Vittorio Gassman pubblicata sulla piattaforma di YouTube<sup>34</sup>; successivamente, attraverso l'estensione di Google Chrome *Audio Capture* si è stati in grado di salvare l'audio del video in formato ".mp3" da poter poi pulire e tagliare nello strumento di *editing* audio REAPER<sup>35</sup>. Da qui, infine, lo si è esportato in formato ".WAV" in UE4 pronto per l'uso.

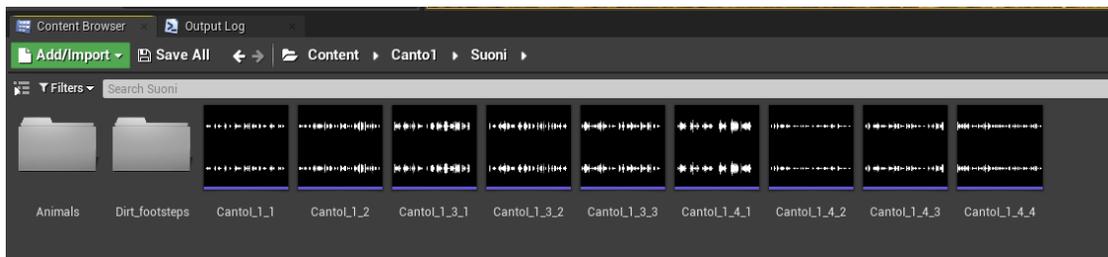


Figura: tracce audio in UE4

## Fuse

Fuse è un software di proprietà della Adobe, scaricabile gratuitamente dalla piattaforma Steam<sup>36</sup> ed utilizzabile se si è in possesso di un account Adobe; è pensato per la creazione e la personalizzazione di modelli umanoidi 3D, ed infatti è stato utilizzato per realizzare i modelli poligonali di Dante e Virgilio.

*"Fuse is a standalone 3D character creator that enables you to make unique characters to use in your game."* <sup>37</sup>

---

<sup>34</sup> PagliaQuotidiana 2012

<sup>35</sup> REAPER è un'applicazione che funge da *workstation* audio digitale e *sequencer* MIDI.

Cockos Incorporated 2021

<sup>36</sup> *Steam* è una piattaforma sviluppata da Valve Corporation che si occupa di distribuzione digitale, di gestione dei diritti digitali, di modalità di gioco multigiocatore e di comunicazione.

Valve Corporation 2021

<sup>37</sup> Definizione dell'applicazione qual è nel negozio virtuale della piattaforma Steam.

Link: <https://store.steampowered.com/app/257400/Fuse/>

Lo strumento si presenta quale un *character editor* con numerose possibilità di modifica e personalizzazione, dalle varie parti anatomiche, fino ad arrivare ai vestiti (gestiti con dei *preset* generici e senza possibilità di modifica della forma dei modelli) e le relative *texture* (le quali consentono di definire la trama del tessuto dell'abito, i colori della stoffa e degli accessori, nonché altre numerose caratteristiche che vanno dalle proprietà di rifrazione luminosa, all'indice di specularità).

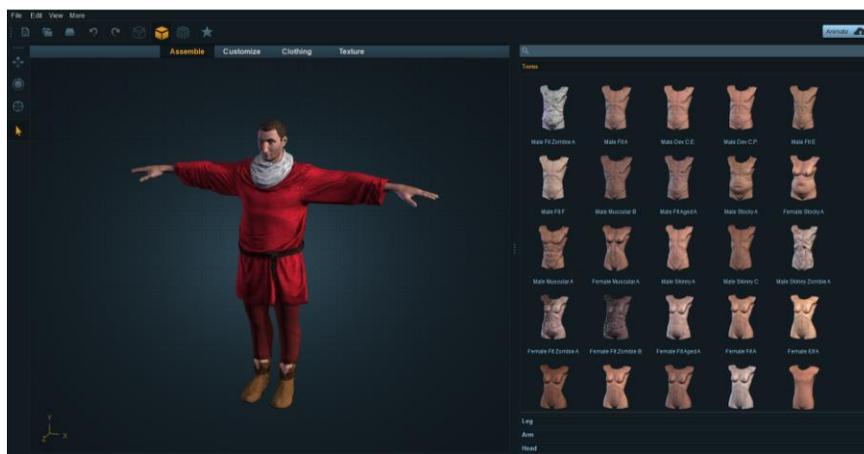


Figure: Interfaccia di Fuse ed esempi di personalizzazione

Una volta creato il personaggio desiderato lo si è esportato in formato “.Obj” per poterlo ulteriormente personalizzare in Blender.

## Blender

Blender è un software libero e multiplatforma di modellazione, rigging, animazione, montaggio video, composizione, rendering e texturing di immagini tridimensionali e bidimensionali. Dispone inoltre di funzionalità per mappature UV, simulazioni di fluidi, di rivestimenti, di particelle, altre simulazioni non lineari e creazione di applicazioni/giochi 3D.”<sup>38</sup>

L’analisi delle vaste e complesse possibilità consentite da Blender per lo sviluppo di ambienti, oggetti e personaggi virtuali va oltre le finalità di questo scritto, ma si porteranno due semplici esempi dell’utilizzo che se ne è fatto per il completamento di questo lavoro: la creazione di un libro animato e la modellazione di una corona di alloro da aggiungere ai modelli dei personaggi di Dante e Virgilio per dare loro ulteriore caratterizzazione.

### Creazione di un libro animato

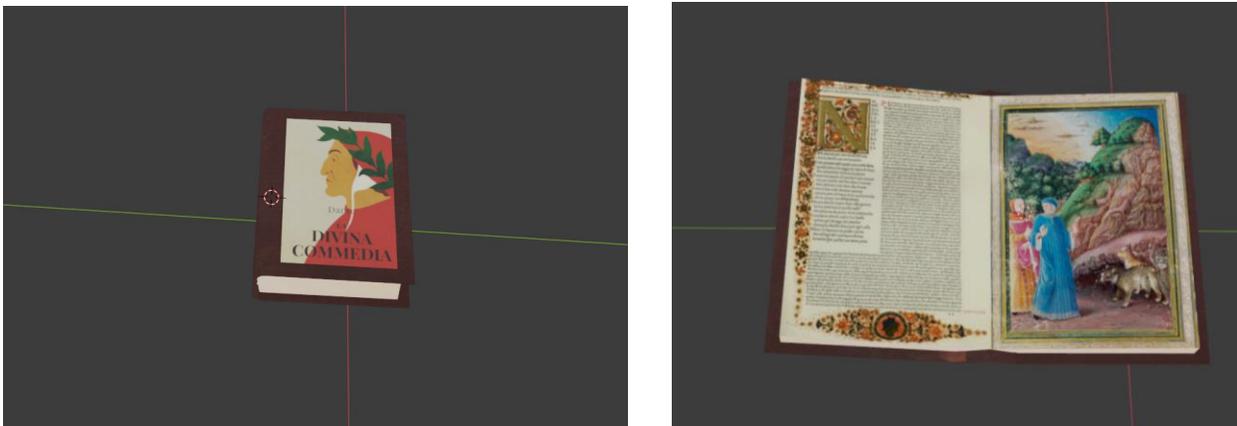


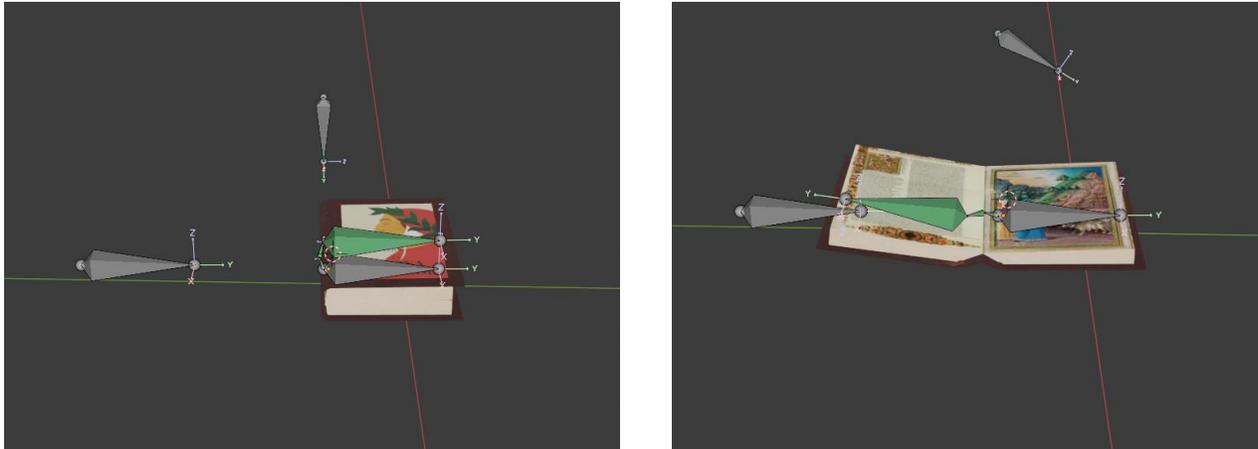
Figure: La *mesh* del libro precedentemente modellata e texturizzata in due posizioni: chiuso e aperto

La tecnica di animazione 3D utilizzata per questo progetto è il *rigging* o *skeletal animation* e necessita di due componenti fondamentali: una *mesh* o *skin*, cioè il modello dell’oggetto o del personaggio da animare, ed un insieme gerarchico di elementi connessi tra loro, detti ossa, che andranno a formare il suddetto scheletro, *rig* o armatura.

---

<sup>38</sup> Definizione da Wikipedia: [https://it.wikipedia.org/wiki/Blender\\_\(programma\)](https://it.wikipedia.org/wiki/Blender_(programma))

La tecnica di *rigging* vera e propria consiste nell'associare i vertici della *mesh* alle ossa dello scheletro attraverso un valore pesato; in modo che, andando ad agire su un osso tramite trasformazioni euclidee (rotazioni, traslazioni o ridimensionamenti) verranno modificati conseguentemente anche i vertici del modello a questo associati. In aggiunta, mediante l'utilizzo di vincoli reciproci tra gli elementi dello scheletro, è possibile generare catene di movimenti partendo dal trasformare un singolo elemento.

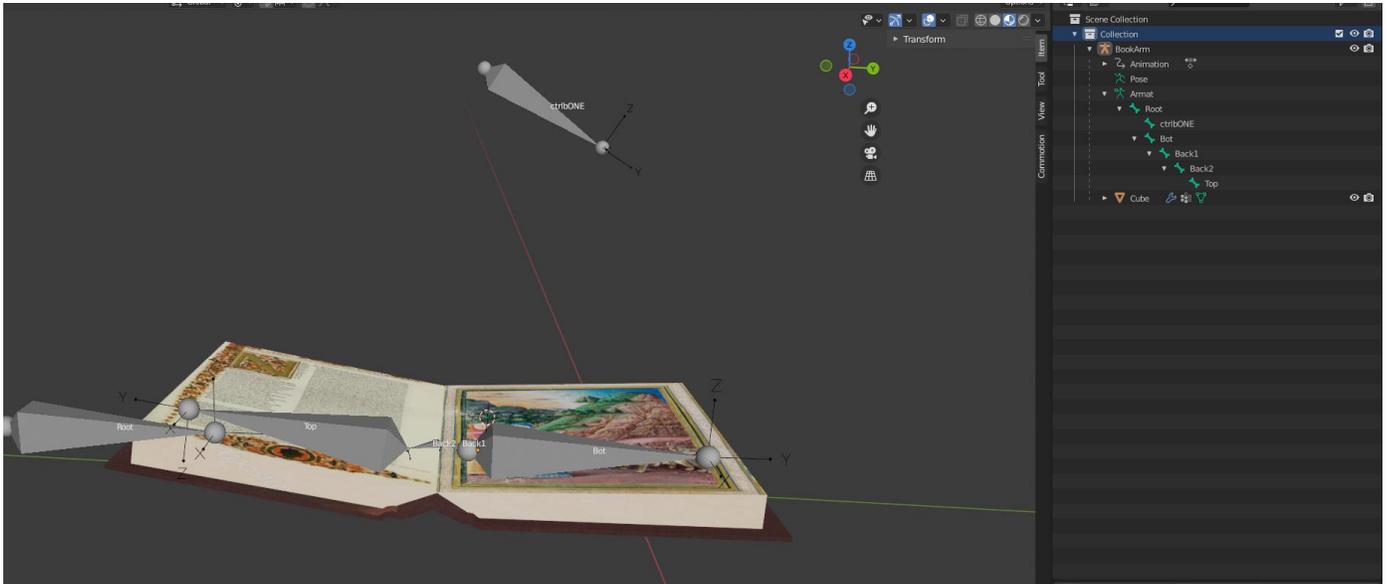


**Figura: il libro sia chiuso che aperto con lo scheletro visibile**

Ad esempio, questo libro 3D è formato da quattro ossa fondamentali, uno per la copertina superiore (“top”), due per le giunture nel retro del libro (“back1” e “back2”) ed uno per la copertina inferiore (“bot”), in aggiunta un osso di controllo “Ctrlbone” ed un osso radice “root”<sup>39</sup>. Per gestire l’animazione di apertura è stato imposto al “top” un vincolo di *copy rotation* dell’asse x rispetto al suo superiore gerarchico “back2”, a sua volta questo ha un medesimo vincolo rispetto a “back1” che, infine, dipende in ugual modo all’osso definito “CtrlBone”; in tal modo, andando ad agire sulla rotazione sull’asse x di quest’ultimo elemento si genererà una catena di rotazioni che andranno ad aprire la *mesh* del libro. Infine, si dispongono le due pose “chiuso” ed “aperto” in due diversi momenti, detti *keyframe*, di una *timeline* per creare l’animazione per interpolazione automatica tra le due pose in un arco temporale e creare così il movimento di apertura e di chiusura.

---

<sup>39</sup> Necessario per la successiva esportazione in UE4 che prende come formato di importazione di armature solo quelle con un solo osso alla radice.

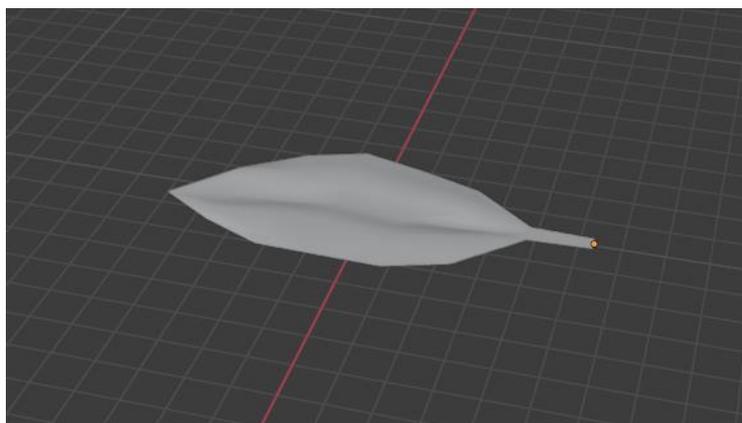


**Figura:** A destra dell'immagine è possibile cogliere i rapporti gerarchici reciproci tra le ossa che compongono lo scheletro

## Creazione della corona di alloro

Entrambi i personaggi umanoidi che compaiono nel *I Canto* della *Divina Commedia* (Dante e Virgilio) sono provvisti di una corona di alloro come simbolo di glorificazione poetica; si è usato Blender proprio per andare a creare questo elemento caratterizzante.

Innanzitutto, si è modellata la forma grezza della foglia di alloro che andrà a costituire, attraverso la sua ripetizione, la corona.



Quindi si è colta una foglia di alloro dal giardino di casa e le si è scattata una foto, successivamente si è andati ad isolare il soggetto in Adobe Illustrator<sup>40</sup>; l'immagine così ottenuta, esportata in file “.png” (con sfondo trasparente), andrà a costituire l'immagine *texture* da associare al modello.



**Figure: a sinistra la foto della foglia, a destra il soggetto scontornato mediante Adobe Illustrator**

Si è quindi tornati in Blender e si è applicato tale elemento al modello attraverso un processo definito *UV mapping* che consiste nel far combaciare i vertici della *mesh* poligonale con l'immagine in questione; quindi, si è andati a modificare le impostazioni degli *shader* per definire come il materiale (inteso come *shader + texture*) risponderà alle fonti luminose.

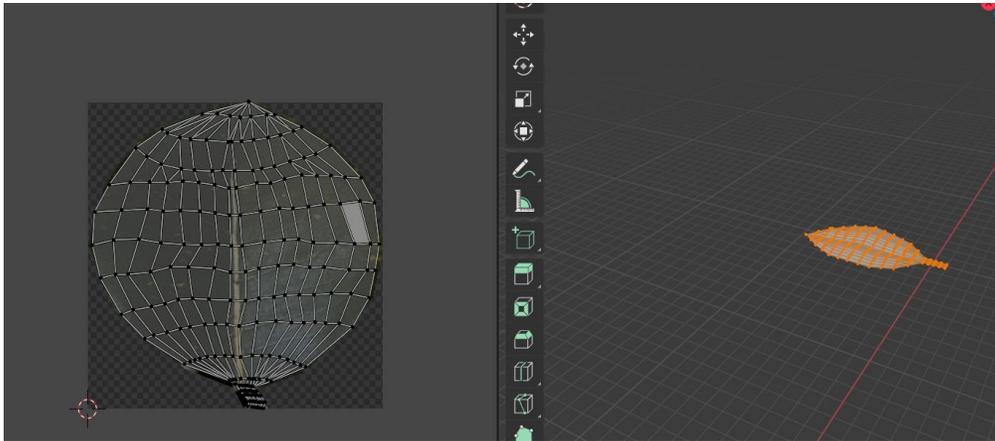
---

<sup>40</sup> Definizione da Wikipedia:

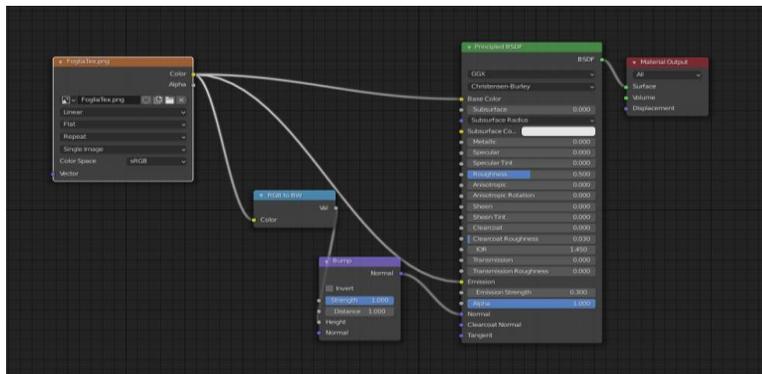
*Adobe Illustrator è un software per l'elaborazione di immagini e per la grafica vettoriale prodotto da Adobe a partire dal 1987. Il programma è utilizzato da milioni di designer e artisti di tutto il mondo per progettare qualsiasi tipo di grafica destinata all'editoria, al web e per dispositivi mobili.*

Link: [https://it.wikipedia.org/wiki/Adobe\\_Illustrator](https://it.wikipedia.org/wiki/Adobe_Illustrator)

Adobe 2020



**Figura 1: UV Mapping**



**Figure: Impostazioni del materiale (il nodo *bump* si è utilizzato per dare una simulazione di tridimensionalità all'immagine 2D) e il risultato**

Infine, si è applicato un emettitore di particelle con oggetto “foglia” attorno ad una curva a semicerchio per formare la corona e la si è aggiunta al modello di Dante precedentemente creato in Fuse.

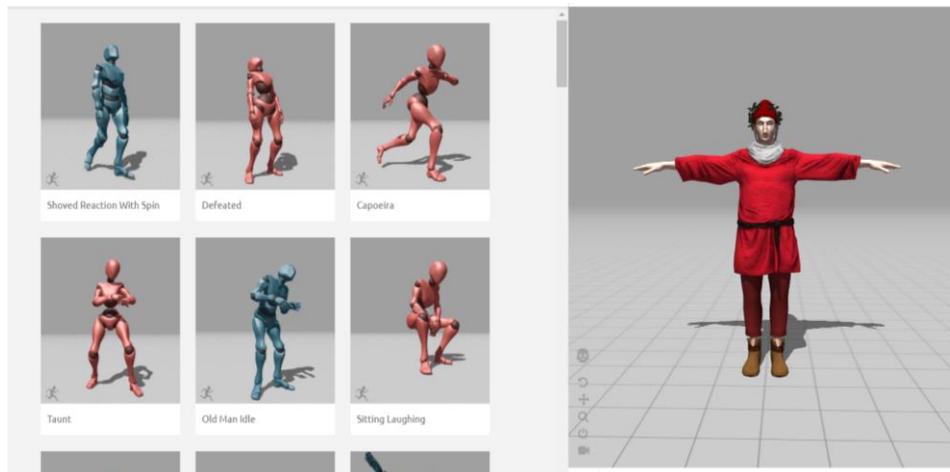


Con il modello completo in tal modo costituito si è passati al software di Mixamo per procedere al *rigging* ed all'animazione della *mesh*.

## Mixamo

*“Mixamo is a 3D computer graphics technology company. Based in San Francisco, the company develops and sells web-based services for 3D character animation. Mixamo’s technologies use machine learning methods to automate the steps of the character animation process, including 3D modeling to rigging and 3D animation.”<sup>41</sup>*

Attraverso questo strumento online, acquistato dal sistema Adobe nel 2015, è possibile caricare un modello umano nella cosiddetta *T pose*, la convenzionale posizione del modello a “braccia aperte” utile per migliorare la precisione del *rigging*. Una volta effettuato l'*upload* del personaggio ed impostati alcuni valori per posizionare correttamente l'ossatura, il programma procede automaticamente a gestire il resto; terminati i calcoli si ottiene un modello *riggato* a cui è possibile far compiere animazioni, alcune di queste vengono messe a disposizione da Mixamo e sono gratuitamente scaricabili in formato “.FBX”.



<sup>41</sup> Definizione da Wikipedia: <https://en.wikipedia.org/wiki/Mixamo>

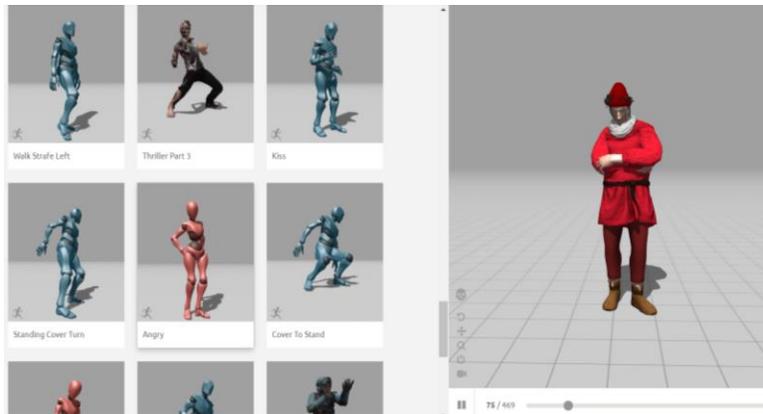


Figure: modello di Dante caricato in "T pose e con animazione applicata

Il tutto è stato quindi importato nel motore grafico Unreal Engine 4 per poter essere utilizzato in *real time*.

Un procedimento analogo a quello di Dante è stato utilizzato anche per il modello di Virgilio:

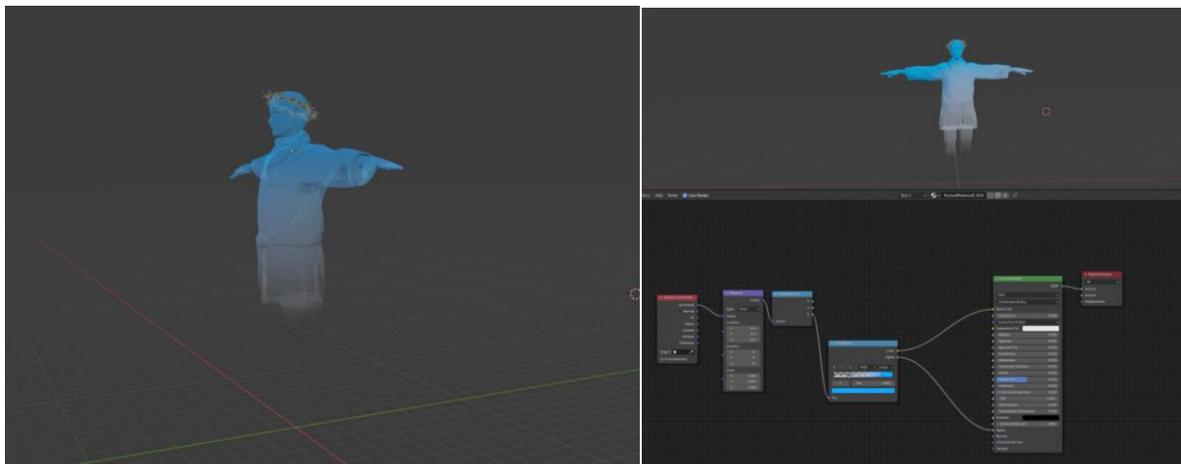


Figure: modello di Virgilio in Blender e relativo *shader node* del materiale; nello specifico si sono tolte tutte le *texture* che si erano esportate da Fuse e si è applicato al modello complessivo, in base alle coordinate geometriche ottenute dalla mappatura del modello, un *color ramp* verticale (sull'asse z) tra il trasparente e l'azzurro proprio per dare un effetto di evanescenza al personaggio che è presente nel poema in forma spiritica; successivamente si è proceduto al *baking* di due *texture*, una per il colore ed una per il canale alfa della trasparenza.

## Unreal Engine 4

Motore grafico sviluppato dalla Epic Games per il rendering in *real-time*:

*Unreal Engine is a complete suite of development tools for anyone working with real-time technology. It gives creators across industries the freedom and control to deliver cutting-edge entertainment, compelling visualizations, and immersive virtual worlds.*<sup>42</sup>

La versione 4 è annunciata già nel 2005. Questa versione aveva lo scopo di essere utilizzabile in parte dalle piattaforme di settima generazione, ma maggiormente da quelle di ottava. [...] A partire dal 2 marzo 2015, l'Unreal Engine 4 è disponibile gratuitamente a tutti, pagando comunque una royalty del 5% sul reddito lordo, dopo i primi 3000\$ per prodotto, per trimestre se il prodotto non verrà pubblicato sullo store proprietario Epic.<sup>43</sup>

Questo software, nella versione 4.27, è stato utilizzato per dar vita al vero e proprio ambiente virtuale della *Divina Commedia*. Quivi è stato creato un progetto ed impostato per poter essere letto dall'Oculus, si sono importati tutti gli elementi precedentemente creati e si sono scaricati alcuni *asset* di modelli per arricchire il mondo virtuale<sup>44</sup>; infine, si è andati a gestire il comportamento del mondo e

---

<sup>42</sup> Epic Games 2021

<sup>43</sup> Definizione da Wikipedia: [https://it.wikipedia.org/wiki/Unreal\\_Engine](https://it.wikipedia.org/wiki/Unreal_Engine)

Epic Games 2021

<sup>44</sup> Dal *Marketplace* di Epic Games è possibile acquistare o, a volte, scaricare gratuitamente una sconfinata varietà di materiale utilizzabile: modelli, mappe, materiali, texture, suoni, funzioni, animazioni e altri strumenti utili per consentire di personalizzare il proprio progetto a seconda della propria volontà.

Link al *Marketplace*: <https://www.unrealengine.com/marketplace/en-US/store>

In particolar modo gli *asset* scaricati per questo progetto dal negozio della Epic sono i seguenti:

1) *Ultimate SFX Bundle*: Un voluminoso pacchetto di suoni registrati e rimasterizzati da sfruttare nei propri progetti, quivi si sono utilizzati i suoni per i passi del modello di Dante sul terreno ed altri suoni ambientali, come il vento, il fruscio delle foglie ed i suoni di animali; tutti elementi utili ad aumentare il potenziale immersivo dell'esperienza e a dare maggior contesto alla Selva Oscura.

Link: <https://www.unrealengine.com/marketplace/en-US/product/ultimate-sfx-bundle>

2) *Realistic Forest Pack*: Un altro *bundle* contenente mesh e texture per la creazione di ambienti boschivi realistici e che si è utilizzato per la texturizzazione dei vari terreni, per la creazione della foresta e la gestione del *foliage* in generale.

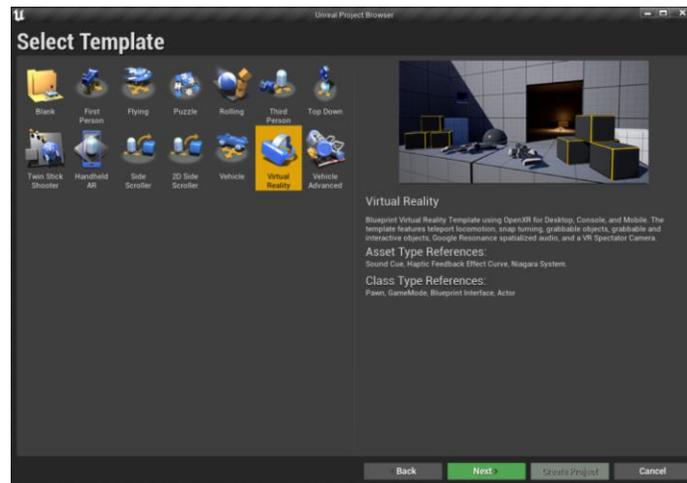
Link: <https://www.unrealengine.com/marketplace/en-US/product/realistic-forest-pack>

di tutti gli elementi costituenti attraverso la programmazione a *blueprint*, come verrà spiegato successivamente.

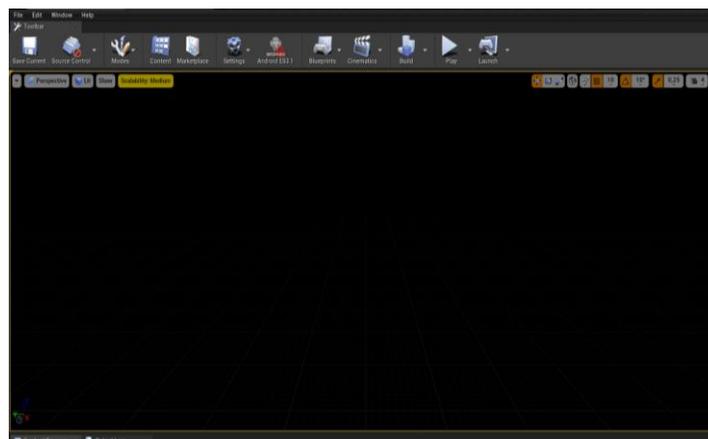
## Esempio di creazione di una mappa in UE4

Si porta ora un esempio di creazione di un livello, tale procedura si è poi ripetuta per la creazione di ciascuna mappa.

Innanzitutto, si crea un nuovo progetto e si seleziona un *template* tra quelli disponibili, ciascuno dei quali dispone già di oggetti e funzioni utili, messi a disposizione da UE4 stesso, per facilitare e velocizzare la programmazione.



Quindi si crea il progetto, lo si nomina e si sceglie se si desidera avere dello *starter content* di *mesh*, materiali e *texture*. Se si decide di non avere contenuto iniziale, ci si trova davanti a una schermata nera.



Innanzitutto, bisogna creare il terreno, andando su *Modes* -> *Landscape*; qui vengono presentate varie opzioni per definire le dimensioni della mappa, si conferma selezionando *create*. In questo modo viene generato un quadrilatero delle dimensioni desiderate senza nessun materiale applicato. Si può ora andare a modificare la conformazione del terreno inserendo piani ed altorilievi a piacimento: UE4 ha numerosi strumenti da usare nell'*editor* ambientale che si possono trovare nella categoria *sculpt* di *Landscape*.

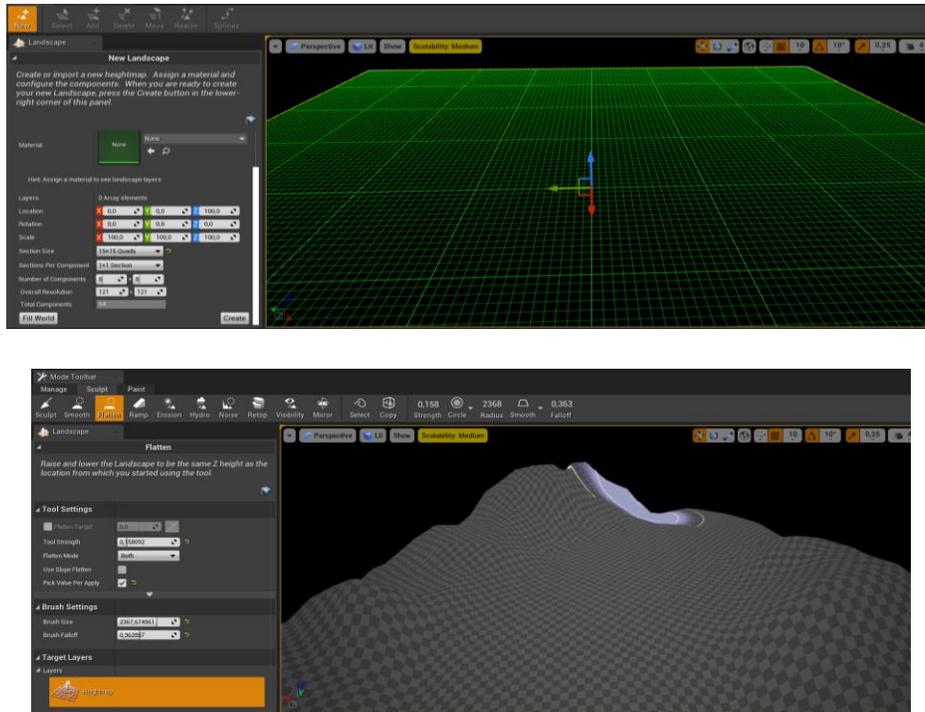
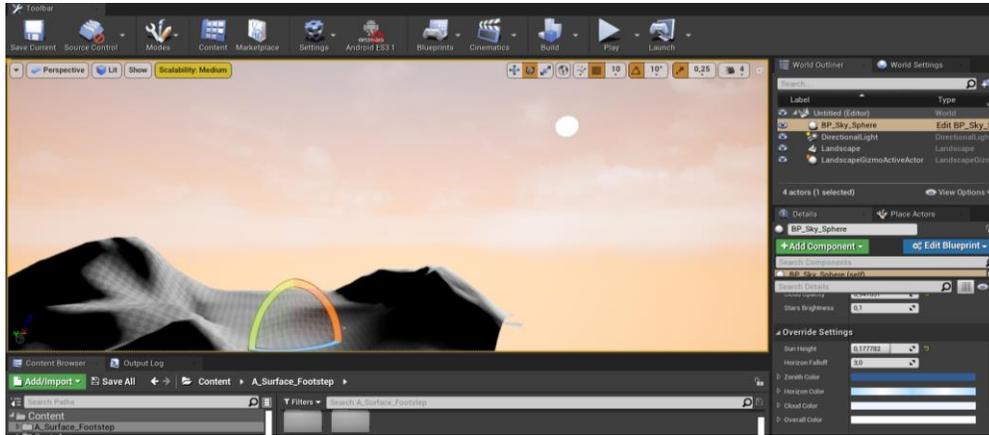
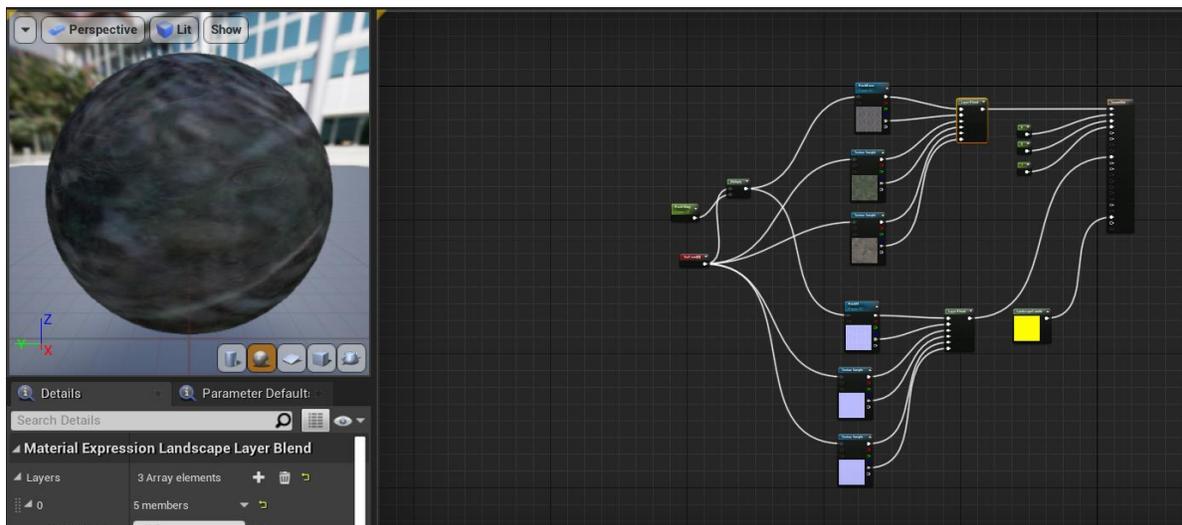


Figure: creazione della mappa e modifica attraverso gli strumenti di UE4

Ora, prima di andare a creare il materiale, bisogna inserire delle luci tornando su *Modes* -> *select*; ci sono varie conformazioni valide, nel progetto in questione si è associato uno *Sky Sphere* ad una luce direzionale, in questo modo è possibile andare ad impostare varie opzioni: la posizione del sole (cambia in base alla rotazione della luce direzionale), il momento della giornata tra alba, giorno, tramonto e sera, ed altre caratteristiche quali le nuvole, vari colori della luce, la presenza della di stelle e la loro densità.



Infine, si va ad aggiungere il materiale che si intende utilizzare per la mappa. Per farlo si deve partire da delle *texture*, generalmente almeno una per ciascun tipo di terreno che si andrà ad inserire; ad esempio, qui si vogliono creare tre *layer*, rispettivamente chiamati: “grass”, “rock” e “gravel”.



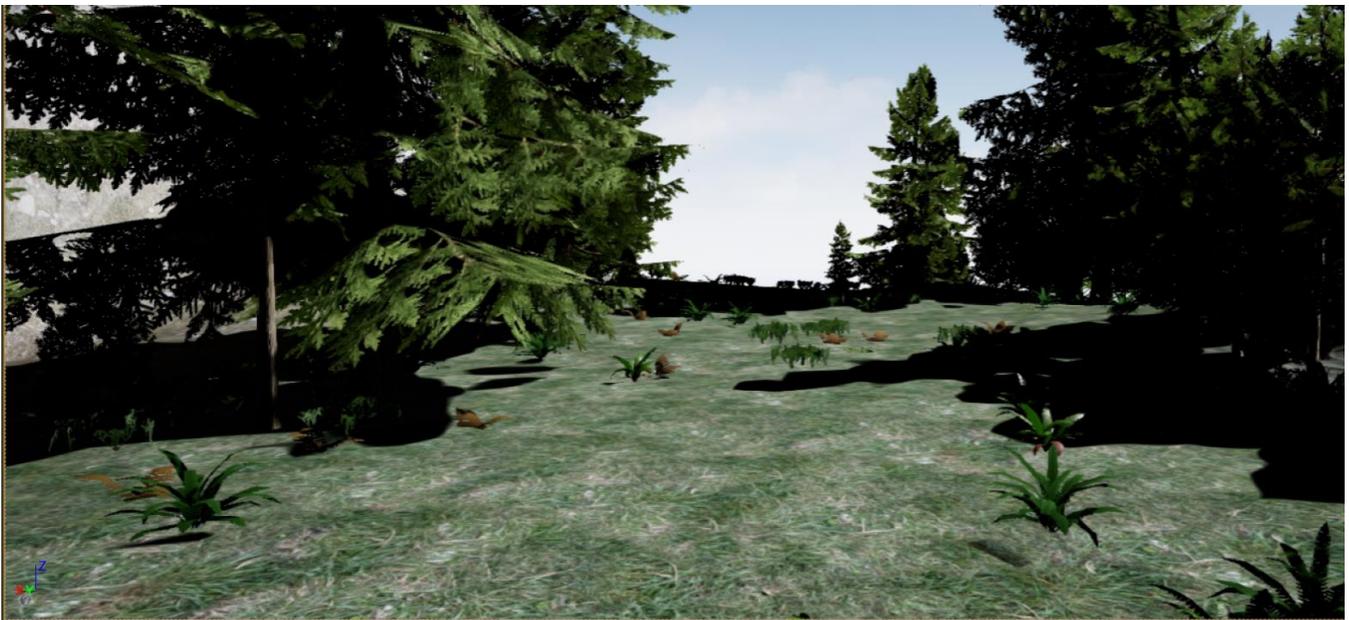
**Figura: Blueprint del materiale, per ciascuno dei tre *layer* è presente una texture di colore ed una per le normali (6 texture in totale), queste ultime servono a dare simulazione di 3D alla texture bidimensionale andando a definire come la luce si comporta quando colpisce il materiale in questione, modificando la traiettoria del raggio luminoso si simulano concavità e convessità senza dover effettivamente aumentare il numero di poligoni.**

Creato il materiale lo si può associare alla mappa e, impostati i cosiddetti *layerInfo* che definiscono come ciascun tipo di terreno si disporrà sulla mappa stessa, si può iniziare a “dipingere” il terreno attraverso uno strumento *brush* con le *texture* selezionate e lo si fa andando alla sottosezione *Paint* dell’*editor Landscape*.



**Figura: Sezione della mappa con materiale applicato e ciascuno dei *layers* inseriti: *grass, gravel, rock***

Infine, si può inserire il cosiddetto *foliage* composto da ulteriori elementi ambientali (alberi, piante o rocce) che vadano a completare la mappa, questi si possono inserire manualmente o attraverso algoritmi di applicazione casuali controllabili attraverso valori modificabili nell'apposito *editor*.



**Figura: Mappa con *foliage* applicato**

## Programmazione in UE4

UE4 utilizza due metodi di programmazione: un metodo “classico” in linguaggio C++ che permette la gestione del progetto mediante compilazione da un *editor* di testo; ed un metodo di *scripting* visivo che sfrutta l’utilizzo di *blueprint*. Per lo sviluppo di questo progetto è stato utilizzato questo secondo approccio.

[...] un’interfaccia che ci consente di programmare dei comportamenti senza agire direttamente sul codice, ma impiegando esclusivamente un sistema di nodi. Tali nodi si presentano come dei pannelli di diverso colore, che potremo connettere tra di loro tramite delle frecce, che ci permetteranno di stabilire dei comportamenti per il nostro gioco.<sup>45</sup>

```
60 template<typename, typename> class TOctree2;
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
```

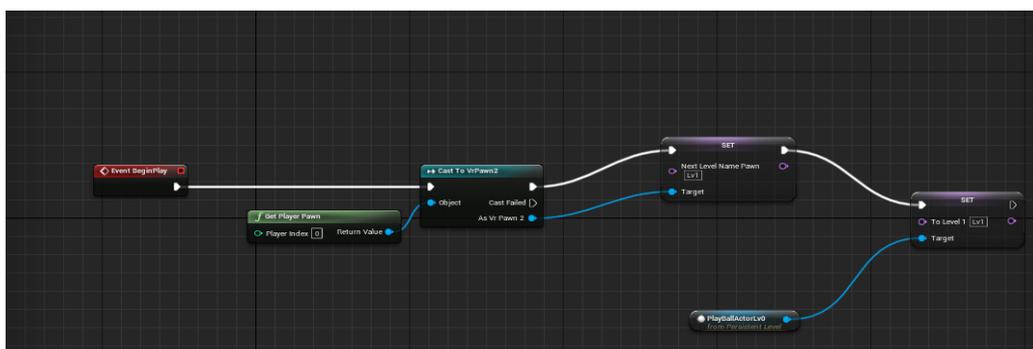


Figure: sopra la programmazione in C++, sotto la programmazione a *blueprint*

<sup>45</sup> Definizione di *blueprint* dal sito di UE4, al link: <https://www.html.it/pag/369129/blueprint-unreal-engine/>

Detto ciò, risulta comodo per semplificare la descrizione tecnica del progetto suddividere i suoi componenti in due categorie: gli “elementi unici”, definiti una volta per tutte, e che si replicano senza modifiche ciascuna volta che vengono *invocati* e degli “elementi variabili” dei quali invece se ne sono create varie istanze per adattarsi alle necessità di ciascuna mappa specifica.

## Descrizione elementi unici

### **La *Game Mode* e il *Virtual Pawn***

Il primo frangente da strutturare in un’applicazione in UE4 è la *Game Mode*, cioè l’insieme di regole macroscopiche atte a definire il “tipo di gioco” che si intende creare, per questo progetto si è partiti dalla classe di *default* di UE4 per la VR.

Even the most open-ended game has an underpinning of rules, and these rules make up a Game Mode. On the most basic level, these rules include:

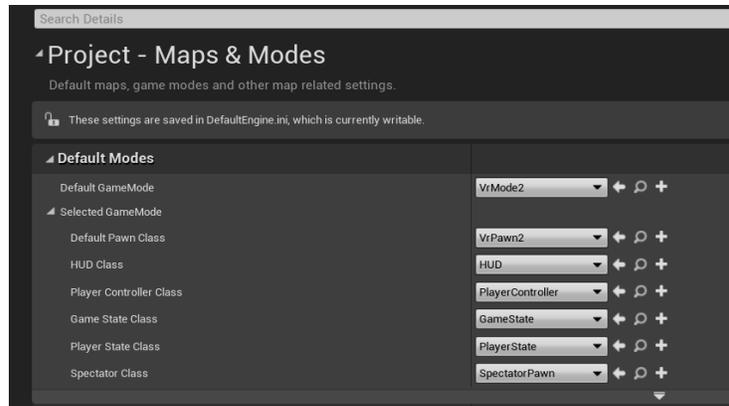
- The number of players and spectators present, as well as the maximum number of players and spectators allowed.
- How players enter the game, which can include rules for selecting spawn locations and other spawn/respawn behavior.
- Whether or not the game can be paused, and how pausing the game is handled.
- Transitions between levels, including whether or not the game should start in cinematic mode.<sup>46</sup>

[...] While certain fundamentals, like the number of players required to play, or the method by which those players join the game, are common to many types of games, limitless rule variations are possible depending on the specific game you are developing. Regardless of what those rules are, Game Modes are designed to define and implement them.

---

<sup>46</sup> Definizione di *Game Mode* dal sito di UE4, al link:

<https://docs.unrealengine.com/4.27/en-US/InteractiveExperiences/Framework/GameMode/>



Tra le varie opzioni che questa schermata permette di gestire, di fondamentale importanza è la *Default Pawn Class*. Un *pawn* è un elemento controllabile da un giocatore o un'IA ed andrà a definire l'*avatar* che l'utente potrà controllare, in questo caso una telecamera in prima persona con delle mani attaccate; tale elemento gli consentirà di essere spettatore nelle scene, di spostarsi nella mappa e di interagire con il mondo virtuale.



**Figura: Il *pawn* utilizzato per quest'applicazione con vista del *viewport* e del grafico in *blueprint***

Il *VrPawn* (nome dato per questo progetto) è l'elemento centrale dell'applicazione, attorno al quale si sviluppano numerosi elementi che si andranno ora a descrivere nello specifico. Dal nodo evento *Begin To Play*, che viene eseguito appena l'elemento *VrPawn* viene creato all'interno del gioco, si diramano numerosi altri elementi attraverso il nodo *sequence*, in particolar modo:

- 1) *Then 0*: si collega alle funzioni addette alla gestione dell'audio relativo a versi della *Divina Commedia* che andranno ad essere letti dalla voce narrante.

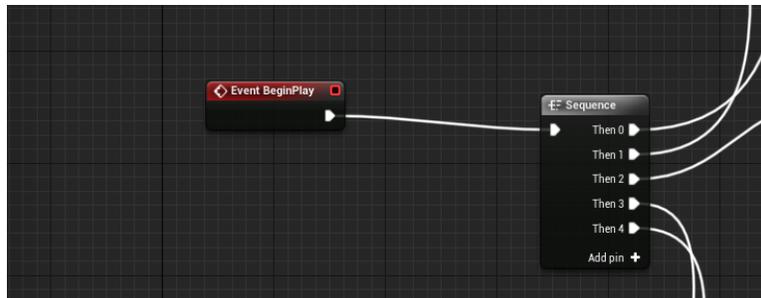


Figura: Nodo *sequence* che agisce dopo il *BeginPlay*

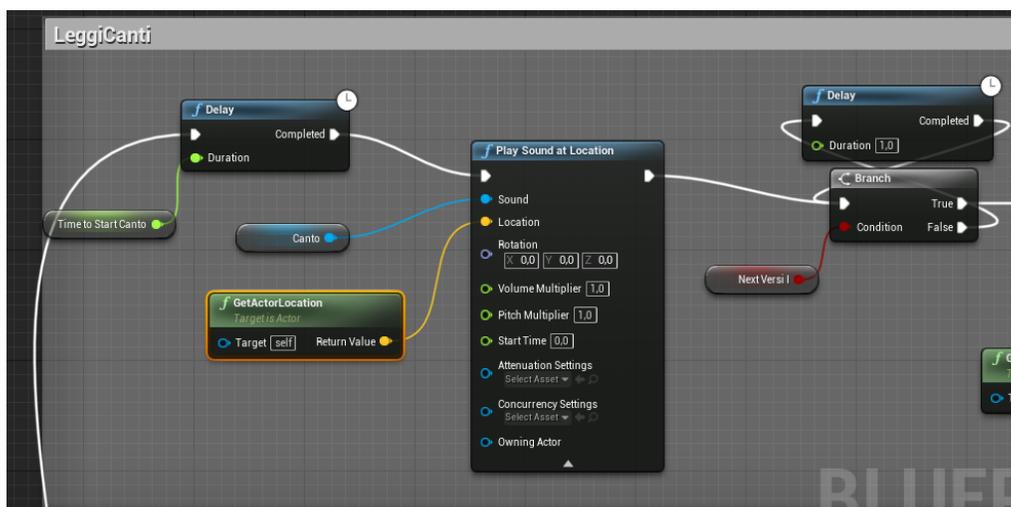
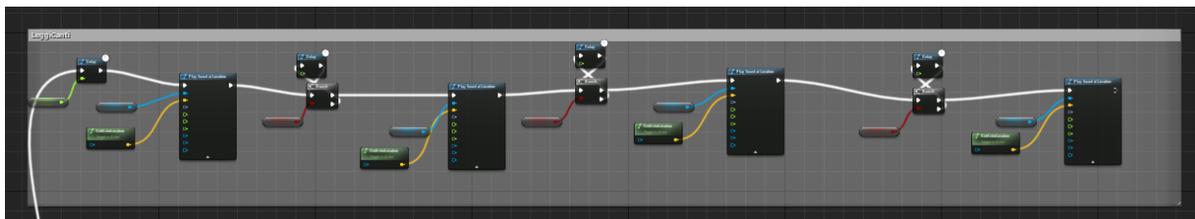


Figure: gestione dei file audio, la funzione *Play Sound At Location* fa partire l'audio connesso alla variabile *Canto* (qui definita, ma inizializzata in ciascun *Level Blueprint*, come verrà spiegato in seguito) nella posizione del *VrPawn*. Siccome in alcuni livelli si è deciso di separare in sottosezioni i versi in modo da poter aggiungere maggior interattività (vedi il capitolo riguardante le sfere interattive), si è inserita una condizione booleana come guardia per la lettura delle terzine successive: quando, durante il gioco, la "sfera di progresso" viene colpita (*On Hit Event*) la condizione *Next Versi I* passa da *false* a *true* e permette la riproduzione audio dei versi successivi.

2) *Then 1*: da questo nodo sono gestiti gli altri suoni ambientali estratti dall' *Ultimate SFX Bundle* tra cui il fruscio del vento tra le gli alberi e i suoni della fauna che abita la Selva Oscura.

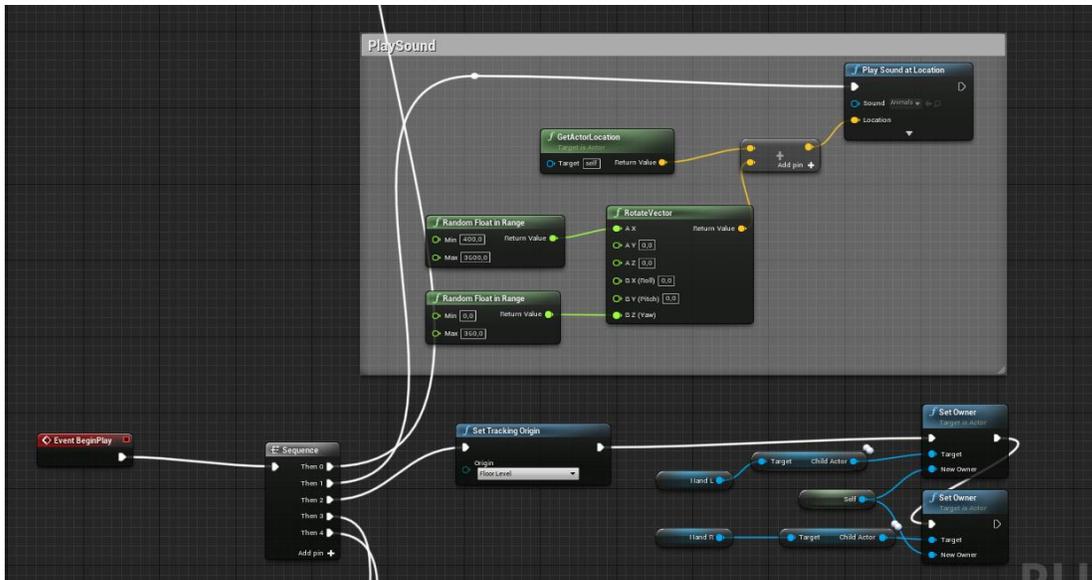
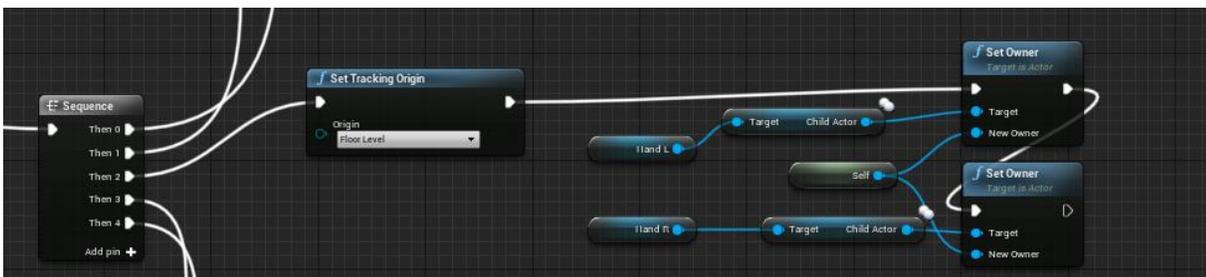
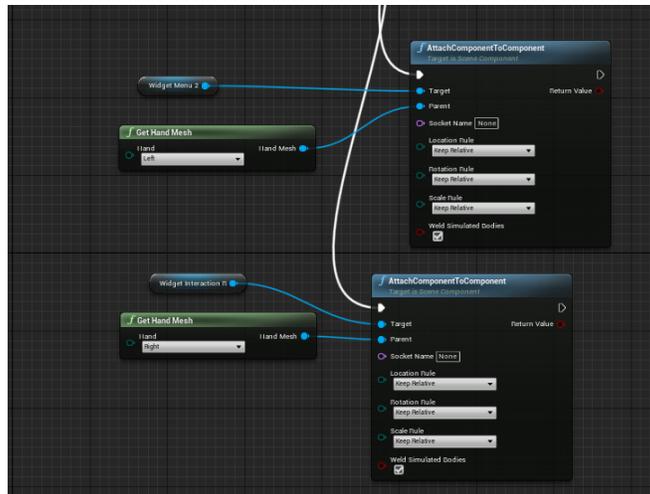


Figura: in particolar modo si è creato un insieme di suoni che vengono emessi in *loop*, ogni volta da una posizione diversa estratta casualmente dalla sezione piana parallela al terreno di una figura toroidale (quindi con un raggio di distanza interno ed uno esterno) che ha il giocatore come centro.

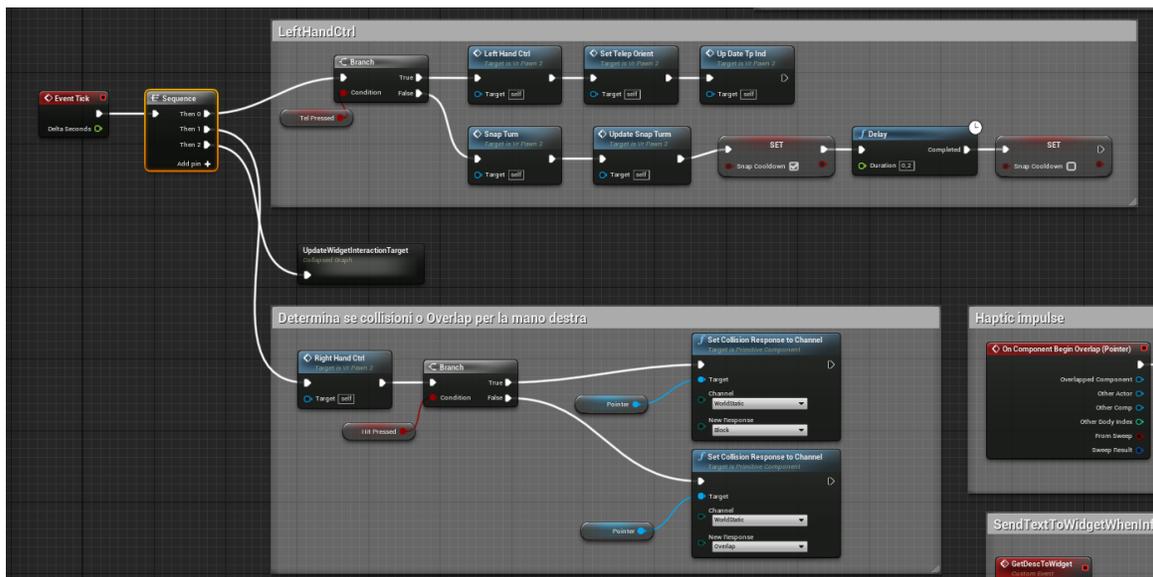
3) *Then 2*: qui vengono associate le *mesh* delle mani virtuali al *pawn* attraverso le apposite funzioni predisposte da UE4.



4) *Then 3* e *Then 4*: si istaurano le dipendenze, rispettivamente con la mano sinistra ed con la mano destra, delle componenti *Widget Menu* (menù interattivo con varie opzioni e funzioni che, come verrà approfondito nell'apposita sezione, è possibile far comparire e scomparire sul proprio braccio virtuale sinistro) e *Widget Interaction* (componente per interagire con l'elemento appena introdotto).



Un altro nodo fondamentale da trattare insieme a *Begin Play*, è il nodo *Event Tick*; mentre il primo viene eseguito una sola volta quando *l'avatar* viene generato, quest altro nodo viene invece eseguito ad ogni *frame* e si utilizza generalmente per l'aggiornamento di variabili. Anche da qui parte una *sequence* di altri nodi:



- 1) *Then 0*: gestisce le funzioni della mano sinistra in base ai pulsanti del *controller* che vengono premuti, cioè l'indicatore di teletrasporto se viene premuto il tasto *left trigger* o la rotazione della telecamera attorno all'asse *z* se viene inclinata la leva analogia. Ad ogni *frame*, se uno di questi pulsanti risulta premuto, si eseguono le funzioni corrispondenti per segnalare dov'è possibile spostarsi o per ruotare la camera.

- 2) *Then 1*: questi nodi collassati (per motivi di spazio nel *blueprint*) permettono al puntatore della mano destra, grazie alla dipendenza dell'elemento *Widget interaction* di intercettare il *Menu Widget* del braccio sinistro quando questo è attivo per poter interagire con i suoi elementi.

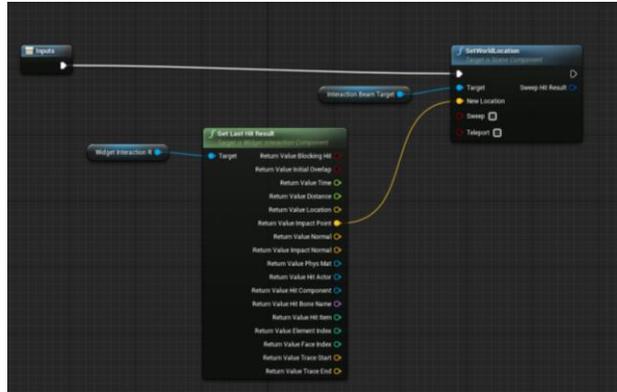


Figura: Nodi collassati di *Update Widget Interaction Target* espansi.

- 3) *Then 2*: gestisce i canali di collisione del puntatore laser della mano destra; in particolar modo, ad ogni *frame*, se il *right trigger* risulta premuto cambia il *collision channel* del suddetto puntatore in modo che, se viene colpito un elemento interattivo, si possa generare un *Hit Event* e far eseguire la funzione annessa (cambiare livello, mandare testi al *widget*, etc...)

Altre funzioni specifiche utilizzate ed eventi secondari del *VrPawn*:

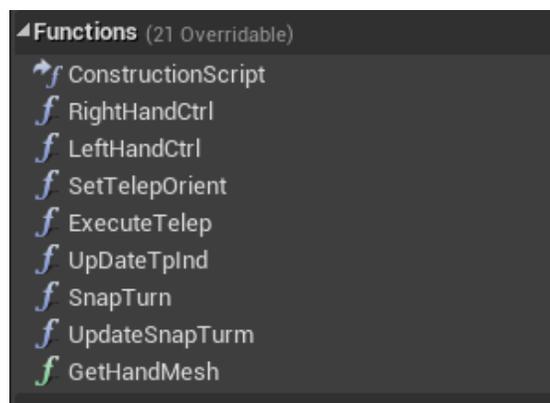


Figura: Queste funzioni sono necessarie per gestire le dinamiche del puntatore laser della mano destra (*RighthHandCtrl*), del teletrasporto e del relativo indicatore per la mano sinistra (*LeftHandCtrl*, *SetTelepOrient*, *ExecuteTelep*, *UpdateTpInd*) e della rotazione della camera attorno al proprio asse z all'inclinazione dell'analogico del controller sinistro (*SnapTurn*, *UpdateSnapTurn*)

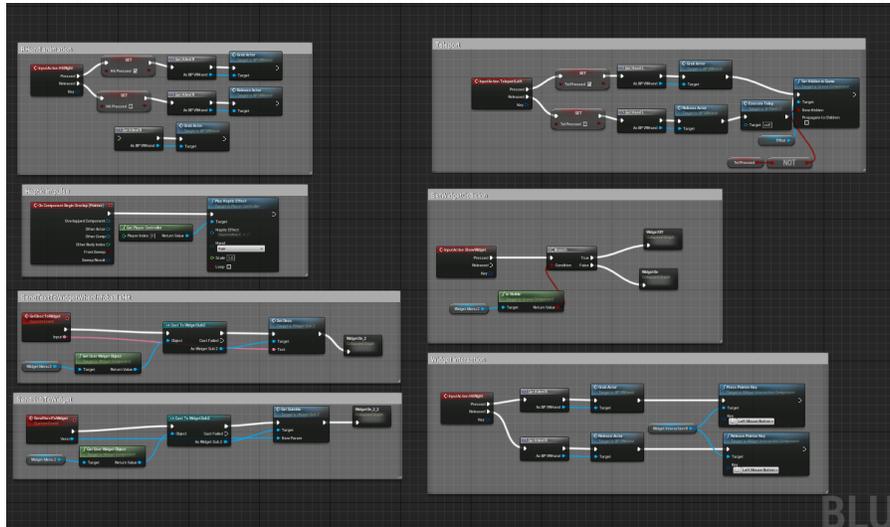


Figura: altri eventi utilizzati definiti riguardano l'interazione con il *Menu Widget*, l'invio di testi a questo componente, impulsi aptici che si attivano quando il puntatore della mano destra passa sopra un elemento interattivo (per aumentare l'usabilità dell'applicazione) e le animazioni di chiusura ed apertura delle due mani virtuali.

## Menu Widget

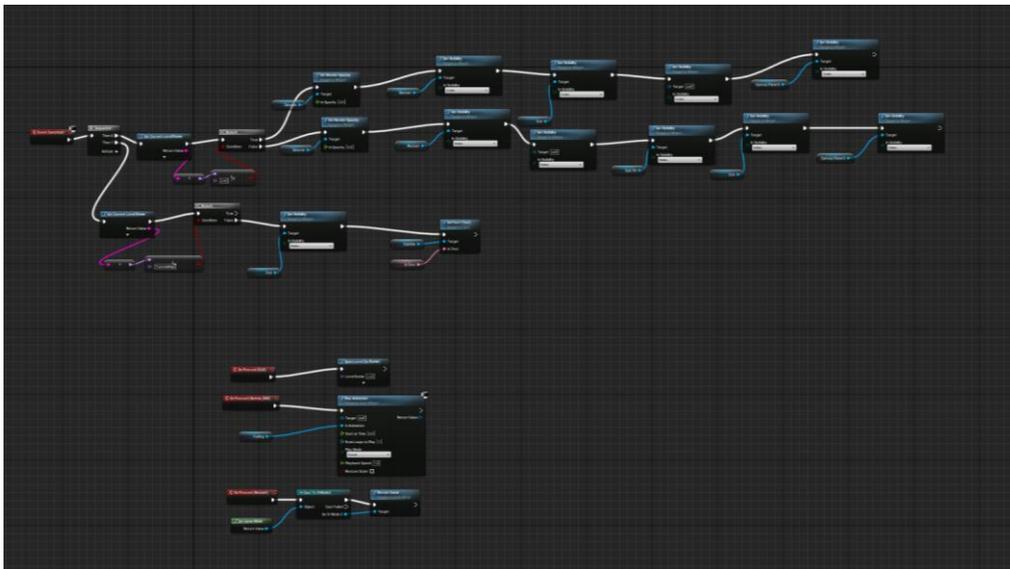


Figura: successivamente, non presente in figura, è stato aggiunto anche un pulsante per mettere pausa e far riprendere l'esperienza

Con *widget* si intende una qualsiasi interfaccia utente di un programma, utile a facilitarne l'interazione; l'idea del *Menu Widget* in questione, un elemento bisimensionale che è possibile render

visibile ed invisibile nel proprio braccio sinistro, nasce dalla necessità di mettere a disposizione uno strumento per facilitare la fruizione di contenuti testuali quali i versi dei canti o i vari commenti con nozioni storiche, filologiche e letterarie; un'ulteriore difficoltà a cui si è dovuto pensare nel progettare tale elemento è stata il doverne permettere l'utilizzo in un ambiente virtuale immersivo stereoscopico (contesto in cui diventano poco pratici i menù statici) e che, allo stesso tempo, non infastidisca l'utente, egemonizzando la schermata, ed impedendogli di osservare le scene narrate nel corso del canto. Qui, oltre ai due *box* testuali, sono altresì rese disponibili alcune funzioni utili, quali il *Restart* che consente di ricaricare da capo la scena attuale o *Esci* che permette di trasportarsi nella mappa iniziale da cui è possibile terminare l'applicazione.

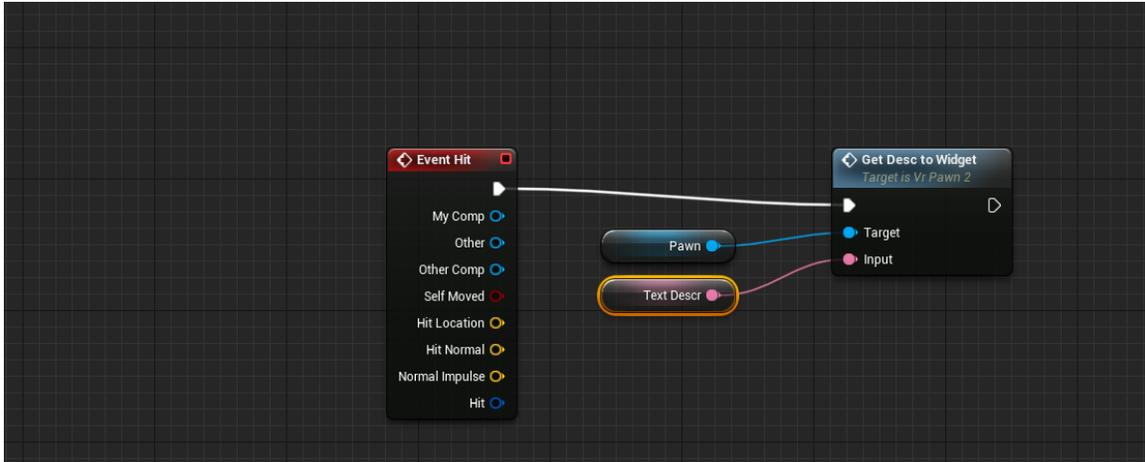
Come si andrà a spiegare questo elemento è strettamente collegato al *VrPawn* in quanto, quando si è *in game*, viene attaccato al braccio virtuale sinistra del giocatore. Il suo funzionamento si basa su poche funzioni ed eventi.



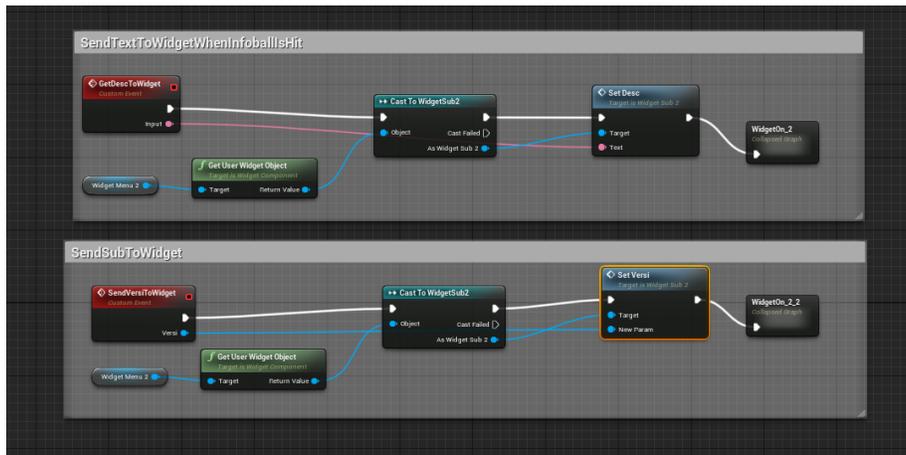
**Figura:** Qui si associano ai pulsanti *Restart*, *Pausa* ed *Esci* le relative funzioni *Restart Game* e *Open level Lv0* Che vengono azionate all'evento *OnPressed* del tasto corrispondente; inoltre, si gestiscono gli elementi relativi alla visibilità dei vari componenti del *widget* in base alla mappa in cui ci si trova (ad esempio, nella mappa chiamata "Lv0" il menù non sarà visibile).

Altre due funzioni, come anticipato, lavorano in dipendenza del *VrPawn*, cioè *SetVersi* e *SetDesc*; utili, rispettivamente, a visualizzare i testi dei versi (nel *box* a sinistra) e degli altri contenuti informativi (*box* a destra) attraverso l'associazione delle informazioni derivanti dal *VrPawn* nelle apposite variabili

associate ai *text box* in questione. Come verrà approfondito in seguito sono presenti delle sfere interattive all'interno dell'esperienza, tra esse ci sono le, così nominate, "sfere dei versi" e le "sfere informative", entrambe denotate da un evento *OnHit*:



Quando il giocatore colpisce uno di questi elementi con il suo puntatore e preme il pulsante *Trigger Destro*, genera un evento attraverso cui viene inviato il testo contenuto in una apposita variabile (inizializzata nel momento del caricamento di una mappa) della sfera al *pawn* e questo, a sua volta, lo spedisce al *widget* che procederà alla visualizzazione.



**Figura: Eventi del *VrPawn* adibiti a spedire testi al *widget***

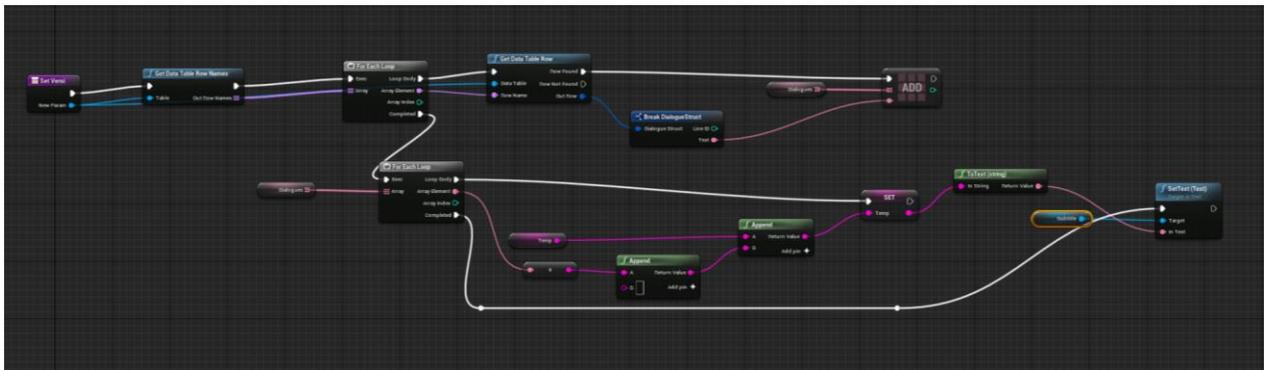
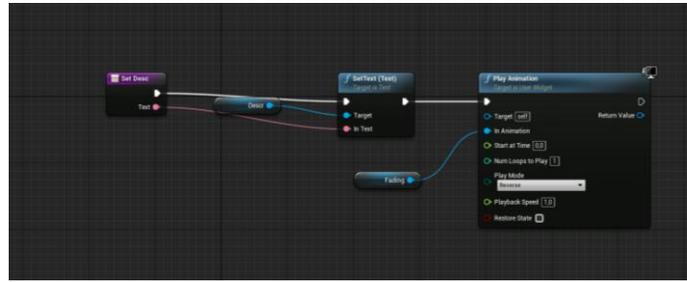
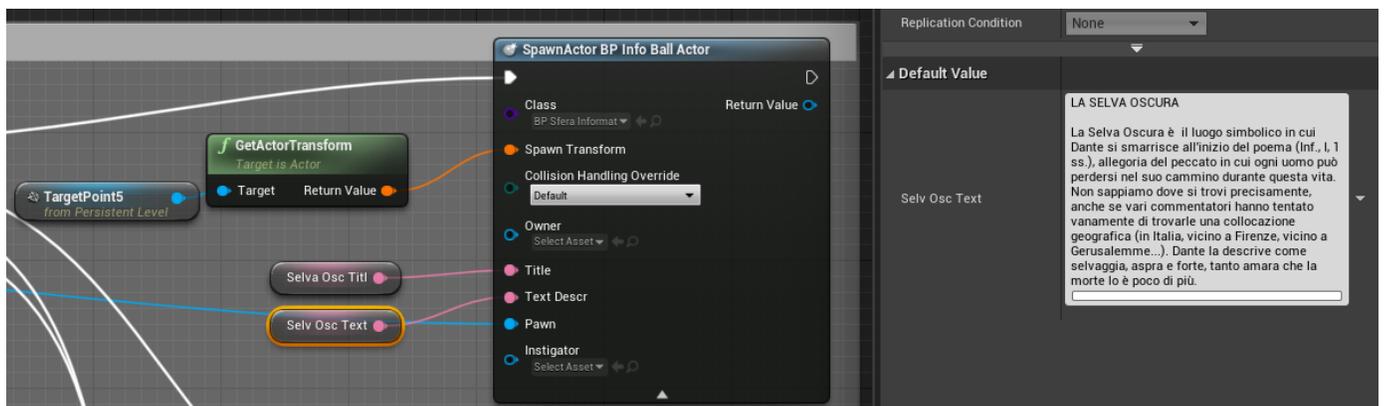


Figura: Funzioni che gestiscono la visualizzazione dei testi nel *Menù Widget*, la funzione *SetDesc* (sopra), gestisce la visualizzazione dei testi provenienti dalle “sfere informative”; mentre *SetVersi* (sotto), gestisce quella dei testi provenienti dalle “sfere dei versi”

Un approfondimento necessario da affrontare riguarda la funzione *SetVersi*; più complessa, come si può notare dall’immagine, rispetto a *SetDesc*, il motivo è semplice: mentre le descrizioni sono contenute testuali semplici, generalmente corti e senza particolari necessità tipografiche, i testi dei versi sono più lunghi e devono mantenere la struttura poetica originaria dei capoversi. Per le descrizioni si sono utilizzate semplici variabili di tipo *text*; mentre per i versi si è utilizzato un diverso tipo di variabile chiamata *Data Table*.



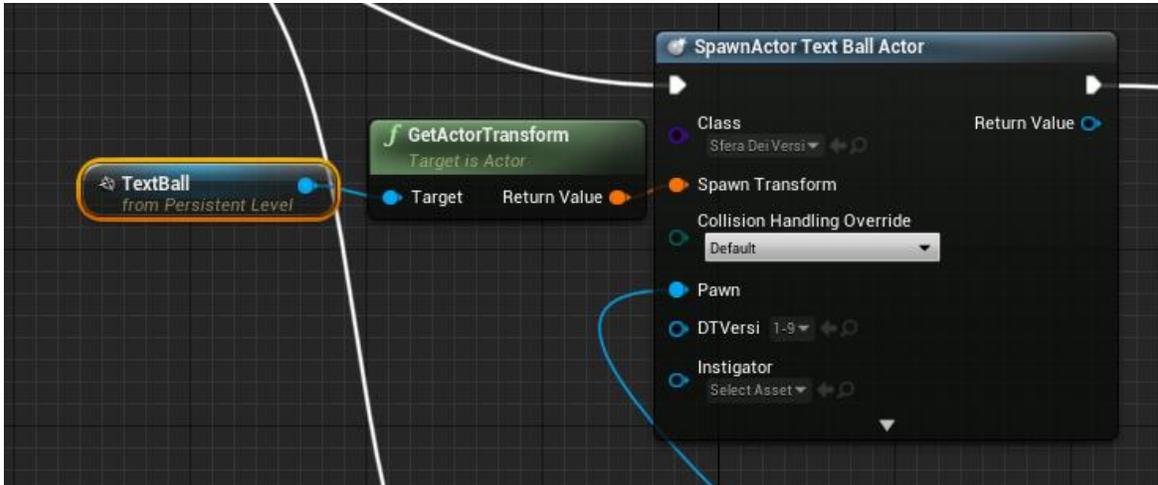


Figure: queste funzioni di *spawn* si occupano della comparsa degli *actors* nei vari livelli e verranno affrontate in seguito; si noti come la *SpawnActor* della “sfera informativa” (sopra) ha una variabile chiamata *TextDesc* a cui viene associata una variabile di testo definita nel livello stesso; al contrario la *SpawnActor* della “Sfera dei versi” (sotto) Prende in input una variabile chiamata *DTVersi (DataTable)*.

Una *Data Table* è una tabella utilizzata per contenere dati in righe e colonne, ed è possibile importare tali elementi in UE4 per varie necessità, in questo frangente si è utilizzata per gestire i versi della *Divina Commedia*. La *Data Table* è stata riempita con Google Sheet. Per quanto riguarda la tabulazione: si è lasciata la prima colonna senza nome e con una numerazione progressiva che definisce il numero di righe utilizzate (prassi necessaria per consentire l’importazione in UE4), poi si è nominata la seconda colonna “*Line\_ID*” che enumera i soli versi (senza lo spazio lasciato per il cambio di terzine) e la terza colonna “*Text*” con il contenuto testuale del verso corrispondente. Completata la tabella in tal modo la si è esportata in formato “.csv” (valori separati da virgole), questo formato è stato poi importato in UE4.

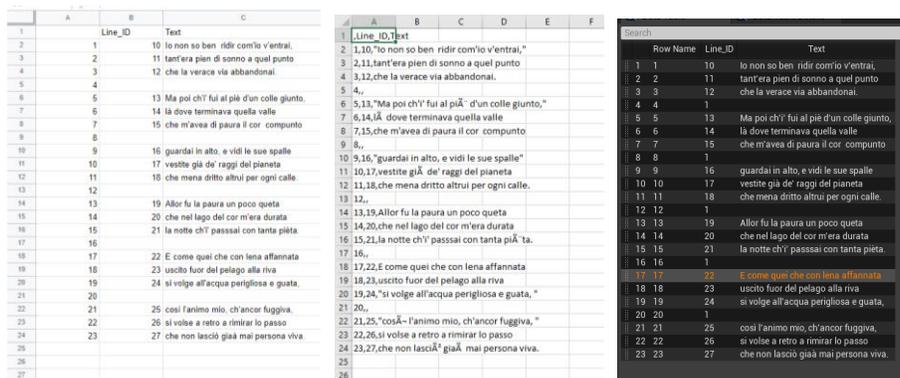


Figure: *Data Table* in Google Sheet (a sinistra), in formato “.csv2” (al centro) e quindi reimportata in UE4 (a destra)



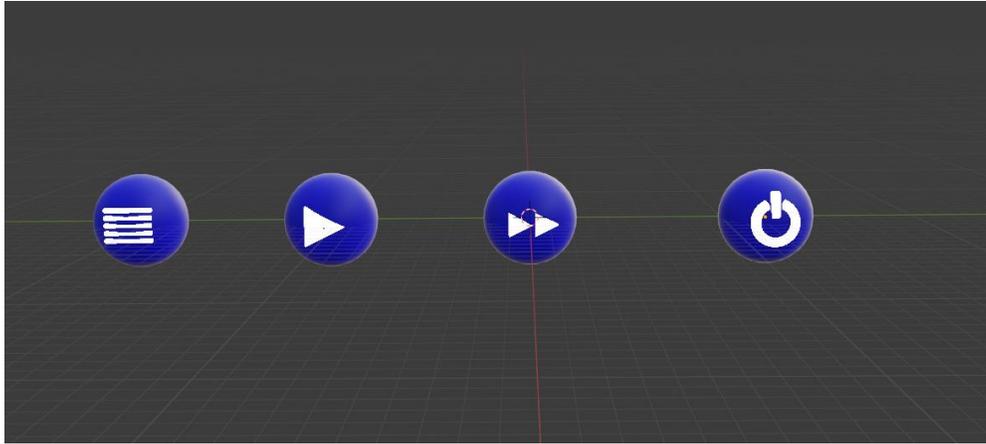
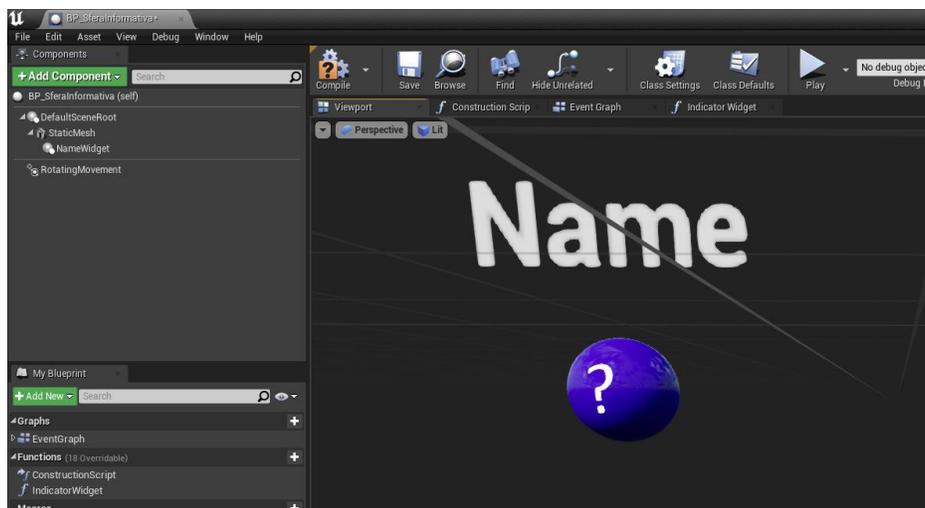


Figura: varie sfere in Blender

Ciascuna sfera si contraddistingue dalle altre da un simbolo in base alla funzione a loro associata, ma prima di analizzare le differenze, verranno analizzate le affinità tra questi oggetti virtuali.



Tutte le sfere sono state gestite come *Actor* per permettere loro di avere dei *blueprint* a cui associare funzioni, variabili ed eventi; un *Actor* ha a sua volta una *mesh* a cui è stata associata una rotazione, per puro effetto grafico, e un *widget* che le sta sopra e ne specifica il nome, coerente alla funzione che svolge; a questo componente è stata quindi associata una funzione, chiamata successivamente nell'*Event Tick*, che ad ogni frame indirizza il testo verso l'utente.

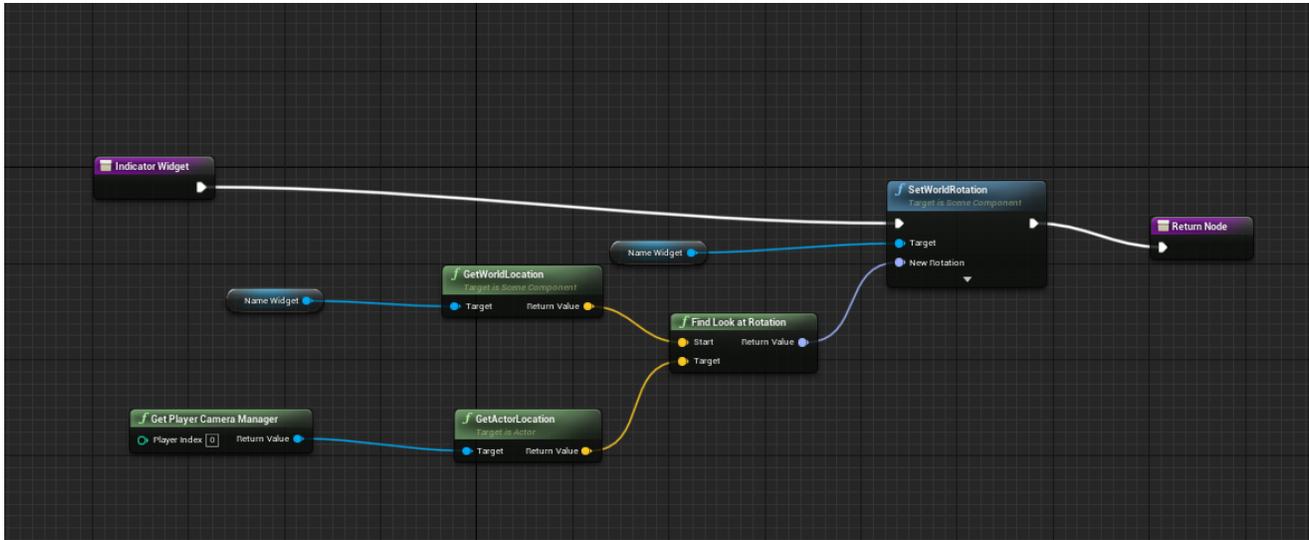


Figura: La funzione *Indicator Widget* viene chiamata dall'*Event Tick*, ad ogni frame si cerca la posizione del *VrPawn* e si fa ruotare il *widget* della sfera in quella direzione

Un aspetto importante da analizzare riguarda le impostazioni per le collisioni in quanto è proprio qui che viene gestita l'interazione tra il *pawn* e queste altre componenti mediante un elemento chiamato *Pointer* (cioè la sfera rossa che si può vedere al termine del puntatore laser che parte dalla mano destra).

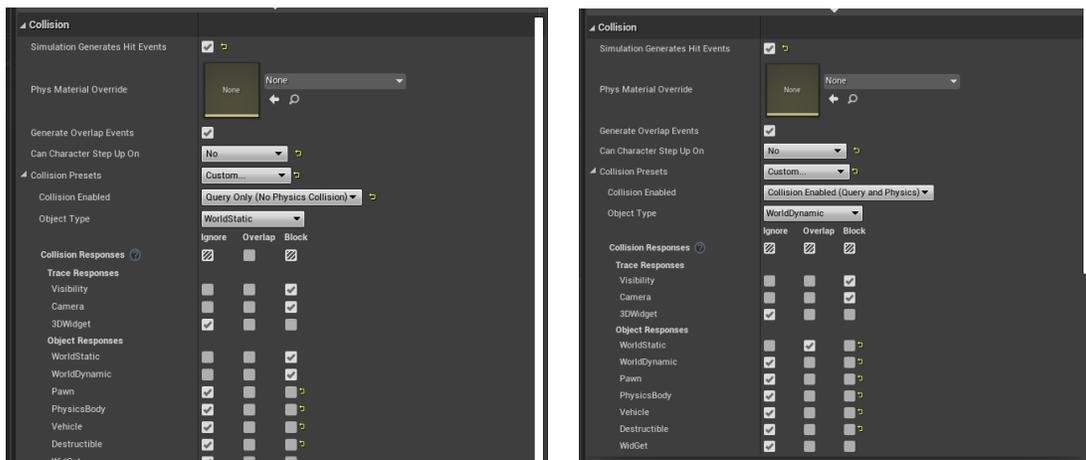
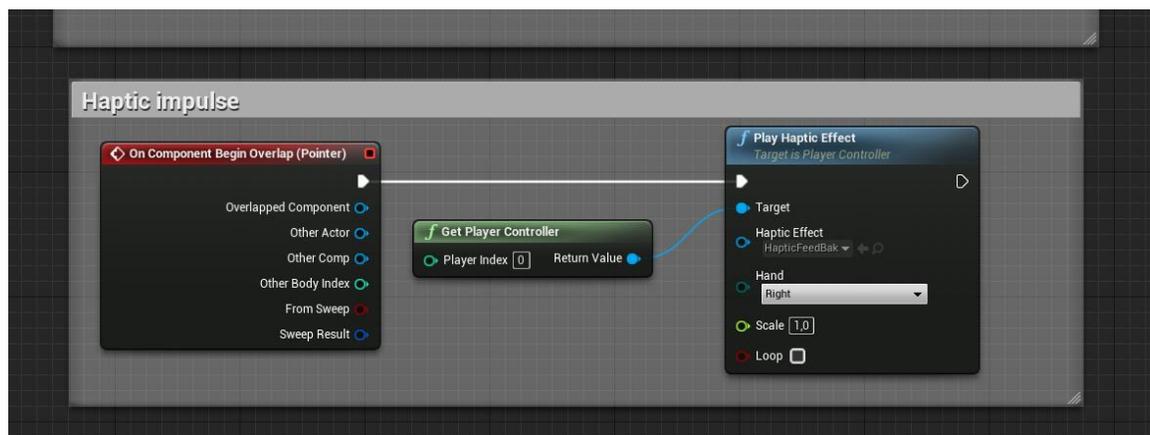


Figure: A sinistra il *Collision Set* delle sfere (uguale per tutte), a destra quello del componente *Pointer* del *VrPawn*

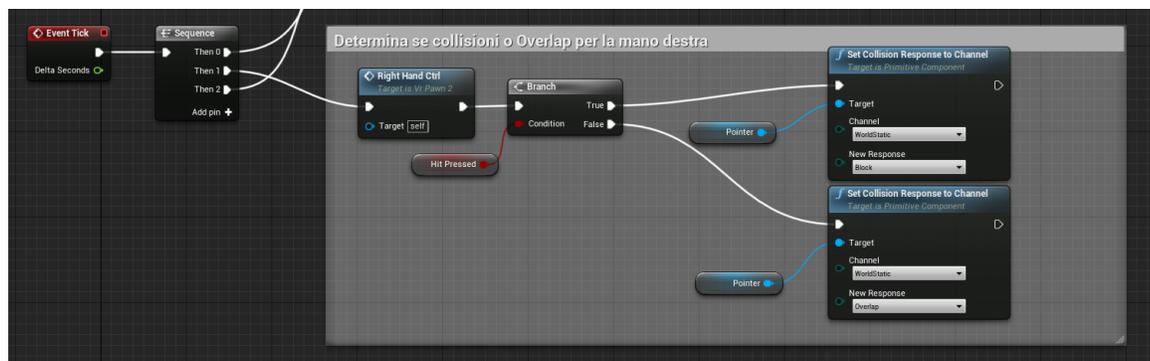
La chiave per far funzionare l'interattività è questa: le sfere sono state impostate come *WorldStatic* in *ObjectType* mentre il *Pointer* come *WorldDynamic* e, sotto la voce *ObjectResponses* di quest'ultimo, si è impostato il valore *Overlap* con oggetti tipo *WorldStatic*; questo significa che quando questi due

elementi (una sfera ed il *Pointer*) entrano in contatto non si bloccano reciprocamente ma viene registrata la sovrapposizione delle loro due *mesh*; a questo punto si passa la gestione al *VrPawn*.

Quando avviene la sovrapposizione si è aggiunta una vibrazione, per far intendere all'utente che l'elemento che si sta indicando con la mano destra è effettivamente interattivo.



Si è impostato il pulsante dell'interazione nel *right trigger* e, quando avviene la pressione del suddetto tasto, si sono associate delle funzioni all' *Event Tick* del *pawn* nel modo che si andrà ora a spiegare.



La funzione *RightHandCtrl* è adibita ad aggiornare (ad ogni frame) la posizione del puntatore, da qui si è imposta una semplice condizione booleana, se *HitPressed* (cioè se il pulsante per l'interazione è premuto durante quel frame), cambia il *collision response* verso gli oggetti di tipo *WorldStatic* passando da *Overlap* (che si riattiva se nei *frame* successivi il *Trigger* viene rilasciato) a *Block*, in questo modo è stato possibile generare degli *HitEvent* che vengono gestiti dalle sfere e ne attivano le varie funzioni.

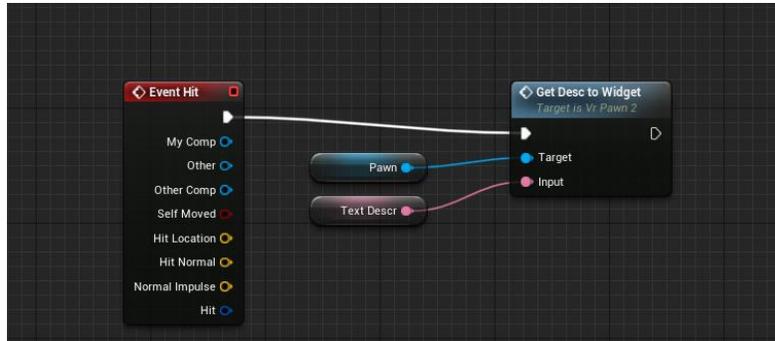


Figura: nel *blueprint* della “sfera informativa”, quando si genera un *HitEvent* si manda il testo (variabile inizializzata, come si vedrà, nel *LevelBlueprint*) al *widget* passando attraverso il *VrPawn* come spiegato le paragrafo dedicato.

### Lista delle sfere



Figura: Questa sfera, se colpita, manda il testo dei versi relativi alla mappa attuale al *Menù Widget*

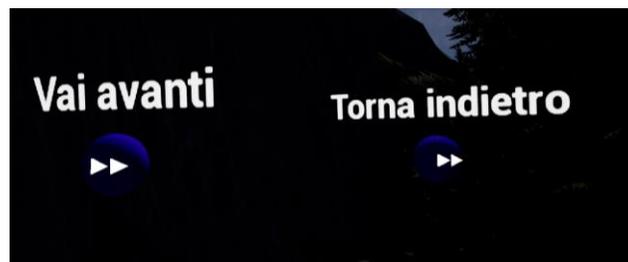


Figura: attraverso queste sfere è possibile cambiare mappa

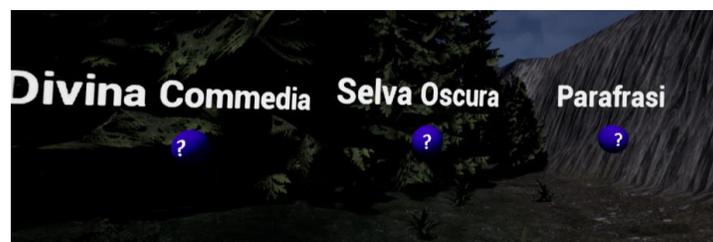


Figura: Le sfere con il simbolo “?” contengono le informazioni meta testuali che si possono leggere nel lato sinistro del *Menù Widget*



**Figure:** La sfera a sinistra permette di chiudere l'applicazione (vedi *Lv0*), quella a destra di mandare avanti la scena nel caso in cui nella mappa in questione i versi siano disposti come una sequenza di azioni (vedi *Lv3* e *Lv4*)

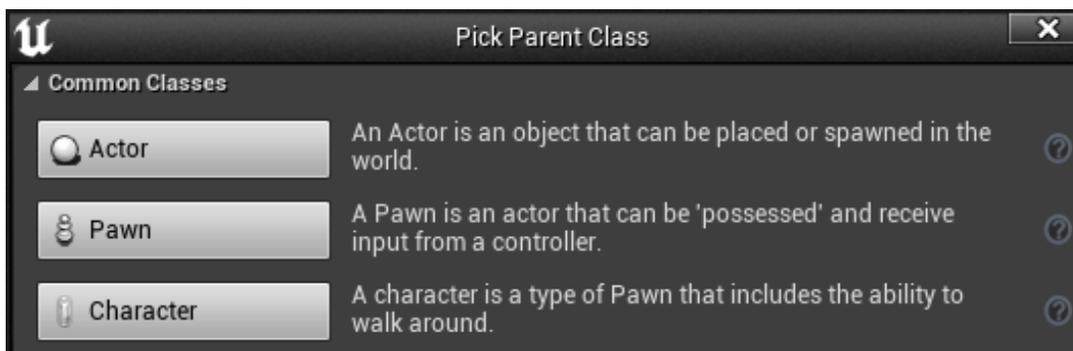
## Descrizione elementi variabili e Mappe

Come anticipato con “elementi variabili” si intendono tutti quegli oggetti che cambiano di comportamento e di funzioni a seconda della mappa e delle azioni che devono svolgere, si tratta dei vari *Character* impostati per Dante e dei relativi *animation blueprint*.

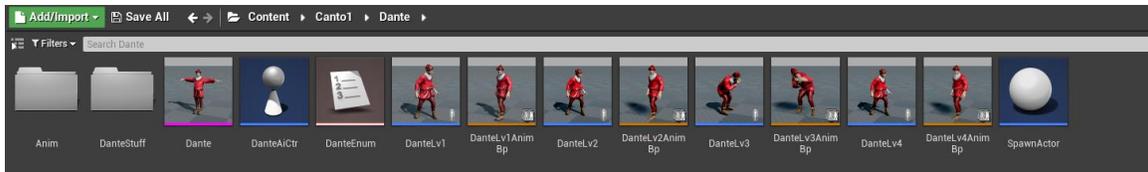
## Dante e Virgilio: IA e *animation blueprint*

Con il termine *Character* si intende, all'interno dell'ambiente di UE4, una classe che deriva dalla classe *Pawn*. Come si era accennato l'ambiente in questione utilizza per la programmazione il linguaggio C++ (di cui i *blueprint* non sono altro che una rappresentazione visiva del codice); essendo questo linguaggio *object oriented*, esso si basa sull'organizzazione in classi, sottoclassi e relazioni di ereditarietà. In particolar modo, parlando di elementi inseribili in una mappa, si segue tale linea gerarchica:

*Object -> Actor -> Pawn -> Character*



E, come in tutti i linguaggi *Object Oriented*, le classi figlio ereditano tutte le funzioni e le variabili della classe genitore più ne aggiungono delle altre, che a loro volta fanno ereditare alle proprie classi figlio. In particolar modo, come detto, per gestire le funzionalità di Dante e Virgilio si è ricorsi alla classe *Character*: ad entrambi gli si è associata un'IA, per consentire lo svolgimento autonomo di azioni, ed un *animation blueprint*, per la gestione delle animazioni. In particolar modo, per ciascuna mappa, poiché il personaggio di Dante doveva compiere azioni differenti e contestuali ai canti, si è preferito andare a creare diverse versioni del *Character*, con modifiche nel comportamento dell'IA e nello svolgimento delle azioni.



Si andrà ad analizzare, come esempio, il *Character* nominato *DanteLv1* ed il relativo *animation blueprint*.



Come si può notare in alto a sinistra un *Character* è dotato di una *Capsule Component* che gestisce la cosiddetta *hit box*<sup>47</sup> cioè il volume che il personaggio occupa nel gioco e con cui si può interagire; si trova poi una *mesh* che dà la forma al personaggio, l'elemento *Arrow* che semplicemente definisce qual è la direzione "avanti" ed il *Character Movement Component*, componente propria della classe *Character* per la gestione dei movimenti. Si è aggiunta una luce dietro il personaggio per far sì che sia visibile nell'ambiente buio della Selva Oscura.

<sup>47</sup> Come si può vedere non corrisponde necessariamente con la *mesh* ma è una sua approssimazione utile a seconda delle necessità; qui ha una forma basilare di capsula ed è necessaria a far sì che il modello con compenetri nella mappa o negli elementi ambientali quali rocce o alberi.

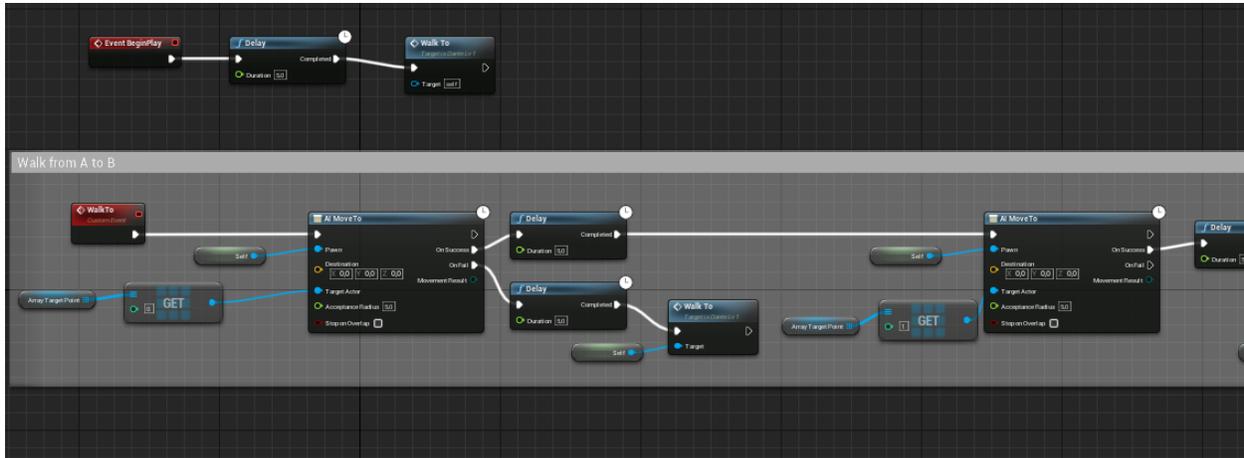


Figura: *blueprint* del comportamento di Dante nella prima mappa

Per quel che riguarda invece il *blueprint* del personaggio, si può notare dall'immagine il semplice comportamento dell'IA in questo primo livello: al *Begin to Play*, cioè quando il personaggio compie il suo *spawn*, dopo un *delay* di 5 secondi, si chiama la funzione *WalkTo* definita sotto, attraverso la quale si danno delle semplici istruzioni all'intelligenza artificiale mediante una successione di funzioni *AI Move To* che prendono in input delle posizioni da un array; quando il personaggio raggiunge uno di questi punti, si fanno trascorrere 5 secondi e poi si muove verso il punto successivo fino all'esaurimento delle variabili nell'array.

Mentre ciò avviene bisogna altresì impostare i movimenti ed i gesti che il personaggio deve compiere e lo si fa attraverso l'*animation blueprint* corrispondente, cioè un'interfaccia specifica per la gestione delle animazioni: questo è a sua formato da un *event graph*, che gestisce variabili e funzioni, e da un *animation graph*, che cambia le animazioni al cambiare dei parametri dell'*event graph*.

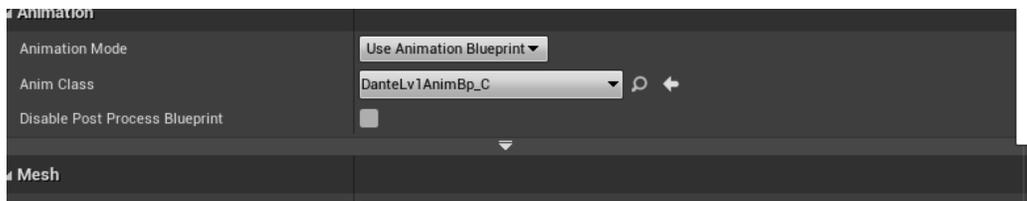


Figura: attraverso questa impostazione, nella classe del *Character* si associa ad un personaggio la classe di animazioni che deve seguire

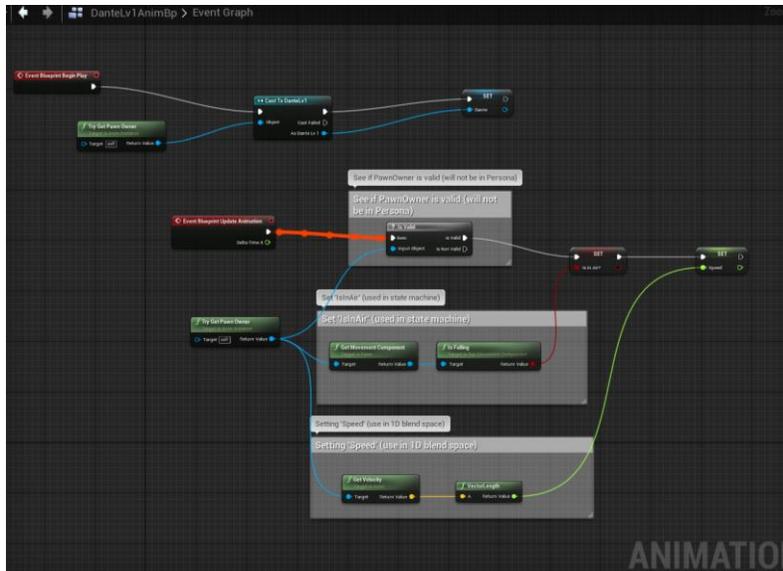


Figura: event graph di Dante Lv1

Si andrà ora descrivere il funzionamento dell'*event graph*. Quando si inizia un livello si attiva l'evento *BeginToPlay* relativo a questo *blueprint* ed associato allo *spawn* del *Character*, si inizializza una variabile della classe atta a contenere l'istanza attuale dell'oggetto *DanteLv1*, poi si gestisce il resto attraverso l'evento *UpdateAnimations* che, se il personaggio è valido e correttamente comparso, setta due variabili, un booleano *IsInAir* che cambia tra gli stati falso o vero a seconda che la *collision capsule* del *Character* tocchi terra o meno; e la variabile di tipo *float* chiamata *speed* che aggiorna ad ogni *frame* la velocità con cui si sta spostando il personaggio; in base a queste variabili avvengono modifiche dell' *animation graph*.

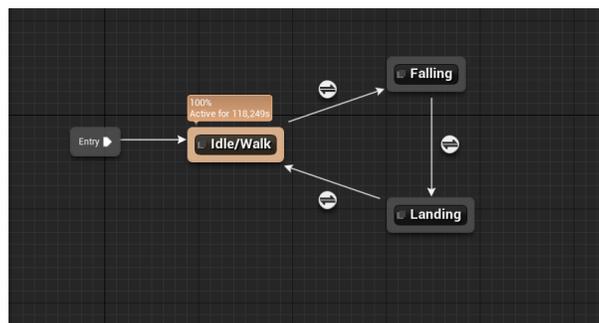
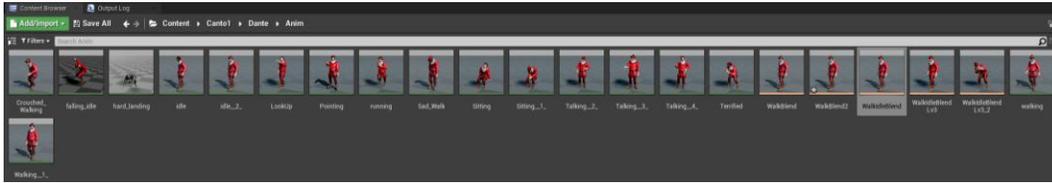


Figura: animation graph di DanteLv1

L'*animation graph* si presenta come un insieme di stati, a ciascuno dei quali è associata un'animazione, tra loro interconnessi mediante condizioni aggiornate e gestite nell'*event graph*:

quando la specifica condizione si realizza avviene il passaggio da uno stato ad un altro; in questo primo livello, ad esempio, si sono utilizzati tre stati cioè *Idle/walk*, *Falling* e *Landing*<sup>48</sup>.



**Figura: Asset di animazioni utilizzate**

Il passaggio tra i tre stati appena detti sono contestuali alle azioni che avvengono nella prima sezione del canto: Dante si ritrova nella Selva Oscura ed inizia a guardarsi intorno per esplorare l'ambiente circostante. Lo stato di partenza è *Idle/Walk* a cui è stato annesso, non un'animazione singola, ma un *blend space*, cioè un'interpolazione tra due animazioni, una di *Idle* (stato "non in movimento" in cui Dante si guarda attorno circospetto) e una di *Walk* (Dante si muove ed inizia l'animazione di "camminata"), il fattore su cui viene interpolato il *blend space* è, appunto, la velocità a cui si sta muovendo il personaggio che viene ricevuta dalla variabile *speed* citata precedentemente.



**Figur2: Preview del *blend space* con velocità uguale a zero e il personaggio in *idle* (sinistra) e con velocità massima con animazione di *walking* (destra)**

<sup>48</sup> Tutte le animazioni sono state scaricate insieme al modello automaticamente riggato da Mixamo, il quale mette a disposizione un vasto numero di animazioni in *motion capture* gratuitamente scaricabili.

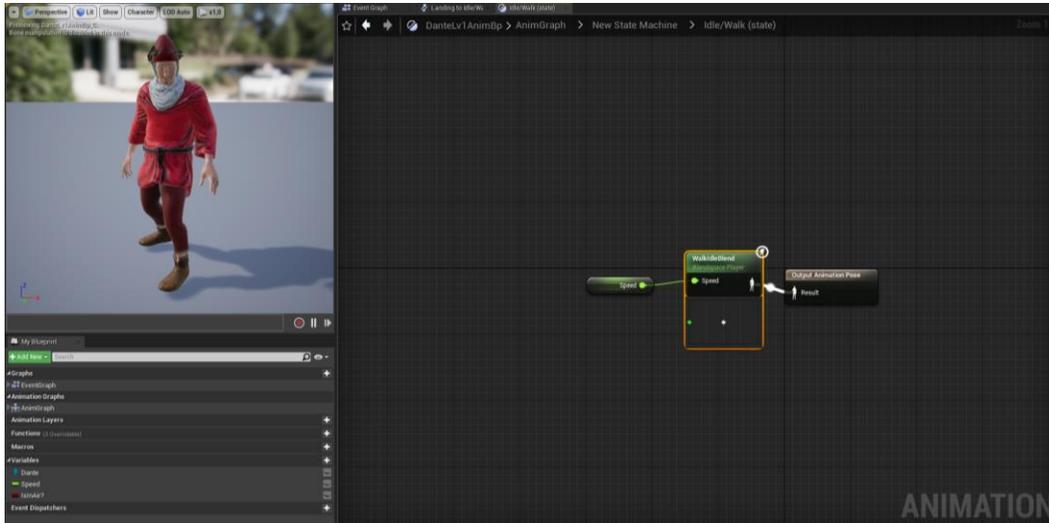


Figura: funzionamento interno dello stato *Idle/Walk*, si mette in input la variabile *float speed* per interpolare il *blend space*

Gli stati di *Falling* e *Landing* sono consecutivi e si basano sulla variabile dell'*event graph* denominata *IsInAir*; una guardia booleana inserita tra lo stato *Walk/Idle* e *Falling*.

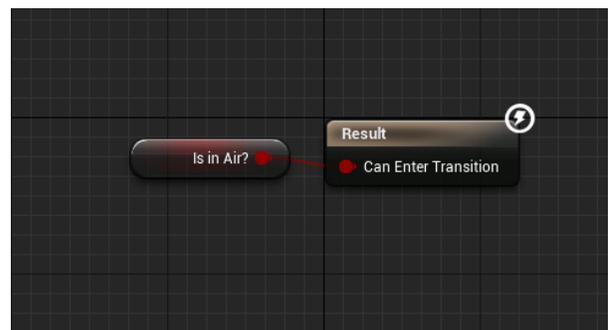
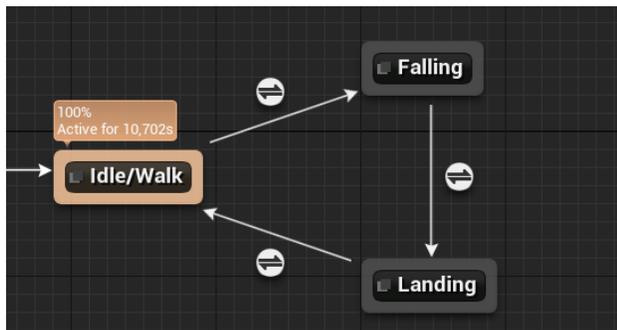


Figura: i cerchi bianchi con le frecce all'interno rappresentano le guardie o regole di transizione tra due stati; a destra la regola di transizione inserita, si può leggere come: *if IsInAir == True => Then Falling*.



Figura: animazione di caduta

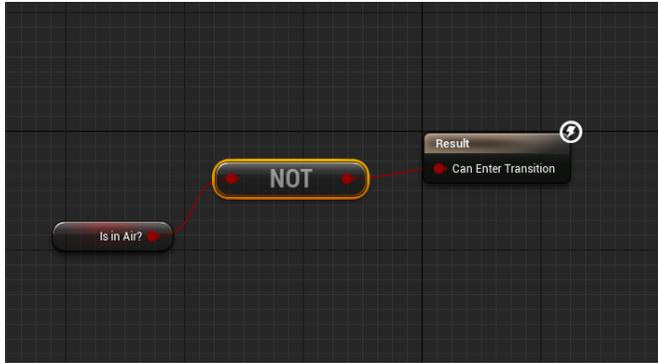


Figure: Quando la *collision component* del *Character* tocca terra, la variabile *IsInAir* diventa *false* e si passa all'animazione di *Landing*

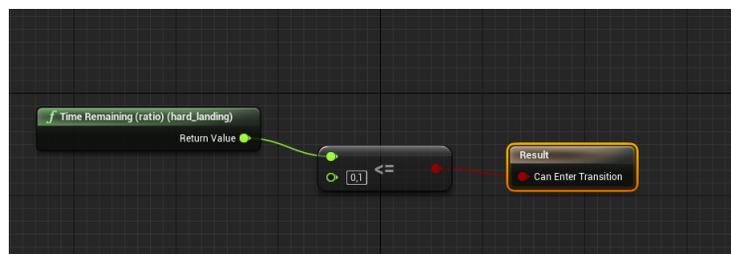
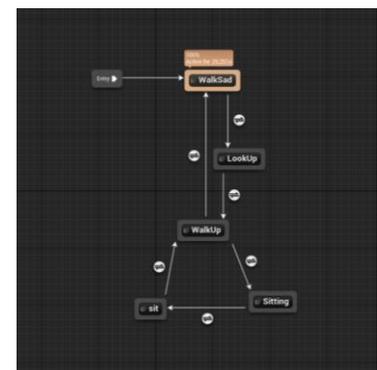
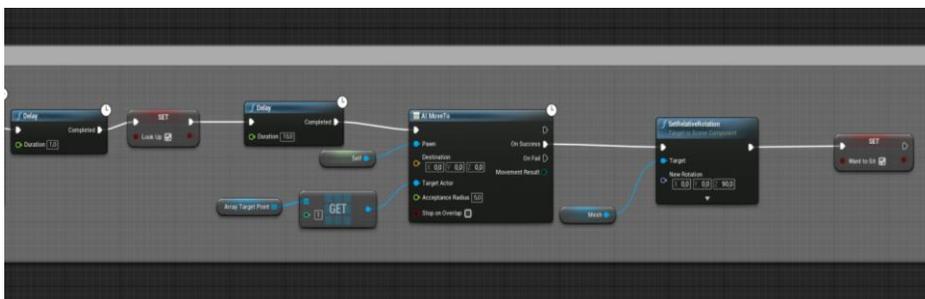


Figura: Infine l'ultima regola di transizione, quando l'animazione di *Landing* sta per finire (*Time remaining ratio <= 0,1*) si torna allo stato di *Idle/Walk*

Come anticipato i comandi alla IA e gli *animation blueprints* cambiano per ogni mappa a seconda delle necessità di movimenti, spostamenti ed animazioni. Si analizzeranno velocemente le altre soluzioni.

## DanteLv2



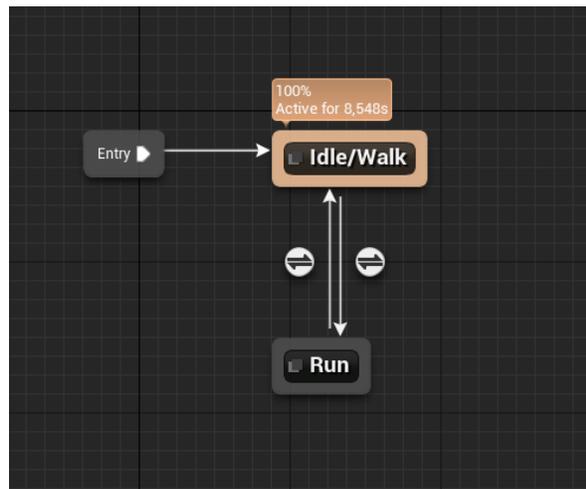
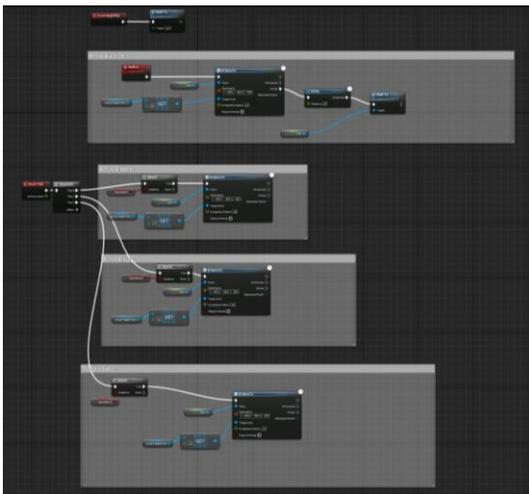
Il funzionamento di *DanteLv2* è simile al primo se non per due variabili booleane che vengono attivate tra due comandi *AIMoveTo* chiamate rispettivamente *LookUp* e *WantToSit*, questo appunto per seguire il flusso narrativo del poema: Dante esce dalla *Selva Oscura* sconfortato e si ritrova ai piedi del

*dilettoso monte* dietro al quale sta sorgendo il sole che lo rinfranca. Dal punto di vista delle animazioni questa scena è stata resa nel seguente modo: Dante si muove fino ad un primo punto della mappa con l'animazione di camminata abbattuta quando, in parallelo all'audio, vede il sole sorgere dietro al monte (*set "LookUp" = true*) e, ripreso coraggio, cambia l'animazione di camminata fino ad arrivare ad una roccia in cui si siede per riposare e contemplare la strada percorsa (*set "WantToSit" = true*).



Figure: animazioni utilizzate nella seconda mappa, da sinistra a destra: Camminata abbattuta, Dante guarda il sole, passo sicuro ed animazione di seduta

### DanteLv3



L'IA di *DanteLv3* è leggermente differente, in quanto si è suddivisa l'azione della scena in sotto parti ed il passaggio tra una parte ad un'altra dipende dall'interazione dell'utente; quindi, per far sì che l'IA ricevesse questo input esterno si è fatto ricorso all' *Event Tick*. I versi in questione sono quelli che riguardano le tre fiere ed il passaggio tra l'incontro di una e la successiva viene dettata

dall'interazione dell'utente con un'apposita sfera che manda avanti l'azione, tale elemento non fa altro che cambiare il valore del booleano, dando accesso al *AIMoveTo* successivo.

### DanteLv4 e Virgilio

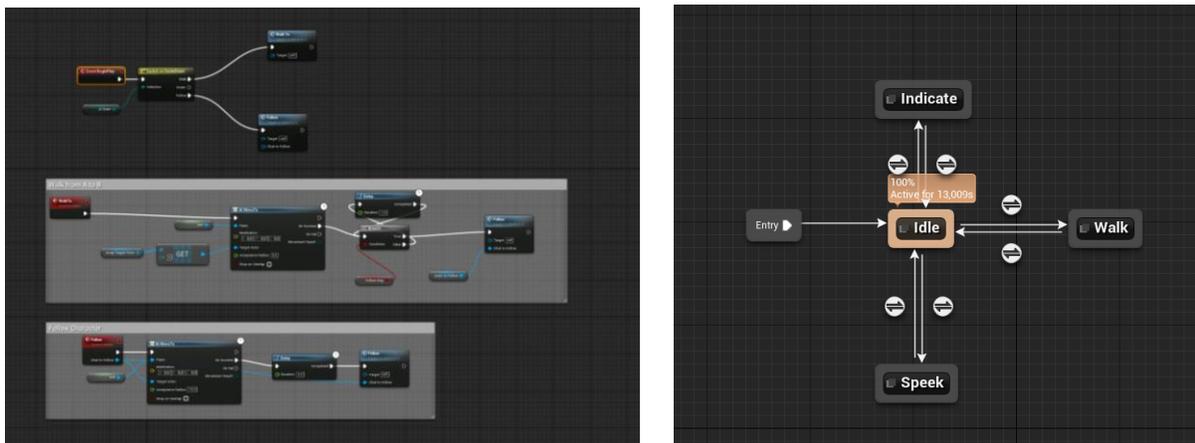


Figura: IA e gestione animazioni di "DanteLv4"

Nella quarta mappa si alternano due comandi per l'IA, il *MoveTo* a cui succede, utilizzando sempre un booleano come guardia, un altro stato di movimento chiamato *Follow* che consente al *Character* di Dante di seguire quello di Virgilio alla fine della loro linea dialogica. Il *loop Branch/Delay* è stato utilizzato per simulare l'azione di un *Event Tick*: ogni secondo (e non ad ogni *frame*) viene valutata la variabile booleana di guardia, se è falsa si ripete il *loop*; invece, se viene restituita vera (il cambio di valore avviene nel *Level blueprint*), significa che la scena volge al termine e Dante inizia a seguire Virgilio. L'*animation graph* gestisce le animazioni di vari momenti del dialogo tra i due poeti.

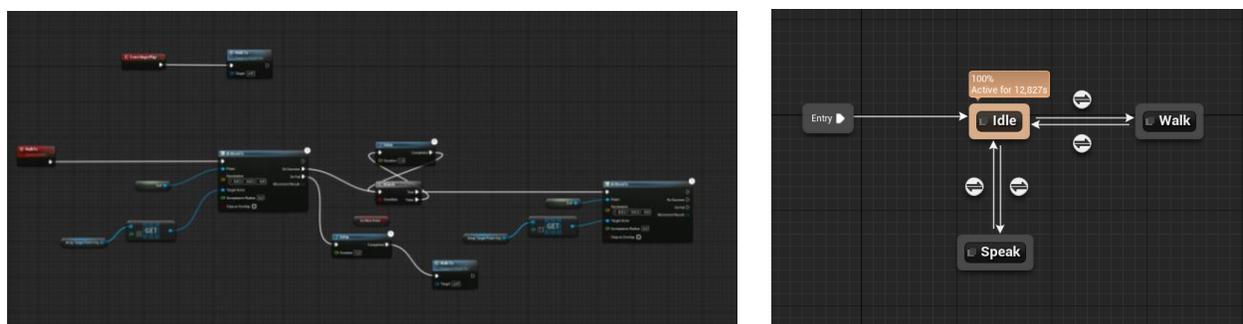


Figure: IA e gestione animazioni di "Virgilio"

I *blueprints* di Virgilio non nascondono particolari novità rispetto a ciò che si è già visto. Si può notare il *loop Branch/Delay* simile a quello appena descritto per Dante e con la medesima funzione, la sola

differenza è che, al cambiare valore della guardia, Virgilio ha un altro *MoveTo* che lo porta fuori dalla scena seguito da Dante mediante il comando *Follow*.

## Descrizione dei *Level blueprint* relativi alle mappe

Si andranno ora a descrivere i livelli creati ed il loro funzionamento logico attraverso quelli che UE4 definisce come *Level blueprint* cioè delle interfacce logiche che permettono agli oggetti di venir istanziati, di cambiare nelle proprie variabili e funzioni, di attivare eventi nonché di interagire e comunicare reciprocamente tra loro e con l'utente.

### *Mappa Tutorial*

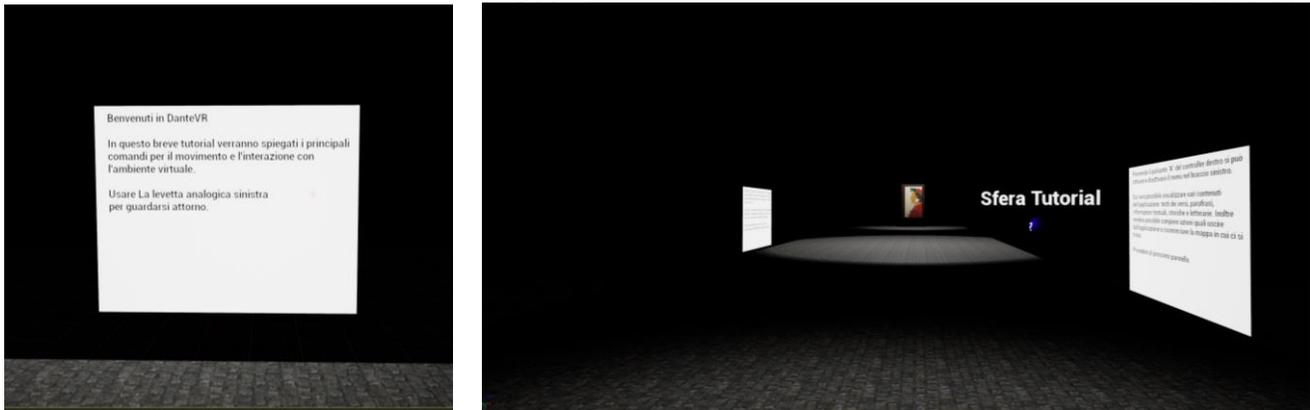


Figure: *screenshot* della mappa tutorial

Questa prima mappa è molto semplice e consiste in un corridoio rettilineo con dei pannelli informativi (si sono utilizzati oggetti di tipo *widget*) il cui scopo è spiegare al giocatore i tasti necessari per interagire con l'ambiente virtuale; in fondo si trova un libro interattivo che consente di passare, una volta conosciuti i comandi base, alla *Mappa 0*.

Il *Level blueprint* di questo livello non è complesso, semplicemente si è utilizzata questa interfaccia per procedere allo *spawn* degli elementi informativi e inizializzare le relative variabili testuali.

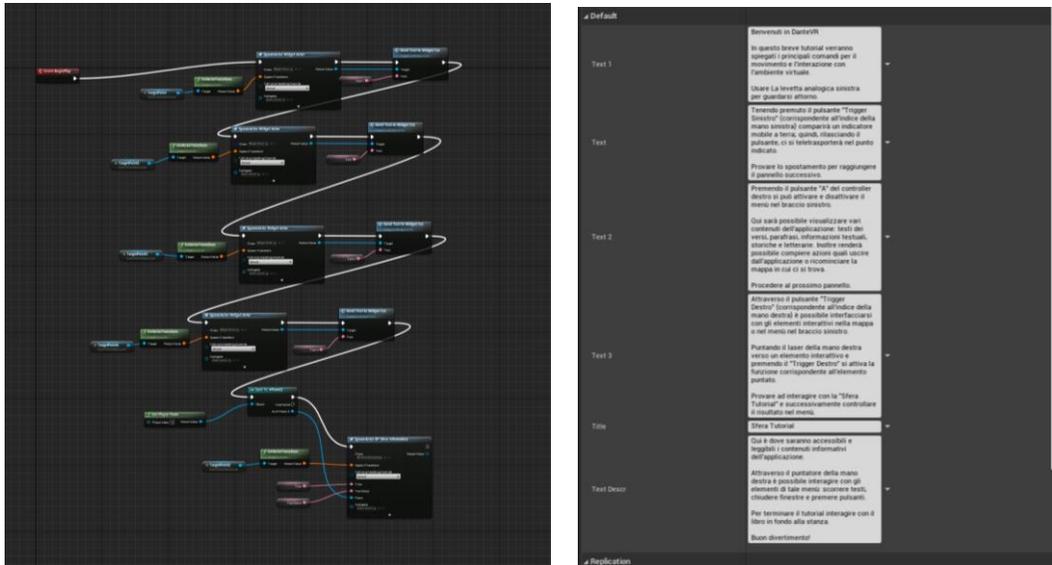


Figure: Level blueprint e variabili di testo inizializzate nei pannelli informativi

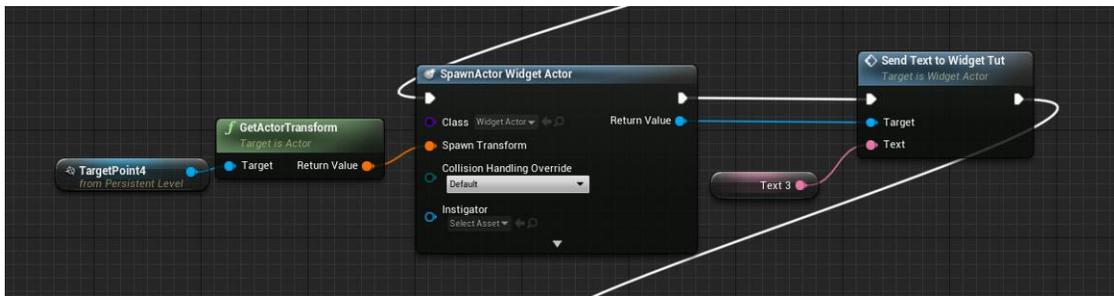


Figura: per ogni pannello sono state svolte le stesse azioni: si è chiamata una funzione *SpawnActor* della classe *WidgetActor*, gli si è fornita la posizione di *spawn* attraverso dei "arquetPoint" posizionati nella mappa, e gli si è associato il testo contenuto nell'apposita variabile

Libro Animato

Il libro animato in Blender e precedentemente descritto verrà ora approfondito nel suo funzionamento e ruolo in UE4 in quanto è stato utilizzato in questa mappa.



Figura: Libro chiuso e libro aperto dopo che è avvenuto l'evento *OnHit*

Questo elemento ha un comportamento simile a quello delle sfere ed agisce attraverso un evento *OnHit* che va ad azionare l'animazione di apertura e rende visibile un menù widget interattivo per poter passare alla *Mappa 0* del primo canto.

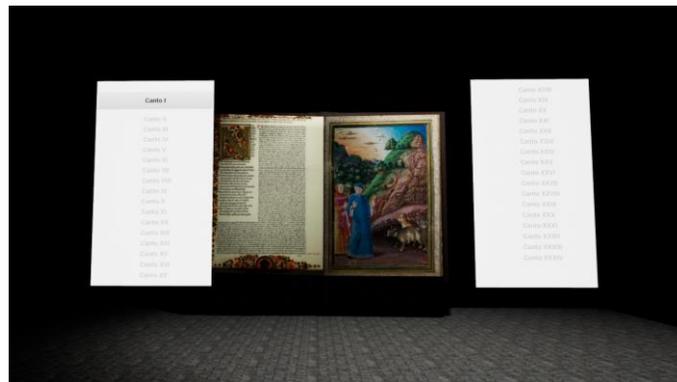


Figura: libro dopo l'interazione e con menù *widget* visibile, solo il primo canto risulta selezionabile

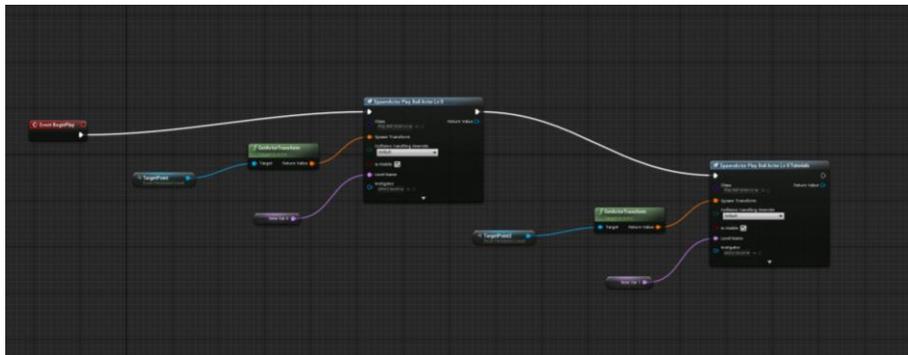
## Mappa 0



La mappa definita *Lv0* è stata pensata per essere un *hub*, uno snodo per poter raggiungere altri ambienti. Sono presenti tre sfere, una per ritornare alla *Mappa Tutorial*, quella con la scritta “Selva Oscura” che permette di spostarsi alla mappa *Lv1* con i primi versi della cantica ed un’ultima per terminare l’applicazione. L’ambiente è stato reso come una piccola area boschiva notturna con la luna ed un cielo stellato, per richiamare da una parte la *Selva Oscura* che si sta per affrontare, da un’altra come citazione all’ultimo verso dell’*Inferno*:

E quindi uscimmo a riveder le stelle<sup>49</sup>

Il *level blueprint* relativo è molto semplice e si occupa dello *spawn* e dell’inizializzazione degli elementi interattivi appena descritti.



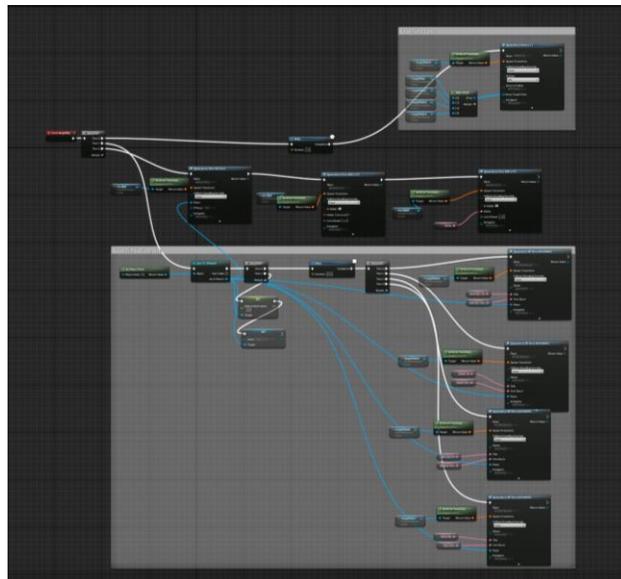
<sup>49</sup> Inferno XXXIV, 139

## **Mappa 1**

Questa mappa riguarda i primi nove versi del primo, si è modellato un ambiente notturno, con il personaggio di Dante che vaga incerto per la zona boschiva mentre la voce di Gassman recita i versi.

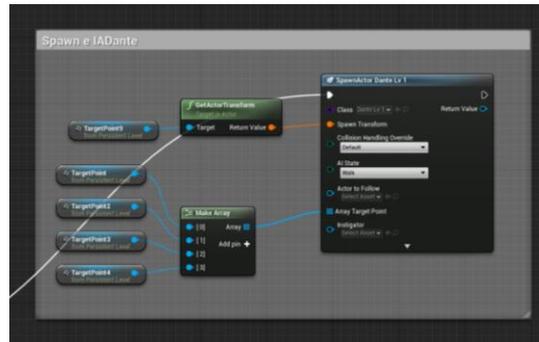


Il *level blueprint* relativo può essere diviso in tre sezioni funzionali che si diramano da una *sequence* derivante dal *Begin to play*.

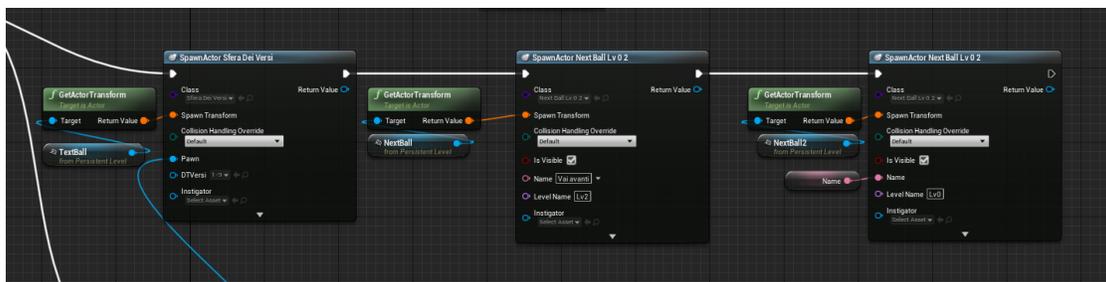


**Figura: level blueprint complessivo**

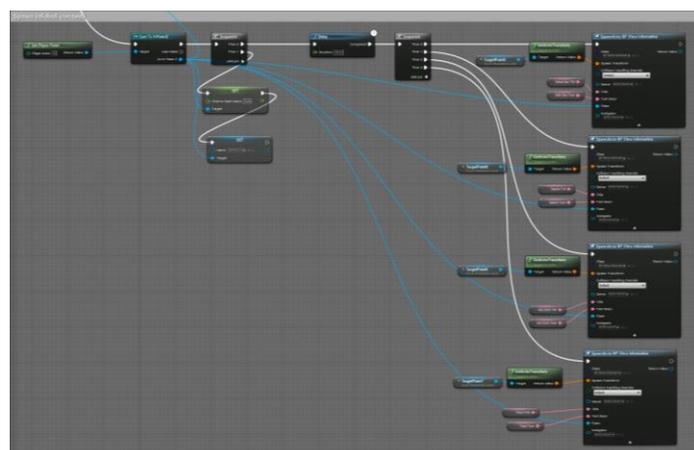
Il primo *then* si occupa dello *spawn* del *character* di Dante a cui viene dato in input un array di posizioni per definire all'IA la successione di punti che deve seguire mediante la funzione *Move to*.



Dal secondo *then* si diramano delle funzioni *SpawnActor* per la comparsa nell'ambiente di alcune sfere interattive: quella per mandare al *widget* il testo dei versi, una per tornare al *Lv0* ed un'altra per andare subito al *Lv2*.



Infine, dall'ultimo *then* dipendono delle funzioni per associare l'audio del canto al *VrPawn* e far comparire, dopo un delay di 55 secondi (corrispondente alla durata dei versi recitati), le sfere con i contenuti informativi; si è deciso di farle comparire dopo la fine della lettura per non distrarre troppo l'attenzione dell'utente dalla scena.



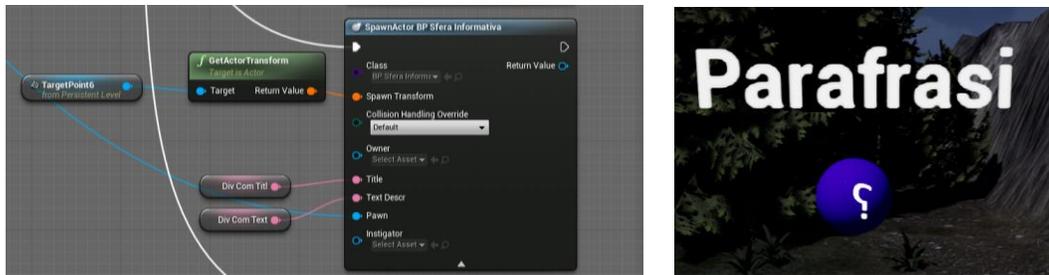


Figura: a ciascuna sfera sono collegate due variabili testo definite nel *level blueprint* stesso, una con il titolo della sfera (quello che compare nell'ambiente virtuale sopra l'oggetto stesso), un'altra con il testo che, in caso di interazione, verrebbe inviato alla *widget* del *VrPawn*

## Mappa 2



Nella seconda mappa si sono rappresentati i versi da 10 a 27, Dante riesce ad uscire dalla Selva Oscura e giunge alle pendici di un monte quando:

16 guardai in alto e vidi le sue spalle  
vestite già de' raggi del pianeta  
che mena dritto altrui per ogni calle.

19 Allor fu la paura un poco queta,  
che nel lago del cor m'era durata  
la notte ch'ì passai con tanta pietà.

Viene rinfrancato dalla vista del sole (allegoria della luce divina) che sorge dietro al monte e si approfitta del momento di sollievo per osservare, alle sue spalle, l'oscuro tragitto che aveva percorso:

22 E come quei che con lena affannata,  
uscito fuor del pelago a la riva,  
si volge a l'acqua perigliosa e guata

25 così l'animo mio, ch'ancor fuggiva,  
si volse a retro a rimirar lo passo  
che non lasciò già mai persona viva.

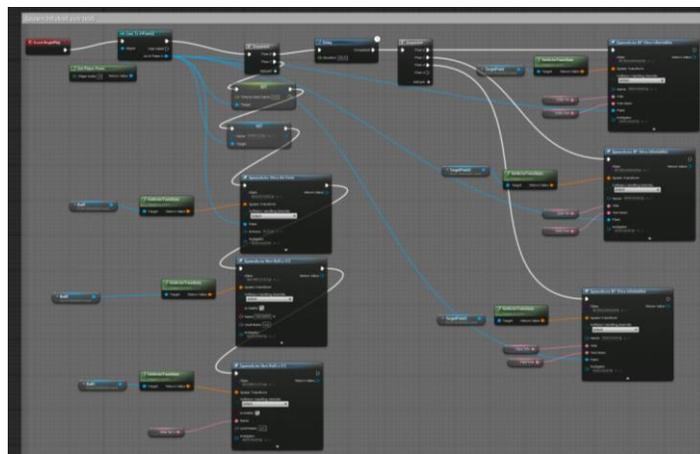
Su queste azioni si basa la struttura narrativa di questo *level blueprint*: Dante cammina sconsolato uscendo dalla selva; ad un certo punto si ferma per guardare in alto i raggi del sole dietro al monte; prende coraggio (azione che si è decisa di esprimere attraverso il cambio di animazione di camminata) e prosegue fino ad arrivare ad un masso su cui si siede per riposarsi e guardare il tragitto percorso.





Figure: sequenza di azioni, Dante inizialmente scontento, scorge il sole dietro al monte e cambia passo, per poi sedersi e guardare il percorso compiuto

Il *level blueprint* di questo livello è simile a quello della mappa precedente e si utilizza per far *spawnare* i vari elementi interattivi e per definire ed inizializzare variabili.



Al contrario della mappa precedente, qui non si è fatto compiere lo *spawn* di Dante dal *level blueprint* ma lo si è posizionato dall'editor del livello e da qui si sono impostate le variabili utili per gestire le animazioni.



### **Mappa 3**

La struttura logica di questa mappa è più complessa. Qui si dà atto ai versi che vanno da 28 a 60 quindi si riferisce all'incontro di Dante con le tre fiere, in ordine: una lonza, un leone ed una lupa; allegorie dei vizi capitali che affliggono l'umanità (lussuria, superbia ed avarizia). Proprio in considerazione del contenuto appena detto si è suddivisa l'azione in tre sotto parti, una per ciascuna fiera che ostacola l'ascesa di Dante alla salita verso la cima del monte; il passaggio tra una sottosezione ed un'altra è data dall'interazione del giocatore con una sfera di progressione.



**Figura: Dante arriva davanti alla Lonza, compare una sfera con l'icona "Play" che, attraverso l'interazione, permette di proseguire nel canto**



Figura: I modelli del leone e della lupa<sup>50</sup>

Come anticipato il *level blueprint* di questa mappa è un po' più complesso dei precedenti:

---

<sup>50</sup> I modelli sono stati scaricati rispettivamente ai seguenti link:

Lince: <https://www.cgtrader.com/free-3d-models/animals/mammal/wild-trot>

Leone: [https://open3dmodel.com/3d-models/3d-model-african-male-lion\\_137732.html](https://open3dmodel.com/3d-models/3d-model-african-male-lion_137732.html)

Lupo: <https://www.cgtrader.com/free-3d-models/animals/mammal/wolf-rigged-animated-and-game-ready>

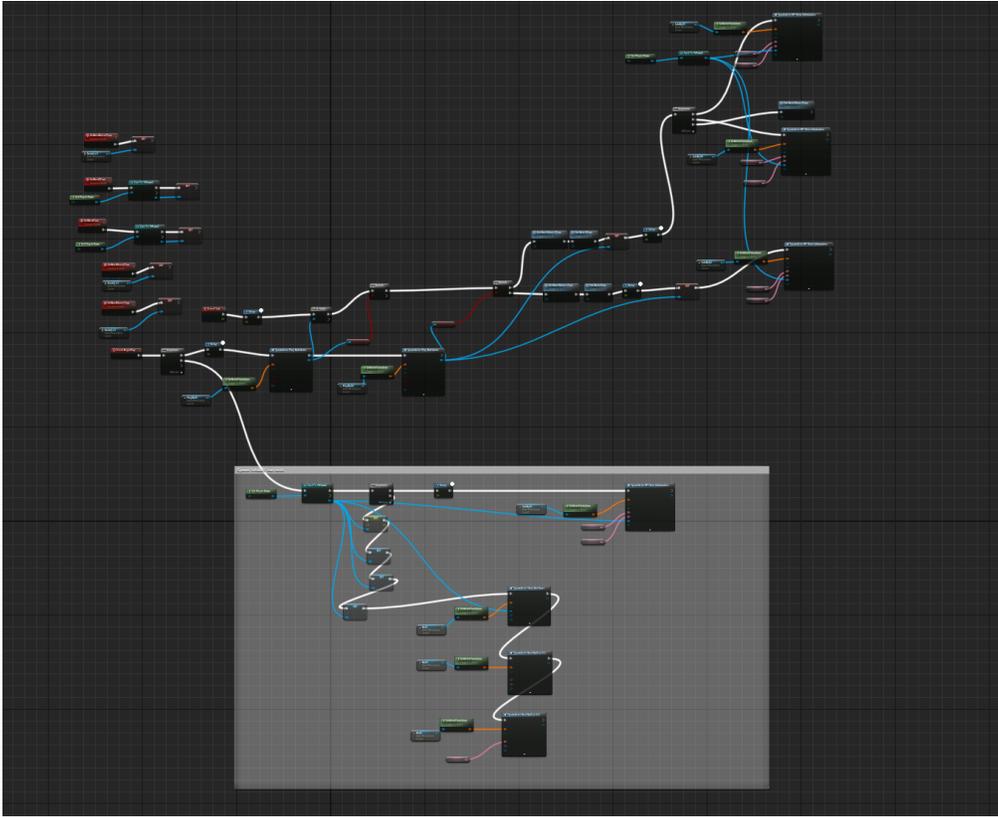


Figura: Level blueprint complessivo di "Lv3"

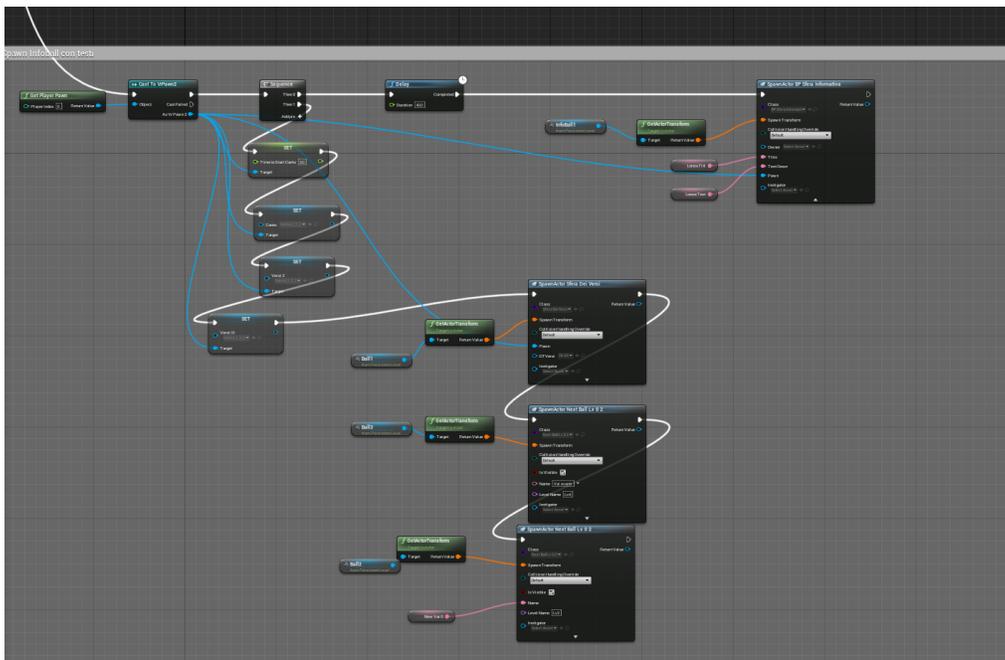


Figura: questa sezione è simile a quella delle mappe precedenti ed è utilizzata per lo spawn di alcune sfere e l'inizializzazione di variabili

Siccome l'avanzamento dell'azione dipende dall'interazione dell'utente non si può far ricorso solo all'evento *Begin to Play* ma si necessita anche dell'*Event Tick*: ad ogni *frame* viene valutato se un'interazione è stata compiuta (attraverso il controllo su alcune variabili booleane appositamente impostate in *DanteLv3* e nel *VrPawn*); nel caso l'utente abbia interagito con la sfera "avanti" la guardi cambia valore e si procede con l'azione successiva.

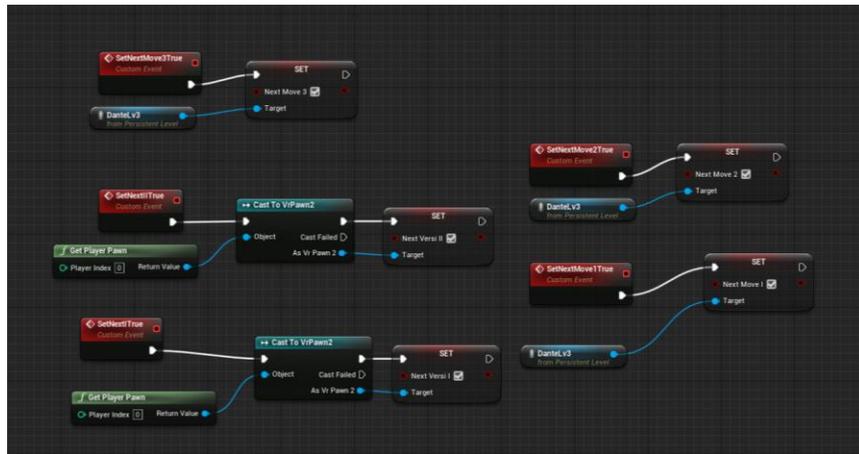
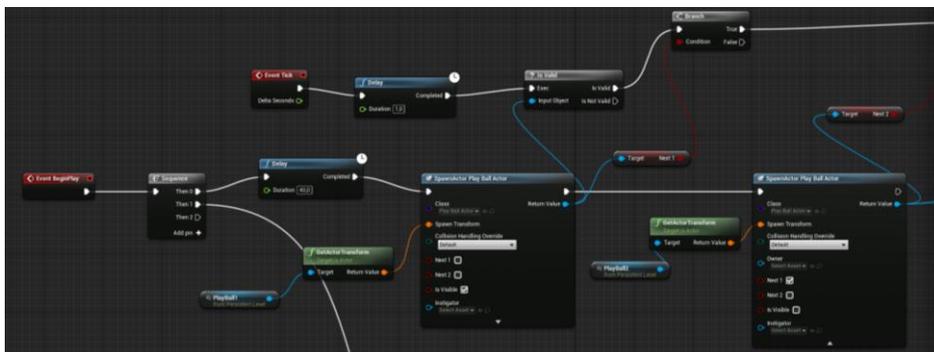


Figura: Variabili booleane in *VrPawn* e *DanteLv3* impostate, attraverso eventi, nel *level blueprint*

Nello specifico, la successione logica di questa mappa è la seguente:

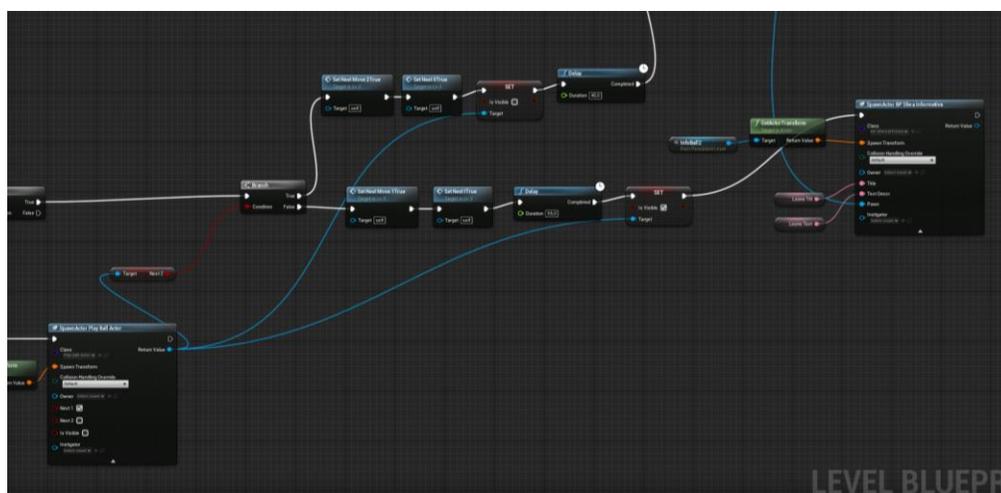
- da *BeginToPlay*, dopo 40 secondi (il tempo di decantare i versi relativi alla prima fiera), fa comparire due sfere (una vicina alla lince, con le booleane *next1* e *next2* false e la variabile *isVisible* vera; e una al leone, con *next1* vera e le altre variabili false)



- Da qui inizia il flusso che parte da *Event Tick*; si è messo un *delay* di 1 secondo per far sì che queste funzioni non vengano eseguite ad ogni *frame* ma ad ogni secondo, alleggerendo il peso

computazionale complessivo. Quando *next1* della prima sfera diventa vero, cioè quando il giocatore ha interagito con essa, si passa ad una condizione annidata.

- Questa seconda valutazione si basa sulla variabile *next2* della seconda sfera (quella vicina al leone); se la valutazione booleana è falsa (quindi se il giocatore non ha ancora interagito con questo elemento) si dà all'IA di Dante indicazione di proseguire (*SetNextMove1True*) e si manda l'input per la lettura dei versi successivi riguardanti la seconda sfera (*SetNextMove1True*), si aspetta un *delay* di 55 secondi per concludere le azioni appena dette e si rende visibile la seconda sfera per avanzare (insieme ad una sfera informativa riguardante il leone).



- Quando l'utente interagisce con l'elemento per avanzare inizia la sezione finale, l'incontro con la lupa e la fuga di Dante che è costretto a ricacciarsi indietro nella Selva Oscura, terminando così le azioni inerenti alla terza mappa.

52 questa mi porse tanto di gravezza  
con la paura ch'uscita di sua vista,  
ch'io perdei la speranza de l'altezza.

55 E qual è quei che volentieri acquista,  
e giugne 'l tempo che perder lo face,  
che 'n tutti suoi pensier piange e s'attrista;

58 tal mi fece la bestia senza pace,  
che, venendomi 'ncontro, a poco a poco  
mi ripigneva là dove 'l sol tace.

Come prima, quando la condizione del secondo *branch* diventa vera, si danno indicazioni di azione alla IA e si prosegue con la lettura della sezione in questione, si aspetta che queste terminino con un *delay* di 40 secondi, poi si fanno comparire delle sfere informative vicino al modello del lupo (una con la parafrasi della sezione completa e una con la descrizione della fiera) e si danno indicazioni a Dante di iniziare a correre indietro verso la Selva Oscura a causa del terrore infusogli dalla belva (attraverso un'animazione impostata nell'apposito *animation blueprint*).

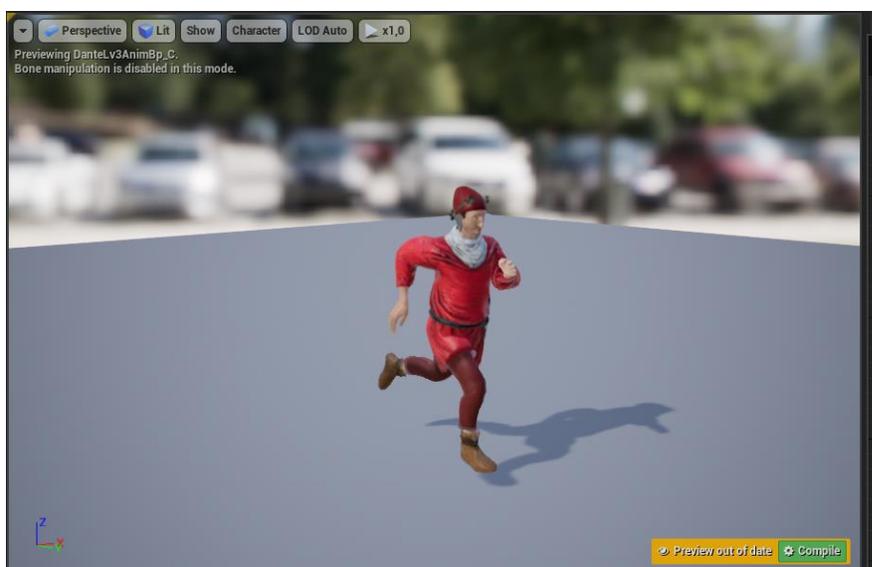
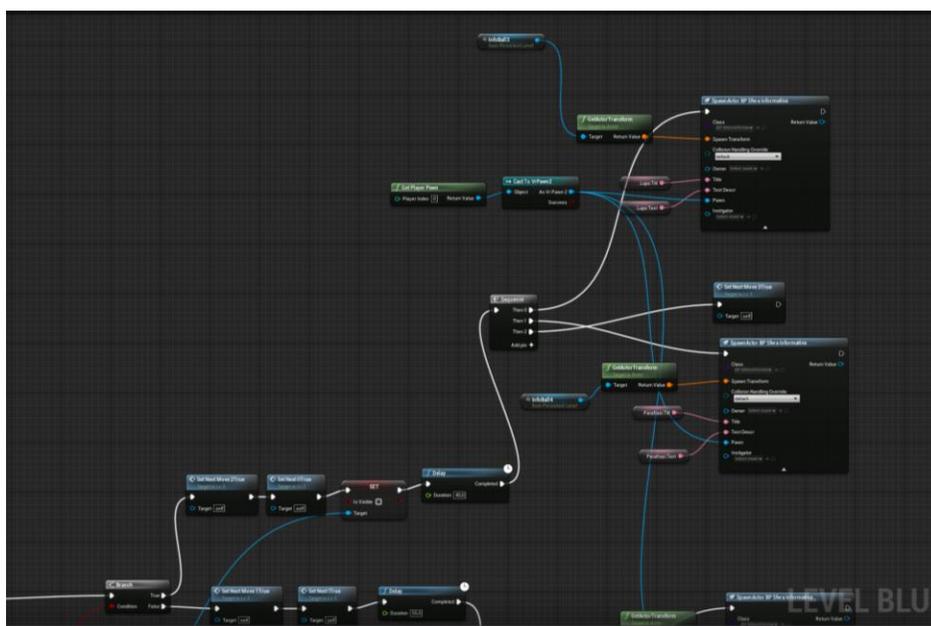


Figura: Animazione di Dante in fuga

## Mappa4

In questa sezione si va verso la conclusione del canto contenendo i versi che vanno da 61 al 136 che riguardano l'incontro tra Dante e Virgilio e la spiegazione da parte di quest'ultimo del viaggio che dovrà percorrere il protagonista attraverso i regni ultraterreni. In queste terzine viene rappresentata una linea dialogica tra Dante e Virgilio e la progressione del discorso viene data in gestione al giocatore attraverso le medesime sfere viste nella mappa precedente. Ogni volta che uno dei due protagonisti prende parola, nei relativi *animation blueprint* vengono attivati degli stati con animazioni di dialogo.

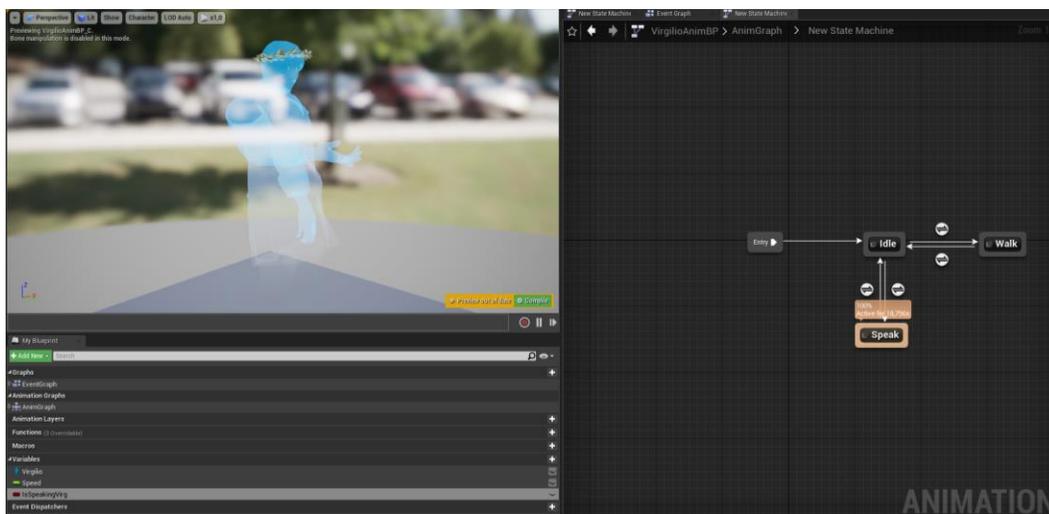


Figura: modello di Virgilio che compie una delle animazioni contestuali al dialogo quando la relativa variabile di guardia viene resa *true*

Siccome la progressione della scena viene data in mano al giocatore, il *level blueprint* sarà simile a quello della mappa precedente, quindi con funzioni che si diramano da *Event Tick* e con condizioni da rendere vere attraverso l'interazione con le sfere per far proseguire l'azione.

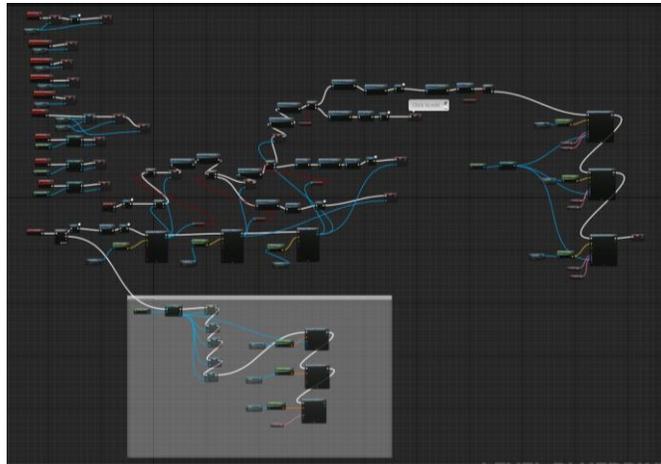


Figura: Level blueprint complessivo

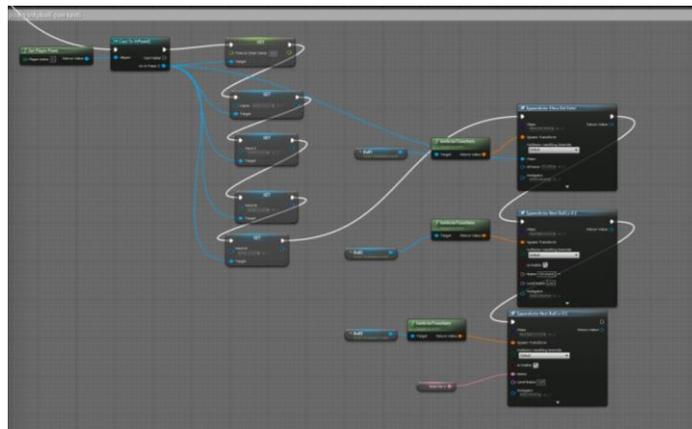


Figura: Qui si trovano funzioni per la comparsa di sfere e l'inizializzazione delle variabili necessarie

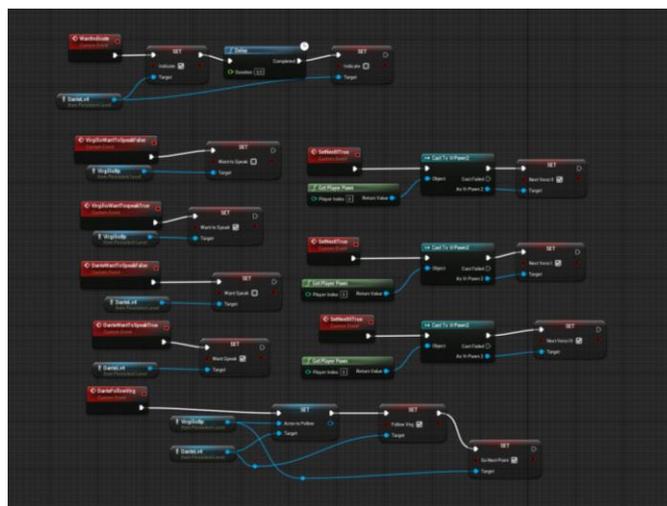
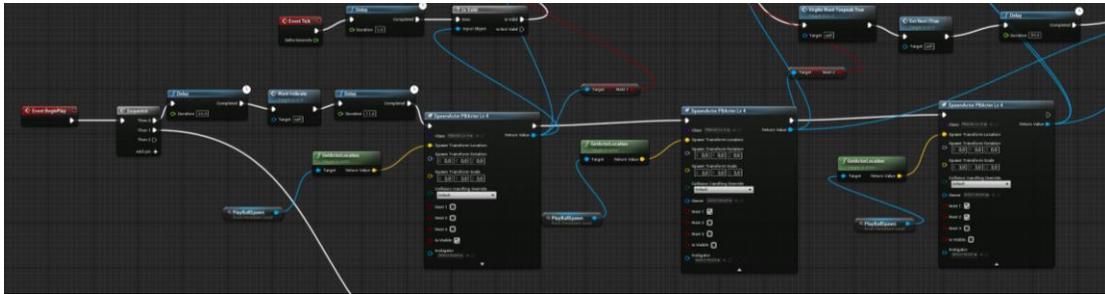


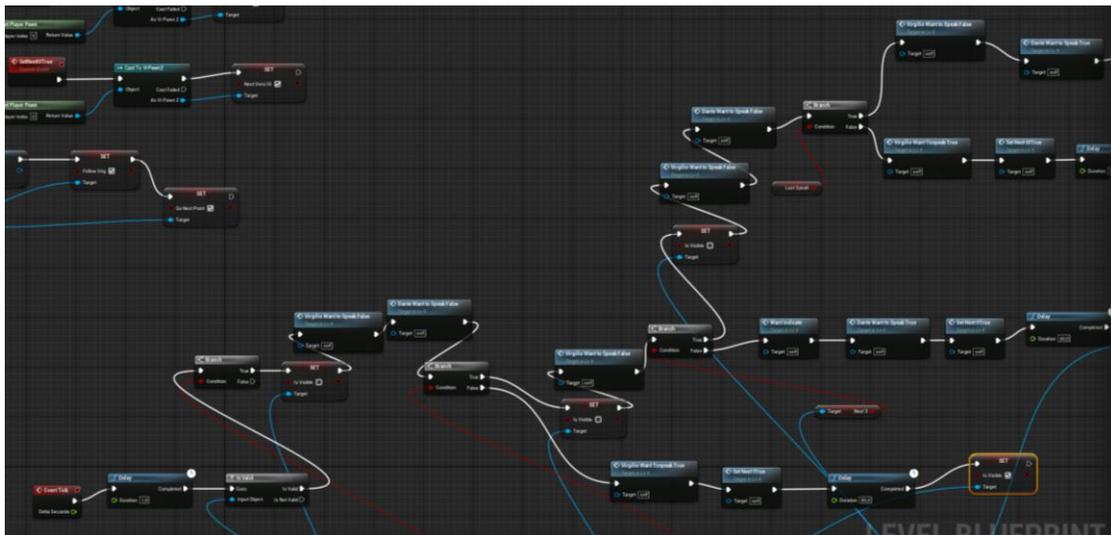
Figura: Questa sezione riguarda gli eventi per rendere *true* le guardie e consentire la progressione della scena e varie altre azioni (come lo *switch* tra le animazioni dei personaggi o l'impostazione *Follow* all'IA di Dante)

Si andrà ora ad analizzare il flusso logico principale:

- Analogamente alla sezione precedente, dal *BeginToPlay* consegue lo *spawn* delle sfere di progressione con variabili booleane di *next* e *visibility* appositamente inizializzate che andranno a fungere da guardie alle condizioni per far proseguire l'azione successivamente all'interazione dell'utente.

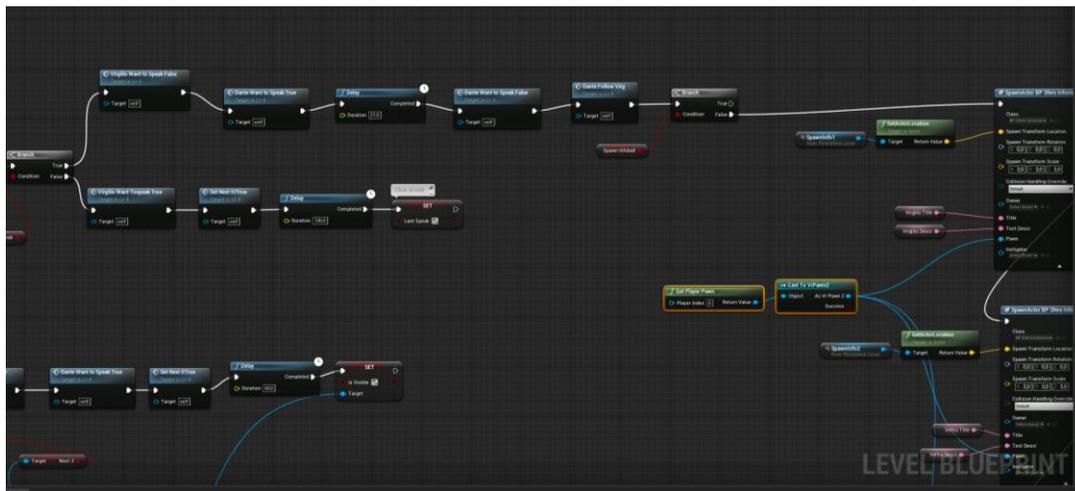


- In aggiunta alla mappa precedente, ci sono da alternare le animazioni di dialogo tra i due personaggi e lo si fa attraverso gli eventi appositi: quando un personaggio inizia a parlare *VirgilioWantToSpeakTrue*, l'altro ferma le animazioni di dialogo *DanteWantToSpeakFalse* e, quando il primo finisce la frase, compare una sfera di progressione per permettere al giocatore di mandare avanti l'azione quando preferisce.



- Come ultimo *step*, alla fine dello scambio dialogico tra i due protagonisti, Dante inizia a seguire Virgilio che condurrà il protagonista attraverso i reami infernali; per far allontanare insieme i due personaggi si imposta l'IA di Dante come "Follow", attraverso l'apposito evento e con

Virgilio come *target*, mentre a quest'ultimo si danno indicazioni di movimento. Oltre a ciò, alla fine del canto si procede allo *spawn* delle ultime sfere informative.



Screenshot dell'applicazione

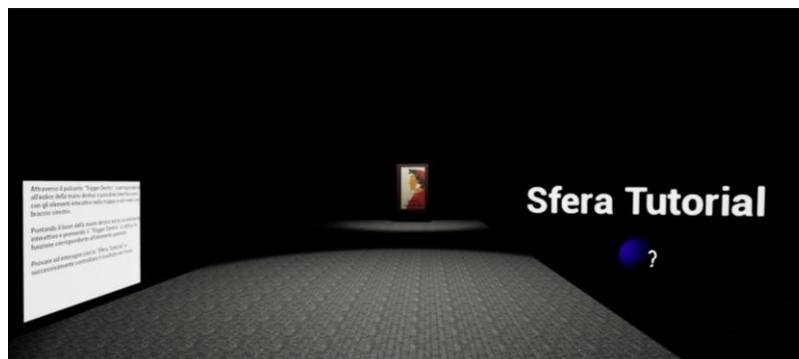
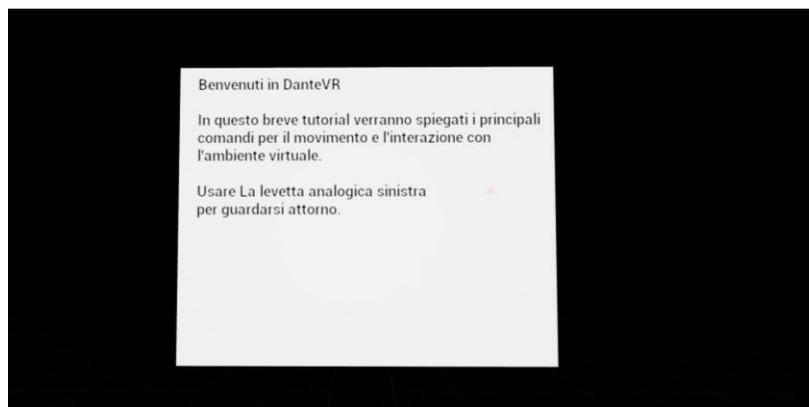


Figure: Screenshot della mappa tutorial.

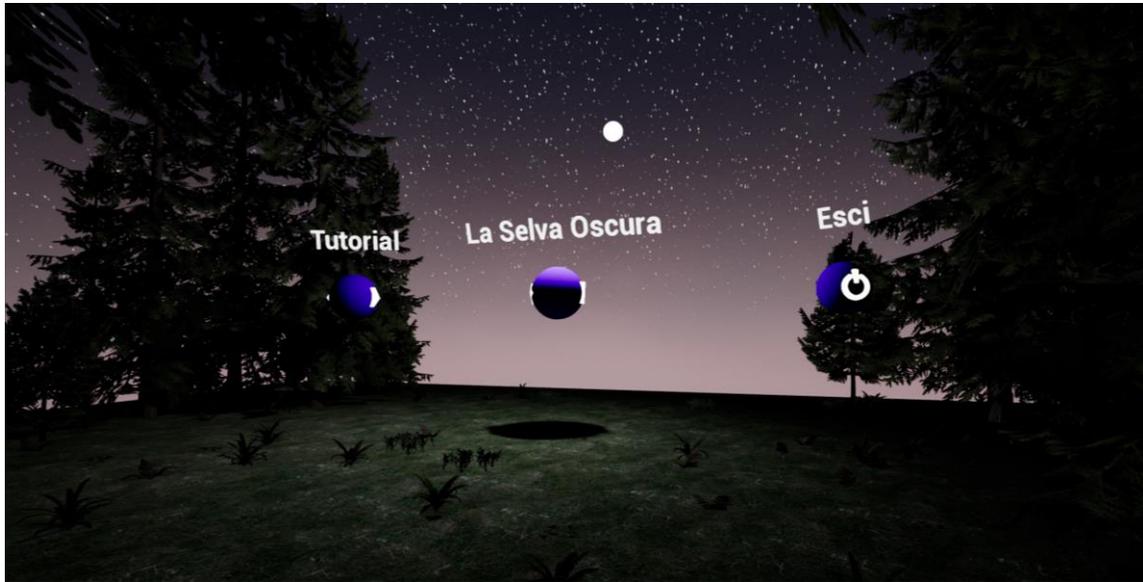


Figura: *Mappa0*.



Figura: *Mappa1*, Dante si ritrova a vagare per la Selva Oscura senza riuscire a ritrovare la retta via.



Figure: *Mappa2*, inizia a sorgere il sole e Dante si trova ai piedi del *Diletto Monte*; la vista della luce gli infonde coraggio e coglie l'occasione per sedersi a riposare guardando il pericoloso percorso che si è lasciato alle spalle.



Figure: *Mappa3*, le tre fiere che bloccano l'ascesa di dante alla cima del monte respingendolo dal terrore a tornare da dove è venuto, in successione: *una lonza, un leone, una lupa*; allegorie dei peccati capitali.

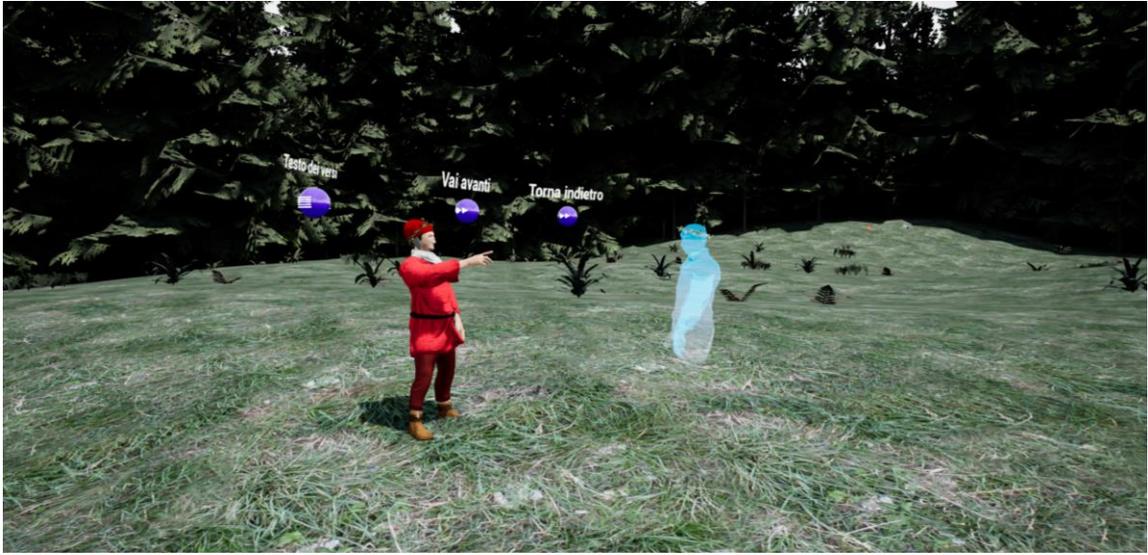


Figura: *Mappa4*, Dante riconosce Virgilio che gli narra del viaggio che dovranno affrontare

# Conclusioni

## Valutazioni esterne

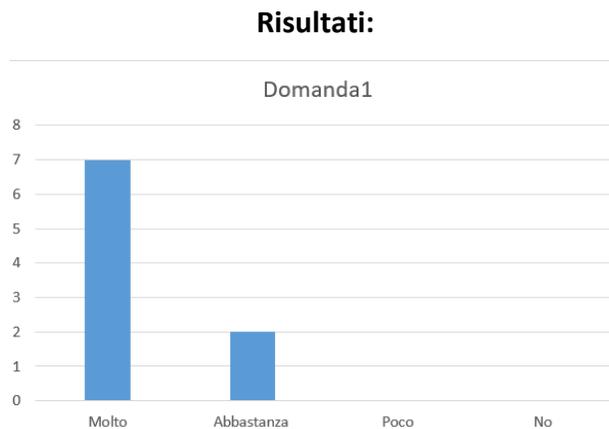
### Il questionario di valutazione e i risultati

Con l'intento di andare a comprendere l'efficacia ed i punti di criticità del progetto finito si è redatto un questionario di valutazione dell'esperienza che si è fatto compilare a dei volontari (un set di nove persone) che si sono prestati a provare l'applicazione; si presentano di seguito le domande con i risultati ottenuti.

### Valutazione sull'immersività e la presenza

1 Hai trovato l'esperienza coinvolgente?

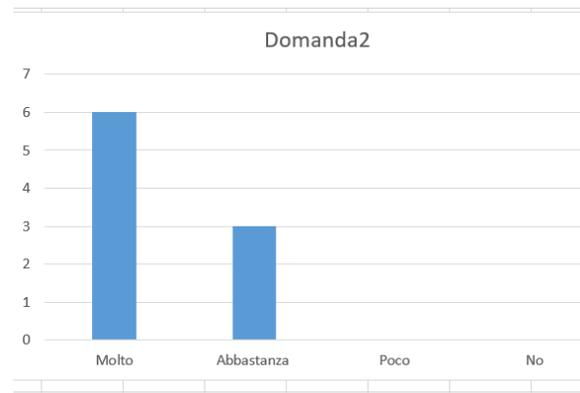
- Molto
- Abbastanza
- Poco
- No



2 Hai avuto l'impressione di essere effettivamente all'interno degli ambienti che hai visitato?

- Molto
- Abbastanza
- Poco
- No

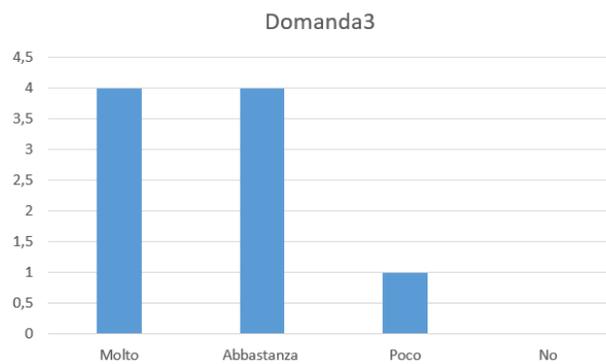
### Risultati:



3 Hai trovato corrispondenza tra le azioni e le scene con il contenuto audio e testuale del canto?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No

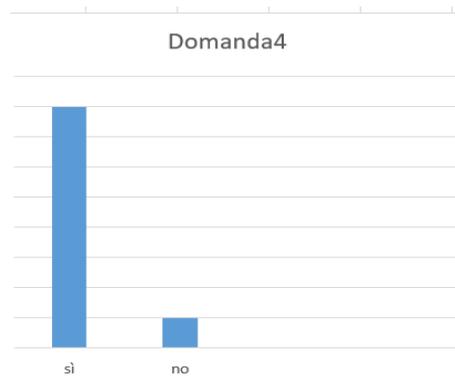
### Risultati:



4 Hai mai letto o ascoltato (a scuola, indipendentemente, a teatro o in tv) questo passo della *Divina Commedia*?

- a. Sì
- b. No

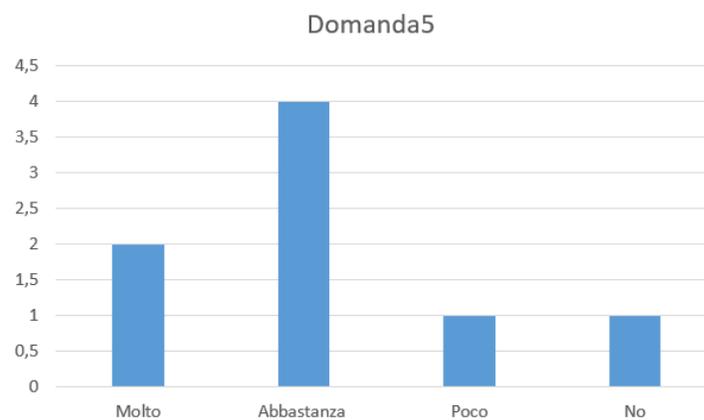
**Risultati:**



5 Se hai risposto positivamente alla domanda 4, le scene che hai visto rispecchiano l'idea o il ricordo che avevi della *Selva Oscura*?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No

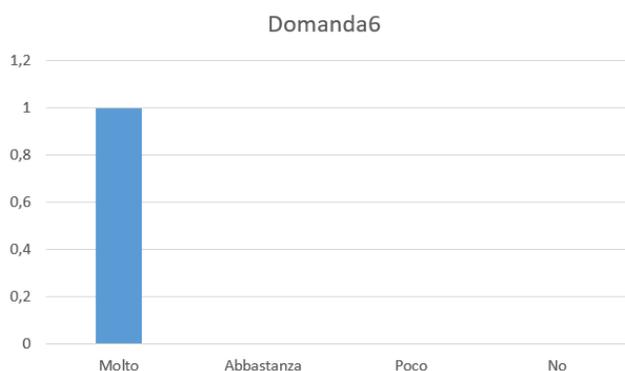
**Risultati:**



6 Se hai risposto negativamente alla domanda 4, ti interesserebbe l'idea di apprendere il contenuto della *Divina Commedia* in questa modalità?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No

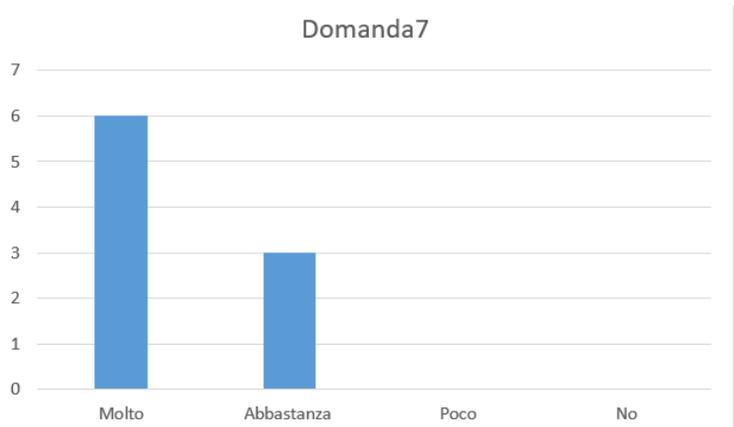
**Risultati:**



7 Saresti interessato a continuare l'esperienza per i canti successivi?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No

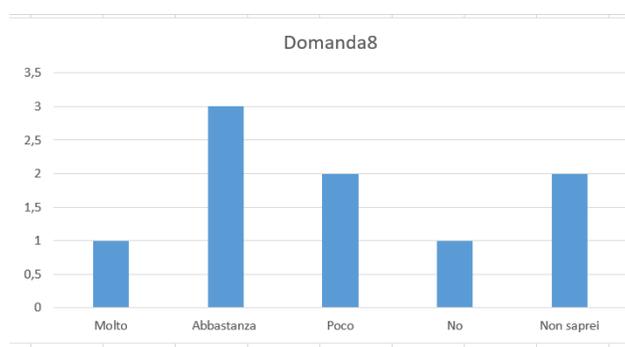
**Risultati:**



8 Pensi che il contenuto virtuale possa rischiare di far passare il contenuto letterario in secondo piano?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No
- e. Non saprei

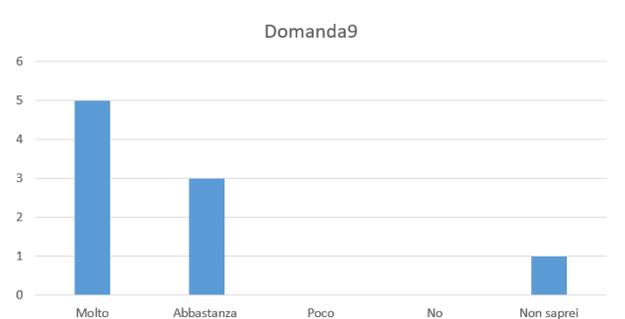
**Risultati:**



9 Pensi che questa modalità possa essere utile per scoprire o approfondire contenuti letterari?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No
- e. Non saprei

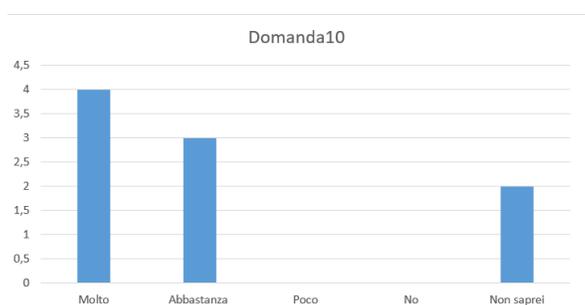
**Risultati:**



10 Pensi che questa modalità possa essere utile per accendere l'interesse per la materia nelle scuole?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No
- e. Non saprei

**Risultati:**

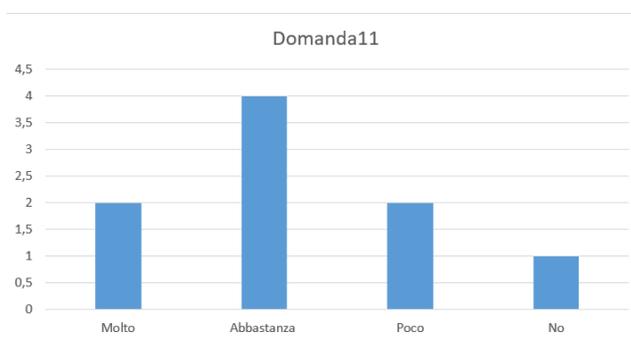


## Valutazione sull'usabilità

11 Hai trovato difficoltà a capire i comandi per interagire con l'ambiente virtuale?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No

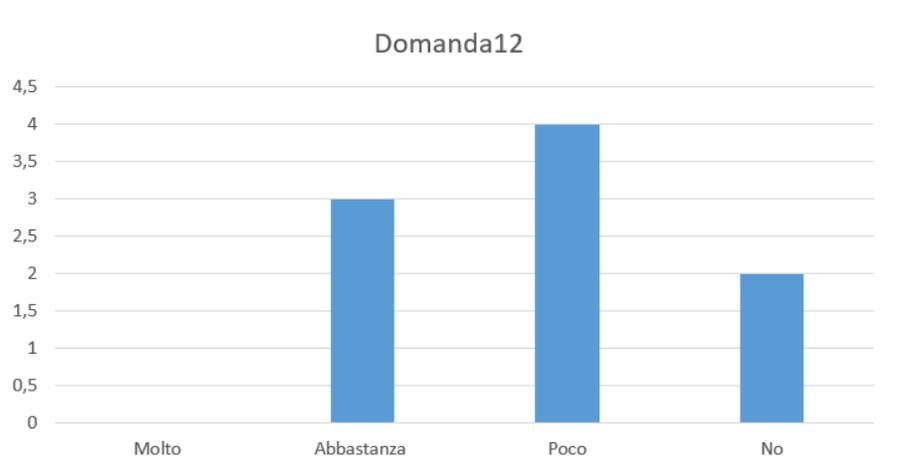
**Risultati:**



12 Hai avuto difficoltà a capire come interagire con i vari elementi?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No

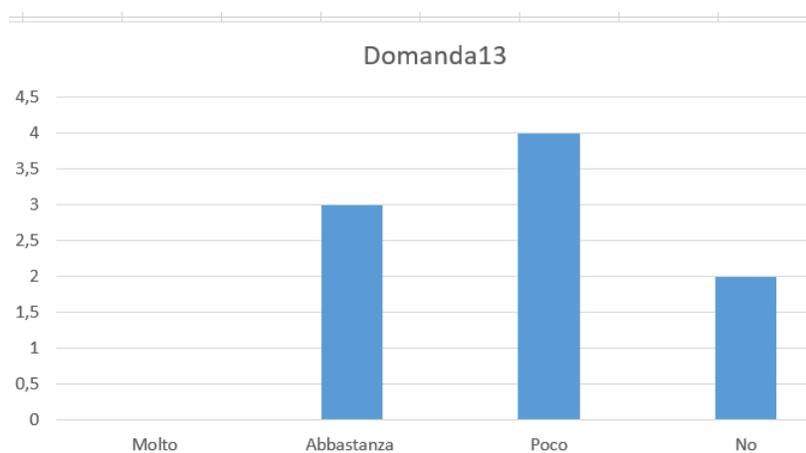
**Risultati:**



13 Hai avuto difficoltà ad usare il puntatore laser per selezionare le sfere?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No

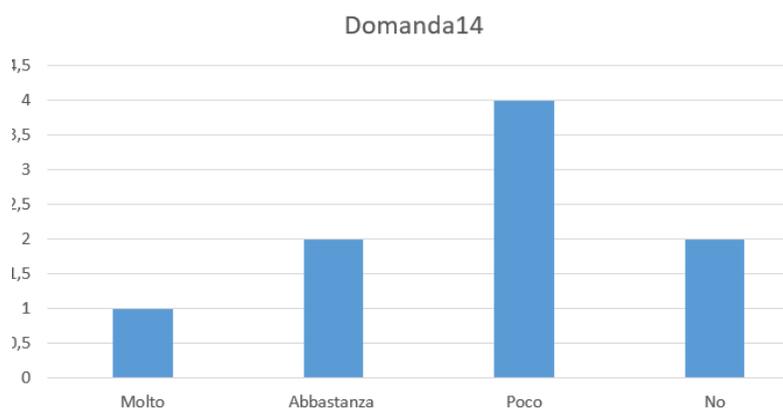
**Risultati:**



14 Reputi comodo dover girare per la mappa per cercare gli elementi interattivi?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No

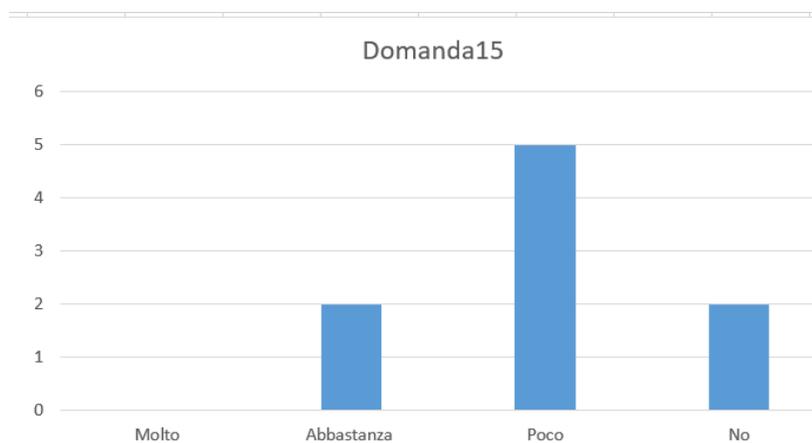
**Risultati:**



15 Reputi comodo il menù sul braccio per leggere i contenuti testuali?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No

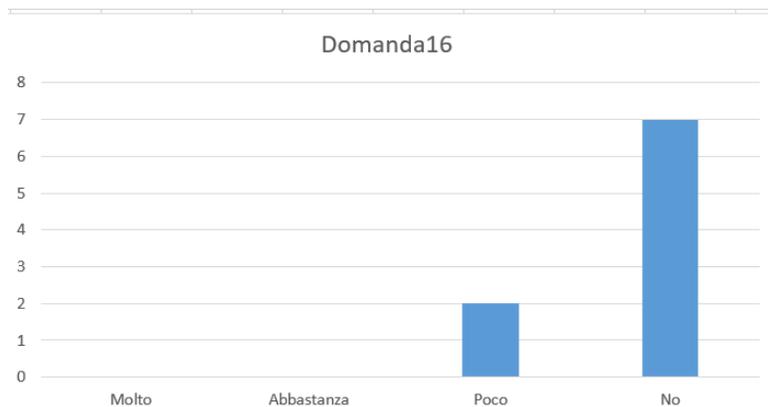
**Risultati:**



16 Hai mai avvertito un senso di confusione o nausea durante l'esperienza?

- a. Molto
- b. Abbastanza
- c. Poco
- d. No

**Risultati:**



**Riflessioni sulle valutazioni ricevute**

Il primo set di domande riguarda il senso di presenza e l'immersività suscitata dagli ambienti virtuali: generalmente tutti gli utenti hanno trovato le scene coinvolgenti e si sono sentiti immersi e presenti nelle situazioni vissute. Si è quindi riusciti nell'intento di sviluppare un contenuto immersivo efficace, con delle logiche di funzionamento coerenti e non estranianti che avrebbero altrimenti potuto infrangere la concentrazione dell'utente o sviarne l'attenzione rispetto il *focus* principale, quello letterario. I soggetti che hanno provato l'applicazione hanno altresì valutato positivamente la corrispondenza tra i versi della *Divina Commedia*, l'interpretazione audio di Vittorio Gassman e le azioni "recitate" dai personaggi. Anche il contesto attorno alla scena è risultato funzionale: si sono ricevuti *feedback* positivi riguardo le mappe, che sono generalmente risultate coerenti all'idea di *Selva Oscura* che l'utente aveva in mente, ed i suoni ambientali che hanno implementato ulteriormente l'immersività dell'esperienza.

Altre domande del questionario ruotano attorno alla posizione in cui vorrebbe porsi la *letteratura immersiva* nei confronti della materia puramente letteraria, cioè come suo strumento di diffusione. È

risultato che la maggior parte degli utenti conoscevano il canto e che gli ambienti predisposti rispecchiavano l'idea ed il ricordo che ne avevano dalla studio scolastico; mentre, chi non ne conosceva affatto il contenuto si è affermato propenso a voler conoscere la *Divina Commedia* attraverso questa modalità; la maggior parte dei soggetti che hanno utilizzato il software hanno mostrato interesse nell'idea di utilizzarlo per procedere ai canti successivi al fine di conoscerli o approfondirli. Dai questionari si evince inoltre che, chi ha provato l'esperienza, reputerebbe utile sfruttare il coinvolgimento vissuto mediante l'applicazione virtuale per scoprire o approfondire contenuti letterari o accendere l'interesse per la materia nelle scuole. Una valutazione interessante che è emersa nel far provare l'applicazione riguarda il concetto di *interpretazione* dell'esperienza nei confronti dell'opera originale: dai pareri ottenuti non risulta il timore che l'utilizzo di tecnologie virtuali possa far passare in secondo piano il contenuto letterario, ma qualcosa si viene effettivamente a perdere rispetto l'esperienza della lettura. Il candidato, nel modellare, posizionare e programmare le scene e gli eventi della *Selva Oscura* ha inevitabilmente costruito un'immagine del poema dettata dalla propria interpretazione; si è strutturato il progetto in base all'immagine mentale sorta dalla lettura della cantica e questa la si è portata a chi ha usufruito del software. Ciò che all'utente viene a mancare quindi, potendo attingere all'immaginario predisposto dall'applicazione, è tutto il processo di creazione di immagini mentali che la lettura, al contrario, impone per essere compresa ed apprezzata; si cita una delle persone che si è prestata a provare l'applicazione, rivolgendosi al candidato: «Questa è la *tua* Selva Oscura; ora però, ogni volta che penserò al primo canto, non posso fare a meno di pensare alle immagini che qui ho visto». Quella che emerge è una questione da analizzare attentamente e che sicuramente necessita di maggiori approfondimenti per valutare effettivamente il rischio e le possibili implicazioni dell'assenza di *interpretazione* da parte dell'utente; d'altra parte, però, si vogliono portare delle argomentazioni. Innanzitutto, questa imposizione delle immagini non è propria del medium virtuale, ma antecedente, andando a porre le sue radici nella televisione e nel cinema, se non addirittura nelle arti figurative ancora precedenti della pittura e della scultura: nel momento in cui viene fatta una trasposizione cinematografica o si compone un dipinto sulla *Divina Commedia* automaticamente si compie la stessa operazione in cui il linguaggio visivo del medium va a sostituire o modificare l'immaginario simbolico mentale scaturente dal testo scritto.



**Figura: *In my middle life*, Gustave Doré, 1861**

Inoltre, un'altra possibile argomentazione riguarda la volontà del lavoro qui portato che è quella di, non sostituire il libro, bensì diffonderne il contenuto; il poema dantesco, come molta della letteratura purtroppo, è un materiale di studio che, per quanto famoso, rimane di nicchia nella lettura effettiva del testo da parte degli individui. Poche sono le persone che possono affermare di aver letto integralmente ed autonomamente tutta la *Divina Commedia*: rimane un poema complesso a cui approcciarsi, scritto in volgare e con termini arcaici, pieno di significati allegorici da dover sciogliere, con contenuto elitario e riferimenti alla cultura greca e latina, rimandi biblici e ad autori o opere ecclesiastiche, citazioni di fatti storici ed a personaggi dell'epoca dantesca o precedenti; in definitiva, non esattamente un testo a cui è facile accedere. Ed è qui che si vorrebbe far intervenire la virtualità ed il suo linguaggio sincretico, proprio con il proposito di valorizzare gli insegnamenti ed i valori umani trasmessi facendo avvicinare quante più persone possibili all'opera; con la speranza che, perché no, si possa altresì accendere la passione ed indirizzare alla lettura dell'opera scritta.

Infine, tornando al questionario, rimane da analizzare la valutazione sull'usabilità. In generale gli utenti non hanno avuto difficoltà a comprendere l'utilizzo dei vari elementi ed a interagirvi, inoltre non vi sono stati problemi legati al *frame drop* ed alla *motion sickness*; ma comunque, in questo frangente di valutazione sono emerse delle criticità. Uno degli aspetti ritenuti più problematici è senza dubbio il *tutorial* che sembra non riuscire a soddisfare il suo scopo primario: far comprendere al giocatore come interagire con il mondo virtuale. In seguito ai *feedback* si sono pensate alcune modifiche che potrebbero essere utili a migliorare tale frangente ad esempio: aggiungere una numerazione progressiva dei pannelli da seguire per completare con successo la mappa iniziale,

inserire nelle descrizioni, per ora solo scritte, descrizioni visive per rendere più immediata la comprensione delle istruzioni, aggiungere informazioni riguardo il funzionamento delle varie sfere e, in generale, dare indicazioni più precise e sintetiche. Per quel che riguarda il progetto nel suo complesso, si potrebbe pensare ad una disposizione alternativa per i contenuti testuali: diversi utenti hanno lamentato la stanchezza del braccio sinistro (dove compare il *widget* del *VrPawn* necessario per seguire il testo dei versi) successivamente all'utilizzo dell'applicazione. Anche il dover spostarsi per la mappa al fine di cercare i vari elementi interattivi è stato giudicato poco utile ed anzi distraente di quella che dovrebbe essere un'esperienza finalizzata a seguire le azioni, la trama ed i personaggi. Per risolvere tali difficoltà bisognerebbe pensare ad un'ottimizzazione dell'informazione nell'interfaccia; si potrebbe ideare, al posto del menù nel braccio, un volume sferico da poter rendere visibile o invisibile alla pressione di un tasto, che circonda il giocatore seguendone gli spostamenti e dove render disponibili tutti i contenuti testuali ed informativi. Inoltre, potrebbe risultare utile aggiungere una modalità *follow* per la telecamera del giocatore in modo che possa spostarsi la visuale comodamente sul protagonista e non togliere il *focus* dall'azione principale.

### *Possibili sviluppi e considerazioni finali*

*Ai giovani d'oggi non è permesso accostarsi al tradizionale patrimonio del genere umano attraverso la porta della consapevolezza tecnologica. Questa porta, l'unica possibile, viene loro sbattuta in faccia da una società che osserva il mondo in uno specchio retrovisore.*

*[...] Lo studente non trova i mezzi che lo coinvolgono e non riesce a scoprire come il sistema educativo possa essere messo in relazione al mondo mitico di dati ed esperienze elettroniche che percepisce in modo chiaro ed immediato.*

Marshall McLuhan,

*Il medium è il messaggio*

I possibili sviluppi dell'applicazione risultano molteplici. Innanzitutto, per quel che riguarda animazioni e modelli si possono fare miglioramenti: i software utilizzati, in particolar modo *Fuse* e *Mixamo*

lavorano con *preset* ed interfacce non completamente libere; si può pensare ad un rimodellamento complessivo dei personaggi e delle creature in modo che più si adattino alle forme convenzionali con cui vengono storicamente ed artisticamente rappresentati: creare delle tuniche, con delle calzature più adatte per l'epoca e colori più convincenti nei limiti di quello che, comunque, un dispositivo hardware per il VR può consentire a livello di peso di calcolo. Anche le animazioni possono essere ampliate e migliorate attraverso una sessione di recitazione di *motion capture* che permetta di creare, attraverso attori reali, le animazioni che servono in maniera più libera rispetto ai vincoli posti dal limitato numero di animazioni messe a disposizione dal software Mixamo; in aggiunta si potrebbero anche implementare animazioni per le tre fiere, presenti ora solo come modelli statici. Inoltre, si potrebbe anche pensare ad un rifacimento delle mappe e degli ambienti in modo che possano risultare più evocativi, un lavoro artistico che potrebbe attingere dalle atmosfere e suggestioni di spunti pittorici o di miniature su manoscritti. La programmazione in UE4 potrebbe essere migliorata per velocizzare ed ottimizzare i processi di calcolo di *rendering* permettendo quindi maggiore fluidità o un maggior numero di poligoni a schermo utili a dare maggior contesto e realismo all'esperienza. Infine, si potrebbe pensare ad una traduzione in altre lingue dei contenuti testuali per poter aumentare il potenziale di diffusione del software e, quindi, rendere il contenuto di *Selva Oscura VR* utilizzabile anche all'estero.

Per quel che riguarda il contenuto, le possibili implementazioni possono seguire due direttive: da una parte si può affrontare il discorso letterario e filologico andando a studiare ed analizzare in modo più specifico ed approfondito quello che riguarda l'informazione storica ed esegetica da inserire nelle sfere informative, uno studio su cosa può essere utile inserire per la comprensione dell'opera e sulla veridicità scientifica dei contenuti. Dall'altra parte un'implementazione contenutistica più verticale, banalmente, aumentando il numero di canti virtualizzati con tutte le migliorie e le specifiche fin qui suggerite. Si è ragionato sulla possibilità di aggiungere una qualche *feature* "sfidante" che avvicini l'esperienza al *gaming*, attraverso collezionabili e punteggi per lo *scoring*, ma l'idea è stata poi accantonata nel timore che avrebbe potuto snaturare la volontà iniziale di rendere un'esperienza che si era proposta di simulare la "lettura di un libro". Infine, per lo sviluppo della *letteratura immersiva*, si potrebbe ovviamente pensare di espandere l'idea della simulazione qui eseguita sulla *Divina Commedia* anche per altre opere letterarie, per poter da una parte arricchire il patrimonio culturale virtuale, e dall'altra dare un nuovo respiro ed una nuova *audience* agli universi letterari creati nel

corso della storia umana; permettendo, si immagini, di assistere: all'assalto di Troia da dentro il cavallo di legno, all'incontro di Ulisse con Polifemo, al viaggio di Astolfo sulla luna alla ricerca del senno di Orlando, alle cariche del prode Don Chisciotte portate contro i mulini a vento; e perché non anche il teatro o la poesia, pensando di spettare in prima persona alle tragedie di Shakespeare, alle notti insonni di Ungaretti in trincea o immaginando di sedersi accanto a Leopardi sul Colle dell'Infinito.

In conclusione, questo è stato il proposito, sempre rimasto saldo, con cui ci si è avvicinati in questo lavoro: portare un'idea che prescindendo dalla sua realizzazione che rimane, per quanto appassionata, pur sempre amatoriale; un'idea che possa far scorgere agli *outsider* del mondo della virtualità le potenzialità e le applicazioni che esso può offrire se canalizzato efficacemente. Un mezzo non solo per trasmettere ma, più semplicemente, per comunicare con le nuove generazioni digitali che tanto sembrano estranee alle modalità di pensiero e di azione rispetto ai loro predecessori. In altre parole, chiedendo nuovamente aiuto al passato, si vuole citare un aforisma estratto dai *Saggi* di Francis Bacon:

*“Se la montagna non è venuta a Maometto,*

*Maometto andrà alla montagna”*

Questa è l'ottica che si propone per la diffusione della letteratura al fine di non rischiare di estrometterla dalla coscienza e dalla vita dell'uomo contemporaneo che, nonostante il cambio di linguaggio, invece necessita di lei, della sua guida, dei suoi insegnamenti ed immaginari, nonché della sua pura e semplice bellezza.

## Bibliografia e sitografia

- Adobe. 2014. *Adobe Fuse CC*. 7 agosto. Consultato il giorno 28 12, 2021. <https://store.steampowered.com/app/257400/Fuse/>.
- . 2020. *Adobe Illustrator*. Consultato il giorno 12 29, 2021. [https://it.wikipedia.org/wiki/Adobe\\_Illustrator](https://it.wikipedia.org/wiki/Adobe_Illustrator).
- Alighieri, Dante. 1968. *La Divina Commedia - Inferno*. Firenze: La Nuova Italia Editrice.
- Bauman, Zygmunt. 2011. *Modernità liquida*. Bari: Laterza.
- Bergenson, Brian. 2006. *Developing serious games*. Boston: Charles River Media.
- Blender Foundation. 2021. *Blender*. 3 dicembre. Consultato il giorno 12 29, 2021. <https://www.blender.org/>.
- Bolter, Jay David, e Richard Grusin. 2000. *Remediation: Understanding New Media*. Cambridge: MIT Press.
- BricksLab. s.d. *LACOM3DIA*. Consultato il giorno 12 28, 2021. <https://lacom3dia.it/>.
- Carraro Lab . 2010. *IDante - La Divina Commedia*. Consultato il giorno 12 28, 2021. <https://www.carraro-lab.com/portfolio-item/idante-la-divina-commedia/>.
- Castells, Manuel. 2002. *La nascita della società di rete*. Milano: Università Bocconi Editore.
- Cockos Incorporated. 2021. *Reaper*. 21 12. Consultato il giorno 12 28, 2021. <https://www.reaper.fm/>.
- Coppola, Andrea. 2019. *Blender 2.8 - La guida definitiva*. Vol. da I a IV. Roma: Blender High School.
- Deterding, Sebastian, Rilla Khaled, Lennart Nacke, e Dan Dixon. 2001. «Gamification: Toward a definition.» *Gamification Workshop Proceedings*. Vancouver. [https://www.researchgate.net/publication/273947177\\_Gamification\\_Toward\\_a\\_definition](https://www.researchgate.net/publication/273947177_Gamification_Toward_a_definition).
- Digital Strategy & Design. s.d. *La Divina Commedia VR*. Consultato il giorno 12 28, 2021. <https://ettsolutions.com/newmedia/progetti/la-divina-commedia-vr/>.
- Eco, Umberto. 2016. *Apocatilliti e Integrati*. Milano: Bompiani.

- Epic Games. 2021. *Unreal Engine 4*. 19 8. Consultato il giorno 12 29, 2021. [https://it.wikipedia.org/wiki/Unreal\\_Engine](https://it.wikipedia.org/wiki/Unreal_Engine).
- Facebook Technologies. 2020. *Meta Quest*. 13 ottobre. Consultato il giorno 12 2021, 28. [https://www.oculus.com/quest-2/?locale=it\\_IT&utm\\_source=gg&utm\\_medium=ps&utm\\_campaign=11140447195&utm\\_term=oculus%20quest%202&utm\\_content=562099288267&utm\\_ad=109261218156&utm\\_location=1008044&utm\\_location2&utm\\_placement=kwd-563298318914&utm\\_device=c&utm\\_m](https://www.oculus.com/quest-2/?locale=it_IT&utm_source=gg&utm_medium=ps&utm_campaign=11140447195&utm_term=oculus%20quest%202&utm_content=562099288267&utm_ad=109261218156&utm_location=1008044&utm_location2&utm_placement=kwd-563298318914&utm_device=c&utm_m).
- Ferri, Paolo. 2011. *Nativi Digitali*. Milano: Pearson.
- Huizinga, Johan. 2002. *Homo ludens*. Torino: Giulio Einaudi editore.
- Mack, Kevin, e Robert Ruud. 2019. *Unreal Engine 4 - Virtual Reality projects*. Birmingham: Packt Publishing Ltd.
- McCaffrey, Mitch. 2017. *Unreal Engine VR - Cookbook*. Boston: Addison - Wesley.
- McLuhan, Marshall. 2019. *Il medium è il messaggio*. Mantova: Corraini Edizioni.
- Michael, David, e Sande Chen. 2006. *Serious games: games that educate, train and inform*. Boston: Thompson.
- Mixamo Inc. 2021. *Mixamo*. Consultato il giorno 12 29, 2021. <https://www.mixamo.com/>.
- Mulargia, Simone. 2016. *Videogiochi Effetti (sociali) speciali*. Milano: Edizioni Angelo Guerini.
- PagliaQuotidiana. 2012. *Vittorio Gassman legge Dante - Commedia - Inferno, Canto I*. 2 febbraio. Consultato il giorno 12 28, 2021. <https://www.youtube.com/watch?v=aBGq11ODudA>.
- Pecchinenda, Gianfranco. 2010. *Videogiochi e cultura della simulazione - La nascita dell' "homo game"*. Bari: Editori Laterza.
- percroart. 2013. «MUBIL - Adam Lonicer's Kreuterbuch "Augmented book".» *Youtube*. 12 11. Consultato il giorno 1 18, 2022. <https://www.youtube.com/watch?v=va8-phTZGjw>.

- Pipitone, Daniele. 2013. *Archeomatica*. 15 ottobre. Consultato il giorno 12 28, 2021. <https://www.archeomatica.it/ict-beni-culturali/mubil-nuove-tecnologie-per-la-fruizione-3d-di-libri-antichi>.
- Prensky, Marc. 2006. *Don't bother me mom. I'm learning!* St. Paul: Paragon House.
- . 2001. «Digital Natives, Digital Immigrants.» *MCB University Press*, October: 1-6. <https://www.marcprensky.com/writing/Prensky%20-%20Digital%20Natives,%20Digital%20Immigrants%20-%20Part1.pdf>.
- Riva, Giuseppe. 2019. *Nativi Digitali - Crescere ed apprendere nel mondo dei nuovi media*. Bologna: il Mulino.
- . 2008. *Psicologia dei nuovi media*. Bologna: Il Mulino.
- Riva, Giuseppe, e Andrea Gaggioli. 2019. *Realtà Virtuali*. Firenze: Giunti.
- Rosselli Del Turco , Roberto. 2021. *Vercelli Book Digitale*. Consultato il giorno 12 28, 2021. <http://vbd.humnet.unipi.it/>.
- Sartori, Giovanni. 2019. *Homo videns*. Bari: Laterza.
- Sauvé, Louise, Lise Renaud, e David Kaufman and Jean-Simon Marquis. 2007. «Distinguishing between games and simulations: A systematic review.» *Jstor* 247-256. <https://www.jstor.org/stable/jeductechsoci.10.3.247>.
- Stella, Renato, Claudio Riva, e Michela Drusian Cosimo Marco Scarcelli. 2018. *Sociologia dei new media*. Novara: UTET Università.
- Sterling, Jim. 2008. *Dante's Inferno on the Commodore 64*. Consultato il giorno 12 28, 2021. <https://www.destructoid.com/dantes-inferno-on-the-commodore-64/>.
- Tapscott, Don. 2011. *Net Generation. Come la generazione digitale sta cambiando il mondo*. Milano: Angeli.
- The Visual Agency. 2017. *Codex Atlanticus*. Consultato il giorno 12 28, 2021. <https://www.codex-atlanticus.it>.

- Torch. 2019. «AR App Ideas: Enriching an Art Book.» *Youtube*. 29 06. Consultato il giorno 01 18, 2022. <https://www.youtube.com/watch?v=G7ZzMX771Ug>.
- Triberti, Stefano, e Luca Argenton. 2019. *Psicologia dei videogiochi - Come i mondi virtuali influenzano mente e comportamento*. Santarcangelo di Romagna: Maggioli Editore.
- Ulibarri, Stephen Seth. 2020. *Unreal Engine C++ the Ultimate Developer's Handbook*. Torino: Amazon Italia Logistica.
- Valve Corporation. 2021. *Steam*. Consultato il giorno 12 28, 2021. <https://store.steampowered.com>.