

Progetto Realtà Virtuale

Progetto Realtà Virtuale	1
Spiegazione progetto	1
Comandi base	2
Packages installati	2
Versione del progetto e dove trovarlo	2
Scena iniziale	2
Descrizione Scena	2
Lista oggetti manipolabili	5
Lista script utilizzati nella scena	7
Descrizione asset utilizzati e modifiche ai prefab	7
Scena Simulazione	9
Descrizione Scena	9
Lista oggetti manipolabili	11
Lista script utilizzati nella scena	14
Descrizione asset utilizzati	14
Prefab	15
Scena Navicella	15
Descrizione scena	15
Lista oggetti manipolabili	18
Lista script utilizzati nella scena	21
Descrizione asset utilizzati	22
Prefab	22
Sviluppi futuri	22

Questo progetto è stato realizzato da tre studenti del CdLM Teoria e Tecnologia della Comunicazione:

- Matteo Ilardo 873723
- Michela Mariani 866056
- Andrea Mazzilli 829700

SPIEGAZIONE PROGETTO

Il progetto è ambientato nello spazio, in particolare ci sono tre scene:

- 1) La prima scena è ambientata su un pianeta, lo scopo dell'utente sarà capire come raggiungere la navicella spaziale;
- 2) La seconda scena è ambientata dentro la navicella, in particolare l'utente, per passare alla scena successiva e quindi essere pronto per far partire la navicella, dovrà svolgere una simulazione;
- 3) La terza scena è ambientata sempre dentro la navicella, l'utente a questo punto dovrà capire come farla partire.

Una volta partita la navicella il gioco terminerà.

Lo scopo dell'applicazione è quindi quello di offrire all'utente l'esperienza di un paesaggio spaziale, accompagnato da delle piccole sfide per passare da una scena all'altra.

Per attuare il progetto abbiamo deciso di suddividerci le scene realizzandone una a testa. Per poter lavorare in contemporanea alle nostre stanze abbiamo creato un repository su Git Hub, in questo modo abbiamo potuto lavorare in maniera indipendente e allo stesso tempo sfruttare gli stessi asset e script senza crearne di nuovi.

Non avendo a nostra disposizione un Oculus o strumenti simili, abbiamo sviluppato il gioco in modo che possa essere disponibile e testabile direttamente dal computer.

Comandi base

- Per muoversi all'interno della scena basta utilizzare WASD e il mouse
- Per far apparire il raggio nella mano quando non presente bisogna prima selezionare la mano (Y) e poi premere il primary button
- Per far apparire e sparire la GUI presente in ogni scena che riassume i comandi principali bisogna premere il tasto "I" (i maiuscolo)
- Per ricaricare la scena bisogna premere il tasto "L"

Packages installati

- AI Navigation
- OpenXR Plugin
- TextMeshPro
- XR Interaction Toolkit (con Samples Starter Asset)
- XR Plugin Management

Versione del progetto e dove trovarlo

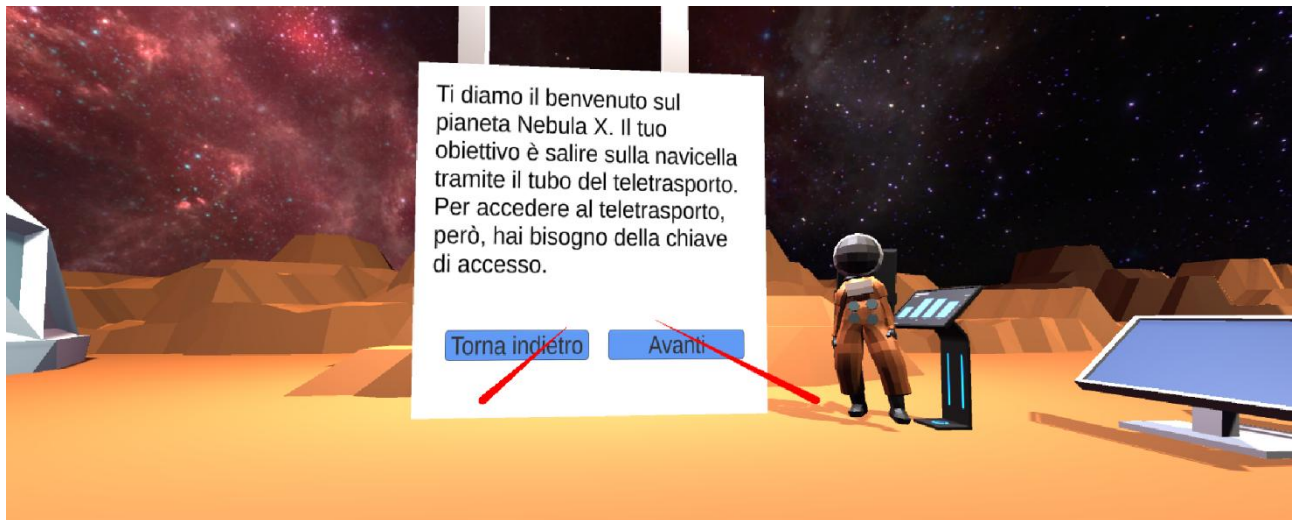
Il progetto si trova all'interno di questa [cartella](#) su OneDrive. Utilizza la versione di Unity 2022.3.5f1 compatibile anche con il progetto utilizzato per le esercitazioni (*almeno, era questa la versione di Unity suggerita nel momento in cui abbiamo scaricato quel progetto*).

SCENA INIZIALE

Descrizione Scena

La prima scena è ambientata su un pianeta tipo Marte chiamato *Nebula X*, in cui sono presenti diversi astronauti che si sono stabiliti sul pianeta, installando pannelli solari, edifici, torre di controllo, veicoli etc.

La scena inizia davanti a un astronauta che sta controllando su uno schermo alcuni parametri legati ai pannelli solari. L'utente vede prima una GUI che spiega quali sono i comandi disponibili per muoversi e giocare e poi una GUI che spiega qual è l'obiettivo della scena corrente, ovvero entrare nella navicella e quindi nella stanza successiva tramite il teletrasporto.



All'interno delle GUI sono presenti i seguenti testi, volti a spiegare come l'utente deve comportarsi nella scena:

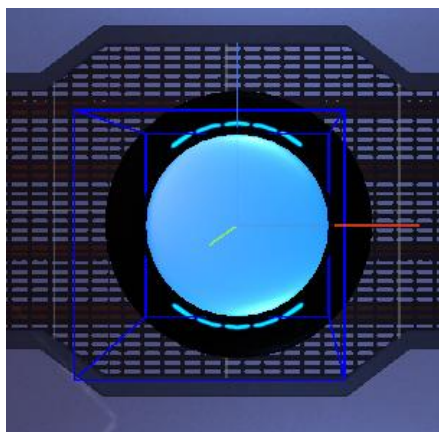
- 1) “Premi il tasto “P” o il bottone qui sotto per interagire”
- 2) “Ti diamo il benvenuto sul pianeta Nebula X. Il tuo obiettivo è salire sulla navicella tramite il tubo del teletrasporto. Per accedere al teletrasporto, però, hai bisogno della chiave di accesso.”
- 3) “L'astronauta Alpha 1 ti guiderà in questo processo. Segui la macchina, ti porterà da Alpha 1. Premi il tasto “C” o il bottone qui sotto “Partiamo” per iniziare”

L'utente una volta letta la prima GUI può passare a quella successiva premendo il bottone "Avanti". Il bottone “Avanti”, così come il pulsante della tastiera *P* disattiva l'attuale GUI, fa girare l'astronauta verso la Main Camera tramite lo script “*Dialogo_Astronauta*” e setta la variabile booleana “Dialogo” nel valore *true*. Questa variabile è un parametro dell'Animator che gestisce il GameObject dell'astronauta dei pannelli solari e che, quando è vera, cambia l'animazione facendo sì che l'astronauta faccia un inchino e poi sia in una posizione di *neutral idle*.

Una volta premuto “Avanti” si può tornare alla GUI precedente tramite il pulsante “Torna indietro”.

Il pulsante “Partiamo”, così come il tasto *C*, riporta la GUI alla prima schermata (“Premi il tasto...”) nel caso l'utente volesse rileggere lo scopo della scena attuale e attiva la GUI che si trova vicino ai due astronauti vicini agli edifici, ovvero *Alpha 1* (nella gerarchia di Unity chiamato “Astronauta edificio 1”) e *X Bot* (nella gerarchia di Unity chiamato “Astronauta edificio 2”).

Se si segue la direzione della macchina (che si muove tramite una semplice animazione di tipo *Transform Position*) si trovano i due astronauti. Per arrivare ai due astronauti si può scegliere se muoversi in maniera naturale, oppure sfruttare i Teleport.



Quando l'utente arriva davanti ad Alpha 1 e X Bot essi stanno parlando e l'utente si trova davanti un'altra GUI con due spiegazioni, ovvero:

1. "Premi il tasto "O" o clicca sul pulsante qui sotto per interagire"
2. "Ciao, ti mostrerò come entrare nella serra. Ti basterà seguirmi. Premi il tasto "C" o il bottone qui sotto quando vuoi partire".



Se si preme "C" o il bottone della seconda GUI (ovvero "Partiamo") X Bot si siede sulla panchina vicino a cui era in piedi fino a quel momento, mentre Alpha 1 inizia a camminare verso il primo obiettivo. Infatti, premendo "C" o il bottone "Partiamo" viene settato:

- Per X Bot il parametro "C_Pressed" del suo Animator che cambia la sua animazione da "Neutral Idle" a "Male sitting pose" (tramite lo script "X_Bot")
- Per Alpha 1 un parametro omonimo che cambia la sua animazione da "Neutral Idle" a "Standard Walk" (tramite lo script "Cammino_Astronauta")

Lo script "Cammino_Astronauta" è lo script principale della scena e fa sì che Alpha 1 cammini in giro per la mappa fino all'obiettivo attuale, indichi all'utente l'oggetto con cui deve interagire e faccia apparire una GUI che spiega all'utente come deve interagire con l'oggetto di volta in volta.

Lo script lo fa utilizzando vari metodi e delle variabili array, ovvero destinazione, oggettoObiettivo e attualeGUI, con delle variabili private con contano i numeri che variano all'interno delle array.

Ciò che l'utente deve fare per arrivare al teletrasporto con cui può cambiare stanza è prendere la chiave blu della serra (che possiede un componente XR Grab Interactable), posizionare la chiave blu della serra

sul socket del tavolo della serra per aprire la porta della serra, prendere la chiave gialla del teletrasporto (che possiede un componente XR Grab Interactable) all'interno della serra, posizionare la chiave gialla del teletrasporto sul socket del tavolo del teletrasporto e, infine, puntare verso il teletrasporto e premere "G" per andare alla scena successiva.

Prima di ognuna di queste azioni compare una GUI con le istruzioni, che scompare ad azione compiuta. Quindi quando Alpha 1 arriva davanti alla chiave blu della serra compare la relativa GUI, che scompare quando l'utente prende in mano la chiave blu della serra. Allo stesso modo, quando l'azione viene compiuta (in questo caso viene raccolta la chiave blu) viene resettata come *true* la variabile booleana "Cambia" che gestisce l'azione principale dell'astronauta.



Le chiavi e i tavoli sono impostati su layer diversi, in modo che sul tavolo della serra possano andarci solamente gli oggetti del layer "ChiaveBlu" (ovvero solo la chiave blu) e sul tavolo del teletrasporto solo gli oggetti del layer "Teletrasporto" (ovvero solo la chiave gialla).



Lista oggetti manipolabili

Astronauti

Nella scena è possibile interagire con tre astronauti: il primo (quello dei pannelli solari), Alpha 1 e X Bot. L'interazione con tutti e tre è gestita sia da animator che dai relativi script, ovvero *Dialogo_Astronauta*, *Cammino_Astronauta* e *X_Bot*.

Ognuno di questi possiede delle variabili sia interne che comuni con l'animator che fa sì che, quando l'utente fa qualcosa – per esempio preme “C” davanti ad Alpha 1 –, lo script e l'animator agiscano contemporaneamente (nell'esempio, Alpha 1 inizia a spostarsi verso la destinazione per via dello script e mentre lo fa possiede un'animazione di una camminata).

Chiavi e tavoli

Le chiavi della serra e del teletrasporto sono GameObject che possiedono delle componenti XR Grab Interactable che fa sì che esse possano essere prese in mano.

Il tavolo della serra e del teletrasporto possiedono degli empty children che fungono da socket, in modo che le chiavi blu e gialla possano essere posizionate sui tavoli corrispondenti per aprire le rispettive porte – ovvero le porte della serra e la porta che preclude l'accesso al teletrasporto.

Sia nel caso della funzione “Select” delle chiavi sia nel caso delle funzioni “Hover Entered” e “Hover Exited”, così come “Select Entered” e “Select Exited” vengono cambiati dei parametri dello script “Cammino_Astronauta” in modo che l'astronauta si diriga verso la destinazione successiva al termine dell'azione oppure che le porte si aprano (e i relativi NavMeshObstacles disattivati).

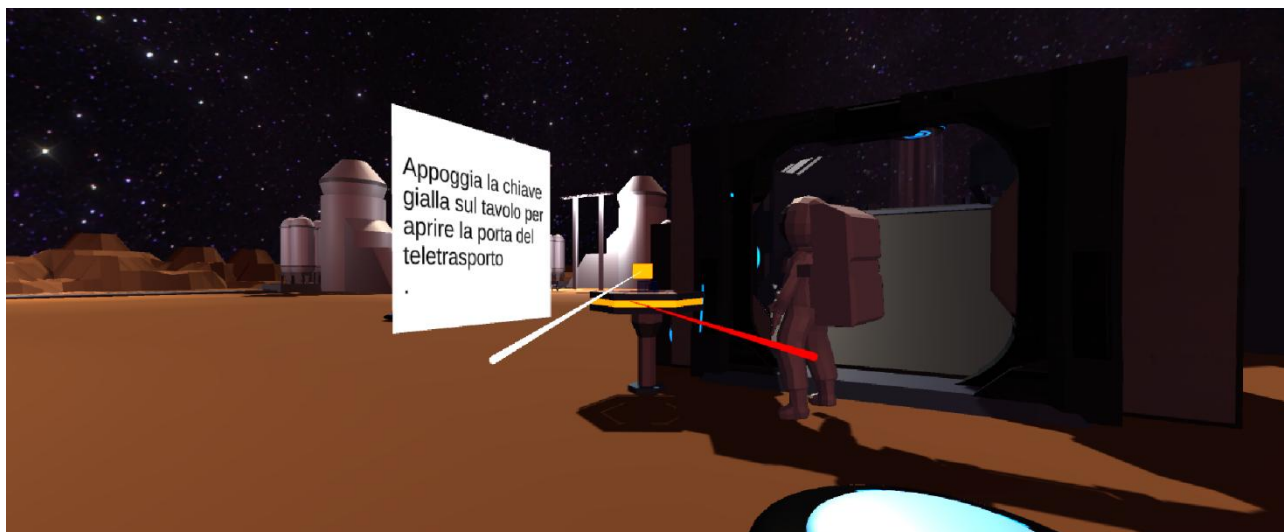
Teleport

Sono i punti di teletrasporto, permettono di muoversi più velocemente nella stanza. L'utente può interagirci grazie alla presenza del Locomotion System e Teleportation Provider nell'XR Rig e grazie alla presenza del Teleportation Anchor all'interno del prefab del teleport.

Teleport speciale

Permette all'utente di passare da una stanza all'altra. L'utente può interagirci grazie alla presenza dell'XR Simple Interactable a cui è associato un Interactable Event nella sezione “Activate” che presenta come oggetto il teleport speciale stesso in quanto lo script “RoomTransition” si trova direttamente sull'oggetto.

Il componente dello script è di default disattivato, viene attivato nella scena iniziale quando l'utente appoggia la chiave gialla del teletrasporto sul relativo tavolo.





Lista script utilizzati nella scena

- **Cammino_Astronauta:** utilizzato per gestire il movimento dell'Astronauta Alpha 1 e tutti gli oggetti relativi, come le GUI che spiegano come interagire con gli oggetti
- **apertura_porta:** utilizzato per cambiare il parametro *character_nearby* dell'Animator associato alle porte della serra e del teletrasporto, in modo che si aprano quando le chiavi della serra e del teletrasporto sono appoggiate sui socket dei rispettivi tavoli
- **Dialogo_Astronauta:** utilizzato per gestire le interazioni con l'astronauta dei pannelli solari, gestisce anche i parametri del relativo Animator in modo che cambi l'animazione dell'astronauta mentre l'utente ci interagisce
- **navmesh_disable:** utilizzato per disabilitare il componente di NavMeshObstacle associato alle porte della serra e del teletrasporto in modo che l'astronauta Alpha 1 possa passarci attraverso. In questo modo quando le porte sono chiuse egli non può passarci attraverso
- **X_Bot:** utilizzato per gestire le interazioni con l'astronauta X Bot (ovvero l'astronauta che parla con Alpha 1 vicino agli edifici), gestisce anche i parametri del relativo Animator in modo che cambi l'animazione dell'astronauta mentre l'utente ci interagisce e si sieda quando l'utente preme "C" per far iniziare il movimento di Alpha 1
- **RoomTransition:** utilizzato per cambiare stanza.

Alcuni metodi utilizzati in questi script sono simili a quelli visti a lezione: per esempio cambiano il valore di un parametro oppure lo impostano solamente come *true* o come *false*. I metodi sono poi associati a XR Grab Interactable oppure XR Socket Interactor.

Ci sono poi anche gli script *"moveMainCamera"* e *"WindVolumeController"* che però non funzionano parzialmente o del tutto (vengono spiegati più dettagliatamente nella sezione degli sviluppi futuri [più sotto](#)).

Descrizione asset utilizzati e modifiche ai prefab

Assets

La maggior parte degli asset presenti in questa scena è stata reperita direttamente da Unity Asset Store. In particolare, sono stati utilizzati all'interno della scena i seguenti asset:

- **Alstra Infinite** per le casse posizionate vicino alla chiave della serra <https://assetstore.unity.com/packages/3d/props/chests-polypack-209167>
- **Flex Unit** per gli UFO che si muovono nello spazio intorno alla scena <https://assetstore.unity.com/packages/3d/vehicles/space/ufo-battleship-289193>
- **Megalopoly.Art Vintage Control** per le chiavi della serra <https://assetstore.unity.com/packages/3d/environments/sci-fi/megalopoly-art-vintage-control-room-190538>
- **Plants Package** per le piante presenti all'interno della serra <https://assetstore.unity.com/packages/3d/vegetation/plants/plants-150261>
- **PolyAngel_Space Pack** per la maggior parte degli oggetti presenti nella stanza: gli astronauti, gli edifici, la torre di controllo, il terreno, la rampa di lancio del teletrasporto, i pannelli solari, la cupola spaziale e le macchine <https://assetstore.unity.com/packages/3d/vehicles/space/poly-angel-space-pack-267010>
- **SpaceSkie Free** per il cielo <https://assetstore.unity.com/packages/2d/textures-materials/sky/spaceskies-free-80503>
- **Scifii Styled Modular Pack** per creare l'ambientazione ideale, quindi il corridoio della serra, le panchine e i tavoli della serra, il monumento vicino alla serra, le porte della serra e del teletrasporto, le luci della serra, <https://assetstore.unity.com/packages/3d/environments/sci-fi/sci-fi-styled-modular-pack-82913>
- **Sci-fi PRB Props** per il teleport speciale, per le componenti hardware della porta del teletrasporto, per gli schermi e i tubi dei pannelli solari, per i bidoni della spazzatura e per alcune scatole sparse nella scena <https://assetstore.unity.com/packages/3d/environments/sci-fi/sci-fi-pbr-props-118783>

Il robot chiamato **X_Bot** (alla fine non inserito nella scena) è stato scaricato da Mixamo <https://www.mixamo.com/#/?page=1&query=x+bot&type=Character>.

L'unico Asset non scaricato direttamente da Unity Asset Store è stato **Unity VR Basics 2023 - Input Systems and VR Hands**. Questo è stato preso da GitHub <https://github.com/Fist-Full-of-Shrimp/VR-Unity-Template-2023>

Prefabs

Oltre ai prefab presenti negli asset scaricati da Unity Asset Store, gli altri prefab di questa scena sono stati creati o sono delle modifiche ai prefab presenti negli asset scaricati.

In particolare, è possibile individuare:

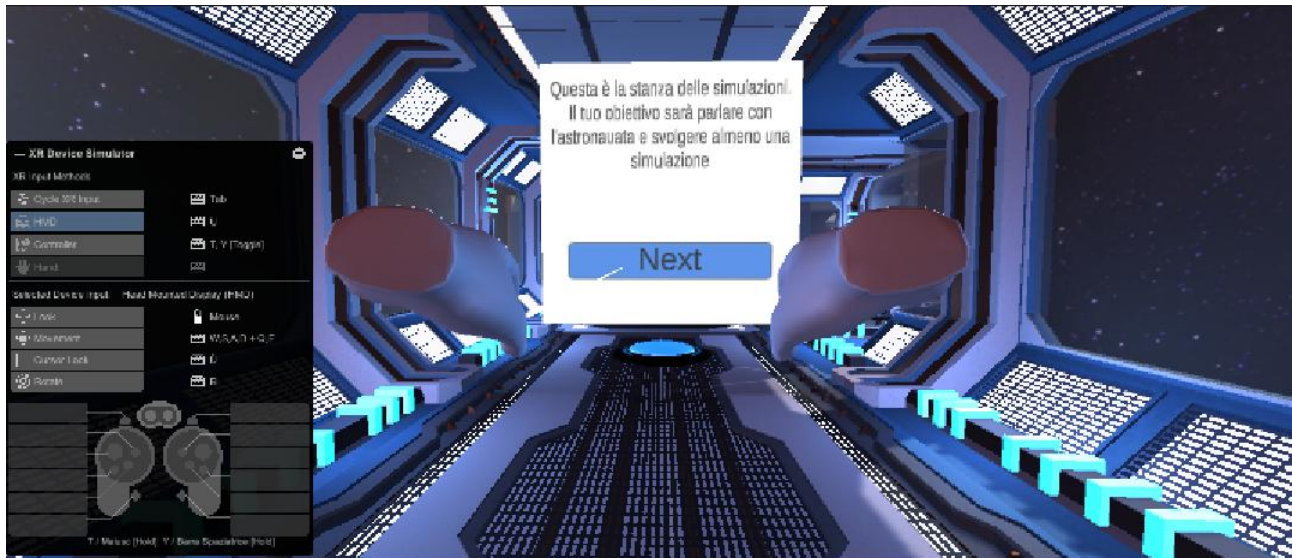
- Lampioni, creati utilizzando dei semplici piani e una luce presa dal pack Scifii Styled Modular Pack;
- Variante gialla del tavolo presente in Sci-Fi Styled Modular Pack > Prefabs > Decorative elements > Tables.

È poi presente nella cartella dei prefab una cartella animazioni, in cui sono presenti le animazioni relative agli UFO, alla macchina "Truck03" e agli astronauti.

SCENA SIMULAZIONE

Descrizione Scena

La scena inizia ad un lato della navicella. La prima cosa che vedere l'utente sono delle GUI che presentano all'utente la stanza.



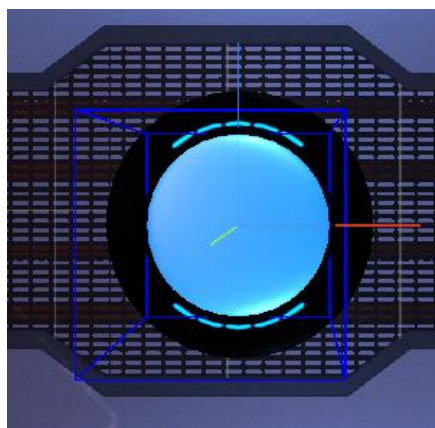
L'utente una volta letta la prima GUI può passare a quella successiva premendo il bottone "Next". Una volta premuto "Next" la GUI precedente non sarà più disponibile. All'interno delle GUI sono presenti i seguenti testi, volti a spiegare come l'utente deve comportarsi nella scena:

- 4) "Questa è la stanza delle simulazioni. Il tuo obiettivo sarà parlare con l'astronauta e svolgere almeno una simulazione"
- 5) "Anche in questa stanza potrai sfruttare il teletrasporto per muoverti più velocemente."
- 6) "Per poter interagire con l'astronauta dovrai cliccare "G" e il tasto sinistro del mouse. I comandi per svolgere la simulazione ti verranno spiegati dopo aver parlato con l'astronauta."

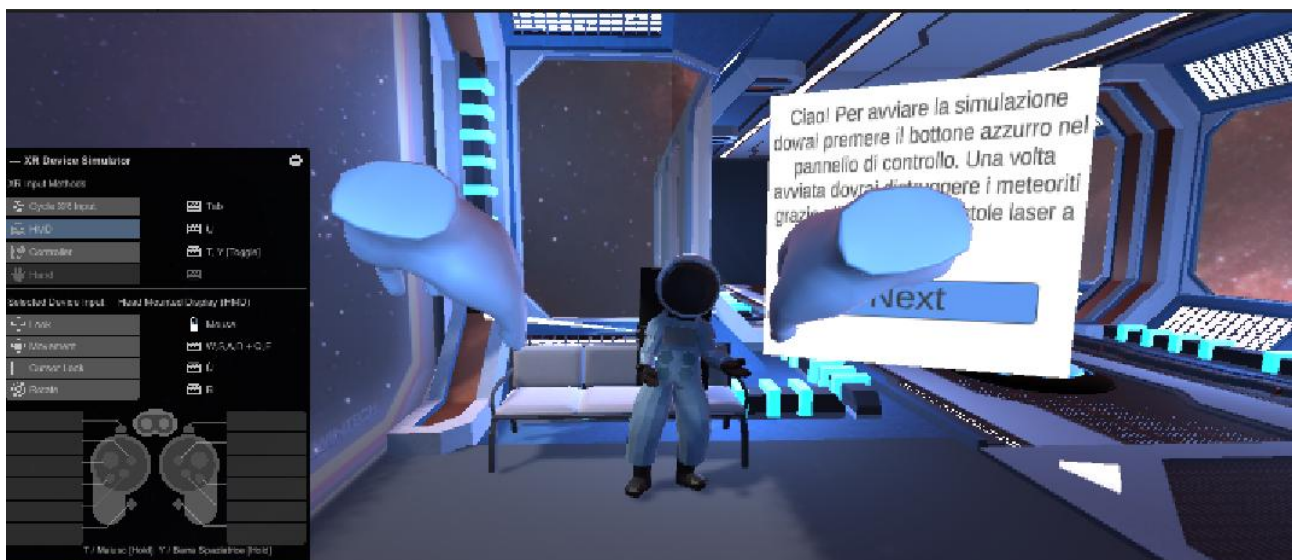
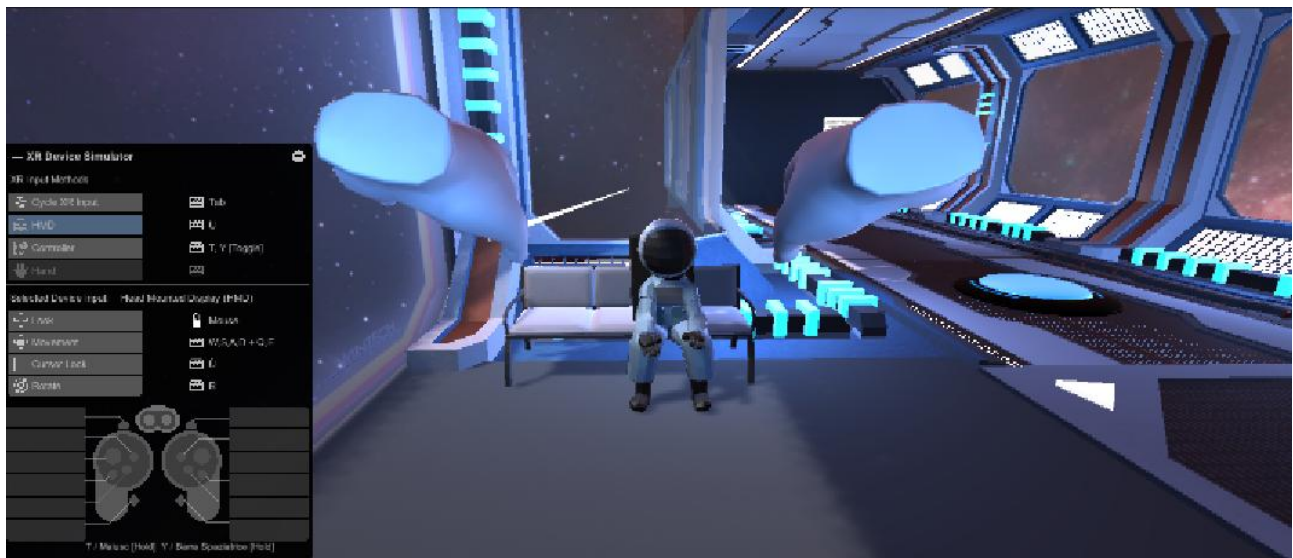
L'utente come prima cosa dovrà quindi dirigersi dall'astronauta.

Per arrivare all'astronauta può scegliere se muoversi in maniera naturale, oppure sfruttare i Teleport.

NOTA: l'utente non riuscirà ad interagire con l'astronauta se non ha premuto "Okay" nell'ultima delle tre GUI di spiegazione iniziale.



L'astronauta, se non sta parlando con l'utente, si trova seduto su una panca, nel momento in cui l'utente ci interagisce appare una GUI e l'astronauta si alza in piedi e inizia a muovere le mani.



La GUI funge da "finestra di dialogo" e serve all'utente per capire come svolgere la simulazione.

All'interno delle GUI sono presenti i seguenti testi:

- 1) "Ciao! Per avviare la simulazione dovrai premere il bottone azzurro nel pannello di controllo. Una volta avviata dovrai distruggere i meteoriti grazie all'ausilio delle pistole laser a disposizione"
- 2) "Per poter premere il pulsante basterà interagirci cliccando il pulsante "G" e il tasto sinistro del mouse. Per sparare con la pistola bisognerà prima equipaggiarla e poi premere la barra spaziatrice."

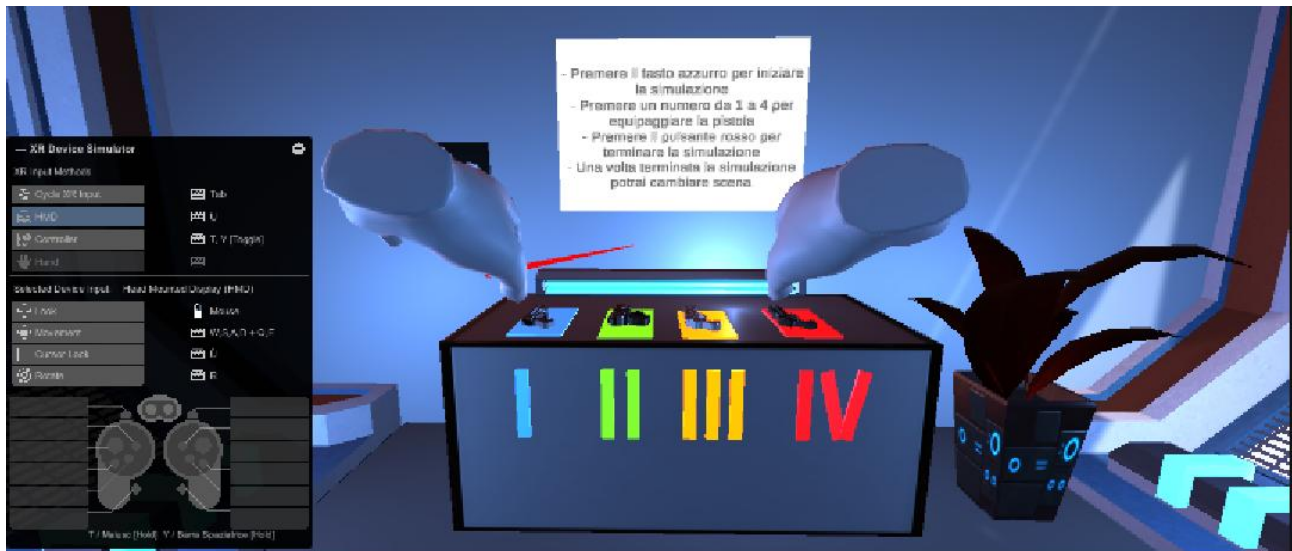
Una volta finito di interagire con l'astronauta la GUI scompare.

A questo punto l'utente si può dirigere verso il lato della stanza dedicato alla simulazione. Le spiegazioni per attivare la simulazione e caricare l'arma si trovano subito sopra il ripiano delle armi.

Le istruzioni sono:

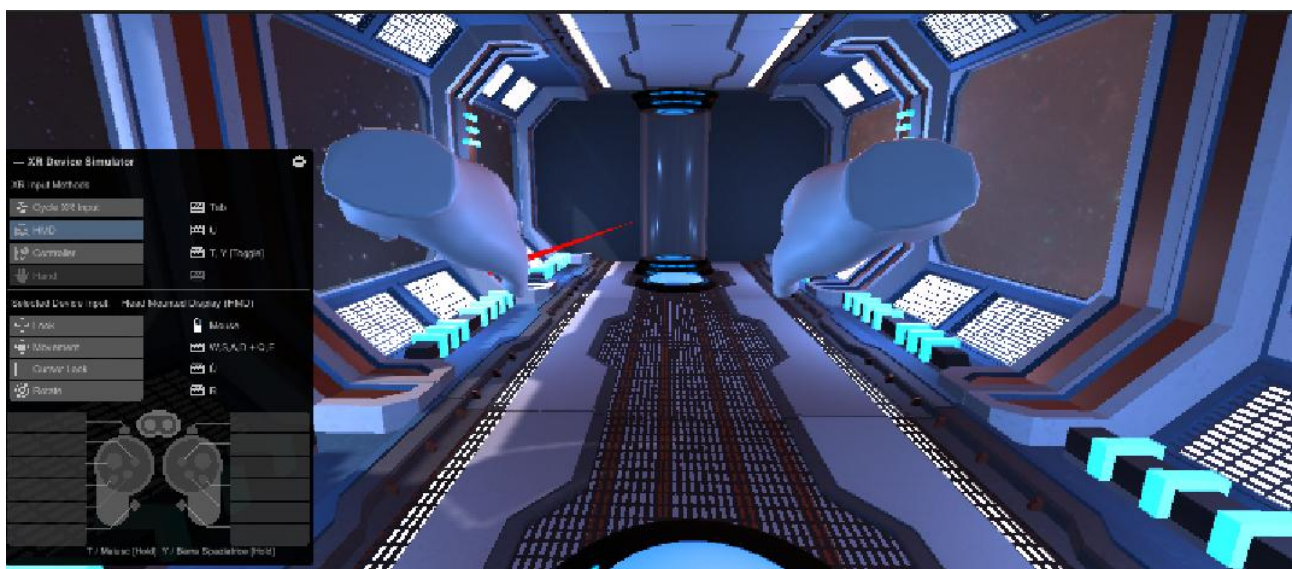
- Premere il tasto azzurro per iniziare la simulazione
- Premere un numero da 1 a 4 per equipaggiare la pistola
- Premere il pulsante rosso per terminare la simulazione
- Una volta terminata la simulazione potrai cambiare scena

NOTA: l'utente non riuscirà a vedere le istruzioni se non ha prima interagito con l'astronauta e premuto il tasto "Next" della prima GUI.



Seguite le istruzioni l'utente potrà svolgere la simulazione e distruggere qualche meteorite.

Una volta terminata la simulazione (bisogna premere il pulsante rosso) apparirà all'utente la "porta" per la stanza successiva.



In realtà si tratta più che di una porta di un teleport "speciale" a cui è associato uno script che permette all'utente di cambiare stanza.

Lista oggetti manipolabili

Astronauta

È possibile interagire con l'astronauta grazie alla presenza dell'XR Simple Interactable. L'interazione con l'astronauta permetterà a quest'ultimo di cambiare animazione alzandosi e iniziando a parlare oppure

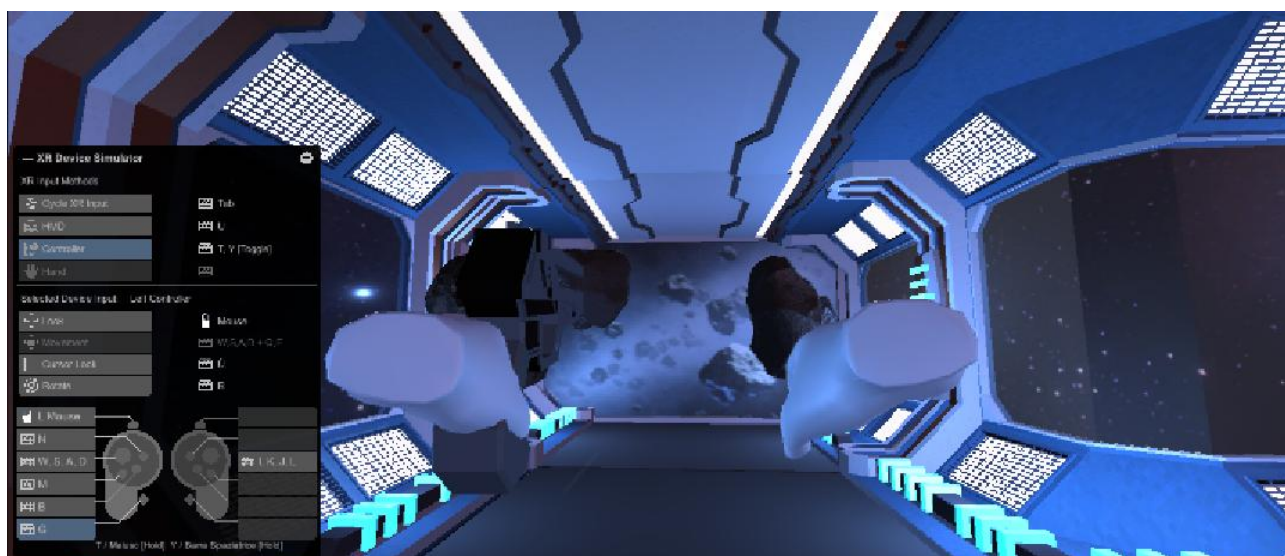
sentendosi e smettendo di parlare. Questo cambio di animazione è regolato anche grazie ad uno script chiamato “Character Interaction” che gestisce due parametri booleani: “Interact” e “Talking”.

Meteoriti

Lo scopo dell'utente è distruggerli, per farlo dovrà necessariamente utilizzare una delle pistole, non è possibile distruggerli in un altro modo. Questo è gestito all'interno dello script “Fracture”.

Una volta distrutti questi si riformano nella stessa posizione e compiendo gli stessi movimenti (questo perché non si tratta di meteoriti "reali" è una simulazione, dunque si presuppone che l'utente possa esercitarsi a sparare all'infinito ipoteticamente).

L'apparizione dei meteoriti dipende dallo script “Visibility Controller”.

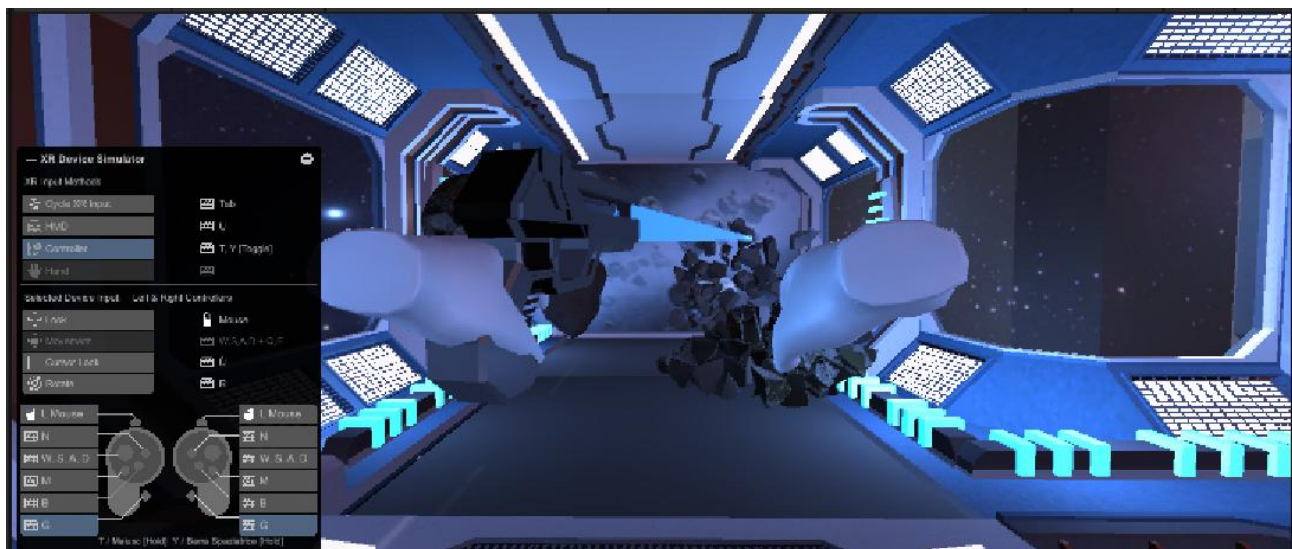


Monitor

Quando viene premuto il pulsante blu, sul monitor apparirà un video ad indicare l'inizio della simulazione. Premendo il pulsante rosso si spegnerà anche il monitor, di conseguenza il video scomparirà.

Pistole

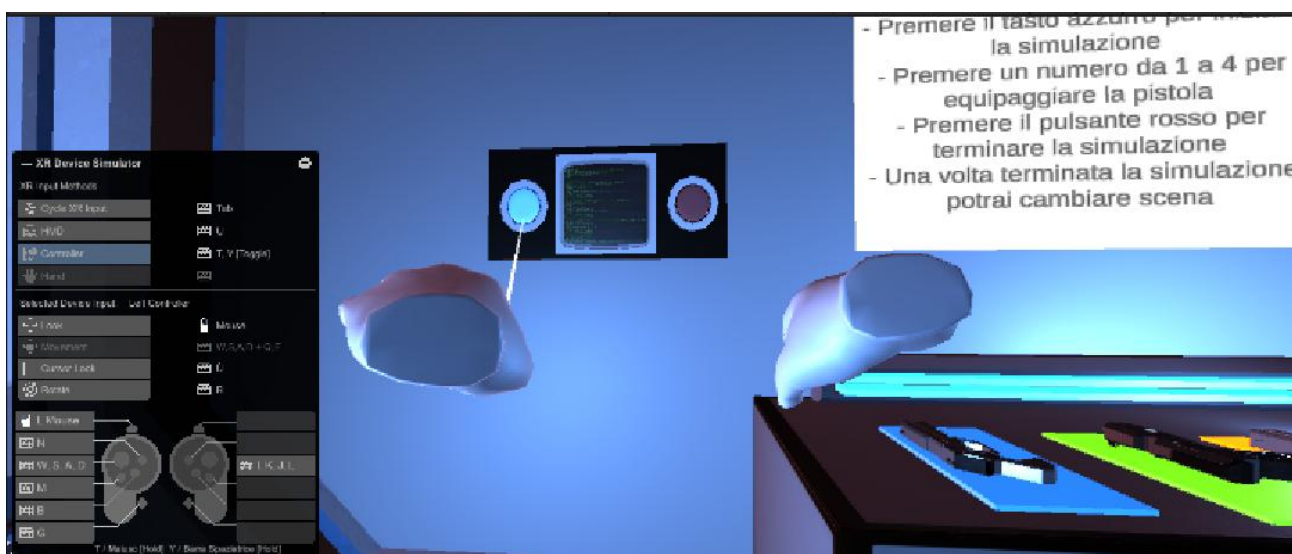
Ad ogni pistola è associato un raggio laser. Per attivarlo basterà premere un numero da 1 a 4 a seconda delle pistole. L'utente può interagirci grazie alla presenza di un XR Grab Interactable e può attivare i raggi laser grazie allo script “Laser Gun”.



Pulsante blu

All'interno dell'XR Simple Interactable, in Interactable Events nella sezione "Activate" sono presenti diverse associazioni.

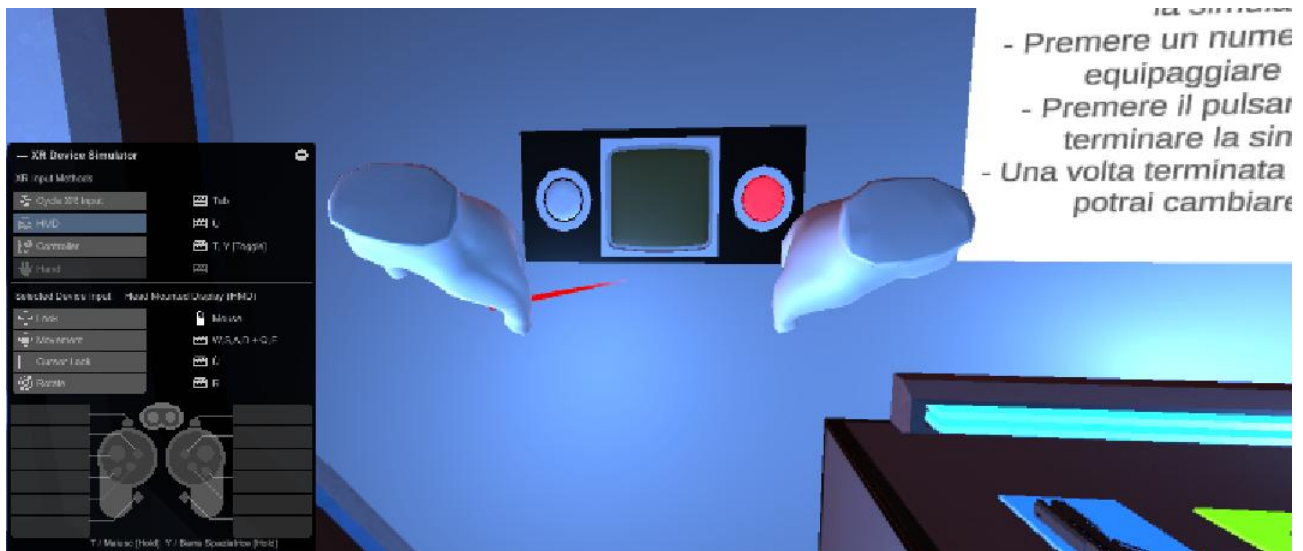
In primo luogo, premendo il pulsante la simulazione avrà inizio e partirà sia un suono di accensione del pulsante, sia il suono di accensione della simulazione associato allo schermo. Il pulsante si illuminerà, mentre il pulsante rosso si spegnerà. Inoltre, l'effetto più importante dell'interazione con il pulsante sarà l'apparizione di quattro meteoriti in movimento.



Pulsante rosso

All'interno dell'XR Simple Interactable, in Interactable Event nella sezione "Activate" sono presenti diverse associazioni. Questo pulsante risulterà "premibile" solo dopo aver interagito con il pulsante blu.

L'interazione con questo pulsante comporta in primo luogo l'attivazione del suono di spegnimento della simulazione e il suono di accensione del pulsante. Il pulsante si illuminerà, mentre il pulsante blu si spegnerà. Inoltre, spariranno i meteoriti e apparirà, in fondo alla stanza, il teleport speciale che permette di cambiare stanza.



Teleport

Sono i punti di teletrasporto, permettono di muoversi più velocemente nella stanza. L'utente può interagirci grazie alla presenza del Locomotion System e Teleportation Provider nell'XR Rig e grazie alla presenza del Teleportation Anchor all'interno del prefab del teleport.

Teleport speciale

Permette all'utente di passare da una stanza all'altra. L'utente può interagirci grazie alla presenza dell'XR Simple Interactable a cui è associato un Interactable Event nella sezione "Activate" che presenta come oggetto il teleport speciale stesso in quanto lo script "RoomTransition" si trova direttamente sull'oggetto. L'apparizione di questo oggetto è regolata dallo script "VisibilityController"

Lista script utilizzati nella scena

- **ChangeMaterial:** utilizzato per cambiare lo sfondo dietro la simulazione degli asteroidi
- **LaserGun:** utilizzato per caricare le pistole e sparare il raggio laser
- **LoadScene:** utilizzato per ricaricare la scena corrente e *in contemporanea teletrasportare l'utente nel punto iniziale*
- **Fracture:** utilizzato per "rompere" gli asteroidi
- **OnButtonPress:** permette di associare a dei comandi altri script e funzioni, è utilizzato soprattutto nell'XR Rig e all'interno delle due mani
- **PlayQuickSound:** utilizzato per far partire il suono dei pulsanti e della simulazione
- **PlayVideo2:** utilizzato per far partire un video sul monitor quando parte la simulazione
- **RoomTransition:** utilizzato per cambiare stanza
- **ToggleRay:** utilizzato per abilitare i raggi della mano destra
- **VisibilityController:** utilizzato per regolare l'apparizione e la scomparsa di vari oggetti
- **UIPositioning:** utilizzato per far apparire la GUI contenente il riassunto dei comandi generali

Alcuni script sono stati presi direttamente dalle esercitazioni fatte in classe (ovvero LoadScene, OnButtonPress, PlayQuickSound, PlayVideo, ToggleRay). Allo script PlayQuickSound è stato aggiunto un metodo Play2 che permette di riprodurre un suono in loop.

Descrizione asset utilizzati

La maggior parte degli asset presenti in questa scena è stata reperita direttamente da Unity Asset Store.

In particolare, sono stati utilizzati all'interno della scena i seguenti asset:

- **Breakable Asteroid** per gli asteroidi <https://assetstore.unity.com/packages/3d/props/breakable-asteroids-167825>
- **Megalopoly.Art Vintage Control** per i comandi della simulazione <https://assetstore.unity.com/packages/3d/environments/sci-fi/megapoly-art-vintage-control-room-190538>
- **Plants Package** per la pianta presente come decorazione <https://assetstore.unity.com/packages/3d/vegetation/plants/plants-150261>
- **SpaceSkie Free** per il cielo <https://assetstore.unity.com/packages/2d/textures-materials/sky/spaceskies-free-80503>
- **Sci-fi HandGun** per la pistola laser <https://assetstore.unity.com/packages/3d/props/guns/sci-fi-handgun-225160>
- **Scifi Styled Modular Pack** per creare l'ambientazione ideale <https://assetstore.unity.com/packages/3d/environments/sci-fi/sci-fi-styled-modular-pack-82913>
- **Sci-fi PRB Props** per il teleport speciale <https://assetstore.unity.com/packages/3d/environments/sci-fi/sci-fi-pbr-props-118783>

L'unico Asset non scaricato direttamente da Unity Asset Store è stato **Unity VR Basics 2023 - Input Systems and VR Hands**. Questo è stato preso da GitHub <https://github.com/Fist-Full-of-Shrimp/VR-Unity-Template-2023>

Prefab

I prefab creati per questa scena sono delle modifiche ai prefab presenti negli asset scaricati. In particolare è possibile individuare:

- Gli asteroidi utilizzati per la simulazione;
- Le pistole con tanto di raggio laser integrato;
- L'oggetto utilizzato per appoggiare le pistole;
- La struttura dei comandi per la simulazione, con lo schermo e i due pulsanti;
- Il teleport utilizzato per muoversi nella stanza;
- Il teleport speciale utilizzato per cambiare stanza;

È poi presente nella cartella dei prefab, una cartella materials in cui è possibile trovare alcuni materiali creati da zero e utilizzati all'interno della scena; e una cartella animazioni, in cui sono presenti le animazioni relative ai meteoriti e all'astronauta.

SCENA NAVICELLA

Descrizione scena

La scena inizia ad un lato della navicella. La prima cosa che vede l'utente sono delle GUI che presentano le varie stanze esplorabili.



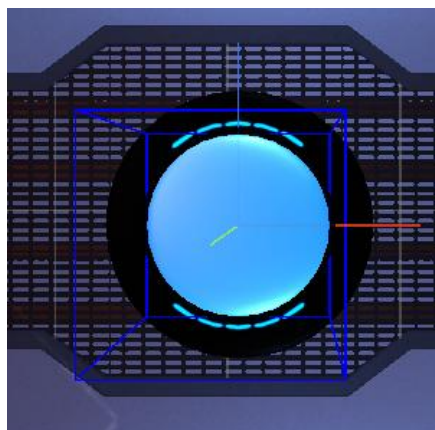
L'utente una volta letta la prima GUI può passare a quella successiva premendo il bottone "Next". Una volta premuto "Next" la GUI precedente non sarà più disponibile.

All'interno delle GUI sono presenti i seguenti testi, volti a spiegare come l'utente deve comportarsi nella scena:

1. Benvenuto nel cuore della navicella.
Ora che hai completato la simulazione sei pronto a partire verso lo spazio.
2. Puoi esplorare 3 stanze:
-La sala motori (a sinistra)
-La sala comandi (dritto)
-La sala relax (a destra)
3. *Per poter far decollare l'astronave dalla sala comandi dovrai prima recarti in sala motori e trovare il pulsante rosso per accendere i motori*

L'utente come prima cosa dovrà quindi raggiungere la sala motori.

Per arrivare alla sala motori può scegliere se muoversi in maniera naturale, oppure sfruttare i Teleport.

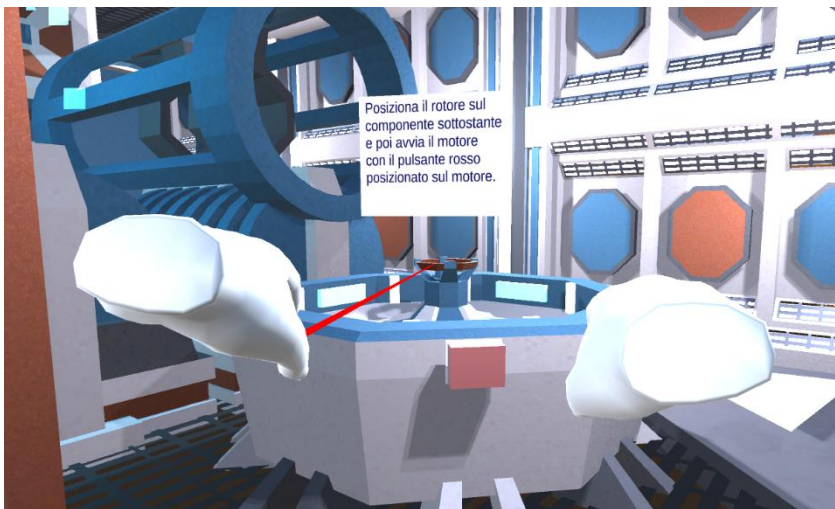


Una volta raggiunta la sala motori, all'ingresso è presente un'altra serie di GUI che spiegano l'attività da svolgere nella stanza.

Nello specifico riportano i seguenti testi:

4. Sei arrivato nella sala motori
Per poter avviare l'astronave dovrai sistemare il motore rimontando un pezzo mancante e poi accenderlo con l'apposito pulsante
5. Esplora la sala motori per trovare il pezzo mancante e il pulsante di accensione

Dopo essersi addentrati nella stanza si noterà un componente del motore appoggiato ad una parete e più avanti sarà visibile una piccola GUI con le indicazioni su come rimontarlo e accendere il motore.



Dopo aver acceso il motore con il pulsante, sarà possibile vedere che il rotore principale del motore inizierà a ruotare, e contemporaneamente apparirà il messaggio *“Ora che hai acceso i motori dirigi ti verso la sala comandi”* in una piccola GUI a fianco.

Dopo aver acceso i motori l'utente può quindi raggiungere la sala comandi. Prima di entrare nella sala troverà una porta chiusa, per aprirla è necessario effettuare il riconoscimento della mano sul display posto a lato della porta, simultaneamente sarà visualizzata una GUI con le indicazioni per aprire la porta. Dopo aver interagito con lo schermo partirà un'animazione e si aprirà la porta per accedere alla sala comandi.

Nella sala comandi si troverà un'ulteriore GUI con le seguenti indicazioni:

6. Benvenuto nella sala comandi.
Hai completato tutti i compiti per poter far decollare l'astronave.
7. Cerca sulle postazioni di comando un pulsante rosso di lancio.
Premendolo verrà dato inizio alla procedura di decollo.
Buon viaggio

Nella sala inoltre saranno presente lo schermo principale situato al centro della stanza.

Premendo il pulsante rosso dalla postazione di comando si avvierà il conto alla rovescia di 20 secondi (visibile sullo schermo centrale) al termine del quale avverrà il decollo dell'astronave (allontanamento del suolo).

Il gioco finisce qui, ma volendo l'utente può continuare ad esplorare le stanze, inoltre come comunicato da una GUI aggiuntiva visibile dopo aver iniziato il decollo è possibile tornare nelle scene precedenti utilizzando dei teletrasporti speciali che compaiono all'inizio della scena.

Lista oggetti manipolabili

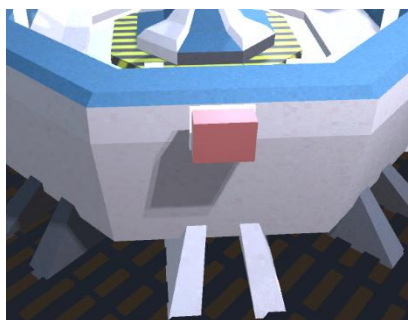
Display porta

Il Display della porta inizialmente visualizza un video di riconoscimento palmare. All'interno di Interactable Event nella sezione "Activate" è presente l'evento che regola il cambio di colore del display (a verde) selezionando MeshRender.material dall'interactor e inserendovi il material "Emission Green" e altri due gli eventi che regolano l'apertura delle porte chiamando dei metodi di due script rispettivamente inseriti nelle due sezioni della porta.



Pulsante motore

All'interno dell'XR Simple Interactable, in Interactable Event nella sezione "Activate" sono presenti diverse associazioni. In particolare, premendo il pulsante partirà un suono relativo all'accensione dei motori, partirà l'animazione del rotore del motore e in secondo luogo sarà da quel momento in poi "visibile" il pulsante rosso per avviare la partenza della navicella, oltre a far progredire le schermate della UI



Pulsante rosso sala comandi

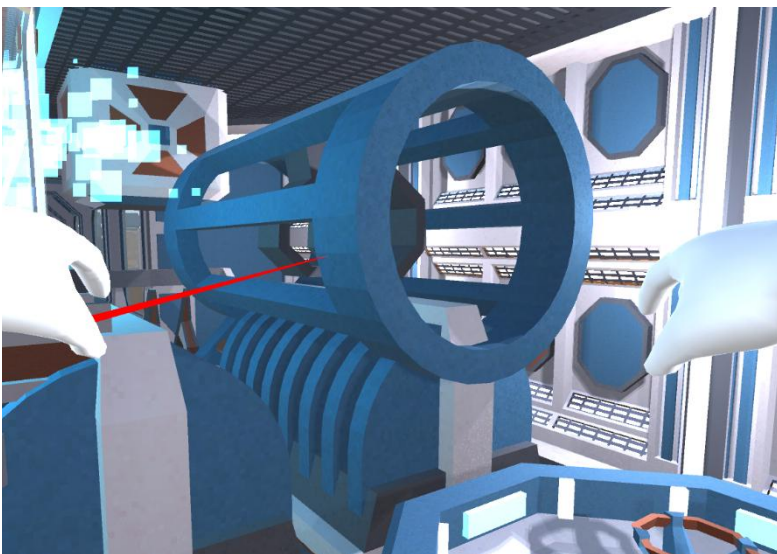
All'interno dell'XR Simple Interactable, in Interactable Event nella sezione "Activate" sono presenti diverse associazioni. Questo pulsante risulterà "visibile" solo dopo aver interagito con il pulsante del motore.

Una volta premuto il pulsante, nello schermo principale della navicella il video con tutti i comandi verrà sostituito da un video di un countdown di 20 secondi e partirà anche un audio volto a segnalare l'avviata partenza (automaticamente avverrà anche il decollo anche se il metodo non è richiamato direttamente da un evento dell'Interactor di questo oggetto), inoltre successivamente all'interazione con questo pulsante saranno visibili i teletrasporti speciali alle scene precedenti.



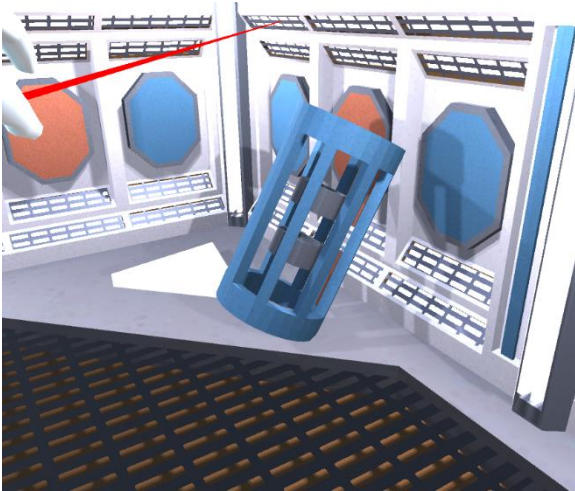
Rotore del motore in sala motori

Su questo componente è presente uno script che contiene un metodo per fare ruotare su sé stesso questo componente usando il metodo "Transform.Rotate". Una volta premuto il pulsante di accensione dei motori, se il componente è stato rimontato verrà chiamato il metodo per far iniziare a ruotare il componente.



Componente del motore da montare in sala motori

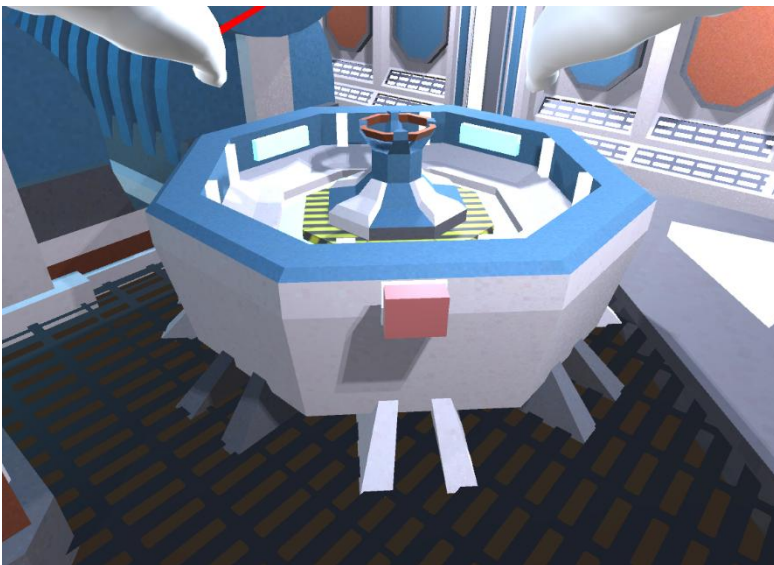
All'interno dell'oggetto è stato inserito il componente "XR Grabbable Interactor" per poterlo afferrare, tenerlo in mano, muoverlo e posizionarlo nel socket e sistemare il motore.



Socket componente da inserire in sala motori

È stato creato un oggetto Socket figlio al componente del motore, all'interno della socket è stato aggiunto il componente "XR Socket Interactor" per poter gestire l'evento nel quale il componente fosse rimontato sulla socket.

All'interno dell' Socket Interactor, in Interactable Event nella sezione "Select Entered" sono presenti diverse associazioni che rispettivamente impostano i valori delle variabili usate per controllare se il motore è rimontato e permettono alle GUI di mostrare la schermata successiva.



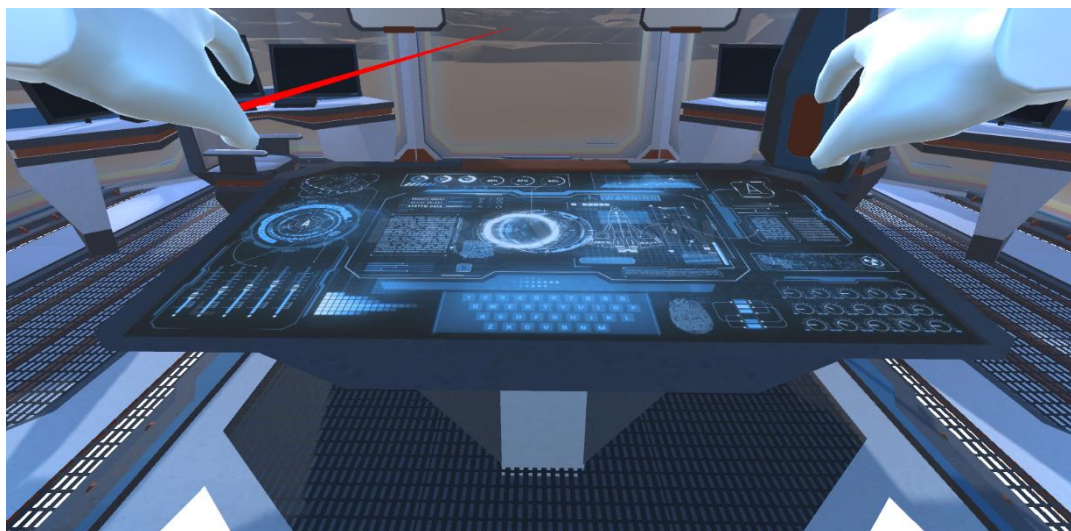
Monitor centrale sala comandi

Questo oggetto ha impostati i seguenti componenti per la visualizzazione video:

- Video Player
- Script per la gestione della visualizzazione delle clip video
- Audio source

In particolare lo script "playvideonav" permette contiene un metodo che se invocato sospende la visualizzazione del video corrente e imposta la visualizzazione sulla clip del countdown, successivamente con un ciclo e un controllo sullo stato del videoplayer aspetta che la visualizzazione della clip di

countdown sia terminata quindi riproduce nuovamente in “loop” la clip di stato generale astronave, successivamente effettua la chiamata al metodo dello script di lancio per effettuare il decollo e dopo un'attesa di 12 secondi (tempo per arrivare in orbita) effettua una chiamata allo script per modificare lo skybox impostando lo spazio come sfondo.



Suolo

Questo oggetto contiene lo script “launch” nel quale è presente il metodo decollo invocato dopo il termine del countdown. Questo metodo permette, modificando il valore di “Transform.position” permette di variare la posizione dell’oggetto, in questo caso è utilizzato per spostare il suolo verso l’alto in modo che dall’interno dell’astronave venga simulato un decollo.

Teleport

Il medesimo descritto nella scena simulazione

Teleport speciale

Il medesimo descritto nella scena simulazione

Lista script utilizzati nella scena

- **ChangeMaterial:** per cambiare il materiale del display da rosso a verde
- **LoadScene:** utilizzato per ricaricare la scena corrente e *in contemporanea teletrasportare l’utente nel punto iniziale*
- **OnButtonPress:** permette di associare a dei comandi altri script e funzioni, è utilizzato soprattutto nell’XR Rig e all’interno delle due mani
- **PlayQuickSound:** utilizzato per far partire il suono dei vari pulsanti della sala comandi, della sala motori e della sala relax
- **playvideonav:** utilizzato per far partire un video sul monitor all’inizio del gioco e per avviare il conto alla rovescia; In particolare una volta messo in riproduzione il conto alla rovescia attende che la riproduzione si concluda e rimette in riproduzione il video con i dati della navicella e chiama un metodo dello script per il decollo.
- **RoomTransition:** utilizzato per cambiare stanza

- **ToggleRay**: utilizzato per abilitare i raggi della mano destra
- **UIPositioning**: utilizzato per far apparire la GUI contenente il riassunto dei comandi generali
- **launch**: script richiamato dallo script "playvideonav" che permette l'allontanamento dell'astro-nave dal suolo per mimare il decollo
- **apertura**: utilizzato per effettuare l'animazione di apertura della porta muovendo la parte superiore verso l'alto
- **apertura2**: utilizzato per effettuare l'animazione di apertura della porta muovendo la parte inferiore verso il basso
- **avviomotore**: script che controlla che il componente sia stato rimontato (controllo variabili booleane) e successivamente attiva la rotazione del rotore del motore
- **ChangeSky**: script usato per cambiare lo skybox a runtime una volta che la navicella arriva in orbita (impostato attesa di tot secondi dopo il decollo)

Descrizione asset utilizzati

- **Megalopoly.Art Vintage Control** per i vari pulsanti utilizzati <https://assetstore.unity.com/packages/3d/environments/sci-fi/megalopoly-art-vintage-control-room-190538>
- **Plants Package** per la pianta presente come decorazione <https://assetstore.unity.com/packages/3d/vegetation/plants/plants-150261>
- **SpaceSkie Free** per il cielo <https://assetstore.unity.com/packages/2d/textures-materials/sky/spaceskies-free-80503>
- **Scifii Styled Modular Pack** per creare l'ambientazione della navicella <https://assetstore.unity.com/packages/3d/environments/sci-fi/sci-fi-styled-modular-pack-82913>
- **Sci-fi PRB Props** per il teleport speciale e per alcuni materiali <https://assetstore.unity.com/packages/3d/environments/sci-fi/sci-fi-pbr-props-118783>

L'unico Asset non scaricato direttamente da Unity Asset Store è stato **Unity VR Basics 2023 - Input Systems and VR Hands**. Questo è stato preso da GitHub <https://github.com/Fist-Full-of-Shrimp/VR-Unity-Template-2023>

Prefab

I prefab creati per questa scena sono delle modifiche ai prefab presenti negli asset scaricati.

Sono stati utilizzati i medesimi prefab descritti nella scena simulazione

SVILUPPI FUTURI

Abbiamo riscontrato delle difficoltà nello svolgere questo progetto e purtroppo non siamo riusciti a sviluppare tutte le funzioni che avevamo pensato.

In primis non siamo riusciti a rendere possibili le collisioni tra XR Rig e altri oggetti presenti nella scena. Abbiamo provato molte volte, sia guardando tutorial che andando a tentativi sfruttando Collider e Rigidbody, ma non ci siamo riusciti.

In secondo luogo, non siamo riusciti a sviluppare al meglio la funzione per ricaricare la scena corrente. La scena viene ricaricata correttamente, ma non si può dire lo stesso del Teleport Player. Non capiamo esattamente perchè, ma anche se da debug risulta la posizione corretta, di fatto l'utente viene teletrasportato da tutt'altra parte.

Avevamo pensato di aggiungere una transizione per far in modo che lo skybox cambiasse gradualmente da quello del pianeta allo spazio, per fare ciò è stata aggiunta una shader a uno che poi è stato impostato come valore di “Skybox Material”, la shader aggiunta permetteva di aggiungere 2 skybox di tipo “Cubemap” e veniva correttamente visualizzata, tuttavia abbiamo riscontrato problemi nel modificare a runtime il valore per controllare il “blend” tra le due skybox.

Nella scena iniziale abbiamo avuto problemi con i due script *moveMainCamera* e *WindVolumeController*.

Il primo script (*moveMainCamera*) aveva il compito di spostare la Main Camera vicino al teletrasporto quando l'utente appoggia la porta gialla del teletrasporto sul socket relativo tavolo, ma non funziona perché *transform.position* probabilmente ha dei problemi con il fatto che la Main Camera sia un elemento child di Camera Offset che è un elemento child dell'XR Origin (XR Rig). Ho quindi aggiunto lo script all'XR Rig invece che alla Main Camera ma questo genera due problemi. Il primo è che non sempre si sposta nella direzione giusta (a volte sposta la Main Camera nella direzione opposta), forse perché ostacolato dalla porta e da Alpha 1 (rispettivamente NavMeshObstacle e NavMeshAgent) e il secondo è che, quando anche si sposta nella direzione corretta, è l'XR Rig a spostarsi nella direzione desiderata e non la Main Camera, per cui l'utente può capitare che finisca in un punto non particolarmente sensato o vicino al teletrasporto.

Il secondo script (*WindVolumeController*) aveva, invece, il compito di controllare il volume del vento (rumore di sottofondo dell'intera prima scena) in modo che esso cessasse o si abbassasse molto quando la Main Camera (e quindi l'utente) entrava nella serra. Questo avrebbe garantito un maggiore grado di realismo alla scena, perché l'utente avrebbe sentito il rumore del vento molto debole e il rumore delle gocce d'acqua nella serra in maniera più marcata. Tuttavia, non siamo riusciti a farlo funzionare correttamente, forse a causa di qualche impostazione sbagliata del BoxCollider a cui è associato lo script, che avrebbe dovuto far avvertire più lieve il suono del vento in tutta la sua zona di collisione.

Abbiamo avuto delle problematiche all'ultimo su Git, di conseguenza alcune modifiche che avevamo fatto sono andate perdute (come, ad esempio, mettere in tutte le GUI il tasto “indietro”), purtroppo non avevamo più tempo di ricorreggere tutto, ma ci tenevamo a specificarlo per far vedere che almeno ci avevamo pensato :)