**Regole gioco:**

-Scegliere con quale mano si vuole pinzare gli oggetti

Esempio con la mano destra (nell'altro caso i tasti saranno invertiti):

-Spuntare i settings che si vogliono adottare durante l'esperimento usando il controller di sinistra con il tasto "trigger"(levetta posteriore)

-Cliccare su start usando la mano sinistra sempre con il tasto "trigger"(levetta posteriore)

-Verrà mostrato l'oggetto da pinzare (l'ordine è fissato, si inizia dalla prima buca a sinistra e si prosegue con le successive)

-Usare il tasto "Grip"(levetta laterale) per grabbare la pinza

-Ci si può tele trasportare usando il bottone "A" o "X" dei rispettivi controller e inoltre ci si può ruotare ogni volta di 15° usando i "joystick"

-Tenere premuto il "Trigger" del controller di destra per chiudere la pinza mentre si mantiene premuto il Grip

-Una volta pinzato l'oggetto con successo, verrà mostrato automaticamente l'oggetto successivo da pinzare.

**Struttura Gioco**

**Scena principale:**

L'utente può scegliere se pinzare con la mano destra o con la sinistra.

**Scena di gioco**

-Pannello UI che mostra la mappatura dei bottoni dei controllers

-Pannello UI che mostra le carte degli oggetti pescati + "start" button

-Pannello UI con 7 settings(sono tutti attivi nel momento in cui si entra nella scena del gioco):

1. **Detect pinza collision:**

Il sistema individua le collisioni tra la pinza e la buca e simula la caduta dell'oggeto resettandolo alla sua posizione iniziale gli errori vengono salvati nel database remoto di firebase.

1. **Detect object collision:**

Il sistema individua le collisioni tra l'oggetto pinzato e la buca e simula la caduta dell'oggeto resettandolo alla sua posizione iniziale gli errori vengono salvati nel database remoto di firebase.

1. **Vibration Pinza:**

E' possibile attivare o rimuovere la vibrazione quando la pinza tocca la buca.

1. **Visual effect Pinza:**

E' possibile attivare o rimuovere gli effetti visivi della pinza (diventa rossa quando attraversa l'oggetto da pinzare o il tavolo)

1. **Visual effect object**

E' possibile attivare o rimuovere gli effetti visivi degli oggetti da pinzare (diventano rossi quando vengono pinzati ma non è il loro turno e quando invece vengono pinzati con successo, si dissolvono con un effetto di shaders).

1. **Sound Effect:**

E' possibile attivare o rimuovere tutti i suoni della scena (suono di vibrazione, suono dell'oggetto quando tocca il bordo elettrico e suono per indicare la buon riuscita del pinzare.

1. **Hide hand:**

E' possibile rendere visibile o invisibile la mano quando si pinza.

-Tavolo da gioco composto da 4 buche e 4 oggetti diversi (cubo, chiave inglese, cornice quadrata, esagono)

**Firebase Realtime Database**

**File Json partita:**

Ogni volta che si entra nella scena di gioco, viene salvato un file json associato ad un codice di sessione che corrisponde alla data e all'ora a cui si è fatto accesso. Vengono salvati tutti i valori booleani dei settings. Inoltre viene salvata una lista di oggetti. Per ogni oggetto che deve essere pinzato, viene creato un file in cui vengono salvate tutte le informazioni necessarie per analizzare i movimenti dell'utente nel cercare di pinzare il dato oggetto.

**JSON con i dati del test di un oggetto da pinzare:**

-start time

-end time

-id object

-number of errors when the object touches the electric edge

-number of errors when the tweezer touches the electric edge

-trajectory list:

La lista è costituita da tante traiettorie quanti sono gli errori che l'utente fa nel pinzare l'oggetto.

Ogni singolo oggetto traiettoria è costituito da una durata, un id, dalla lista dei vettori posizione e dalla lista dei vettori rotazione.

**Domande:**

-Servono altri settings?

-Quale metodo per teletrasportarsi è meglio, quello della scena con la mano destra dove ci si muove liberamente o quello della mano sinistra dove ci sono delle “postazioni” predefinite dove potersi teletrasportare?

-Dobbiamo salvare altre cose nel database?

-Al momento salviamo le liste dei vettori di posizione e rotazione della mano, dobbiamo salvare anche quelle della punta della pinza?

-Quante buche dobbiamo fare?

-Quanti Oggetti diversi dobbiamo realizzare?

-Va bene spezzare la traiettoria in più traiettorie in base a quanti errori l'utente compie?

-Gli oggetti che vengono pescati durante il gioco devono seguire un ordine preciso oppure randomico? (noi pensavamo di farlo con un ordine sempre fissato così da mettere ogni utente di fronte alla stessa difficoltà in modo tale da avere degli esperimenti più omogenei e confrontabili).

**Esempio struttura Json:**