#### Version 1.5

SolutionVR Unity Asset

Asset pack sur-couche de MiddleVR, qui permet quelques fonctions supplémentaires ainsi qu’une configuration des scènes plus personnalisable.

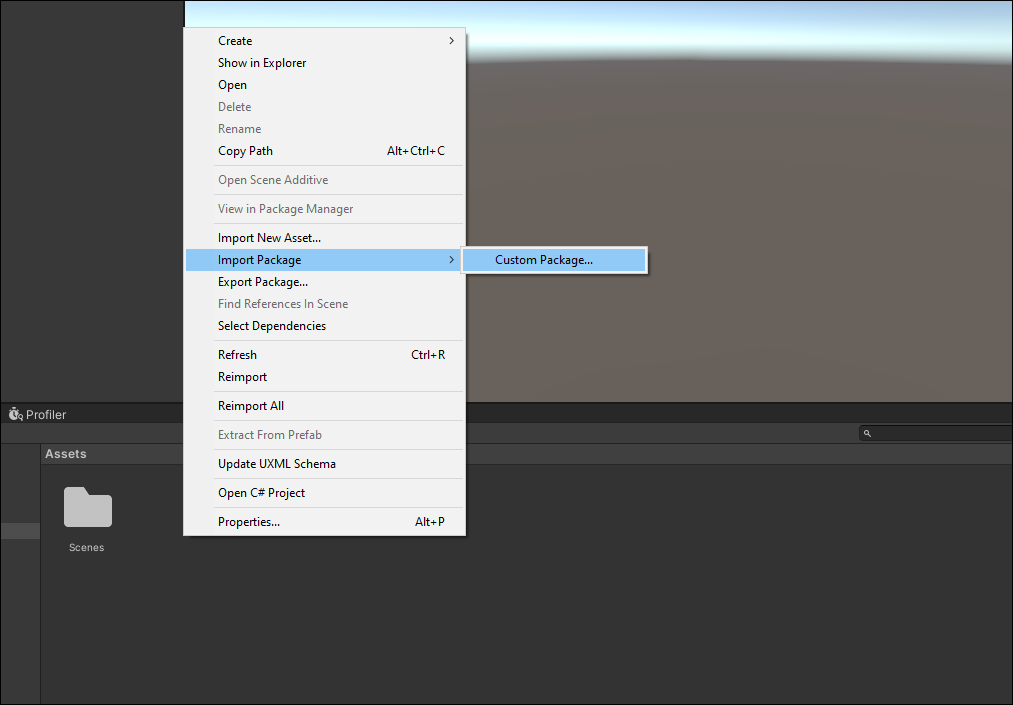
# Fonctionnalitées:

* Pouvoir naviguer dans une scène unity sans casques (Debug, etc..) via une config Default
* Prendre des photos
* Naviguer à des points d'intérêts
* Activer ou non le fly-mode

# Installation:

Tout d'abord, vous devez avoir un projet d'unité complet avec un préfabriqué de lecteur.

Importez le package dans votre projet unity



Glissez désormais le SolutionVRManager situé dans le dossier ./SolutionVR

# Configuration

## Ajouter une fonctionnalité

Pour ajouter une fonctionnalité, il y a plusieurs étapes.

### -Ajouter un bouton dans le menu:

Chaque bouton dans le panneau inférieur est identique sur la structure, un panneau qui contient le texte, l'image et le background.

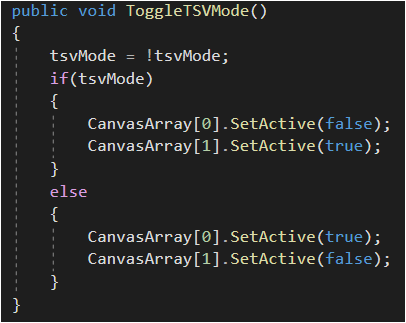
Il suffit donc de dupliquer un des boutons, et le positionner en dessous du dernier par exemple.

Le container des boutons est un “scrollview”, il faudra donc ajuster la propriété “Height“ dans le rect transform de l’objet “Content” afin de faire entrer tous les boutons dedans.

Modifiez ensuite le texte, l’image ainsi que les actions du nouveau bouton en suivant l'étape suivante.

### -Ajouter le script:

Dans le script SVRMenu, ajouter une fonction public qui servira de récepteur au clic sur le bouton, (exemple de la fonction ToggleTSVMode) et connectez la à votre bouton.



# Informations Techniques

## SVR Manager

Sur l’objet *Solution Manager*, le script *SVRManager* est le cerveaux de *SVR*, il possède trois catégories de paramètres:

**-Configuration:** Non nécessaire au fonctionnement de SVR, les paramètres passés ici sont des ajout adjacent servant l'amélioration ou l’ajout de fonctionnalités

* **Session path:** Choix du nom du fichier dans lequel seront enregistrées les informations lors du runtime (screenshots, etc..)
* **Using Config:** Ici sera mit un script servant à la modification de plusieurs configuration navigables au choix

**-WandCube:** Paramètre de personnalisation du WandCube par défaut de MiddleVR

* **Wand Cube Mesh:** Détermine la forme du Wand, le Cylindre est préféré
* **Wand Mat:** La texture du Wand

**-SceneOptions:** Informations globales sur la scène

* **Flying Mode:** Détermine si le joueur débute en mode de déplacement fly ou walk

**Awake:** Création du dossier portant le nom de “**Session path**” et attribution des textures choisies au wand.

**Update:** Check si le mode de déplacement est fly ou walk et envoie l’information au script **VRManagerScript**.

## Screenshot Camera

C’est le script qui vas gérer la capture et l’enregistrement d’images avec la camera libre

**start:** initialisation des variables

**Update:** Check si le bouton du wand choisi en paramètre est pressé pour prendre un screenshot

**ScreenshotName:** Définit le nom de la capture en fonction du nom de fichier racine (définis dans le **SVR Manager**) de la date et de l’heure

**TakeScreenShot:** Prend une capture de l’image de la caméra, de la taille définie en paramètres, appelle la fonction **ScreenShotName** et lance la coroutine **Cooldown** afin d'éviter les captures d'images massives

**Cooldown:** Coroutine qui modifie le texte de la caméra et empêche la prise de capture pendant 1.5 secondes

## ConfigScript

Ce script est modifiable, il sert de template déjà mis en place pour les différentes configurations utilisateurs, il contient un Switch-Case de base connecter au Menu via le script **SVRMenu**

**Update:** Récupère les inputs de l’utilisateurs via la fonction **GetInputs** et check le Switch-Case

**GetInputs:** Incrémente ou Décrémente la variable condition en fonction de l’input reçu

**IncrementVal** && **DecrementVal:** fonctions publiques appelée par les boutons du Menu dans le script **SVRMenu**

## SVR Menu

Ce script gère le fonctionnement du menu SVR

**Start:** Initialise les variables, cache les menus autre que le premier, check si le Bouton téléportation est masqué ou non en récupérant la liste des TSV dans le script **TSVManager** et check si l’user a le droit d'accéder au panel des configs.

**Update:** Modifie le texte des panels alternatifs ( config et playerspeed )

**Toggle Fly Mode:** Change la valeur du paramètre flyingMode du script **SVR Manager** et modifie le texte des boutons attribués.

**ToggleTSVMode:** Change la valeur du paramètre \_tsvMode et affiches ou cache les canvas associés

**ToggleScreenshotCam:** Change la valeur du paramètre \_screenCamEnabled, affiche ou cache la cam et modifie les textes attribués

**ToggleConfigGUI:** Change la valeur du paramètre \_showConfig, affiche ou masque le canva correspondant, change le texte et la couleur du bouton

**ToggleplayerSpeedGUI:** Change la valeur du paramètre \_showPlayerSpeed, affiche ou masque le canva correspondant et change la couleur de l’image du bouton

**ReloadScene:** Recharge la scene..

**TSVInit:** Récupère les instances du scripts TSVFlag présent sur la scène, les met dans une liste et prend une image de chacune des vue de caméra des TSV avant de les ranger dans le GUI correspondant, la taille du container des images est dynamique et gérée dans ce script

## SVR Canvas

Le même script que le MiddleVR (**VRCanvas**) en changeant uniquement le mode d’interaction (Pressed à Toggled)

## SVR Menu Manager

Le même script que le MiddleVR (**VRManager**) en changeant les instantiations et valeurs

## TSV Flag

Contient uniquement le nom de la localisation du TSV, ce script est utilisé pour compter le nombre de TSV et récupérer leurs informations

## TSV Navigation

Gère la navigation après clic sur l’une des images de TSV dans le menu

**TeleportView:** Fonction appelée par le clic, elle récupère l'indice du TSV et appelle la Coroutine **Teleport**

**Teleport**: Coroutine qui désactive les collisions du joueurs et le téléporte au coordonnée du TSV qui appelle la Coroutine.

## TSV Manager

Sur l’objet *Solution Manager*, le script *TSVManager* gère l'instantiation des TSV et leurs gestion, il possède deux parametres publics:

-**TSV Prefab**: Le prefab utiliser pour les TSV (par défaut celui dans le chemin “*SolutionVR/Prefabs/TSV”*  
-**Nbr tsv:** Définit le nombre de TSV à instancer, s’occupe aussi de les supprimer s’il y en a en trop, il ne faut donc pas supprimer les TSV(Clone) soit même

**Fonctionnement:** Lors de la définition du nombre *x* de TSV, Ce script instancie *x* TSV(“Clone”) Et envoie leurs informations dans une liste communiquée au script *SVRMenu* .

# Glossaire

**SVR**: Solution Virtual Reality

**TSV**: Téléporteur Sur Vue