Aim: To develop a simple UI(User interface) menu with images, canvas, sprites and button.

Theory:

Unity's XR Interaction Toolkit:
Unity's XR Interaction Toolkit is a comprehensive package designed to streamline the development of VR applications. It offers a set of tools and assets that facilitate the integration of VR functionalities across a variety of VR headsets. This toolkit simplifies complex tasks such as handling input from VR controllers, managing interactions, and creating user interfaces tailored for immersive experiences.

XR Plugin Management:
XR Plugin Management is a crucial component for ensuring compatibility with specific VR platforms. By selecting the appropriate VR plugin (such as Oculus XR Plugin), developers establish a connection between Unity and the target VR hardware. This step is essential for enabling seamless communication between the Unity environment and the VR headset.

XR Origin and World Space:
The XR Origin serves as the central point in the virtual environment, defining the coordinate system for the entire scene. By establishing this origin, objects within the virtual space can be accurately positioned, oriented, and scaled relative to the user's perspective. Setting the Tracking Origin Mode to 'Floor' ensures that the virtual world aligns with the physical floor, enhancing realism and user comfort.

Hand Controller Configuration:
Hand controllers are fundamental to the VR experience, acting as the user's primary means of interaction. In this experiment, the XR Interaction Toolkit is utilized to customize the behavior of these controllers. By adding components like XR Direct Interactor and Sphere Collider, developers define how the controllers interact with objects in the virtual environment. This step lays the foundation for precise and intuitive user interactions.

GameManager Script:
The GameManager script serves as the backbone of the VR application, managing game logic and interactions. Through this script, developers establish the rules governing how the application responds to user inputs. In this experiment, the GameManager script orchestrates actions such as scoring points upon successful interactions, providing a dynamic and engaging user experience.

User Interface (UI) Design:
Creating an intuitive and visually appealing UI is crucial for a successful VR application. Within Unity's UI system, elements like buttons and images are arranged within a Canvas object. This canvas serves as a container for UI components, allowing developers to design a user-friendly interface that seamlessly integrates with the VR environment.

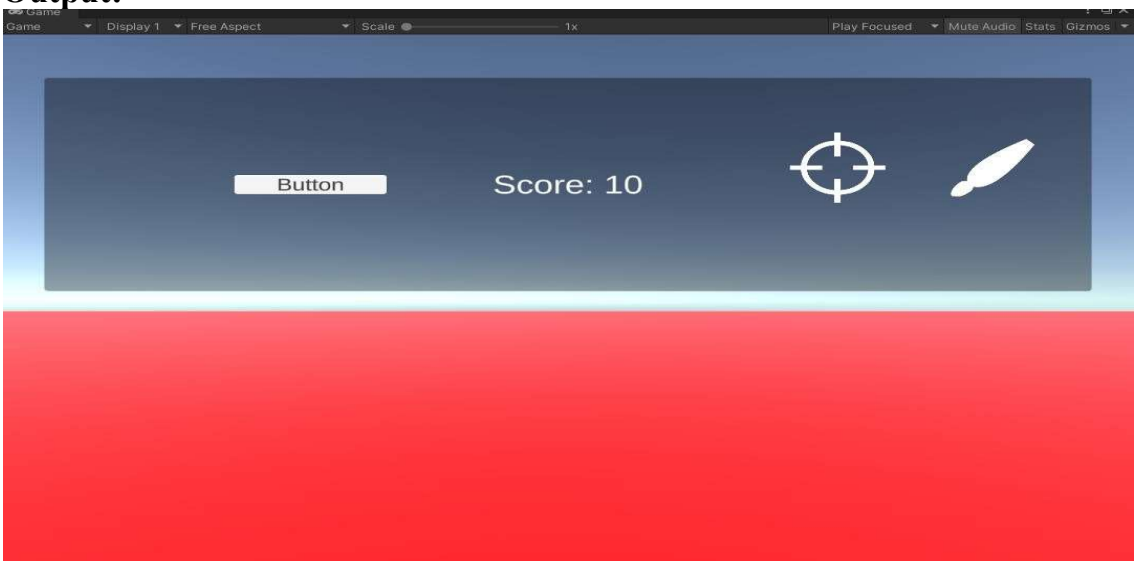
Interacting with UI Elements in VR:

Interacting with UI elements in a VR environment introduces unique challenges. The Tracked Graphic Raycaster is added to the Canvas object, enabling the detection of user interactions within the virtual space. By integrating the GameManager script, UI elements can trigger specific actions, providing users with a natural and immersive means of navigation and interaction.

Procedure:

1. Install XR Interaction Toolkit via Package Manager.
2. Configure XR Plugin Management for your VR platform.
3. Create an XR Origin for tracking and positioning.
4. Customize hand controllers with XR Direct Interactor and Sphere Collider.
5. Write a GameManager script for game logic.
6. Create UI elements like Canvas, Button, and Text.
7. Position and style UI elements.
8. Enable VR interaction with UI through ComponentEvent Trigger.
9. Implement a World Space Canvas for VR visibility.
10. Add Ray Interactors for UI interaction.
11. Play the game and use VR controllers to interact with UI elements while observing score updates on the Canvas.

Output:



Game
Game
Display 1
Free Aspect
Scale
1x
Play Focused
Mute Audio
Stats
Gizmos
Button
Score: 0

Conclusion: