**Entwicklerdokumentation**

***Kamera***

Die Kamera wurde so angepasst, dass sie mit WASD gesteuert werden kann. Dabei geht sie nach forward/backward/left/right. Mit dem Mausrad kann man jeweils mit Y nach unten oder nach oben gehen. Mit der rechten Maustaste kann man sich mit einer Kamera rotation in der Welt umsehen.

***UI***

Es wurde ein unteres UI hinzugefügt. Dieses befindet sich im HUD und heißt Inventory. Das Inventory besitzt eine Horizontal Layout group , die childs (slots) sind im Modus „middle Center“ dargestellt. Diese Horizontal Layout group ordnet alle slots automatisch an. Die slots sind die jeweiligen Items in der unteren Leiste. Sie beinhalten mehrere ineinander geschachtelte Images. Eines sorgt dafür, dass ein Randeffekt entsteht. Dieses Image besitzt auch einen Button. Wenn dieser Button später gedrückt oder drübergehovered wird, wird eine kleine „blauanimation“ abgespielt.

Auf dieses Image kam ein weiteres Image, welches etwas kleiner ist und das jeweilige Bild enthält, welches letztendlich unten angezeigt werden soll. Das Bild muss ein 2D sprite sein.

Der Hintergrund von dem Bottom UI ist ein ganz normales Panel in schwarz mit 50% alpha kanal (Inventory). Der Canvas heißt „HUD“.

Neben dem Bottom UI wurde ein weiteres Panel hinzugefügt, welches dem Nutzer gelegentlich messages überbringen soll. Dieses enthält ein entsprechendes TMP (Textmesh Pro) Object, welches den eigentlichen Text halten soll.

***ItemClickHandler***

Jedes Border Object enthält ein Script namens „ItemClickHandler“. Dieses enthält die Methode „OnButtonClicked“. Wenn man nun auf einen Button drückt, wird ein entsprechender Integerwert an diese Methode übergeben und kann dort verarbeitet werden.

In Update wird geprüft ob die entsprechende Zahl (1-9+0) gedrückt wurde. Die entsprechenden Keycodes wurden vorher bei den Border Objekten festlgelegt. Wenn das der Fall sein sollte, wird die selbstgeschriebene Methode „FadeToColor“ aufgerufen, welche eine Farbe entgegen nimmt und mit der Unity eigenen Methode „CrossFadeColor“ über die „Graphic“ (vorher über this geholt), die Farbe des Buttons ändert. Sobald man nun also den Button runter drückt, wird die Farbe geändert und der Button getriggert. Wenn man die Zahl auf der Tastatur wieder los lässt, wird die Farbe wieder zu der normal Color zurückgesetzt.

***Radial Menu [TODO]***

Radial Menu ist für die verschiedenen Buttons verantwortlich. Jeder button ist für sein eigenes aussehen verantwortlich (color, sprite).