



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΕΛΟΠΟΝΝΗΣΟΥ
UNIVERSITY *of the* PELOPONNESE

Π.Μ.Σ. στην Επιστήμη Υπολογιστών

Μεταπτυχιακή Διπλωματική Εργασία

Σχεδίαση και Υλοποίηση 3D ηλεκτρονικού παιχνιδιού ενός παίκτη, κατηγορίας περιπέτειας-τρόμου

Κυριαζή Στέλλα - Σαρέλα Δήμητρα
2022202202009 - 2022202202016

Επιβλέποντες: Λέπουρας Γεώργιος - Καθηγητής
Πλατής Νικόλαος - Επίκουρος Καθηγητής

Τρίπολη, Μάιος 2024



Abstract

This thesis analyses the main categories of computer games, with a greater emphasis on adventure and horror games. It presents the basic design concept of a first-person 3D adventure game with horror elements, and analyses the story, characters and main objectives of the game. It then details the steps of implementation of this idea, through the powerful game creation tool Unreal Engine, combining technical expertise and artistic vision. Finally, the challenges encountered during the implementation are highlighted, as well as the solutions applied to overcome them. This thesis presents a vision of future improvements to our game and explores possible routes for its evolution.

Περίληψη

Η παρούσα διπλωματική εργασία επικεντρώνεται στην ανάλυση των βασικών κατηγοριών ηλεκτρονικών παιχνιδιών, με μεγαλύτερη έμφαση στα παιχνίδια περιπέτειας και τρόμου. Παρουσιάζεται η βασική ιδέα σχεδιασμού ενός first-person 3D παιχνιδιού περιπέτειας με στοιχεία horror, και αναλύονται η ιστορία, οι χαρακτήρες και οι κύριοι στόχοι του παιχνιδιού. Στη συνέχεια, περιγράφονται λεπτομερώς τα βήματα υλοποίησης αυτής της ιδέας μέσω του ισχυρού εργαλείου δημιουργίας παιχνιδιών Unreal Engine, συνδυάζοντας την τεχνική εξειδίκευση και το καλλιτεχνικό όραμα. Τέλος, επισημαίνονται οι προκλήσεις που προέκυψαν κατά την υλοποίηση, καθώς και οι λύσεις που εφαρμόστηκαν για την αντιμετώπισή τους. Η παρούσα διατριβή οραματίζεται μελλοντικές βελτιώσεις του παιχνιδιού μας και διερευνά πιθανούς δρόμους για την εξέλιξή του.

Ευχαριστίες

Εκφράζουμε την ειλικρινή εκτίμησή μας στους επιβλέποντες της διατριβής μας, κ. Λέπουρα Γεώργιο και κ. Πλατή Νικόλαο, για την καθοδήγηση και την υποστήριξή τους καθ' όλη τη διάρκεια αυτής της προσπάθειας. Είμαστε επίσης ευγνώμονες στα μέλη της επιτροπής μας για τις πολύτιμες γνώσεις και επισημάνσεις τους.

Θα θέλαμε επιπλέον, να ευχαριστήσουμε τον συνάδελφό μας, κ. Κυριαζή Άγγελο, για τη βοήθεια και την ενθάρρυνσή του, καθώς επίσης τους φίλους και τις οικογένειές μας για την αμέριστη υποστήριξή τους.

Τέλος, αναγνωρίζουμε τη συμβολή όλων εκείνων που επηρέασαν στην επιλογή του προσανατολισμού αυτής της διατριβής και συνέβαλαν στην ανάπτυξή της.

Πίνακας Περιεχομένων

1. Εισαγωγή	15
2. Εισαγωγή στο Gaming	16
2. 1. Διαφοροποίηση στις κατηγορίες παιχνιδιών	16
2. 2. Ιστορική Εξέλιξη των παιχνιδιών (gaming)	18
2. 3. Παιχνίδια Περιπέτειας	19
Χαρακτηριστικά, Είδη και Εξέλιξη	19
Η σημασία της Ιστορίας (Story)	23
2. 4. Horror Games	24
Είδη και χαρακτηριστικά horror	25
User Experience	26
2. 5. Κοινωνικοποίηση μέσω του Gaming	28
3. Σχεδιασμός	29
3. 1. Ιστορία - Εκκίνηση Παιχνιδιού - Τέλος	29
3. 2. Χάρτης Παιχνιδιού	30
3. 3. Γρίφοι	31
3. 4. Χαρακτήρες παιχνιδιού - Backstories	33
3. 5. Χειρισμός Παίκτη	35
4. Υλοποίηση	36
4.1. Unreal Engine 5	36
Unreal Vs Unity	36
4.2. Περιβάλλον-Σχεδιασμός levels	37
Island level	37
Dungeon Level	40
Metahumans	42
4.3. Blueprints	43
Level Blueprints	43
Blueprint Class Actor	44
Widget Blueprints	56
Blueprints Χαρακτήρων	84
4.4. Cinematics	91
Sequences	91
5. Προκλήσεις-Μελλοντικές Βελτιστοποιήσεις	104
Προκλήσεις	104
Χρόνος	104
Μέγεθος Project	104
Cinematics Live Rendering	105
Επικοινωνία Levels - Widgets	105
Μελλοντικές Βελτιώσεις	106
6. Συμπεράσματα	107
7. Βιβλιογραφία	109

Λίστα Εικόνων

- [Εικόνα 1. Λίστα Δωματίων-Υποδοματίων παιχνιδιού](#)
- [Εικόνα 2. Χάρτης Παιχνιδιού](#)
- [Εικόνα 3. Μονοπάτι νησιού](#)
- [Εικόνα 4. Ηλιοβασίλεμα στο νησί](#)
- [Εικόνα 5. Η παραλία του νησιού](#)
- [Εικόνα 6. Τρισδιάστατο Μοντέλο καλύβας](#)
- [Εικόνα 7. Τελικό αποτέλεσμα καλύβας](#)
- [Εικόνα 8. Εσωτερικό καλύβας](#)
- [Εικόνα 9. Εσωτερικό καλύβας άλλη οπτική γωνία](#)
- [Εικόνα 10. Διάδρομος μπουντρουμιών](#)
- [Εικόνα 11. Great Hall](#)
- [Εικόνα 12. Δωμάτιο Βασανιστηρίων](#)
- [Εικόνα 13. Metahumans](#)
- [Εικόνα 14. MainMenuTest Event Graph](#)
- [Εικόνα 15. DungeonNew Event Graph](#)
- [Εικόνα 16. Ρύθμιση φακού στο First Person](#)
- [Εικόνα 17. Mapping First Person “F”](#)
- [Εικόνα 18. Flashlight Event Graph](#)
- [Εικόνα 19. Viewport Doors](#)
- [Εικόνα 20. Unlocked Door Event Graph \(Collision and Visibillity\)](#)
- [Εικόνα 21. Unlocked Door Event Graph \(Player Interaction\)](#)
- [Εικόνα 22. DoorKey](#)
- [Εικόνα 23. DoorKey Event Graph](#)
- [Εικόνα 24. DoorCode Event Graph \(Interaction\)](#)
- [Εικόνα 25. DoorCode Event Graph \(Open the Door Event\)](#)
- [Εικόνα 26. Viewport DoubleDoor](#)
- [Εικόνα 27. DoubleDoor Event Graph](#)
- [Εικόνα 28. DoubleDoor Τελικό αποτέλεσμα](#)
- [Εικόνα 29. Viewport Desk](#)
- [Εικόνα 30. Desk Event Graph \(Collision\)](#)
- [Εικόνα 31. Desk Event Graph \(Interaction\)](#)
- [Εικόνα 32. 3D Key Model](#)
- [Εικόνα 33. 3D key Model with Rust](#)
- [Εικόνα 34. Key Event Graph \(Interaction\)](#)
- [Εικόνα 35. Desk and Key Τελικό αποτέλεσμα](#)
- [Εικόνα 36. PhotoFrame Τελικό αποτέλεσμα](#)
- [Εικόνα 37. Folder Τελικό αποτέλεσμα](#)
- [Εικόνα 38. Letter Τελικό αποτέλεσμα](#)
- [Εικόνα 39. Skull Event Graph \(Interaction\)](#)
- [Εικόνα 40. Skull Τελικό αποτέλεσμα](#)
- [Εικόνα 41. Main Menu Widget Designer](#)
- [Εικόνα 42. Main Menu Widget Event Graph \(Button_Play\)](#)
- [Εικόνα 43. Main Menu Widget Event Graph \(Button_Quit\)](#)
- [Εικόνα 44. Main Menu Widget Event Graph \(Button_Options\)](#)
- [Εικόνα 45. Main Menu Widget Event Graph \(Mouse Movement /Load Settings\)](#)
- [Εικόνα 46. Main Menu Τελικό αποτέλεσμα](#)
- [Εικόνα 47. Option Menu Designer](#)

- [Εικόνα 48. Option Menu \(Variables/Functions\)](#)
- [Εικόνα 49. Function Binding GetText](#)
- [Εικόνα 50. Option Menu Event Graph \(Window Mode+/-\)](#)
- [Εικόνα 51. Option Menu Event Graph \(Graphics+/-\)](#)
- [Εικόνα 52. Option Menu Event Graph \(Apply Button\)](#)
- [Εικόνα 53. Option Menu Event Graph \(Apply settings\)](#)
- [Εικόνα 54. Option Menu Event Graph \(Back Button\)](#)
- [Εικόνα 55. Option Menu Event Graph \(Function Get Text\)](#)
- [Εικόνα 56. Option Menu Event Graph \(Function Get Text_0\)](#)
- [Εικόνα 57. Option Menu Τελικό αποτέλεσμα](#)
- [Εικόνα 58. Pause Menu Designer](#)
- [Εικόνα 59. Pause Menu Event Graph \(Mouse Movement\)](#)
- [Εικόνα 60. Pause Menu Event Graph \(Button Resume\)](#)
- [Εικόνα 61. Pause Menu Event Graph \(Button Quit\)](#)
- [Εικόνα 62. Mapping First Person “Tab”](#)
- [Εικόνα 63. Pause Menu in First Person Event Graph](#)
- [Εικόνα 64. Pause Menu Τελικό αποτέλεσμα](#)
- [Εικόνα 65. Photo Designer](#)
- [Εικόνα 66. Photo Event Graph \(Button Flip\)](#)
- [Εικόνα 67. Photo Event Graph \(Mouse movement\)](#)
- [Εικόνα 68. Photo in First Person Event Graph](#)
- [Εικόνα 69. Photo Back Designer](#)
- [Εικόνα 70. Photo Back Event Graph \(Button Remember\)](#)
- [Εικόνα 71. Photo Back Event Graph \(Mouse Movement\)](#)
- [Εικόνα 72. Photo και Photo Back Τελικό αποτέλεσμα](#)
- [Εικόνα 73. Open Folder Designer](#)
- [Εικόνα 74. Open Folder Event Graph \(Exchange widgets\)](#)
- [Εικόνα 75. John Trueman Designer](#)
- [Εικόνα 76. John Trueman Event Graph \(Status Coloring\)](#)
- [Εικόνα 77. John Trueman Event Graph \(Mouse Movement/Audio\)](#)
- [Εικόνα 78. All Sailors Designer](#)
- [Εικόνα 79. All Sailors Event Graph \(Mouse Movement/Audio\)](#)
- [Εικόνα 80. Letter Designer](#)
- [Εικόνα 81. Letter Event Graph \(Button Drop\)](#)
- [Εικόνα 82. Letter Event Graph \(Audio/Drop Button Delay\)](#)
- [Εικόνα 83. Letter Τελικό αποτέλεσμα](#)
- [Εικόνα 84. Numpad Designer](#)
- [Εικόνα 85. Numpad Event Graph \(Buttons 0-9\)](#)
- [Εικόνα 86. Numpad Event Graph \(Button Clear\)](#)
- [Εικόνα 87. Numpad Event Graph \(Button Enter\)](#)
- [Εικόνα 88. Numpad Event Graph \(Button Exit\)](#)
- [Εικόνα 89. Numpad Τελικό αποτέλεσμα](#)
- [Εικόνα 90. Button Sequence Door Designer](#)
- [Εικόνα 91. Button Sequence Door Event Graph \(Button Left/Right\)](#)
- [Εικόνα 92. Button Sequence Door \(Print Sequence, Buttons Reset/Ok Visibility\)](#)
- [Εικόνα 93. Button Sequence Door \(Button Reset\)](#)
- [Εικόνα 94. Button Sequence Door \(Button Ok\)](#)
- [Εικόνα 95. Button Sequence Door \(Button Reset συνέχεια\)](#)
- [Εικόνα 96. Button Sequence Door \(Button Exit\)](#)
- [Εικόνα 97. Button Sequence Door Τελικό αποτέλεσμα](#)

- [Εικόνα 98. Inventory Designer](#)
- [Εικόνα 99. Inventory Event Graph \(Visibility of Letter\)](#)
- [Εικόνα 100. Inventory in First Person Event Graph](#)
- [Εικόνα 101. Inventory in First Person Event Graph συνέχεια](#)
- [Εικόνα 102. Inventory Τελικό αποτέλεσμα](#)
- [Εικόνα 103. Ending Choice Designer](#)
- [Εικόνα 104. Ending Choice Event Graph \(Audio\)](#)
- [Εικόνα 105. Ending Choice Event Graph \(Audio on hovered\)](#)
- [Εικόνα 106. Ending Choice Event Graph \(Buttons Felipe/Tobias Choice\)](#)
- [Εικόνα 107. Ending Choice Event Graph \(End Widget depend on Choice\)](#)
- [Εικόνα 108. Ending Choice in First Person Event Graph](#)
- [Εικόνα 109. End Felipe/Tobias Designer](#)
- [Εικόνα 110. End Felipe/Tobias Event Graph \(Video player\)](#)
- [Εικόνα 111. Viewport Felipe](#)
- [Εικόνα 112. Felipe Animation \(Body\)](#)
- [Εικόνα 113. Felipe Animation \(Face\)](#)
- [Εικόνα 114. Npc2look AnimGraph](#)
- [Εικόνα 115. Npc2look Event Graph](#)
- [Εικόνα 116. Look At node Bone Modifications \(Left/Right Eye\)](#)
- [Εικόνα 117. Face_AnimBP1FORNPCs AnimGraph](#)
- [Εικόνα 118. Face_AnimBP1FORNPCs Event Graph \(Find Location\)](#)
- [Εικόνα 119. Face_AnimBP1FORNPCs Event Graph \(Set Location\)](#)
- [Εικόνα 120. Dialogue Designer](#)
- [Εικόνα 121. Felipe Dialogue](#)
- [Εικόνα 122. Tobias Dialogue](#)
- [Εικόνα 123. Felipe/Tobias Event Graph \(Variables\)](#)
- [Εικόνα 124. Felipe Event Graph \(Event talk\)](#)
- [Εικόνα 125. Felipe Event Graph \(Text/Audio Dialogue Widget\)](#)
- [Εικόνα 126. Felipe in First Person Event Graph](#)
- [Εικόνα 127. Felipe/Tobias in First Person Event Graph \(Spoke with both check\)](#)
- [Εικόνα 128. Felipe/Tobias - Dialogue Τελικό αποτέλεσμα](#)
- [Εικόνα 129. Play.HT Online AI tool](#)
- [Εικόνα 130. Metahuman Performance](#)
- [Εικόνα 131. Mixamo Tool for Body Animations](#)
- [Εικόνα 132. Helicopter Opening Scene](#)
- [Εικόνα 133. Helicopter Landing](#)
- [Εικόνα 134. Interview Scene](#)
- [Εικόνα 135. Interview Scene Editing Process](#)
- [Εικόνα 136. Interview Scene Close up](#)
- [Εικόνα 137. Hill Walk Scene](#)
- [Εικόνα 138. Talk On The Beach Scene](#)
- [Εικόνα 139. Inside The Cabin Scene](#)
- [Εικόνα 140. Inside The Cabin Scene \(Tobias Camera\)](#)
- [Εικόνα 141. Ending Felipe Scene](#)
- [Εικόνα 142. Ending Felipe Scene \(Salvation\)](#)
- [Εικόνα 143. Ending Tobias Scene](#)
- [Εικόνα 144. Ending Tobias Scene \(Doom\)](#)
- [Εικόνα 145. Neon Sign Blender](#)
- [Εικόνα 146. Neon Sign Closing Scene](#)
- [Εικόνα 147. DaVinci Resolve Editing for Rendering](#)

Λίστα Πινάκων

- [Πίνακας 1. Είδη παιγνιδιών, χαρακτηριστικά και παραδείγματα](#)
- [Πίνακας 2. Χαρακτηριστικά Παιγνιδιών Περιπέτειας και Ανάλυση](#)
- [Πίνακας 3. . Στάδια φόβου, επεξήγηση και παραδείγματα](#)

1. Εισαγωγή

Η εμφάνιση όλο και πιο ισχυρών μηχανών ανάπτυξης παιχνιδιών έφερε επανάσταση στη βιομηχανία παιχνιδιών, επιτρέποντας στους δημιουργούς να ζωντανέψουν φανταστικούς κόσμους με πρωτοφανή επίπεδα λεπτομέρειας και διαδραστικότητας. Τα παιχνίδια περιπέτειας, με έμφαση στην εξερεύνηση, την επίλυση γρίφων και την εξέλιξη της ιστορίας, έχουν γοητεύσει τους παίκτες εδώ και δεκαετίες με τις συναρπαστικές ιστορίες και τους καθηλωτικούς κόσμους τους.

Κύριος στόχος σε αυτή τη διπλωματική εργασία, είναι η αξιοποίηση της δύναμης του εργαλείου Unreal Engine για τη δημιουργία ενός παιχνιδιού περιπέτειας, που όχι μόνο διασκεδάζει τον παίκτη, αλλά και διεγείρει την περιέργειά του μέσα από ελκυστικούς γρίφους, δυναμικό περιβάλλον και πλούσια αφήγηση.

Η παρούσα διπλωματική εργασία περιλαμβάνει τη διαδικασία πλήρους ανάπτυξης ενός παιχνιδιού, από το στάδιο σχεδιασμού έως και το στάδιο υλοποίησης, αξιοποιώντας τις δυνατότητες και τα εργαλεία που παρέχει το εν λόγω περιβάλλον του Unreal Engine. Εμβαθύνει σε διάφορα μέρη της ανάπτυξης του παιχνιδιού, συμπεριλαμβανομένου του σχεδιασμού επιπέδων, της δημιουργίας στοιχείων (assets), του σεναρίου και της βελτιστοποίησης. Ένας από τους στόχους είναι, εκτός άλλων, να παρέχει γνώσεις και πρακτικές λύσεις για τους επίδοξους προγραμματιστές παιχνιδιών.

Η εργασία οργανώνεται σε ενότητες, αρχικά με μια επισκόπηση του θεωρητικού πλαισίου και της βασικής προσέγγισης της διαδικασίας, και στη συνέχεια την πορεία της σταδιακής υλοποίησης. Τέλος, παρουσιάζονται οι προκλήσεις που αντιμετωπίστηκαν επιτυχώς, όπως επίσης και το περιθώριο μελλοντικής βελτιστοποίησης.

2. Εισαγωγή στο Gaming

Τα ηλεκτρονικά παιχνίδια αποτελούν την κύρια ψυχαγωγία, τόσο μικρών όσο και μεγάλων, της εποχής μας. Επειδή ακριβώς μεγάλος αριθμός παικτών αφιερώνουν ώρες από την καθημερινότητά τους για να απολαύσουν το ψυχαγωγικό αυτό χόμπι, τα παιχνίδια αποτελούν πλέον σημαντικό και αναπόσπαστο κομμάτι της ζωής (Qaffas, 2020). Αυτές οι τάσεις έχουν κεντρίσει το ενδιαφέρον τόσο επενδυτών όσο και ερευνητών όσων αφορά την ανάπτυξη και την προώθηση των ηλεκτρονικών παιχνιδιών αλλά και τον αντίκτυπο αυτών σε διάφορους τομείς, όπως η εκπαίδευση και η οικονομία.

Εκτός αυτών, τα ηλεκτρονικά παιχνίδια προσφέρουν πολλαπλά οφέλη στους παίκτες. Οι παίκτες αναπτύσσουν δεξιότητες όπως η συνεργασία, η λήψη αποφάσεων και η ενίσχυση της δημιουργικής σκέψης. Επιπλέον, τα παιχνίδια αναδεικνύουν τη δύναμη της διαδραστικής εμπειρίας και οδηγούν στην ανάπτυξη νέων δυνατοτήτων.

Ακόμη, η εξέλιξη των κονσολών έχει φέρει μια νέα διάσταση στον κόσμο του gaming. Οι σύγχρονες εκδόσεις έχουν ενισχύσει τα επίπεδα διασκέδασης, καθώς και τις απαιτήσεις του κοινού. Τέλος, με την εισαγωγή της εικονικής πραγματικότητας (VR) και την ανάπτυξη της διαδραστικής τεχνολογίας, οι παίκτες έχουν πλέον την ευκαιρία να ζήσουν μοναδικές εμπειρίες που προσφέρουν μία ολοκληρωμένη αίσθηση ρεαλισμού.

2. 1. Διαφοροποίηση στις κατηγορίες παιχνιδιών

Τα ηλεκτρονικά παιχνίδια καλύπτουν μεγάλο φάσμα προτιμήσεων, και λόγω αυτού υπάρχει συζήτηση ως προς τον τρόπο κατηγοριοποίησης τους (Qaffas, 2020). Παρακάτω παρουσιάζονται κάποιες βασικές κατηγορίες παιχνιδιών και παραδείγματα των πιο επιτυχημένων παραγωγών των τελευταίων δεκαετιών.

- **Περιπέτεια (Adventure):** Παιχνίδια με έμφαση στην ανακάλυψη, την εξερεύνηση και την αφήγηση ιστοριών. Οι παίκτες συνήθως λύνουν γρίφους και ακολουθούν μια περιπέτεια μέσα σε ένα φανταστικό κόσμο.
- **Παιχνίδια Ρόλων (Role-playing Game - RPG):** Παιχνίδια όπου οι παίκτες αναλαμβάνουν το ρόλο ενός χαρακτήρα και εξερευνούν έναν φανταστικό κόσμο, επιτρέποντας την προσαρμογή των χαρακτήρων και την ανάπτυξη των δεξιοτήτων τους.
- **Σκοπευτές Πρώτου Προσώπου (First-person Shooters - FPS):** Παιχνίδια όπου ο παίκτης βλέπει τη δράση μέσα από τα μάτια του χαρακτήρα του και κυνηγάει εχθρούς με όπλα.
- **Παιχνίδια Πλατφόρμας (Platformers):** Παιχνίδια όπου οι παίκτες πρέπει να περάσουν διάφορα εμπόδια και να φτάσουν σε διάφορα επίπεδα χρησιμοποιώντας κυρίως τις δεξιότητες του άλματος.
- **Παιχνίδια Γρίφων (Puzzle Games):** Παιχνίδια όπου οι παίκτες πρέπει να λύσουν διάφορους γρίφους ή παζλ για να προχωρήσουν στο παιχνίδι.
- **Στρατηγικής (Strategy Games):** Παιχνίδια όπου οι παίκτες λαμβάνουν αποφάσεις για την κατεύθυνση και τις ενέργειες των χαρακτήρων ή των στρατευμάτων τους με σκοπό τη νίκη σε έναν φανταστικό ή πραγματικό κόσμο.
- **Αγωνιστικά (Racing Games):** Παιχνίδια όπου οι παίκτες ανταγωνίζονται μεταξύ τους σε αγώνες οχημάτων.

- **Αθλητικά (Sports Games):** Παιχνίδια που προσομοιώνουν διάφορα αθλήματα, επιτρέποντας στους παίκτες να αναλάβουν τον έλεγχο των αθλητών τους.

Στον πίνακα παρακάτω (βλ. [Πίνακας 1.](#)) παρατίθενται κάποια είδη με τα χαρακτηριστικά τους και κάποια παραδείγματα παιχνιδιών ανά είδος.

Είδος	Χαρακτηριστικά	Παραδείγματα
Arcade & Action	Προσανατολισμός στη δράση, συντονισμός χεριών-ματιών	Doom, Mortal Kombat, Pacman
Adventure	Ο παίκτης αλληλεπιδρά με τους χαρακτήρες και τα αντικείμενα του παιχνιδιού για να ολοκληρώσει τις εργασίες του, προσπαθώντας να πετύχει έναν τελικό στόχο ("quest"). Συνήθως περιλαμβάνει μια καλή πλοκή, ενδιαφέροντες χαρακτήρες και ένα φανταστικό σκηνικό. Ο παίκτης πρέπει να προσαρμοστεί στους κανόνες και τη νοοτροπία του νέου κόσμου. η απόκτηση και η χρήση πόρων είναι σημαντικές.	King's Quest, Monkey Island, Full Throttle
Computer RPG	Ηλεκτρονικές εκδόσεις των παιχνιδιών ρόλων. Η ιστορία είναι επαναλαμβανόμενη και δευτερεύουσα σε σχέση με τη μάχη και την ανάπτυξη των χαρακτήρων.	Dungeons and Dragons, Wizardry, Ultima
Simulation (Sim)	Τα παιχνίδια μοντελοποιούν αντικείμενα του πραγματικού κόσμου, όπως μαχητικά αεροσκάφη, και επιτρέπουν στους παίκτες να προσομοιώσουν τη χρήση τους.	Falcon, Fleet Defender, Mig 29
Building Sim	Ο παίκτης είναι υπεύθυνος για την οικοδόμηση και τη διαχείριση μιας κοινωνίας, όπως μια πόλη ή μια φάρμα. Περιλαμβάνουν επίσης τα " God games ", όπου ο παίκτης έχει ικανότητες που προσομοιάζουν αυτές του Θεού.	SimCity, SimAnt, Civilization, Populus
Strategy	Ανταγωνιστικά παιχνίδια στα οποία ο παίκτης πρέπει να αναπτύξει σχέδια για να ξεγελάσει έναν αντίπαλο ή τον υπολογιστή.	Command and Conquest, Warcraft I & II, Warlords II
War	Παιχνίδια στρατηγικής turn-based, τα οποία συνήθως αναπαριστούν μια ιστορική σύγκρουση. Ωμές προσεγγίσεις της πραγματικότητας.	Panzer General, Koei's Chinese Saga
Sports	Hybrid μεταξύ arcade και simulation, που απαιτούν συντονισμό χεριού-ματιού, αλλά αναπαριστούν πραγματικούς αθλητικούς χώρους ή μοντελοποιούν πραγματικούς	FIFA Soccer '96, Virtual Pool, ESPN Extreme Sports

	(διάσημους) αθλητές.	
Puzzles	Εφαρμογές μαθηματικών ή λογικών γρίφων σε υπολογιστή. Συχνά ενσωματωμένες σε άλλους τύπους παιχνιδιών.	Puzzle components in Seventh Guest
Educational	Σχεδιασμένα για να διδάξουν στον παίκτη ένα σύνολο εκπαιδευτικών δεξιοτήτων, όπως μαθηματικά, ανάγνωση, δακτυλογράφηση ή μια ξένη γλώσσα. Τα παιχνίδια έχουν διάφορες μορφές, όπως επίλυση γρίφων ή arcade.	Carmen San Diego, Mario Teaches Typing

Πίνακας 1. Είδη παιχνιδιών, χαρακτηριστικά και παραδείγματα

Καθώς οι παίκτες εξερευνούν αυτές τις κατηγορίες παιχνιδιών, επιλέγουν ποια εμπειρία ταιριάζει καλύτερα στα προσωπικά τους ενδιαφέροντα και προτιμήσεις. Από τις μεγάλες ανοιχτές περιπέτειες με συναρπαστικές ιστορίες, μέχρι τις ευφάνταστες λύσεις γρίφων και τις ατελείωτες αγωνιώδεις κόντρες, το gaming προσφέρει μια ποικιλία εμπειριών που καλύπτει κάθε είδος παίκτη. Με τη συνεχή εξέλιξη της τεχνολογίας, οι παίκτες έχουν πάντα κάτι νέο να ανακαλύψουν και να απολαύσουν στον κόσμο των video games.

2. 2. Ιστορική Εξέλιξη των παιχνιδιών (gaming)

Η εξέλιξη του gaming από τα πρώτα παιχνίδια έως σήμερα αποτελεί μια συναρπαστική πορεία που αντανακλά την τεχνολογική πρόοδο και τις κοινωνικές αλλαγές. Από τα απλά pixelated παιχνίδια των δεκαετιών του '70 και του '80, μέχρι τα υπερ-ρεαλιστικά γραφικά και τις εντυπωσιακές εμπειρίες εικονικής πραγματικότητας στις μέρες μας, το gaming έχει διαμορφώσει έναν πλούσιο και πολυδιάστατο κόσμο.

Η εξέλιξη του gaming αντικατοπτρίζει επίσης τις κοινωνικές τάσεις και τις προτιμήσεις των παικτών (Valentine & Jensen, 2016). Από την αρχή, τα παιχνίδια ήταν κυρίως μια ατομική εμπειρία, αλλά σήμερα πολλά παιχνίδια προσφέρουν multiplayer λειτουργίες, που επιτρέπουν στους παίκτες να αλληλεπιδρούν και να ανταγωνίζονται μεταξύ τους σε πραγματικό χρόνο. Η δημιουργία online κοινοτήτων γύρω από παιχνίδια έχει επίσης αλλάξει τον τρόπο που οι άνθρωποι αντιλαμβάνονται την κοινωνικοποίηση και την επικοινωνία στον ψηφιακό κόσμο.

Η εξέλιξη του gaming εξαρτάται άμεσα από την εξέλιξη του εξοπλισμού που χρησιμοποιείται, ο οποίος έχει βιώσει μια εντυπωσιακή πορεία τα τελευταία χρόνια. Από τις απλές κονσόλες και τα φορητά παιχνίδια του παρελθόντος, έχουμε μεταβεί σε πολυσύνθετα gaming PCs και εξελιγμένα gaming laptops που επιτρέπουν τη λειτουργία παιχνιδιών σε υψηλή ανάλυση και μεγάλη ταχύτητα. Οι κονσόλες παιχνιδιών έχουν επίσης γνωρίσει σημαντική εξέλιξη, με τις τελευταίες γενιές να προσφέρουν υψηλή απόδοση γραφικών και εκτεταμένες δυνατότητες δικτύωσης για online παιχνίδια. Πέραν των κονσολών και των PCs, έχουμε δει επίσης την εμφάνιση νέων ειδών εξοπλισμού για το gaming, όπως τα VR (Virtual Reality) συστήματα και τα παιχνίδια, που προσφέρουν στους παίκτες μια πλήρη εμπειρία εμβάθυνσης με εντυπωσιακά εικονικά περιβάλλοντα. Επιπλέον, οι ειδικοί gaming περιφερειακοί εξοπλισμοί, όπως gaming ποντίκια, πληκτρολόγια και ακουστικά, έχουν σχεδιαστεί με γνώμονα τις

ανάγκες των παικτών, προσφέροντας βελτιωμένη απόδοση και άνεση κατά τη διάρκεια του gaming.

Εκτός αυτών, μια μεγάλη μετάβαση έχει παρατηρηθεί στις ίδιες τις κατηγορίες παιχνιδιών. Κάποτε, τα παιχνίδια περιορίζονταν σε γραμμική πλοκή, προσφέροντας λιγοστές επιλογές και περιορισμένο πλαίσιο αναζήτησης. Πλέον, με την εμφάνιση των Open World παιχνιδιών οι παίκτες καλούνται να εξερευνήσουν εκτενείς εικονικούς κόσμους και να πλοηγηθούν πέραν της βασικής ιστορίας, ολοκληρώνοντας επιπλέον αποστολές. Ακόμη, η φύση των αποστολών και του χειρισμού, έχουν εξελιχθεί σε πιο πολύπλοκες και απαιτητικές διαδικασίες. Οι λειτουργίες του ίδιου του παιχνιδιού, έγιναν πιο περίτεχνες και ολοκληρωμένες, και η διεπαφή του χρήστη πιο ρεαλιστική.

Με τη συνεχή εξέλιξη της τεχνολογίας, την αυξανόμενη δημιουργικότητα των developers και τη συνεχή βελτίωση του gaming εξοπλισμού, το μέλλον του gaming φαίνεται λαμπρό και γεμάτο υποσχέσεις προσφέροντας στους παίκτες ακόμη πιο εντυπωσιακές και εκτεταμένες εμπειρίες.

2.3. Παιχνίδια Περιπέτειας

Τα παιχνίδια περιπέτειας είναι ένα δημοφιλές είδος παιχνιδιού εδώ και δεκαετίες. Αγαπήθηκαν από τους παίκτες για το μοναδικό συνδυασμό της αφήγησης ιστοριών, της επίλυσης γρίφων και της εξερεύνησης. Συχνά προσφέρουν μια πιο ελκυστική εμπειρία από άλλα είδη, καθιστώντας τα ιδανικά για όσους απολαμβάνουν την εξερεύνηση νέων κόσμων (Fernandez-Vara & Osterwil, 2010).

Τα παιχνίδια περιπέτειας δίνουν συνήθως έμφαση στην εξερεύνηση και την επίλυση γρίφων έναντι της δράσης. Λαμβάνουν χώρα σε φανταστικούς κόσμους ή σκηνικά και ο παίκτης είναι συνήθως ο πρωταγωνιστής, ο οποίος πρέπει να ξεπεράσει διάφορα εμπόδια για να προχωρήσει παρακάτω στην ιστορία. Δίνουν μεγάλη έμφαση στην αφήγηση, προσφέροντας στον παίκτη μια καθηλωτική και συναρπαστική ιστορία που πρέπει να ακολουθήσει. Οι παίκτες πρέπει συνήθως να συλλέγουν ή να κατασκευάζουν αντικείμενα που χρησιμοποιούνται μέσα στο παιχνίδι ή που είναι απαραίτητα για την έκβαση του παιχνιδιού.

Χαρακτηριστικά, Είδη και Εξέλιξη

Τα παιχνίδια περιπέτειας συνήθως διαθέτουν χαρακτηριστικά που τα διαφοροποιούν από άλλες κατηγορίες παιχνιδιών. Παρακάτω παρουσιάζονται κάποια από τα σημαντικά χαρακτηριστικά της αρχιτεκτονικής αυτών των παιχνιδιών που μπορεί να χρησιμεύσουν ως την αρχή για το σχεδιασμό τέτοιων παιχνιδιών (Ju & Wagner, 1997). Τα παιχνίδια αυτά, αποτελούνται από τρία βασικά δομικά στοιχεία: την ιστορία, την εξέλιξη και την υλοποίηση.

- Η Ιστορία (**Story**) (καθώς και η Μεταφορά, η Αναλογία και η "πλοκή") καθορίζει τα κύρια χαρακτηριστικά της υπόθεσης (π.χ. το θέμα), τους χαρακτήρες, τα αντικείμενα (πόρους), τα καθήκοντα που πρέπει να επιτελεστούν, καθώς και τον τόπο και τον χρόνο στον οποίο αυτή διαδραματίζεται.
- Η εξελικτική προσέγγιση (**Development**) αναφέρεται στην προσέγγιση του προγραμματιστή του παιχνιδιού ως προς τον τρόπο με τον οποίο θα παρουσιαστούν τα οπτικά στοιχεία του παιχνιδιού από πλευράς των χαρακτήρων και του περιβάλλοντος.

- Η υλοποίηση (**Implementation**) αναφέρεται στις τεχνικές λεπτομέρειες για την ανάπτυξη των μηχανισμών του παιχνιδιού.

Χαρακτηριστικά	
Story Characteristics	<ul style="list-style-type: none"> • Θέμα • Είδος (π.χ. κωμωδία) • Επίλυση της πλοκής (π.χ., " διαστρεβλωμένη πλοκή") • Αφηγηματική ταχύτητα (π.χ., συνοπτική και γρήγορη κίνηση, μακροσκελής και λεπτομερής) • Φαντασία και ποικιλομορφία • Παιχνίδι ρόλων
Actors and their characteristics	<ul style="list-style-type: none"> • Κατηγορία (παίκτης, alter ego, non-player) • Ταυτότητα (όνομα, περιγραφή υπόβαθρου κ.λπ.) • Επίπεδο νοημοσύνης (βουβό-ντετερμινιστικό, πιθανολογικό, μαθησιακό) • Ρόλος (εκπαιδευτής, απειλή, πάροχος πόρων) • Κινητικότητα (μετακινείται, σταθερή θέση στο χρόνο, τόπος) • Απαιτήση ύπαρξης (πρέπει να ζήσει, πρέπει να πεθάνει)
Resources	<ul style="list-style-type: none"> • Πόροι • Αξία (τιμή, κόστος, σπανιότητα) • Αναλωσιμότητα (αναλώσιμο, μόνιμο, ανανεώσιμο) • Απαιτήση κατοχής • Άλλες προδιαγραφές (βάρος, μέγεθος κ.λπ.)
Tasks	<ul style="list-style-type: none"> • Στόχοι εργασιών (ανώτερο επίπεδο, σκηνικό επίπεδο, χαμηλό επίπεδο) • Συνθήκες και δυσκολία εκτέλεσης της αποστολής • Χρονοδιάγραμμα (πότε πρέπει να ολοκληρωθεί;) • Απαιτήσεις πόρων (τι πόρους πρέπει να διαθέτουμε;) • Υποεργασίες (η αποστολή αποτελείται από;..) • Φορέας (ποιος πρέπει να εκτελέσει το έργο;)
Setting	<ul style="list-style-type: none"> • Στοιχεία • Χρόνος (παρελθόν, παρόν, μέλλον, μείγμα αυτών) • Τοποθεσία • "Δωμάτια" • Γενική περιοχή

Πίνακας 2. Χαρακτηριστικά Παιχνιδιών Περιπέτειας και Ανάλυση

Η ιστορία (**Story**) ενός παιχνιδιού περιπέτειας διαμορφώνεται από τα ακόλουθα στοιχεία: κύρια χαρακτηριστικά, πρωταγωνιστές, πόροι, στόχοι (καθήκοντα) και περιβάλλοντα (βλ. [Πίνακας 2](#)).

Στα χαρακτηριστικά της ιστορίας περιλαμβάνονται το θέμα ("topic") της ιστορίας, το είδος (πολλά είναι κωμωδίες, ωστόσο ορισμένα παιχνίδια περιπέτειας είναι πιο

σκοτεινού χαρακτήρα) και διάφορες πτυχές που σχετίζονται με το στυλ. Για παράδειγμα, τα παιχνίδια μπορεί να διαφέρουν ως προς τον τρόπο με τον οποίο εκτυλίσσεται η πλοκή και την ταχύτητα με την οποία εξιστορείται. Σημαντικά για την ιστορία στοιχεία είναι επίσης η φαντασία και η ποικιλομορφία, καθώς και το στοιχείο του παιχνιδιού ρόλων. Μια πλούσια ιστορία έχει πολλαπλά μονοπάτια με τα οποία μπορούν να επιτευχθούν οι στόχοι (όχι μόνο μια "σωστή" λύση) που συχνά εκπλήσσει τον παίκτη. Το επίπεδο του παιχνιδιού ρόλων προσδιορίζει πόσο μεγάλη επιρροή μπορεί να ασκήσει ο παίκτης στην έκβαση των γεγονότων.

Οι φορείς (χαρακτήρες) μπορεί να είναι δύο τύπων: χαρακτήρες παίκτη ("**player**") και μη-παίκτες ("**non-player characters**"). Συχνά όμως, μόνο ένας χαρακτήρας χρησιμεύει ως εκπρόσωπος του παίκτη στο παιχνίδι. Οι μη παίκτες εμφανίζονται στον κόσμο του παιχνιδιού για να εμπλουτίσουν την εμπειρία. Μπορεί να είναι φιλικόι σύμμαχοι, ουδέτερα μέρη ή εχθρικές οντότητες. Προστίθενται για να παρέχουν πόρους, να βοηθούν τον χαρακτήρα-παίκτη στην απόκτηση ειδικών δεξιοτήτων, να δίνουν συμβουλές ή να χρησιμεύουν ως εκπαιδευτές για να βοηθήσουν τον παίκτη να αναπτύξει δεξιότητες επίλυσης προβλημάτων. Ένας ειδικός τύπος μη παίκτη έχει ως πρωταρχικό στόχο να βλάψει τον χαρακτήρα-παίκτη. Αυτοί οι χαρακτήρες προσθέτουν κίνδυνο και ενθουσιασμό στο παιχνίδι και βελτιώνουν τις δεξιότητες επίλυσης προβλημάτων του παίκτη καθώς το παιχνίδι εξελίσσεται.

Οι πόροι στα παιχνίδια περιπέτειας μπορεί να είναι κοινά αντικείμενα (π.χ. φακός) ή πολύτιμα και μοναδικά αντικείμενα (π.χ. χρυσός, "σκήπτρο της δύναμης"). Μπορεί να είναι αναλώσιμοι ή μη αναλώσιμοι (μόνιμοι). Μπορούν επίσης να είναι άυλοι, όπως ειδικές δεξιότητες (π.χ. σκοποβολή, κλοπή) που πρέπει να μάθει ο χαρακτήρας-παίκτη για να φέρει εις πέρας μια αποστολή. Συχνά, οι πόροι συνδέονται με ένα σκορ, το οποίο υποδηλώνει την αξία, τη σπανιότητα, τον αντίκτυπο ή την πρόοδο του παιχνιδιού. Για ορισμένους πόρους μπορεί να υπάρχει απαίτηση κατοχής, όπως για να ολοκληρωθεί μια εργασία (π.χ. κλειδί για να ξεκλειδωθεί μια πόρτα).

Οι στόχοι ή καθήκοντα (**Tasks**) είναι οι ενέργειες που πρέπει οι παίκτες να ολοκληρώσουν για να μπορέσουν να προχωρήσουν στην ιστορία. Προηγούμενες έρευνες δείχνουν ότι οι στόχοι είναι σημαντικοί για την αρέσκεια των ανθρώπων στα παιχνίδια (Malone, 1981). Ο κορυφαίος στόχος ενός παιχνιδιού οδηγεί στην επιτυχή ολοκλήρωση του παιχνιδιού. Ένας στόχος μεσαίου επιπέδου (που αναφέρεται επίσης ως στόχος ενός κεφαλαίου ή στόχος μιας σκηνής) μπορεί να οδηγεί σε ένα cut scene animation, που υποδηλώνει την ολοκλήρωση μιας πράξης. Οι στόχοι χαμηλού επιπέδου διευκολύνουν την επίτευξη των στόχων κεφαλαίου και κορυφαίου επιπέδου και αποτελούν απαιτούμενα βήματα για τη συνέχιση της διαδικασίας, αλλά δεν περιλαμβάνουν cut scene animation ή άλλες ειδικές ανταμοιβές.

Τέλος, οι σχεδιαστές πρέπει να καθορίσουν τις ρυθμίσεις του παιχνιδιού, οι οποίες σε συνδυασμό με το ρολόι και τις επιλογές κίνησης του παιχνιδιού δημιουργούν μια σωστή αίσθηση του χρόνου και της τοποθεσίας. Αφού αποφασιστεί η γενική τοποθεσία (περιοχή), είναι επίσης απαραίτητο να καθοριστούν τα δωμάτια (θέσεις σκηνών) που είναι ανοικτά για εξερεύνηση από τον παίκτη.

Η εξελικτική προσέγγιση (**Development**) καθορίζει τη μορφή παρουσίασης ενός παιχνιδιού. Επικεντρώνεται σε δύο τομείς: τους χαρακτήρες (ηθοποιούς) και το περιβάλλον (σκηνικά/φόντο). Ο προγραμματιστής πρέπει να αποφασίσει αν θα τα δημιουργήσει εικονικά ή χρησιμοποιώντας πραγματικούς ηθοποιούς και σκηνικά. Για παράδειγμα, ιστορικά, τα παιχνίδια περιπέτειας γραφικών αναπτύχθηκαν με τη χρήση

χειροποίητων χαρακτήρων και φόντου. Αρκετά από τα νεότερα παιχνίδια είτε χρησιμοποιούν πραγματικούς ηθοποιούς και ψηφιοποιούν απλουστευμένες εκδοχές των εικόνων τους, είτε χρησιμοποιούν ακόμη και animations.

Τα ζητήματα υλοποίησης (**Implementation**) που αντιμετωπίζουν οι περισσότεροι προγραμματιστές παιχνιδιών περιλαμβάνουν τον έλεγχο του χρήστη, την παρουσίαση της εξόδου (βίντεο και ήχος), τη μέτρηση της επίτευξης στόχων, τον χρόνο και την αβεβαιότητα.

Μεταξύ των στοιχείων εισόδου, ένα βασικό ζήτημα είναι η επιλογή της συσκευής ή των συσκευών. Τα πρώτα παιχνίδια περιπέτειας καθορίζονταν αυστηρά από το πληκτρολόγιο, πράγμα που ήταν κατάλληλο για την κειμενική φύση τέτοιων παιχνιδιών. Σήμερα, το ποντίκι είναι η τυπική συσκευή εισόδου για τον χρήστη που αλληλεπιδρά με τα περισσότερα παιχνίδια περιπέτειας- και η διεπαφή χρήστη έχει βελτιωθεί ξανά και ξανά για να παρέχει μία καλύτερη εμπειρία παιχνιδιού.

Η μορφή εξόδου των πρώτων παιχνιδιών περιπέτειας στον υπολογιστή (π.χ. Zork) περιοριζόταν σε οπτική απεικόνιση κειμένου, η οποία αργότερα ενισχύθηκε με γραφικά. Καθώς τα συστήματα υπολογιστών έγιναν πιο ισχυρά και λιγότερο ακριβά, τα γραφικά βελτιώθηκαν σε έγχρωμες απεικονίσεις και από στατικό φόντο σε κινούμενο φόντο με κίνηση των χαρακτήρων και άλλων αντικειμένων στο προσκήνιο. Το κείμενο σταδιακά καταργήθηκε ή ενισχύθηκε από προφορικές γραμμές με τη μορφή ψηφιοποιημένης ομιλίας.

Ορισμένα από τα σημερινά παιχνίδια τελευταίας τεχνολογίας χρησιμοποιούν ψηφιοποιημένο βίντεο πλήρους κίνησης ως μέσο οπτικής παρουσίασης. Το "Game accounting" είναι ένας τρόπος αξιολόγησης της προόδου του παιχνιδιού, ο οποίος απαιτείται σε όλα τα παιχνίδια περιπέτειας. Η πρόοδος μετριέται με βάση τους πόντους που κερδίζονται, τα γεγονότα που ενεργοποιούνται ή τα αντικείμενα που αποκτώνται στην αποθήκη του παίκτη (Inventory). Ορισμένα παιχνίδια περιπέτειας ακολουθούν το σύστημα πόντων, κατά το οποίο η επίτευξη στόχων αποφέρει πόντους στον παίκτη, ενώ άλλα εκτελούν ελέγχους απογραφής για να δουν αν έχουν αποκτηθεί οι απαιτούμενοι πόροι. Ωστόσο, τα περισσότερα χρησιμοποιούν την προσέγγιση της ενεργοποίησης γεγονότων για τον στόχο κορυφαίου επιπέδου.

Οι προγραμματιστές παιχνιδιών μπορούν επίσης να αποφασίσουν να ενσωματώσουν στοιχεία αβεβαιότητας για να ενισχύσουν την αξία επανάληψης του παιχνιδιού και να αποφύγουν μια μοναδική λύση στο παιχνίδι. Η αβεβαιότητα μπορεί να δημιουργηθεί μέσω της τυχαίας τοποθέτησης πόρων (π.χ. αντικειμένων-κλειδιών), διαφορετικής αλληλουχίας γρίφων ή τυχαίας εμφάνισης γεγονότων.

Υπάρχουν πολλές υποκατηγορίες παιχνιδιών περιπέτειας, η καθεμία με το δικό της μοναδικό gameplay και χαρακτηριστικά. Με την ευρύτερη έννοια, υπάρχουν παιχνίδια δυτικής και ιαπωνικής επιρροής. Τα δυτικά παιχνίδια περιπέτειας συχνά δίνουν στον παίκτη μεγάλη ελευθερία να εξερευνήσει και να ανακαλύψει τον κόσμο του παιχνιδιού, ενώ τα ιαπωνικά παιχνίδια περιπέτειας τείνουν να είναι πιο γραμμικά και επικεντρώνονται στην αφήγηση μιας συγκεκριμένης ιστορίας. Εντός αυτών των στυλ, υπάρχει περαιτέρω διαφοροποίηση. Κάποιες από αυτές τις υποκατηγορίες παρατίθενται παρακάτω.

- **Choose your own adventure:** Παρέχουν στον παίκτη επιλογές που επηρεάζουν την έκβαση του παιχνιδιού.

- **Escape:** Με έμφαση στην επίλυση γρίφων, οι παίκτες πρέπει να αποδράσουν από ένα δωμάτιο ή κτίριο. Αυτοί οι τύποι παιχνιδιών έχουν συχνά χρονικά όρια και απαιτούν γρήγορη σκέψη και παρατηρητικότητα για την επίλυση (The Room).
- **Hybrid:** Παιχνίδια που δεν εντάσσονται με σαφήνεια σε κάποια υποκατηγορία και, αντίθετα, συνδυάζουν στοιχεία από πολλούς τύπους (Alan Wake).
- **Interactive movies/visual novels:** Με έμφαση στην αφήγηση, οι παίκτες κάνουν επιλογές που επηρεάζουν την ιστορία. Τα παιχνίδια αυτά έχουν συχνά πολλαπλές ιστορίες που εξελίσσονται παράλληλα και διαφορετικά τέλη παιχνιδιού (Detroit Becomes Human).
- **Point-and-click:** Παιχνίδια που βασίζονται σε μεγάλο βαθμό στην είσοδο του ποντικιού, με τους παίκτες να κάνουν κλικ για να αλληλεπιδράσουν με αντικείμενα και χαρακτήρες. Σε αυτά τα παιχνίδια υπάρχουν μερικές φορές πιο πολύπλοκοι γρίφοι από άλλες υποκατηγορίες και συχνά απαιτείται δοκιμή και σφάλμα (Monkey Island).
- **Puzzle:** Τα συγκεκριμένα παιχνίδια έχουν συχνά πολύπλοκους γρίφους που απαιτούν πολύπλευρη σκέψη (The Witness).
- **Text-based:** Η κλασική υποκατηγορία που βασίζεται σε μεγάλο βαθμό στην εισαγωγή κειμένου. Οι παίκτες πληκτρολογούν εντολές για να αλληλεπιδράσουν με το παιχνίδι (Zork).
- **Walking simulator:** Παιχνίδια που εστιάζουν στην εξερεύνηση, με τους παίκτες να περπατούν και να ανακαλύπτουν νέες περιοχές και προκλήσεις. Αυτά έχουν ελάχιστους έως καθόλου γρίφους και βασίζονται σε μεγάλο βαθμό στην αφήγηση (Fire Watch).

Με τα σύγχρονα τεχνολογικά άλματα, τα παιχνίδια περιπέτειας υπέστησαν μετασχηματισμό στους μηχανισμούς παιχνιδιού, ενσωματώνοντας στοιχεία από άλλα είδη όπως η δράση, το μυστήριο και το παιχνίδι ρόλων. Αυτή η διαφοροποίηση έχει διευρύνει την απήγηση των παιχνιδιών περιπέτειας, προσελκύοντας νέο κοινό, διατηρώντας παράλληλα την κεντρική ουσία της εξερεύνησης και του αφηγηματικού gameplay.

Επιπλέον, η εικονική πραγματικότητα (VR) θα προσφέρει νέους και καθηλωτικούς τρόπους παιχνιδιού, ενώ η τεχνητή νοημοσύνη (AI) θα δημιουργήσει πιο αληθοφανείς χαρακτήρες που θα αλλάζουν συνεχώς με το παιχνίδι. Αυτές οι τεχνολογίες θα βοηθήσουν τα παιχνίδια περιπέτειας να εξελιχθούν και να παραμείνουν σημαντικά στο προσκήνιο. Οι αναβιώσεις και οι επανεκδόσεις κλασικών παιχνιδιών παίζουν επίσης σημαντικό ρόλο, επαναφέροντας αγαπημένους ήρωες σε μια νέα γενιά παικτών, διατηρώντας παράλληλα τη γοητεία και τους μηχανισμούς παιχνιδιού τους.

Η σημασία της Ιστορίας (Story)

Οι άνθρωποι ανέκαθεν χρησιμοποιούσαν τις ιστορίες για να μοιραστούν παραδόσεις, πολιτιστικές αξίες, ιστορικά γεγονότα, ακόμη και για να μεταδώσουν τη γνώση στην επόμενη γενιά. Όπως είναι φυσικό, η αφήγηση ιστοριών είναι επίσης παρούσα και στα παιχνίδια, παρουσιάζεται μέσω διαλόγων, σκηνών, ενεργειών των χαρακτήρων, ιστορίας εντός του παιχνιδιού, μηχανισμών παιχνιδιού και πολλών άλλων. Τα παιχνίδια έχουν μετατραπεί σε ένα μέσο αφήγησης που ανταγωνίζεται τις ταινίες και τη

λογοτεχνία (Ju & Wagner, 1997). Στον επίκεντρο αυτής της εξέλιξης βρίσκονται τα περίπλοκα δομικά στοιχεία που κατασκευάζουν την αφήγηση ενός παιχνιδιού.

- **Δημιουργία κόσμου:** Το θεμέλιο κάθε αξιομνημόνευτης αφήγησης παιχνιδιού είναι η κατασκευή του κόσμου του. Πρόκειται για τη διαδικασία δημιουργίας ενός καθηλωτικού, αληθοφανούς κόσμου που αποτελεί το σκηνικό της ιστορίας. Σε αντίθεση με τα παραδοσιακά μέσα, τα παιχνίδια προσφέρουν ένα διαδραστικό περιβάλλον, όπου οι παίκτες μπορούν να εξερευνήσουν και να εμπλακούν με τον κόσμο γύρω τους. Ο κόσμος ενός παιχνιδιού δεν είναι απλώς ένα σκηνικό αλλά ένας ενεργός συμμετέχων στην αφήγηση. Κάθε στοιχείο, από το αρχιτεκτονικό στυλ των κτιρίων μέχρι τα κοινωνικά πρότυπα των κατοίκων του, συμβάλλει στη βαθύτερη κατανόηση και εμπάθυση στον κόσμο του παιχνιδιού. Η αποτελεσματική δημιουργία περιβάλλοντος εξασφαλίζει ότι οι παίκτες αισθάνονται μέρος του κόσμου που εξερευνούν, καθιστώντας το ταξίδι τους μέσα στο παιχνίδι πιο ελκυστικό και ουσιαστικό.
- **Χαρακτήρες και Δημιουργία συνδέσεων:** Οι χαρακτήρες στα παιχνίδια χρησιμεύουν ως συναισθηματική γέφυρα μεταξύ του παίκτη και της ιστορίας. Η ανάπτυξη ελκυστικών, πολυδιάστατων χαρακτήρων είναι ζωτικής σημασίας για να συνδεθούν οι παίκτες και να επενδύσουν στην αφήγηση. Αυτοί οι χαρακτήρες συχνά εξελίσσονται με την πάροδο του χρόνου, αντιδρώντας στις αποφάσεις του παίκτη και στην εξέλιξη της ιστορίας, δημιουργώντας μια δυνατή σχέση μεταξύ του παίκτη και του παιχνιδιού. Οι παίκτες επενδύουν βαθιά στο ταξίδι των πρωταγωνιστών, βιώνοντας από πρώτο χέρι τον αγώνα, τους θριάμβους και την ανάπτυξή τους. Αυτή η συναισθηματική επένδυση είναι που ωθεί τον παίκτη να συνεχίσει να παίζει, ανυπομονώντας να δει πώς θα εξελιχθεί η ιστορία των αγαπημένων του χαρακτήρων.
- **Πλοκή και σύγκρουση:** Η πλοκή και η σύγκρουση είναι αυτά που προωθούν την ιστορία σε ένα παιχνίδι. Εισάγουν προκλήσεις και εμπόδια που πρέπει να ξεπεράσει ο παίκτης, δημιουργώντας ένταση και οδηγώντας τον άξονα της αφήγησης. Μια καλά δομημένη πλοκή σε ένα παιχνίδι είναι ευέλικτη, επιτρέποντας στις επιλογές του παίκτη να επηρεάσουν σημαντικά την κατεύθυνση και την έκβαση της ιστορίας. Η σύγκρουση στα παιχνίδια δεν περιορίζεται σε φυσικές αντιπαραθέσεις, περιλαμβάνει επίσης ηθικά διλήμματα και συναισθηματικές μάχες. Αυτή η μη γραμμική προσέγγιση της αφήγησης εμπλέκει τους παίκτες με έναν βαθύτερο, πιο προσωπικό τρόπο, καθώς οι επιλογές τους επιδρούν άμεσα στον κόσμο του παιχνιδιού.

Με λίγα λόγια, η αφήγηση και η ιστορία στα παιχνίδια, από τον κόσμο έως και την τελική σύγκρουση του παίκτη, είναι ένα μέσο που όχι μόνο ψυχαγωγεί αλλά και μας συνδέει με νέους κόσμους, χαρακτήρες και εμπειρίες με βαθύτατους τρόπους.

2. 4. Horror Games

Ο φόβος αποτελεί έμφυτη έννοια στον άνθρωπο από τις απαρχές του χρόνου. Συνιστά την αντανάκλαση της ανθρώπινης περιέργειας προς το άγνωστο, το υπερφυσικό και τις σκοτεινότερες πτυχές της ανθρώπινης ψυχής.

Η κατηγορία horror έχει μια ιδιαίτερη επιρροή στο κοινό σε διάφορες μορφές ψυχαγωγία, όπως η λογοτεχνία και ο κινηματογράφος. Εμπνευσμένη από τις ταινίες τρόμου, ενσωματώθηκε ως στοιχείο στα παιχνίδια κατά τα τέλη του 20ου αιώνα. Στον κόσμο του gaming έχει αναδειχθεί ως ένα από τα πιο συναρπαστικά είδη, παρέχοντας στους παίκτες την ευκαιρία να αντιμετωπίσουν τους βαθύτερους φόβους τους και να βυθιστούν σε συναρπαστικές σκοτεινές ιστορίες. Εγκαταλελειμμένα σπίτια και κτίρια, στοιχειωμένα δάση, τρομακτικές σκάλες και κάθε τοποθεσία μπορεί να είναι ένα μέρος τρόμου. Ο τρόπος με τον οποίο χτίζεται και εφαρμόζεται το στοιχείο του horror στα παιχνίδια, επηρεάζει σε μεγάλο βαθμό τον παίκτη και την επιτυχία του ίδιου του παιχνιδιού. Μαζί με τις κατηγορίες sci-fi και fantasy, αποτελούν μια “μαγική” τριάδα, και πολλές φορές συνδυάζονται και εμπλουτίζουν η μια την άλλη.

Πολλές φορές στις ταινίες και στα παιχνίδια τρόμου υπάρχει συνδυασμός του καλού και του κακού (Krzywinska, 2012). Για παράδειγμα, στο resident evil 3 ο βασικός πρωταγωνιστής προσπαθεί να φέρει την ισορροπία σε έναν κόσμο που οι δαιμονικές δυνάμεις απειλούν την ανθρωπότητα. Ωστόσο, υπάρχουν διαφορές ανάμεσα στα παιχνίδια τρόμου και τις ταινίες. Τα παιχνίδια δεν σταματούν στην αφήγηση, αλλά προσφέρουν και τη συμμετοχή του παίκτη στην ιστορία. Στις ταινίες ο θεατής ελπίζει στην επιβίωση του πρωταγωνιστή, ενώ στα παιχνίδια συμμετέχει ενεργά στη δράση (Graja, Chanel and Lopes, 2021). Στα νεότερα παιχνίδια μπορεί να παίρνει αποφάσεις για τη συνέχεια του παιχνιδιού και καταλήγει σε διαφορετικά αποτελέσματα (Lin, 2013). Για παράδειγμα στο Witcher 3 ή το Walking Dead, ο διάλογος ανάμεσα στους παίκτες και τους NPCs επηρεάζουν την πλοκή του παιχνιδιού, επομένως ο παίκτης πρέπει να επιλέγει με προσοχή τις αποφάσεις του.

Είδη και χαρακτηριστικά horror

Είναι δύσκολο να οριστεί με σαφήνεια η κατηγορία Horror, καθώς περιλαμβάνει πολλές δικές της κατηγορίες, υποκατηγορίες και χαρακτηριστικά. Κάθε μια από αυτές περιλαμβάνει δικά της χαρακτηριστικά και στοιχεία, και προσεγγίζει με διαφορετικό τρόπο την εμπειρία του φόβου-τρόμου των παικτών.

Η κατηγοριοποίηση των παιχνιδιών horror είναι περισσότερο γενική, παρά σαφής. Αφορούν περισσότερο την εμπειρία του παίκτη σε τέτοιου είδους περιβάλλοντα, και λιγότερο τον σχεδιασμό των ίδιων των παιχνιδιών (Prohászkoná, 2012).

- **The Uncanny:** περιέχει υπερφυσικά στοιχεία, εξωπραγματικά ή παράλογα, ή γεγονότα που διέπουν τους νόμους της φυσικής αλλά είναι απίστευτα ή συγκλονιστικά (Little Nightmares).
- **The marvelous horror:** περιλαμβάνει παράλογα και ακατανόητα φαινόμενα που μπορούν να εξηγηθούν μόνο με την αποδοχή της ύπαρξης του υπερφυσικού π.χ. βαμπίρ, λυκάνθρωποι (Outlast).
- **The fantastic horror:** δεν προσφέρει σαφείς εξηγήσεις για το παράλογο και ο παίκτης έχει τη δυνατότητα με τη φαντασία του να επιλέξει αν θα εξηγήσει τα φαινόμενα ως παραφυσικά ή ως παραισθήσεις του κύριου πρωταγωνιστή (Until Dawn).

Οι υποκατηγορίες των παιχνιδιών αποκαλύπτουν την πολυπλοκότητα και την ποικιλία που χαρακτηρίζει το είδος. Κάθε μια από αυτές συνδυάζει την πτυχή του φόβου με

στοιχεία από άλλες κατηγορίες παιχνιδιών και θέτει μοναδικές προκλήσεις για τους παίκτες.

- **Survival Horror:** Ο παίκτης εστιάζει στην επιβίωση σε ένα περιβάλλον με περιορισμένους πόρους και εχθρούς που απειλούν την ζωή του χαρακτήρα. Συχνά περιλαμβάνουν τρομακτικό περιβάλλον, ενώ ο χρόνος αποτελεί συνεχή μάχη (The Last of Us).
- **Action Horror:** Ο παίκτης επικεντρώνεται στη δράση ενάντια σε εχθρούς. Συνήθως περιλαμβάνει έντονες μάχες και υπερφυσικούς, τερατώδεις αντιπάλους (Resident Evil Series).
- **Psychological Horror:** Εστιάζει στον τρόμο μέσω της ψυχολογικής αγωνίας και φόβου. Συχνά, περιλαμβάνει πολύπλοκη ιστορία με ανατριχιαστικές αποκαλύψεις, ενώ η απειλή του άγνωστου κρατάει τον παίκτη σε εγρήγορση (Silent Hill).

Χαρακτηριστικά Horror

Ο σχεδιασμός των παιχνιδιών horror είναι ιδιαίτερης σημασίας για την απήχηση και επιτυχία του παιχνιδιού. Αν και νέες παραγωγές ενσωματώνουν πρωτότυπες προσεγγίσεις στην υλοποίηση των παιχνιδιών τρόμου, υπάρχουν κάποια βασικά χαρακτηριστικά που διέπουν τα περισσότερα από αυτά.

- **Ατμόσφαιρα:** Συνήθως αποτελείται από έντονες σκοτεινές τοποθεσίες με αμυδρό φωτισμό, δυσοίωνη μουσική, ξαφνικούς ήχους και τρομακτικά αντικείμενα, ώστε να προκαλέσουν το αίσθημα του φόβου και της ανασφάλειας.
- **Αίσθημα επιβίωσης και Αδρεναλίνης:** Οι παίκτες συνήθως αισθάνονται ευάλωτοι και αδύναμοι σε σχέση με την απειλή του παιχνιδιού. Συνδυαστικά, το παιχνίδι προσφέρει περιορισμένους πόρους και κινήσεις, έτσι ώστε η επιβίωση να αποτελεί βασικό κίνητρο του παίκτη.
- **Εχθροί-Απειλές:** Ο παίκτης, συνήθως, καλείται να αντιμετωπίσει εχθρούς με υπερφυσικές δυνάμεις ή απειλές με τρομακτική υπόσταση.
- **Suspense και plot-twist:** Οι ιστορίες προσφέρουν σασπένς και απροσδόκητες αλλαγές ώστε να καθηλώσουν τον παίκτη και να τον κρατήσουν σε εγρήγορση.
- **Επίλυση γρίφων:** Ο παίκτης εμπλέκεται στην αποκάλυψη του μυστηρίου επιλύοντας γρίφους και puzzles για να προχωρήσει.
- **Συμμετοχή του παίκτη:** Τα παιχνίδια συνήθως χρησιμοποιούν μηχανισμούς ώστε να προσελκύσουν τον παίκτη συναισθηματικά. Αυτοί περιλαμβάνουν το first person gameplay, τα ρεαλιστικά γραφικά και δυναμικούς ήχους.

User Experience

Τα horror παιχνίδια δημιουργούν στον παίκτη μοναδικά συναισθήματα όπως ο φόβος, το άγχος, η πίεση (Gowler & Iacovides, 2019). Παρά την αρνητική τους φύση, πολλές φορές μπορούν να χρησιμοποιηθούν και για ψυχαγωγικούς σκοπούς, σε ένα ελεγχόμενο περιβάλλον και να δημιουργήσουν ένα τρομακτικό και καθηλωτικό παιχνίδι που προσφέρει ευχαρίστηση και υψηλή εμπειρία παιχνιδιού (Demarque and Lima, 2019). Η ενσωμάτωση των ψυχολογικών στοιχείων στη διαδικασία σχεδιασμού του παιχνιδιού,

ιδιαίτερα στην εικονική πραγματικότητα, μπορεί να ενισχύσει το συνολικό τρόμο που βιώνουν οι παίκτες.

Γιατί όμως οι παίκτες συνεχίζουν να αναζητούν αυτά τα παιχνίδια; Γιατί απολαμβάνουν αυτές τις τρομακτικές και άβολες εμπειρίες; Για να κατανοήσουμε την ελκυστικότητα αυτών των παιχνιδιών, πρέπει να εξετάσουμε την συμπεριφορά των ίδιων των παικτών, χωρίζοντάς τους σε δυο κατηγορίες. Η πρώτη κατηγορία είναι οι παίκτες που απολαμβάνουν τον φόβο και την αδρεναλίνη, καθώς τους κάνει να νιώθουν πιο ζωντανοί και ανακουφισμένοι όταν ο τρόμος τελειώνει. Η δεύτερη κατηγορία είναι οι παίκτες που παίζουν για να αναδειχθούν ή να εντυπωσιάσουν, δείχνοντας πως είναι ατρόμητοι. (Ntokos, 2018). Παρακάτω, αναλύονται τα στάδια του φόβου (βλ. [Πίνακας 3.](#)) με κάποια χαρακτηριστικά παραδείγματα:

Επίπεδο	Αίσθημα	Κατάσταση παιχνιδιού	Παράδειγμα
1	Calmness (Ηρεμία)	Απουσία φόβου, ο παίκτης μπορεί να εξερευνήσει ελεύθερα	Ασφαλές μέρος, save-point
2	Mild Anxiety (νευρικότητα)	Μερική ύπαρξη φόβου, ο παίκτης δεν ξέρει ακόμη πολλές πληροφορίες	Ίχνη ανεξήγητων ή παραφυσικών φαινομένων στο χώρο
3	Moderate Anxiety (Επαγρύπνηση)	Ο φόβος εμφανίζεται, ο παίκτης πρέπει να έχει επίγνωση του περιβάλλοντος	Σημάδια από αίμα, αποδείξεις άγνωστων δολοφονικών δυνάμεων
4	Severe Anxiety (ανησυχία)	Ο φόβος προκαλεί προβλήματα προσαρμοστικότητας στον παίκτη	Ηχητικά στοιχεία που αυξάνουν την ένταση, αντικείμενα κινούνται μόνα τους
5	Mild Stress (υπερένταση)	Ο παίκτης φοβάται αυτό που δεν μπορεί να δει ή να εξηγήσει, μικρές αλλαγές στο περιβάλλον πιθανόν τώρα να τον τρομάζουν	Ακούγοντας φωνές ή βήματα αλλά χωρίς να βλέπει κάτι, αντικείμενα κινούνται, σπάνε
6	Severe Stress (δυσφορία)	Ο φόβος απομακρύνει τη λογική από τον παίκτη και αναδύονται ένστικτα επιβίωσης	Μάρτυρας σε πράξη βίας μέσω εικόνας ή/και ήχου

7	Mild Fear (Ηπιος φόβος)	Ένστικτα επιβίωσης και γρήγορα αντανακλαστικά απομακρύνουν την λογική, ο παίκτης είναι πιθανό να φοβάται με μικρές αλλαγές στο περιβάλλον	Στοιχεία για το που βρίσκεται μια μακρινή απειλή, αδύναμα jumpscares, κρύψιμο
8	Severe Fear (Έντονος φόβος)	Ο παίκτης επικεντρώνεται στο να κρυφτεί από την απειλή, ο κίνδυνος ελλοχεύει	Στοιχεία για το που βρίσκεται μια κοντινή απειλή, συνατά jumpscares
9	Terror (Τρόμος)	Ο παίκτης ή θα επιβιώσει ή όχι, γρήγορες κινήσεις έχουν καταβάλει τη λογική	Ο εχθρός κυνηγάει τον παίκτη, απουσία πυρομαχικών κατά τη διάρκεια της μάχης
10	Panic (Πανικός)	Πλήρης απουσία ελέγχου, κατάσταση μάχης ή φυγής	Έξοδος από το παιχνίδι, αφαίρεση ακουστικών, αιφνιδιαστική επίθεση, ο εχθρός έχει σχεδόν πιάσει τον παίκτη

Πίνακας 3. . Στάδια φόβου, επεξήγηση και παραδείγματα

2.5. Κοινωνικοποίηση μέσω του Gaming

Το gaming έχει δημιουργήσει μια πιστή και γεμάτη ενθουσιασμό κοινότητα, η οποία προσελκύεται από την έμφαση που δίνει το είδος στην ιστορία, την ατμόσφαιρα και την εμπάθυση. Αυτή η κοινότητα καλύπτει διάφορες πλατφόρμες, από online φόρουμ και ομάδες κοινωνικής δικτύωσης μέχρι εξειδικευμένες ιστοσελίδες και ομάδες θαυμαστών. Σε αυτές, οι φίλοι των παιχνιδιών ανταλλάσσουν τις εμπειρίες τους, μιλούν για τα αγαπημένα τους παιχνίδια και δίνουν συμβουλές και προτάσεις στρατηγικής για την επίλυση δύσκολων γρίφων. Η αίσθηση της συντροφικότητας και του κοινού ενθουσιασμού μέσα σε αυτές τις κοινότητες δημιουργεί ένα φιλόξενο περιβάλλον όπου οι παίκτες μπορούν να έρθουν σε επαφή με άτομα με τις ίδιες ιδέες που μοιράζονται την αγάπη τους για το gaming.

Ειδικές ιστοσελίδες και forums χρησιμεύουν ως κέντρα συζητήσεων, ειδήσεων και κριτικών, προσφέροντας χρήσιμες πηγές τόσο για έμπειρους παίκτες όσο και για αρχάριους. Αυτές οι πλατφόρμες συχνά διαθέτουν οδηγίες περιήγησης, τέχνη και μυθοπλασία θαυμαστών, πλουτίζοντας τη συνολική εμπειρία για τους οπαδούς και συμβάλλοντας στη ζωντανή κουλτούρα που περιβάλλει το είδος. Επιπλέον, τα συνέδρια και οι εκδηλώσεις παιχνιδιών περιπέτειας παρέχουν την ευκαιρία στους fans να συναντηθούν από κοντά, να συμμετάσχουν σε συζητήσεις και workshops, να γνωρίσουν τους προγραμματιστές και να γιορτάσουν το κοινό τους πάθος. Μέσω αυτών των κοινών χώρων, οι λάτρεις των παιχνιδιών δημιουργούν μόνιμες συνδέσεις,

ανταλλάσσουν ιδέες και συμβάλλουν σε έναν συνεχή διάλογο σχετικά με την εξέλιξη και τον αντίκτυπο των παιχνιδιών (Valentine & Jensen, 2016).

3. Σχεδιασμός

Στο πλαίσιο αυτής της διπλωματικής σχεδιάσαμε ένα συναρπαστικό παιχνίδι περιπέτειας, που συνδυάζει μυστήριο και στοιχεία horror σε 3D περιβάλλον. Δημιουργημένο ως εμπειρία ενός παίκτη, το “ The Journalist: The Triangle of Mysteries” ακολουθεί ένα συναρπαστικό ταξίδι μέσα από μια απίστευτη ιστορία. Καθώς ο παίκτης εμβαθύνει στο παιχνίδι, καλείται να λύσει γρίφους, να συλλέξει στοιχεία για να λύσει το μυστήριο και να λάβει καθοριστικές αποφάσεις που διαμορφώνουν την πλοκή του παιχνιδιού. Επιπλέον, πρέπει να καταφέρει να παραμείνει ασύλληπτος από το κακό που ελλοχεύει, αυξάνοντας έτσι τη δραστηριότητα του παιχνιδιού και ενισχύοντας την αδρεναλίνη.

3.1. Ιστορία - Εκκίνηση Παιχνιδιού - Τέλος

Ο αρχισυντάκτης του CNN στη Νέα Υόρκη του 1970, αναθέτει στην πιο έμπιστη και ταλαντούχα ερευνητική δημοσιογράφο του καναλιού του, Rebecca Smith, ένα νέο θέμα. Η Rebecca θα πρέπει να πάρει συνέντευξη από έναν ναυαγό που διασώθηκε πρόσφατα από ένα έρημο νησί κοντά στο τρίγωνο των Βερμούδων. Ο John Trueman ήταν χαμένος για 25 χρόνια, μέχρι που τυχαία διασώθηκε από μια ομάδα του λιμενικού. Συμφωνεί να επιστρέψει στο νησί για την συνέντευξη, ώστε να μπορέσει το κανάλι να πάρει και πλάνα από το καταφύγιο που του έσωσε τη ζωή αυτά τα 25 χρόνια απομόνωσης.

Η Rebecca ξεκινά το ταξίδι με τον John, τον εικονολήπτη και παιδικό της φίλο Tobias Lee και τον πιλότο του ελικοπτερου που θα τους μεταφέρει στο νησί, Felipe Hernan. Η συνέντευξη ήδη αρχίζει μέσα στο ελικόπτερο, όπου ο John περιγράφει τα γεγονότα που οδήγησαν στο βασανιστήριο του. Ξετυλίζει τις μνήμες του και αφηγείται πως το αλιευτικό σκάφος που βρισκόταν μαζί με την ομάδα του, συνετρίβη πάνω σε υποθαλάσσια βράχια, πως το κύμα τον ξέβρασε σε μια ακτή όπου και κατάλαβε ότι είναι ο μοναδικός επιζών. Διέσωσε ο,τι μπορούσε από τα συντρίμια του σκάφους ώστε να φτιάξει μια καλύβα για να προφυλαχθεί. Φθάνοντας στο νησί, ο John δεν είναι σε ψυχολογική θέση να συνεχίσει προς την καλύβα και προτρέπει την Rebecca και τον Tobias να συνεχίσουν χωρίς αυτόν, καθώς θέλει να περιμένει στο ελικόπτερο με τον Felipe.

Η Rebecca με τον Tobias πεζοπορούν το λόφο, και βρίσκονται μπροστά από την καλύβα του John. Καθώς εξερευνούν τα πράγματα στην καλύβα, τις αυτοσχέδιες παγίδες κυνηγιού και τα ζωικά λάφυρα, συνειδητοποιούν πως υπάρχουν δύο κρεβάτια και πως όλα δείχνουν ότι υπήρχαν δύο άτομα που ζούσαν εκεί. Δίπλα από τα κρεβάτια κρυμμένη κάτω από ένα χαλί βρίσκεται μια σιδερένια καταπακτή. Την ώρα αυτής της ανακάλυψης, ο John ορμάει μέσα στην καλύβα και ακινητοποιεί τον Tobias, χτυπώντας τον.

Το παιχνίδι μας ξεκινάει όταν η ηρωίδα μας πέφτει στις κρυφές κατακόμβες που οδηγεί η καταπακτή για να σωθεί από την ξαφνική επίθεση του John. Η καταπακτή από πάνω της σφραγίζεται, επομένως χάνεται και η μοναδική γνωστή έξοδος από τις κατακόμβες. Πλέον η Rebecca βρίσκεται αντιμέτωπη με το σκοτάδι και το άγνωστο.

Με όπλο τον φακό της περιηγείται στις κατακόμβες. Σχεδόν όλες οι πόρτες είναι περίτεχνα κλειδωμένες και ο παίκτης, μέσα από τα μάτια της Rebecca, πρέπει να διαθέτει ευρηματικότητα, εξυπνάδα και μνήμη για να καταφέρει να τις ανοίξει. Ήδη σε πολύ αρχικό στάδιο συλλέγοντας στοιχεία, μπορεί να καταλάβει πως ο John Trueman δεν είναι αυτός που ισχυρίζεται. Στην πραγματικότητα είναι ο Ezekiel Asir, ένας αιμοσταγής σαδιστής που συνδέεται με το θάνατο του πραγματικού John. Όσο ο παίκτης περιηγείται στα πρώτα δωμάτια, αγνοεί την ύπαρξη μιας μυστικής εισόδου από την οποία τρυπώνει ο Ezekiel μαζί με τους δύο αναίσθητους συνεργάτες της. Τους οδηγεί στο δωμάτιο των βασανιστηρίων (D2) και ανά τακτά χρονικά διαστήματα (σε checkpoint που ενεργοποιεί ο παίκτης), ο Ezekiel ψάχνει την Rebecca.

Τα στοιχεία που μαζεύει στα δωμάτια, δεν δίνουν μόνο τρόπους να ανοίξουν οι καλά κλειδωμένες πόρτες, αλλά σιγά σιγά ξετυλίγεται μια ιστορία που κανείς δεν μπορούσε να φανταστεί. Ο Ezekiel δεν ναυάγησε στο νησι, αλλά μεγάλωσε εκεί. Ο πατέρας του (Samuel Asir) ένας συλλέκτης περιέργων τεχνουργημάτων και αντικειμένων, μεγάλωσε τον Ezekiel μέσα σε αυτές τις κατακόμβες. Μαζί βασάνιζαν και κατέγραφαν τις διάφορες αντιδράσεις των θυμάτων τους, ανίδεων ψαράδων που περνούσαν από το νησί και τους δελιάζαν να κατέβουν για να τους βοηθήσουν.

Στην πορεία του παιχνιδιού η Rebecca καταλαβαίνει πως οι συνεργάτες της είναι ζωντανοί αλλά είναι επίσης και οι μόνοι που γνωρίζουν την μυστική έξοδο για να σωθούν. Καταφέρνει να παγιδεύσει τον Ezekiel και να κερδίσει ένα προβάδισμα για να σώσει του συνεργάτες της. Εκεί ο παίκτης βρίσκεται σε ένα δίλημμα, ποιον να σώσει. Τον αγαπημένο της φίλο; Τον οδηγό του ελικοπτέρου; Στην ιστορία μας υπάρχουν δύο πιθανά τέλη:

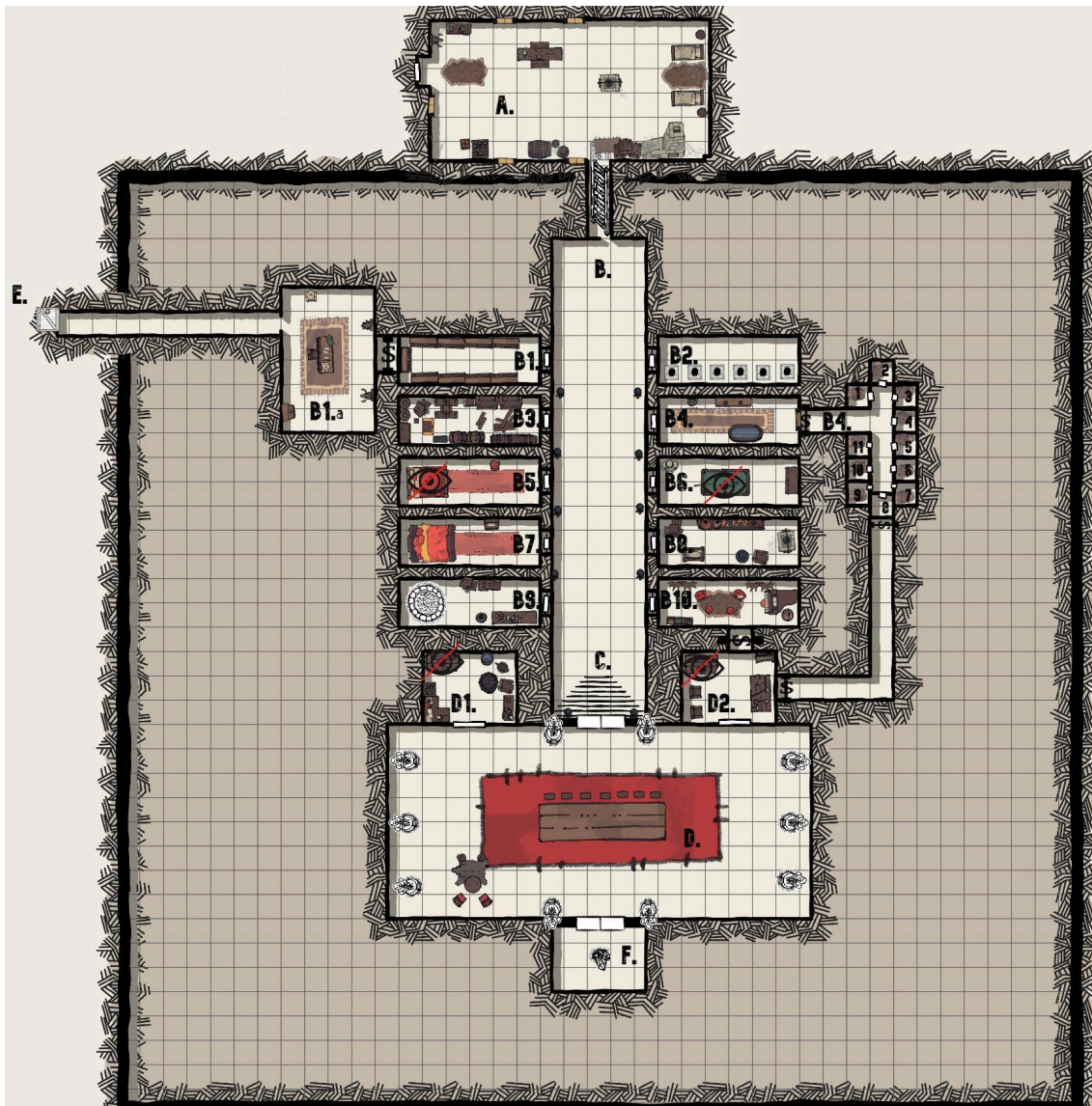
- Αν ο παίκτης σώσει τον Tobias (cameraman) τότε βγαίνοντας από τη μυστική έξοδο τρέχοντας μακριά από την καλύβα, ο Ezekiel τους προλαβαίνει και τους πιάνει (evil end).
- Αν ο παίκτης σώσει τον Felipe (πιλότο) τότε μαζί με την Rebecca τρέχουν στην παραλία και βλέπουν ένα ελικόπτερο που έστειλε το κανάλι αφού δεν είχε νέα τους, και σώζονται (happy end).

3. 2. Χάρτης Παιχνιδιού

Η δόμηση του χάρτη ξεκινά με την καλύβα του Ezekiel και μέσα από μια καταπακτή κατεβαίνει επίπεδο, στις υπόγειες κατακόμβες. Στις εικόνες παρακάτω, ονοματίζονται τα δωμάτια του χάρτη (βλ. [Εικόνα 1.](#)), επισημασμένα με πράσινο χρώμα αυτά που τελικά υλοποιήθηκαν, και παρουσιάζεται επίσης ο αρχικός σχεδιασμός του χάρτη (βλ. [Εικόνα 2.](#)).

- A. Cabin
- B. Catacombs
 - B1. Library
 - B1.a. Secret Office
 - B2. Privy
 - B3. Storage
 - B4. Bathroom
 - B4.a. Secret Prison with cells 1-11
 - B5. Ezekiel's Bedroom
 - B6. Play Room
 - B7. Samuel's Bedroom
 - B8. Kitchen
 - B9. Pantry
 - B10. Tea Room
- C. Stair Gate
- D: Great Hall
 - D1: Armory
 - D2: Torture Room
- E: Secret Exit
- F: Crypt

Εικόνα 1. Λίστα Δωματίων-Υποδοματίων παιχνιδιού



Εικόνα 2. Χάρτης Παιχνιδιού

3.3. Γρίφοι

Όπως αναφέρεται στην ιστορία του παιχνιδιού, ο παίκτης θα πρέπει να λύσει κάποιους γρίφους για να μπορέσει να εισέλθει και να εξερευνήσει τα παραπάνω δωμάτια του χάρτη. Αυτό το καταφέρνει μαζεύοντας διάφορα στοιχεία, τα οποία δεν αποτελούν

μόνο στοιχεία για την επίλυση των γρίφων αυτών, αλλά βοηθούν στην εξέλιξη της ιστορίας του παιχνιδιού.

Γρίφος Κλειδί

Στο πρώτο κιάλας δωμάτιο (Secret Office), ο παίκτης βρίσκει μέσα στο συρτάρι του γραφείου ένα κλειδί. Αυτό το κλειδί, χωρίς ο παίκτης να το γνωρίζει ακόμα, είναι και το κλειδί που χρειάζεται για να ανοίξει η πόρτα του δωματίου του Ezekiel (Ezekiel 's Bedroom). Μέσα στο συρτάρι, κάτω από το κλειδί, για πρώτη φορά ο πρωταγωνιστής έρχεται σε επαφή με το όνομα Ezekiel Asir, χωρίς ακόμα αυτό να ξέρει τι σημαίνει.

Γρίφος Φωτογραφία

Μέσα στο ίδιο δωμάτιο που βρίσκεται το κλειδί, πάνω σε μια από τις βιβλιοθήκες, υπάρχει μία κορνίζα με μία φωτογραφία. Στο πίσω μέρος της φωτογραφίας υπάρχει η εξής σημείωση: “Dad, 1956”. Η φωτογραφία απεικονίζει τον πατέρα του Ezekiel και η χρονολογία αφορά το έτος θανάτου του. Η χρονολογία αυτή, είναι και ο κωδικός της πόρτας που ανοίγει το δωμάτιο της αποθήκης (Storage).

Γρίφος Γράμμα

Μέσα στο δωμάτιο της αποθήκης (Storage), κάτω από μία σιδερένια ντουλάπα, βρίσκεται ένα γράμμα, το οποίο ανήκει στον ίδιο τον John Trueman. Στο γράμμα αναφέρονται τα γεγονότα που συνέβησαν στον πραγματικό John Trueman και μέσα από αυτό δίνει στοιχεία για την επίλυση του γρίφου της διπλής πόρτας στο δωμάτιο Stair Gate. Το γραμμα έχει ως εξής:

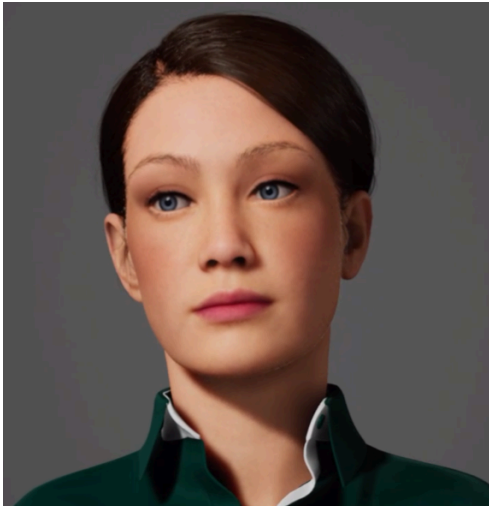
“ i’ve been heRe foR eLeven weeks
every singLe day we are being tortured,
and some of us neveR come back
i don’t know how long i am going to Live
he is keeping us fed so we can last Longer
i tried to escape this helL,
but when i managed to reach the stair gate,
he was already there opening the door and
he caught me again
if you ReAd this, there is no hope
you aRe doomed ,
J.T.“

Τα κόκκινα γράμματα σχηματίζουν μια ακολουθία, “RLLRLLRR”, η οποία και αποτελεί και το σωστό συνδυασμό στα κουμπιά- μάτια ενός κρανίου δίπλα στη διπλή πόρτα, όπου **R** - Right Eye και **L** - Left Eye.

Συμπέρασμα - Ναύτες

Μέσα στο δωμάτιο του Ezekiel (Ezekiel 's Bedroom), σε ένα κιβώτιο με διπλωμένους παπύρους, υπάρχει ένας φάκελος που περιέχει καταγραφές των βασανιστηρίων που εκτελούσε ο Ezekiel στους ναύτες. Σε αυτό το σημείο ο παίκτης διαβάζει τον τρόπο με τον οποίο πέθανε ο πραγματικός John Trueman, και η πρωταγωνίστρια έρχεται στο συμπέρασμα ότι αυτός ο John Trueman που γνώρισε είναι κάποιος άλλος.

3. 4. Χαρακτήρες παιχιδιού - Backstories



Όνομα: Rebecca Smith

Ηλικία: 35 ετών

Καταγωγή: Boston, USA

Γονείς: Derek Smith
Samantha Johnson

Ύψος: 173 cm

Βάρος: 65 kg

Επάγγελμα: Ερευνητική δημοσιογράφος (CNN)

Χαρακτηριστικά: Μελαχρινή, κοντά μαλλιά, πράσινα μάτια, έντονες γωνίες προσώπου

Προσωπικότητα: Εργασιομανής, θαρραλέα, οξυδερκής, έλλειψη προσωπικής ζωής

Backstory: Έζησε ανέμελα παιδικά χρόνια στη Βοστώνη, μοναχοκόρη σε μια ευκατάστατη οικογένεια. Σπούδασε δημοσιογραφία όπου και γνώρισε τον Matthew Sanders, τον μεγάλο έρωτα της ζωής της. Πριν προλάβει να ολοκληρώσει τις σπουδές της, ο Matthew βρέθηκε νεκρός υπό άγνωστες συνθήκες στο διαμέρισμά τους. Από τότε έχει ως σκοπό ζωής να μην αφήσει καμία υπόθεση ανεξιχνίαστη, με την ελπίδα κάποια στιγμή να λύσει το μυστήριο πίσω από τον θάνατο του Matthew.



Όνομα: Tobias Lee

Ηλικία: 30 ετών

Καταγωγή: New York, USA

Γονείς: Wu Lee
Rose Black

Ύψος: 190 cm

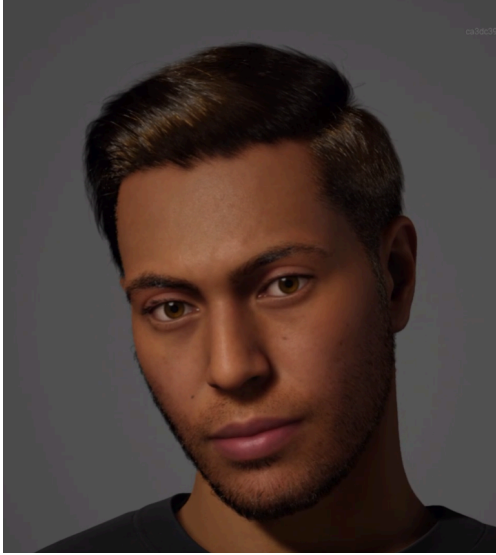
Βάρος: 85 kg

Επάγγελμα: Εικονολήπτης (CNN)

Χαρακτηριστικά: Μελαχρινός, μαύρα μάτια, ασιατικά και αφρικανικά χαρακτηριστικά, άφρο μαλλιά, γεροδεμένος

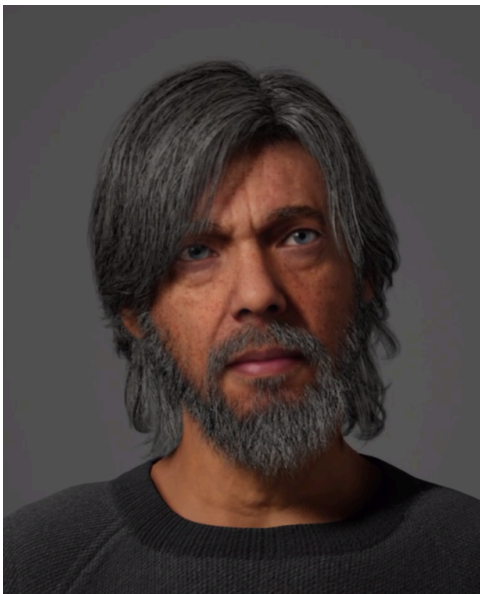
Προσωπικότητα: Κοινωνικός, φιλικός, ευγενικός, πρόθυμος, αγχώδης, υποχόνδριος

Backstory: Γεννήθηκε στη Νέα Υόρκη από μητέρα αφρικανικής καταγωγής και πατέρα Κινεζικής καταγωγής. Μεγάλωσε σε μια πολυμελή οικογένεια και από μικρή κιόλας ηλικία ανέλαβε την ανατροφή των μικρότερων αδερφών του. Όταν τα αδέρφια του μεγάλωσαν αρκετά, ασχολήθηκε με την εικονοληψία και προσλήφθηκε στο CNN, όπου και συνεργάστηκε με την Rebecca. Είναι πολύ καλός φίλος της Rebecca, και ίσως ο μοναδικός της.



Όνομα: Felipe Hernan
Ηλικία: 20 ετών
Καταγωγή: Morelia, Mexico
Γονείς: Maria - Luisa Perez
Αγνώστου πατρός
Ύψος: 168 cm
Βάρος: 73 kg
Επάγγελμα: Μισθωτός πιλότος
Χαρακτηριστικά: Καστανά μαλλιά και μάτια, σκουρόχρωμη επιδερμίδα, έντονα φρύδια και μουστάκι, πλατύ χαμόγελο
Προσωπικότητα: Αυθόρμητος, ευδιάθετος με χιούμορ, αδυναμία στο γυναικείο φύλο, άγνοια κινδύνου, εγωιστής

Backstory: Μεγάλωσε σε μονογονεϊκή οικογένεια χωρίς πατρική φιγούρα. Περνάει πολύ χρόνο με φίλους και λατρεύει τα ταξίδια. Έχει μεγάλη αδυναμία στη μητέρα του και δίψα για περιπέτειες. Όσες φορές έχει συνεργαστεί με τη Rebecca είναι εμφανής η έλξη που νιώθει για αυτή, γεγονός που κολακεύει τη Rebecca.



Όνομα: John Truman (Ezekiel Asir)
Ηλικία: 50 ετών
Καταγωγή: Virginia, USA
Γονείς: Samuel Asir
Luisa Portman
Ύψος: 199 cm
Βάρος: 85 kg
Επάγγελμα: Άγνωστο
Χαρακτηριστικά: Χλωμό δέρμα, μπλε μάτια, λιπόσαρκη φιγούρα με έντονα ζυγωματικά, ατημέλητα μαύρα μαλλιά
Προσωπικότητα: Αντικοινωνικός, μυστηριώδης, νευρικός με κινητικά τικ, γοητευτικός, ευφραδής

Backstory: Ο μόνος κόσμος που γνώρισε ήταν το έρημο νησί όπου μεγάλωσε με τον πατέρα του, Samuel. Έγινε υποχείριό του από μικρή ηλικία με αποτέλεσμα να γίνει σαδιστής και να απολαμβάνει τον βασανισμό των θυμάτων του. Για τον Ezekiel, αυτή η τάση του να βασανίζει ανθρώπους, είναι ο τρόπος να καταλάβει την ανθρώπινη φύση και τα συναισθήματα, κάτι που δεν έμαθε ποτέ από τον πατέρα του. Ο πατέρας του πέθανε πριν είκοσι χρόνια, και έζησε μόνος του μέχρι που βρέθηκε κατά τύχη από ομάδα λιμενικών. Καθώς δεν ήθελε να αποκαλύψει την ταυτότητά του, έδωσε ψεύτικα στοιχεία για το ποιος είναι, κλέβοντας την ταυτότητα ενός ναύτη που κάποτε ναυάγησε στο νησί.



Όνομα: Samuel Asir (αποθανόν)

Καταγωγή: Virginia, USA

Γονείς: Άγνωστοι

Τέκνα: Ezekiel Asir

Χαρακτηριστικά: επιβλητική φυσιογνωμία, μεγάλη ομοιότητα από τον Ezekiel

Προσωπικότητα: Αυστηρός, σκοτεινός, ζητήματα διαχείρισης θυμού

Backstory: Ο Samuel ήταν ένας άντρας της υψηλής κοινωνίας της Virginia. Παντρεύτηκε νέος μια γυναίκα την οποία δεν ήθελε, μόνο και μόνο για το κοινωνικό της “στάτους”. Την έλλειψη μιας βαθιάς σχέσης, ξεκίνησε να γεμίζει με συλλογή περιέργων αντικειμένων ή αντικών. Ξεκίνησε τα ταξιδεύει τον κόσμο με στόχο να ανακαλύψει νέα και σπάνια αντικείμενα. Κάποια στιγμή άκουσε για την ύπαρξη ενός σπάνιου και πολύτιμου λίθου σε ένα νησί στα ανοιχτά της Virginia. Έχοντας αποκτήσει τον πρώτο του γιο, σε ηλικία 2 περίπου ετών, αποφάσισε να τον πάρει μαζί του στην αναζήτηση του πετραδιού. Μετά από πολύκαιρη αναζήτηση, ανακάλυψε το νησί του πετραδιού, βέβαια ακόμα και μετά τον θάνατό του, δεν έχει βρεθεί το πετράδι. Η αδυναμία συνάμα με την μανία του να το βρει, τον έκανε να πάρει την απόφαση να μείνει για πάντα στο νησί.

3. 5. Χειρισμός Παίκτη

Το παιχνίδι ξεκινάει στις υπόγειες κατακόμβες μέσα στο σκοτάδι. Ο παίκτης έχει τις βασικές δυνατότητες κίνησης αλλά και κάποιες που προστέθηκαν κατά την υλοποίηση. Παρακάτω αναγράφεται αναλυτικά ότι χρειάζεται να ξέρει ο παίκτης για να παίξει το συγκεκριμένο παιχνίδι.

- **Κίνηση:** Πλήκτρα W, A, S, D ή Πλήκτρα βέλους.
Χρησιμοποιούνται για την κίνηση προς τα εμπρός (W), προς τα πίσω (S) και την κίνηση προς τα αριστερά (A) και προς τα δεξιά (D). Τα πλήκτρα βέλους μπορούν να χρησιμοποιηθούν αντίστοιχα αντί των W, A, S, D.
- **Κάμερα:** Ποντίκι
Η μετακίνηση του ποντικιού ελέγχει την κατεύθυνση της κάμερας.
- **Φακός:** Πλήκτρο F
Ενεργοποιεί/Απενεργοποιεί το φακό για να βλέπει ο παίκτης στο σκοτάδι.
- **Pause Menu:** Πλήκτρο Tab
Χρησιμοποιείται για την παύση του παιχνιδιού.
- **Inventory:** Πλήκτρο I
Χρησιμοποιείται για να ανοίξετε την αποθήκη του παίκτη, υπενθυμίζοντας βασικές πληροφορίες.
- **Interact:** Πλήκτρο E
Χρησιμοποιείται για να αλληλεπιδράσει ο παίκτης με αντικείμενα και να μιλήσει με NPCs.

- **Κουμπιά:** Αριστερό κλικ ποντικιού
Η επιλογή του κουμπιού των widgets που επιθυμεί ο παίκτης.
- **Quit Game:** Πλήκτρο Esc
Ο παίκτης παραιτείται από το παιχνίδι οριστικά.

4. Υλοποίηση

Σε αυτό το κεφάλαιο, θα εμβαθύνουμε στην πολύπλοκη διαδικασία υλοποίησης των ιδεών μας για το παιχνίδι χρησιμοποιώντας την Unreal Engine, μια ισχυρή και ευέλικτη πλατφόρμα ανάπτυξης παιχνιδιών. Η υλοποίηση σηματοδοτεί το κομβικό στάδιο όπου οι έννοιες μετατρέπονται σε διαδραστικές εμπειρίες, συνδυάζοντας το δημιουργικό όραμα με την τεχνική εμπειρογνομosύνη για να δώσουμε ζωή στην ιδέα μας.

4.1. Unreal Engine 5

Το Unreal Engine είναι μια μηχανή σχεδιασμού και δημιουργίας ηλεκτρονικών παιχνιδιών που δημιουργήθηκε από τη Epic Games. Είναι δωρεάν προς το κοινό, και παρέχει στους developers μεγάλη γκάμα εργαλείων και ρυθμίσεων για την δημιουργία 3D παιχνιδιών. Γραμμένη σε γλώσσα C++ , εκτός από υπολογιστές υποστηρίζει και άλλες πλατφόρμες, όπως κονσόλες (PS, Xbox), φορητές συσκευές και VR headsets. Πλέον η εταιρία Epic games , μετά από ενημερώσεις έχει φτάσει στη έκδοση 5.3 της μηχανής, την οποία χρησιμοποιήσαμε και για το σχεδιασμό του παιχνιδιού μας. Η έκδοση αυτή παρέχει προηγμένα γραφικά, ισχυρή προσομοίωση της φυσικής, μεγάλη συλλογή από έτοιμα 3D assets και materials, και επεκτάσεις. Εκτός από την ανάπτυξη παιχνιδιών, το Unreal Engine υποστηρίζει και άλλες εφαρμογές, όπως η αρχιτεκτονική απεικόνιση, κινηματογραφικά video και σχεδιασμό 3D μοντέλων. Κάποια από τα παιχνίδια που έχουν σχεδιαστεί πάνω σε αυτή την πλατφόρμα είναι το Hogwarts Legacy, Fortnite, Hellblade: Senua's Sacrifice και Final Fantasy 7:Remake.

Unreal Vs Unity

Μια απόφαση που πρέπει να πάρει κάθε προγραμματιστής παιχνιδιού (και εμείς), είναι ποια πλατφόρμα θα χρησιμοποιήσει για τον σχεδιασμό του παιχνιδιού του. Οι δύο καλύτερες επιλογές στην αγορά, έως τώρα, είναι το Unreal Engine και το Unity.

Το Unity είναι μια πλατφόρμα ανοιχτού κώδικα για τον σχεδιασμό 2D και 3D παιχνιδιών από την Unity Technologies. Χρησιμοποιεί C# και υποστηρίζει εξίσου πολλαπλές πλατφόρμες. Επίσης, είναι κατάλληλο για δημιουργία παιχνιδιών VR και AR. Πλέον, οι προγραμματιστές μπορούν να αναπτύξουν και να πουλήσουν τα δικά τους assets. Είναι beginner-friendly πλατφόρμα και ευέλικτη στη χρήση.

Κάθε μια από τις μηχανές αυτές έχει τις δικές της διαφορές και πλεονεκτήματα. Και οι δυο πλατφόρμες μπορούν να δημιουργήσουν AAA ποιότητα γραφικών, υποστηρίζουν τα περισσότερα λογισμικά της βιομηχανίας και παρέχουν πολλά εργαλεία όπως επεξεργασία, animation, προσομοίωση της φυσικής, VR κ.α. (Simran Kaur Arora, 2021). Κάποιες από τις βασικές διαφορές τους είναι:

- **Γλώσσα προγραμματισμού:** Το Unreal Engine χρησιμοποιεί C++, που αν και θεωρητικά δύσκολη, αποτελεί την καλύτερη επιλογή για προγραμματισμό παιχνιδιών σήμερα. Αντίστοιχα, το Unity χρησιμοποιεί C# που είναι μια πιο εύκολη γλώσσα και πιο φιλική προς τους ερασιτέχνες developers.
- **Ποιότητα γραφικών:** Αν και οι δυο πλατφόρμες μπορούν να παράγουν υψηλής ποιότητας γραφικά, το Unreal Engine έχει το πλεονέκτημα καθώς παρέχει ρεαλιστικές οπτικοποιήσεις και καλύτερο φωτισμό (lumen), σκιές.
- **Rendering:** Το Unreal Engine υποστηρίζει καλύτερη απόδοση μετά την επεξεργασία, ενώ στην περίπτωση του Unity το rendering είναι πιο αργό.

Στην δική μας περίπτωση, επιλέξαμε το Unreal Engine και οι λόγοι είναι διάφοροι. Αρχικά, το Unreal διαπρέπει στο σχεδιασμό 3D παιχνιδιών. Επιπλέον, παρέχει μια βιβλιοθήκη με έτοιμα ρεαλιστικά materials και assets δωρεάν, το Quixel Bridge. Από το Marketplace for UE που είναι μια πλατφόρμα ηλεκτρονικού εμπορίου, οι προγραμματιστές συνδέονται με σχεδιαστές περιεχομένου (assets, materials, έτοιμο κώδικα παιχνιδιού) και μπορούν να αποκτήσουν πληθώρα στοιχείων. Τέλος, το Unreal Engine δίνει την δυνατότητα να αποφευχθεί η συγγραφή κώδικα μέσω της εναλλακτικής των blueprints που θα δούμε στη συνέχεια.

4.2. Περιβάλλον-Σχεδιασμός levels

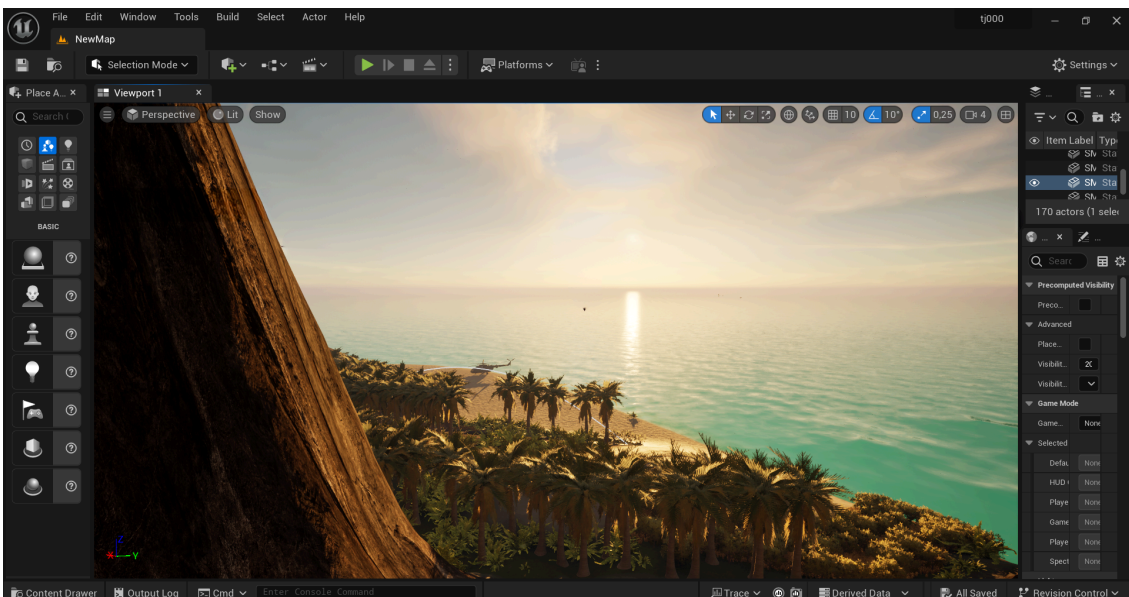
Το πρώτο βήμα μας στην υλοποίηση του παιχνιδιού ήταν ο αρχιτεκτονικός σχεδιασμός των levels. Δημιουργήσαμε δύο βασικά levels, το νησί που χρησιμοποιήθηκε μόνο για τα cinematics και τις κατακόμβες που γίνεται το gameplay. Για όλο το σχεδιασμό χρησιμοποιήσαμε δωρεάν assets, materials και textures από τις συνεργαζόμενες πλατφόρμες Quixel Bridge και Marketplace. Επιπλέον, χρησιμοποιήσαμε το Blender για να σχεδιάσουμε τα δικά μας assets.

Island level

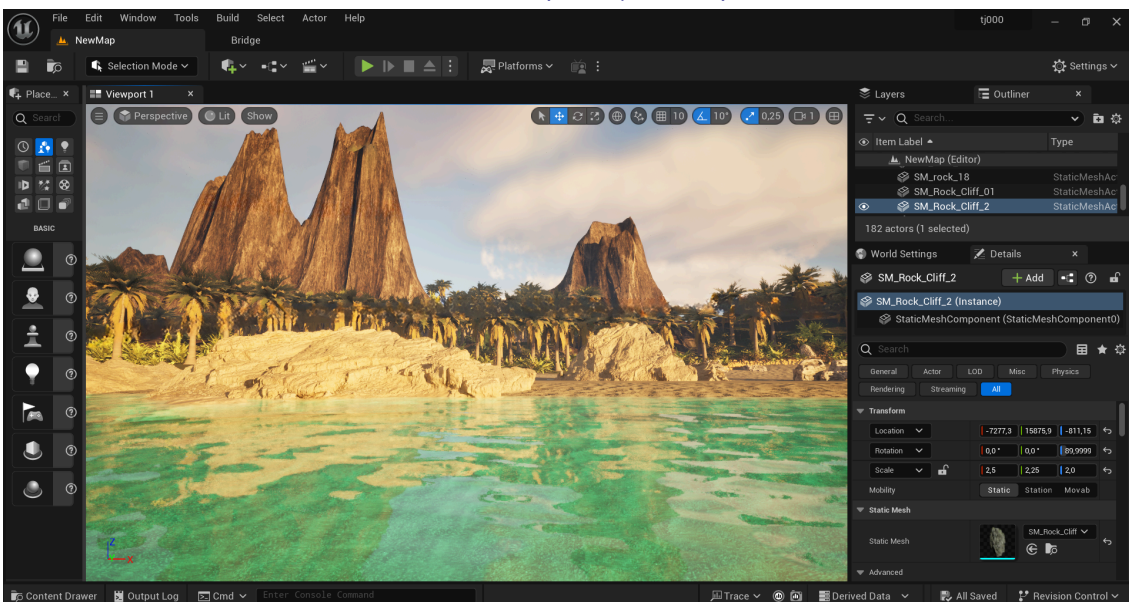
Κατά το άνοιγμα ενός νέου project στο Unreal Engine υπάρχει η επιλογή για το είδος του παιχνιδιού first person, third person, VR κ.λπ. Επιλέξαμε το First person, και στη δημιουργία level επιλέξαμε ένα basic level. Αρχικά προσθέσαμε παράγοντες φωτισμού, ουρανό, ήλιο, ομίχλη. Για τον σχεδιασμό του νησιού χρησιμοποιήσαμε το εργαλείο Landscape για να μπορέσουμε να σμιλεύσουμε το έδαφος ώστε να μοιάζει με νησί. Το εργαλείο αυτό δίνει επιλογές για paint, smooth, flatten, κ.α. για το έδαφος. Στη συνέχεια, προσθέσαμε κάποια plugins στο Unreal για τη δημιουργία του ωκεανού και τη διαχείρισή του, όπως το χρώμα, το ύψος των κυμάτων και την τυχαιότητα τους για να φαίνονται πιο φυσικά (βλ. [Εικόνα 4.](#)). Μόλις καταλήξαμε στο σχήμα του νησιού, εφαρμόσαμε στις επιφάνειες τα materials που επιλέξαμε από το Quixel Bridge για την άμμο, τα βράχια και τους λόφους (βλ. [Εικόνα 5.](#)). Το δεύτερο εργαλείο που χρησιμοποιήσαμε ήταν το Foliage. Αφού επιλέξαμε τα φυτά και τα δέντρα που θέλαμε να προσθέσουμε στο περιβάλλον, με το εργαλείο αυτό μπορέσαμε να σχεδιάσουμε εύκολα την τροπική χλωρίδα του νησιού, που θα υπάρχει βλάστηση και που θα υπάρχει μονοπάτι (βλ. [Εικόνα 3.](#)). Τελικά, δημιουργήσαμε το νησί που φαίνεται στις παρακάτω εικόνες.



Εικόνα 3. Μονοπάτι ησιού

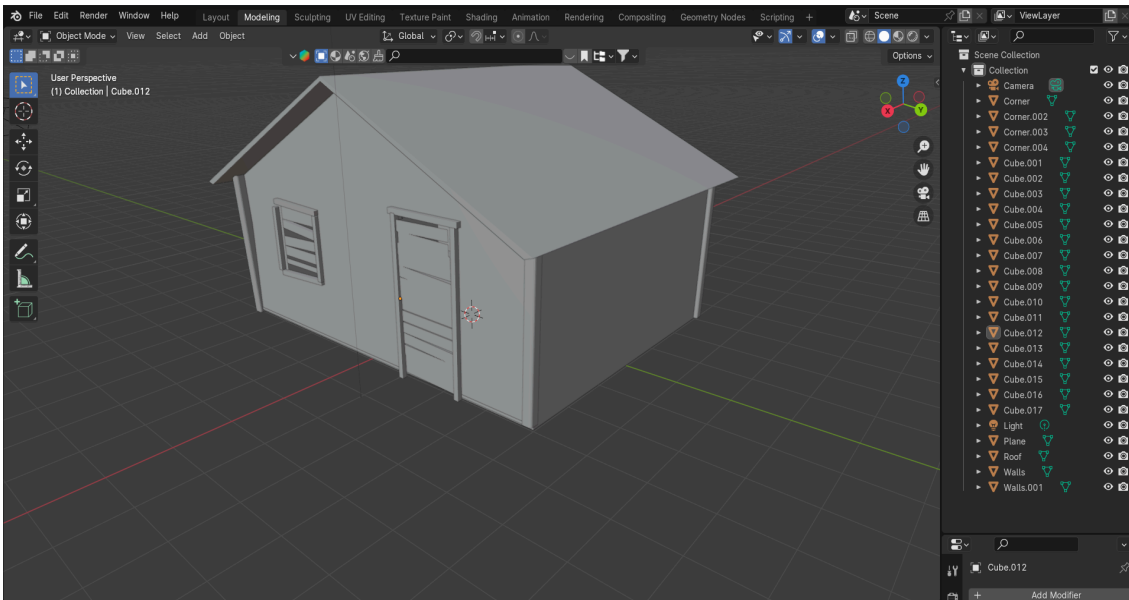


Εικόνα 4. Ηλιοβασιλέμα στο ησι

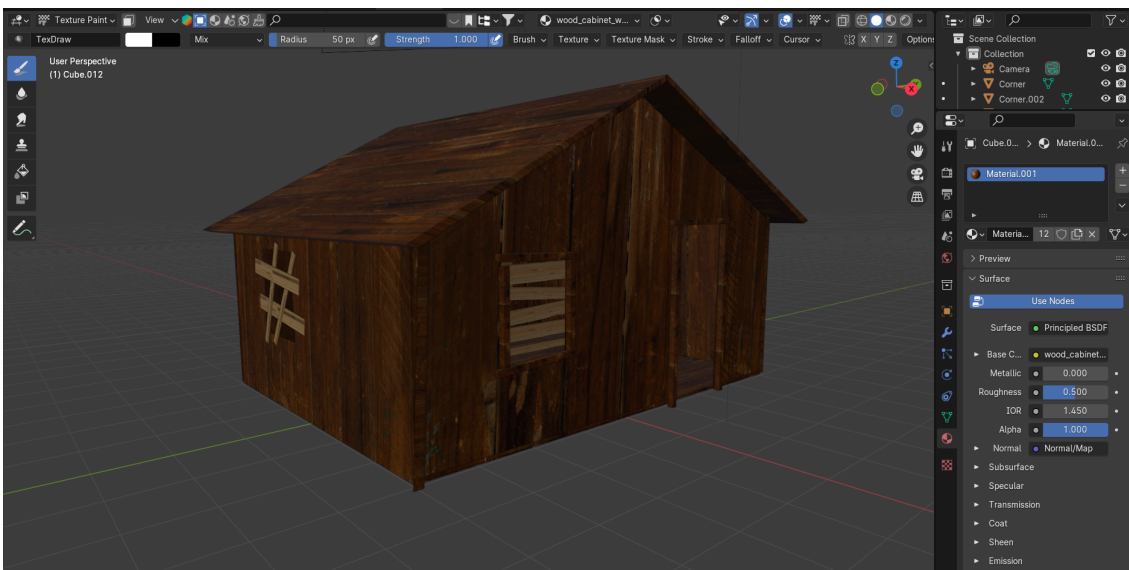


Εικόνα 5. Η παραλία του ησιού

Για τον σχεδιασμό της καλύβας, χρησιμοποιήσαμε το Blender ώστε να φτιάξουμε ακριβώς το μοντέλο που ταιριάζει στο παιχνίδι μας . Επιπλέον του βασικού σχεδίου, προσθέσαμε στα παράθυρα σανίδες, σχεδιάσαμε την πόρτα από λεπτά κομμάτια ορθογώνιων και κρατήσαμε στον σχεδιασμό μια ασυμμετρία. Με αυτό τον τρόπο καταφέραμε να μοιάζει περισσότερο φτιαγμένο από ανθρώπινο χέρι σε απομονωμένο νησί με περιορισμένους πόρους. Παρακάτω παρουσιάζουμε το τρισδιάστατο μοντέλο της καλύβας που σχεδιάσαμε (βλ. [Εικόνα 6.](#)), καθώς και το τελικό μοντέλο μετά την εφαρμογή του υλικού του ξύλου (βλ. [Εικόνα 7.](#)).



Εικόνα 6. Τρισδιάστατο Μοντέλο καλύβας



Εικόνα 7. Τελικό αποτέλεσμα καλύβας

Στη συνέχεια φορτώσαμε το αρχείο μας στο Unreal, τοποθετήσαμε την καλύβα στο κατάλληλο σημείο και προχωρήσαμε στον εσωτερικό της σχεδιασμό. Η τεράστια ποικιλία δωρεάν assets, μας έδωσε την δυνατότητα να δημιουργήσουμε μια εγκαταλελειμμένη ατμόσφαιρα με χειροποίητα έπιπλα και διακοσμητικά. Στις επόμενες εικόνες (βλ. [Εικόνα 8.](#) και [Εικόνα 9.](#)) βλέπουμε το τελικό αποτέλεσμα της καλύβας στο εσωτερικό της.



Εικόνα 8. Εσωτερικό καλύβας



Εικόνα 9. Εσωτερικό καλύβας άλλη οπτική γωνία

Dungeon Level

Το δεύτερο level που δημιουργήσαμε ήταν αυτό με τις κατακόμβες, το οποίο συνιστά το βασικό πυρήνα του gameplay.

Ξεκινώντας πάλι από ένα κενό βασικό level, οριοθετήσαμε με διαφορετικές διαστάσεις ορθογώνιων την περίμετρο του κτιρίου και τα διαφορετικά δωμάτια που απαιτούνταν. Για τα κατάλληλα πλαίσια όπου θα ενσωματώνονταν οι πόρτες, προσθέσαμε ορθογώνια και τα ορίσαμε ως αφαιρετικά (abstractive), δηλαδή να αφαιρούν τη συγκεκριμένη γεωμετρία και να δημιουργείται κενός χώρος. Επιπλέον, καθορίσαμε το collision για τους τοίχους ώστε να μην μπορεί ο παίκτης να περνάει από μέσα (BlockAll), ενώ για τα κενά των πορτών αφαιρέσαμε το collision ώστε να μπορεί να περάσει (NoCollision). Τέλος, αφού διαλέξαμε και ενσωματώσαμε τα materials που ταίριαζαν από το Quixel Bridge, προχωρήσαμε στον εσωτερικό σχεδιασμό των δωματίων.

Στο πρώτο δωμάτιο, δημιουργήσαμε ένα χώρο γραφείου. Προσθέσαμε δυο παλιές ραφιέρες γεμάτες με βιβλία, μια καρέκλα και ένα παλιό γραφείο, το οποίο στο συρτάρι του περιείχε το πρώτο στοιχείο που παιχνιδιού.

Στο δεύτερο δωμάτιο δημιουργήσαμε μία κρεβατοκάμαρα, η οποία περιέχει ένα κρεβάτι, ένα μικρό καθιστικό, μια βιβλιοθήκη με κάποια βιβλία και είδη γραφής, καθώς επίσης και κάποια ακόμα μικροαντικείμενα ώστε να θυμίζει υπνοδωμάτιο.

Το τρίτο δωμάτιο περιέχει σιδερένιες ντουλάπες, πιθάρια, κιβώτια και γενικά αντικείμενα που βρίσκει κανείς σε μια αποθήκη.

Το επόμενο δωμάτιο διαμορφώθηκε ως κρατητήρια. Προσθέσαμε δυο μεγάλα κελιά μέσα στα οποία κρέμονταν αλυσίδες και χειροπέδες. Επιπλέον, προστέθηκαν κάποια μικρά αντικείμενα, όπως κεριά και παλιά κουτιά, ώστε να προσδωθεί στο χώρο περισσότερη πυκνότητα και ρεαλισμός. Μέσα σε αυτό το δωμάτιο τοποθετήσαμε τον Felipe, που συναντούμε κατά τη διάρκεια του παιχνιδιού μας.



Εικόνα 10. Διάδρομος μπουντρουμιών

Στο τέλος του διαδρόμου (βλ. [Εικόνα 10.](#)), διαλέξαμε μια μεγάλη εντυπωσιακή πόρτα για να επισημαίνει την είσοδο στο Great Hall. Μέσα σε αυτό, προσθέσαμε πέτρινες ψηλές κολώνες, μεσαιωνικά αγάλματα, σκελετούς και πολυελαίους να κρέμονται από το ταβάνι. Σε αυτό το δωμάτιο τοποθετήσαμε τον Ezekiel, και φυλάσσει το τελευταίο δωμάτιο του level.



Εικόνα 11. Great Hall

Μέσα στο Great Hall (βλ. [Εικόνα 11.](#)), υπάρχει και η είσοδος για το τελευταίο δωμάτιο που δημιουργήσαμε στο παιχνίδι μας. Στο τελευταίο αυτό δωμάτιο προσθέσαμε δύο μεγάλα κλουβιά, μέσα σε ένα από τα οποία τοποθετήσαμε τον Tobias. Εμπλουτίσαμε το δωμάτιο με αλυσίδες και εργαλεία βασανισμού, ίχνη αίματος και σχοινιά, για να δημιουργήσουμε έναν χώρο βασανιστηρίων (βλ. [Εικόνα 12.](#)).



Εικόνα 12. Δωμάτιο Βασανιστηρίων

Τέλος, σε όλο το κτίριο επιστρέψαμε και προσθέσαμε τον κατάλληλο φωτισμό. Ο στόχος μας ήταν να διατηρηθεί η ατμόσφαιρα σκοτεινή και μυστηριώδης. Έτσι, τοποθετήσαμε σε μερικούς τοίχους αναμμένους πυρσούς, ενώ σε κάποια ακόμη σημεία ενισχύσαμε το φωτισμό με κεριά ή παλιές λάμπες. Επιπλέον, για ακόμη περισσότερη εμπύθιση στο περιβάλλον, τοποθετήσαμε σε διάφορα σημεία του κτιρίου ηχητικά assets που αναπαριστούν ήχους από μπουντρούμια.

Metahumans

Για τη δημιουργία των χαρακτήρων μας χρησιμοποιήσαμε το Metahuman Creator, ένα συνεργαζόμενο λογισμικό της Unreal Engine. Αυτό το εργαλείο δίνει τη δυνατότητα στους σχεδιαστές να δημιουργήσουν υψηλής ποιότητας, φωτορεαλιστικούς ψηφιακούς ανθρώπους ή χαρακτήρες.

Ένα από τα βασικά του πλεονεκτήματα ήταν εύχρηστη διεπαφή προς τον χρήστη, η οποία διευκόλυνε και τη δημιουργία των χαρακτήρων (Metahumans). Το Metahuman Creator προσφέρει μια εκτεταμένη σειρά επιλογών προσαρμογής των χαρακτήρων, όπως τη γλυπτική των χαρακτηριστικών του προσώπου, τη χρωματική παλέτα για ματια και τα μαλλιά, τον καθορισμό του σωματότυπου και επιλογή ρουχισμού. Ακόμη, οι Metahumans μπορούν εύκολα να ενσωματωθούν σε οποιοδήποτε project και συνοδεύονται με χαρτογραφημένο ανθρώπινο σκελετο, πράγμα που καθιστά απλούστερο το χειρισμό του σώματος και την εφαρμογή animations στα σώματα τους.

Στο δικό μας project χρησιμοποιήσαμε το Metahuman Creator για τη δημιουργία των τεσσάρων δικών μας χαρακτήρων και καταλήξαμε στην τελική τους μορφή που φαίνεται στην παρακάτω (βλ [Εικόνα 13.](#)).



Εικόνα 13. Metahumans

4.3. Blueprints

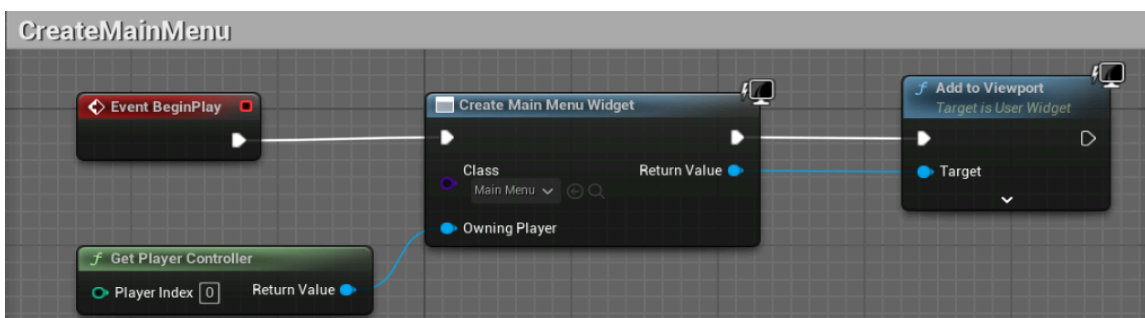
Το Blueprint Visual Scripting είναι ένα πλήρες σύστημα scripting που βασίζεται στη χρήση κόμβων για τη δημιουργία στοιχείων του παιχνιδιού μέσα στο Unreal Editor. Είναι ιδιαίτερα ισχυρό εργαλείο καθώς δίνει τη δυνατότητα στους σχεδιαστές να χρησιμοποιήσουν οπτικά σχεδόν όλα τα εργαλεία της μηχανής, ενώ το πιο δυνατό πλεονέκτημά τους είναι ότι επιτρέπουν πολύ γρήγορη δημιουργία κώδικα εντολών, εύκολα προσιτή και επεξηγηματική. Ουσιαστικά τα Blueprints με σύνδεση κόμβων, events, συναρτήσεων και μεταβλητών όπως ακριβώς σε ένα διάγραμμα ροής, είναι δυνατό να δημιουργηθούν πολύπλοκα στοιχεία παιχνιδιού. Στη συνέχεια παρουσιάζονται οι βασικές κατηγορίες Blueprints

Level Blueprints

Κάθε level έχει ένα δικό του Blueprint μέσα στο οποίο μπορεί να χειρίζεται στοιχεία του παιχνιδιού όπως τα cinematics, checkpoints, triggers κ.α.

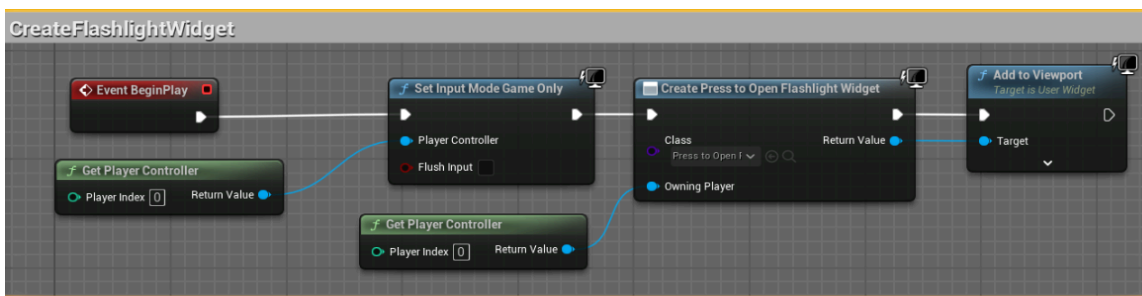
Στο βασικό μας level δημιουργήσαμε το παρακάτω blueprint, το οποίο υλοποιείται με την είσοδο στο Dungeon (ώστε να οργανώσουμε την ακολουθία του παιχνιδιού).

- **MainMenuTest:** Όταν ανοίγει το Level (Event BeginPlay), το οποίο δημιουργήθηκε για να στεγάσει το μενού έναρξης, δημιουργείται το widget MainMenu και εμφανίζεται στην οθόνη του παίκτη (βλ. [Εικόνα 14.](#)).



Εικόνα 14. MainMenuTest Event Graph

- **DungeonNew:** Με το που ανοίγει το Level (βλ. [Εικόνα 15.](#)), αμέσως μετά το κινηματογραφικό βίντεο (Event BeginPlay), το παιχνίδι τίθεται σε λειτουργία Game Only και δημιουργείται το widget Press to Open Flashlight το οποίο και εμφανίζεται στην οθόνη του παίκτη και περιέχει ένα κείμενο “F for flashlight”.



Εικόνα 15. DungeonNew Event Graph

Blueprint Class Actor

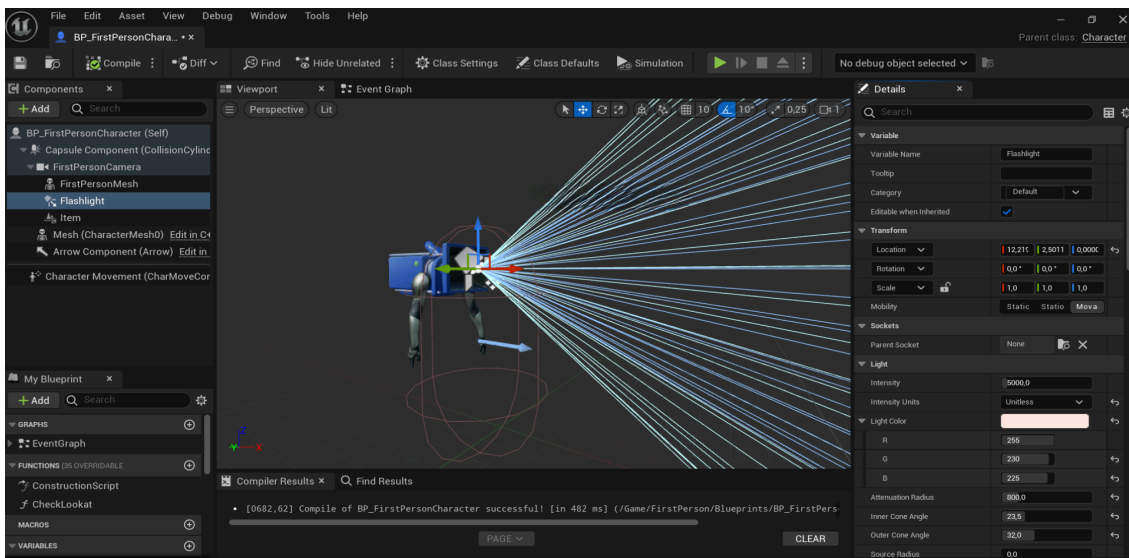
Είναι ένα θεμελιώδες στοιχείο που χρησιμεύει για την προσθήκη λειτουργικότητας και διαδραστικότητας σε στοιχεία του παιχνιδιού. Ουσιαστικά, με αυτή την κατηγορία blueprint οι προγραμματιστές μπορούν να ορίσουν τη συμπεριφορά και τις ιδιότητες των στοιχείων, καθώς επίσης και τις αλληλεπιδράσεις μεταξύ παίκτη και αντικειμένων.

Οι Blueprint Classes επιτρέπουν τη δημιουργία λογικής και λειτουργικότητας μέσω οπτικών interfaces, μπορούν να συνδεθούν με άλλες ή να κληρονομήσουν στοιχεία και ιδιότητες και, τέλος, μπορούν να διαμορφωθούν καλώντας εξατομικευμένα συμβάντα.

Κατά τον σχεδιασμό του δικού μας παιχνιδιού, αξιοποιήσαμε σε βάθος τις ευκαιρίες που προσφέρονται από τα blueprint classes. Παρακάτω παρουσιάζεται πως σχεδιάσαμε τους χαρακτήρες, τις πόρτες και άλλα αντικείμενα με τα οποία μπορεί να αλληλεπιδράσει ο παίκτης.

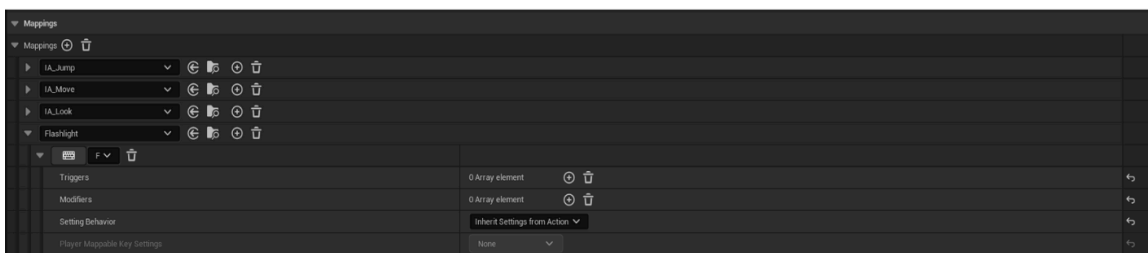
Flashlight

Ξεκινώντας από το Blueprint του First Person, προσθέσαμε από το viewport ένα spotlight στο ύψος της κάμερας, όπως φαίνεται στην παρακάτω εικόνα (βλ. [Εικόνα 16.](#)). Από το Details panel ορίσαμε από προεπιλογή να είναι άορατο, προσαρμόσαμε το χρωματισμό του φωτός ώστε να είναι πιο φυσικό, και ορίσαμε την ακτίνα που το φως θα είναι έντονο (Inner Cone Angle) και την ακτίνα διάχυσης του φωτός εξωτερικά (Outer Cone Angle).



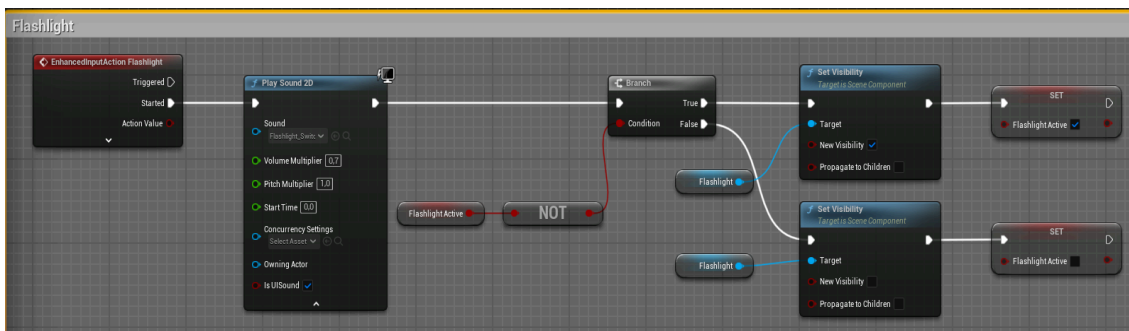
Εικόνα 16. Ρύθμιση φακού στο First Person

Στη συνέχεια, δημιουργήσαμε μια ενέργεια για τον φακό, ώστε να μπορούμε να το προσδιορίσουμε στο input mapping του first person (βλ. [Εικόνα 17.](#)), δηλαδή να μπορούμε να χαρτογραφίσουμε ένα συγκεκριμένο κουμπί εισόδου για να μπορεί να το χειρίζεται ο παίκτης. Αφού ορίσαμε το κουμπί που συνδέεται με το φακό να είναι το “F”, επιστρέψαμε στο blueprint για να προσδώσουμε λειτουργικότητα σε αυτό.



Εικόνα 17. Mapping First Person “F”

Στο event graph, ορίσαμε μια Boolean μεταβλητή (Flashlight Active). Όταν πατηθεί το κουμπί “F” (βλ. [Εικόνα 18.](#)) που προσδιορίσαμε (EnhanceInputAction Flashlight), ακούγεται ένας ήχος κλείστρου (Play Sound 2D) και γίνεται έλεγχος (Branch) αν η μεταβλητή είναι αληθής, δηλαδή ο φακός είναι ενεργός, ή όχι. Ανάλογα με την κατάσταση του φακού τότε γίνεται ορατός ή όχι (Set visibility) και η μεταβλητή γίνεται εκ νέου αληθής ή ψευδής.

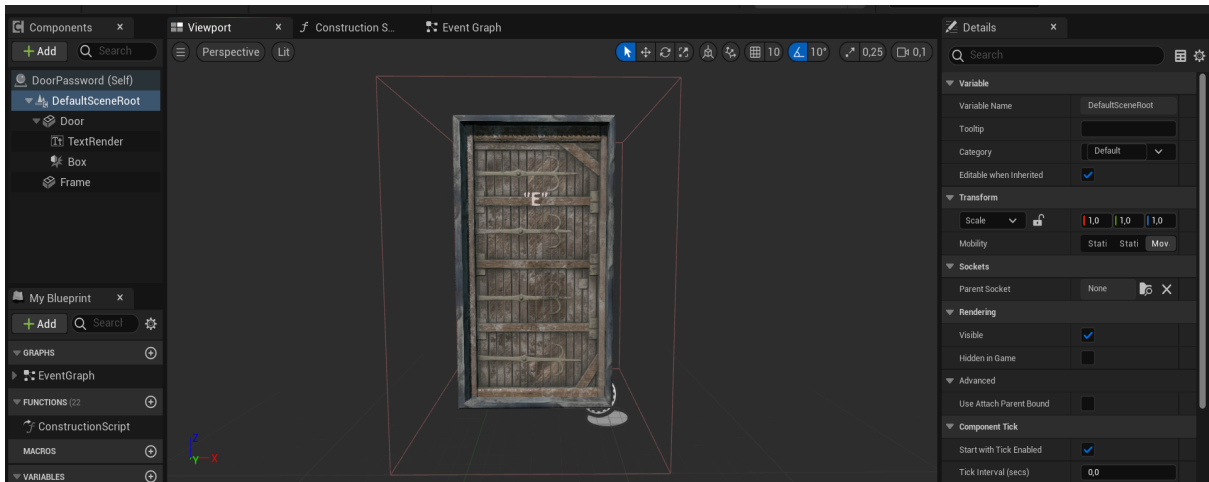


Εικόνα 18. Flashlight Event Graph

Doors

Οι πόρτες που συναντάμε στο παιχνίδι μας, αποτελούν τους βασικούς γρίφους που καλείται να λύσει ο παίκτης. Ο εξωτερικός σχεδιασμός στις εξωτερικές πόρτες είναι ο ίδιος για να υπάρχει μια ομοιομορφία, ενώ για την πόρτα που καταλήγει στο great hall διαλέξαμε μια μεγάλη εντυπωσιακή διπλή πόρτα. Κάθε μια από αυτές τις πόρτες καταλήγει στα δωμάτια τα οποία περιέχουν είτε στοιχεία για τη λύση του μυστηρίου, είτε στοιχεία για να ανοιχτούν οι επόμενες πόρτες, όπως θα δούμε στη συνέχεια.

Αρχικά προσθέσαμε το 3D μοντέλο της πόρτας (StaticMesh), ένα πλαίσιο (Box Collision) για να προσδιορίσουμε την ελάχιστη απόσταση που πρέπει να έχει ο παίκτης για να ενεργοποιηθεί η αλληλεπίδραση με την πόρτα (βλ. [Εικόνα 19.](#)), και ένα δυναμικό “E” (text render), τη λειτουργικότητα του οποίου ρυθμίζουμε στο blueprint.

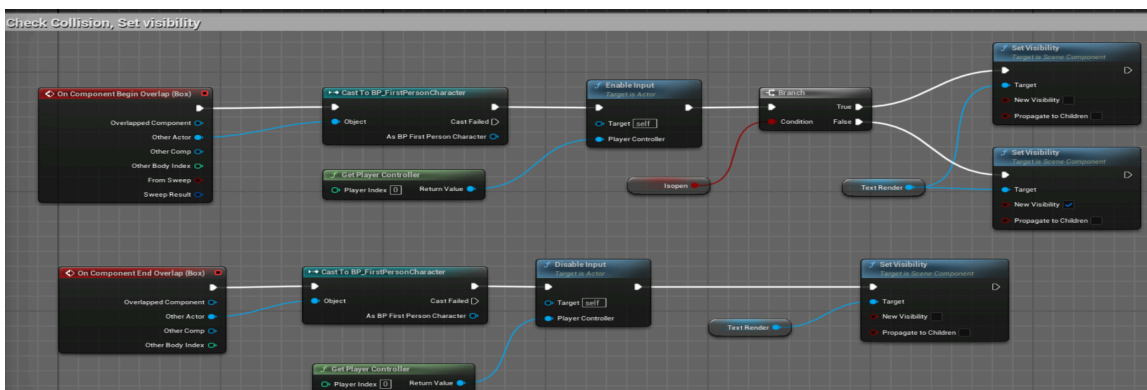


Εικόνα 19. Viewport Doors

Ξεκλειδωτή πόρτα (UnlockedDoor)

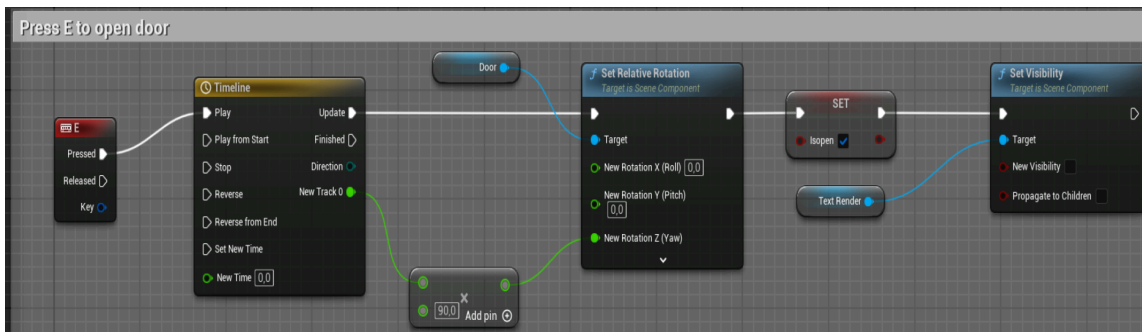
Πρόκειται για την πρώτη πόρτα που συναντάει ο παίκτης κατά την είσοδο στις κατακόμβες. Για την συγκεκριμένη πόρτα δεν θέσαμε κάποιο περιορισμό για το άνοιγμα της και οδηγεί στο δωμάτιο γραφείο που αναφέραμε προηγουμένως.

Στο event graph δημιουργούμε μια Boolean μεταβλητή (Isopen). Ελέγχουμε αν ο παίκτης βρίσκεται μέσα στο box collision (On Component Begin/End Overlap) (βλ. [Εικόνα 20.](#)), ενεργοποιούμε τα πλήκτρα του παίκτη (Enable/Disable Input), και ρυθμίζουμε την ορατότητα του Text Render αν ο παίκτης είναι εκτός του box, ή εντός αλλά έχει ήδη ανοίξει την πόρτα και η μεταβλητή Isopen έχει γίνει αληθής.



Εικόνα 20. Unlocked Door Event Graph (Collision and Visibility)

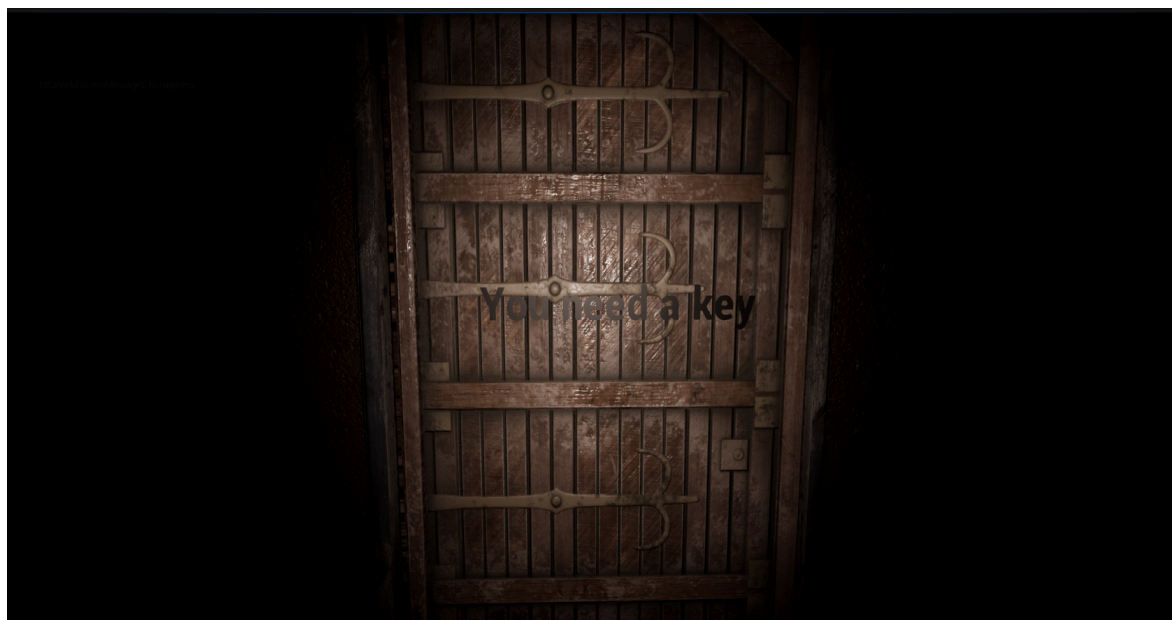
Επιπλέον, όταν ο παίκτης πατήσει το πλήκτρο “E” (βλ. [Εικόνα 21](#)) ορίζουμε το χρονικό διάστημα της κίνησης της πόρτας (Timeline) και θέτουμε να κάνει περιστροφή 90° στον άξονα z (Set Relative Rotation). Τέλος, αλλάζουμε τη τιμή της Isopen σε αληθής (Set) και απενεργοποιούμε την ορατότητα του Text Render. Έτσι όταν ο παίκτης ανοίξει την πόρτα μια φορά το Text Render γίνεται μη ορατό ακόμη και αν αλληλεπιδρά με το box collision.



Εικόνα 21. Unlocked Door Event Graph (Player Interaction)

Πόρτα με Κλειδί (DoorKey)

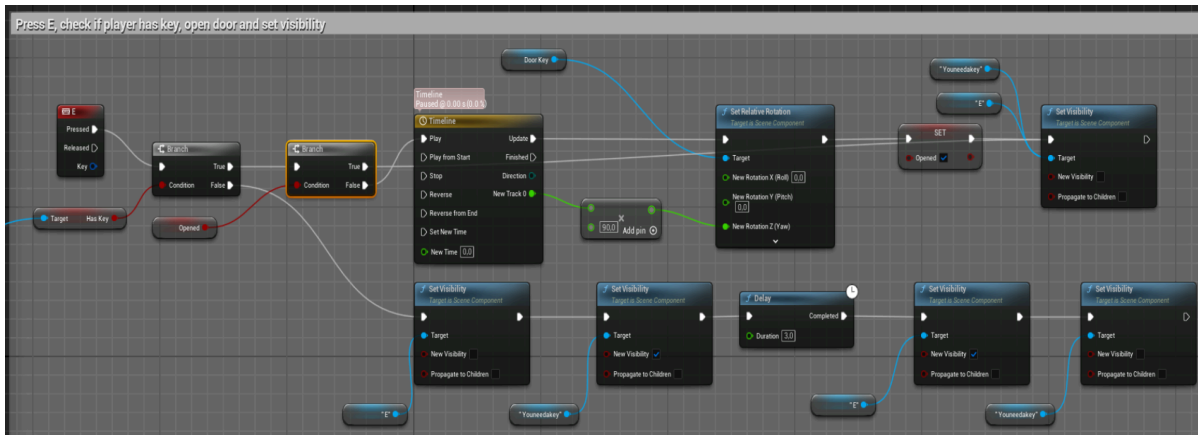
Η δεύτερη πόρτα που σχεδιάσαμε είναι η πόρτα που για να ανοίξει απαιτείται ο παίκτης να έχει βρεί το κλειδί (βλ. [Εικόνα 22](#)). Αρχικά, στο blueprint του First Person δημιουργήσαμε μια Boolean μεταβλητή (HasKey) με αρχική τιμή ψευδής, η οποία γίνεται αληθής μόνο αν ο παίκτης έχει σηκώσει το κλειδί. Στο Event graph της πόρτας η αρχική διαδικασία για τον έλεγχο της αλληλεπίδρασης του παίκτη με το box collision είναι η ίδια, καθώς επίσης και ο καθορισμός της ορατότητας του Text Render (“E”).



Εικόνα 22. DoorKey

Επιπλέον προσθέσαμε ένα ακόμη Text Render (“You need a key”) το οποίο αρχικά είναι μη ορατό. Αν ο παίκτης βρίσκεται μέσα στο box collision και πατήσει το “E”, κάνουμε έλεγχο την τιμή της HasKey (βλ. [Εικόνα 23](#)). Αν είναι αληθής, δηλαδή ο

παίκτης έχει το κλειδί, κάνουμε δεύτερο έλεγχο αν η πόρτα έχει ήδη ανοίξει. Αν όχι, τότε η πόρτα ανοίγει. Αν ο παίκτης δεν έχει το κλειδί, τότε στη θέση του “E” εμφανίζεται το “You need a key”, και μετά από 3 δευτερόλεπτα (Delay) επιστρέφουν στην αρχική κατάσταση.



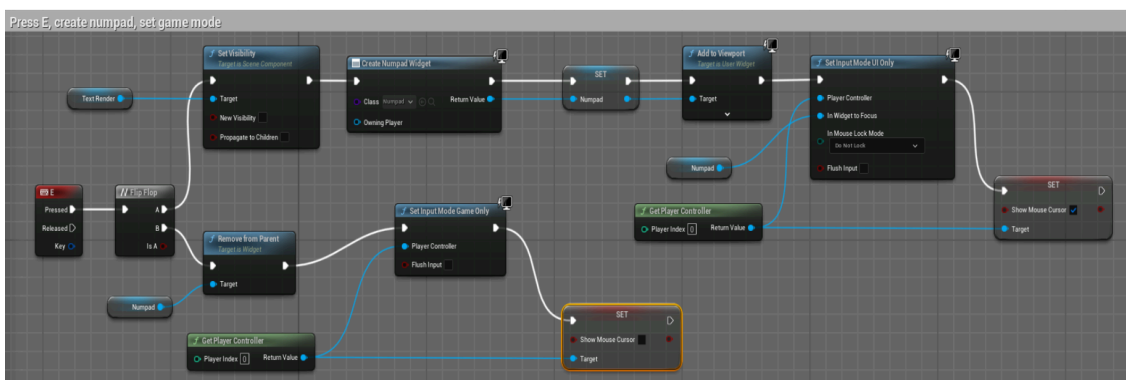
Εικόνα 23. DoorKey Event Graph

Πόρτα με Κωδικό (DoorCode)

Η επόμενη πόρτα ανοίγει μόνο με την προϋπόθεση ότι ο παίκτης γνωρίζει τον κατάλληλο συνδυασμό αριθμών. Για την υλοποίηση αυτού του γρίφου και του πλαισίου που ο παίκτης πληκτρολογεί τον κωδικό, δημιουργήσαμε και ένα widget blueprint (Numpad) που θα δούμε παρακάτω.

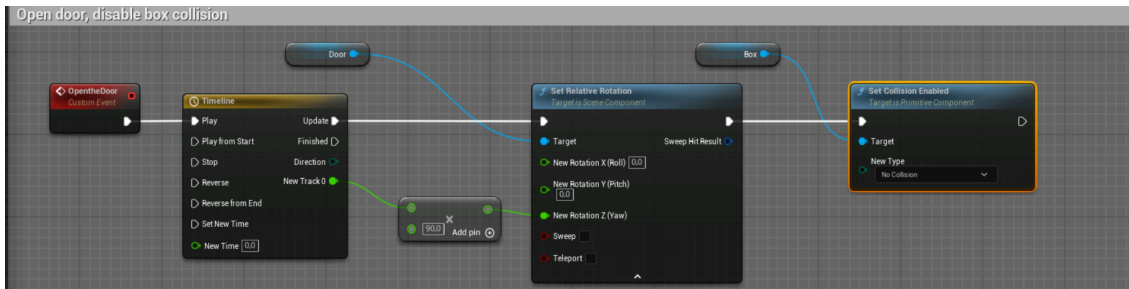
Το αρχικό κομμάτι και για αυτή την πόρτα παραμένει το ίδιο, δηλαδή γίνεται έλεγχος αν ο παίκτης βρίσκεται εντός του box collision και γίνεται ορατό ή αόρατο αντίστοιχα το “E” που βλέπει ο παίκτης.

Αν ο παίκτης βρίσκεται εντός του box collision και πατήσει το πλήκτρο “E” (βλ. [Εικόνα 24.](#)), τότε δημιουργείται το widget με το Numpad (Create Numpad Widget) και εμφανίζεται στην οθόνη του (Add to viewport). Επιπλέον, παιχνίδι τίθεται σε Mode UI και εμφανίζεται ο κέρσορας ώστε να μπορεί να αλληλεπιδράσει με το Numpad (Set Show Mouse Cursor). Η χρήση της Flip Flop στην αρχή του διαγράμματος δίνει την δυνατότητα κάθε φορά που ο χρήστης πατάει το πλήκτρο “E”, να εναλλάσσει τον κώδικα που θα τρέξει μεταξύ A ή B. Έτσι, κάθε φορά που ο παίκτης πατάει το “E” το widget εμφανίζεται στην οθόνη ή εξαφανίζεται (Remove from Parent).



Εικόνα 24. DoorCode Event Graph (Interaction)

Αφού ο παίκτης καταφέρει να πληκτρολογήσει το σωστό κωδικό, από το Event Graph του Widget ενεργοποιείται ένα συμβάν (Event trigger) το OpenTheDoor. Όταν κληθεί αυτό το συμβάν (βλ. [Εικόνα 25.](#)), τότε η πόρτα ανοίγει και το Box Collision γίνεται ανενεργό (Set Collision Enabled) ώστε να μην μπορεί πλέον ο παίκτης να αλληλεπιδράσει με την πόρτα, και κατ' επέκταση να δημιουργηθεί ξανά το Widget Numpad.

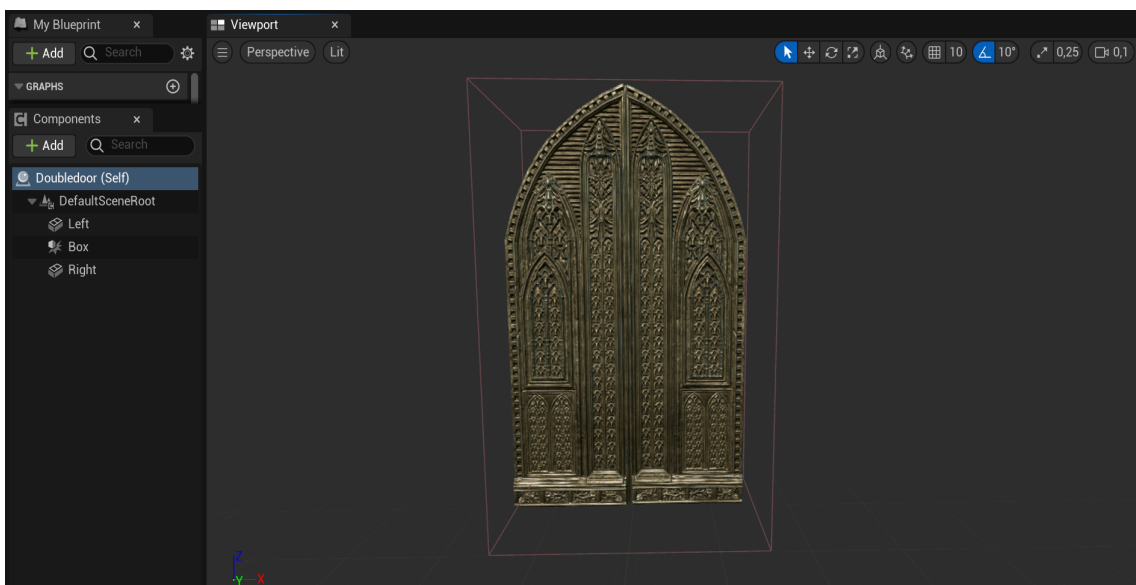


Εικόνα 25. DoorCode Event Graph (Open the Door Event)

Διπλή πόρτα Great Hall (Double Door)

Αποτελεί την κεντρική πόρτα που οδηγεί στο Great Hall. Η πόρτα αυτή ανοίγει μόνο με την προϋπόθεση ότι ο παίκτης γνωρίζει την κατάλληλη αλληλουχία δεξί/αριστερό κουμπί. Για την υλοποίηση αυτού του γρίφου και του πλαισίου που ο παίκτης εισάγει την αλληλουχία, δημιουργήσαμε και ένα widget blueprint (βλ [ButtonSequenceDoorWidget](#)) που θα δούμε παρακάτω.

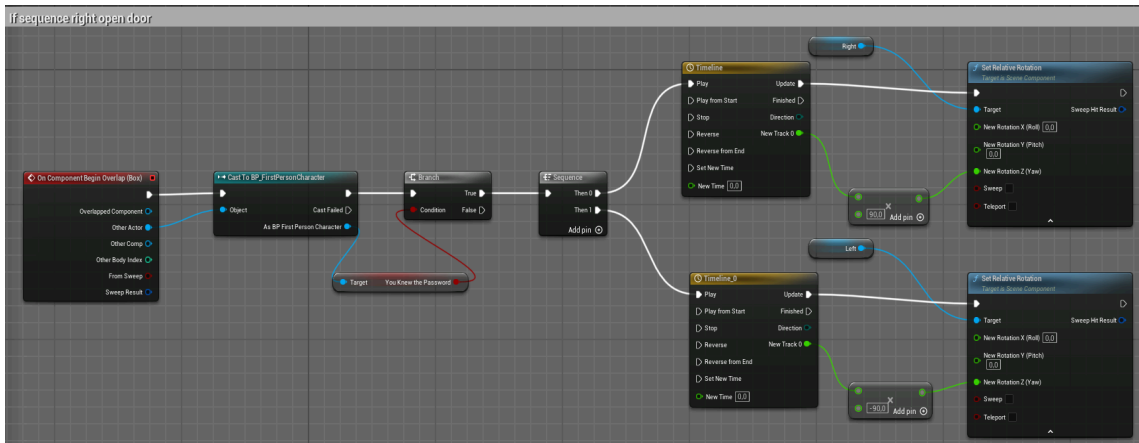
Στο Viewport αυτής της πόρτας ορίσαμε το δεξί και αριστερό κομμάτι της πόρτας καθώς επίσης και ένα box collision (βλ. [Εικόνα 26.](#)). έτσι , στη συνέχεια μπορούμε να διαχειριστούμε τη στροφή κάθε κομματιού ξεχωριστά.



Εικόνα 26. Viewport DoubleDoor

Στο Event Graph της πόρτας, όταν ο παίκτης βρεθεί εντός του box collision (βλ. [Εικόνα 27.](#)) γίνεται έλεγχος της μεταβλητής “You knew the password” (γίνεται αληθής ή ψευδής στο widget του κρανίου). Αν η τιμή είναι αληθής και παίκτης μάντεψε σωστά

τον κωδικό της πόρτας, τότε τίθεται σε λειτουργία ένα sequence, και κάθε φύλλο της πόρτας (left,right) ανοίγει κατά 90° αριστερόστροφα και δεξιόστροφα αντίστοιχα.



Εικόνα 27. DoubleDoor Event Graph

Στη παρακάτω εικόνα (βλ. [Εικόνα 28.](#)) φαίνεται η διπλή πόρτα που οδηγεί στο Great Hall κατά τη διάρκεια του Gameplay.



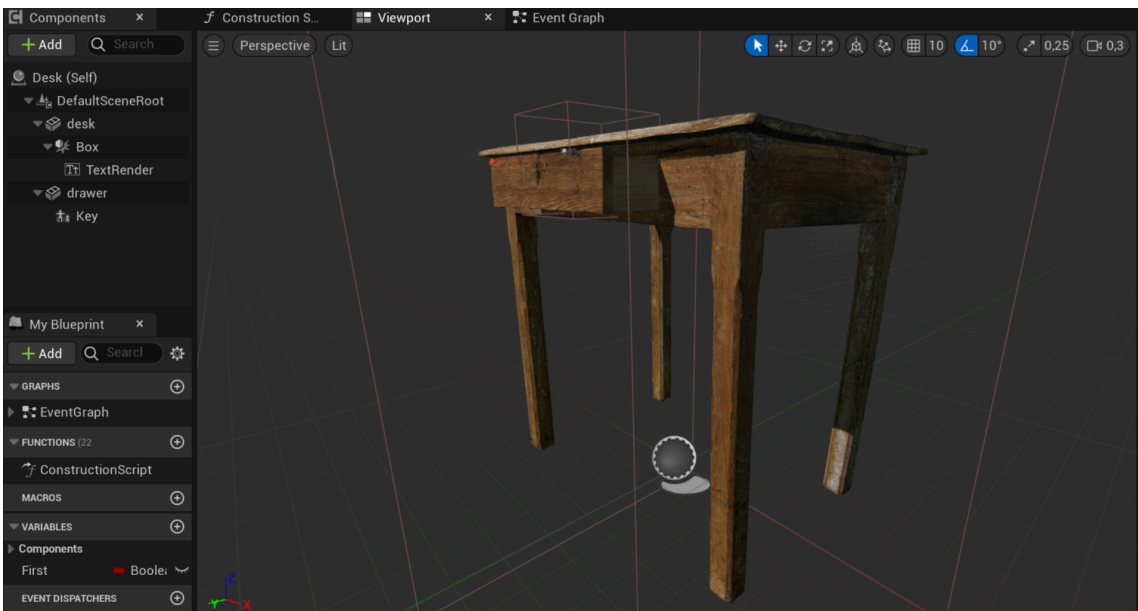
Εικόνα 28. DoubleDoor Τελικό αποτέλεσμα

Assets

Στο Unreal Engine, τα assets είναι τα δομικά στοιχεία των εικονικών κόσμων και μπορούν να περιλαμβάνουν από τρισδιάστατα μοντέλα και υφές μέχρι κινούμενα σχέδια, ηχητικά εφέ και πολλά άλλα.

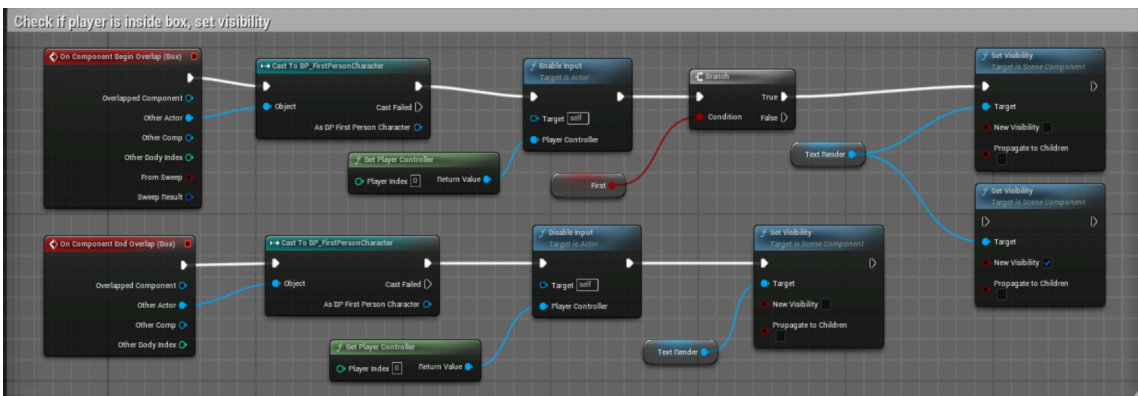
Γραφείο με Συρτάρι

Μέσα στο δωμάτιο γραφείο που εισέρχεται ο παίκτης από την ξεκλειδωτή πόρτα, έχουμε τοποθετήσει ένα παλιό γραφείο με συρτάρι που βρήκαμε από το Quixel Bridge. Τοποθετήσαμε γύρω του ένα box collision (βλ. [Εικόνα 29.](#)), ώστε να μπορεί να αλληλεπιδράσει ο παίκτης και ένα Text Render για το γράμμα "E" της αλληλεπίδρασης.



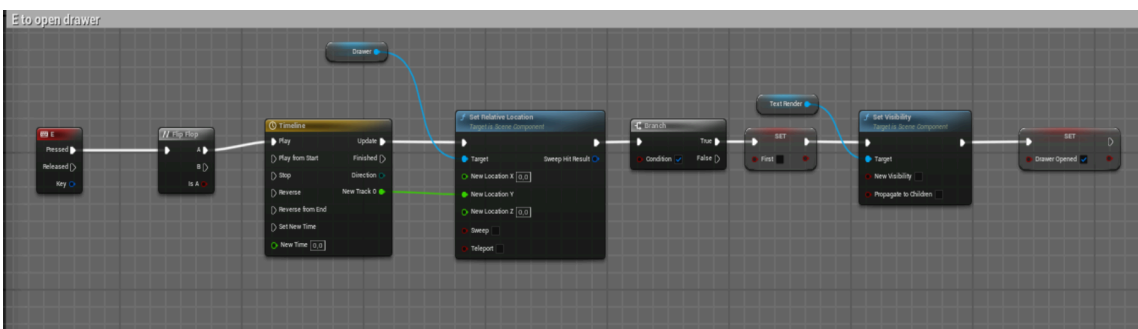
Εικόνα 29. Viewport Desk

Στο event graph, (βλ. [Εικόνα 30.](#)) δημιουργήσαμε μια Boolean μεταβλητή (First) για να μπορούμε να καθορίσουμε αν το συρτάρι έχει ανοιχτεί και ορίσαμε τον έλεγχο αν ο παίκτης βρίσκεται εντός του box collision. Αν ναι, τότε το γίνεται έλεγχος αν το συρτάρι έχει ανοιχτεί και εμφανίζεται ή εξαφανίζεται το Text της αλληλεπίδρασης αντίστοιχα.



Εικόνα 30. Desk Event Graph (Collision)

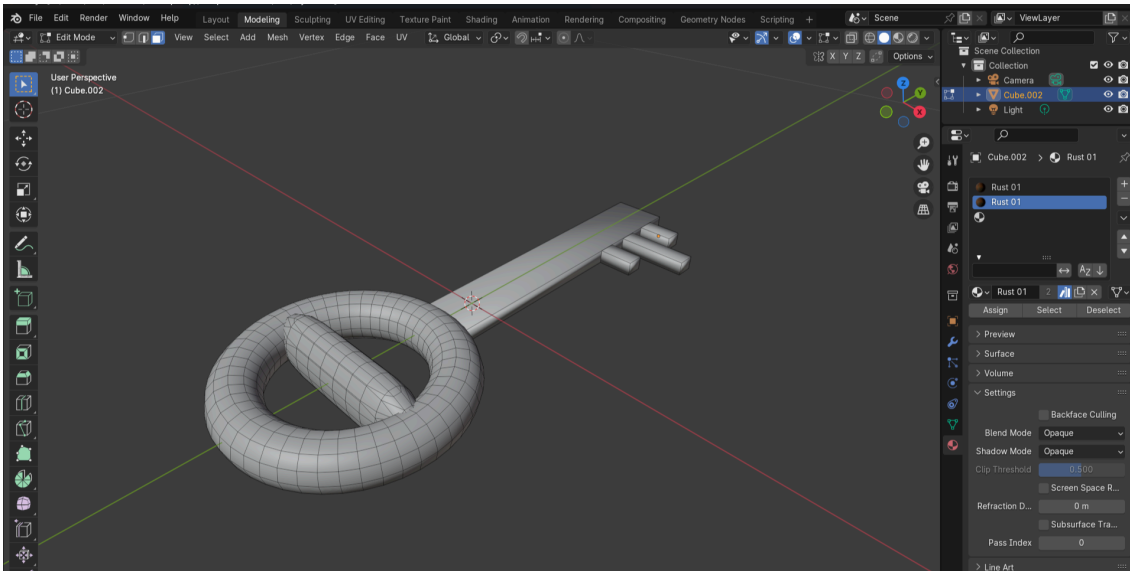
Αν ο παίκτης βρίσκεται εντός του box και πατήσει το E (βλ. [Εικόνα 31.](#)), τότε ενεργοποιείται το άνοιγμα του συρταριού (Set Relative Location) και η μεταβλητή γίνεται αληθής ώστε το “E” να σταματήσει να εμφανίζεται στον παίκτη.



Εικόνα 31. Desk Event Graph (Interaction)

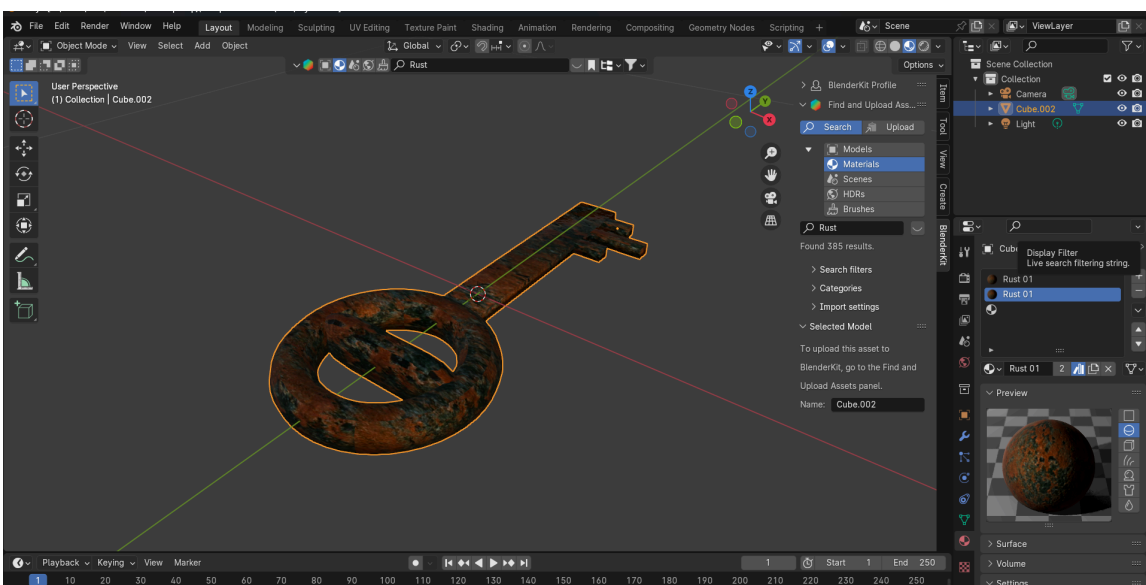
Κλειδί (Key)

Μέσα στο συρτάρι βρίσκεται το πρώτο στοιχείο που βρίσκει ο παίκτης, ένα κλειδί που χρησιμοποιείται για το άνοιγμα της KeyDoor. Το συγκεκριμένο κλειδί το σχεδιάσαμε στο Blender. Για το 3D μοντέλο (βλ. [Εικόνα 32.](#)) χρησιμοποιήσαμε έναν τόρο, ένα ορθογώνιο και μερικούς κυλίνδρους, και διαμορφώνοντας κατάλληλα τις διαστάσεις τους καταλήξαμε στο αποτέλεσμα της παρακάτω εικόνας. Τέλος, κάναμε μια σύνδεση σε όλα τα σχήματά μας, και δημιουργήσαμε ένα μεμονωμένο αντικείμενο, ώστε να μπορούμε κατα την εισαγωγή του στο Unreal Engine να το διαχειριστούμε σαν ένα ενιαίο asset.



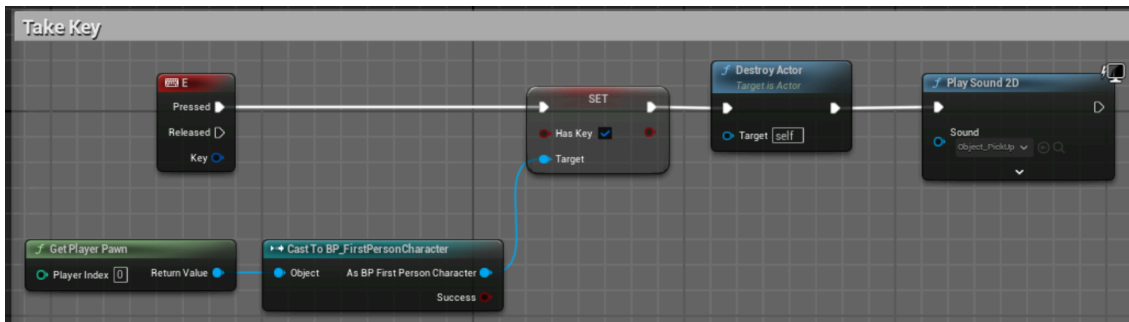
Εικόνα 32. 3D Key Model

Μετά το σχεδιασμό, αναζητήσαμε το material (βλ. [Εικόνα 33.](#)) που εξυπηρετούσε τις ανάγκες του παιχνιδιού από το BlenderKit (επέκταση του Blender με εκτεταμένη ποικιλία δωρεάν 3D μοντέλων και materials), και μετά την εφαρμογή του, καταλήξαμε στο παρακάτω μοντέλο κλειδιού.



Εικόνα 33. 3D key Model with Rust

Αφού εισάγαμε το μοντέλο στο Project μας, δημιουργήσαμε ένα Blueprint Actor ώστε να μπορούμε να επιτύχουμε αλληλεπίδραση με τον παίκτη. Αρχικά ορίσαμε ένα Box Collision και ένα text “E” με την ίδια λειτουργικότητα που είχαν και στα προηγούμενα Event Graphs. Όταν ο παίκτης πατήσει το “E” (βλ. [Εικόνα 34.](#)), τότε η μεταβλητή HasKey γίνεται αληθής και το κλειδί εξαφανίζεται από την εικόνα του παίκτη (DestroyActor).



Εικόνα 34. Key Event Graph (Interaction)

Τελικά καταλήξαμε στο αποτέλεσμα της παρακάτω εικόνας (βλ. [Εικόνα 35.](#)).



Εικόνα 35. Desk and Key Τελικό αποτέλεσμα

Φωτογραφία (PhotoFrame)

Το επόμενο στοιχείο που προσθέσαμε στο παιχνίδι μας είναι η φωτογραφία (βλ. [Εικόνα 36.](#)) του πατέρα του Ezekiel. Ο παίκτης μπορεί να τη σηκώσει και στο πίσω μέρος να βρει τη χρονολογία/ κωδικό που χρησιμοποιείται για το άνοιγμα της DoorCode. Για τη δημιουργία του συγκεκριμένου Actor χρησιμοποιήσαμε έναν κύβο, αλλάζοντας τις διαστάσεις του οποίου καταφέραμε να μοιάζει με κορνίζα. Στη συνέχεια, προσθέτοντας μια φωτογραφία δημιουργήσαμε ένα material το οποίο εφαρμόσαμε στην επιφάνεια του κύβου. Επιπλέον, με τη χρήση του Box Collision και του Text Render, δώσαμε λειτουργικότητα στην κορνίζα. Τέλος, με τη δημιουργία ενός User Interface (BPI_Photo) που περιέχει μια συνάρτηση (ReadPhoto) καταφέραμε να συνδέσουμε το asset με τον First Person. Μέσω της σύνδεσης αυτής καλείται ένα Widget (Photo) που επιτρέπει στον παίκτη να δει το μπροστά και πίσω μέρος της φωτογραφίας.



Εικόνα 36. PhotoFrame Τελικό αποτέλεσμα

Φάκελος (Folder)

Με τον ίδιο ακριβώς τρόπο δημιουργίας του PhotoFrame, δημιουργήσαμε το asset του φακέλου (Folder) (βλ. [Εικόνα 37.](#)), το οποίο συνδέεται επίσης με ένα UI (BPI_Sailors) που περιέχει τη συνάρτηση Sail, και το widget Open_folder, ώστε ο παίκτης να περιηγηθεί στα περιεχόμενα του φακέλου. Αυτά περιέχουν πληροφορίες για κάποια από τα θύματα του Ezekiel.



Εικόνα 37. Folder Τελικό αποτέλεσμα

Γράμμα (Letter)

Ομοίως, δημιουργήσαμε και το γράμμα (Letter) (βλ. [Εικόνα 38.](#)) το οποίο χρησιμεύει για την αναγνώριση του μοτίβου Right/Left που πρέπει ο παίκτης να εισάγει για το άνοιγμα της πόρτας DoubleDoor. Είναι συνδεδεμένο, εξίσου, με ένα UI (BPI_Letter) που περιέχει τη συνάρτηση Letter Function, και το widget Letter, με το οποίο εμφανίζεται η εικόνα του γράμματος στην οθόνη.

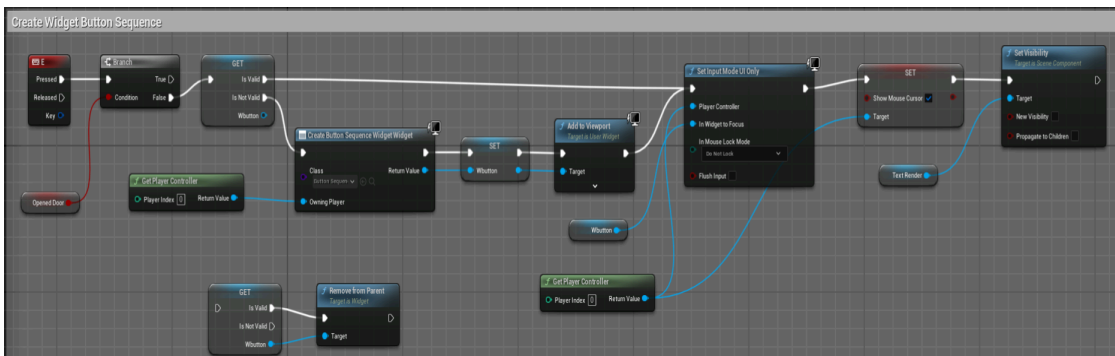


Εικόνα 38. Letter Τελικό αποτέλεσμα

Κρανίο (Skull)

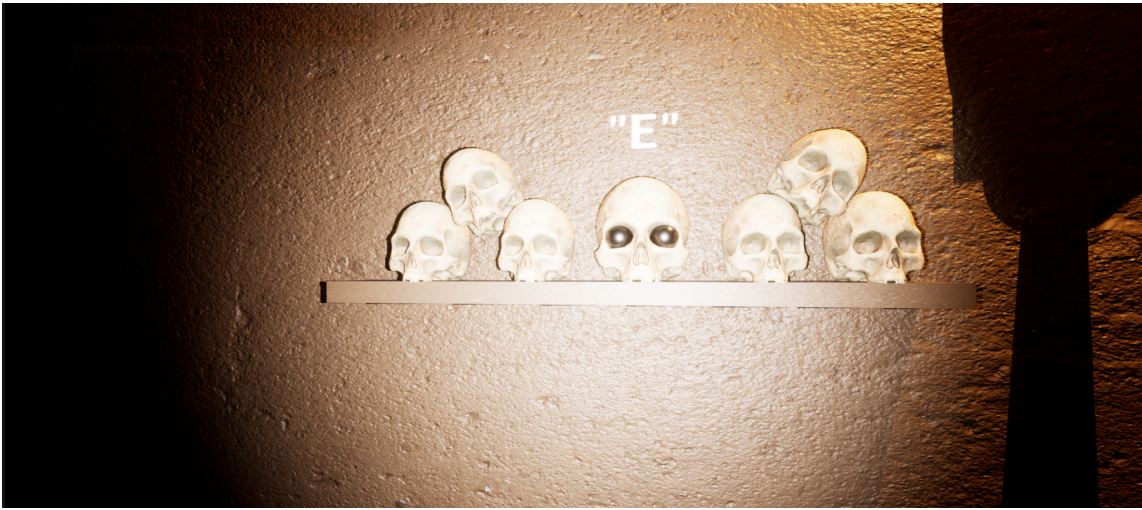
Τελευταίο Actor, είναι το κρανίο (βλ. [Εικόνα 40.](#)) που βρίσκεται δίπλα από τη Double Door. Με αυτό το asset αλληλεπιδρά ο παίκτης και δημιουργείται το widget ButtonSequence που προστίθεται στην εικόνα του, και είναι αυτό που επιτρέπει στον χρήστη να εφαρμόσει τη σωστή ακολουθία για να ανοίξει η πόρτα.

Όπως φαίνεται στο παρακάτω διάγραμμα του κρανίου, όταν ο παίκτης πατήσει το πλήκτρο “E” (βλ. [Εικόνα 39.](#)), γίνεται έλεγχος για το αν η πόρτα έχει ανοίξει ήδη ή όχι, και αν δεν έχει ανοίξει, τότε δημιουργείται το αντίστοιχο Widget και το παιχνίδι τίθεται σε λειτουργία UI only με ορατό κέρσορα.



Εικόνα 39. Skull Event Graph (Interaction)

Στην παρακάτω εικόνα, φαίνεται το asset του κρανίου (βλ. [Εικόνα 40.](#)) κατά τη διάρκεια του Gameplay.



Εικόνα 40. Skull Τελικό αποτέλεσμα

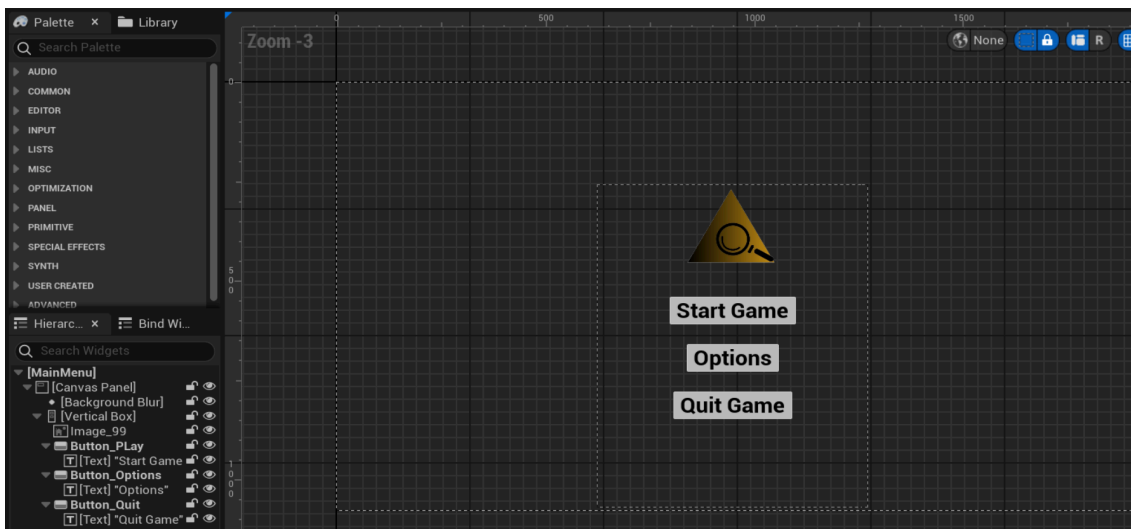
Widget Blueprints

Στα περισσότερα παιχνίδια, θέλει κανείς να μοιραστεί κάποιες πληροφορίες με τους παίκτες μέσω μιας διεπαφής χρήστη (User Interface) του παιχνιδιού. Αυτό μπορεί να περιλαμβάνει πράγματα όπως κύρια μενού, μενού παύσης εντός του παιχνιδιού, στοιχεία του HUD όπως η υγεία ή η εμπειρία, στοιχεία απογραφής ή κείμενο βοήθειας για να καθοδηγήσει τους παίκτες τι πρέπει να κάνουν σε μια δεδομένη κατάσταση. Στο Unreal Engine μπορεί να δημιουργηθεί ένα Widget Blueprint το οποίο και χειρίζεται την εμφάνιση των στοιχείων UI στο παιχνίδι. Το Widget Blueprint χρησιμοποιεί το Blueprint Visual Scripting για να σχεδιάσει τη διάταξη (design layout) καθώς και τη λειτουργικότητα σεναρίου (script functionality) μέσω του graph για τα στοιχεία User Interface (UI), όπως το τι συμβαίνει όταν γίνεται κλικ σε ένα κουμπί ή αν αλλάζει μια τιμή.

Στο συγκεκριμένο παιχνίδι έχουν δημιουργηθεί UIs σε διάφορες φάσεις του παιχνιδιού. Κάποιες από αυτές είναι το μενού έναρξης, μενού παύσης, μενού επιλογών, διάλογοι με NPC, όπως επίσης και κάποιοι γρίφοι. Παρακάτω αναλύονται βήμα-βήμα τα στάδια υλοποίησης για κάθε widget ξεχωριστά.

Μενού Έναρξης (MainMenu)

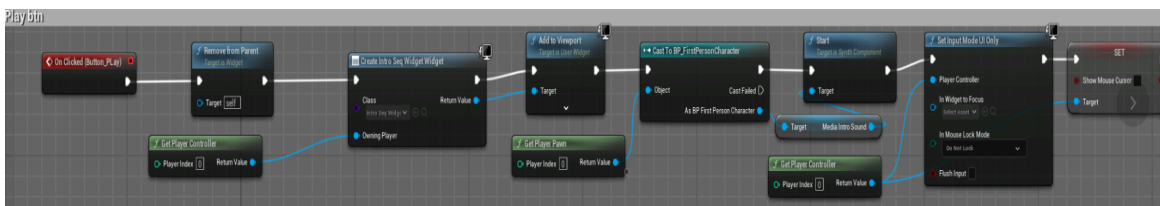
Η διάταξη designer του MainMenu (βλ. [Εικόνα 41.](#)) περιλαμβάνει μέσα στο canvas panel ένα κουτί για οριοθέτηση της διάταξης (vertical box) μέσα στο οποίο υπάρχουν το logo (μορφή image), κουμπί (Button_Play) με text “Start Game”, κουμπί (Button_Options) με text “Options”, κουμπί (Button_Quit) με text “Quit Game”. Η εφαρμογή του MainMenu γίνεται στο τρίτο level που σχεδιάσαμε μόνο για αυτό το σκοπό. Έτσι, διαμορφώνοντας τον χώρο κατάλληλα, καταφέραμε το background του widget να θυμίζει διάδρομο από τις κατακόμβες, με αναμμένους πυρσούς και προσθέσαμε ήχους που θυμίζουν μπουντρούμι.



Εικόνα 41. Main Menu Widget Designer

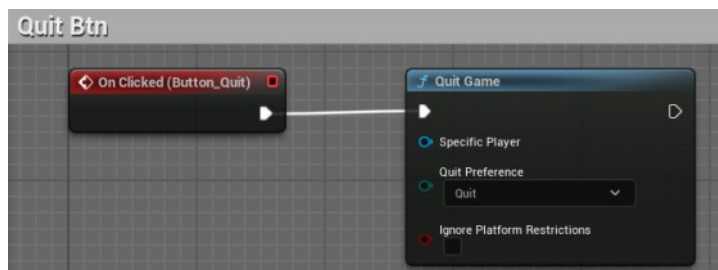
Στη σελίδα του Graph, το Blueprint του συγκεκριμένου widget αποτελείται από τέσσερα διαγράμματα ροής. Παρακάτω αναλύονται οι λειτουργικότητες που υλοποιήθηκαν.

- **Button_Play:** Όταν πατηθεί το κουμπί Start Game (On_Clicked(Button_Play)) (βλ. [Εικόνα 42.](#)), εξαφανίζεται το widget από την οθόνη του παίκτη (Remove from Parent), δημιουργείται (create widget) το Intro Seq Widget και εμφανίζεται στην οθόνη (Add to Viewport). Παίζει (Start) ο ήχος που βρίσκεται στον First Person (Media Intro Sound) του video που υπάρχει στο widget και το παιχνίδι τίθεται σε λειτουργία UI Only με κρυφό τον κέρσορα.



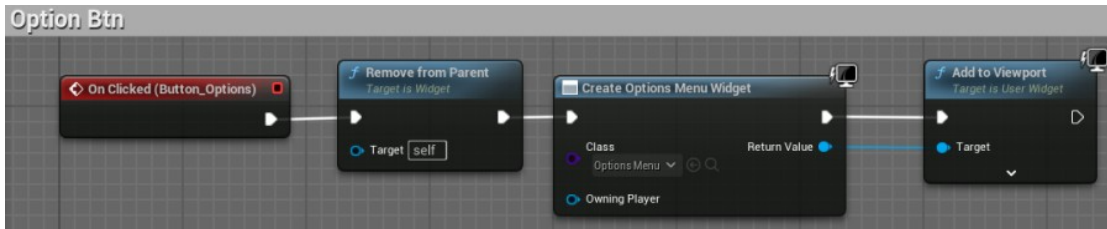
Εικόνα 42. Main Menu Widget Event Graph (Button_Play)

- **Button_Quit:** Όταν πατηθεί το κουμπί Quit Game (βλ. [Εικόνα 43.](#)) το παιχνίδι τερματίζει (Quit Game).



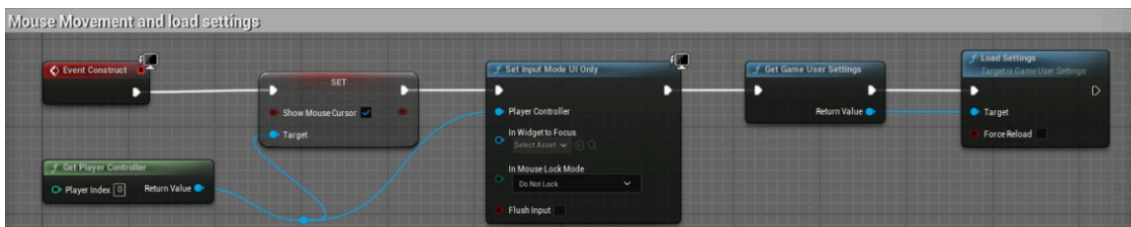
Εικόνα 43. Main Menu Widget Event Graph (Button_Quit)

- **Button_Options:** Όταν πατηθεί το κουμπί Options (βλ. [Εικόνα 44.](#)) το widget εξαφανίζεται από την οθόνη (Remove from Parent), δημιουργείται το widget που έχει φτιαχτεί για τις επιλογές (create Option Menu Widget) και προβάλλεται στην οθόνη (Add to Viewport).



Εικόνα 44. Main Menu Widget Event Graph (Button_Options)

- Mouse Movement and Load Settings:** Όταν δημιουργείται το widget (Event Construct) (βλ. [Εικόνα 45.](#)) θέλουμε να μπορεί ο χρήστης να βλέπει τον κέρσορα (Show mouse Cursor και Mode UI Only) ώστε να επιλέξει κάποιο από τα κουμπιά-επιλογές που του δίνονται και έπειτα σε περίπτωση που έχει αλλάξει τις ρυθμίσεις από το μενού επιλογών (βλ. [Μενού Επιλογών \(OptionsMenu\)](#)) να φορτώνονται αυτές στο παιχνίδι (Get and Load Settings), διαφορετικά φορτώνονται οι default επιλογές.



Εικόνα 45. Main Menu Widget Event Graph (Mouse Movement /Load Settings)

Τέλος, παρουσιάζεται το αποτέλεσμα που φαίνεται στον παίκτη (βλ. [Εικόνα 46.](#)) κατά τη διάρκεια του παιχνιδιού όταν βρίσκεται στο level του MainMenu.

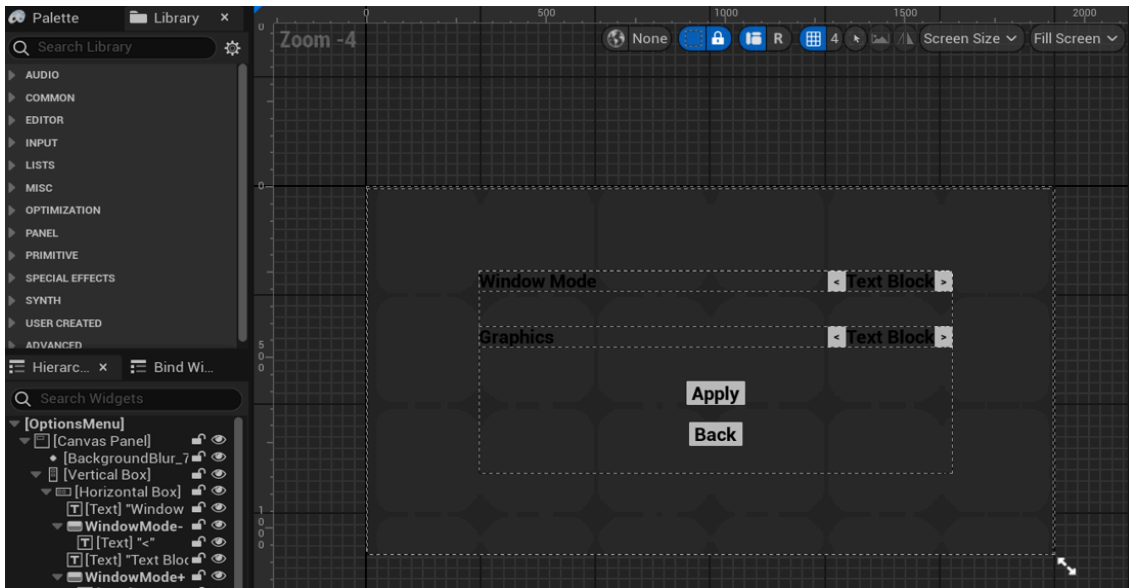


Εικόνα 46. Main Menu Τελικό αποτέλεσμα

Μενού Επιλογών (OptionsMenu)

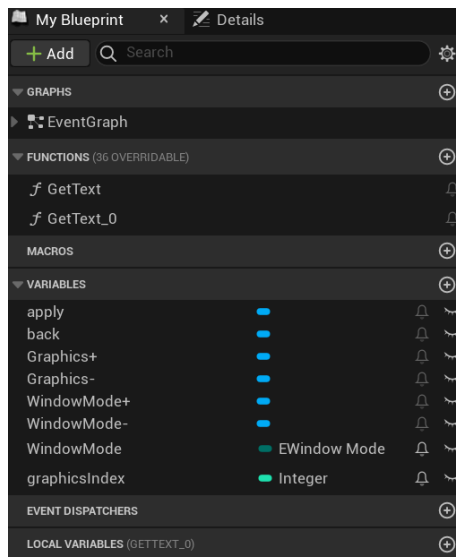
Η διάταξη Design του OptionsMenu (βλ. [Εικόνα 47.](#)) περιλαμβάνει μέσα στο canvas panel ένα κουτί για οριοθέτηση της διάταξης (vertical box) με δύο διαφορετικής οριοθέτησης κουτιά (horizontal box). Το κάθε οριζόντιο κουτί περιέχει με τη σειρά

text-button με text-text-button με text. Έτσι δημιουργήσαμε την επιλογή Window Mode και Graphics. Τα κουμπιά μοιάζουν με βελάκια και ονομάστηκαν WindowMode- και WindowMode+, αντίστοιχα και για την επιλογή Graphics. Τα λεκτικά ανάμεσα στα κουμπια τα αφήσαμε Text Block καθώς δημιουργήσαμε 2 συναρτήσεις (Βλέπε Functions παρακάτω) που θα τραβούν το text της επιλογής. Τέλος, προστέθηκαν 2 κουμπιά, Apply και Back.



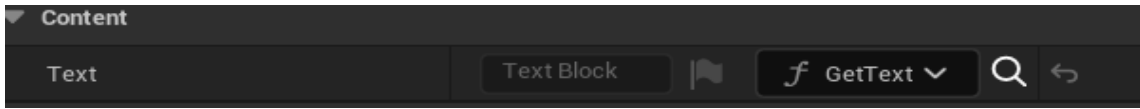
Εικόνα 47. Option Menu Designer

Για να προχωρήσουμε στην υλοποίηση των διαγραμμάτων ροής του Graph, δημιουργούμε δύο μεταβλητές (βλ. [Εικόνα 48.](#)), την Window Mode (έτοιμη αριθμημένη λίστα του unreal, EWindow Mode Type, με επιλογές Fullscreen, Windowed Fullscreen και Windowed) και την Graphics Index (integer με default value 0).



Εικόνα 48. Option Menu (Variables/Functions)

Έπειτα δημιουργούμε δύο συναρτήσεις για να καλούμε την ονομασία τους στην οθόνη του χρήστη, τις οποίες τις αναθέτουμε στο αντίστοιχο text box (βλ. [Εικόνα 49.](#)) στην επιλογή των details, content-> text.

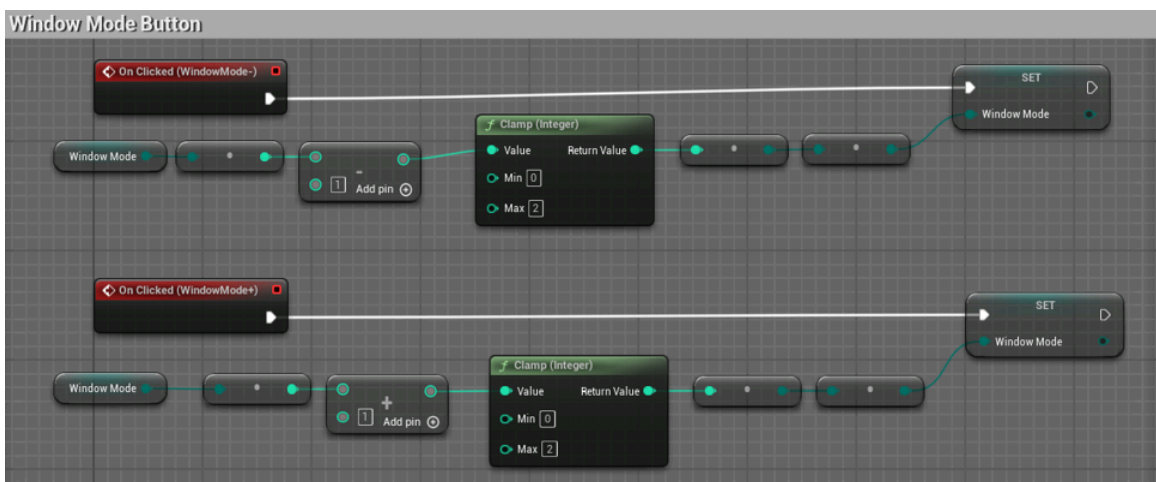


Εικόνα 49. Function Binding GetText

Παρακάτω αναλύονται οι λειτουργικότητες που υλοποιήθηκαν.

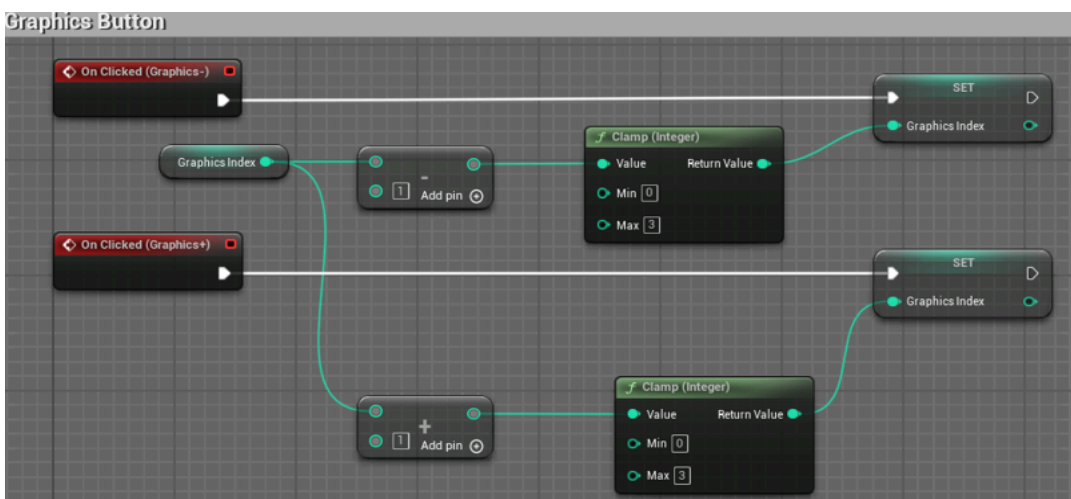
- WindowMode-/+:** Όταν πατηθεί το κουμπί WindowMode- (<) (βλ. [Εικόνα 50.](#)), παίρνουμε τη λίστα EWindow Mode Type και αφού μετατραπεί σε integer αφαιρούμε 1 κάθε φορά που πατιέται το κουμπί. Έχουμε βάλει περιορισμό από 0-2 καθώς στη συγκεκριμένη λίστα η αρίθμηση είναι Fullscreen(0), Windowed Fullscreen(1) και Windowed(2). Τέλος αποθηκεύουμε το αποτέλεσμα στην μεταβλητή.

Ομοίως και όταν πατηθεί το WindowMode+(>), όμως σε αυτήν την περίπτωση αντί να αφαιρούμε, προσθέτουμε 1.



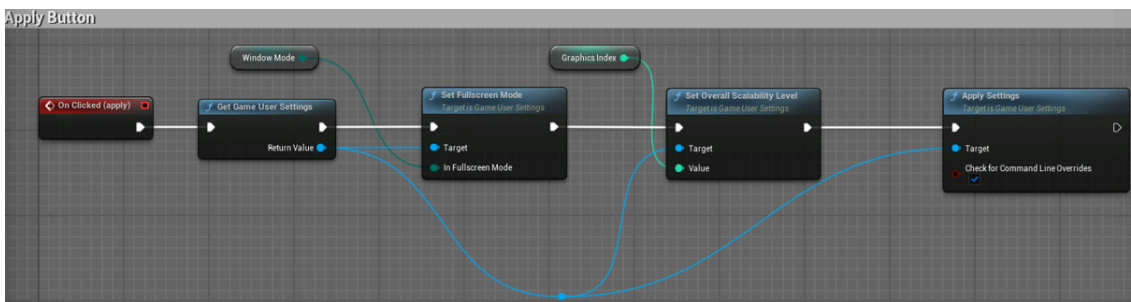
Εικόνα 50. Option Menu Event Graph (Window Mode+/-)

- Graphics-/+:** Όμοια με το Window Mode Button, με τη διαφορά πως ήδη δουλεύουμε με integer, οπότε δεν χρειάζεται μετατροπή (βλ. [Εικόνα 51.](#)). Ο περιορισμός είναι 0-3 βάση οριοθέτησης των γραφικών στη συνάρτηση get_text_0 που θα δούμε παρακάτω (Functions).



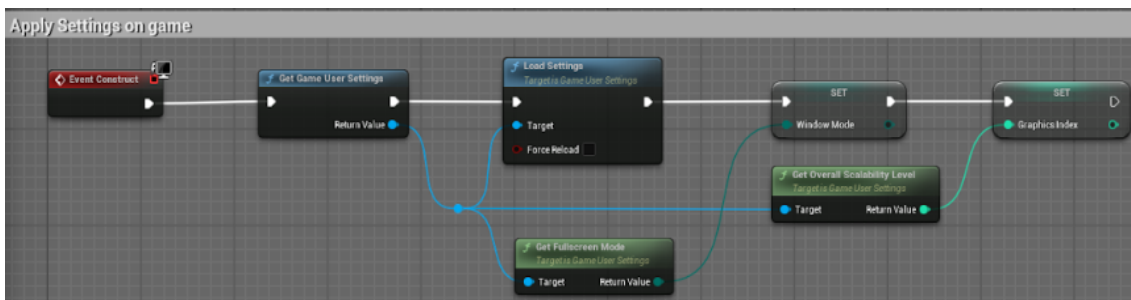
Εικόνα 51. Option Menu Event Graph (Graphics+/-)

- **apply:** Όταν πατηθεί το κουμπί Apply (βλ. [Εικόνα 52.](#)) παίρνουμε τις ρυθμίσεις του παιχνιδιού και τις αλλάζουμε με αυτές που έχουν αποθηκευτεί στις μεταβλητές Window Mode και Graphics Index, δηλαδή αυτές που έχει επιλέξει ο χρήστης-παίκτης. Τέλος, τις κάνουμε apply στο παιχνίδι.



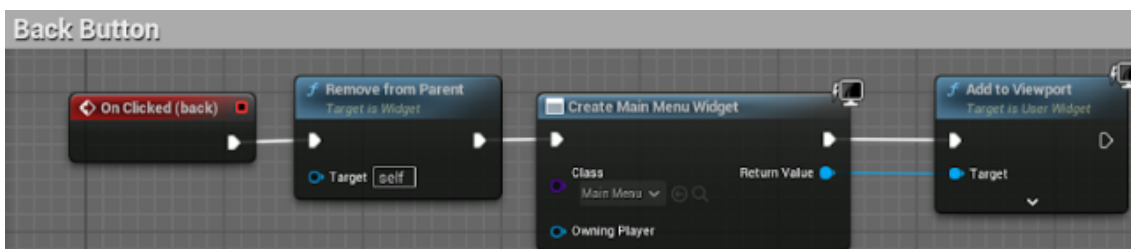
Εικόνα 52. Option Menu Event Graph (Apply Button)

- **Apply Settings on game:** Όταν δημιουργείται το widget (βλ. [Εικόνα 53.](#)) θέλουμε να τραβηχτούν οι default ρυθμίσεις του παιχνιδιού σε περίπτωση που ο χρήστης δεν θέλει να τις αλλάξει. Τραβάμε αυτές τις ρυθμίσεις και τις “σετάρουμε” με τις μεταβλητές μας, Window Mode και Graphics Index.



Εικόνα 53. Option Menu Event Graph (Apply settings)

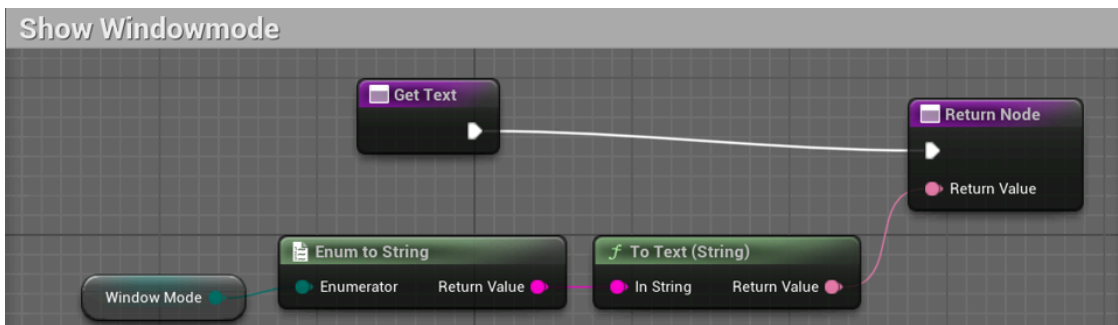
- **Back:** Όταν πατηθεί το κουμπί Back (βλ. [Εικόνα 54.](#)) το widget εξαφανίζεται από την οθόνη (Remove from Parent), δημιουργείται το widget που έχει φτιαχτεί για το μενού (create Main Menu Widget) και προβάλλεται στην οθόνη (Add to Viewport).



Εικόνα 54. Option Menu Event Graph (Back Button)

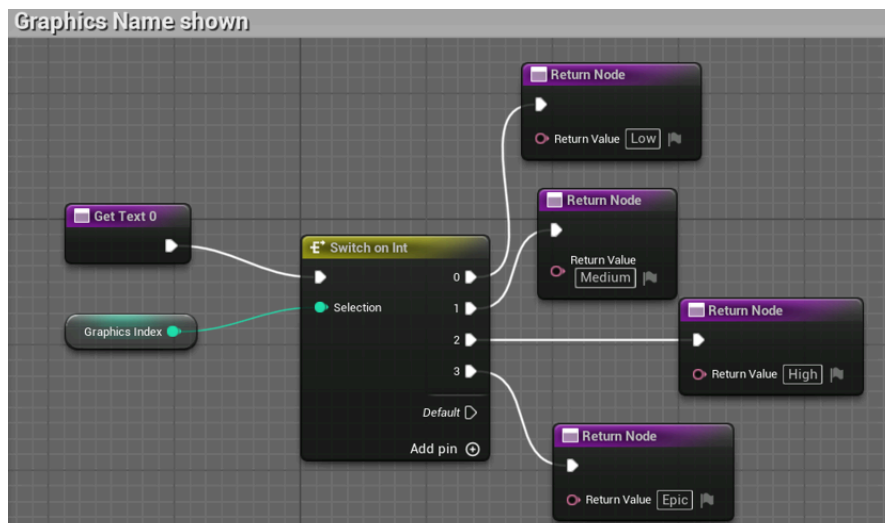
Στο συγκεκριμένο Graph υλοποιήθηκαν και δύο συναρτήσεις όπως προαναφέραμε οι οποίες αναλύονται ως εξής:

- **Get_text:** Αυτή η συνάρτηση (βλ. [Εικόνα 55.](#)) τραβά την επιλογή που είναι αποθηκευμένη στη μεταβλητή Window Mode, την μετατρέπει σε string και έπειτα τυπώνει σε text.



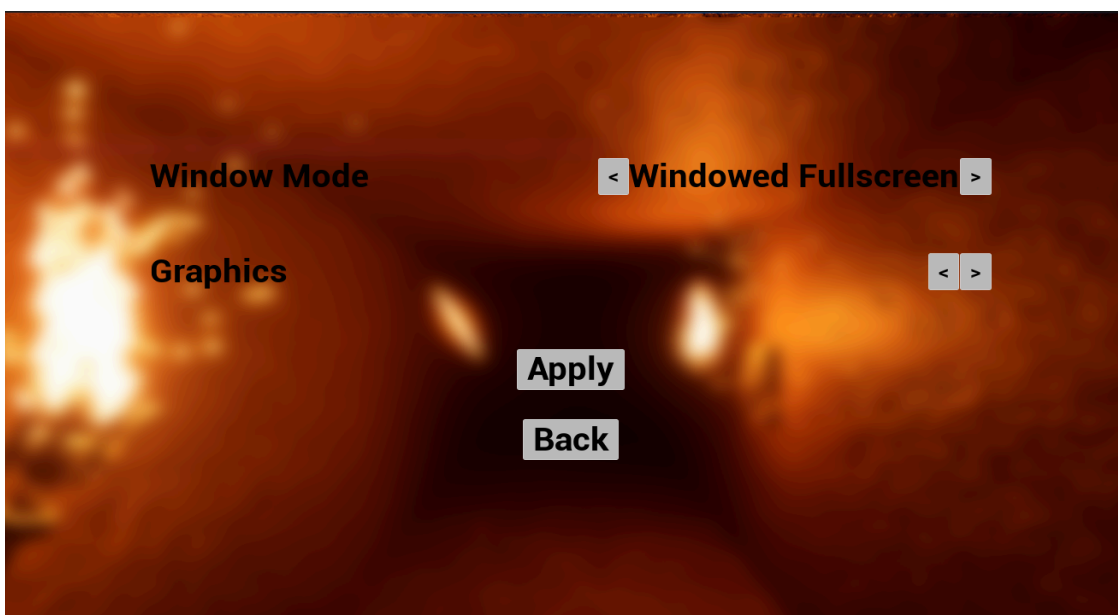
Εικόνα 55. Option Menu Event Graph (Function Get_Text)

- **Get_text_0:** Αυτή η συνάρτηση (βλ. [Εικόνα 56.](#)) τραβά την επιλογή που είναι αποθηκευμένη στη μεταβλητή Graphics Index και ανάλογα την επιλογή του χρήστη τυπώνει για Low(0), Medium(1), High(2) και Epic(3).



Εικόνα 56. Option Menu Event Graph (Function Get_Text_0)

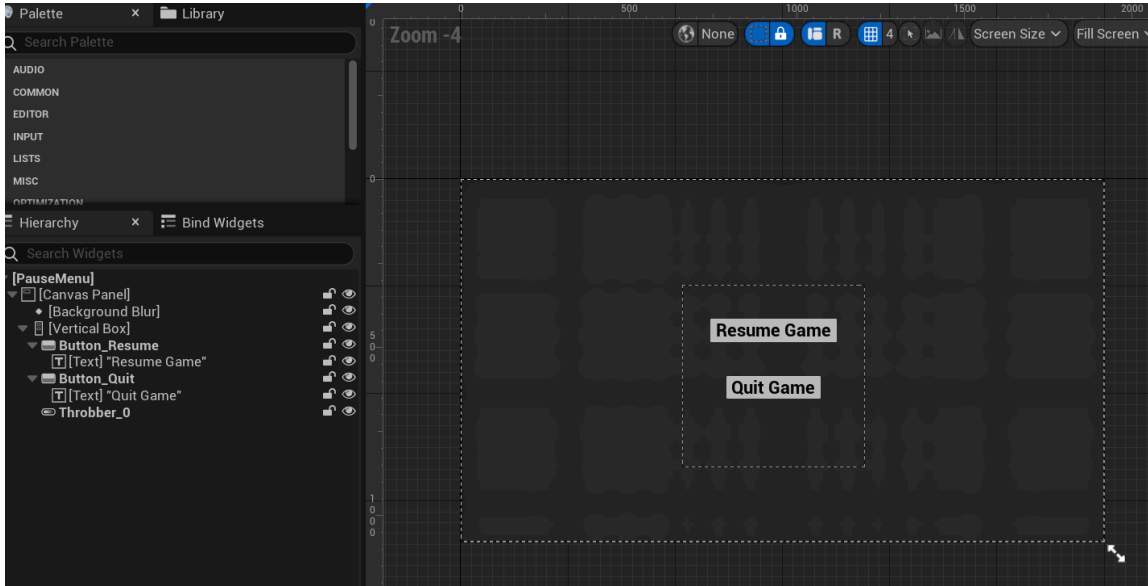
Παρακάτω (βλ. [Εικόνα 57.](#)) παραθέτουμε το τελικό αποτέλεσμα στον χρήστη όταν θέτει σε λειτουργία το OptionsMenu.



Εικόνα 57. Option Menu Τελικό αποτέλεσμα

Μενού Παύσης (PauseMenu)

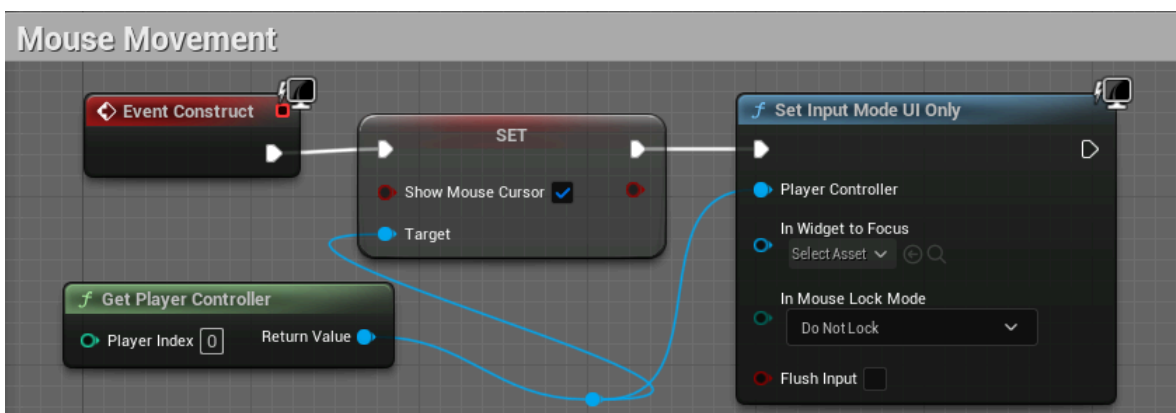
Η διάταξη στη σελίδα Designer του PauseMenu (βλ. [Εικόνα 58.](#)) περιλαμβάνει μέσα στο canvas panel ένα κουτί για οριοθέτηση της διάταξης (vertical box) μέσα στο οποίο υπάρχουν δύο κουμπιά και ένα κινούμενο που απεικονίζει φόρτωση (Throbber), το κουμπί (Button_Resume) με text “Resume Game” και κουμπί (Button_Quit) με text “Quit Game”. Επίσης έχει ένα Background Blur για να διαφοροποιείται από το περιβάλλον του παιχνιδιού.



Εικόνα 58. Pause Menu Designer

Στη σελίδα του Graph, το Blueprint του συγκεκριμένου widget αποτελείται από τρία διαγράμματα ροής. Οι υλοποιήσεις του Blueprint παρατίθενται παρακάτω.

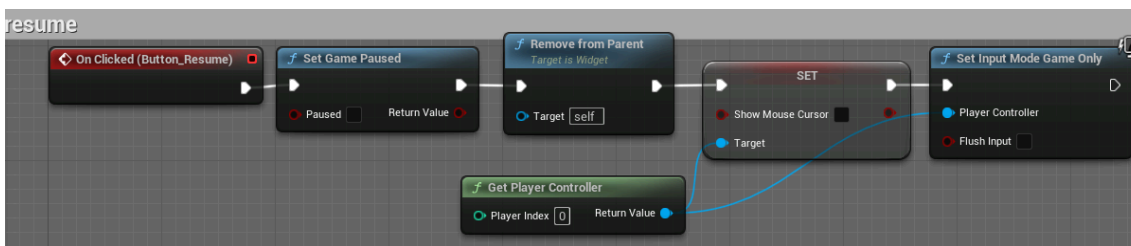
- **Mouse Movement:** Όταν δημιουργείται το widget (βλ. [Εικόνα 59.](#)) βάζουμε τον κέρσορα του ποντικιού να φαίνεται στον παίκτη (set show mouse cursor) και ρυθμίζουμε την λειτουργία σε UI Only (Set Input Mode UI Only) ώστε να μην υπάρχει αλληλεπίδραση με το κυρίως παιχνίδι.



Εικόνα 59. Pause Menu Event Graph (Mouse Movement)

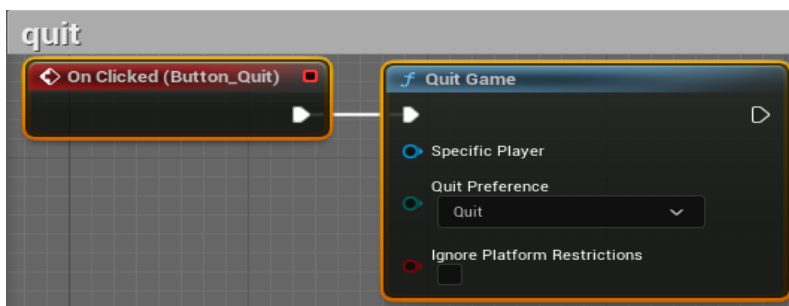
- **Button_Resume:** Όταν πατηθεί το κουμπί Resume Game (βλ. [Εικόνα 60.](#)) ρυθμίζουμε το παιχνίδι να μην βρίσκεται πλέον σε παύση (Set Game

Paused(false)), αφαιρούμε το widget από την οθόνη του χρήστη (Remove From Parent) και επιστρέφουμε την λειτουργία σε Mode Game Only.



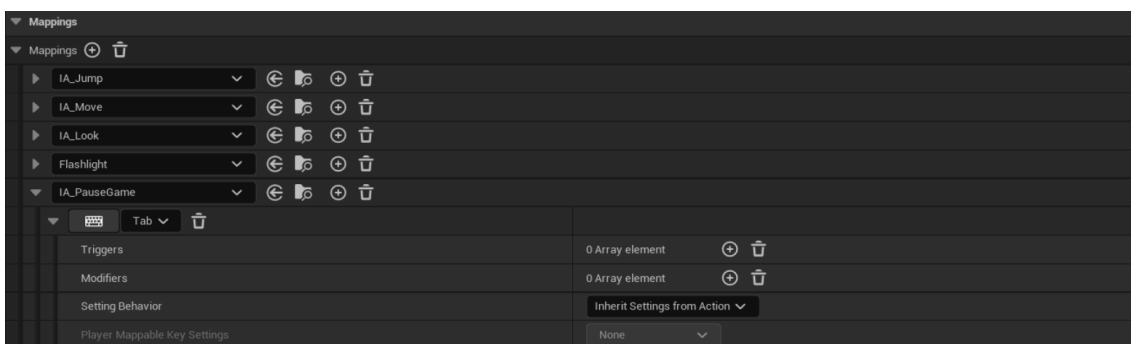
Εικόνα 60. Pause Menu Event Graph (Button_Resume)

- **Button_Quit:** Όταν πατηθεί το κουμπί Quit (βλ. [Εικόνα 61.](#)) καλούμε τη συνάρτηση τερματισμού του παιχνιδιού (Quit Game).



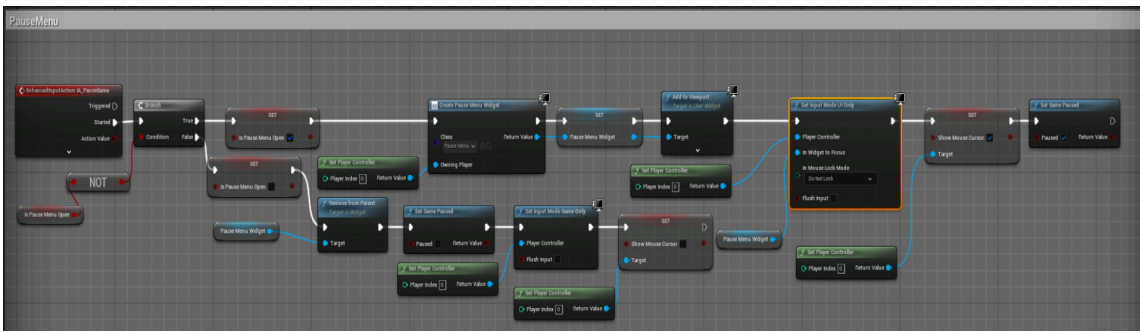
Εικόνα 61. Pause Menu Event Graph (Button_Quit)

Για να αλληλεπιδράσει ο παίκτης με αυτό το widget, χρειάστηκε να δημιουργηθεί ένα διάγραμμα ροής στο Blueprint του FirstPerson. Για να λειτουργήσει αυτό, προσθέσαμε ένα ακόμη mapping στο First Person και το συνδέσαμε με το πλήκτρο “Tab” (βλ. [Εικόνα 62.](#)). Με το πάτημα του πλήκτρου “Tab” ενεργοποιείται το pausemenu.



Εικόνα 62. Mapping First Person “Tab”

Στο Event Graph του FirstPerson, ορίζουμε μια μεταβλητή boolean (IsPauseMenuOpen). Με το πάτημα του κουμπιού “tab”, (βλ. [Εικόνα 63.](#)), γίνεται έλεγχος αυτής της μεταβλητής για το αν είναι ήδη ενεργό το pausemenu. Αν δεν είναι ενεργό, τότε δημιουργείται το widget του PauseMenu, εμφανίζεται στην οθόνη του παίκτη, το παιχνίδι τίθεται σε λειτουργία UI Only, εμφανίζεται ο κέρσορας και το παιχνίδι γίνεται Paused. Αν είναι ενεργό, τότε αφαιρείται από την οθόνη του παίκτη, η λειτουργία του παιχνιδιού τίθεται Game Only και εξαφανίζεται ο κέρσορας.



Εικόνα 63. Pause Menu in First Person Event Graph

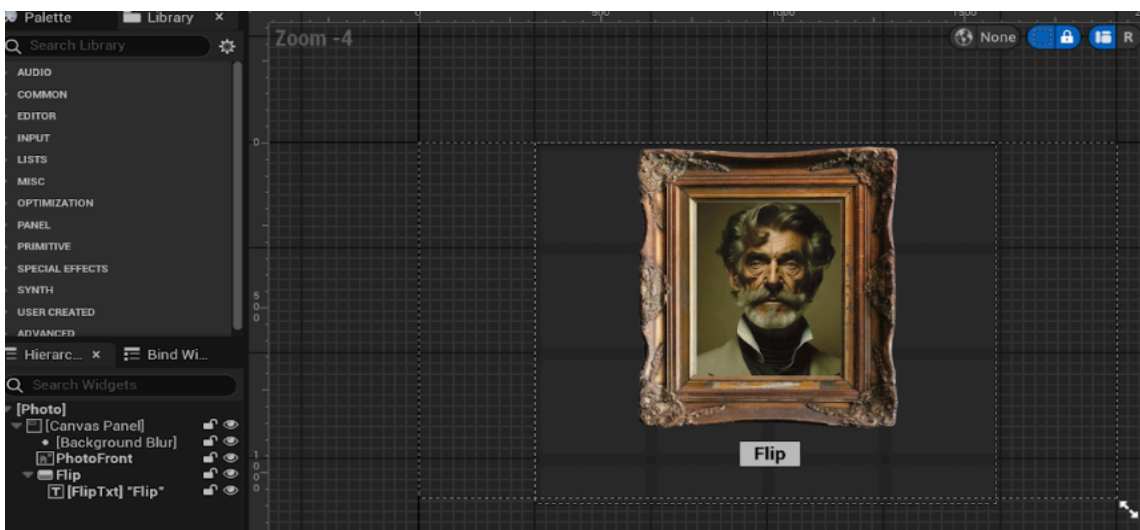
Στη συνέχεια, παρατίθεται το αποτέλεσμα του widget (βλ. [Εικόνα 64.](#)).



Εικόνα 64. Pause Menu Τελικό ποτέλεσμα

Γρίφος Φωτογραφία (Photo)

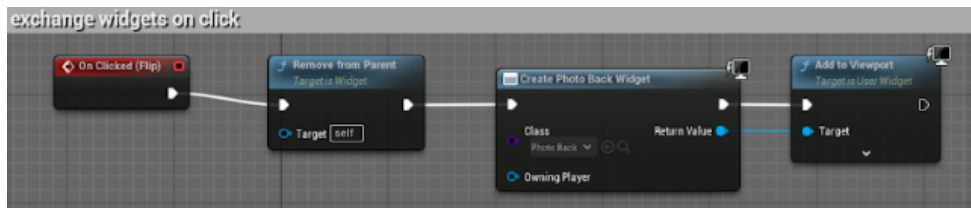
Στη σελίδα του Designer, η διάταξη του Photo (βλ. [Εικόνα 65.](#)), περιλαμβάνει μέσα στο canvas panel (Οθόνη που βλέπει ο χρήστης) μια εικόνα μιας κορνίζας με φωτογραφία και ένα κουμπί (Flip) με text “Flip”.



Εικόνα 65. Photo Designer

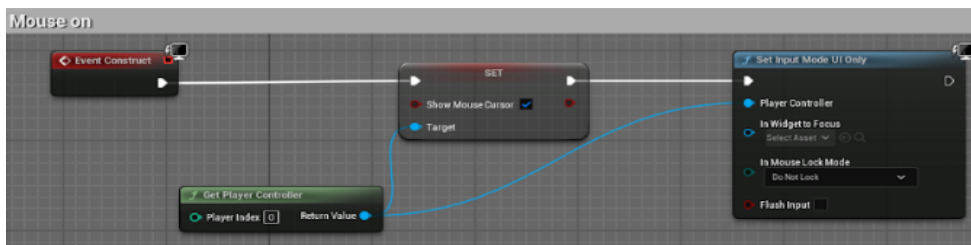
Το Blueprint του συγκεκριμένου widget αποτελείται από δύο διαγράμματα ροής τα οποία τα βρίσκει κανείς στη σελίδα του Graph.

- **Flip:** Όταν πατηθεί το κουμπί Flip (βλ. [Εικόνα 66.](#)) το widget φεύγει από την οθόνη του παίκτη, καλείται το widget photoback και εμφανίζεται στην οθόνη του χρήστη.



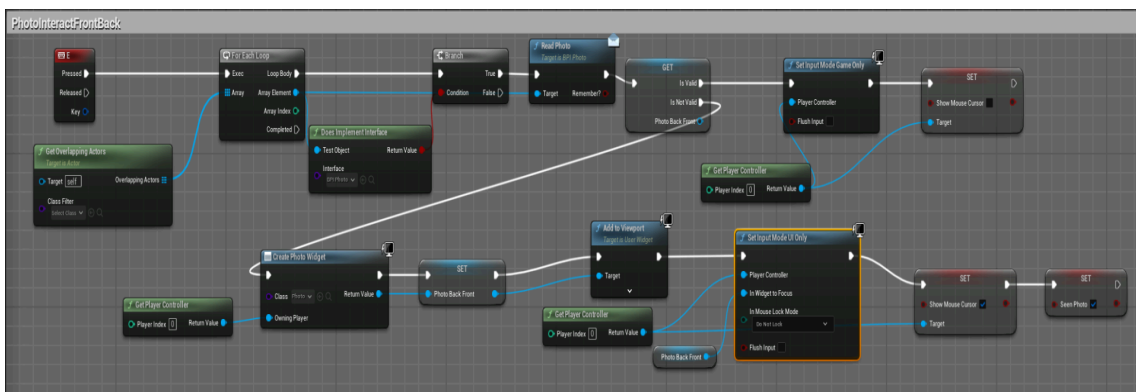
Εικόνα 66. Photo Event Graph (Button Flip)

- **Mouse on:** Όταν δημιουργείται το widget (βλ. [Εικόνα 67.](#)) θέλουμε να μπορεί ο χρήστης να βλέπει τον κέρσορα (Show mouse Cursor και Mode UI Only) ώστε να μπορεί να επιλέξει το κουμπί Flip.



Εικόνα 67. Photo Event Graph (Mouse movement)

Για να αλληλεπιδράσει ο παίκτης με αυτό το widget, χρειάστηκε να δημιουργηθεί ένα διάγραμμα ροής στο Blueprint του FirstPerson (βλ. [Εικόνα 68.](#)). Αρχικά ορίζουμε μια μεταβλητή boolean (SeenPhoto). Όταν πατηθεί το πλήκτρο “E” και ο παίκτης βρίσκεται μέσα στο Box Collision του PhotoFrame (Get Overlapping Actors), τότε καλείται μέσω του UI BPI_photo (Does Implement Interface) η σύνδεση actor-widget. Έπειτα, αν η σύνδεση είναι σωστή τότε καλείται η συνάρτηση Read Photo. Αν δεν έχει δημιουργηθεί το widget Photo (Validation Get), το δημιουργούμε, το προσθέτουμε στην οθόνη του παίκτη με λειτουργία παιχνιδιού UI Only και εμφανίζουμε τον κέρσορα. Η μεταβλητή SeenPhoto γίνεται true. Σε περίπτωση που έχει δημιουργηθεί το παιχνίδι συνεχίζει κανονικά, λειτουργία Game Only.



Εικόνα 68. Photo in First Person Event Graph

Γρίφος Φωτογραφία πίσω μέρος (Photo Back)

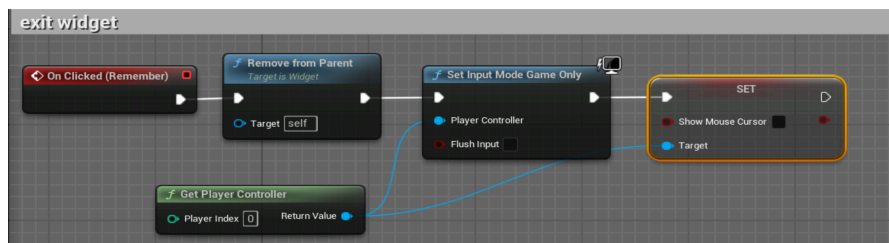
Η διάταξη στο Designer του Photo (βλ. [Εικόνα 69.](#)) περιλαμβάνει μέσα στο canvas panel μια εικόνα μιας κορνίζας με φωτογραφία, και ένα κουμπί (Flip) με text “Flip”.



Εικόνα 69. Photo Back Designer

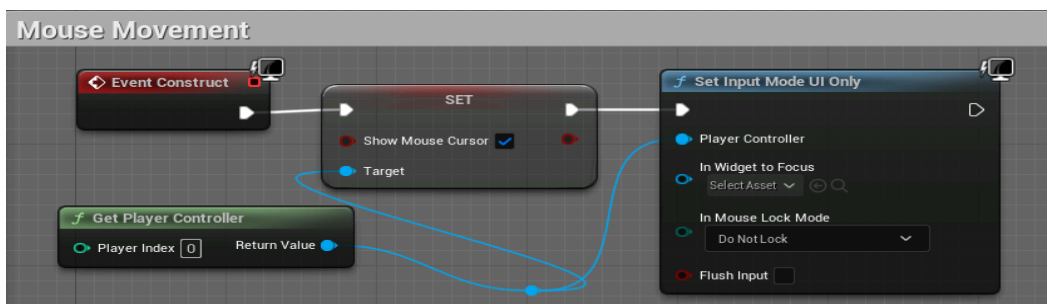
Το Blueprint αυτού του widget αποτελείται από δύο διαγράμματα ροής τα οποία βρίσκονται στο Graph.

- **Remember:** Όταν πατηθεί το κουμπί Remember (βλ. [Εικόνα 70.](#)) το widget φεύγει από την οθόνη του παίκτη και αφού το παιχνίδι τεθεί σε Mode Game Only, και ο κέρσορας αφαιρείται από την οθόνη (Set Show Mouse Cursor).



Εικόνα 70. Photo Back Event Graph (Button Remember)

- **Mouse movement:** Όταν δημιουργείται το widget (βλ. [Εικόνα 71.](#)) θέλουμε να μπορεί ο χρήστης να βλέπει τον κέρσορα (Show mouse Cursor και Mode UI Only) ώστε να μπορεί να επιλέξει το κουμπί Flip. (Βλ [Γρίφος Φωτογραφία \(Photo\).](#))



Εικόνα 71. Photo Back Event Graph (Mouse Movement)

Σε αυτό το σημείο, παρουσιάζεται το τελικό αποτέλεσμα που φαίνεται στον παίκτη, των widgets photo και photo back (βλ. [Εικόνα 72.](#)), καθώς το ένα widget είναι αλληλένδετο κομμάτι του άλλου.

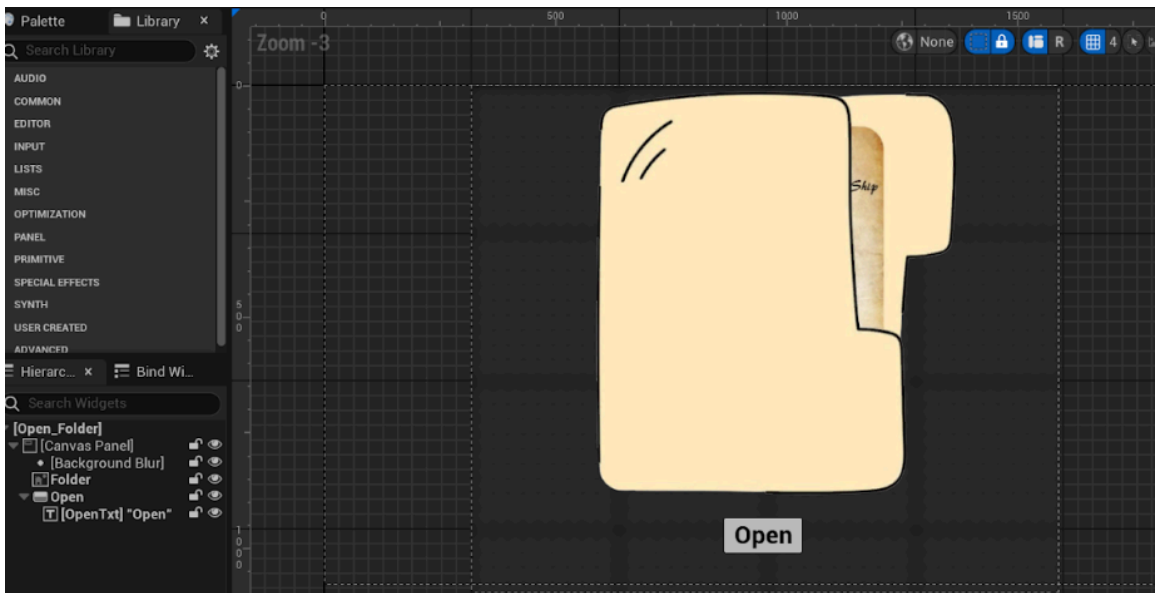


Εικόνα 72. Photo και Photo Back Τελικό αποτέλεσμα

Γρίφος Ναύτες (Sailors)

Πρόκειται για ένα σύνολο από widgets, τα οποία συνδέονται ομοίως με ένα κουμπί. Το σύνολο αυτό αποτελείται από το widget **Open_Folder**, που είναι και αυτό που καλείται πρώτο από το χρήστη, στη συνέχεια από τα widgets με την αντίστοιχη σειρά εμφάνιση στο χρήστη, **BobJameson**, **JackPortman**, **LeonardAsher**, **MarkusDoyle**, **PatrickAllen**, **StefanDjokovic**, **JohnTrueman** και τελευταίο το **AllSailors**.

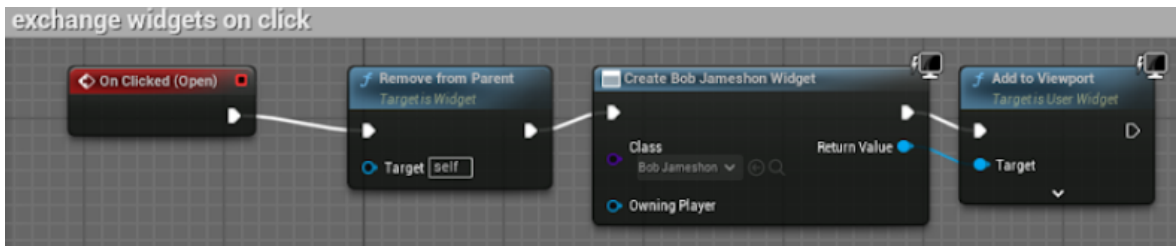
Αρχικά, η διάταξη του widget **Open_Folder**, (βλ. [Εικόνα 73.](#)), βρίσκεται στη σελίδα Design και περιλαμβάνει μέσα στο canvas panel μια εικόνα ενός κλειστού φακέλου (Folder) και ένα κουμπί (Open) με text “Open”.



Εικόνα 73. Open Folder Designer

Προχωρώντας στη σελίδα Graph, η οποία αποτελείται από δύο διαγράμματα ροής, γίνεται κατά τη δημιουργία του widget εμφάνιση του κέρσορα στην οθόνη όπως ακριβώς και στο widget photo (βλ. διάγραμμα ροής [Γρίφος Φωτογραφία](#)) και στη

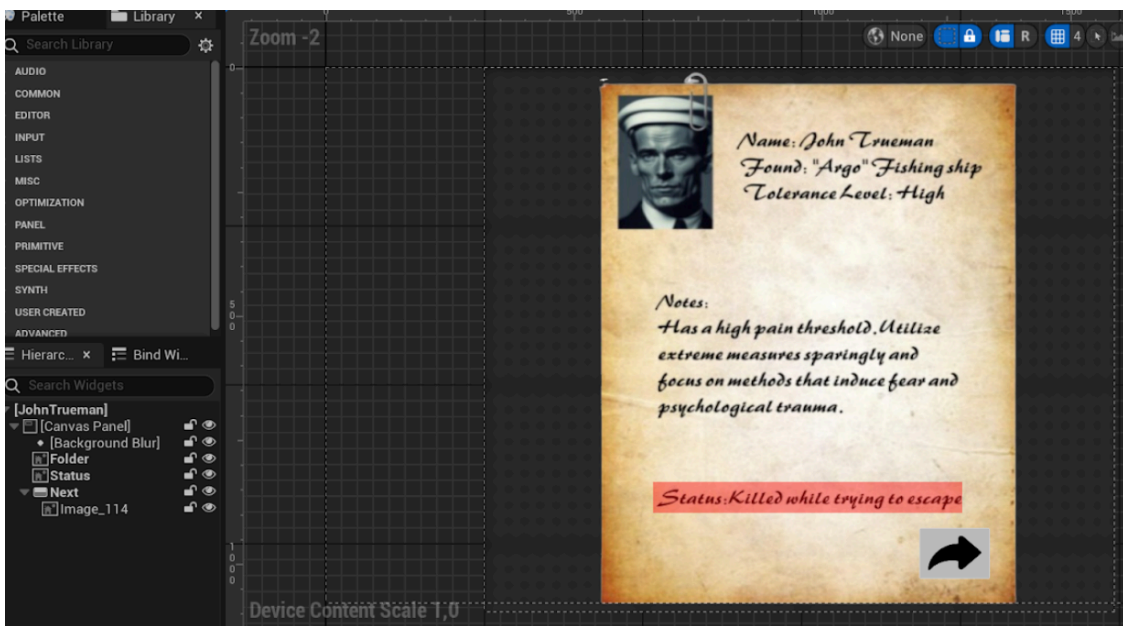
συνέχεια όταν ο χρήστης πατήσει το κουμπί Open (βλ. [Εικόνα 74.](#)) , το συγκεκριμένο widget δίνει τη θέση του στην οθόνη στο widget BobJameson.



Εικόνα 74. Open Folder Event Graph (Exchange widgets)

Στη συνέχεια, ακολουθούν επτά widgets που έχουν όμοια υλοποίηση. Τα widgets αυτά είναι BobJameson, JackPortman, LeonardAsher, MarkusDoyle, PatrickAllen, StefanDjokonic και JohnTrueman. Σε αυτή την ενότητα θα αναλυθεί ένα από αυτά λόγω ομοιότητας υλοποίησης, το JohnTrueman, καθώς και είναι πρόσωπο ενδιαφέροντος για την ιστορία μας.

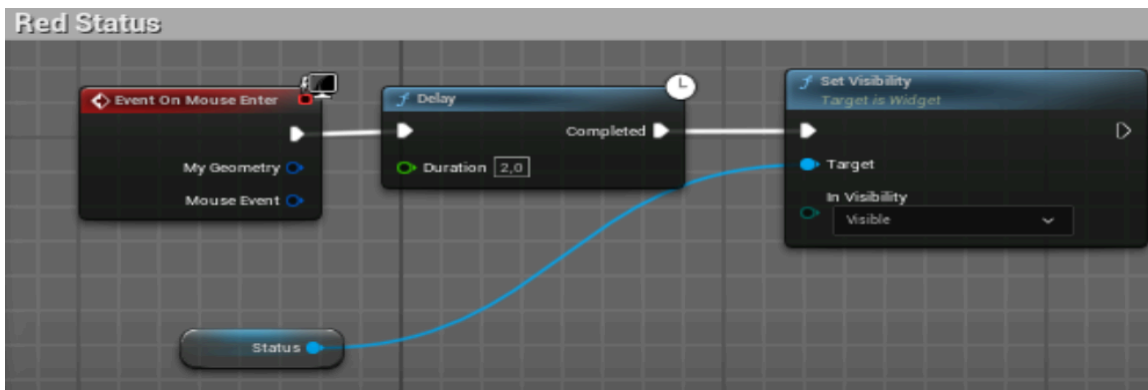
Η διάταξη του widget JohnTrueman, που βρίσκεται στη σελίδα Design (βλ. [Εικόνα 75.](#)), περιλαμβάνει μέσα στο canvas panel μια εικόνα ενός εγγράφου καταγραφής βασανιστηρίων του Ezekiel που διαβάζει ο παίκτης (Folder), μία κόκκινη αχνή εικόνα (Status) για highlight η οποία αποτελεί και μεταβλητή, και ένα κουμπί (Next) με image (Image_114) ένα βέλος.



Εικόνα 75. John Trueman Designer

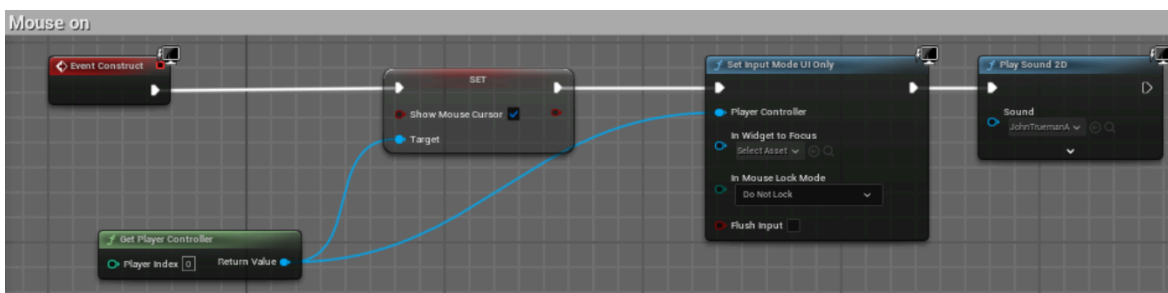
Το Graph του συγκεκριμένου widget , όπως και των ομοίων του, αποτελείται από τρία διαγράμματα ροής.

- **Highlight Status:** Στην κίνηση του ποντικιού (Event On Mouse Enter) και με καθυστέρηση 2 δευτερολέπτων, η εικόνα status που έχει γίνει μεταβλητή γίνεται ορατή (Set Visibility(Visible)) (βλ. [Εικόνα 76.](#)) για να τονίσει στο χρήστη το status των ναυτών.



Εικόνα 76. John Trueman Event Graph (Status Coloring)

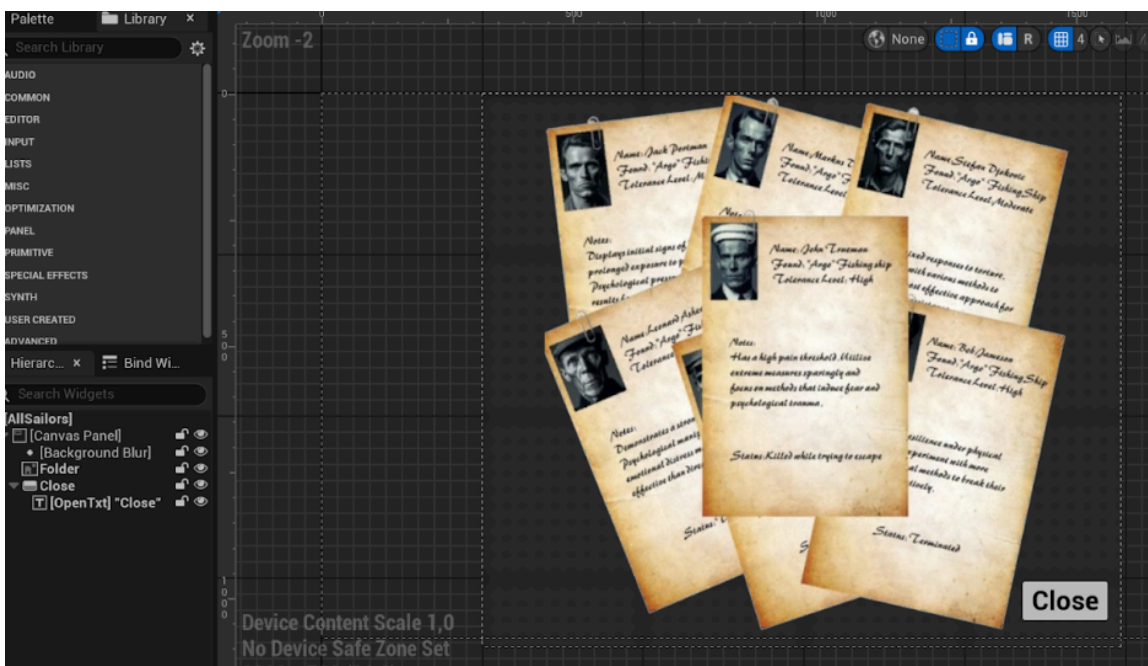
- **Mouse movement and Audio Play:** Κατα τη δημιουργία του widget (βλ. [Εικόνα 77.](#)) εμφανίζεται ο κέρσορας, βάζουμε λειτουργία UI Only και παίζει ένα ηχητικό με τη φωνή του Ezekiel.



Εικόνα 77. John Trueman Event Graph (Mouse Movement/Audio)

- **Next:** Με το πάτημα του κουμπιού Next, ομοίως με το widget Folder, δίνει τη θέση του στο widget AllSailors.

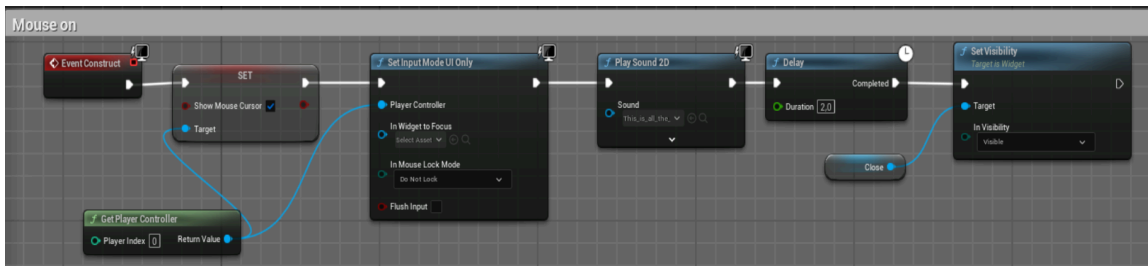
Τέλος, το τελευταίο widget είναι το AllSailors, το οποίο έχει όμοια διάταξη στη σελίδα Design με το Folder, αλλά αντί για κουμπί και text “Open” έχει Close (βλ. [Εικόνα 78.](#)).



Εικόνα 78. All Sailors Designer

Όσον αφορά το Graph αυτού του widget, περιέχει δύο διαγράμματα ροής.

- **Close:** Όταν πατηθεί το κουμπί Close(On_Clicked(Close)) το widget φεύγει από την οθόνη του παίκτη και αφού το παιχνίδι τεθεί σε Mode Game Only, και ο κέρσορας αφαιρείται από την οθόνη (βλ. διάγραμμα ροής [Γρίφος Φωτογραφία πίσω μέρος \(Photo Back\)](#)).
- **Mouse Movement and Audio:** Όταν δημιουργείται το widget (βλ. [Εικόνα 79.](#)) θέλουμε να μπορεί ο χρήστης να βλέπει τον κέρσορα (Show mouse Cursor και Mode UI Only) ώστε να μπορεί να επιλέξει το κουμπί Close. Ακούγεται ένα ηχητικό από την Rebecca να βγάζει κάποια συμπεράσματα για την ιστορία του παιχνιδιού και με μια καθυστέρηση 2 δευτερολέπτων εμφανίζεται το κουμπί Close, το οποίο είναι αρχικά Hidden, όπως φαίνεται στην παρακάτω εικόνα.

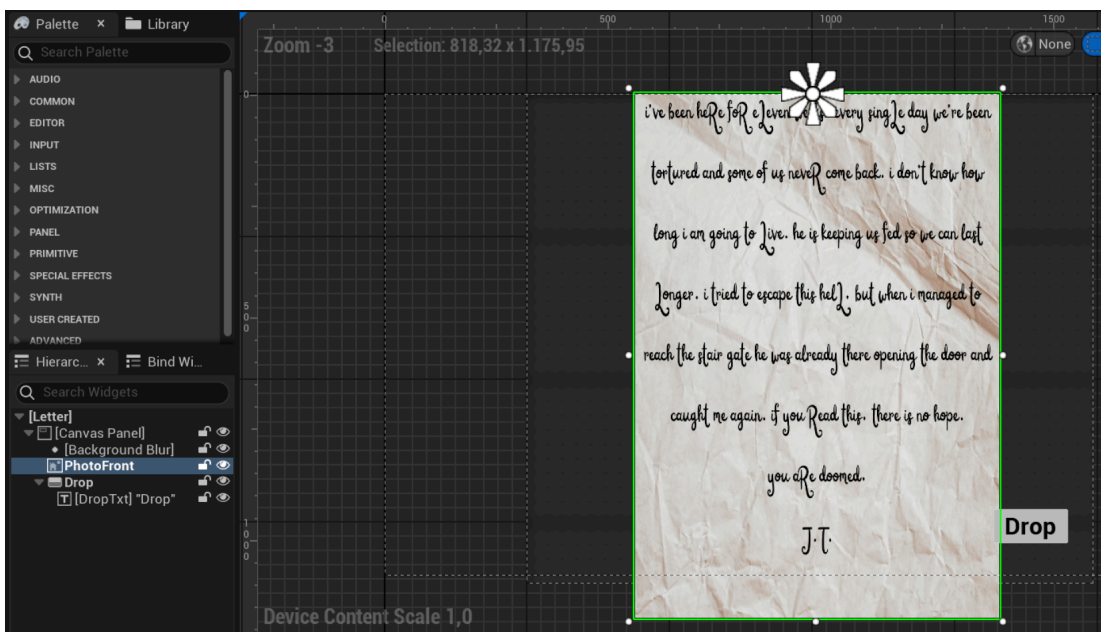


Εικόνα 79. All Sailors Event Graph (Mouse Movement/Audio)

Για να ενεργοποιήσει ο παίκτης το widget Open_Folder, χρειάστηκε να δημιουργηθεί ένα διάγραμμα ροής στο Blueprint του FirstPerson όμοιο με εκείνο του Photo Widget. Η σύνδεση γίνεται μέσω του UI BPI_Sailors που περιέχει τη συνάρτηση Sail.

Γρίφος Γράμμα (Letter)

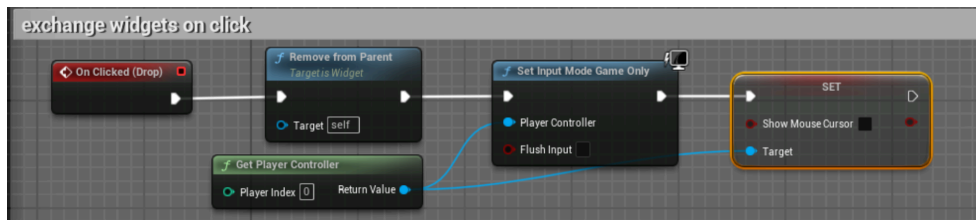
Η διάταξη του Letter βρίσκεται στη σελίδα Design (βλ. [Εικόνα 80.](#)) και περιλαμβάνει μέσα στο canvas panel μια εικόνα του γράμματος που διαβάζει ο παίκτης και ένα κουμπί (Drop) που είναι αόρατο αρχικά στο χρήστη με text “Drop”.



Εικόνα 80. Letter Designer

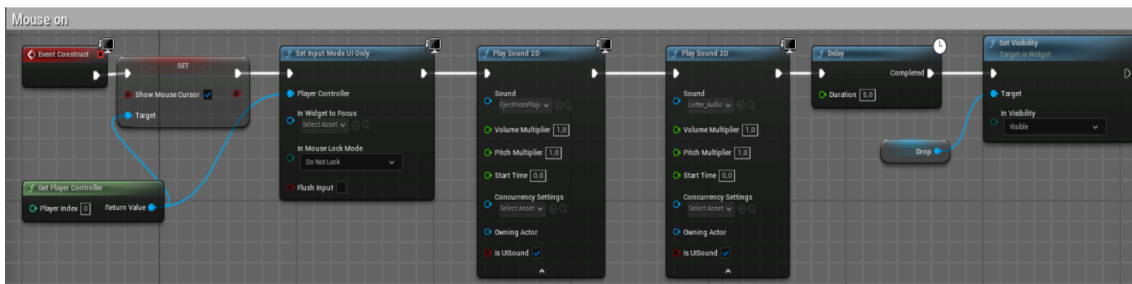
Το Blueprint του συγκεκριμένου widget, της σελίδας Graph, αποτελείται από δύο διαγράμματα ροής με την υλοποίηση αυτών να αναλύεται παρακάτω.

- **Drop:** Όταν πατηθεί το κουμπί Drop (βλ. [Εικόνα 81.](#)) το widget φεύγει από την οθόνη του παίκτη.



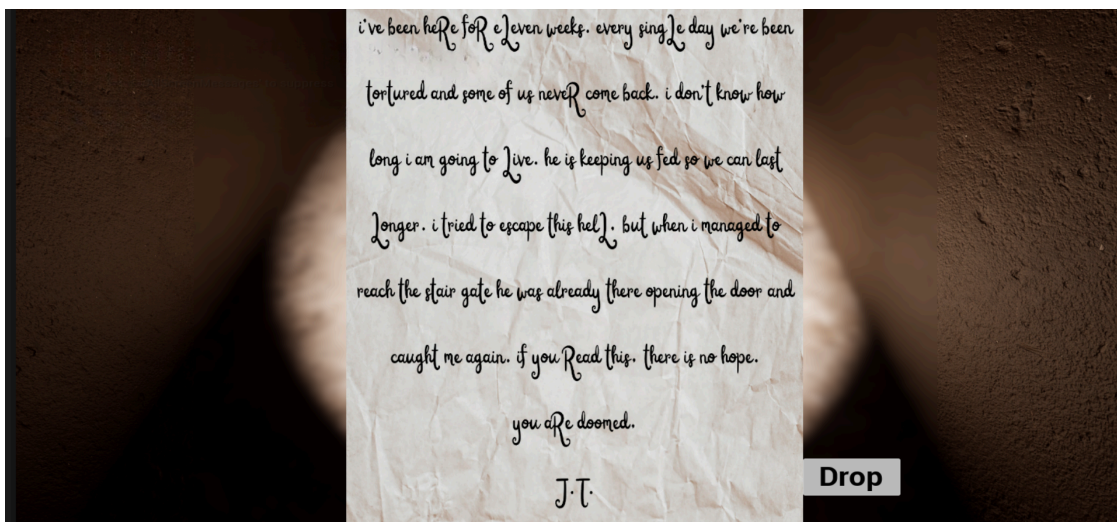
Εικόνα 81. Letter Event Graph (Button Drop)

- **Audio and drop btn delay:** Όταν δημιουργείται το widget (βλ. [Εικόνα 82.](#)) θέλουμε να μπορεί ο χρήστης να βλέπει τον κέρσορα (Show mouse Cursor και Mode UI Only) ακούγεται ένα ηχητικό (LetterAudio), από τον ίδιο τον John Trueman, και με καθυστέρηση (Delay) 5 δευτερολέπτων εμφανίζεται στην οθόνη το κουμπί Drop (Set Visibility).



Εικόνα 82. Letter Event Graph (Audio/Drop Button Delay)

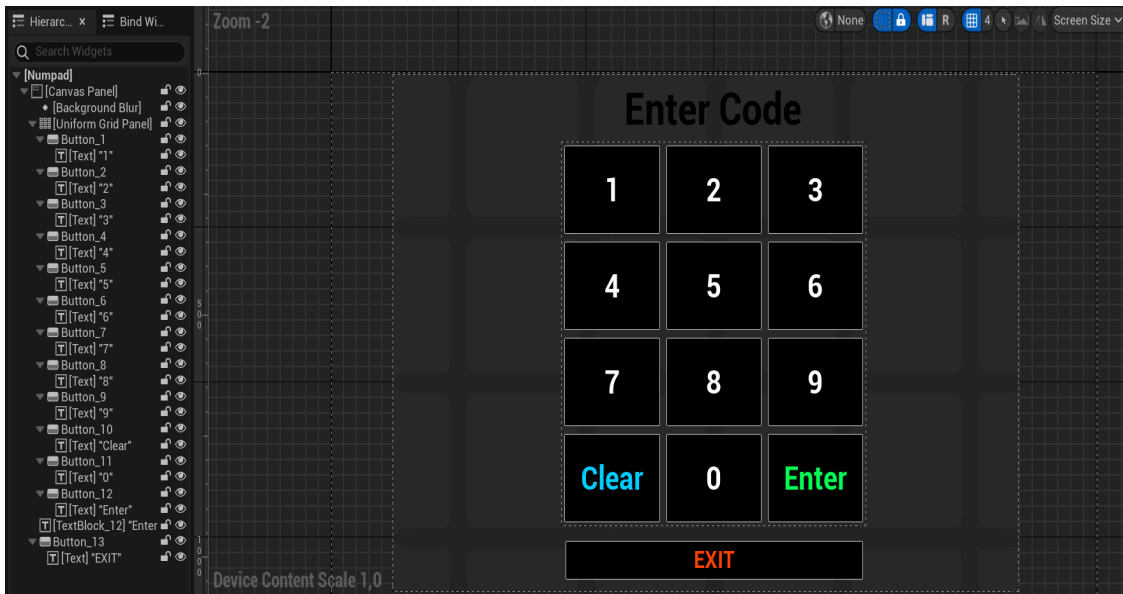
Για να ενεργοποιήσει ο παίκτης το widget Letter, χρειάστηκε να δημιουργηθεί ένα διάγραμμα ροής στο Blueprint του FirstPerson όμοιο με εκείνο του Photo Widget και του Open Folder. Η σύνδεση γίνεται μέσω του UI BPI_Letter που περιέχει τη συνάρτηση Letter Function. Ακολουθεί το αποτέλεσμα που βλέπει ο χρήστης στην οθόνη του (βλ. [Εικόνα 83.](#)) κατά τη διάρκεια του παιχνιδιού.



Εικόνα 83. Letter Τελικό αποτέλεσμα

Πόρτα με κωδικό (Numpad widget)

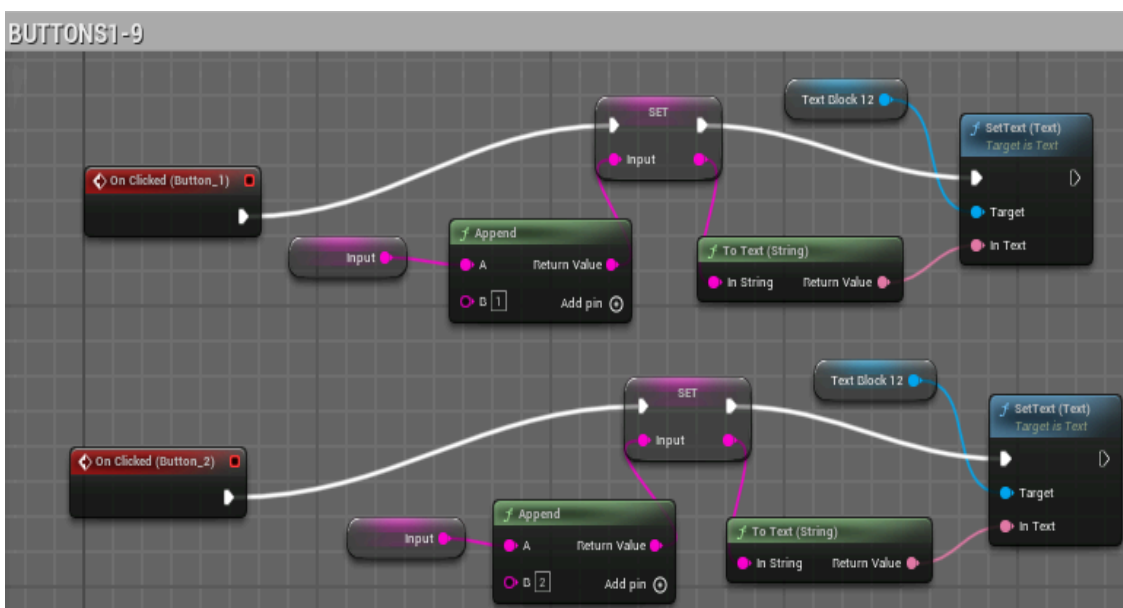
Στο design του συγκεκριμένου widget (βλ. [Εικόνα 84.](#)), μέσα στο canvas panel, περιλαμβάνονται εννέα κουμπιά με την αρίθμηση 0 έως 9 και τρία ακόμη για τις επιλογές Enter, Clear και Exit, ως επιλογές του χρήστη. Επιπλέον, υπάρχει και ένα Text (Enter Code), στο στη συνέχεια εμφανίζεται ο κωδικός που πληκτρολογεί ο χρήστης.



Εικόνα 84. Numpad Designer

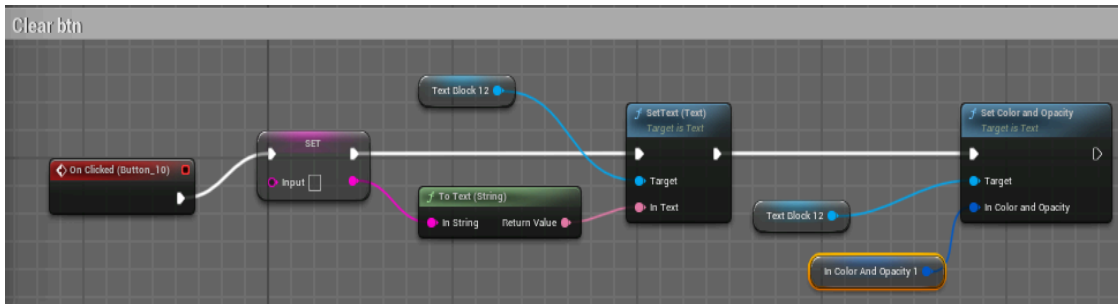
Στο event graph του widget, αρχικά ορίσαμε δύο μεταβλητές τύπου string, τις Input για τον κωδικό που πληκτρολογεί ο παίκτης και CorrectPassword με default value τον σωστό κωδικό που ανοίγει την πόρτα (1956).

- **Buttons 0-9:** Για όλα τα κουμπιά αριθμών, δημιουργήσαμε το αντίστοιχο γράφημα με το παρακάτω (βλ. [Εικόνα 85.](#)). Αν ο παίκτης πατήσει το κουμπί με τον αριθμό 1 (On Clicked(Button1)), τότε στο string Input προστίθεται στο τέλος ο αριθμός 1 (Append) και εμφανίζεται στη θέση του Text (SetText).



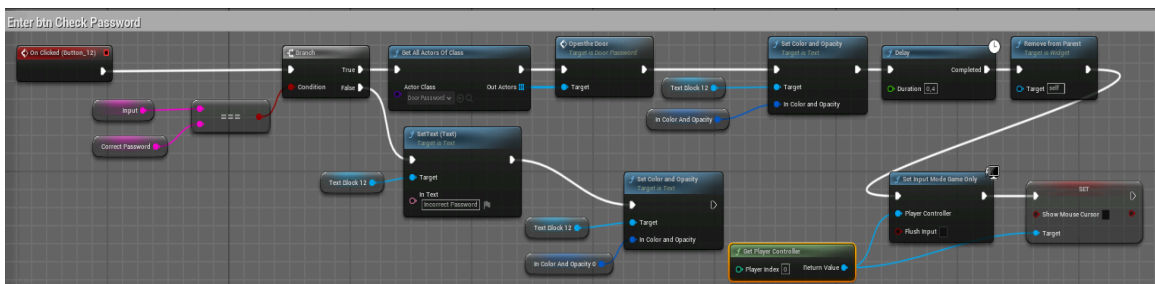
Εικόνα 85. Numpad Event Graph (Buttons 0-9)

- **Button_Clear:** Αν ο παίκτης πατήσει το κουμπί Clear (βλ. [Εικόνα 86.](#)), τότε η σειρά αριθμών που έχει μέχρι στιγμής πληκτρολογήσει αδειάζει και μπορεί να ξεκινήσει από την αρχή.



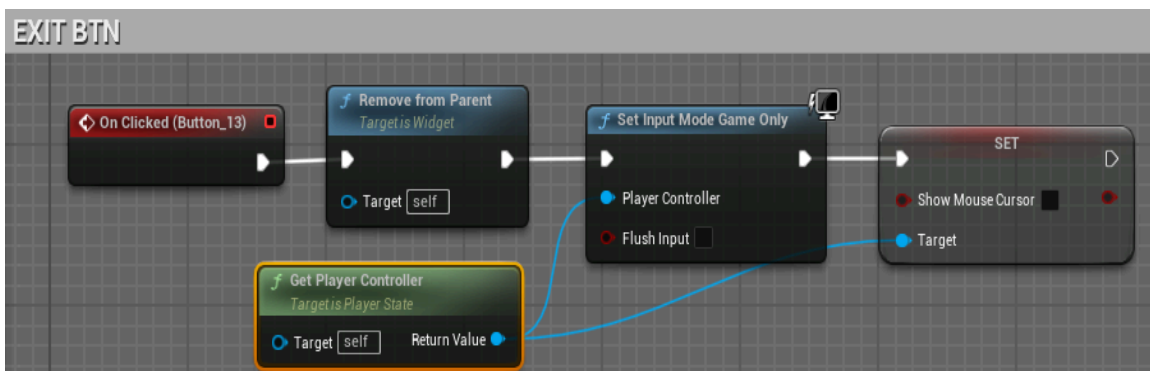
Εικόνα 86. Numpad Event Graph (Button Clear)

- **Button Enter:** Αν πατήσει το κουμπί Enter (βλ. [Εικόνα 87.](#)), τότε γίνεται έλεγχος αν ο Input κωδικός και ο CorrectPassword κωδικός ταυτίζονται. Αν όχι, τότε στο χρήστη εμφανίζεται το μήνυμα “Incorrect Password”. Αν ναι, τότε ενεργοποιείται το συμβάν OpenTheDoor της πόρτας με τον κωδικό ώστε να ανοίξει, και αφού εμφανιστεί το μήνυμα “Correct Password” το widget Numpad αφαιρείται από την οθόνη του παίκτη.



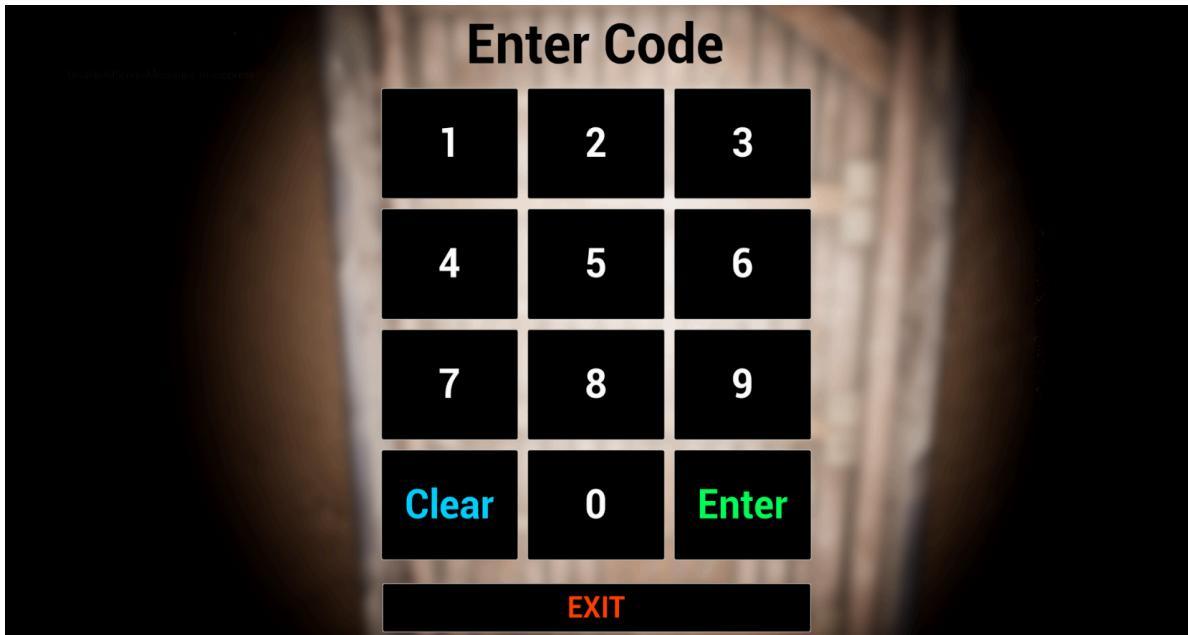
Εικόνα 87. Numpad Event Graph (Button Enter)

- **Exit_Button:** Τέλος, αν ο παίκτης πατήσει το πλήκτρο Exit (βλ. [Εικόνα 88.](#)), τότε το Numpad αφαιρείται από την οθόνη και το παιχνίδι τίθεται και πάλι σε λειτουργία Game only με απενεργοποιημένο τον κέρσορα.



Εικόνα 88. Numpad Event Graph (Button Exit)

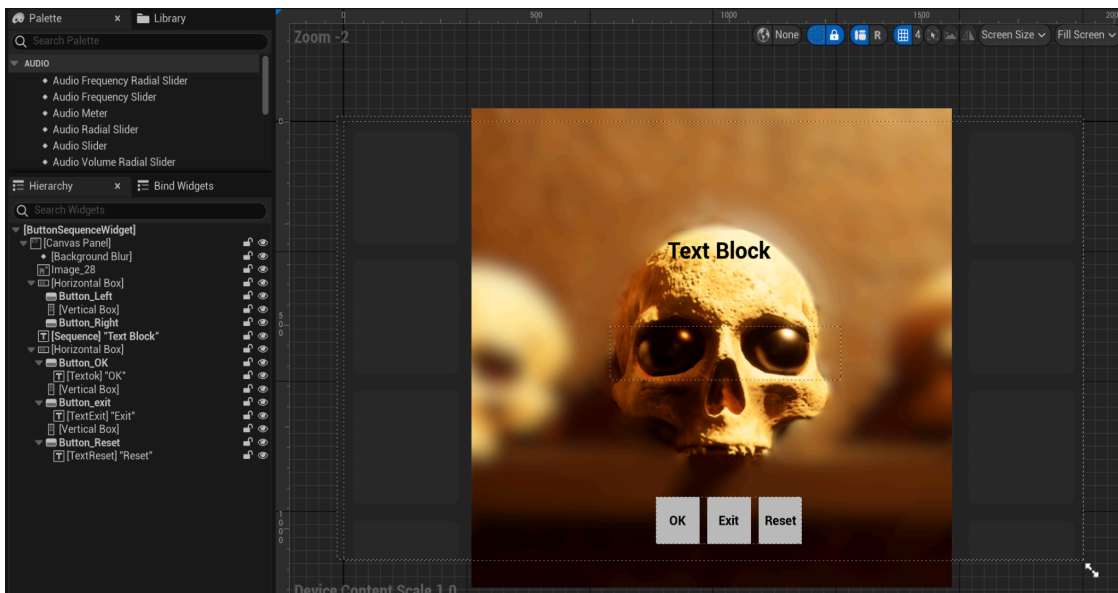
Στη παρακάτω εικόνα (βλ. [Εικόνα 89.](#)), παρουσιάζουμε το τελικό αποτέλεσμα του widget στη οθόνη του παίκτη, όταν αλληλεπιδράσει με την πόρτα κατά τη διάρκεια του Gameplay.



Εικόνα 89. Numpad Τελικό αποτέλεσμα

Διπλή πόρτα Great Hall/ Κρανίο (ButtonSequenceDoorWidget)

Αρχικά, η διάταξη του ButtonSequenceDoorWidget, που βρίσκεται στη σελίδα Design (βλ. [Εικόνα 90.](#)), περιλαμβάνει μέσα στο canvas panel μια εικόνα ενός κρανίου με δύο κουμπιά για μάτια, τα Button_Left και Button_Right αντίστοιχα. Στο μέτωπο του κρανίου υπάρχει ένα text (Sequence), που τυπώνει δυναμικά την μεταβλητή PlayerInput, που αναλύουμε στην ενότητα του graph του συγκεκριμένου widget. Υπάρχουν τρία ακόμα κουμπιά τα οποία είναι Button_Ok, Button_Exit και Button_Reset εκ των οποίων μόνο το δεύτερο είναι αρχικά Visible.

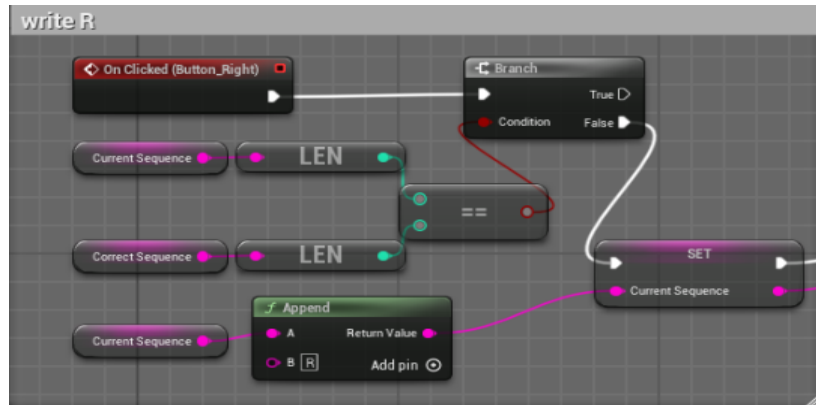


Εικόνα 90. Button Sequence Door Designer

Το Blueprint του συγκεκριμένου widget, της σελίδας Graph, αποτελείται από αρκετά διαγράμματα ροής με την υλοποίηση αυτών να αναλύεται βήμα βήμα παρακάτω.

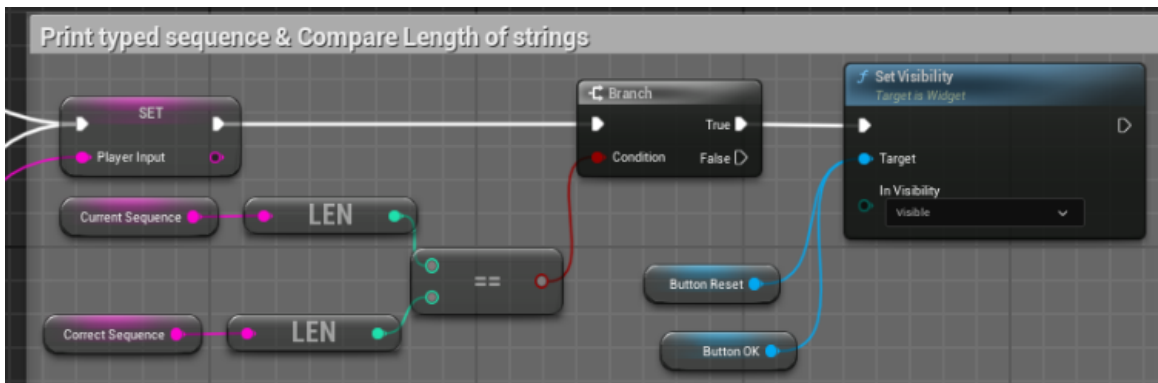
Αρχικά δημιουργήσαμε τις string μεταβλητές CorrectSequence, CurrentSequence και PlayerInput. Η CorrectSequence έχει ως default value τον σωστό συνδυασμό κουμπιών για να ανοίξει η πόρτα DoubleDoor (RLLRLLRR).

- **"Button_Left/Right:** Όταν πατηθεί είτε το δεξί κουμπί είτε το αριστερό (On Button Clicked) ακολουθούμε την ίδια υλοποίηση (βλ. [Εικόνα 91.](#)). Μέχρι ο παίκτης να φτάσει το ίδιο πλήθος χαρακτήρων στα αλφαριθμητικά CorrectSequence και CurrentSequence συμπληρώνονται χαρακτήρες στο CurrentSequence. Για το Button_Left το "L" και αντιστοίχα για το Button_Right το "R".



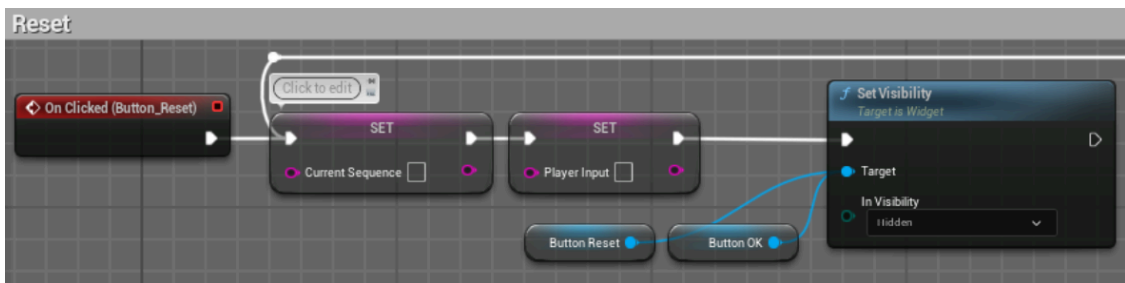
Εικόνα 91. Button Sequence Door Event Graph (Button Left/Right)

- **Print Sequence & Visibility:** Ότι πατάει ο παίκτης, αριστερό ή δεξί κουμπί, το ενσωματώνουμε στην PlayerInput (βλ. [Εικόνα 92.](#)), η οποία τυπώνει το αποτέλεσμα στη θέση του Sequence στο Design. Όταν ο παίκτης δώσει τους προαπαιτούμενους χαρακτήρες στο CurrentSequence μέσω των κουμπιών, τότε εμφανίζονται τα κουμπιά Button_Ok και Button_Reset.



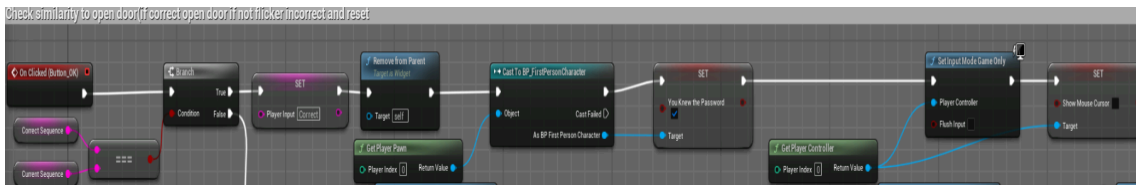
Εικόνα 92. Button Sequence Door (Print Sequence, Buttons Reset/Ok Visibility)

- **Button_Reset:** Όταν ο παίκτης πατήσει το Reset (βλ. [Εικόνα 93.](#)), τότε ορίζουμε τα strings CurrentSequence και PlayerInput κενά και εξαφανίζουμε τα κουμπιά Button_Ok και Button_Reset. Την ίδια ακριβώς υλοποίηση ακολουθούμε αν ο παίκτης δώσει λάθος κωδικό όπως θα δούμε παρακάτω.



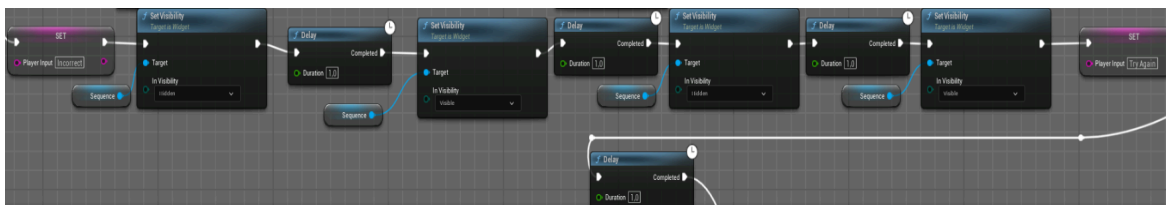
Εικόνα 93. Button Sequence Door (Button Reset)

- Button_Ok:** Όταν πατηθεί το κουμπί ok (βλ. [Εικόνα 94.](#)), γίνεται ένας έλεγχος ορθότητας της αλληλουχίας που έδωσε ο παίκτης. Αν έχει δώσει τη σωστή αλληλουχία τότε τυπώνεται “Correct” και το widget εξαφανίζεται από την οθόνη του παίκτη. Καλώντας τον First Person, ορίζουμε τη μεταβλητή του YouKnewThePassword ως true. Θέτουμε τη λειτουργία του παιχνιδιού σε Game only και εξαφανίζουμε τον κέρσορα, ώστε να προχωρήσουμε στο άνοιγμα της διπλής πόρτας.



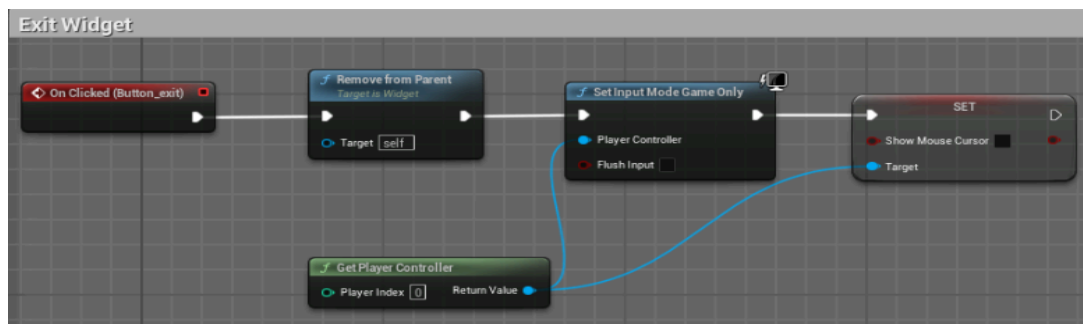
Εικόνα 94. Button Sequence Door (Button Ok)

Αν ο παίκτης δεν δώσει την σωστή αλληλουχία (βλ. [Εικόνα 95.](#)), τυπώνεται “Incorrect” το οποίο αναβοσβήνει στην οθόνη και μετά τυπώνεται το μήνυμα “Try Again”. Τέλος, όπως προαναφέρθηκε επαναλαμβάνεται η διαδικασία του reset.



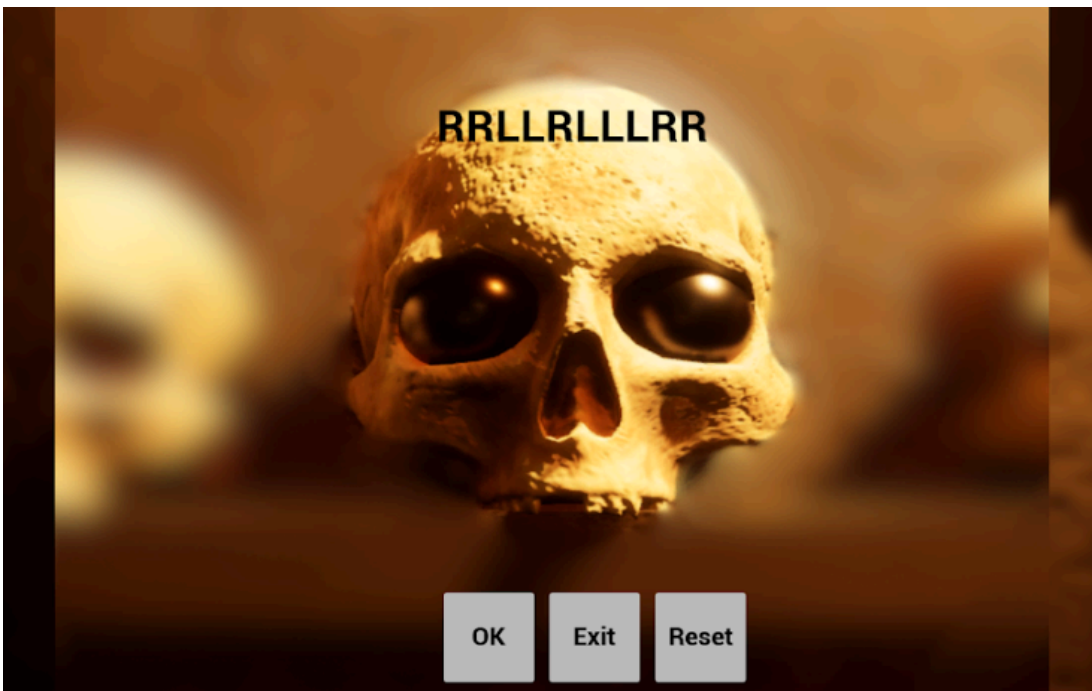
Εικόνα 95. Button Sequence Door (Button Reset συνέχεια)

- Button_Exit:** Όταν ο παίκτης πατήσει το Exit (βλ. [Εικόνα 96.](#)), το widget αφαιρείται από την οθόνη του παίκτη και το παιχνίδι μπαίνει σε λειτουργία Game Only με τον κέρσορα να μην φαίνεται.



Εικόνα 96. Button Sequence Door (Button Exit)

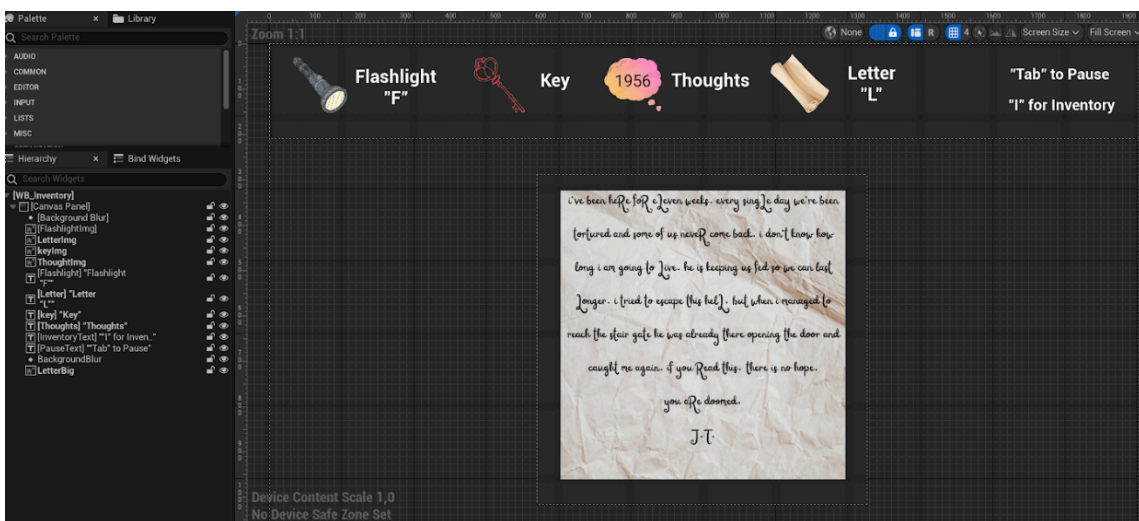
Στην παρακάτω εικόνα (βλ. [Εικόνα 97.](#)) είναι το αποτέλεσμα αυτής της υλοποίησης στην οθόνη του παίκτη.



Εικόνα 97. Button Sequence Door Τελικό αποτέλεσμα

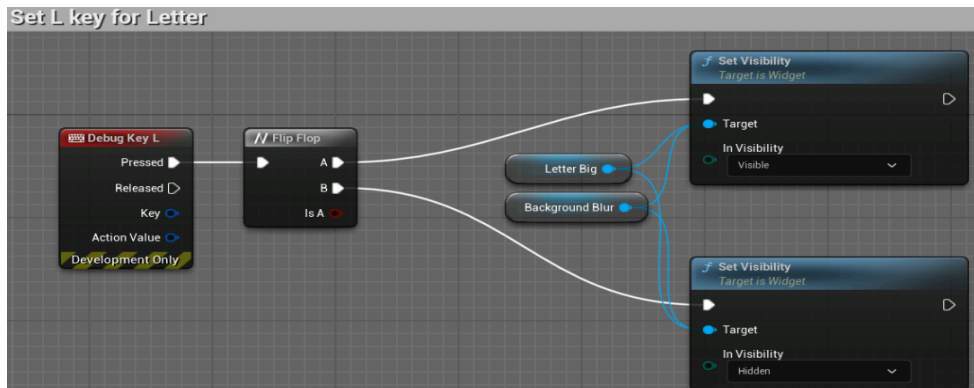
Αποθήκη Παίκτη (Inventory)

Στη σελίδα του Designer, η διάταξη του Inventory, (βλ. [Εικόνα 98.](#)), περιλαμβάνει μέσα στο canvas panel πέντε εικόνες και έξι κείμενα. Κάθε μια εικόνα από τις πάνω αντιπροσωπεύει τι έχει ο παίκτης πάνω του από αντικείμενα και τι έχει βρει από στοιχεία. Η εικόνα FlashlightImg και τα λεκτικά flashlight, InventoryText και PauseText είναι μόνιμα Visible στον παίκτη, τα υπόλοιπα όμως που αφορούν στοιχεία είναι Hidden και έχουν οριστεί ως μεταβλητές.



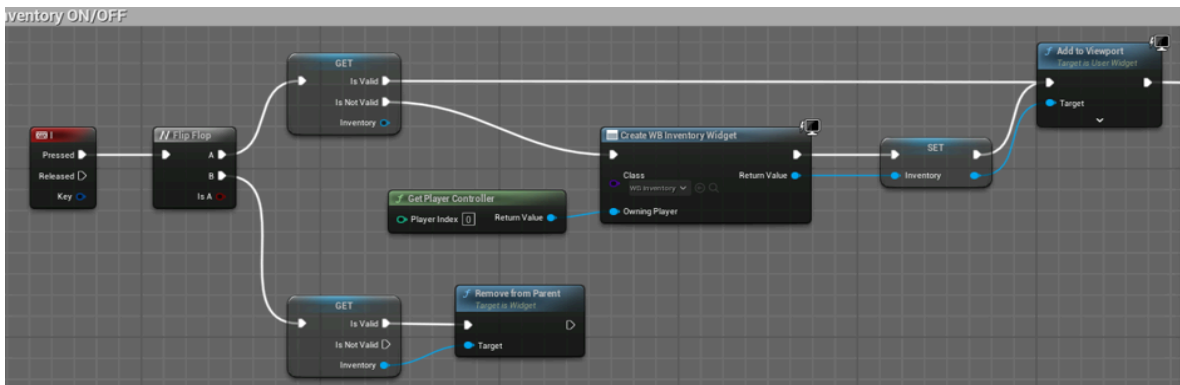
Εικόνα 98. Inventory Designer

Όσον αφορά το graph του widget, όταν ο παίκτης έχει το βρει γράμμα, με το πάτημα του πλήκτρου “L”, (βλ. [Εικόνα 99.](#)), το εμφανίζει και εξαφανίζει από την οθόνη, σε περίπτωση που χρειάζεται να ανατρέξει και να ξαναδιαβάσει.



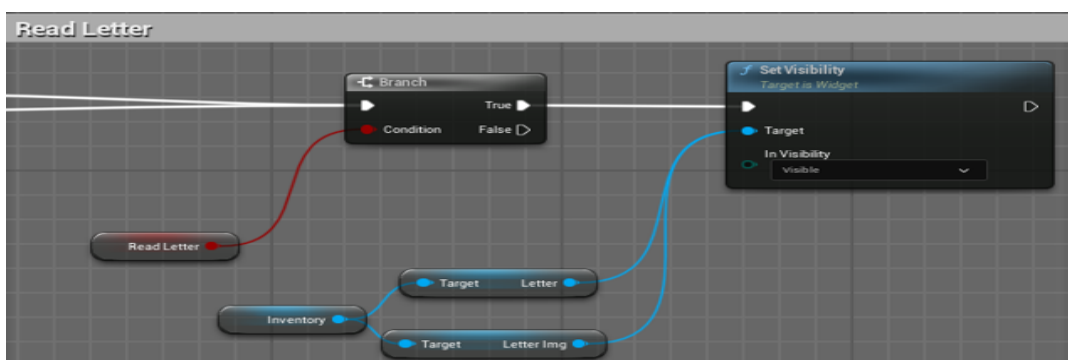
Εικόνα 99. Inventory Event Graph (Visibility of Letter)

Δεν γίνεται όμως να εξηγήσουμε την λειτουργία του Inventory αν δεν αναλυθεί το διάγραμμα ροής αυτού στο Blueprint του First Person. Όταν ο παίκτης πατήσει το πλήκτρο “I”, (βλ. [Εικόνα 100.](#)), μπορεί και εναλλάσει άνοιγμα/κλείσιμο του widget.



Εικόνα 100. Inventory in First Person Event Graph

Ακολουθούν τρεις έλεγχοι. Ελέγχονται οι boolean μεταβλητές SeenPhoto, HasKey και ReadLetter (βλ. [Εικόνα 101.](#)). Όποια από αυτές τις μεταβλητές είναι true εμφανίζει την εικόνα και το λεκτικό που αντιστοιχεί στον εκάστοτε γρίφο. Πιο αναλυτικά, η SeenPhoto εμφανίζει την εικόνα ThoughtsImg και το λεκτικό Thoughts, η HasKey εμφανίζει την εικόνα KeyImg και το λεκτικό Key και τέλος η ReadLetter εμφανίζει την εικόνα LetterImg και το λεκτικό Letter.



Εικόνα 101. Inventory in First Person Event Graph συνέχεια

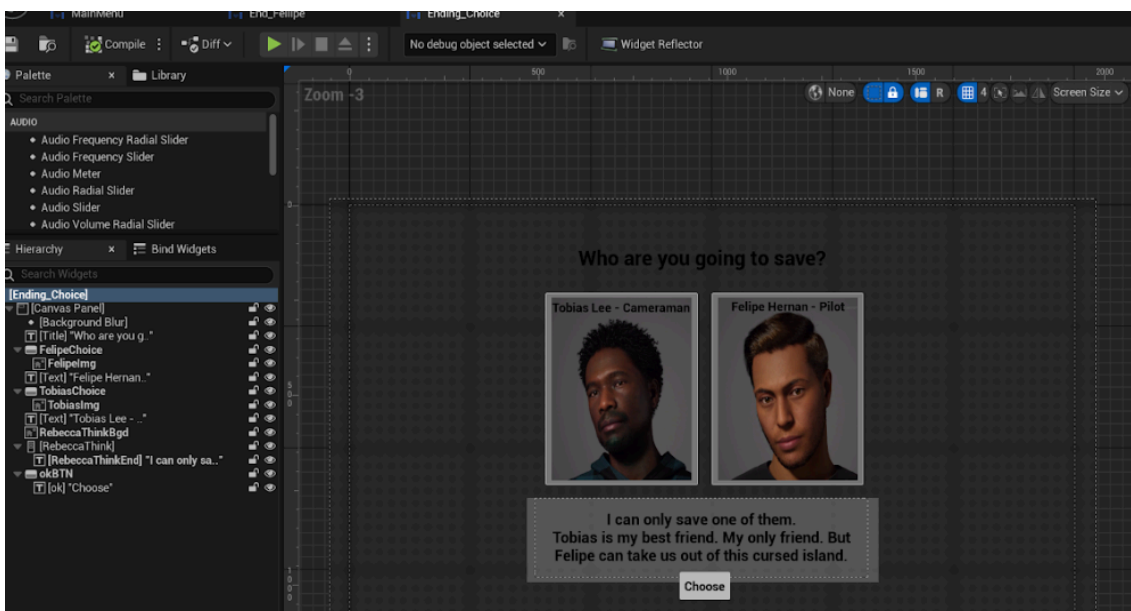
Τέλος, παρατίθεται το τελικό αποτέλεσμα στο χρήστη, (βλ. [Εικόνα 102.](#)), χωρίς αυτός να έχει μαζέψει στοιχεία ή αντικείμενα.



Εικόνα 102. Inventory Τελικό αποτέλεσμα

Επιλογή Τέλους (EndingChoice)

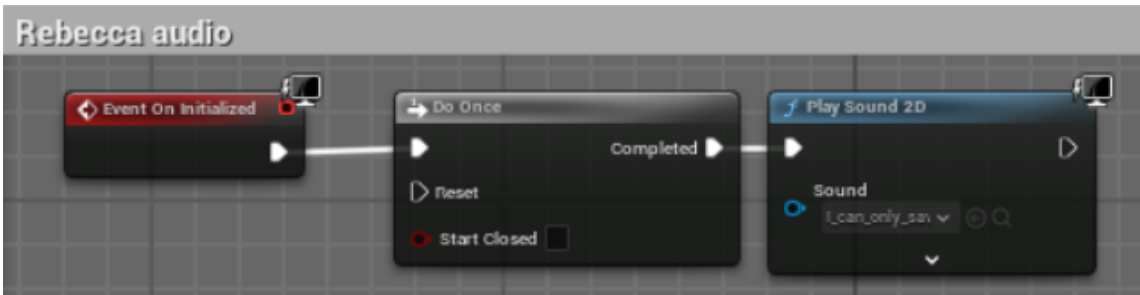
Η διάταξη στη σελίδα Designer του EndingChoice, (βλ. [Εικόνα 103.](#)), περιλαμβάνει μέσα στο canvas panel ένα τίτλο (text) κάτω από τον οποίο υπάρχουν δύο κουμπιά, το κουμπί (FelipeChoice) με εικόνα τον Felipe και λεκτικό με το όνομα και την ιδιότητα του και ομοίως κουμπί (TobiasChoice). Μια ωχρή εικόνα για φόντο του κειμένου - συμπεράσμα της Rebecca και ένα κουμπί (okBTN) που γράφει “Choose” που αρχικά είναι Hidden.



Εικόνα 103. Ending Choice Designer

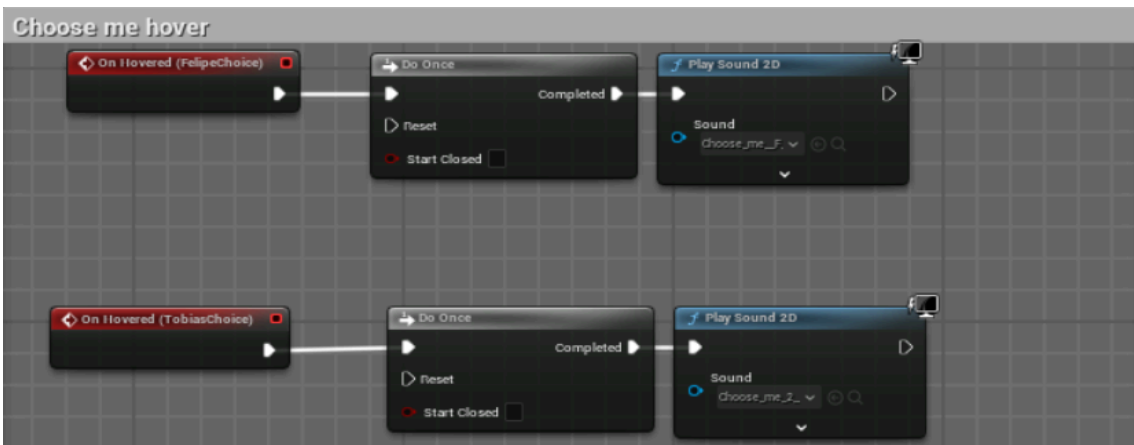
Σχετικά με το graph αυτού του widget, το οποίο αποτελείται από τέσσερα διαγράμματα ροής, δημιουργήσαμε μια μεταβλητή Choise(Integer).

- **Audio Rebecca:** Όταν το widget αρχικοποιηθεί (Event on Initialized) (βλ. [Εικόνα 104.](#)) θα παίξει μία φορά το συμπέρασμα σε ηχητικό από την πρωταγωνίστρια.



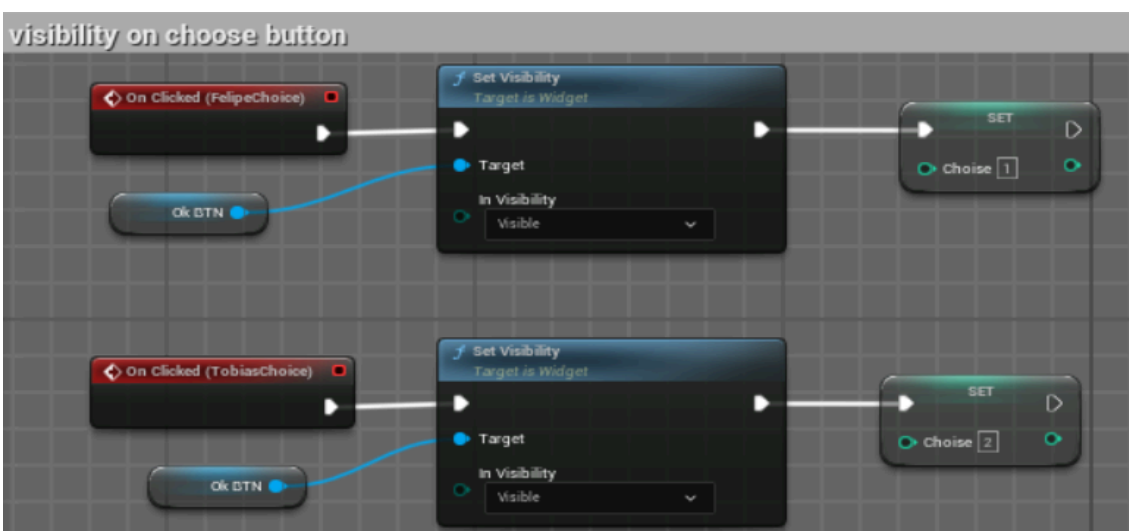
Εικόνα 104. Ending Choice Event Graph (Audio)

- **Hover Sound “Choose me”:** Όταν ο κερσορας περάσει πάνω είτε από το κουμπί FelipeChoice είτε TobiasChoice (On Hovered) (βλ. [Εικόνα 105.](#)) ακούγονται ηχητικά προτροπής για την επιλογή διάσωσης από τον παίκτη.



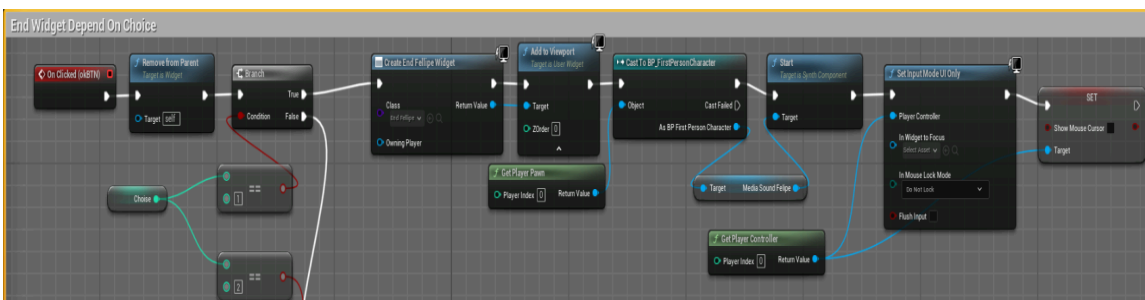
Εικόνα 105. Ending Choice Event Graph (Audio on hovered)

- **FelipeChoice/TobiasChoice Button:** Πατώντας ο παίκτης είτε την εικόνα με τον Felipe είτε του Tobias (βλ. [Εικόνα 106.](#)), εμφανίζεται το okBTN και ορίζεται η μεταβλητή choise (Felipe=1 & Tobias=2).



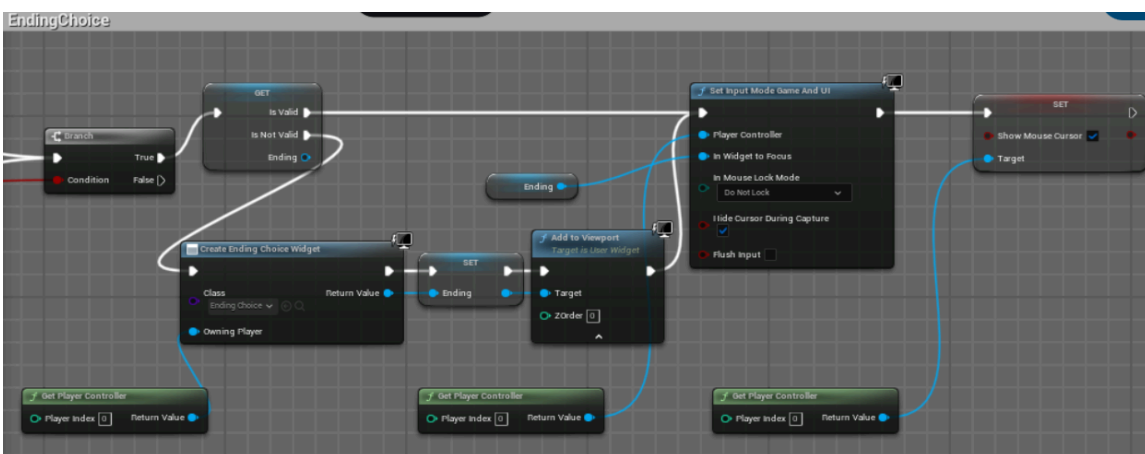
Εικόνα 106. Ending Choice Event Graph (Buttons Felipe/Tobias Choice)

- End Widget Depend on Choice (okBTN):** Πατώντας το κουμπί Choose (okBTN), (βλ. [Εικόνα 107.](#)), το widget εξαφανίζεται από την οθόνη του παίκτη. Ελέγχεται η τιμή της μεταβλητής Choise. Η υλοποίηση που πραγματοποιήθηκε ανεξάρτητα επιλογής είναι η ίδια με διαφορά τα widgets που δημιουργούνται και τα ηχητικά των Videos που ακούγονται. Είτε ο παίκτης επιλέξει 1 είτε 2, δημιουργείτε το widget της αντίστοιχης επιλογής (EndFelipe για choise=1 και EndTobias για choise=2) και εμφανίζεται στην οθόνη του παίκτη. Γίνεται cast ο First Person ώστε να τραβήξουμε το αντίστοιχο ηχητικό του εκάστοτε Video, Media Sound Felipe και Media Sound Tobias, το οποίο και ξεκινά να παίζει. Τέλος, το παιχνίδι τίθεται σε λειτουργία UI Only με κρυφό κέρσορα.



Εικόνα 107. Ending Choice Event Graph (End Widget depend on Choice)

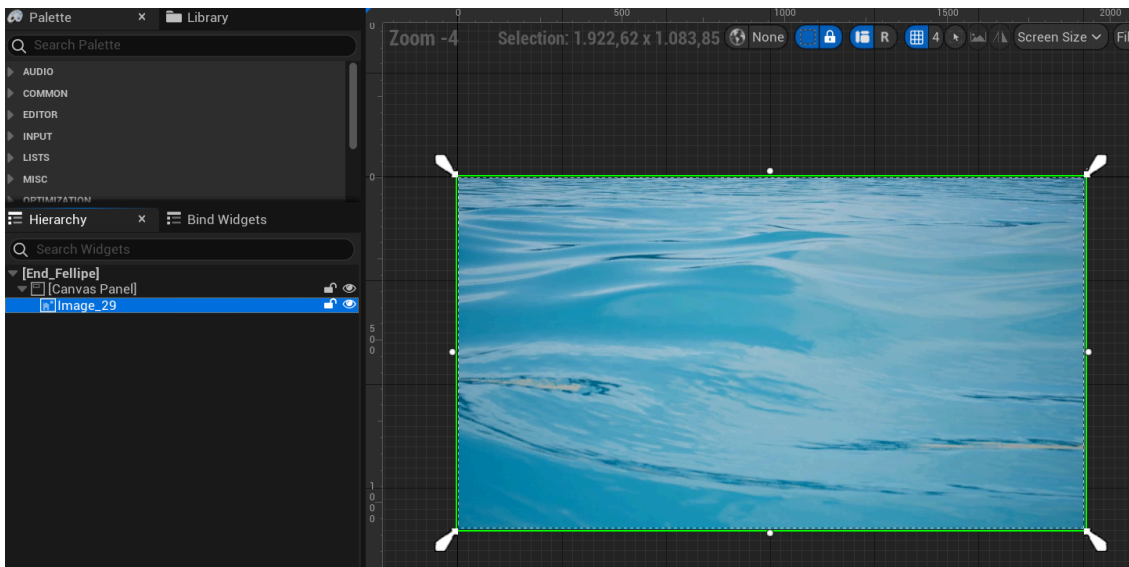
Το συγκεκριμένο widget καλείται από το Blueprint του First Person, (βλ. [Εικόνα 108.](#)), όταν και εάν ο παίκτης έχει μιλήσει και με τους δυο Npcs, όπως θα δούμε αναλυτικά στην επόμενη ενότητα. Η υλοποίηση είναι παρόμοια αυτήν του widget Photo.



Εικόνα 108. Ending Choice in First Person Event Graph

Τέλος Felipe/Tobias (EndFelipe)/(EndTobias)

Αυτά τα δύο widget διαφορετικού τέλους, είναι αρκετά όμοια εκτός κάποιων επιλογών. Η διάταξη της σελίδας Designer (βλ. [Εικόνα 109.](#)) αποτελείται από ένα Canvas Panel του οποίου την επιφάνεια καλύπτει πλήρως μια εικόνα, στην οποία και εφαρμόσαμε το αντίστοιχο video τέλους ανάλογα το widget και αρχικά είναι Hidden.



Εικόνα 109. End Felipe/Tobias Designer

Προτού αναλύσουμε το graph, θα πρέπει να εξηγήσουμε πως εισαγάγαμε τα video μας μέσα στο Unreal Engine και τα καταστήσαμε λειτουργικά. Αρχικά κάναμε import τα δύο αρχεία πιθανού τέλους .mp4 και στη συνέχεια δημιουργήσαμε ένα Media Player για το καθένα, επιλέγοντας επίσης τη δυνατότητα να χρησιμοποιηθούν ως textures. Σε κάθε Media Player φορτώσαμε το αντίστοιχο video. Από το texture που προέκυψε δημιούργησαν ένα υλικό (material). Στη σελίδα του material, αλλάξαμε την επιλογή material από Surface σε User Interface, έτσι ώστε να μπορεί να χρησιμοποιηθεί μέσω blueprint. Συνδέσαμε στο graph που προέκυψε το RGB με το Final Color και τα texture μας ήταν έτοιμα για χρήση, χωρίς όμως να περιλαμβάνουν ήχο.

Το Blueprint του συγκεκριμένου widget, στην σελίδα Graph (βλ. [Εικόνα 110.](#)), αποτελείται από ένα διάγραμμα ροής. Αρχικά, δημιουργήσαμε μία μεταβλητή Media Player τύπου Media Player Type και στο default value της εφαρμόσαμε το Media Player του αντίστοιχου video τέλους. Κατά τη δημιουργία του widget (Event Construct) καλέσαμε την Open Source με target το Media Player που μόλις δημιουργήσαμε. Με μια μικρή καθυστέρηση, κάναμε την εικόνα με το texture του video Visible. Προσθέσαμε μια καθυστέρηση που ισούται με το μήκος του video συν την προηγούμενη καθυστέρηση και έπειτα αφαιρέσαμε το widget από την οθόνη του παίκτη. Τέλος, ανοίξαμε το Level MainMenu, ώστε όταν τελειώσει το video να πετάει τον παίκτη στο μενού του παιχνιδιού και ενεργοποιήσαμε το ποντίκι.



Εικόνα 110. End Felipe/Tobias Event Graph (Video player)

Εισαγωγικό Video (Intro Seq Widget)

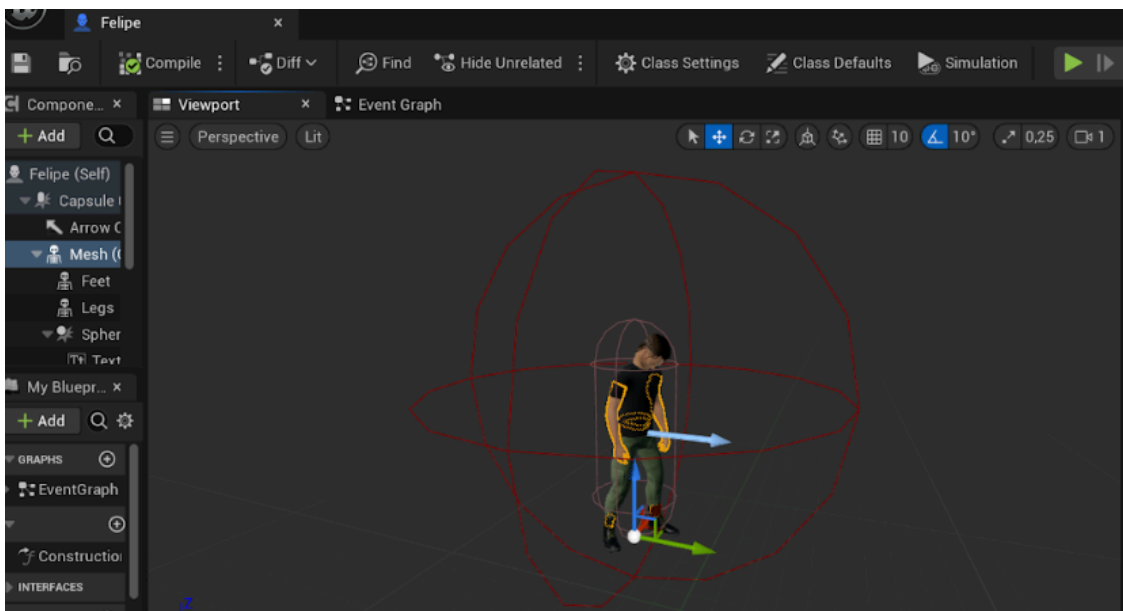
Όσον αφορά το video κατά την έναρξη του παιχνιδιού, ακολουθήσαμε την ίδια διαδικασία ακριβώς με αυτήν του Ending Felipe/Tobias, προσαρμόζοντας τα delays στο μέγεθος του Video εισαγωγής.

Blueprints Χαρακτήρων

Για την υλοποίηση των NPCs δημιουργήσαμε δύο χαρακτήρες, τον Felipe και τον Tobias, χρησιμοποιώντας την Blueprint Class, τύπου Character. Τα βήματα υλοποίησης που ακολουθήθηκαν είναι ακριβώς τα ίδια και για τους δύο Non-Player Characters, όπως αναλύεται παρακάτω, με ελάχιστες διαφορές.

Δημιουργία Character - Metahuman

Αφού δημιουργήσαμε τον Character Felipe, στο Viewport (βλ. [Εικόνα 111.](#)) ορίσαμε ως σώμα το mesh του metahuman που φτιάξαμε για αυτόν τον χαρακτήρα. Απλά αντιγράφοντας τα Face (Beard, Mustache, Eyelashes, Fuzz, Hair, Eyebrows), Torso, Legs, Feet καταφέραμε να έχει αυτός ο Character τα επιθυμητά χαρακτηριστικά του Metahuman που σχεδιάσαμε.

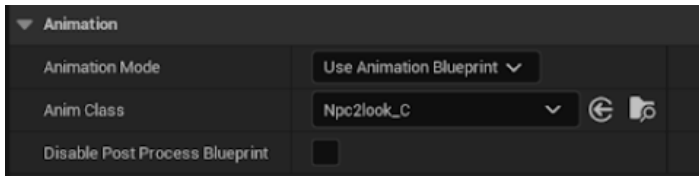


Εικόνα 111. Viewport Felipe

Εκτός της εμφάνισης, και οι δύο χαρακτήρες χρειάστηκε να έχουν κάποιου είδους κίνηση. Έτσι, έπρεπε να δημιουργηθεί ένα animation, το οποίο θα επιτρέπει στους χαρακτήρες να έχουν κάποια κινησιολογία. Για τους σκοπούς αυτής της υλοποίησης, δώσαμε έμφαση στο κεφάλι και στα μάτια, τα οποία ακολουθούν τον παίκτη, ενώ το υπόλοιπο σώμα είναι σε Idle Animation.

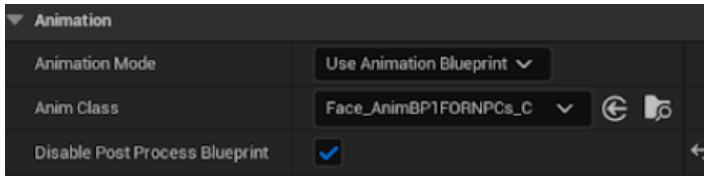
Animations In Game

Όσον αφορά την κίνηση του λαιμού, δημιουργήσαμε ένα Animation Blueprint (Npc2look), το οποίο και εφαρμόσαμε (βλ. [Εικόνα 112.](#)) στο Mesh, Torso, Feet και Legs του Character.



Εικόνα 112. Felipe Animation (Body)

Για το Head, εφαρμόσαμε το animation Face_AnimBP1FORNPCs (βλ. [Εικόνα 113.](#)) που είναι αντίγραφο του ήδη υπάρχοντος Face_AnimBP.

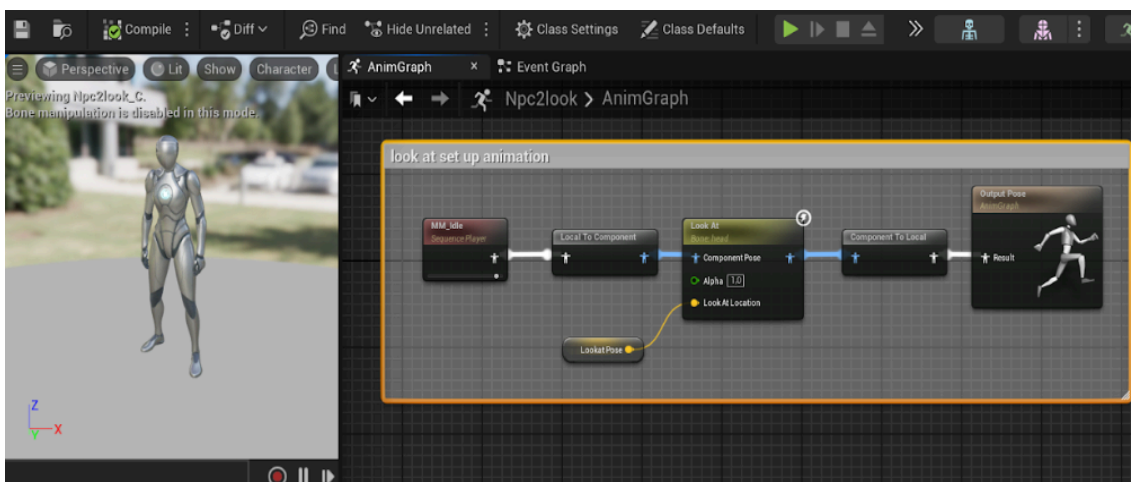


Εικόνα 113. Felipe Animation (Face)

Παρακάτω αναλύονται το Npc2look animation και οι αλλαγές που κάναμε στο Face_AnimBP1FORNPCs για την κίνηση των ματιών.

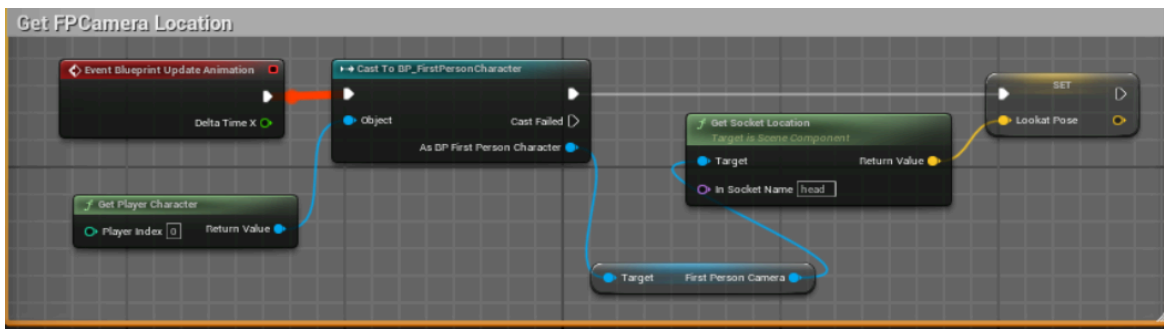
Npc2look

Ξεκινώντας με το AnimGraph της Npc2look, επιλέξαμε από τον Asset Browser το MM_Idle με λειτουργία Loop Animation. Στη συνέχεια, χρησιμοποιήσαμε την Look At (βλ. [Εικόνα 114.](#)), ώστε να ορίσουμε που θα κοιτάει το κεφάλι του χαρακτήρα, με Look At Location το διάνυσμα LookAtPose, όπου και ορίζουμε στην Event Graph παρακάτω. Τέλος, ολοκληρώνεται η τελική πόζα του χαρακτήρα.



Εικόνα 114. Npc2look AnimGraph

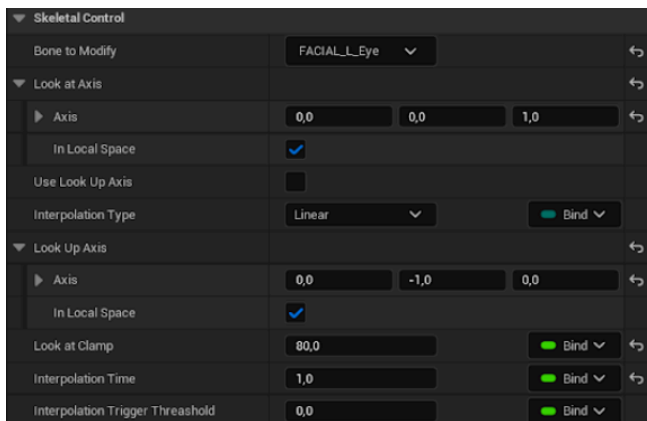
Στη σελίδα του Event Graph (βλ. [Εικόνα 115.](#)), ορίσαμε την ακριβή τοποθεσία που θέλουμε να κοιτάει ο χαρακτήρας που δεν είναι άλλη από την Camera του First Person. Για να επιτευχθεί αυτό, καλούμε τον First Person, κάνουμε get την First Person Camera, όπου υποθετικά είναι το κεφάλι του παίκτη, και αφού ορίσουμε μια μεταβλητή διάνυσμα (LookAtPose(vector)), της αναθέτουμε την τιμή που επιστράφηκε.



Εικόνα 115. Npc2look Event Graph

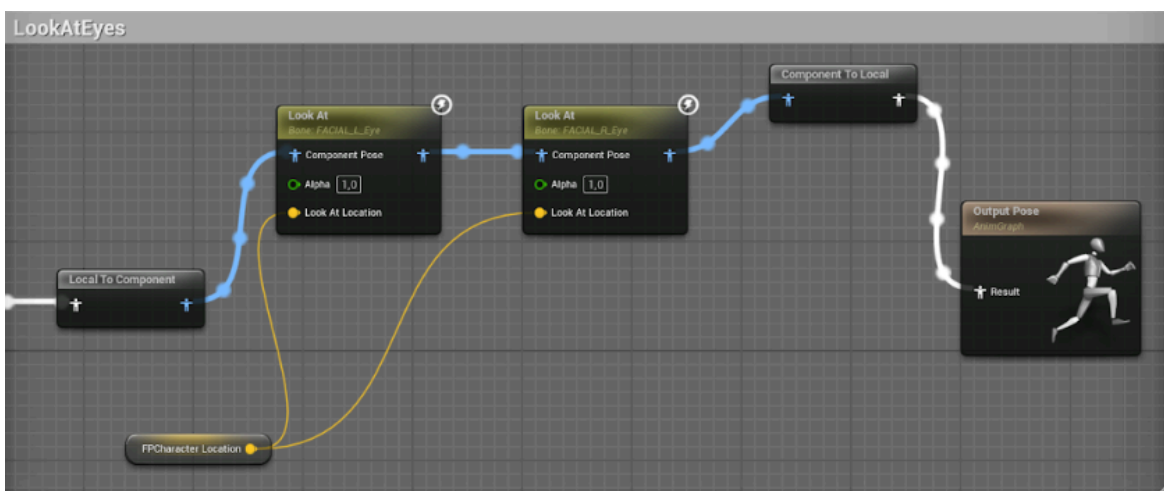
Eyes Movement

Προχωρώντας στα μάτια, στο animation Face_AnimBP1FORNPCs, στη σελίδα του AnimGraph, ακριβώς πριν τον κόμβο Output Pose και μετά το ήδη υπάρχον ARKit Head Rotation, προσθέσαμε δύο Look At, η μία ορισμένη στο Bone FACIAL_L_Eye η άλλη FACIAL_R_Eye. Χρειάστηκε να ορίσουμε τα Axis Look At και Look Up (βλ. [Εικόνα 116.](#)) για να πετύχουμε το επιθυμητό αποτέλεσμα σε κάθε μάτι, όπως επίσης και το Clamp για να μην χάνονται στο εσωτερικό του κεφαλιού.



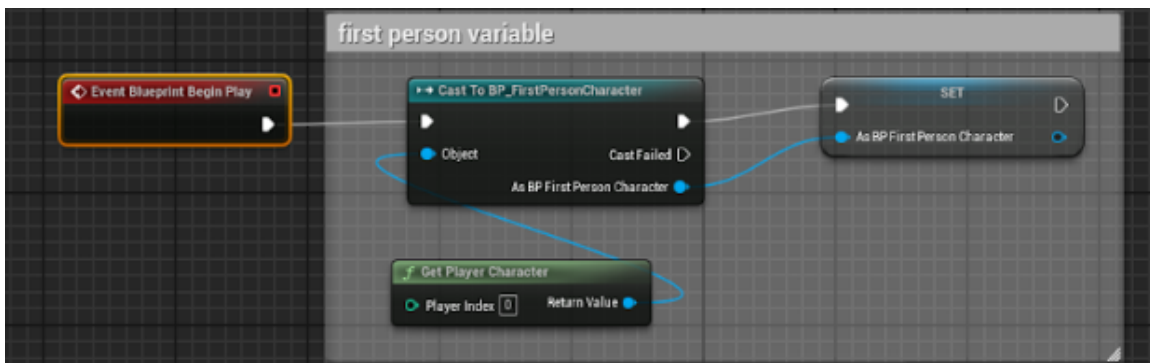
Εικόνα 116. Look At node Bone Modifications (Left/Right Eye)

Το Look At Location και των δύο ματιών τίθεται ως FPCharacterLocation (βλ. [Εικόνα 117.](#)), το οποίο και ορίζεται στο Event Graph της Face_AnimBP1FORNPCs.



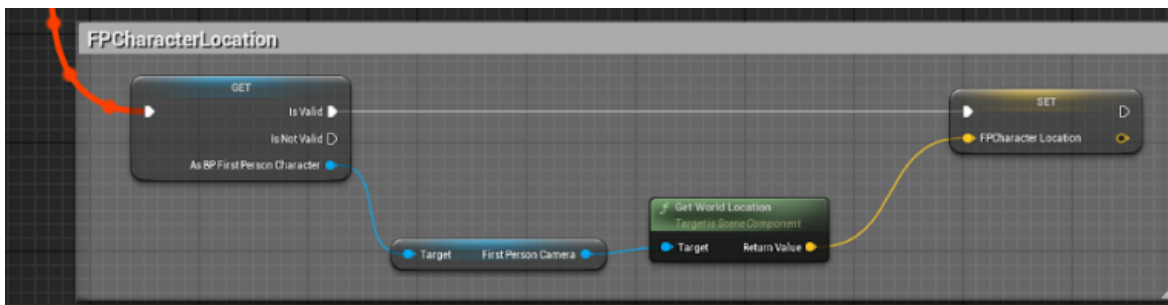
Εικόνα 117. Face_AnimBP1FORNPCs AnimGraph

Αρχικά, καλούμε τον First Person (βλ. [Εικόνα 118.](#)) και τον ορίζουμε ως μεταβλητή.



Εικόνα 118. Face_AnimBP1FORNPCs Event Graph (Find Location)

Έπειτα, από συνδέουμε στο False του branch της Set ARKit Head Rotation το παρακάτω διάγραμμα ροής. Έχοντας δημιουργήσει μια vector μεταβλητή FPCharacterLocation, αν η μεταβλητή BPFirstPeronCharacter έχει δημιουργηθεί (βλ. [Εικόνα 119.](#)), κάνουμε get την FirstPersonCamera και βρίσκοντας την τοποθεσία του παίκτη σε σχέση με τον NPC ορίζουμε και την τελική τοποθεσία που κοιτάει.



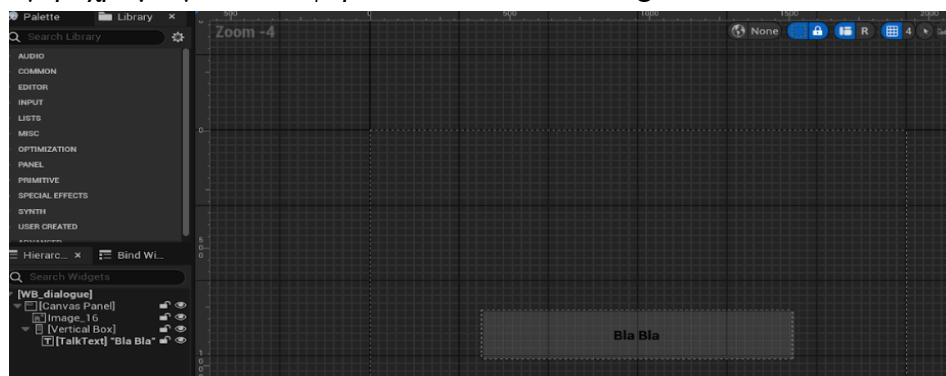
Εικόνα 119. Face_AnimBP1FORNPCs Event Graph (Set Location)

NPCs - Dialogue Widget

Τα Event Graphs και των δύο χαρακτήρων, αφορούν την υλοποίηση του διαλόγου μεταξύ του παίκτη και των NPC. Αυτό επιτυγχάνεται μέσω ενός widget (Dialogue) το οποίο συνδέεται με το UI BP_Dialogue μέσω της συνάρτησης Talk.

Διάλογος NPC (Dialogue)

Η διάταξη του Dialogue στο Designer (βλ. [Εικόνα 120.](#)) περιλαμβάνει μέσα στο canvas panel ένα κουτί για οριοθέτηση της διάταξης (vertical box) και από πίσω του μια εικόνα με αχνό γκρι χρώμα για να διαφοροποιείται από το background.



Εικόνα 120. Dialogue Designer

Μέσα στο vertical box υπάρχει ένα text, το TalkText, το οποίο δεν μας ενδιαφέρει να γράφει κάτι συγκεκριμένο καθώς το κείμενο που βλέπει ο χρήστης αντλείται δυναμικά από τα blueprints των NPC που εξηγούνται παρακάτω.

- **Διάλογοι Felipe Hernan (Pilot - NPC):**

Ο διάλογος που κάνει ο Felipe με τη Rebecca φαίνεται στην εικόνα παρακάτω (βλ. [Εικόνα 121.](#)). Για τις συγκεκριμένες προτάσεις υπάρχουν και τα αντίστοιχα audio.

Dialogue Felipe		5 Array elements
Index [0]		Rebecca? Is this really you?
Index [1]		I am so happy you are alive!
Index [2]		Trueman was torturing us for hours. To see what makes
Index [3]		Tobias is really hurt.
Index [4]		Can you help us? Can you break us out of here?

Εικόνα 121. Felipe Dialogue

- **Διάλογοι Tobias Lee (Cameraman - NPC):**

Αντίστοιχα με τον Felipe, ο διάλογος του Tobias (βλ. [Εικόνα 122.](#)).

Dialogue Tobias		4 Array elements
Index [0]		Rebecca help us.
Index [1]		I cannot move my legs.
Index [2]		I think i will never see my daughter again.
Index [3]		Please tell me you know a way out!?

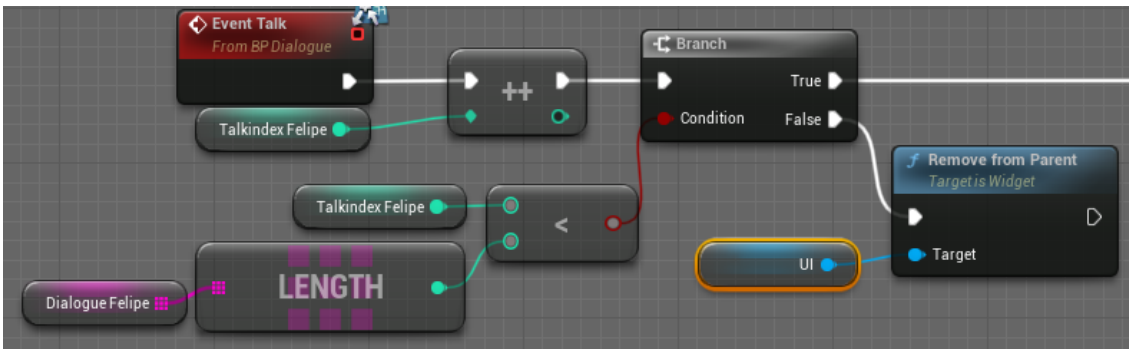
Εικόνα 122. Tobias Dialogue

Στη σελίδα του Event Graph, και των δύο χαρακτήρων ακολουθεί η ίδια υλοποίηση (βλ. [Εικόνα 123.](#)). Για λόγους συντομίας, σε αυτό το σημείο θα περιγράψουμε το graph του ενός εκ των δύο. Αρχικά, δημιουργήσαμε τις εξής μεταβλητές, ένα μετρητή integer (TalkIndexFelipe (-1)), ένα πίνακα string (DialogueFelipe), όπου κάθε θέση του πίνακα είναι και μια πρόταση (Βλ. Εικόνες από πάνω), ένας πίνακας ομιλίας (SpeechFelipe), όπου κάθε θέση του είναι η αντιστοιχία ήχου audio με τις προτάσεις.

VARIABLES	
Components	
DialogueFelipe	String
TalkindexFelipe	Integer
UI	WB Dialogue
Txt_Render	Text Render Actor
SpeechFelipe	Sound Wave

Εικόνα 123. Felipe/Tobias Event Graph (Variables)

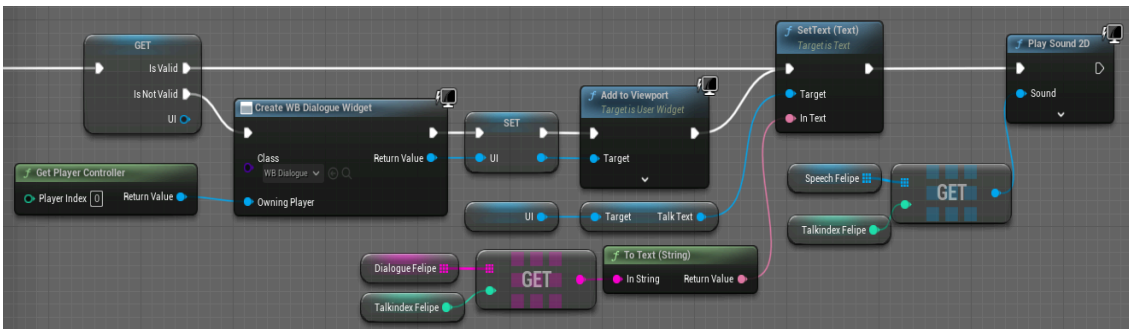
Το graph μας αποτελείται από ένα διάγραμμα ροής (βλ. [Εικόνα 124.](#)) που ξεκινά με ένα γεγονός, Event Talk. Ο μετρητής αρχίζει να μετρά για κάθε θέση του πίνακα Dialogue, και αν τελειώσουν οι προτάσεις αφαιρεί το widget Dialogue από την οθόνη του παίκτη.



Εικόνα 124. Felipe Event Graph (Event talk)

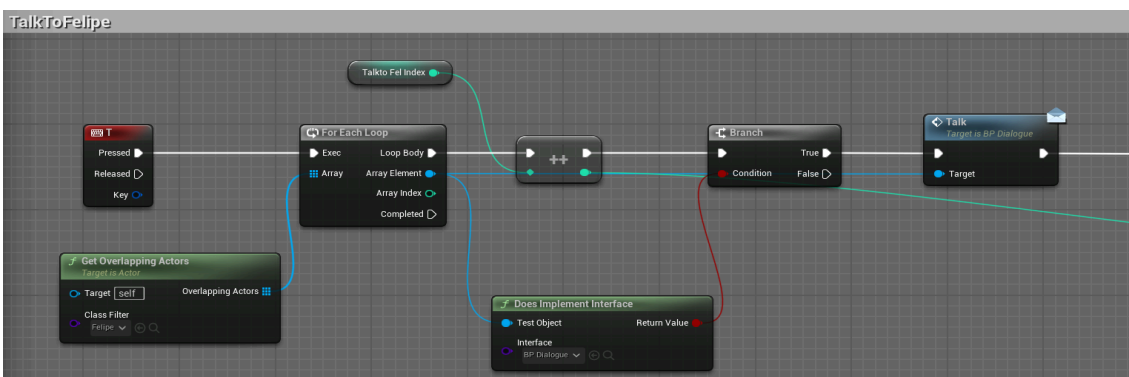
Όσο όμως υπάρχουν θέσεις στον πίνακα (βλ. [Εικόνα 125.](#)), αν δεν έχει δημιουργηθεί το widget Dialogue, τότε δημιουργείται και προστίθεται στην οθόνη του παίκτη (πρώτη φορά) και έπειτα στη θέση του TalkText του Widget γράφεται το string του πίνακα διαλόγου με αριθμό τον εκάστοτε TalkIndex.

Ταυτόχρονα ακούγεται από τον πίνακα Speech το αντίστοιχο ηχητικό. Αν έχει δημιουργηθεί το widget προχωράμε κατευθείαν στην αντικατάσταση του TalkText μέχρι να τελειώσει ο πίνακας.



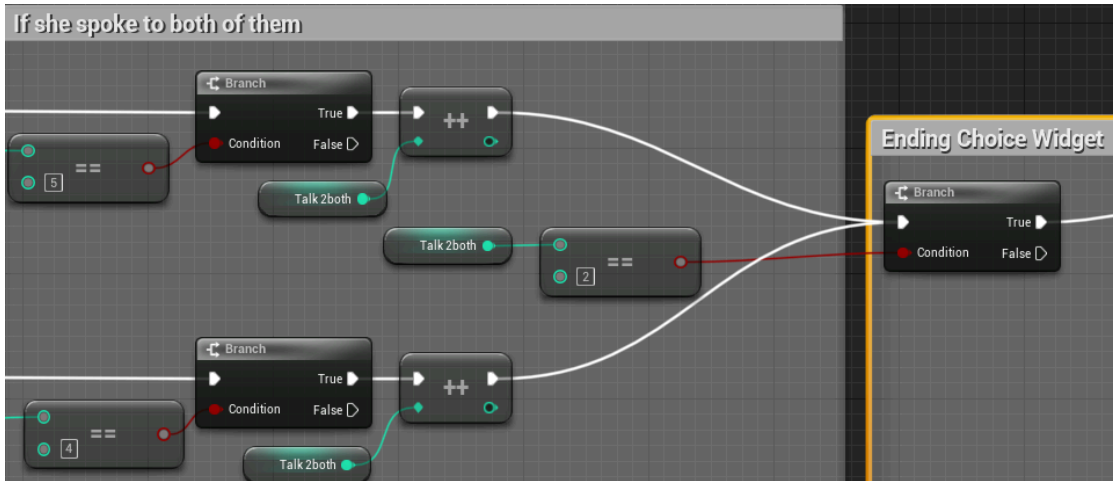
Εικόνα 125. Felipe Event Graph (Text/Audio Dialogue Widget)

Από πλευράς υλοποίησης του παίκτη, για να μπορέσει να αλληλεπιδράσει με τους Npc, χρειάστηκε ένα διάγραμμα ροής στο Blueprint του First Person (βλ. [Εικόνα 126.](#)) για τον καθένα. Παρακάτω φαίνεται αυτό του Felipe που είναι όμοιο με αυτό του Tobias. Αρχικά ορίσαμε μια μεταβλητή μετρητή (TalktoFelIndex) για να παρακολουθούμε τις φορές που μίλησε ο παίκτης με τον Npc. Καθώς το widget Dialogue καλείται στο Blueprint του κάθε Npc ξεχωριστά, το διαφορετικό από την υλοποίηση των άλλων widget (photo, letter, open_folder) είναι ότι απλά καλούμε την συνάρτηση Talk. Αυτή ενεργοποιεί το Event Talk στους Npc.



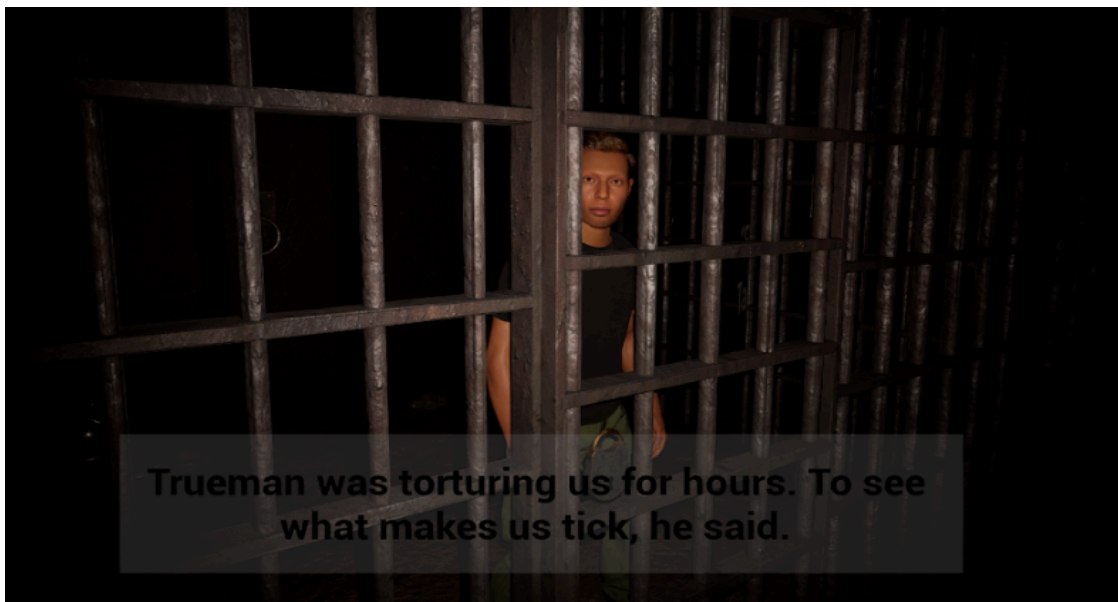
Εικόνα 126. Felipe in First Person Event Graph

Αφού ορίσαμε τα διαγράμματα ροής για τους διαλόγους και των δύο Npcs, δημιουργούμε μια νέα μεταβλητή μετρητή (Talk2both), η οποία όπως λέει και το όνομά της, καταγράφει αν έχει μιλήσει και με τους δυο. Τα διαγράμματα ροής των δύο Npcs συνεχίζονται με έναν έλεγχο το καθένα (βλ. [Εικόνα 127.](#)), για το αν ο παίκτης έχει πάρει όλη την πληροφορία που του προσέφερε ο Npc (Felipe (max 5) και Tobias (max 4)) και αν ισχύει, αυξάνει την Talk2both. Όταν ο παίκτης βρει στα δωμάτια τους Npcs και καταφέρει να μιλήσει και με τους δύο, είναι πια σε θέση να αποφασίσει ποιόν θα σώσει και να ενεργοποιήσει το widget της τελικής επιλογής (EndingChoice).



Εικόνα 127. Felipe/Tobias in First Person Event Graph (Spoke with both check)

Το αποτέλεσμα της παραπάνω υλοποίησης φαίνεται στον παίκτη παρακάτω (βλ. [Εικόνα 128.](#)):



Εικόνα 128. Felipe/Tobias - Dialogue Τελικό αποτέλεσμα

4.4. Cinematics

Το Unreal Engine περιέχει ισχυρά εργαλεία που επιτρέπουν την δημιουργία εντυπωσιακών κινηματογραφικών πλάνων εντός του παιχνιδιού. Μέσω της ενσωμάτωσης τεχνολογιών αιχμής, όπως το Nanite και το Lumen, δημιουργούνται νέες δυνατότητες οπτικής ποιότητας και λεπτομέρειας.

Το κυριότερο από τα εργαλεία που περιλαμβάνονται είναι το Sequence, μέσα στο οποίο επιτρέπεται ο σχεδιασμός και η δημιουργία οποιασδήποτε κινηματογραφικής σκηνής. Ουσιαστικά, αποτελεί μια ακολουθία γεγονότων, δράσεων και οπτικών στοιχείων που συνδυάζονται δυναμικά για τη δημιουργία συναρπαστικών στιγμών αφήγησης. Μέσα στο sequence υπάρχει πληθώρα λειτουργιών και επιλογών για τη δημιουργία πλάνων, όπως θα παρουσιάζονται παρακάτω.

Για την ένωση των sequences και τη διαχείριση της ροής τους, χρησιμοποιείται το Master Sequence. Πρόκειται για ένα sequence υψηλότερου επιπέδου, που μπορεί να περιέχει περισσότερα μικρά ολοκληρωμένα sequences και επιτρέπει την οργάνωση τους και να παρέχει ευελιξία και ευκολία στους δημιουργούς.

Για το δικό μας Project δημιουργήθηκαν συνολικά δεκαπέντε sequences, τα οποία ενώθηκαν σε τρία Master sequences. Το πρώτο αφορούσε την εισαγωγή του παιχνιδιού, και την γνωριμία του παίκτη με τους χαρακτήρες και την ιστορία, ενώ τα άλλα δύο αφορούν τα δύο πιθανά τέλη που μπορεί να καταλήξει ο παίκτης διαλέγοντας να σώσει είτε τον Tobias είτε τον Felipe. Τα cinematics που δημιουργήσαμε αφορούσαν μόνο το level του νησιού.

Sequences

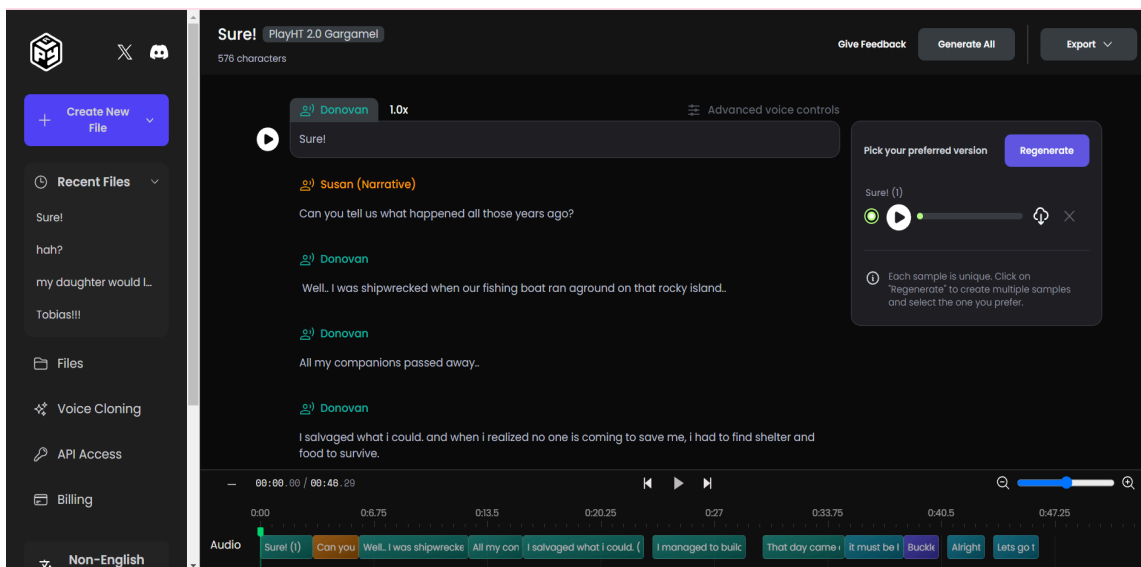
Για την δημιουργία των cinematics του “The Journalist” χρησιμοποιήσαμε τα περισσότερα από τα εργαλεία που προσφέρονται στο Unreal Engine. Επιπλέον, χρησιμοποιήσαμε άλλα εξωτερικά προγράμματα: PlayHT, Mixamo-Mixamo Converter, LiveLink και DaVinci Resolve. Γενικότερα, για τη δημιουργία όλων των cinematics, υλοποιήσαμε τα εξής:

- **Body animations:** για την προσομοίωση της κίνησης του σώματος, Χρησιμοποιήσαμε Sitting, Sitting Talking, Walking, Idle, Run κ.α. Σε διάφορα στυλ.
- **Face Animations:** για τον σχεδιασμό της έκφρασης και της ομιλίας στα πρόσωπα των παικτών. Με βάση τα λόγια του καθενός στο ηχητικό υλικό, το πρόσωπο και τα χείλη συγχρονίζονται.
- **Audio:** προσθήκη ηχητικών για τις συνομιλίες των χαρακτήρων, αλλά και ήχους περιβάλλοντος, όπως ήχοι ελικοπτέρου, κύματος, κελαηδίσματα πουλιών, μουσική κ.α. Επιπλέον, εφαρμόσαμε μερικές ρυθμίσεις, όπως η ένταση, η ταχύτητα, το βάθος κάθε ήχου.
- **Transform:** για τον ορισμό της μετατόπισης και στροφής (location and rotation) κάθε actor που μετακινήθηκε σε κάθε sequence, όπως χαρακτήρες, κάμερες κ.α.
- **Camera Actors:** προσθήκη καμερών και εναλλαγή πλάνων σε κάθε sequence. Ρυθμίστηκε το focus, το blur κ.α. ανάλογα με τις ανάγκες κάθε σκηνής. Η ένωση μετά την εναλλαγή των πλάνων έγινε μέσω Camera Cuts.
- **Light Transform:** Σε κάθε sequence μετατοπίσαμε κατάλληλα τη θέση του ήλιου, την ένταση και τον χρωματισμό, ώστε να κολακεύουν την κάθε σκηνή στο μέγιστο.

Επιπλέον, για να μην επηρεάζουν το περιβάλλον του νησιού όλοι οι actors κάθε cinematic sequence, δηλαδή να ανήκουν μόνο στο εκάστοτε sequence και να εξαφανίζονται όταν αυτό τελειώσει, μετατρέψαμε την κατάσταση τους σε spawnable.

Play.HT

Στη συνέχεια, χρησιμοποιήσαμε το δωρεάν AI Online Tool “Play.HT” (βλ. [Εικόνα 129.](#)), και δώσαμε σε όλους τους Metahumans συγκεκριμένη φωνή. Το Play.HT περιέχει μεγάλη γκάμα επιλογών, όπως φωνές, ταχύτητα, ύφος κ.λπ. Με αυτό το Text-to-Speech εργαλείο, δημιουργήσαμε όλες τις συνομιλίες που χρειαστήκαμε για τα κινηματογραφικά πλάνα, και τις εισάγαμε στο Project μας. Στην παρακάτω εικόνα, παρουσιάζουμε το πρόγραμμα κατά την δημιουργία μιας εκ των συνομιλιών που δημιουργήσαμε.



Εικόνα 129. Play.HT Online AI tool

LiveLink

Το επόμενο πρόγραμμα είναι το LiveLink και χρησιμοποιήθηκε για τη δημιουργία των animations του προσώπου που εφαρμόσαμε στους Metahumans. Αποτελεί ένα ισχυρό εργαλείο που επιτρέπει την καταγραφή της κίνησης του προσώπου και υποστηρίζει τον συγχρονισμό ακόμη και σε πραγματικό χρόνο. Μέχρι στιγμής είναι συμβατό μόνο με λογισμικό iOS και όχι με Android.

Αρχικά, ορίσαμε τη σύνδεση μέσω iPhone και υπολογιστή συνδέοντας τα στο ίδιο δίκτυο, μέσω της IP address. Στη συνέχεια, απαιτείται ένα αρχικό video με περιφερειακή λήψη του προσώπου του χρήστη (calibration), που χρησιμοποιείται για την αναγνώριση του προσώπου και των χαρακτηριστικών (μάτια, μύτη, δόντια) του χρήστη, ώστε αυτά στη συνέχεια να αντιστοιχιστούν στα χαρακτηριστικά προσώπου των Metahumans. Τέλος, βιντεοσκοπούμε όσες ακόμη λήψεις χρειάζεται να μετατραπούν σε facial animations.

Στο Unreal Engine, δημιουργήσαμε τα εξής:

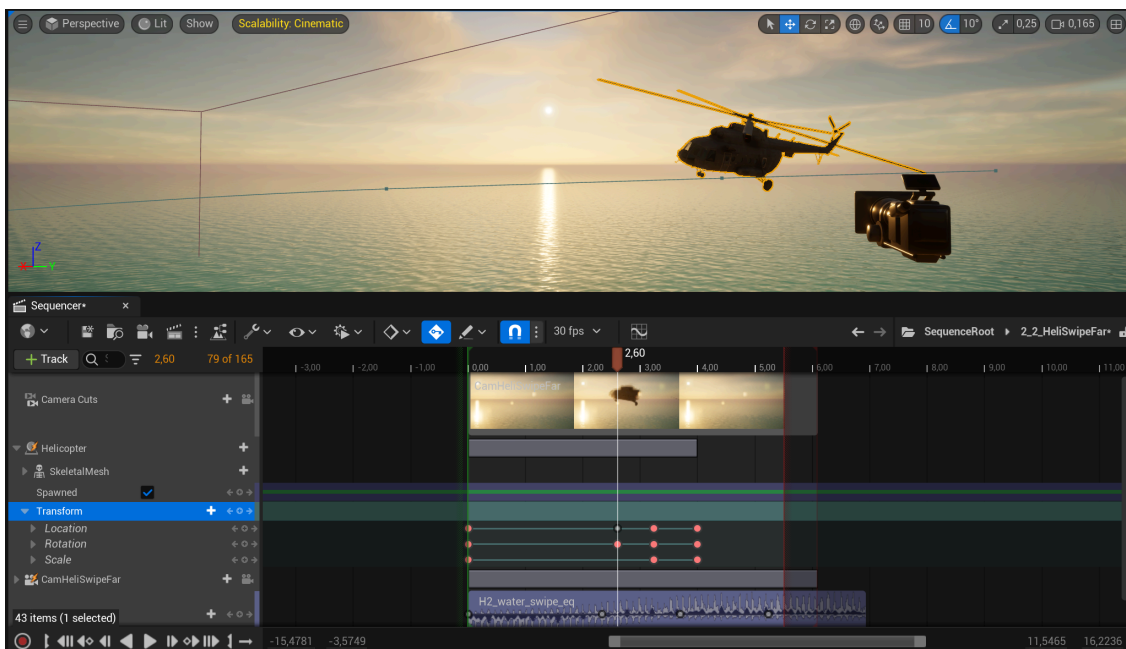
- **Capture Source:** εμφανίζει τα διαθέσιμα video προς μετατροπή από τη συνδεδεμένη συσκευή iPhone
- **Metahuman Identity:** δημιουργεί το πρόσωπο το οποίο απεικονίζεται στο Calibration ως πρόσωπο Metahuman. Εδώ ορίζουμε τις θέσεις των ματιών, χειλιών

Σχεδιασμός σκηνικών

Για τη δημιουργία κάθε sequence, έπρεπε να στήσουμε το σκηνικό για κάθε πλάνο ξεχωριστά. Αρχικά, προσθέσαμε assets τα οποία αποκτήσαμε από το Marketplace και το Sketchfab, μία ακόμη πλατφόρμα με συλλογές από 3D μοντέλα. Τα assets τοποθετήθηκαν ανάλογα με τις ανάγκες κάθε σκηνής.

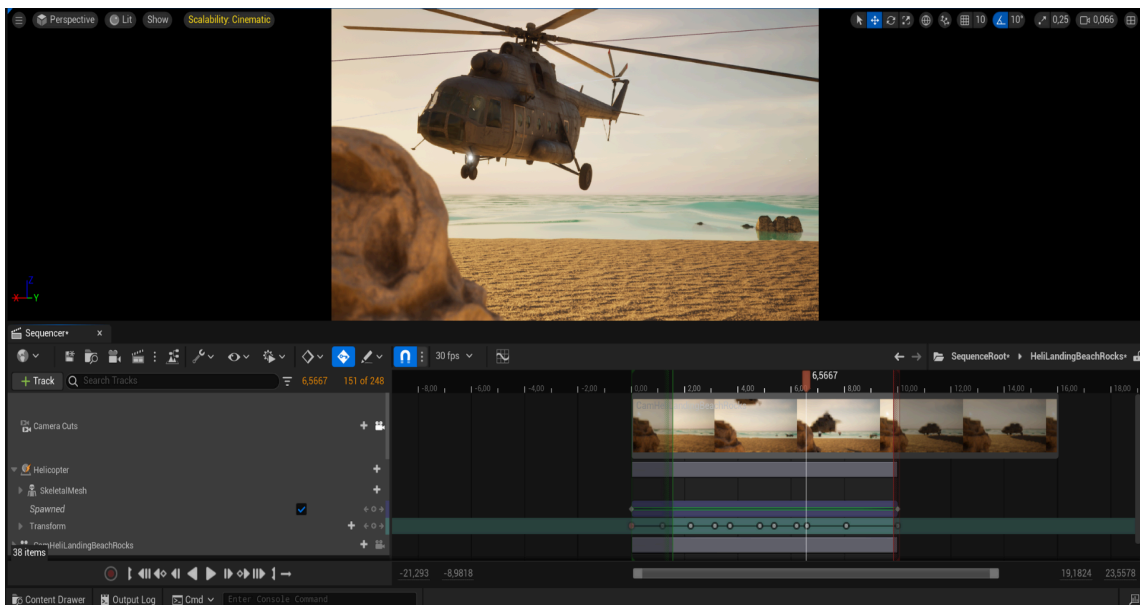
Helicopter Scenes

Τα πρώτα πλάνα που σχεδιάσαμε (βλ. [Εικόνα 132.](#)), ήταν εισαγωγικά και χρησιμοποιήθηκε ένα μοντέλο ελικοπτέρου που αποκτήσαμε από το Marketplace. Για το ελικοπτερο ορίσαμε ρυθμίσεις όπως η ταχύτητα του κεντρικού και πίσω έλικα για την προσομοίωση της κίνησης. Σχεδιάσαμε την πορεία του ελικοπτέρου πάνω από τη θάλασσα, αλλά και άλλες σκηνές όπως η προσγείωση στο νησί, με τη χρήση της επιλογής Transform. Ουσιαστικά, μετακινώντας τη θέση του ελικοπτέρου ανά συγκεκριμένο χρονικό διάστημα και προσθέτοντας keyframes, δημιουργήσαμε μια διαδοχική συνεχή κίνηση.



Εικόνα 132. Helicopter Opening Scene

Επιπλέον, προσαρμόσαμε τις ρυθμίσεις του φωτισμού (Directional Light, Volumetric Cloud) προκειμένου οι σκηνές να διαδραματίζονται τις πρωινές ώρες. Τέλος, προσθέσαμε ήχους ελικοπτέρου τους οποίους επεξεργαστήκαμε στο DaVinci Resolve, προσθέτοντας Equalizer για την αφαίρεση υψηλών συχνοτήτων και αυτοματοποιώντας το Panning, ώστε ο ήχος να έχει βάθος και να μετατοπίζεται βάσει της θέσης του ελικοπτέρου. Τοποθετώντας πολλαπλά camera actors σε διάφορες οπτικές γωνίες και μετακινώντας τες με όμοιο τρόπο, καταφέραμε να δημιουργήσουμε τις πρώτες σκηνές, την ροή των οποίων διευθετήσαμε μέσω των Camera Cuts (βλ. [Εικόνα 133.](#)).



Εικόνα 133. Helicopter Landing

Interview Scenes

Στη συνέχεια, προχωρήσαμε στο σχεδιασμό της βασικής σκηνής που αφορούσε τη συνέντευξη του John Trueman (Ezekiel Asir) στο εσωτερικό του ελικοπτερου (βλ. [Εικόνα 134](#)). Σε αυτό το sequence, παρουσιάζονται οι χαρακτήρες μας καθώς επίσης και η ιστορία του John Trueman πίσω από τα 25 χρόνια μοναξιάς στο έρημο νησί.

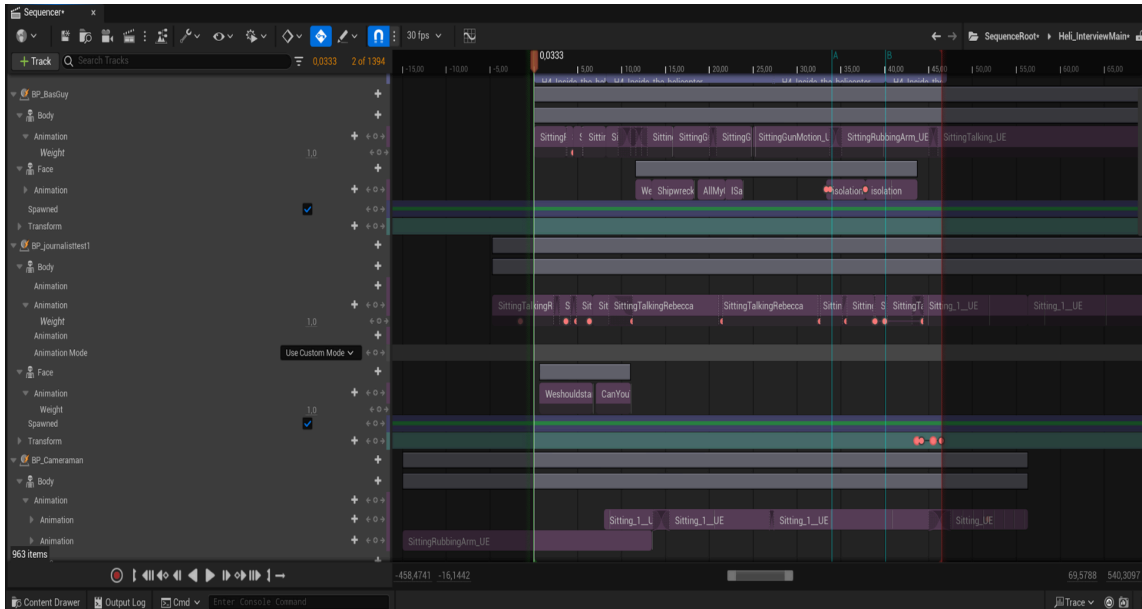
Αρχικά, διακοσμήσαμε το εσωτερικό του ελικοπτερου με διάφορα αντικείμενα (camera lens, tape recorder, exit sign, handbags) που συνέβαλαν στη ρεαλιστικότητα της ιστορίας.



Εικόνα 134. Interview Scene

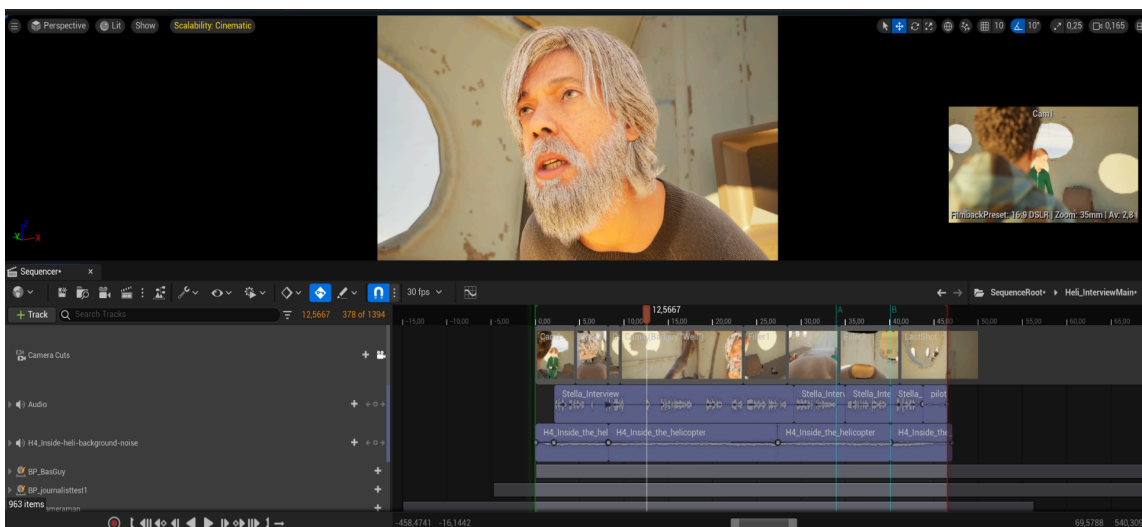
Τοποθετήθηκαν οι χαρακτήρες και το ηχητικό υλικό της συνέντευξης που δημιουργήσαμε στο PlayHT. Σύμφωνα με το audio, φτιάξαμε ένα πλάνο για τις καταστάσεις κίνησης των Metahumans, είτε βρίσκονταν σε αδρανή κατάσταση είτε σε συνομιλία. Έτσι, για κάθε χαρακτήρα προσθέσαμε τα κατάλληλα body animations όπως Idle, Sitting, Sitting Talking κ.α., για τα οποία επεξεργαστήκαμε τη διάρκεια, την

ταχύτητα και το blend με τα επόμενα στη σειρά (βλ. [Εικόνα 135.](#)). Αφού ολοκληρώσαμε τις κινήσεις σώματος, εισάγαμε τα face animations που δημιουργήσαμε στο LiveLink. Ρυθμίσαμε τα Additive Settings στο Panel των Asset Details του κάθε Face animation, ως Additive Anim Type το “Mesh Space” και ως Base Pose Type το “Frame from this animation”, προκειμένου η κίνηση του κεφαλιού να ορίζεται από τα body animations. Μετά την εφαρμογή όλων των animations, πραγματοποιήσαμε τις τελευταίες τροποποιήσεις για τον συγχρονισμό της κίνησης, της έκφρασης, της ομιλίας και του ήχου. Η συγκεκριμένη διαδικασία αποτέλεσε μια από τις προκλήσεις που αντιμετωπίσαμε και αναλύεται παρακάτω.



Εικόνα 135. Interview Scene Editing Process

Επιπλέον, δημιουργήσαμε Camera Actors τα οποία ακολουθούν από διάφορες οπτικές τους χαρακτήρες αλλά και το σκηνικό για τη λήψη της σκηνής. Αυτοματοποιήθηκε η κίνηση τους, το blur και focal length μέσω keyframes, έτσι ώστε να ενισχυθεί η αφηγηματική ένταση και μέσω των camera cuts τοποθετήθηκαν διαδοχικά οι κατάλληλες εναλλαγές των λήψεων. Τέλος, προσθέσαμε έναν ακόμη ήχο ελικοπτερου χαμηλής έντασης, ώστε αποδοθεί φυσικότητα στη σκηνή (βλ. [Εικόνα 136.](#)).

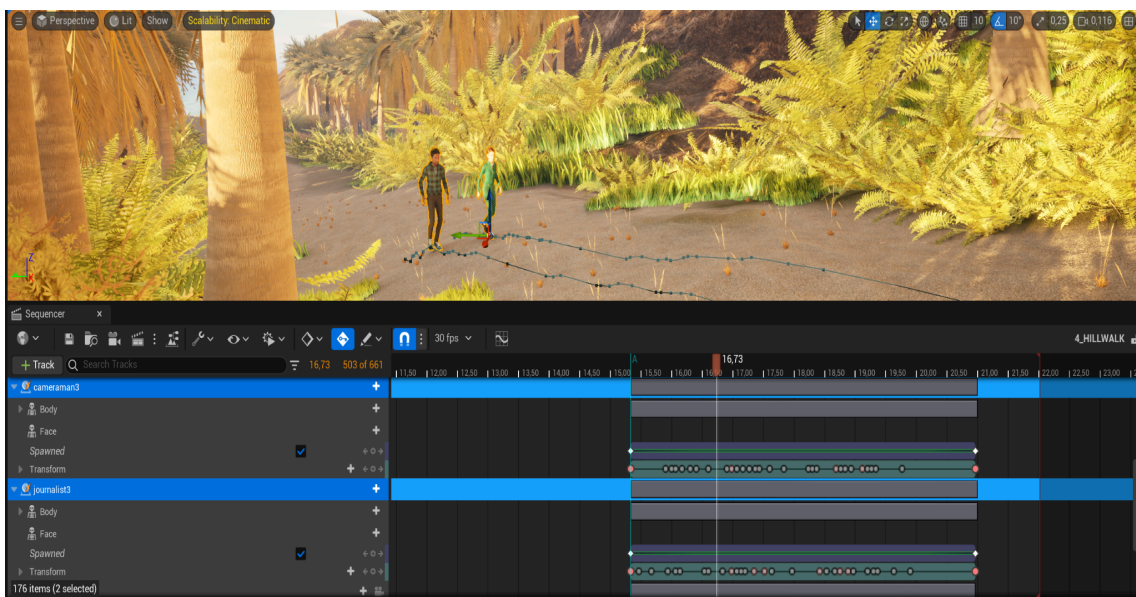


Εικόνα 136. Interview Scene Close up

Talk on the beach & hillwalk scene

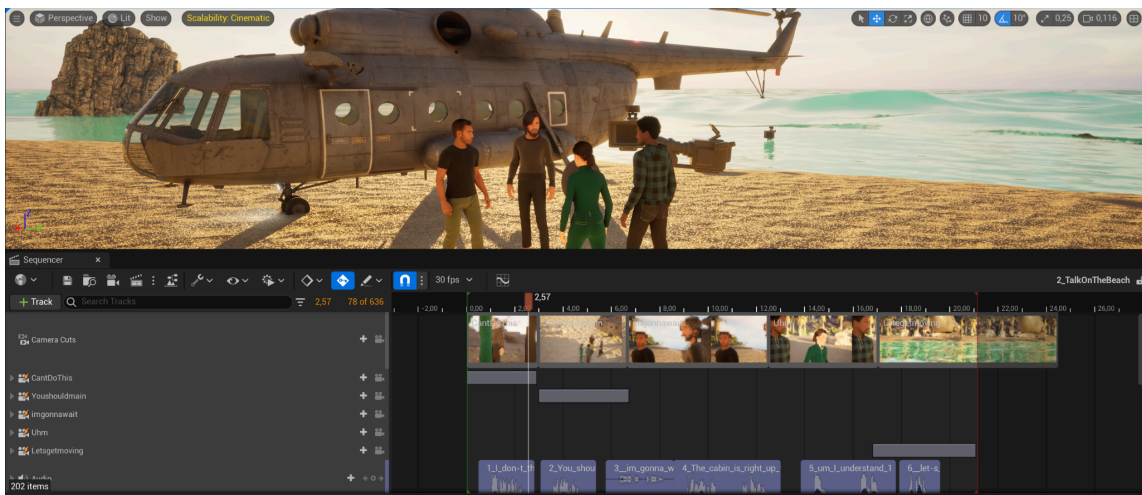
Το επόμενο πλάνο που δημιουργήσαμε αφορούσε τη σκηνή μετά την προσγείωση. Οι χαρακτήρες κατεβαίνουν από το ελικόπτερο και συνομιλούν. Στη συνέχεια η Rebecca με τον Tobias ξεκινούν την πορεία τους προς την καλύβα, αφού ο Ezekiel αποφασίζει να μην ακολουθήσει μαζί τους και να παραμείνει στην παραλία με τον Felipe.

Με αντίστοιχο τρόπο όπως στη σκηνή της συνέντευξης, προσθέσαμε στους χαρακτήρες face και body animations, και τα συνδιάσαμε κατάλληλα, ώστε μετά την προσθήκη του αντίστοιχου ηχητικού υλικού, να συγχρονίζονται ομοιόμορφα κίνηση και ομιλία. Σε αυτό το sequence έπρεπε, επιπλέον, να χρησιμοποιήσουμε και body animation για το περπάτημα των χαρακτήρων. Αφού διαλέξαμε για κάθε έναν ξεχωριστά ένα στυλ βαδίσματος, που όμως ο χαρακτήρας βαδίζει στατικά, έπρεπε να ορίσουμε και τη μετατόπιση του όσο το animation είναι ενεργό. Έτσι, με τη βοήθεια των keyframes του Transform, καταφέραμε οι χαρακτήρες να μετατοπίζονται κατάλληλα στο ρυθμό του animation (βλ. [Εικόνα 137.](#)).



Εικόνα 137. Hill Walk Scene

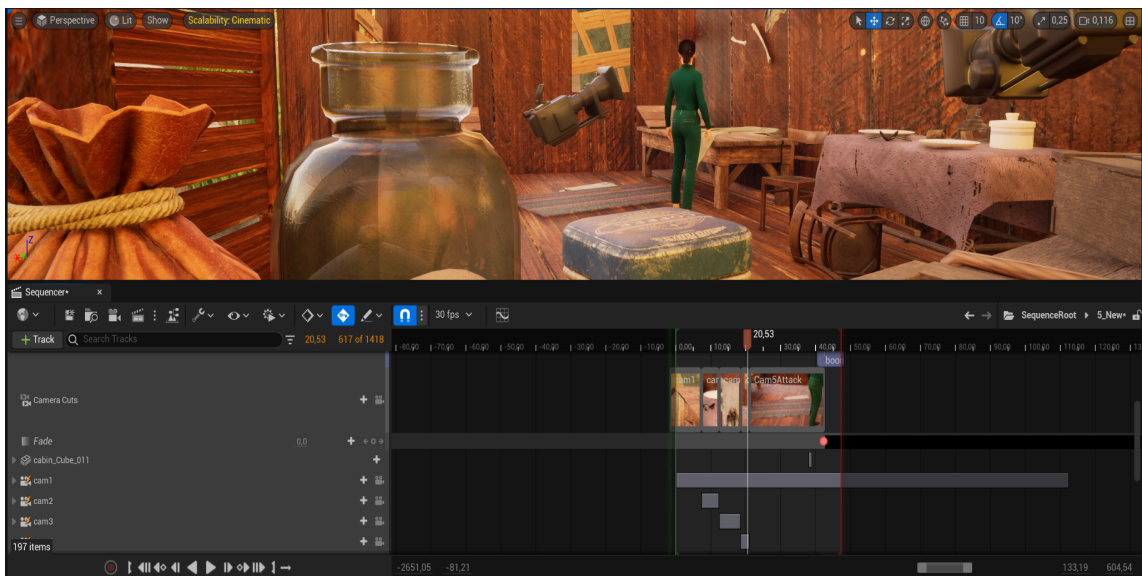
Αφού σχεδιάσαμε όλες τις κινήσεις και τις συγχρονίσαμε με την ομιλία των χαρακτήρων, προσθέσαμε Camera Actors, και πήραμε τα κατάλληλα πλάνα για αυτή τη σκηνή. Έτσι, φτιάξαμε το πλάνο που κατεβαίνουν από το ελικόπτερο, που συνομιλούν στην παραλία (βλ. [Εικόνα 138.](#)), που περπατάνε και συνομιλούν στο δρόμο προς την καλύβα.



Εικόνα 138. Talk On The Beach Scene

Inside The Cabin Scene

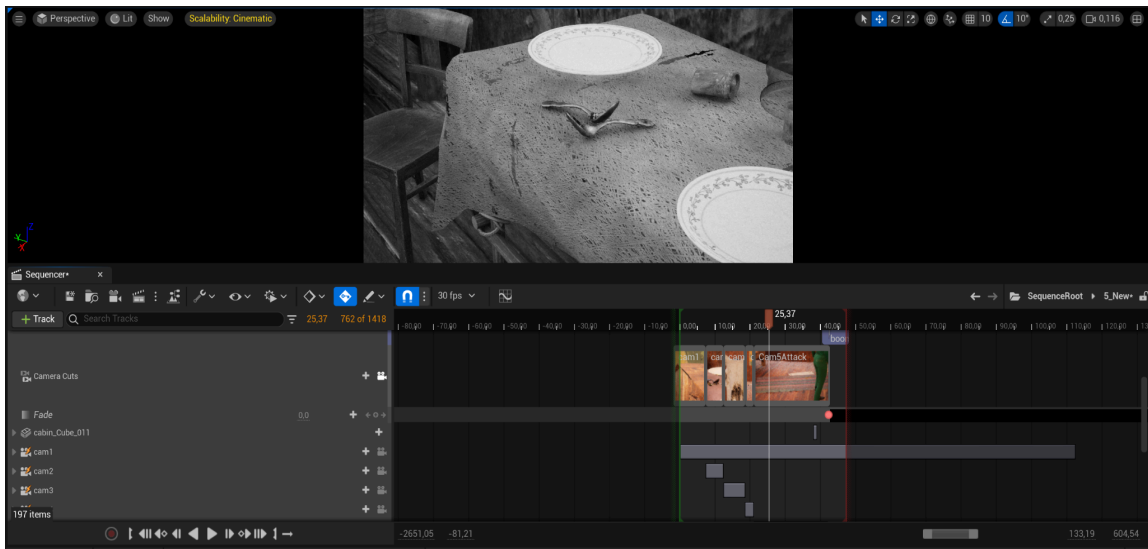
Το τελικό πλάνο του εισαγωγικού κινηματογραφικού video, ήταν αυτό που η Rebecca και ο Tobias καταφθάνουν στην καλύβα. Κατά την είσοδο τους θαυμάζουν τα αντικείμενα που βρίσκονται στο χώρο και συζητούν για την επιβίωση του Ezekiel (βλ. [Εικόνα 139](#)). Ο Tobias ξεκινά να βιντεοσκοπεί με την κάμερα του, η Rebecca βρίσκει κάτω από το χαλί μια σιδερένια καταπακτή και συνειδητοποιούν πως ο Ezekiel δεν ζούσε μόνος του εκεί, όπως ισχυριζόταν.



Εικόνα 139. Inside The Cabin Scene

Σε αυτό το sequence χρησιμοποιήσαμε και πάλι body animations για τους χαρακτήρες, ηχητικό υλικό, και camera actors, όμως μορφοποιήσαμε τον τρόπο λήψης της συγκεκριμένης σκηνής. Χρησιμοποιήσαμε έναν Camera Actor, και αφού ρυθμίσαμε τη λήψη ως ασπρόμαυρη, δημιουργήσαμε ένα εφέ πως το πλάνο αυτό είναι από την κάμερα που κρατάει ο ίδιος ο Tobias. Δημιουργήσαμε κίνηση για την κάμερα, πιο απότομη και έντονη με γρήγορες μετακινήσεις, ώστε να αποδώσουμε ένα πιο φυσικό αποτέλεσμα με περισσότερη αγωνία (βλ. [Εικόνα 140](#)). Στο τέλος της συνομιλίας, προσθέσαμε ένα μικρό ήχο που θυμίζει άνοιγμα πόρτας, προσθέσαμε Attack Animation

στον Ezekiel, και με κατάλληλη μετακίνηση της κάμερας, καταφέραμε η κάμερα να στρέφεται προς την πόρτα την ώρα που επιτίθεται ο Ezekiel. Τελικά, εφαρμόσαμε στην κάμερα ένα Fade out εφέ, και το πλάνο σβήνει απότομα. Μετά από αυτό το πλάνο, η Rebecca έχει μπει μέσα από την καταπακτή στα μπουντρούμια απ'όπου και ξεκινά ο παίκτης το παιχνίδι.



Εικόνα 140. Inside The Cabin Scene (Tobias Camera)

Ending Felipe/ Ending Tobias Scene

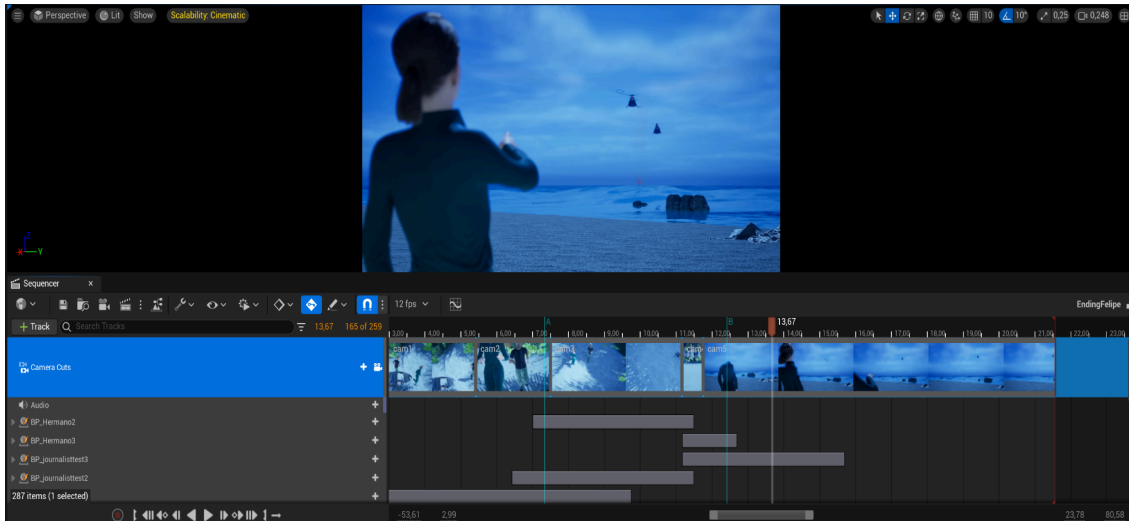
Τα sequences Ending Tobias και Ending Felipe, δημιουργήθηκαν για τα δύο πιθανά τέλη που μπορεί να καταλήξει ο παίκτης. Στο τέλος του παιχνιδιού, ο παίκτης έρχεται αντιμέτωπος με την απόφαση για το ποιον από τους δυο χαρακτήρες θα διαλέξει να σώσει. Ανάλογα με την απάντηση του ενεργοποιείται το αντίστοιχο τελικό video.

Στο sequence του Felipe, μετατρέψαμε τη θέση του ηλίου ώστε να δημιουργήσουμε νυχτερινή ατμόσφαιρα, και τοποθετήσαμε τους χαρακτήρες Rebecca και Felipe στο εσωτερικό της καλύβας. Χρησιμοποιώντας body animations με κίνηση τρεξίματος “Running” και μέσω του Transform (βλ. [Εικόνα 141.](#)), ρυθμίσαμε τη μετατόπιση τους ώστε να φαίνεται πως απομακρύνονται από την καλύβα με κατεύθυνση προς την παραλία.



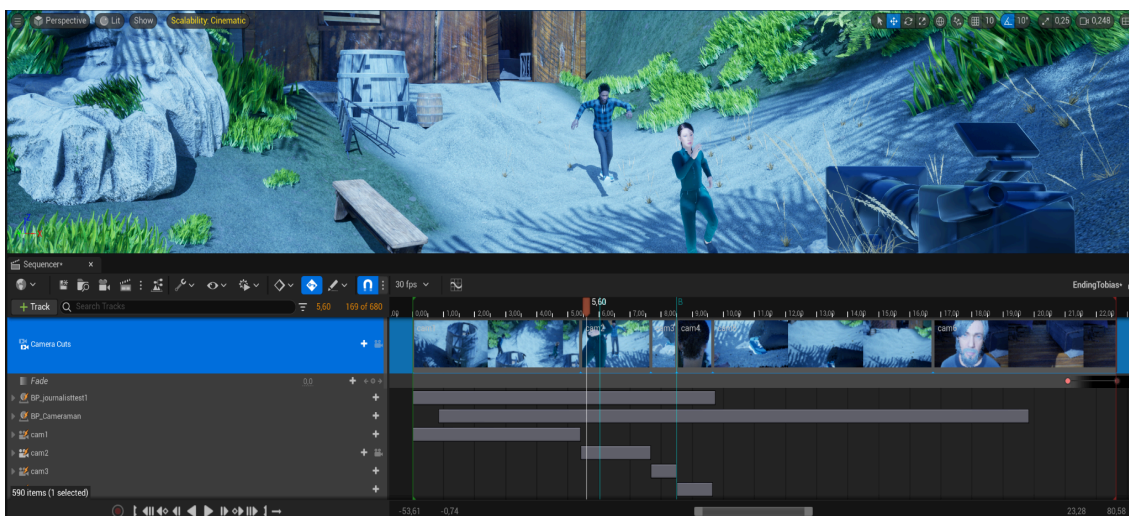
Εικόνα 141. Ending Felipe Scene

Κατά την άφιξη τους στην παραλία, προσθέσαμε body animations για τον τερματισμό του τρεξίματος και στη Rebecca μια ακόμη κίνηση όπου σηκώνει το χέρι και δείχνει προς τη θάλασσα. Προς την κατεύθυνση που δείχνει η Rebecca εισάγαμε δύο μοντέλα ελικοπτέρων τα οποία μετακινούνται προς το νησί με ενεργοποιημένους έλικες (βλ. [Εικόνα 142.](#)). Τέλος, συνοδεύσαμε τη σκηνή suspenseful μουσική και δημιουργήσαμε Camera Actors από διαφορετικές γωνίες για να καταγράψουμε τα κατάλληλα πλάνα. Έτσι, το τελικό αποτέλεσμα, με την συνένωση των πλάνων στα Camera Cuts, αντικατοπτρίζει τους δύο χαρακτήρες να τρέχουν προς την παραλία και να έρχονται αντιμέτωποι με την επικείμενη σωτηρία τους από τον Ezekiel.



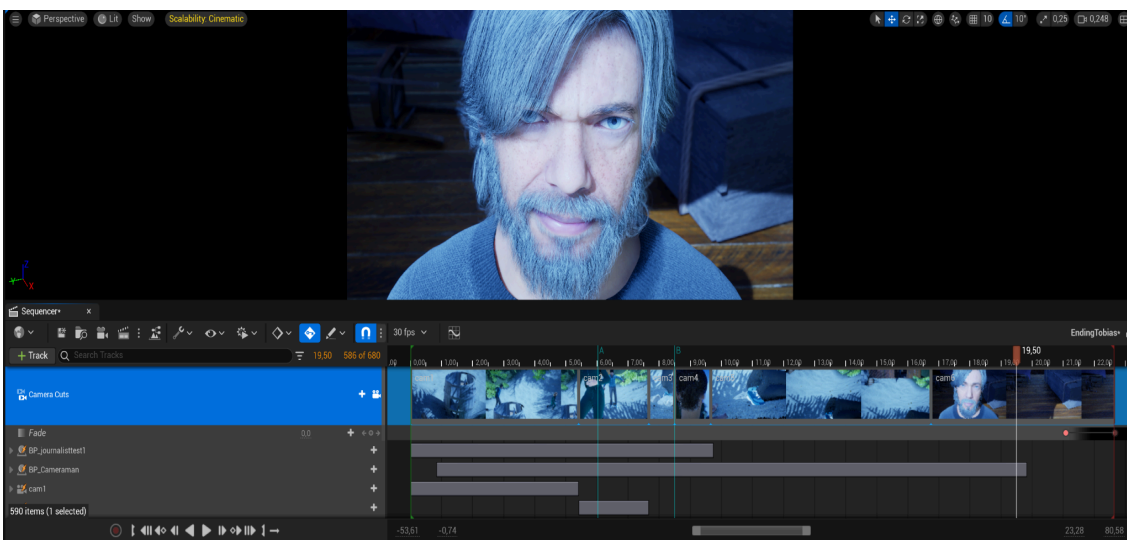
Εικόνα 142. Ending Felipe Scene (Salvation)

Στο sequence του Tobias, ξεκινήσαμε τοποθετώντας τους δύο χαρακτήρες στο εσωτερικό της καλύβας. Αρχικά, επιλέξαμε τα κατάλληλα body animations, δηλαδή για τη Rebecca “Running” και τον Tobias “Running Injured”, και ρυθμίσαμε κατάλληλα το Transform τους ώστε να μετακινούνται ανάλογα με την ταχύτητα των βημάτων τους (βλ. [Εικόνα 143.](#)). Λίγο εκτός της καλύβας, αλλά σε οπτική επαφή με αυτή, αλλάξαμε το body animation του Tobias σε “Falling” δημιουργώντας την ψευδαίσθηση της πτώσης, και το body animation της Rebecca ώστε να σταματάει να τρέχει και να κοιτάει προς τα πίσω τον Tobias πεσμένο στην άμμο.



Εικόνα 143. Ending Tobias Scene

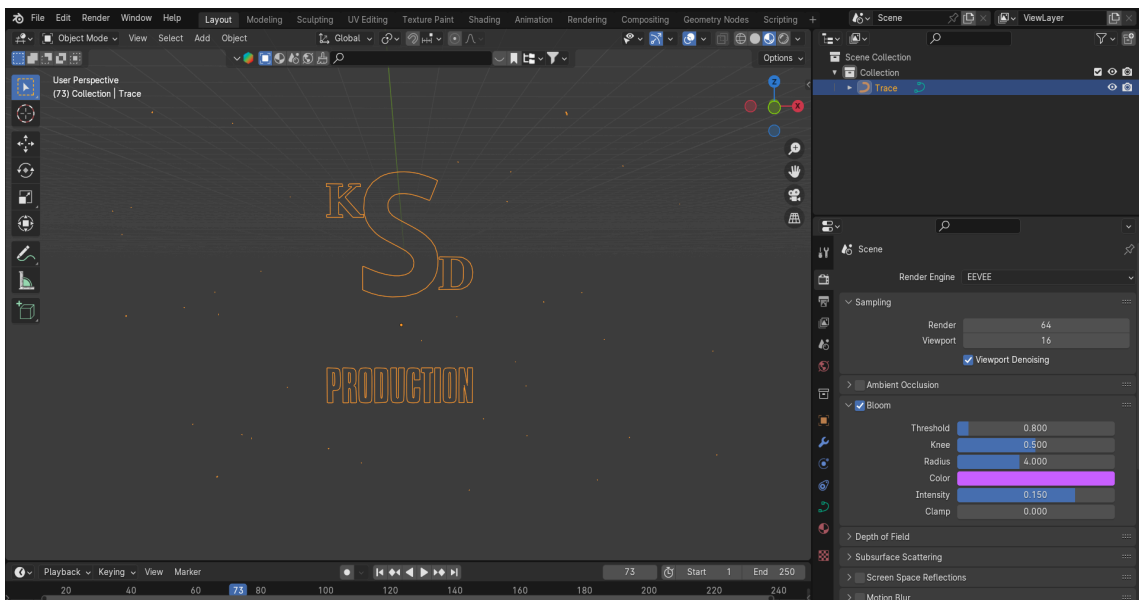
Στη συνέχεια, τοποθετήσαμε τον Ezekiel στο εσωτερικό της καλύβας και του προσθέσαμε body animations, προσδίδοντάς του μερικά βήματα προς την πόρτα της καλύβας, όπου σταματάει και κοιτάζει στην κατεύθυνση του Tobias. Τέλος, προσθέσαμε Camera actors που εναλλάσσονται και ακολουθούν την κίνηση των χαρακτήρων. Στην τελική σκηνή, η κάμερα εστιάζει στον Tobias πεσμένο, ενώ το focus απομακρύνεται πίσω του που ξεπροβάλλει ο Ezekiel στην πόρτα, δίνοντας έμφαση στην ατμόσφαιρα της σκηνής. Το τελευταίο πλάνο εστιάζει στο πρόσωπο του Ezekiel, για τον οποίο μέσω του Control Rig του προσώπου εντοπίσαμε τους κατάλληλους μύες και μετακινώντας τους κατάλληλα, δημιουργήσαμε μια τρομακτική έκφραση με ανατριχιαστικό χαμόγελο (βλ. [Εικόνα 144.](#)). Στο τελικό αποτέλεσμα, οι χαρακτήρες προσπαθούν να δραπετεύσουν, αλλά με την απρόσμενη πτώση του Tobias, ο Ezekiel τους προλαβαίνει. Με ένα σαρκαστικό χαμόγελο, δίνει την εντύπωση ότι η διάσωση είναι φαινομενική και δεν έχουν ελπίδα να ξεφύγουν.



Εικόνα 144. Ending Tobias Scene (Doom)

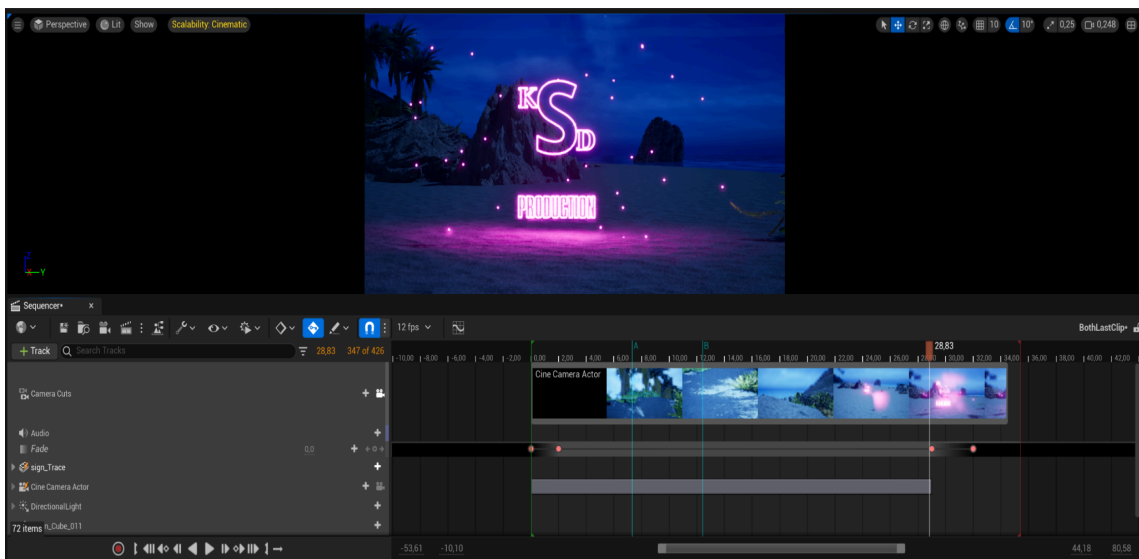
Both Last Clip Scene

Το τελευταίο sequence που δημιουργήσαμε αποτελεί το ομαλά κλείσιμο του παιχνιδιού μετά τα δύο πιθανά τέλη. Αρχικά, δημιουργήσαμε έναν Camera Actor και προσαρμόσαμε την κίνηση του, ώστε να φαίνεται πως βγαίνει μέσα από την καταπακτή που υπάρχει στην καλύβα, ακολουθεί τη διαδρομή μέχρι την παραλία. Μέσω του Blender, δημιουργήσαμε μια πινακίδα neon (βλ. [Εικόνα 145.](#)) την οποία τοποθετήσαμε στην παραλία. Αρχικά δημιουργήσαμε το σχέδιο που θέλαμε να μετατρέψουμε σε Neon, και το μετατρέψαμε σε Trace. Στη συνέχεια, αυξήσαμε το Depth για να δημιουργήσουμε 3D μορφή, διαλέξαμε το χρωματισμό που μας εξυπηρετούσε (Emission) και το εισάγαμε στο Project μας.



Εικόνα 145. Neon Sign Blender

Τοποθετήσαμε την πινακίδα στην παραλία στην κατάλληλη θέση και προσαρμόσαμε κατάλληλα την κάμερα ώστε να σταματάει μπροστά της (βλ. [Εικόνα 146.](#)). Τέλος, προσθέσαμε ένα Fade Out εφέ, προσδίδοντας στην εικόνα αργό σβήσιμο και δημιουργώντας μια σκοτεινή ατμόσφαιρα που αποπνέει μυστήριο.



Εικόνα 146. Neon Sign Closing Scene

Master Sequences-Rendering Rendering (FFMPEG)

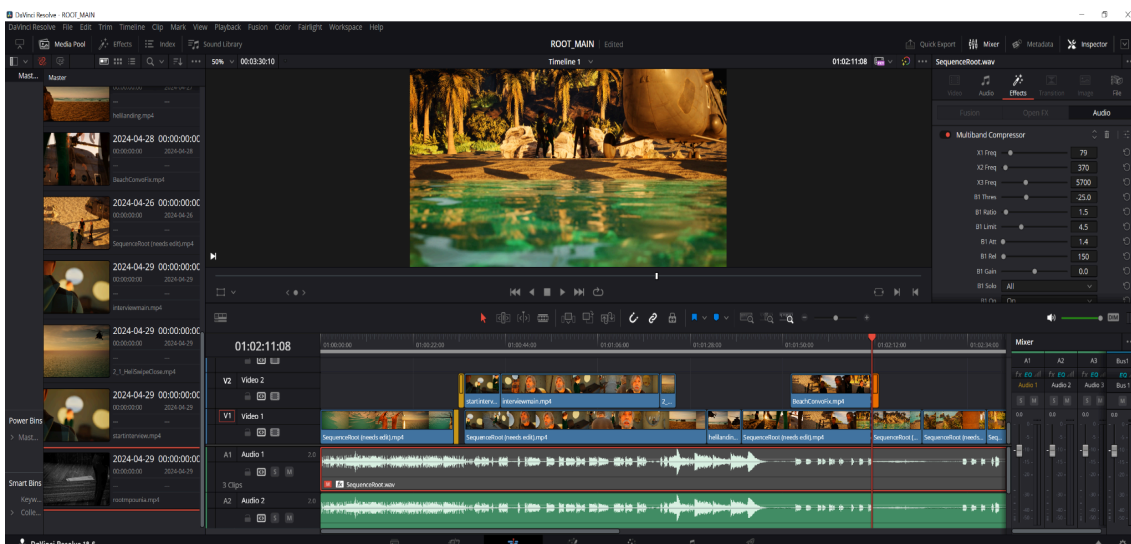
Τα παραπάνω sequences που δημιουργήθηκαν, ενώθηκαν σε τρία Master Sequences κάθε ένα από τα οποία αφορούσε την εισαγωγή, το τέλος αν διασωθεί ο Tobias και το τέλος αν διασωθεί ο Felipe.

Για κάθε Master sequence, προσθέσαμε τα ανάλογα sequences και τα ενώσαμε με τη σειρά μέσω των Camera Cuts, τροποποιώντας το χρόνο προβολής του καθενός και τις εναλλαγές των λήψεων ώστε να συμβαδίζουν με το ηχητικό υλικό. Στη συνέχεια, προχωρήσαμε στο Render των κινηματογραφικών. Χρησιμοποιήσαμε μορφή .exr για

την εξαγωγή των videos ως αλληλουχία εικόνων με τη μικρότερη δυνατή απώλεια ποιότητας. Επιπλέον, προσθέσαμε Anti-aliasing με Spatial και Temporal Sample Count, ένα και δεκαέξι αντίστοιχα, για περισσότερη ευκρίνεια και ομαλότητα. Τέλος, προσαρμόσαμε την ανάλυση σε 2560x1440p και 30fps, επιτυγχάνοντας υψηλής ποιότητας render σε εύλογο χρονικό διάστημα επεξεργασίας. Από προεπιλογή στο Unreal Engine δεν υπάρχει η δυνατότητα το αρχείο εξόδου να είναι video (π.χ. mp4, avi), καθώς μπορεί να παράγει μόνο αλληλουχίες εικόνων (π.χ. Png, jpg) που απαιτούν περαιτέρω επεξεργασία για μετατροπή σε video. Για την άμεση έξοδο σε αρχείο mp4 εντός του Unreal Engine, χρησιμοποιήσαμε το FFmpeg, έναν Command Line Encoder. Κάθε master sequence έγινε render πολλαπλές φορές, λόγω αποκλίσεων του τελικού αποτελέσματος από στοιχεία που προσομοιώνουν τη φυσική (σκιές, κίνηση νερού) και κινήσεων των Metahumans (βλ. [Προκλήσεις](#)). Η εξαγωγή του ηχητικού υλικού έγινε ξεχωριστά, σε μορφή .wav, ώστε να έχουμε περισσότερη ευελιξία κατά την τελική επεξεργασία.

Στο τελευταίο στάδιο, προσθέσαμε τα παραγόμενα αρχεία ήχου και video στο DaVinci Resolve, ένα πρόγραμμα παραγωγής και επεξεργασίας video, που περιλαμβάνει μεγάλο πλήθος δυνατοτήτων και εργαλείων. Η επεξεργασία περιλάμβανε διαλογή των καλύτερων πλάνων μέσα από τις διαφορετικές εκδόσεις των videos και τη σύνθεσή τους σε συγχρονισμό με τον ήχο (βλ. [Εικόνα 147.](#)). Επιπλέον, χρησιμοποιήθηκαν effects, όπως overlay για το πλαίσιο της κάμερας και green screen effect για διάβρωση της εικόνας στα πλάνα μέσα από την κάμερα του Tobias.

Τέλος, ακολούθησε render για κάθε Master Sequence με τις ίδιες ρυθμίσεις ως προς την ανάλυση και το ρυθμό ανανέωσης καρτέ, τα οποία αποτέλεσαν τα τελικά κινηματογραφικά videos που χρησιμοποιήθηκαν στο παιχνίδι.



Εικόνα 147. DaVinci Resolve Editing for Rendering

5. Προκλήσεις-Μελλοντικές Βελτιστοποιήσεις

Στην περίπλοκη διαδικασία ανάπτυξης παιχνιδιών, οι προκλήσεις και τα ζητήματα δημιουργούνται αναπόφευκτα και αποτελούν εμπόδια προς την ομαλή υλοποίηση μιας δημιουργικής διαδικασίας. Σε αυτό το κεφάλαιο παρουσιάζονται οι βασικές προκλήσεις που παρουσιάστηκαν κατά τη διάρκεια της ανάπτυξης του “The Journalist”.

Προκλήσεις

Παρακάτω αναλύονται οι λεπτομέρειες και η πολυπλοκότητα των ζητημάτων, καθώς και οι λύσεις για την αντιμετώπιση τους. Από το τεράστιο μέγεθος του αρχείου μέχρι τις τεχνικές δυσλειτουργίες, κάθε πρόκληση δημιούργησε μοναδικές ευκαιρίες για ανάλυση και εντοπισμό των προβλημάτων, καθώς και στρατηγική προσέγγιση για την επίλυσή τους.

Χρόνος

Σε κάθε έργο υλοποίησης ενός παιχνιδιού με χρονοδιάγραμμα, η αποτελεσματική διαχείριση του χρόνου είναι μεγάλης σημασίας, λόγω της πολυπλοκότητας της ανάπτυξης παιχνιδιών και των διαφόρων εργασιών που απαιτούνται. Μια σημαντική πρόκληση είναι η αποτελεσματική κατανομή του χρόνου σε διάφορες πτυχές του έργου, όπως ο σχεδιασμός της ιστορίας, η σχεδίαση, η ανάπτυξη, η δοκιμή και η επανάληψη.

Απρόβλεπτα εμπόδια και τεχνικά ζητήματα που προέκυψαν κατά τη διάρκεια της ανάπτυξης, σπατάλησαν πολύτιμο χρόνο και δυνητικά εκτροχίασαν σε σημαντικό βαθμό το χρονοδιάγραμμα του έργου. Όμως, η ικανότητα λήψης αποφάσεων υπό πίεση και η συνεχής αναζήτηση εναλλακτικών λύσεων, όπως θα δούμε πιο αναλυτικά στις παρακάτω προκλήσεις, μας επέτρεψαν να μείνουμε εντός των χρονικών περιορισμών του έργου.

Ο συντονισμός των ξεχωριστών προσπαθειών δύο ατόμων με σκοπό την ολοκλήρωση του έργου αποτέλεσε επίσης πρόκληση. Η διασφάλιση της αποτελεσματικής επικοινωνίας, της συνεργασίας και του καταμερισμού των καθηκόντων μεταξύ των μελών της ομάδας, ήταν ζωτικής σημασίας για την τήρηση της προθεσμίας και την παράδοση ενός συνεκτικού τελικού προϊόντος.

Μέγεθος Project

Ένα αξιοσημείωτο εμπόδιο που καταλήξαμε κατά τη διαδικασία ανάπτυξης του παιχνιδιού, ήταν το μέγεθος του Project, το οποίο κατά την τελική έκδοση έφτασε τα 69,8GB. Αυτό δημιούργησε αρκετά ζητήματα, όπως περιορισμό στην αποθήκευση, δυσκολία στη μεταφορά του αρχείου, καθώς και χαμηλότερη απόδοση. Υπάρχουν αρκετές τεχνικές βελτιστοποίησης που μπορεί να ακολουθήσει κανείς για τη μείωση του μεγέθους ενός Project στο Unreal Engine. Κάποιες από αυτές τις τεχνικές είναι Texture και Audio Compression ώστε να μειωθεί το μέγεθος με ελάχιστη απώλεια ποιότητας, Asset streaming ώστε τα assets να φορτώνονται δυναμικά κατά τη διάρκεια του Gameplay ή LOD (Level of Detail) Models ώστε τα assets να εμφανίζονται σε lower-polygon ανάλυση όταν βρίσκονται σε απόσταση, μειώνοντας τη χρήση της μνήμης. Για τους σκοπούς του δικού μας παιχνιδιού, χρησιμοποιήσαμε 3D μοντέλα υψηλής ποιότητας, 4K textures στο περιβάλλον και Metahumans στην υψηλότερη

δυνατή ανάλυση. Για τη μείωση του μεγέθους του Project ακολουθήσαμε μια άλλη στρατηγική, το Migration. Με τη διαδικασία αυτή, το Project σαρώνεται και διαγράφονται όλα τα assets, textures και materials που δεν έχουν χρησιμοποιηθεί. Με τον τρόπο αυτό, το μέγεθος του παιχνιδιού μειώθηκε στα 30GB, κρατώντας όλες τις κατασκευές στην υψηλότερη δυνατή ποιότητα, που ήταν και ο σκοπός του Demo.

Cinematics Live Rendering

Στην διαδικασία δημιουργίας των κινηματογραφικών του παιχνιδιού, προέκυψε μια ακόμη σημαντική πρόκληση κατά την προσπάθεια του Live Rendering των σκηνών που περιείχαν Metahumans. Σε αρκετές από αυτές τις σκηνές προέκυψαν ζητήματα με δυσλειτουργίες των Metahumans, ασύνδετες κινήσεις του κεφαλιού ή αιωρούμενα μαλλιά. Παρά τις προσπάθειες για επιδιόρθωση μέσω την αλλαγής της ανάλυσής τους από High σε Medium, και μερικών αλλαγών στις ρυθμίσεις των Face Animations, δεν εντοπίστηκε η ακριβής λύση του προβλήματος. Έτσι, αντί για την ζωντανή μετάδοση των cinematics κατά τη διάρκεια του παιχνιδιού, ακολουθήσαμε μια διαφορετική προσέγγιση. Χρησιμοποιήθηκε η δυνατότητα του Unreal Engine για Render των cinematics και η εξαγωγή τους σε αρχείο video. Στο Rendering εφαρμόστηκαν ρυθμίσεις όπως το επιθυμητό resolution (2560x1440), η μορφή του αρχείου (mp4) και το πλήθος των frames (30fps) που επεξεργάζεται το Unreal Engine ανά πλάνο. Με αυτό τον τρόπο, η αργή ενδεδειγμένη επεξεργασία που εφάρμοσε το Unreal Engine οδήγησε στη δημιουργία υψηλής ποιότητας κινηματογραφικών χωρίς ζητήματα με τους Metahumans. Στη συνέχεια, επεξεργαστήκαμε τα cinematics στο DaVinci Resolve και τα ενσωματώσαμε εκ νέου στο Project ως αρχεία MP4, παρακάμπτοντας με επιτυχία τις προκλήσεις που παρουσιάστηκαν σε σχέση με το Live Rendering.

Επικοινωνία Levels - Widgets

Η επικοινωνία μεταξύ levels και widgets στο Unreal Engine μπορεί να δημιουργήσει αρκετές προκλήσεις. Ένα πρωταρχικό πρόβλημα που προέκυψε ήταν από την ανάγκη χρήσης στοιχείων μεταξύ διαφορετικών levels. Κατά τη μετάβαση μεταξύ levels, τα widgets δεν ανταποκρίνονταν, με αποτέλεσμα να πρέπει να βρεθεί άλλος τρόπος επικοινωνίας των δύο. Η εμφάνιση των πληροφοριών που αφορούσαν το level και απαιτούνταν για την επιτυχή διασύνδεση, δεν κατέστη ποτέ εφικτή. Αυτές οι πληροφορίες αφορούσαν τα cinematics, τα οποία, ενώ βρίσκονταν στο level του νησιού, έπρεπε με κάποιο τρόπο να κληθούν σε widget στο κεντρικό level του παιχνιδιού στην προσέγγιση του Live Rendering. Το ζήτημα αυτό επιλύθηκε με τον ίδιο τρόπο που επιλύθηκε το παραπάνω πρόβλημα με τους Metahumans. Από τη στιγμή που μετατρέψαμε τα cinematics σε αρχείο Video, καταφέραμε να τα καλέσουμε απευθείας μέσα στα widgets με την απλή χρήση ενός Media Player (βλ. [Τέλος Felipe/Tobias Widget](#)).

Μελλοντικές Βελτιώσεις

Σε αυτόν τον δυναμικό και διαρκώς μεταβαλλόμενο τομέα, η αναζήτηση για βελτίωση είναι διαρκής με κύριο σκοπό την ανταπόκριση στις ολοένα εξελισσόμενες προσδοκίες των παικτών. Οραματιζόμενοι μελλοντικές βελτιώσεις, στοχεύουμε όχι μόνο να βελτιώσουμε την υλοποίηση του παιχνιδιού μας, αλλά και να εμβαθύνουμε σε αυτή, με πιθανές βελτιώσεις πιο συγκεκριμένα στους μηχανισμούς του παιχνιδιού και το σχεδιασμό της διεπαφής χρήστη (UI) έως και τη βελτιστοποίηση των επιδόσεων και την επέκταση του περιεχομένου.

Πιο συγκεκριμένα, μια πολλά υποσχόμενη οδός, για την ενίσχυση της εμπύθισης και της αλληλεπίδρασης των παικτών στο παιχνίδι μας, περιλαμβάνει την ενσωμάτωση έξυπνων NPCs εξοπλισμένων με προηγμένες δυνατότητες τεχνητής νοημοσύνης (AI). Αξιοποιώντας αλγορίθμους δέντρων αποφάσεων (Decision Tree Algorithms) και άλλες τεχνικές τεχνητής νοημοσύνης, αυτοί οι NPC μπορούν να δείχνουν πιο αληθοφανείς συμπεριφορές, να ανταποκρίνονται δυναμικά στις ενέργειες του παίκτη και να συμβάλλουν σε έναν πιο πλούσιο και πιο καθηλωτικό κόσμο παιχνιδιού.

Όσον αφορά τις μελλοντικές βελτιώσεις των widgets, σε αυτές περιλαμβάνονται η βελτιστοποίηση της διάταξης και του οπτικού σχεδιασμού, οι οποίες εκτείνονται μέχρι και τη βελτίωση της λειτουργικότητας και της διαδραστικότητας. Με την ενσωμάτωση μελλοντικών κριτικών από δοκιμές παικτών, μπορούμε να εντοπίσουμε σημεία του παιχνιδιού που χρήζουν βελτίωση, και παράλληλα, βελτιώνοντας τα υπάρχοντα widgets, να μπορέσουμε να ανταποκριθούμε καλύτερα στις προτιμήσεις και στις προσδοκίες των παικτών. Επιπλέον, στόχος μας είναι να διερευνήσουμε νέα χαρακτηριστικά και λειτουργίες που αξιοποιούν τις δυνατότητες των σύγχρονων τεχνολογιών UI, όπως η δυναμική παραγωγή περιεχομένου (dynamic content generation), τα προσαρμόσιμα widgets και η απρόσκοπτη ενσωμάτωση με τα συστήματα παιχνιδιού.

Διατυπώνοντας το όραμά μας για μελλοντικές βελτιώσεις, στοχεύουμε να εμπνεύσουμε περαιτέρω διερεύνηση, συνεργασία και πειραματισμό στην κοινότητα των παιχνιδιών, προωθώντας την πρόοδο και την καινοτομία σε αυτόν τον ζωντανό και δυναμικό κλάδο.

6. Συμπεράσματα

Στο κλείσιμο αυτής της εργασίας, έχοντας περιηγηθεί ανάμεσα στο σχεδιασμό και την υλοποίηση αυτού του παιχνιδιού περιπέτειας, ξεκίνησε ένα ταξίδι που χαρακτηρίζεται από δημιουργικότητα, εφευρετικότητα αλλά, επίσης, και τεχνική ικανότητα. Από την αρχική ιδέα του παιχνιδιού μας έως τον σχολαστικό σχεδιασμό των μηχανισμών του αλλά και την αυστηρή υλοποίηση, η προσπάθειά μας ωθήθηκε από το πάθος για μια διαδραστική αφήγηση και τη δέσμευση να προσφέρουμε, μια όσο το δυνατόν καλύτερη, καθηλωτική εμπειρία παιχνιδιού.

Μέσα από την διαδικασία εκπόνησης της ανάπτυξης του παιχνιδιού μας, αντιμετωπίσαμε προκλήσεις και θριάμβους, με γέλιο και με κλάμα, καθεμία από τις οποίες συνέβαλε στην εξέλιξη του έργου μας και στην ανάπτυξη των δεξιοτήτων μας ως προγραμματιστές παιχνιδιών. Από την ενασχόληση με τις προκλήσεις σε επίπεδο σχεδιασμού, μέχρι την τελειοποίηση των αλληλεπιδράσεων του παίκτη, κάθε εμπόδιο αντιμετωπίστηκε με επιμονή και αποφασιστικότητα, διαμορφώνοντας το τελικό αποτέλεσμα του παιχνιδιού μας.

Καθώς ολοκληρώνουμε αυτό το έγγραφο, πρέπει να τονιστεί η δυνατότητα των παιχνιδιών να εμπνέουν, να ψυχαγωγούν και να προκαλούν σκέψη, όχι μόνο στους παίκτες αυτών αλλά και στους δημιουργούς τους. Μέσα από αυτό το ταξίδι σχεδιασμού και υλοποίησης του παιχνιδιού, διευρύνουμε τις γνώσεις μας σε σχέση με τον σχεδιασμό ηλεκτρονικών παιχνιδιών και εξερευνήθηκε σε βάθος ένα από τα πιο δυνατά σύγχρονα εργαλεία δημιουργίας παιχνιδιών, το Unreal Engine.

7. Βιβλιογραφία

- Edward Ju and Christian Wagner. (1997). *Personal computer adventure games: their structure, principles, and applicability for training*. SIGMIS Database 28, 2 (Spring 1997), 78–92. <https://doi.org/10.1145/264701.264707>
- Fernandez-Vara, C., & Osterwil, S. (2010). *The key to adventure game design: Insight and sense-making*. <http://hdl.handle.net/1721.1/100238>
- Valentine, K. D. & Jensen, L. J. (Eds.). (2016). *Examining the Evolution of Gaming and Its Impact on Social, Cultural, and Political Perspectives*. IGI Global. <https://doi.org/10.4018/978-1-5225-0261-6>
- Thomas Malone. 1981. *What makes computer games fun?* (abstract only). SIGSOC Bull. 13, 2–3 (1982), 258-277. <https://doi.org/10.1145/1015579.810990>
- Qaffas, A. A. (2020). *An Operational Study of Video Games' Genres*. International Journal of Interactive Mobile Technologies (IJIM), 14(15), pp. 175–194. <https://doi.org/10.3991/ijim.v14i15.16691>
- Chad Phoenix Rose Gowler and Ioanna Iacovides. 2019. "Horror, guilt and shame" -- Uncomfortable Experiences in Digital Games. In Proceedings of the Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '19). Association for Computing Machinery, New York, NY, USA, 325–337. <https://doi.org/10.1145/3311350.3347179>
- Prohászková, V. (2012). *The genre of horror*. American International Journal of Contemporary Research, 2(4), 132-142.
- Ntokos, Konstantinos. "“Level of fear” : Analysis of fear spectrum into a tool to support horror game design for immersion and fear" Computer Game Development and Education: An International Journal (CGDEIJ) 1 , no. 1 (2018): 33-43.
- Soares de Lima, Edirlei & Demarque, Thainá. (2013). *Auditory Hallucination: Audiological Perspective for Horror Games*.
- S. Graja, P. Lopes and G. Chanel, "Impact of Visual and Sound Orchestration on Physiological Arousal and Tension in a Horror Game," in IEEE Transactions on Games, vol. 13, no. 3, pp. 287-299, Sept. 2021, [doi: 10.1109/TG.2020.3006053](https://doi.org/10.1109/TG.2020.3006053)
- Krzywinska, T. (2002). *Hands-on horror*. Spectator, 22(2), 12-23.
- Lin, J. H. (2013). Identification matters: *A moderated mediation model of media interactivity, character identification, and video game violence on aggression*. Journal of Communication, 63(4), 682-702.
- Unreal Engine Documentation Forum, <https://dev.epicgames.com/documentation/en-us/unreal-engine/unreal-engine-5-4-documentation>
- Epic Games Development Documentation, <https://dev.epicgames.com/documentation/>