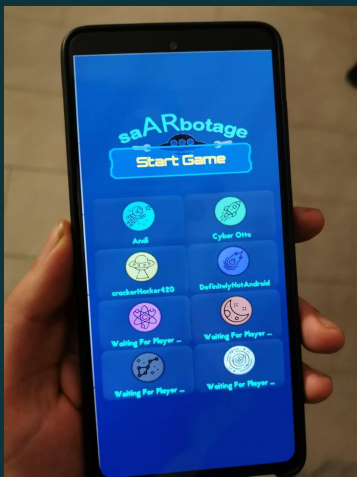


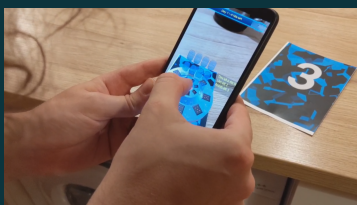
SaARbotage

Konzeptbeschreibung

SaARbotage ist ein interaktives Augmented Reality Mobile Spiel mit klassischem Werwolf Charakter. Eine Gruppe möchte gemeinsam von einer kaputten Raumstation auf einen neuen Planeten fliehen. Der Weg ist weit, der Sauerstoff ist knapp, die Zeit rennt ihnen davon. Sie müssen zusammenhalten, um das Raumschiff rechtzeitig an ihr Ziel zu bringen. Allerdings gibt es unter ihnen Verräter - Androide. Diese wollen das Ankommen der Menschheit auf dem neuen Planeten unbedingt verhindern. Geschicktes untertauchen, Lügen und saarbortieren ist zwingend notwendig, um unentdeckt Teil der Crew zu bleiben. Denn am ende jedes Tages findet sich die Crew zusammen und ein verdächtiger wird von Bord geworfen. Aber war es wirklich ein Android?



SaARbotage ist ein herausforderndes AR echtzeit Abenteuerspiel für 4 - 8 Personen. Das Spiel wird zusammen an einem beliebigen Ort gespielt. Jeder Spieler benötigt die SaARbotage Anwendung auf seinem Smartphone. Zu Beginn des Spieles werden über mehrere physikalische Räume ausgedruckte Spielkarten verteilt. Alle interaktionen mit dem Raumschiff (der App) finden über das Scannen der Spielkarten statt. Interaktionen mit den Crewmitglieder finden jedoch in Angesicht zu Angesicht statt.



Die Reise zum neuen Planeten dauert mehrere in-game Tage. Täglich müssen Maschinen repariert werden, dazu werden die Maschinen an den jeweiligen Stationen gescannt und ein Minispiel startet. Verschiedene Spiele verlangen eine unterschiedliche Anzahl an Spielern. Androiden und Menschen müssen im direkten Kontakt miteinander Aufgaben bewältigen, ohne zu wissen wer von ihnen auf welcher Seite steht. Androiden versuchen das Leben der Crewmitglieder zu erschweren, indem sie Stationen manipulieren. diese verbrauchen dann mehr Sauerstoff und verkürzen das

Leben der menschlichen Crewmitglieder. Dabei müssen sie geschickt vorgehen, um unentdeckt zu bleiben oder sich elegant aus brenzligen Situationen raus reden können. Denn einmal täglich kommt die gesamte Crew zusammen. Sie diskutieren den Verlauf des Tages und Verdächtiges Verhalten. Schaffen es die Androiden, böses Blut unter die Crewmitglieder zu verbreiten? Oder werden sie direkt auffliegen und von Bord geworfen?



Technischer Aufbau

Das Spiel ist mit Unity 3D entwickelt.

Um das Scannen der Spielkarten zu ermöglichen, haben wir das Unity Plugin [Vuforia](#) benutzt.

Multiplayer

Keiner von uns beiden hatten bisher Erfahrung mit der Entwicklung von Multiplayer spielen oder den verschiedenen Möglichkeiten diesen in Unity einzubauen. Wir hatten uns dann fuer die neue Hauseigene Integration von Unity selbst entschieden, [MLAPI](#). Wir hatten sowieso vor, uns Multiplayer integrationen in Unity genauer anzuschauen, also bot sich die Gelegenheit gut an. Allerdings haben wir uns an dieser Stelle etwas überschätzt bzw. den Aufwand unterschätzt. MLAPI ist sehr neu und noch in der Entwicklung. Die Dokumentation ist **sehr Lückenhaft** und die ersten Tutorials kamen leider auch erst während des Semesters. Wir wurden immer wieder zurückgeworfen, da wir bei jedem neuen Feature wieder neue Probleme hatten. Gegen Ende des Semesters waren wir dann aber doch zu Stolz, um den gesamten Multiplayer Code nochmal neu zu machen und durch eine andere Integration wie zum Beispiel Photon zu ersetzen. Im Nachhinein hätte uns das sicher viel Zeit erspart.

Unser Spiel wird nicht von einem Server gehostet, sondern der Host ist selbst ein Spieler. Zu Beginn hatten wir überlegt, Stationen über Bluetooth innerhalb der HBK zu erkennen. Dazu wäre die Architektur sehr praktisch gewesen. Allerdings hat sich unser Konzept im Laufe des Semesters sehr verändert und einen dedicated Server irgendwo zu hosten und die Spieler dahin zu verbinden wäre eine (nicht kostenlose) Alternative gewesen, und hätte die Kommunikation unter den Spielern erleichtert. Im allgemeinen sind wir trotzdem sehr froh, dass wir den Multiplayer so aufsetzen konnten wie wir uns das vorgestellt hatten. Allerdings haben wir zu viel Zeit daran verloren, die wir lieber in andere Bereiche das Spiel gesteckt hätten, vor allem in die Sichtbaren.

Assets

Die Assets und das Spielkarten Design wurden alle in Blender erstellt. Keiner von uns beiden hatte wirklich Erfahrung mit Blender zu Beginn des Semesters. Da wir uns 3D Modellierungen aber ebenfalls gerne anschauen wollten, war das ebenfalls eine gute Gelegenheit. Zu Beginn hatten wir mit vielen Problemen zu kämpfen, wie der Material Export zu Unity oder das UV-mapping in Blender. Hier haben Prozesse auch länger gedauert als erwartet.

Source Control

Während des Semesters hatten wir ein privates Git Repository für das Spiel. Wir haben es zur Abgabe nun public gemacht: <https://github.com/hannahkriegler/SaARBortage>. Da sind noch weitere Links zu Dokumentationen die wir im Laufe des Semesters erstellt hatten. Das Spielkonzept ist da auch noch deutlicher dokumentiert, allerdings sind da auch noch features drin beschrieben, die es nicht mehr mit ins Game geschafft haben.

Fazit

Allgemein sind wir etwas traurig, dass wir unser Spiel nicht mehr so fertig bekommen habe, wie wir es ursprünglich gedacht hatten. Viele der eigentlichen Features mussten im Laufe des Semesters aus Zeitgründen gestrichen werden. Leider haben wir verschiedenste Prozesse total unterschätzt und uns etwas zu sehr verrannt. Besonders, da es schwierig ist, unsere Fortschritte zu sehen und

auszuprobieren. Wir hoffen, wir konnte das vermitteln, was wir umsetzen konnten. Denn am meisten haben wir ein modulares Konzept implementiert, dass man super durch verschiedene Features erweitern kann, für die wir nun nur keine Zeit mehr hatten.

Trotzdem hatten wir sehr viel Spass mit dem Projekt. Besonders als wir soweit waren, das Spiel in einer Gruppe auszuprobieren. Zuzusehen, wie die Mitspieler sich Ihre Rolle überlegt haben und die Stationen zu suchen war sehr cool! Allgemein wollen wir uns nochmal bei euch für euer Feedback im Laufe des Semesters bedanken und hoffen, dass unser Spielkonzept einigermaßen verständlich übergekommen ist! :)

Disclaimer: Es gibt Bugs. Manchmal werden Stationen nicht richtig registriert wenn sie geschafft werden. Dadurch wird der Tag nie beendet und die Abstimmung startet nicht. Bei der Abstimmung kann es passieren, dass die Clients sich untereinander nicht finden und das Spiel crashed.
