


Entwicklungsprojekt WS 21/22

Audit 2



Korrektur der Idee für das Entwicklungsprojekt aus Audit 1

- Zielgruppe: 10-14 Jahre, entspricht Ende der Primary School und Anfang der Junior Secondary School bei Kindern in Afrika
- Zielort: Nairobi, Kairo, Lagos und Johannesburg
- Idee wird als PC-Version umgesetzt

- 
- > Zielgruppe: Wir empfanden diese Altersgrenze für passend da im Alter von 10-14 Jahren die Kinder noch neugierig genug sind aber auch noch nicht zu alt um Interesse daran zu verlieren.
 - > Zielort: Diese Großstädte haben alle eine Bevölkerungszahl von über 4 Mio. Einwohnern und hat Vorteile im Bereich der erforderlichen Ausstattung und der Reichweite unsere Vorhabens.
 - > Entscheidung für ein PC-Programm da Schulen eher mit PC's ausgestattet sind, als mit Tablets o.Ä.

Projektrisiken

- Programm stürzt während spielen/lernen ab
- Nutzer schafft ein Level nicht
- Sprachbarriere
- Über- oder Unterforderung



... und die daraus abgeleiteten Proof of Concepts

Absturz

- Beschreibung: Das Programm wird ausgeführt und das Spiel wird gespielt, bis man es geschafft hat.
- Exit: Programm läuft einwandfrei. Spiel wurde abgeschlossen.
- Fail: Programm stürzt ab
- Fallback: Automatisches Zwischenspeichern, damit bei erneutem Start des Spiels kein bzw. nur ein geringer Fortschritt verloren ist; Kontaktieren des Supports, falls Zwischenspeichern erfolglos war.

Level-/Spielabschluss

- Beschreibung: Der Nutzer spielt ein Level/Spiel, bis er es geschafft und damit erfolgreich abgeschlossen hat.
- Exit: Nutzer absolviert das Spiel ohne Hilfe
- Fail: Nutzer schafft es nicht, das Spiel zu absolvieren. Es wird Hilfe benötigt.
- Fallback: Es gibt einen Button, um sich Hilfe zu holen in Form von Hinweisen, Tipps oder Lösungsbeispielen (Lösung einer ähnlichen Aufgabe); Gewisse Anzahl an Leveln kann übersprungen werden, falls es trotz der Hilfe nicht geschafft wird.



... und die daraus abgeleiteten Proof of Concepts

Sprachbarriere

- Beschreibung: Der Nutzer startet das Programm in einer bestimmten Sprache.
- Exit: Der Nutzer versteht die Sprache.
- Fail: Der Nutzer versteht die Sprache nicht.
- Fallback: Am Anfang erscheint ein Dialog, in welchem alle möglichen Sprachen auf der jeweiligen Sprache aufgelistet sind. Der Nutzer wählt die gewünschte Sprache aus.

Unter-/Überforderung

- Beschreibung: Der Nutzer spielt das Spiel. Er versteht das Spiel und die Sprache und hat keine technischen Probleme.
- Exit: Der Nutzer hat Spaß am Spiel und schließt es erfolgreich ab. Am Ende kann er es bewerten.
- Fail: Dem Nutzer gefällt das Spiel/Level nicht, es fällt ihm zu schwer oder zu leicht. Er möchte es überspringen oder beenden.
- Fallback1: Es gibt einen Button, um eine Beschwerde bzw. einen Verbesserungsvorschlag einzureichen. So ist man ständig im Austausch und kann besser auf die Bedürfnisse der Zielgruppe eingehen.
- Fallback2: Wenn ein Nutzer zu schnell durch alle Spiele kommt, wird ihm nach einigen Spielen angeboten, die Schwierigkeit zu erhöhen. Wenn ein Nutzer ständig Probleme hat, ohne Hilfe weiterzukommen oder nur Level überspringt, wird ihm angeboten, die Schwierigkeit herunterzustellen.



Unity

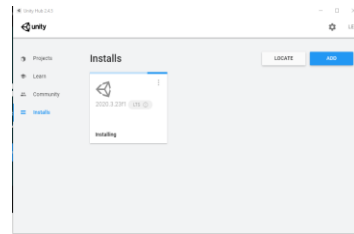
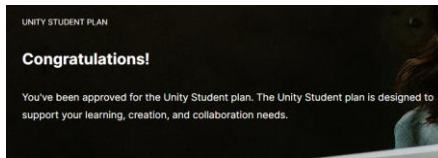
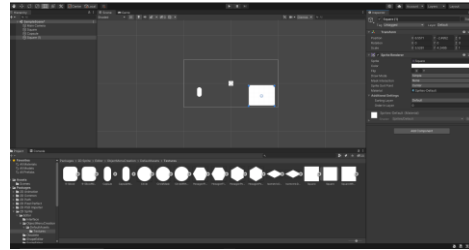
➤ Lernspiel in 2D

➤ Relevante Videos:

<https://www.youtube.com/watch?v=on9nwbZngyw>

<https://www.youtube.com/watch?v=2MimgNvgrNg>

<https://www.youtube.com/watch?v=pwZpJzpE2IQ>



-
- > Unity ist sehr bekannt und verbreitet vor allem auch in der Spielewelt
 - > Für 2D-Spiele auch benutzerfreundlicher als beispielsweise OpenGL
 - > Es gibt mehr und bessere Tutorials, die einem bei Hilfe und Fragen auch gut weiterhelfen können