

Kids Secure

Entwicklungsprojekt 23/24 - 2. Audit

Fortschritt seit Audit 1

- Projektrisiken
- Anforderungsermittlung
- Funktionale Use Cases
- Proof of Concept Konzept
- Aktualisierung der Artefakte anhand des Feedbacks
- Wireframes

Weitere Informationen zu der Bearbeitung findet sich auf der letzten Folie unter Projektplan, oder im Wiki unter dem selbem Artefakt.

https://github.com/AnoukMartinez/EP_WS23_Ganz_Khamkaew_Martinez/blob/main/Artefakte/Projektplan_v2.pdf



Wir haben uns dazu entschieden, die Projektrisiken nach Einflussgrad von unserer Seite zu ordnen. Kleine Werte (siehe Wiki) stellen Risiken dar, auf die wir als Programmierer und Entwickler der Projekts einen großen Einfluss haben können, größere Werte stellen Risiken dar, die eventuell außerhalb unseres Einflussbereiches liegen.

Unsere vollständigen und geordneten Projektrisiken finden sich auch noch einmal im Github Wiki:

https://github.com/AnoukMartinez/EP_WS23_Ganz_Khamkaew_Martinez/wiki/06-Projektrisiken

Wie realistisch ist die Lösung der wichtigsten Risiken in unserem Prototypen?

Schüler*innen werden nicht genügend vom System motiviert und brechen die Aufgaben beim Bearbeiten ab	Lehrer*innen fehlt fachliches Wissen um Schüler*innen bei Fragen/ Problemen zu unterstützen	Fehlerhafte / Ungenaue Darstellung an einem zu kleinen Display	Fragen / Aufgabestellungen werden nicht von allen verstanden	Fortschritt ist unübersichtlich für Lehrkräfte, und führt dazu, dass das System ungeeignet für eine Bewertung der Schüler ist	Die Aufgaben des Systems werden nicht in der vorhergesehenen Reihenfolge gestellt
Schlechtes Feedback nach einer nicht richtig beantworteten Aufgabe wird ersetzt durch neutralere Erklärungen und Tipps mit einem Tippmünzen System	Das Spiel hat selber interne Erklärungen zu den Aufgaben, um den Lehrern diese Aufgabe im schlimmsten Fall "abzunehmen" Lehrer sollen auch in der Lage sein, das Spiel zu spielen und zu lernen	In der Implementierung könnten wir genauer auf Plattformspezifische Schwierigkeiten achten	Kleine Tooltips sind mit einem Klick auf ein Fragezeichen bei jeder Aufgabe erhältlich, falls der Aufgabentyp für manche Schüler nicht intuitiv genug ist	Sollten wir bei der Implementierung drauf achten, dass das Design des UI einfach verständlich ist	Sollten wir bei der Implementierung drauf achten, dass man sich nicht selbst "softlocken" kann

Wir haben uns überlegt, wie wir die Projektrisiken vermeiden oder minimieren können. Hierzu haben wir die wichtigsten Risiken in unserem Einflussbereich noch einmal näher angeschaut, und uns konkrete Lösungen überlegt, die wir in unserem Prototypen verwirklichen können. Mehr hierzu findet sich ebenfalls im Github Wiki:

https://github.com/AnoukMartinez/EP_WS23_Ganz_Khamkaew_Martinez/wiki/06-Projektrisiken

Wie realistisch ist die Lösung der wichtigsten Risiken in unserem Prototypen?



Wir haben uns überlegt, wie wir die Projektrisiken vermeiden oder minimieren können. Hierzu haben wir die wichtigsten Risiken in unserem Einflussbereich noch einmal näher angeschaut, und uns konkrete Lösungen überlegt, die wir in unserem Prototypen verwirklichen können. Mehr hierzu findet sich ebenfalls im Github Wiki:

https://github.com/AnoukMartinez/EP_WS23_Ganz_Khamkaew_Martinez/wiki/06-Projektrisiken

Beispiele unserer Anforderungsanalyse

Funktionale Anforderungen

Das System sollte den Schüler*innen die Möglichkeit bieten, für ihre Lernfortschritte Feedback zu erhalten.

Das System muss den Schüler*innen die Möglichkeit bieten, durch Klicken und Texteingaben die Aufgaben zu bearbeiten.

Qualitative Anforderungen

Das System sollte den Schüler*innen die Möglichkeit bieten, Spaß bei der Bearbeitung der Aufgaben zu haben, damit diese nicht die Motivation verlieren.

Organisationale Anforderungen

Das System sollte von dem Schulministerium gefördert werden, und von Interesse zu sein in das Curriculum integriert zu werden.

Hier einige Beispiele unserer Anforderungsanalyse. Die vollständige Anforderungsanalyse findet sich in unserem Github unter Artefakten.

https://github.com/AnoukMartinez/EP_WS23_Ganz_Khamkaew_Martinez/blob/main/Artefakte/Anforderungsermittlung.pdf

Proof Of Concept

- Prototyp mit zwei Beispielaufgaben
- Feedback nach Beantwortung
- Spezifische Exit- und Fail-Kriterien

Als Proof Of Concept entwickeln wir einen Prototypen für eine Webapp. Der Prototyp soll ein Menü und zwei Beispielaufgaben enthalten. Mit dem Menü kann interagiert, und eine Aufgabe gestartet werden. Die Aufgabe wird dann angezeigt und kann bearbeitet werden. Beim richtigen beantworten der Frage wird eine entsprechende Punktzahl zum Score hinzugefügt. Wenn eine Frage falsch beantwortet wurde, wird eine Erklärung angezeigt.

Wenn die letzte Aufgabe erfolgreich bearbeitet wurde, wird ein finaler Score ausgegeben. Je nachdem, wie gut die Fragen beantwortet wurden, wird entsprechend Feedback an den Nutzer ausgegeben. Bei <50% richtig beantworteten Fragen wird ein negatives Feedback gegeben. Bei >=50% wird ein gutes Feedback gegeben. Bei 100% wird ein perfektes Feedback gegeben.

Hier unser Proof of Concept. Die Implementierung wird bis zum nächsten Audit erfolgen, und ist bereits in unserem Projektplan vermerkt.

https://github.com/AnoukMartinez/EP_WS23_Ganz_Khamkaew_Martinez/wiki/07-Proof-of-Concept

Proof Of Concept

Exit-Kriterien

Die Aufgabe kann erfolgreich abgeschlossen werden. Es wird ein Endergebnis angezeigt.

Fail-Kriterien

Daten aus der Datenbank können nicht richtig interpretiert werden.

Gespeicherte Daten werden beschädigt.

Verbindung zum Server kann nicht hergestellt werden.

Verbindung zum Server dauert länger als 10 Sekunden.

Verbindung zum Server bricht während des Bearbeiten der Aufgaben ab.

Die Schüler verlieren während der Bearbeitens der Aufgabe die Motivation, und brechen während des bearbeitens ab.

Die Schüler werden währene des Schulalltages nicht fertig und haben keine Möglichkeit, die Aufgaben zu Hause weiter zu bearbeiten.

Fallbacks

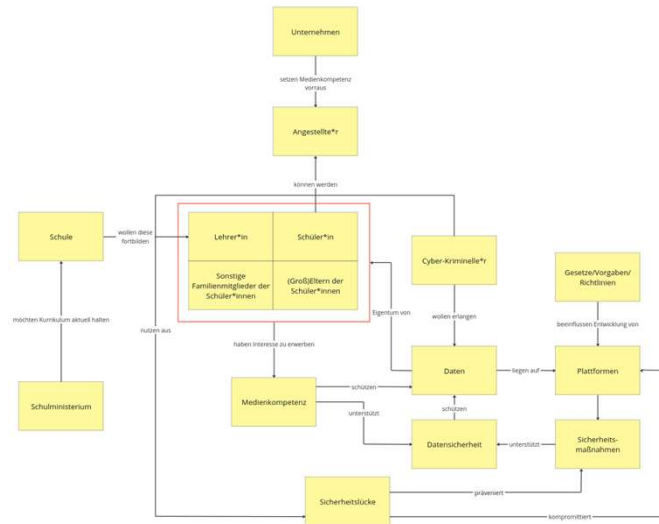
Der Fortschritt wird zwischengespeichert, damit bei Verlust der Verbindung nicht immer alle Aufgaben neu bearbeitet werden müssen.

Das System erzählt eine Geschichte, die die Schüler nur zuende erleben können, wenn sie alle Aufgaben bearbeiten.

Noch einmal eine ausführlichere Erläuterung der Exit- und Failkriterien, sowie den Fallbacks. Eine vollständige Dokumentation lässt sich im Wiki finden.


https://github.com/AnoukMartinez/EP_WS23_Ganz_Khamkaew_Martinez/wiki/07-Proof-of-Concept

Angepasstes Domänenmodell



Wir haben das Domänenmodell so angepasst, dass Familienmitglieder, v.a. Großeltern und Geschwister repräsentiert werden. Der rote Rahmen bedeutet, dass wir unser System hauptsächlich an Schüler*innen und Lehrer*innen ausrichten. Diese können zu Angestellten in Unternehmen werden. Die Angestellten sind nicht die Benutzer unseres Systems.

Aktualisierte Persona: Barbara Arnolds

- Alter: 70
 - Geschlecht: weiblich
 - Bildung: Rentnerin, gelernte Floristin
 - Einkommen: bezieht Rente
 - Familienstand: verheiratet mit Wolfgang, zwei Katzen, drei Kinder, vier Enkel
 - Wohnsituation: Vorstadt, Haus
 - Stakeholder: Familie
- 
- Kenntnisse von Technologie: Barbara hat kein IT Verständnis, einen Laptop oder Computer besitzt sie nicht. Auf der Arbeit früher kam sie auch ohne eine digitale Kasse aus.
 - Vor kurzem wurde sie Opfer eines Trickbetrügers. Jemand rief sie an und gab sich als ihr Enkelkind aus und konnte sich auf diese Weise mehrere tausend Euro von Barbara auszahlen lassen. Diese Erfahrung hat die ältere Dame sehr eingeschüchert.
 - Endgeräte: Barbara ist im Besitz eines Smartphones, welches sie nicht ohne die Hilfe ihrer Enkel benutzen kann. Bisher mussten sie diese immer sehr geduldig mit ihrer Oma sein, da ihr das Lernen und Handhaben mit der Technik schwer fällt. Barbara bekam das Smartphone während der Pandemie geschenkt, damit sie während der Lockdowns mit ihrer Familie in Verbindung bleiben konnte, ohne zu vereinsamen. Seit ihrer Negativerfahrung liegt ihr Gerät jedoch unbenutzt im Schrank.

Zwei Beispiele zu unseren aktualisierten Personas. Weitere Personas finden sich in unserem Github unter Artefakten, insbesondere unter Personas die mit V2 gekennzeichnet sind.

https://github.com/AnoukMartinez/EP_WS23_Ganz_Khamkaew_Martinez/tree/main/Artefakte/Personas

Aktualisierte Persona: Raphaela Bayer

- Alter: 10
- Geschlecht: weiblich
- Bildung: Gymnasium, 5. Klasse
- Einkommen: Grundsicherung
- Familienstand: ledig, Einzelkind
- Wohnsituation: Dorf
- Stakeholder: Schülerschaft



- Kenntnisse von Technologie: Raphaela hat bei ihren Freundinnen von der App TikTok gehört und sich diese installiert. Raphaela ist hin und weg von der App und verbringt jede freie Minute mit ihr. Sie hat sogar angefangen eigene Videos zu drehen und zu veröffentlichen. Ohne groß darüber nachzudenken - oder gar abschätzen zu können -, welche Konsequenzen Raphaelas Verhalten mit sich führen könnte, teilt sie auch private Dinge wie Namen, Orte und andere sensible Daten.
- Endgeräte: Raphaela lebt mit ihren Eltern in einem sehr abgelegenen Ort, weshalb sie zur Einschulung auf die weiterführende Schule ein Smartphone geschenkt bekommen hat. Der Bus fällt öfter aus und so kann sie jederzeit um Hilfe bitten.

Zwei Beispiele zu unseren aktualisierten Personas. Weitere Personas finden sich in unserem Github unter Artefakten, insbesondere unter Personas die mit V2 gekennzeichnet sind.

https://github.com/AnoukMartinez/EP_WS23_Ganz_Khamkaew_Martinez/tree/main/Artefakte/Personas

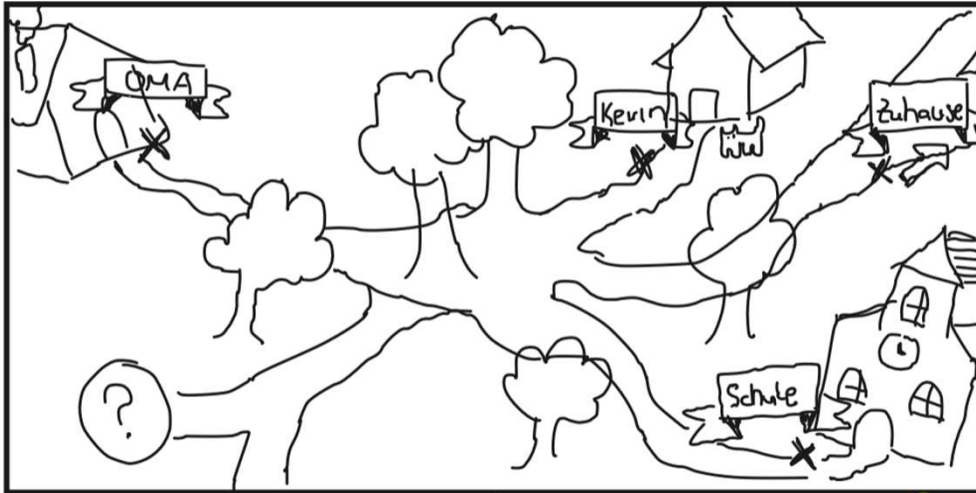
KidsSecure

- Aufklärung über Phishing durch die Bearbeitung von Aufgaben
- Storytelling als Motivationsfaktor
- Die Aufgaben hängen mit der Story zusammen
 - Point and Click
- NPCs liefern Informationen und Tipps zur Bearbeitung von Aufgaben
- Hinweismünzen

Als eines der Projektrisiken hatten wir definiert, dass die Schüler die Bearbeitung der Aufgaben aufgrund eines Mangels an Motivation bereits frühzeitig abbrechen. Um dies zu vermeiden, hatten wir überlegt, eine zusammenhängende Geschichte mit einem Protagonisten im Alter der Zielgruppe zu entwerfen. So müssen die Schüler die Aufgaben immer weiter bearbeiten, um die Fortsetzung der Story zu bekommen.

Um mögliche Frustrationen zu vermeiden, haben wir ein System entwickelt, in dem NPCs Hinweise an die Schüler geben, falls diese vermehrt eine Aufgabe nicht korrekt bearbeiten. Zudem soll es begrenzte Hinweismünzen geben, die es den Schülern ermöglicht, Tipps für Aufgaben zu bekommen, in denen sie sich besonders schwer tun.

Wireframes: Die Oberwelt



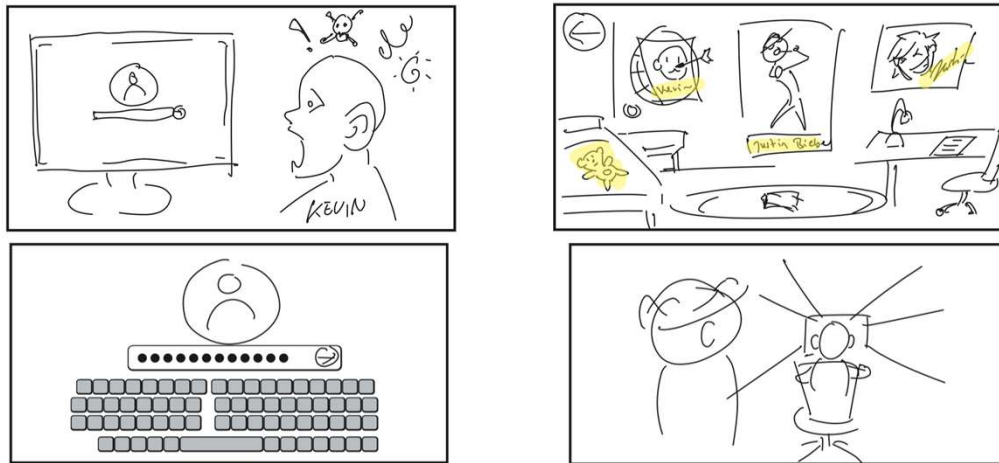
Die Oberwelt ist die Karte von der aus wir in einzelne Level navigieren können. Sie repräsentiert den Ort, in dem unser Protagonist und derjenige, mit dem wir die Aufgaben lösen, wohnt. Der Protagonist ist derjenige, mit dem wir die Aufgaben lösen, der Gespräche mit NPCs führt und uns manchmal mit internen Monologen Tipps gibt. Von der Karte der Oberwelt aus kann unser Protagonist sein Zuhause, das Haus seiner Oma, das von Kevin, die Schule, etc. erreichen. Je nach System wählen wir die Levels durch Mausklick oder Touch.

Wireframes: Kevins Zimmer



Das hier ist ein Beispiel für ein Level. Wir befinden uns in Kevins Zimmer. Wir können mindestens mit Kevin und seinem Computer interagieren. Es könnte auch darauf hinauslaufen, dass das Betreten eines Levels eine Aufgabe oder ein Gespräch mit einem NPC auslöst. Wir haben mindestens zwei Aufgabe geplant, bei der wir mit dem Computer interagieren müssen.

Wireframes: Aufgabenbearbeitung



Die Aufgaben werden teil der Story sein. Die Aufgabe werden mit einer Geschichte verbunden, was die Erinnerung an den Inhalt fördert. Hier z.B. Sehen wir Kevin, der seinen Computer mit seiner Schwester teilt. Beim letzten Login hat sie das Passwort geändert und Kevin kann sich nicht mehr einloggen. Als Kennworthinweis hat sie „süßcooldoof“ hinterlegt. Wir können von Kevins Zimmer in das Zimmer seiner Schwester navigieren. In ihrem Zimmer finden wir Hinweise darauf, was das Passwort sein könnte: Ihr Teddy ist süß; es hängt ein Foto von Kevin auf der Dartscheibe und sie besitzt zwei Poster von Justin Bieber. Einige Elemente sollen interagierbar sein (wenn man z.B. Auf die Poster klickt, soll Kevin reagieren: „Sie ist schon mega lange Fan von ihm. Sagt immer OMG Justin ist soo cool!!“) Das Kennwort ist teddyjustinkevin. Wir kehren zurück in Kevins Zimmer und wählen den Computer aus. Hier können wir nun das Passwort eingeben und die Aufgabe endet damit, dass Kevin an seinem Computer spielt.

Diese Aufgabe soll darauf hinweisen, die Passwörter nicht zu einfach zu vergeben und das Kennworthinweis, wenn es geht nicht zu benutzen. Wir haben uns eine andere Aufgabe überlegt, die Tipps gibt, wie man Passwörter vergibt.

Vielen Dank für die Aufmerksamkeit!