

Zuerst Allgemeinen Domäne und Artefakte

Danach konkreter auf Social Computing Bezug eingehen

Zum Schluss Umsetzung mit Hinblick auf das nächste Audit

Zeitplan

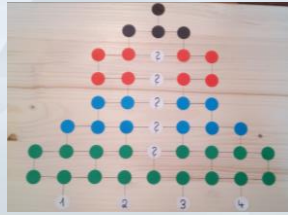
11.11.19	16.12.19	20.01.20	24.02.20
Problembeschreibung	Erste Funktionen programmieren	Ausarbeitung des Prototyps	Dokumentation überarbeiten
Zielsetzung und Anforderungen	Prototyp	Datenerhebung	Feinarbeiten am Prototyp
Domänenmodell	Ethische Aspekte	Gestaltungsprozess	Arbeit an Nebenperspektive
Aufgabenverteilung	Motivationen ausarbeiten	Systemanalyse	
Projektplan	Motive	Bibliotheken und Frameworks	
Zielhierarchie	Evaluation	Kernfunktionalitäten	
Unterschied analog und digital	Entwicklungsprozess	Poster	
Modellierungen	Nebenperspektive erarbeiten	kritische Reflektion	
Serious Game Analyse	Kommunikation im System		
Einarbeitung in Unity			

fppt.com

Für diesen Termin Fokus auf allgemeine Domäne und Social Computing
 Alle eigens gesetzte Ziele für 1. Audit erreicht
 Danach Beginn Programmierung und somit auch Erarbeitung der Nebenperspektive
 Für den letzten Termin Raum für Nacharbeitung durch mögliche Probleme gelassen

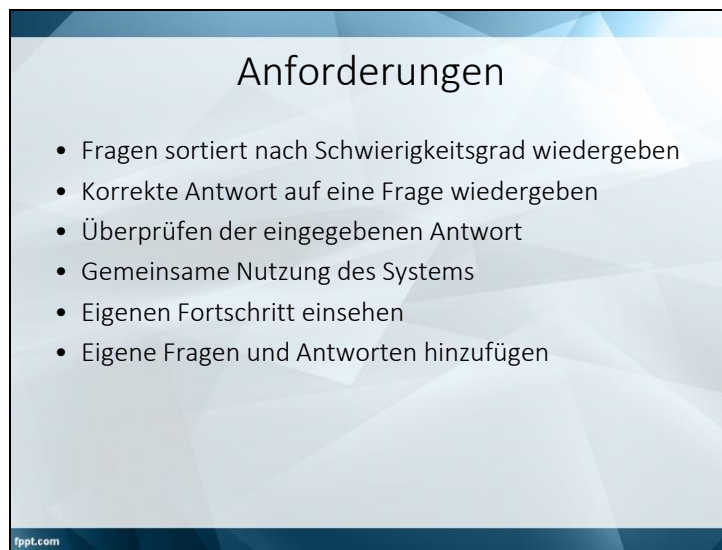
Projektidee

- Lernen als Schwierigkeit für Studierende
- Lernstoff kann nicht vereinfacht werden
→ dafür aber der Lernprozess
- Lernen durch spielerische Mittel angenehmer gestalten
- Datenbank-Lerner
"Datenbiz"



fppt.com

- Lernen = unangenehm/anstrengend – Bulimie Lernen
- Lernprozess zum Teil vereinfachen, damit der Stoff besser vertieft wird → Stoff bleibt länger beibehalten
- Rückblick auf SC: DB-Lernspiel – analoges Brettspiel
 - DB, da Modul sehr umfassend und gut abfragbar
 - Spiel hat Fragen in verschiedenen Schwierigkeitsstufen, steigt von unten nach oben
 - Pro Feld ist eine Frage richtig zu beantworten
 - Ziel: als erster oben ankommen
- In SC Regelwerk und analoges Design erstellt. Regelwerk dient als Basis, musste jedoch bereits in großen Teilen abgeändert werden
 - Im Wiki zu findende Recherche der Artefakte sind nur während EP entstanden



Technische Anforderungen an System

Fragen und Antworten korrekt wiedergeben

Antworten vom System überprüfen (durch MC oder Algorithmus)

Gemeinsame Nutzung → an einem Gerät 2 sozialen Aspekt fördern

Eigener Fortschritt sowie eigene Fragen und Antworten optional (kann, aber kein muss)



Klare Prioritätensetzung ist wichtig, um das Ziel nicht aus den Augen zu verlieren

Fundament: zum Einen Fachliche Korrektheit des Lernstoffs, da sonst

--> Pädagogischer Lehrauftrag nicht erfüllt wäre

Fehlerfreies Durchführen von Basisfunktionen sonst -> Motivationslöcher durch fehlerhafte Bedienung

Das führt uns zur nächsten Ebene, Berücksichtigung wichtiger Faktoren wie Spiellänge oder Spielgefühl, die Begründung ist die gleiche.

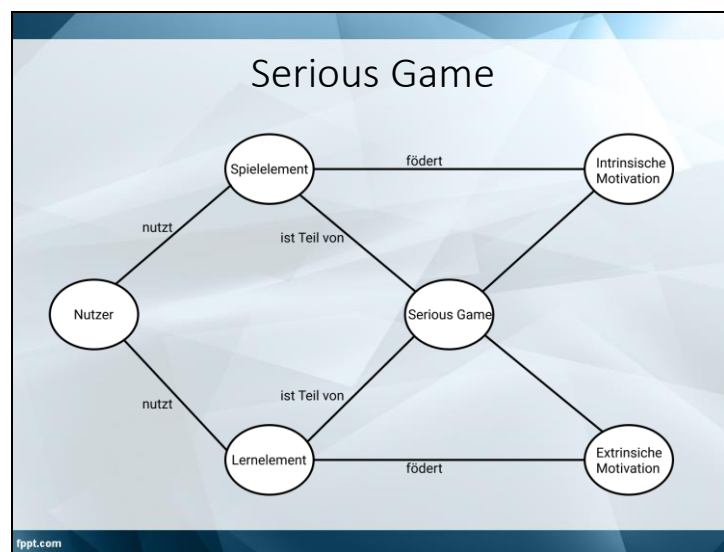
Balance zwischen intrinsischer und extrinsischer Motivation

Stringenter Designstil sorgt für gesteigerte Motivation vor allem, wenn man das Spiel über längeren Zeitraum spielt

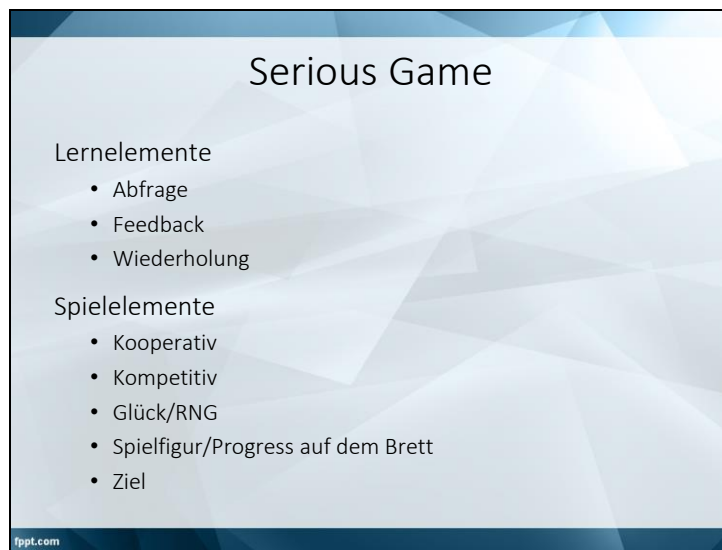
Gilt auch für die Speicherung des Fortschritts.

Am Ende: Polishing

Einbindung neuer Funktionen (z.B eigene Karten erstellen oder Sounds, Quality of Life changes etc)



Nutzer interagiert mit Serious Game → nutzt dabei Spiel- und Lernelemente
Dabei fördern Spielelemente die intrinsische Motivation; Lernelemente die extrinsische
Beide Arten der Motivation sind essentiell wichtig für einen stetigen und konsistenten
Lernerfolg



Abfrage: 2 Optionen: alleine oder mit anderen Mitspielern

Feedback: Feedback für die eigene Leistung außerhalb von Lehrstunden (Übung, Probeklausur, Test)

Wiederholung: Bei mehrfachem Spielen treten Fragen wiederholt auf → nochmals Vertiefung des Stoffs

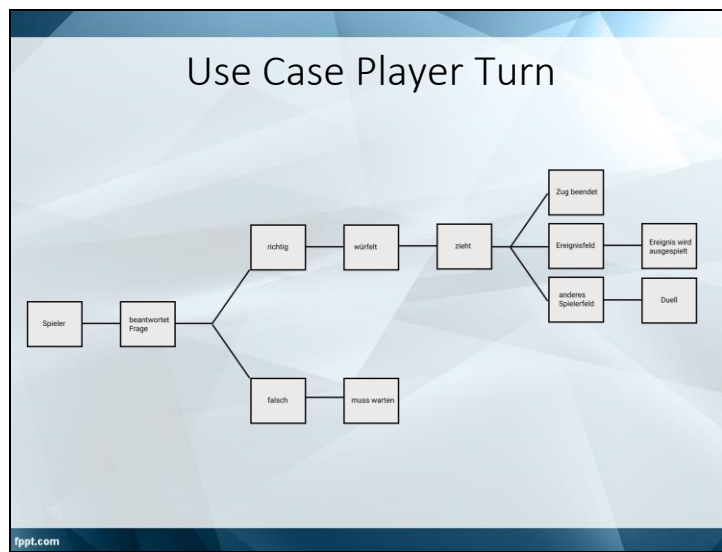
Kooperativ: Gestaltung des Spiels durch zB. Ereigniskarten ermöglicht es, anderen Spielern zu helfen oder gemeinsam andere zu behindern

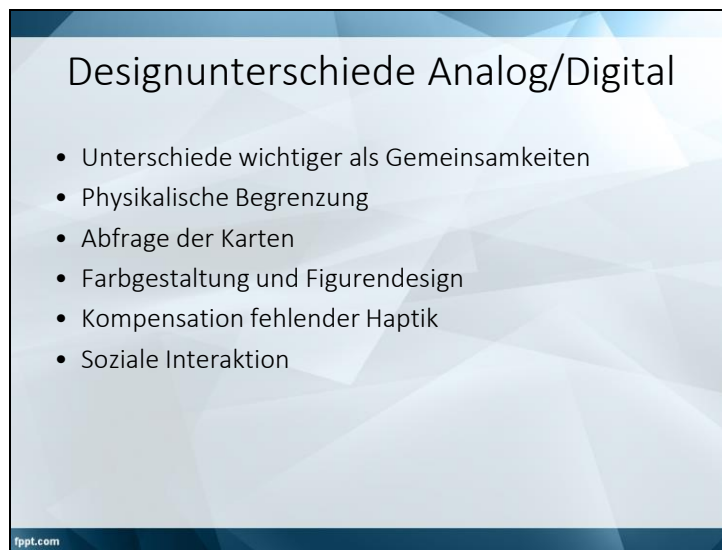
Kompetitiv: Konkurrieren mit anderen; mehr präsent als kooperativ, da jeder zuerst das Ziel erreichen möchte

Glück/ Random Number Generator: Würfeln, Ereigniskarten führt zu emotionaler Bindung der Spieler

Spielfigur: Wodurch der Spieler sich besser mit Spiel identifizieren kann

Ziel: Ist natürlich notwendig, damit ein kontinuierliches Interesse an dem Spiel besteht





Die Unterschiede sind wichtiger als Gemeinsamkeiten, da

Die Gemeinsamkeiten größtenteils keine Änderung benötigen, um auch im Digitalen zu funktionieren

Bei den Unterschieden ist genau das der Fall

Die Physikalische Begrenzung ist nicht mehr vorhanden im Digitalen, man hat keinen Verpackungsraum mehr um den man sich sorgen muss oder kann nahezu unbegrenzt Karten erstellen

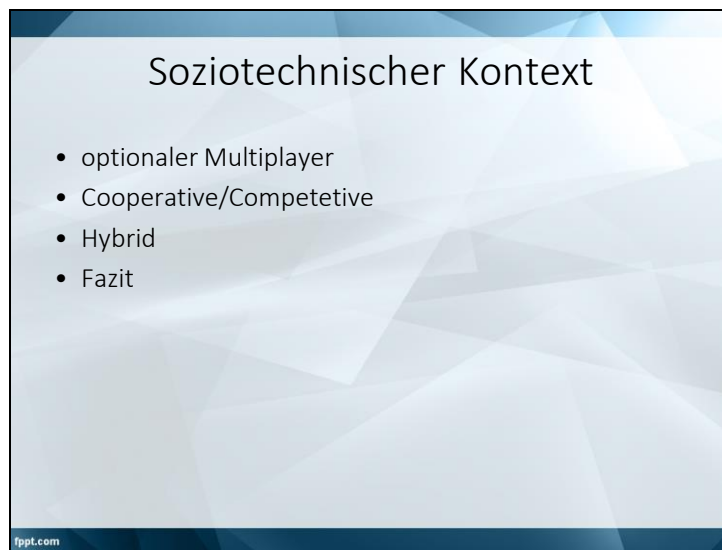
Die Abfrage der Karten läuft anders

Die Farbgestaltung und das Figurendesign finden anders statt, es muss auf andere Punkte geachtet werden (Sichtbarkeit auf dem Bildschirm anstatt Haptik z.B.)

Eben diese Haptik muss kompensiert werden, sodass es ein schönes Spielgefühl entsteht.

Die Soziale Interaktion, die bei einem Brettspiel gegeben ist, muss ebenfalls im Digitalen ermöglicht werden

--> haben wir durch gemeinsames Spielen des Programmes zusammen vor einem PC



Wir haben einen optionalen Multiplayer. Warum?

Die beiden Begriffe Cooperative/Competitive

Cooperative: gemeinsames Spielen auf ein gemeinsam gesetztes Ziel. Steigert Motivation und Spielspaß durch das vermitteln von zugehörigkeit zu einer Gruppe.

Competitive: Der Wettstreit fördert ebenfalls schon seit es Menschen gibt deren Motivation. Es gibt natürlich auch Hybriden wo Teams gegen andere Teams spielen.

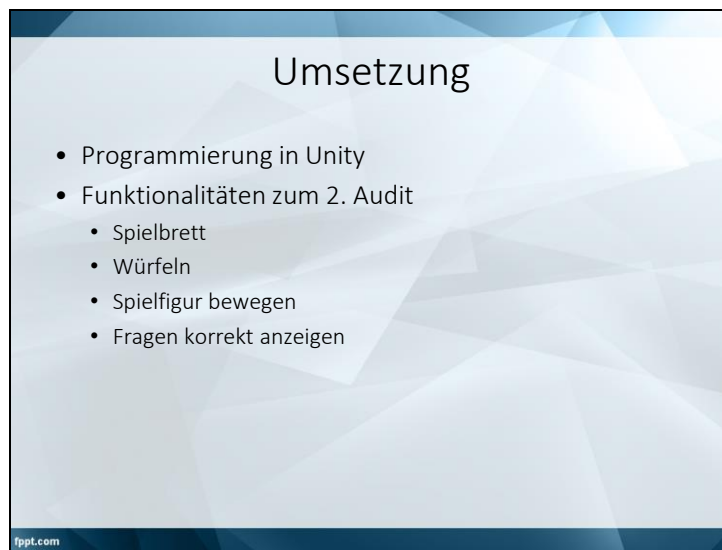
Fazit:

Gemeinsames Spielen den Spielspaß fördern kann und den Spieler möglicherweise emotional mehr an das Spiel bindet

Und der kompetitive Aspekt kann ebenfalls anspornen, also einen großen Schub in der Motivation geben.

Wir wollten aber auch ermöglichen, das Spiel alleine zu spielen, da nicht jeder Freunde hat oder die auch nicht immer können.

Man kann sogar wenn man möchte das Spiel als Hybriden nutzen und in Teams gegeneinander antreten.



Warum Unity?

- viele Möglichkeiten zur 2D Spielentwicklung
- Bietet Support
- Genügend Guides für einen guten Einstieg in die Entwicklungsumgebung

Fragen korrekt anzeigen, ggf. mit Antwort-Eingabe, ist jedoch abhängig davon, wie schnell die Implementierung der anderen Funktionen voran geht

