

Entwicklungsprojekt Medieninformatik WiSe 22/23 – Audit 3

Huyen Trang Nguyen, Uoc Linh Tony Nguyen, Julia Tretjakov



Übersicht

- Änderungen des bisherigen Projektablaufs
- Aktuelle Resultate
- Iteration Brainstorming
- Durchführung des POC's "Barcode Scannen"
- Weiterer Projektverlauf



Änderungen des bisherigen Projektablaufs



Änderungen des bisherigen Projektablaufs

Änderungen der Modellierungstechniken

- Präskriptive HTA entfernt
- Experience Mapping entfernt

Nun Conceptual Design mit:

- Essential & Use Cases nach Constantine & Lockwood
- Content Model durch Affinity Diagramming
- Navigation Model



Aktuelle Resultate

- Essential & Concrete Use Case nach Constantine & Lockwood
- Content Modelling



Essential und Concrete Use Case nach Constantine & Lockwood

wesenFüttern	
USER INTENTION	SYSTEM RESPONSIBILITY
Wesen füttern	Produktliste anzeigen
auswählen	das Füttern beenden

Abbildung 1: Essential Use Case nach Constantine & Lockwood

wesenFüttern	
USER ACTION	SYSTEM RESPONSE
App öffnen	Hauptseite anzeigen
das Füttern auswählen	Produktliste anzeigen
Produkt auswählen	Hauptseite anzeigen
	Werte aktualisieren

Abbildung 2: Concrete Use Case nach Constantine & Lockwood

[Komplete Use Cases](#)

Auf Basis der Use Cases kann nun die KJ-Methode (Affinity Diagram) angewendet werden.

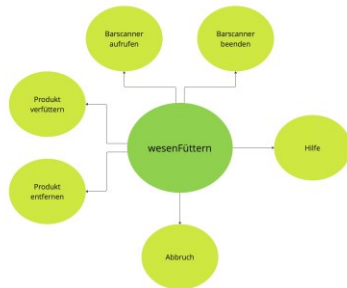
- Essential und Concrete Use Case nach Constantine & Lockwood besteht aus 3 Bestandteile:
 1. Name des Use Cases, der den allgemeinen Zweck ausdrückt
 2. User Intention/ User Action
 3. System Responsibility/ System Response
- Jeder Use Case beschreibt eine Menge verschiedener Interaktionssequenz vom Anfang eines Use Cases bis zu einem bestimmten Ende.
- Use Case oder Anwendungsfall enthält die Menge der möglichen Szenarien für Erreichen eines Ziels. Alle Interaktionen beziehen sich auf das gleiche Ziel desselben Hauptakteurs. Der Use Case beginnt mit dem auslösenden Ereignis und dauert so lange an, bis das Ziel erreicht oder aufgegeben wurde und das System seine Aufgaben in Bezug auf die Interaktion erfüllt hat.
- Hier als Beispiel haben wir das Use Cases für wesenFüttern gewählt. Die Interaktionen beziehen sich auf das Ziel, das Wesen mit dem ausgewählten Produkt zu füttern, daraufhin dessen Status aktualisiert und auf dem Hauptbildschirm der App angezeigt wird.

Das Essential Use Case beginnt mit der Absicht des Benutzer, der sein Wesen füttern will, und das Concrete Use Case beginnt mit der ersten Aktivität – App öffnen, um sein Wesen anzusehen und zu füttern. Die Use Cases dauern an, bis das Wesen erfolgreich gefüttert wird.

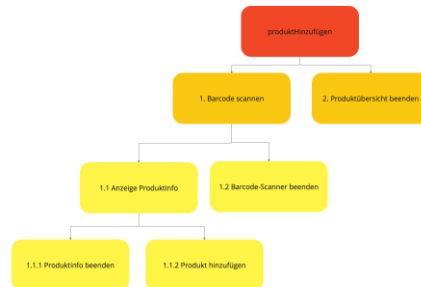
Content Modelling

- Insgesamt 5 **Task-Objekte** wurden anhand der Use Cases abgeleitet:
 - o Anwendungsobjekt „wesenErstellen“
 - o Anwendungsobjekt „wesenAnzeigen“
 - o Anwendungsobjekt „wesenFüttern“
 - o Anwendungsobjekt „produktHinzufügen“
 - o Anwendungsobjekt „miniGameSpielen“

- Affinity Diagram



- Task oriented



- Die Handlungsfelder "wesenFüttern" ist objects-oriented modelliert. Dies resultiert daraus, da untereinander keine Abhängigkeiten im Bedienungsablauf bestehen.
- Da der Task "produktHinzufügen" sich aus Aufgaben und mit ihnen verbunden Unteraufgaben aufbaut, ist die task-oriented modelliert.

Iteration Brainstorming

- Mindmap mit Beschreibungen der Anwendungslogik
- UI-Sketch



Was ist neu?


Die Attribute des Wesens wurden geändert/ erweitert und dessen Anwendungslogik zur Berechnung wurden festgehalten. Diese Attribute sind Hunger, Durst, Energie und Zufriedenheit sowie Diät.

Ein Minigame wurde hinzugefügt, dessen Konzept aus der Heldenmission des Spiels Cartoon Wars 2 entspringt.

Siehe: <https://youtu.be/C2xPj0JPqmw?t=2455>

In eigenem Projekt sind die Alleinstellungsmerkmale eine eigene Story im Bezug zur Nachhaltiger Ernährung und das Zusammenspiel mit einem Pet-Game.

[Github-Link zur Mindmap](#)



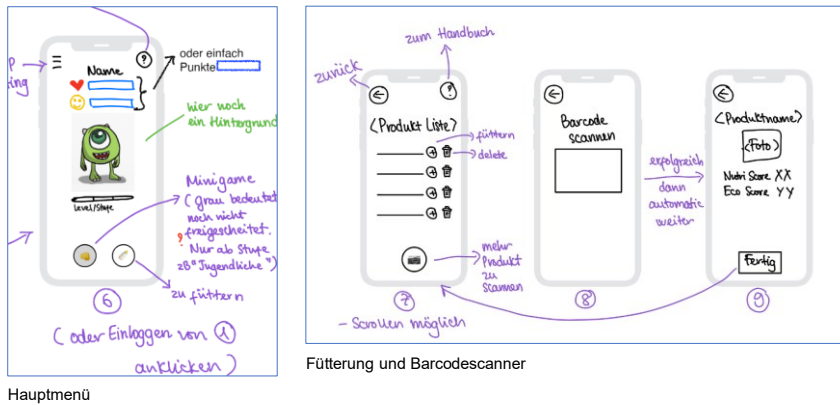
Während der Durchführung des Conceptional Designs sind uns Lücken bei den Anforderungen und UseCases aufgefallen.

Die Vorstellungen über die Konzeption des Systems war zwischen den Teammitgliedern unterschiedlich, sodass entschieden wurde hier eine Iteration anzusetzen.

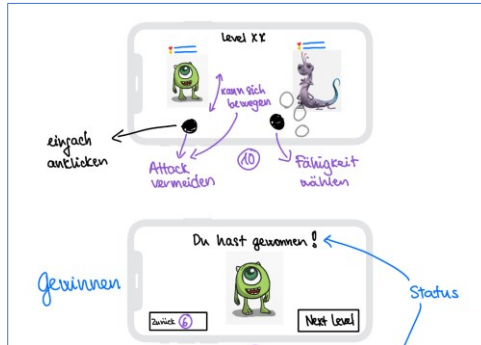
Zudem ist aufgefallen, dass ein reines Petgame mit der einzigen interaktiven Funktion "wesenFüttern" nicht ausreichend ist, um den Spieler längerfristig zu motivieren und auf eine nachhaltige Ernährung aufmerksam zu machen.

Daraufhin wurde ein Minigame hinzugefügt, weitere Ideen in einer Mindmap festgehalten und durch einen knappen UI-Sketch visualisiert. Der UI-Sketch dient ausschließlich zur Orientierung und repräsentiert nicht die vollständigen bzw. Tatsächlich umzusetzenden Funktionalitäten des Systems.

Erste Entwürfe



- Hier werden erstmal die Hauptfunktionen skizziert, die wir implementieren wollen. Erst ist der Hauptbildschirm bzw. das Hauptmenü der App, wo man sein Wesen mit dessen Status anschauen, sein Wesen füttern oder das Mini Game spielen kann. Und auf der oben linken und rechten Seite kann man auf die Einstellungen und das Handbuch der App zugreifen.
- Die essentiellen Funktionen unserer App sind die Fütterung und der Barcodescanner. Mit einem Klick auf den Babyflasche-Button kann man diese Funktionen abrufen. Hier hat man die Möglichkeit, die ganze Produktliste anzusehen und das Hinzufügen bzw. Löschen eines Produkt auszuwählen. Das Löschen wird einfach durch einen Klick auf den Korb-Button durchgeführt. Das Hinzufügen wird dann durch den Klick auf den Kamera-Button durchgeführt. Wenn das Barcodescannen von den Benutzer erfolgreich ist, wird dann eine Seite mit der Information des Produkts automatisch erscheinen, wo man da der Name, das Foto und das Nutri- und Eco Score des Produkts nachschauen kann.



Minigame

[der gesamte UI-Sketch](#)

Vergleiche Cartoon Wars 2

Video : <https://youtu.be/C2xPj0JPqmw?t=2455>


Entwickler war das Studio GAMEVIL, das von COM2US aufgekauft wurde. Die aktuelle Version des Spiels ist im heißt Cartoon Wars 3 und ist im Google Playstore/ Apple Appstore erhältlich.

COM2US: <https://www.com2us.com/holdings/game>

Cartoon Wars 2 APK Download: <https://cartoon-wars-2.de.uptodown.com/android>

Durchführung des POC's "Barcode Scannen"

Live Präsentation der Funktionen zum Abruf der
Produktinformationen mit Code Einsicht



Es wurde noch kein vertikaler rapider Prototyp erstellt, da wir uns entschieden haben zu iterieren und somit Teilfunktionen abgeändert werden könnten.

Jedoch ist das Barcode Scannen und die resultierenden ECO&Nutri Scores abzufragen ein Hauptbestandteil unseres Systems, das nicht mehr geändert wird.

Diese Funktionalitäten sind kritisch für die weitere Implementation des Systems, weswegen dieses POC priorisiert durchgeführt wurde.

Weiterer Projektverlauf

- POC's an das neue Spielkonzept anpassen
- Konkurrenzanalyse anpassen
- Anforderungen & Use Cases iterieren
- Conceptual Design fortführen
- Wireframes
- Implementierung





Vielen Dank für Ihre Aufmerksamkeit