

Audit 2 Vorstellung

Marko Kalaburda
Annika Lennep
Christian Pankiv

Inhalt

- Überarbeitetes Domänen Modell s.3
- Use Case Flow Chart s.4
- Ausarbeitung Phasen Konzept Entwurf
 - – Analyse Phase S.5-6
 - – Design Phase S.7-8
 - – Übung Phase S.9-10
 - – Review Phase S.11
- Erweiterte Risikoanalyse s.12-17
- PoCs s.18-23

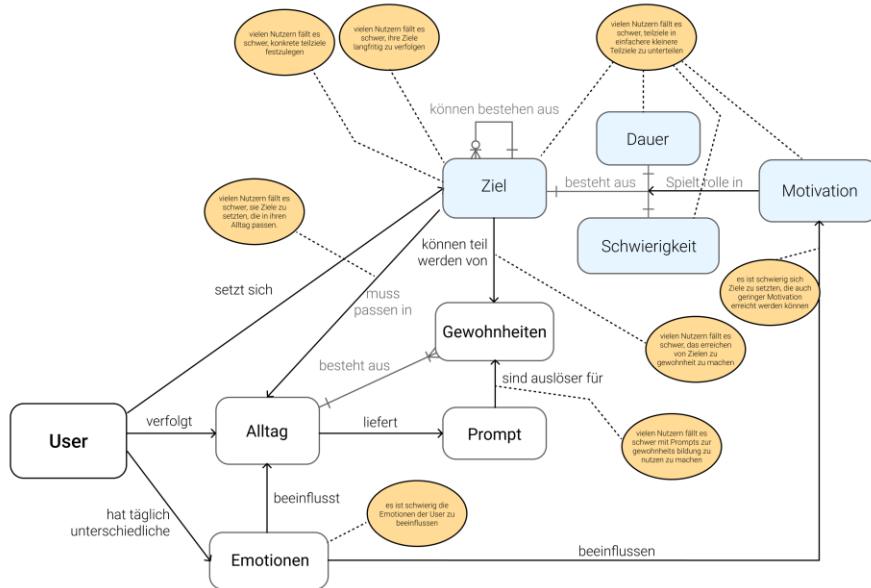
2

In der Präsentation zum 2. Audit werden zunächst die Artefakte vorgestellt, die nach dem Feedback aus dem 1. Audit und den Open Spaces überarbeitet und erweitert wurden. Dazu gehört das Domänenmodell, der Use Case Flow Chart und die Risikoanalyse.

Zusätzlich wurde der Ablauf der einzelnen Phasen in einer Schritt-für-Schritt-Modellierung dargestellt, um einen gedanklichen "Walk-Through" zu vollziehen und weitere Risiken, Schwachstellen und Herausforderungen bei der Implementierung zu identifizieren.

Die zum Schluss vorgestellten PoCs beziehen sich jeweils auf konkrete Risiken oder Machbarkeits-Aspekte.

Überarbeitetes Domänenmodell



3

Das Domänen Modell wurde nochmal überarbeitet.

Menschen setzen sich Ziele, die aus mehreren Teilzielen bestehen können. Ziele haben eine Erreichungs-Dauer und eine Schwierigkeit und werden durch die Motivation stark beeinflusst, da die Ziele nur erreicht werden können, wenn sie auch bei verschiedenen Motivationsstufen verfolgt werden können.

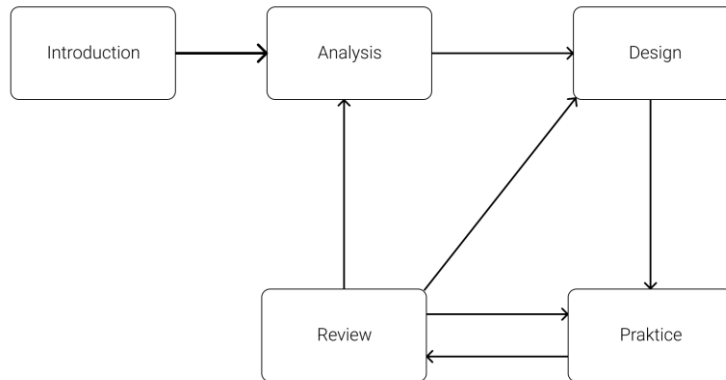
Die Motivation hingegen wird durch verschiedene Emotionen beeinflusst, die auch den Alltag beeinflussen.

Menschen verfolgen jeden Tag ihren normalen Alltag, in dem auch verschiedene Gewohnheiten stattfinden die teilweise durch Prompts ausgelöst werden. Die Ziele die der User sich dabei setzt müssen zwingend in den Alltag des Users Passen und können auch teil der Gewohnheit werden.

Verschiedenste Probleme die der User bei der Erstellung neuer Ziele überwinden muss sind in den gelben Blasen dargestellt.

Alle Diagramme sind auch Einzeln als PNG im GitHub Repository unter Artefakte/Audit 2 auffindbar.

Überarbeitete Use Case Flow Chart



4

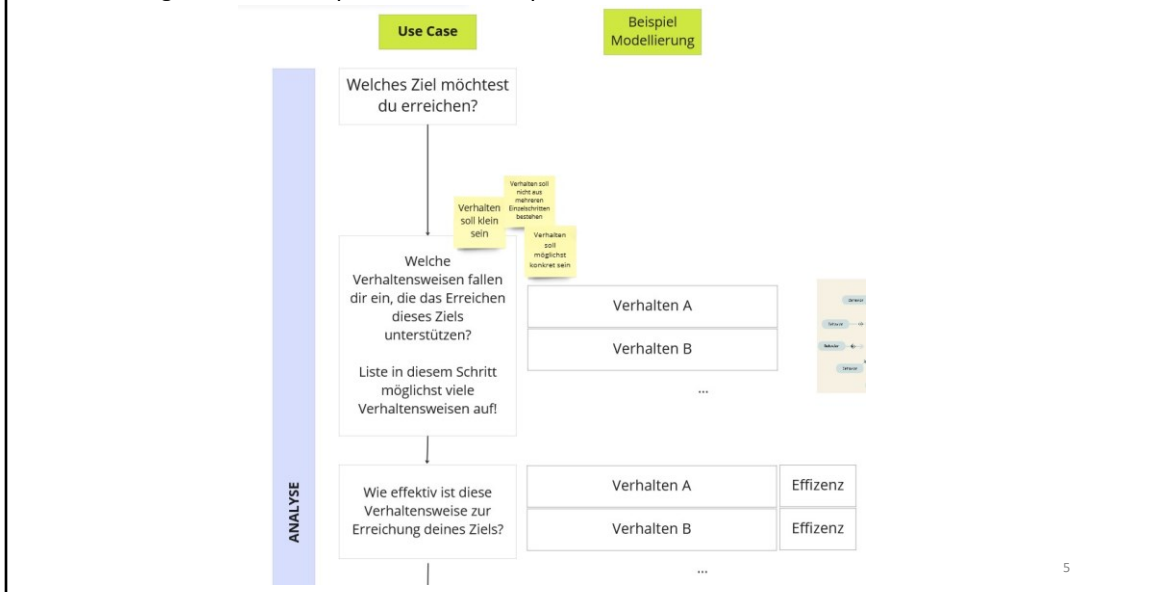
Im überarbeiteten Use Case wurde der Zyklus iteriert.

Nach der Analyse(Analysis), welche die Verhalten analysiert und sortiert, folgt das Design, welches dem Verhalten eine Einheit (Zeit/Wiederholung) und einen Prompt zuteilt.

Nach dem Design folgt die Übung(Practice), wo der Benutzer sein Verhalten in sein Leben überführen soll.

Das Review zeigt dann an an welchen Stellen der Nutzer eine Verbesserung anstreben kann.

Ausarbeitung Phasen Konzept Entwurf - Analyse



Der Phasen Konzept Entwurf dient der Identifikation weiterer Risiken und Stellt die Grundlage für den System Ablauf bei der Nutzung dar und zeigt die prinzipielle Aufteilung der User-System-Interaktionen in den aufeinander folgenden Phasen. Der Entwurf basiert auf den Phasen der Use Case Flow Chart. Die Risiko-Tabelle wurde auf der Grundlage dieses Entwurfs erstellt bzw. ergänzt.

In der Analyse-Phase werden die Ziele, Teilschritte und Verhaltensmöglichkeiten zur Zielerreichung erfasst und analysiert.

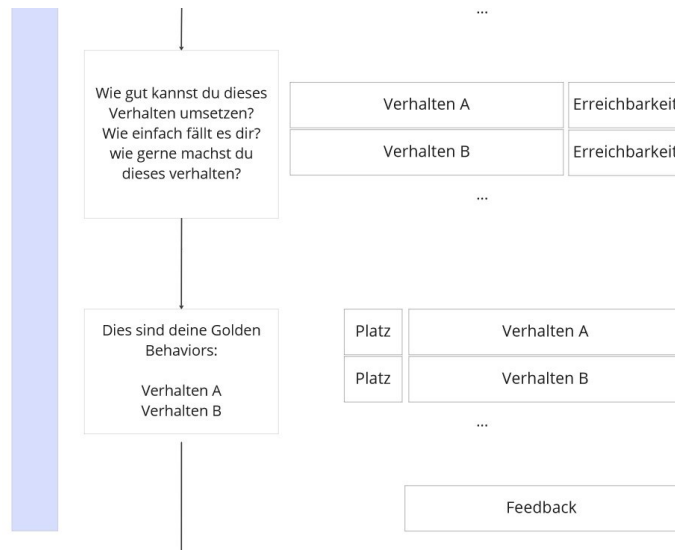
Der/Die UserIn legt im System das zu erreichendes Ziel und entsprechende Verhaltensweisen, mit ihr Effizienz das Ziel zu erreichen, fest. Diese werden für die Auswahl und die Bestimmung der „Golden Behaviors“ nach Fogg, zusammen mit der Erreichbarkeit des Verhaltens, benötigt.

Die Ausarbeitung ist detaillierter, als der zuvor gezeigte Phasen-Flow-Chart, entspricht jedoch zum gegebenen Projektzeitpunkt noch nicht der tatsächlichen Interaktion mit dem User, sondern stellt eine Annäherung zum Zweck der Risiko-Identifikation dar.

Das gesamte Modell ist in Miro einsehbar:

https://miro.com/app/board/o9J_IJqd05Y=/?share_link_id=11552071440

Ausarbeitung Phasen Konzept Entwurf - Analyse



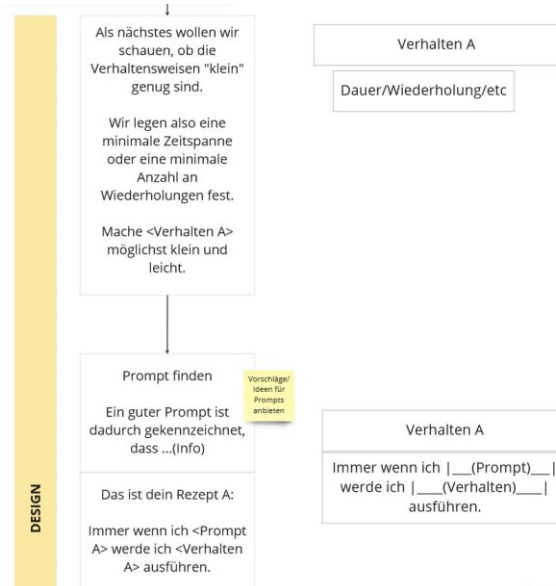
6

Anschließend legt der/die UserIn ihrer Einschätzung nach die Erreichbarkeit bzw. Schwierigkeit der einzelnen Verhalten fest.

Die besten Verhalten für den Nutzer ergeben sich dann aus einer Rangordnung der besten Effizienz und Erreichbarkeit für den Nutzer. (sie werden addiert und dann absteigend Sortiert)

Der/Die UserIn erhält eine Liste mit den besten Verhalten woraus dann eine gewählt werden kann, welche dann für die Erzeugung einer Gewohnheit genutzt werden soll.

Ausarbeitung Phasen Konzept Entwurf - Design



7

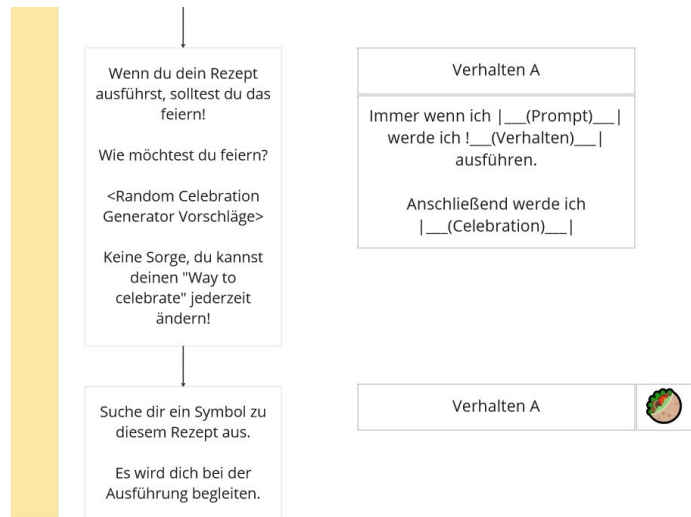
In der Designphase wollen wir die Eckdaten von dem/der BenutzerIn sammeln, die uns zeigen wie viel sich der/die BenutzerIn zutraut.

Dies messen wir an einer Einheit, in welcher das Verhalten ausgeführt wird.

Diese Einheit soll klein gehalten werden, um dem / der BenutzerIn die Leichtigkeit dieser Tätigkeit nahezubringen und für größere Einheiten vorbereiten.

Außerdem soll ein Prompt gefunden werden, welcher den/die UserIn an das Verhalten bindet, indem täglich ausgeführte Prompts verwendet werden. (Beispiele für Prompts werden als Beispiel angezeigt, um dem/der UserIn zu helfen eins für sich auszuwählen)

Ausarbeitung Phasen Konzept Entwurf - Design



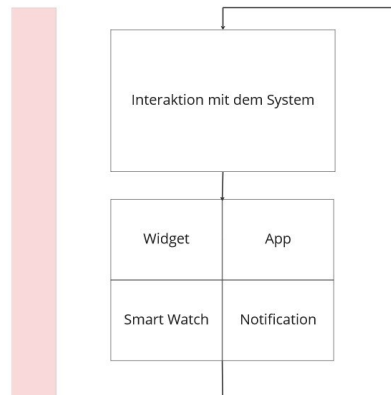
8

Um den/die BenutzerIn langlebig an ein Verhalten zu binden, sollte er/ sie eine positive Emotion damit verbinden.

Dies wird durch eine positiven Zuruf am Ende der Ausführung des zu erlernenden Verhaltens angewendet, um sich und sein Erfolg zu feiern.

Ein Symbol wird herausgesucht, welcher einen in diesem System begleitet, um sich selbst in diesem System zu identifizieren.

Ausarbeitung Phasen Konzept Entwurf - Übung

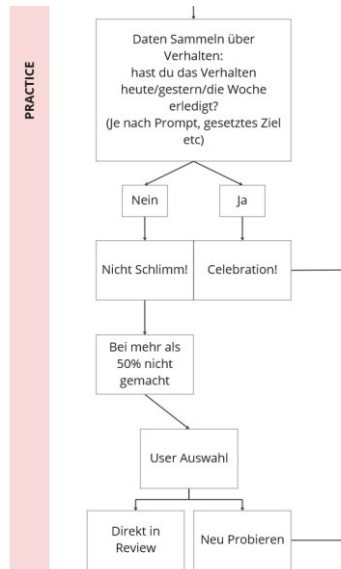


9

In der Übungsphase, soll der / die BenutzerIn das Verhalten üben, indem er/ sie es anwendet.

Das System kann auf mehreren Geräten mit einem interagieren (Widget/ App/ Smart Watch/ Notification).

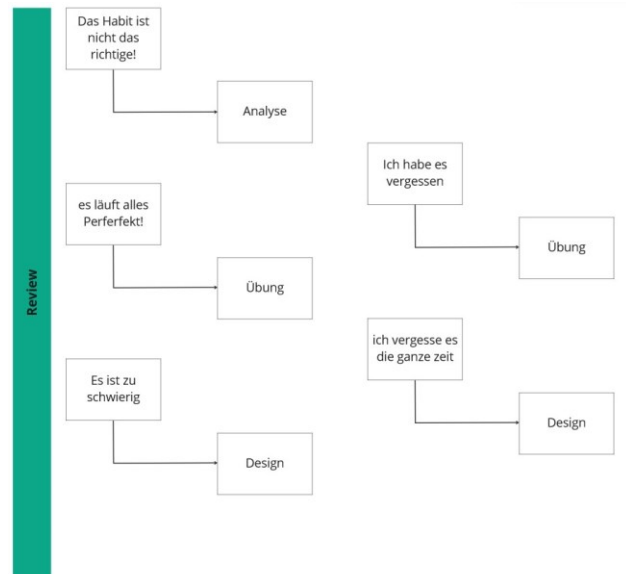
Ausarbeitung Phasen Konzept Entwurf - Übung



10

Es wird gesammelt wie oft der / die BenutzerIn das Verhalten angewendet hat. Am Ende der Übung werden die Schwierigkeiten aufgefasst und das Review vorbereitet.

Ausarbeitung Phasen Konzept Entwurf - Review



11

In der Review Phase werden die Daten die während der Übungs-Phase Gesammelt wurden Ausgewertet und dem User Präsentiert. Der User hat dann die Möglichkeit aus verschiedensten Optionen je nach Ergebnissen der Übungs-Phase zu wählen. Wenn der Nutzer nicht mit dem Habit zufrieden ist, kann er in der Analyse Phase ein anderes Habit wählen, die verschiedenen Verhalten anpassen oder neue Hinzufügen. Bei guter Ausführung oder Einmaligem vergessen des Habits und sonstiger guter Umsetzung sollte der User unverändert mit dem Habit fortsetzen. Falls der User jedoch mehr als die Hälfte der Zeit die Habits vergisst sollte an dem Design der Prompt verändert werden. Ähnlich sollte der User auch bei zu großer Schwierigkeit zu der Design-Phase zurück kehren um das Verhalten zu vereinfachen/verkleinern.

Erweiterte Risikoanalyse

User Risiken nach Phasen: Introduction

Phase	Risiko	Ursache	Auswirkung	Eintrittswahrscheinlichkeit	Schaden	Maßnahme
Introduction	Mangelnde Aufmerksamkeit	Ablenkung, Unkonzentriertheit	Mangelndes Verständnis, mangelnde Überzeugung	mittel	mittel	Einfache Gestaltung, kurze Sätze, kleine Interaktionen für Aufmerksamkeitserhalt
	Fehlende Motivation	Zweifel am Konzept, Versagensängste	Fehlende Ernsthaftigkeit bei der Interaktion, Auslassen von Schritten, Übergang in neue Phase wird nicht vollzogen	niedrig	hoch	Angsten begegnen, gute Erklärung anbieten, Beispiel-Rezepte zum Ausprobieren anbieten
	Zeitmangel	Arbeit, Privatleben, Termine	Keine Interaktion mit der App, mangelnde Ausarbeitung des Behavior-Change-Rezeptes	hoch	mittel	Mechanismen des Behavior Changes gut erklären, geringen Zeitaufwand betonen
	Versagensängste	Bestehende Vorstellungen von Motivation/ Verhalten	Vermeidungsverhalten, Startschwierigkeiten	hoch	sehr hoch	Betonen der Unbeständigkeit des Faktors „Motivation“, „We are not the problem. Our approach to change is. It's a design flaw – not a personal flaw.“
	Zweifel am Konzept der App	Fehlendes Verständnis	Positive Gefühle bleiben aus, Startschwierigkeiten	mittel	hoch	Wissenschaftlichen/ empirischen Hintergrund liefern, gute Erklärungen, Erfolgsbeispiele

12

Die Risiko-Tabelle zeigt die Risiken aufgeteilt nach User- und System. Hier werden zunächst die User-Risiken nach Phasen eingeteilt vorgestellt. In den PoCs werden einige dieser Risiken aufgegriffen und mögliche Maßnahmen vorgeschlagen. Den User-Risiken kann an vielen Stellen nicht technisch begegnet werden, weshalb hier die konzeptionelle Arbeit einen höheren Stellenwert einnimmt. Da beispielsweise "Versagensängste" eine hohe Eintrittswahrscheinlichkeit sowie einen hohen Schaden aufweisen, werden diese in einem eigenen PoC behandelt. Der Bezug zu den Risiken wird in den Notizen zu den PoCs herausgestellt.

Erweiterte Risikoanalyse

User Risiken: Analysis

Analysis	Unrealistische Zielsetzung	Hohe Ansprüche, Ehrgeiz, starke	Motivationseinbruch nach Anfangs-Peak	hoch	mittel	Aufforderung, realistisches Ziel zu wählen, Kontrollfragen
		Unzufriedenheit mit derzeitigem Verhalten				
	Nur wenige Verhaltensweisen gesammelt	Ideenlosigkeit, mangelnde Übung	„Swarm of Behaviors“ bleibt klein/ unvollständig	hoch	mittel	Hilfen vorschlagen, Beispiele, Methoden (z.B. Freunde fragen)
	Verhaltensweisen passen nicht zur Zielsetzung	Gedanken nur wenig systematisiert, unklare Vorstellungen	Ziel schwer oder nicht zu erreichen, Erfolge bleiben aus	niedrig	hoch	Beispiele, Leitfragen
	Effektivität der Verhaltensweisen falsch eingeschätzt	Fehlendes Vorwissen	Ziel schwer oder nicht zu erreichen, Erfolge bleiben aus	mittel	mittel	Behebung der Analyse-Probleme in der Review-Phase
	Umsetzbarkeit der Verhaltensweisen falsch eingeschätzt	Wenig Erfahrung, zu hohe Ansprüche an sich selbst	Verhalten nicht oder nicht immer durchführbar	mittel	mittel	Behebung der Analyse-Probleme in der Review-Phase

13

Auch bei der Analyse zeigt sich die Relevanz der konzeptionellen Ausarbeitung. Maßnahmen zur Begegnung der Risiken werden direkt in das Gesamtkonzept eingearbeitet. So sollen dem User zu den verschiedenen Schritten stets Beispiele und Leitfragen zur Verfügung stehen sowie die durch ihn zu erledigende Aufgabe (z.B. Verhaltensweisen identifizieren) möglichst präzise erklärt werden.

Erweiterte Risikoanalyse

User Risiken : Design

Design	Verhaltensweisen bleiben zu komplex/ aufwändig	Schwierigkeiten beim „Verkleinern“ der Verhaltensweisen, falsche Einschätzung der Umsetzbarkeit	Verhalten nicht oder nicht immer durchführbar, negative Gefühle	mittel	hoch	Leitfragen (z.B. „Ist das gewählte Verhalten auch an besonders stressigen Tagen ausführbar?“) -> Design-Flow (Fogg)
	Es wird kein Prompt gefunden	Ideenlosigkeit, fehlende Routinen im Alltag, Verfügbarkeits-Problem	Stopp der App-Interaktion, Zweifel	mittel	sehr hoch	Beispiele, Angebot System-Prompt zu setzen
	Prompt ist ungeeignet	Falsche Auswahl	Prompt führt nicht zu definiertem Verhalten	mittel	hoch	Beispiele, Kontrollfragen
	Rezept unklar formuliert/ nicht spezifisch genug	Fehlendes Verständnis, fehlende Orientierung an der Vorlage	Unsicherheit, wann Verhaltensweise zu zeigen ist, Verhaltensweise zu unspezifisch, um ausgeführt werden zu können	mittel	hoch	Beispiele, Kontrollfragen

14

Es gelten dieselben Bemerkungen wie zu den vorangegangenen Folien.

Erweiterte Risikoanalyse

User Risiken : Practice

Practice	Verhalten wird nicht eingehalten	<ul style="list-style-type: none">• Vergesslichkeit• Zeitmangel• Falsche einteilung der Parameter des zu ändernden Verhaltens• etc.	Die Applikation hat keinen Nutzen für den Benutzer	mittel	mittel	Auswirkung erkennen und positives Feedback geben
-----------------	----------------------------------	--	--	--------	--------	--

15

Es gelten dieselben Bemerkungen wie zu den vorangegangenen Folien.

Erweiterte Risikoanalyse

User Risiken : Review

Review	Daten Werden Falsch Interpretiert	<ul style="list-style-type: none">• User im Urlaub/ nicht Zuhause• kurzzeitig keine Zeit• etc.	Das System nimmt an, dass der Prompt nicht zu dem User passt	Hoch	niedrig	User die möglichkeit geben selber über Fortfahren zu entscheiden.
---------------	-----------------------------------	--	--	------	---------	---

16

Der Nutzer kann während einer Übungsphase bsw. kurzzeitig verhindert sein, was dazu führt, dass die Daten, die das System über die Übungsphase sammelt, falsch interpretiert werden können. Der User sollte daher selber das Fortfahren und die nächste Phase wählen können.

Erweiterte Risikoanalyse System-Risiken

System-Risiken

Risiko	Ursache	Auswirkungen	Eintritts- wahrscheinlichkeit	Schaden	Maßnahmen
Probleme mit Datenbankzugriff/ Datenverlust	Internetausfall, Probleme beim DB- Anbieter, Bugs	Daten nicht	hoch	mittel	Wichtige Anwendungsdaten im Cache speichern
Internetausfall	Probleme bei Netzanbietern, technische Störungen	Online-Daten nicht abrufbar, DB-Anfragen nicht möglich	hoch	niedrig	Möglichkeit, Behavior- Change-Konzepte lokal zu speichern
Datenschutz	Datenleck, fehlende Sicherheit	Rechtliche Konsequenzen, mangelndes Vertrauen, Image-Schaden	mittel	sehr hoch	Verschlüsselung, rechtliche Vorgaben beachten
Systemabsturz	Bugs, Absturz des technischen Geräts	Datenverlust	hoch	mittel	Fehler catchen, guter Code, Zwischenspeichern
Unklare Navigation	Schlechtes Interaktions- Design, schlechtes UI- Design	Frustration, fehlendes Verständnis der Funktionsweisen	mittel	hoch	MO-Grundlagen beachten

17

Bei den System-Risiken wurden die allgemeinen Risiken angeschaut, welche an verschiedenen Schnittstellen zum System .

PoCs

1. Introduction Slides

- Kein klassischer PoC möglich, da keine Exit-Kriterien, Fail-Kriterien und Fallbacks angebracht werden können. Trotzdem begegnen die Introduction Slides einigen Risiken aus der Risikoanalyse und sollen zum 3. Audit vorgelegt werden.
- Literatur & empirische Forschung zurate ziehen

18

In der Tabelle zu den User-Risiken sind in der Einführung, aber auch in den darauf folgenden Phasen, sind einige User-spezifische Risiken aufgeführt, denen durch eine gelungene Einführung begegnet werden kann. Dazu gehört beispielsweise die auf vergangene Fehlschläge und äußere Erwartungen zurückführbare "Versagensangst". Menschen neigen laut Fogg dazu, den Fehler bei der fehlenden Umsetzung neuer Verhaltensweisen bei sich selbst zu suchen und sich schuldig zu fühlen. Ein Grundsatz von "Tiny Habits" lautet: "It's a design flaw, not a personal flaw". Und so soll es ein Ziel der "Introduction" sein, diesen Grundsatz überzeugend zu transportieren. Auch Zweifel am Konzept der App und Verständnisschwierigkeiten zum Ablauf (siehe ebenfalls User-Risiken-Tabelle) sollen hier früh adressiert werden.

Tragende Exit- und Fail-Kriterien könnten an dieser Stelle definiert werden, wenn Daten von Probanden durch empirische Forschung erhoben würden. Da dies im Rahmen des Entwicklungsprojekts für uns nur schwer umzusetzen ist, werden stattdessen bereits vorliegende Forschungsergebnisse und allgemeine Literaturrecherchen zur Beurteilung der Risikobewältigung herangezogen.

PoCs

2. Analyse-Part in Code umsetzen

- Vom Setzen eines Ziels über das Sammeln und Auswerten von Verhaltensweisen hin zur Ausgabe der Golden Behaviors
 - Exit-Kriterium: die richtigen Golden Behaviors werden ausgegeben
 - Fail-Kriterium: falsche GBs, keine Ausgabe, Absturz etc.

19

In der Modellierung zum Ablauf der Analyse-Einheit ist dargelegt, wie der User zunächst ein Ziel bestimmt, dann Verhaltensweisen sammelt und diese nach den Kriterien "Effektivität" und "Umsetzbarkeit" bewertet. Anschließend werden die besten Verhaltensweisen ("Golden Behaviors") auf Grundlage der User-Eingaben bestimmt und ausgegeben. Dies soll zum 3. Audit in Kotlin-Code umgesetzt werden, um die Machbarkeit zu zeigen und verschiedene Inputs zu testen. Hierfür muss eine passende Bewertungsskala definiert werden sowie verschiedene Intervalle, die personalisierte Rückmeldungen ermöglichen (z.B.: Sind dies "nur" die besten gefundenen Behaviors oder sind sie tatsächlich "gut" oder "sehr gut" geeignet). Sollte keine der gesammelten Verhaltensweisen im oberen rechten Feld der Focus Map (vgl. Fogg) landen, kann zudem ein Ergänzen der Sammlung vorgeschlagen/ angeregt werden.

PoCs

3. Random Celebration Generator

- „Ways to celebrate“ in Datenbank speichern und eine randomisierte Ausgabe in der App erreichen
 - Exit-Kriterium:
 - bei jedem Aufruf wird ein neuer „Way to celebrate“ ausgegeben
 - Fail-Kriterien:
 - Verbindung zur Datenbank kann nicht hergestellt werden
 - Verbindung besteht, aber es wird kein „Way to celebrate“ ausgegeben

20

Bei der Verbindung mit einer Datenbank handelt es sich um eine der technischen Voraussetzungen, welche das System erfüllen muss, um zu funktionieren. Beispielhaft kann die funktionierende Verbindung anhand des "Random Celebration Generators" demonstriert werden. Hierfür werden 20-30 "Ways to Celebrate" in einer Datenbank hinterlegt und randomisiert abgerufen.

Es wird also gezeigt, dass die Verbindung mit der Datenbank hergestellt werden kann. Diese wird insbesondere auch für die persistente Speicherung von User-Daten benötigt. Das Fehlschlagen oder Ausfallen des "Random Celebration Generators" stellt an sich ein zu vernachlässigendes Risiko dar, da die Ausgabe theoretisch auch über ein Array im Backend erfolgen kann. Trotzdem hat sich die Gruppe für die Umsetzung dieses PoCs entschieden, um die Kommunikation mit der Datenbank grundsätzlich zu demonstrieren, ohne bereits die genaue Anwendungslogik und innere Architektur des Systems erarbeitet zu haben.

Eine weitere Idee besteht darin, dem User auch selbst die Möglichkeit zu geben, "Ways to celebrate" hinzuzufügen und das System so noch stärker zu personalisieren. Diese Option zeigt die grundsätzliche Möglichkeit und das Funktionieren des Sendens und Speicherns von Daten (im Gegensatz zum bloßen Abfragen).

PoCs

4. Regelmäßiges Senden von Notifications

- Notifications sollen von der App gesendet werden können, auch wenn diese nicht im Hintergrund läuft
 - Exit-Kriterium: Notification erscheint zum geplanten Zeitpunkt, auch bei geschlossener App
 - Fail-Kriterium: Notification erscheint nicht, regelmäßige Notification erscheint nur einmal, Notification erscheint nur bei laufender App

21

Die Gruppe hat bislang keine Erfahrung mit dem Senden von Notifications auf einem Smartphone. Mit diesem Use Case wird sichergestellt, dass dieses Wissen zur Entwicklung des Systems erworben und sicher angewendet wird. Das adressierte Risiko liegt hier also in der Kompetenz des Entwicklerteams begründet.

Die Relevanz von Notifications hängt stark davon ab, wie regelmäßig und eigenständig der User mit dem System interagiert. Die Funktionsweise des App/Smartwatch-Systems ist nicht zwingend beeinträchtigt, wenn keine Notifications möglich sind. Bei Usern, die regelmäßige Erinnerungen benötigen, stellt das Fehlen ein höheres Risiko dar. Diese könnten die Interaktion vernachlässigen oder vergessen (siehe User-Risiken-Tabelle) und somit die Auswertung der Erfolge unmöglich machen.

PoCs

5. Verbindung zwischen App und Smartwatch (in unserem Fall über Browser)

- Demonstration der Funktionsweise der Schnittstelle
 - Exit-Kriterien:
 - Widget ist bedienbar
 - Notifications erscheinen auch auf Smartwatch (im Browser), wenn sie gesendet werden
- Fail-Kriterien

22

Die Verbindung zwischen App und Smartwatch sollte für die Nutzung des Systems gewährleistet werden. Zwar ist grundsätzlich auch eine reine Interaktion über die Smartphone-App denkbar (und erweitert den Kreis potentieller Nutzer), jedoch werden die identifizierten User-Risiken wie Vernachlässigung, Motivationsnachlass und Vergessen höher eingeschätzt, als beim (zusätzlichen) Einsatz der Smartwatch. Dafür gibt es verschiedene Gründe: Die Vielzahl an Apps und Notifications, die auf dem Smartphone stattfinden. Die gute Sichtbarkeit der Smartwatch. Die Freude am technischen Gerät.

PoCs

6. Wege der Datenerhebung

- Wichtige User-Daten während der Praxis-Phase sind die Anzahl der Prompts und die Anzahl der Male, die das Verhalten ausgeführt wurde (jeweils pro Behavior-Change-Rezept). Es sind verschiedene Interaktionsmöglichkeiten denkbar. Diese sollen in PoCs durchgespielt werden.
- z.B.:
 - Notification: User erhält eine Notification mit dem Symbol, das zu einem bestimmten Rezept gehört. Er wird abgefragt, wie oft der Prompt aufgetreten ist und wie oft im Anschluss das Verhalten ausgeführt wurde (Verhältnis). -> Wie häufig soll die Abfrage stattfinden
 - Der User klickt selbständig über ein Widget auf der Smartwatch/ dem Smartphone auf das Symbol zum Rezept, sobald er dieses ausgeführt hat. Hier ließe sich die Celebration mit der Handlung verbinden. Sie wird im Anschluss an das Tippen auf das Symbol ausgeführt und durch einen Soundeffekt unterstützt. Problem: Es muss zusätzlich bestimmt werden, wie häufig der Prompt aufgetreten ist, um das Erfolgsverhältnis zu bestimmen.
- Exit-Kriterien: Es wird ein Weg oder es werden mehrere Wege gefunden, ein Verhältnis zwischen Prompt und Verhalten zu bestimmen. Die Daten sind zuverlässig
- Fail-Kriterien: Es wird kein Weg gefunden, Daten zuverlässig zu erheben

23

Die Datenerhebung ist kritisch für den Erfolg des Systems in der Practice- und Reviewphase. Es soll ermittelt werden, wie gut der User es schafft, die erstellten Verhaltens-Rezepte umzusetzen. Dafür muss einerseits die Anzahl der Prompts und andererseits die Anzahl der Male, die das Verhalten im Anschluss ausgeführt wurde erfasst werden. Von dem "Erfolgsquotienten" hängt ab, ob der User in der Practice-Phase verbleibt, die Verhaltensweisen der Analyse-Phase überarbeitet oder in der Design-Phase den Prompt verändert/ das Verhalten weiter vereinfacht/ verkleinert. Es wurden bereits Ideen der Erhebung besprochen, von denen zwei hier skizziert sind. Bei der Entwicklung des PoCs können noch weitere Alternativen einfließen und getestet werden. Es ist wichtig, dass die Daten zuverlässig genug sind, um den User passend anzuleiten.

Ein möglicher Fallback für den Fall, dass zuverlässige Daten nicht oder nur unter größeren Nachteilen erhoben werden können, wäre, die Einschätzung im Review grundsätzlich dem User zu überlassen. Dann wäre nicht der vom System erfasste Erfolgsquotient entscheidend für die Fortführung, sondern lediglich die Meinung des Users selbst.

Wichtig: Auch bei einer systematischen Erhebung und Auswertung soll dem User die

Wahl gelassen werden, wie und an welcher Stelle er sein Verhaltensrezept verändern möchte. Das System macht hier lediglich begründete Vorschläge.