C:\Users\Edwin\Google Drive\Studium\Studium\4 Semester\Software Engineering 1\Allergeek\Dokumentation\Konzeption\Corporate Design\Logo.png

Projekt Edible

Abschlussdokumentation

25.04.2015

Inhalt

[1 Intention und Motivation 3](#_Toc417755038)

[1.1 Anwendungsbereich 3](#_Toc417755039)

[1.2 Zielgruppen 3](#_Toc417755040)

[2 Komponenten 3](#_Toc417755041)

[3 Anforderungen 4](#_Toc417755042)

[4 Architektur 4](#_Toc417755043)

[4.1 Backend 4](#_Toc417755044)

[4.1.1 Datenmodell 4](#_Toc417755045)

[4.1.2 Klassendiagramme 4](#_Toc417755046)

[4.1.3 Libraries und Framworks 4](#_Toc417755047)

[4.2 Backend 4](#_Toc417755048)

[4.2.1 Klassendiagramme 4](#_Toc417755049)

[4.2.2 Libraries und Frameworks 4](#_Toc417755050)

[4.3 HMD Applikation 4](#_Toc417755051)

[4.4 Klassendiagramme 4](#_Toc417755052)

[4.5 Libraries, Frameworks und Intents 4](#_Toc417755053)

[5 Bewertung der Ergebnisse 4](#_Toc417755054)

[6 Beschreibung des Projektablaufs 4](#_Toc417755055)

[6.1 Vorgehensmodell 4](#_Toc417755056)

[6.2 Rollen 4](#_Toc417755057)

[6.3 Testkonzeption 4](#_Toc417755058)

[7 Fazit und Ausblick 4](#_Toc417755059)

# Intention und Motivation

Das Ziel des Projektes Edible ist es eine Anwendung bereit zu stellen, mit der Allergiker ein Produkt auf seine Inhaltsstoffe prüfen können. Die Realisierung der Anwendung soll auf einem Head-Mounted Display (HMD) erfolgen, welches den Produktbarcode einliest und die Inhaltsstoffe per Web-Request von einem Server zugeschickt bekommt. Durch den Abgleich einer Blacklist wird so dem Allergiker Auskunft gegeben ob er dieses Produkt konsumieren kann oder nicht.

## Anwendungsbereich

Der Hauptanwendungsbereich von Edible liegt im regelmäßigen Einkauf von Lebensmitteln der Anwender, bzw. auch vor dem Konsum eines Produktes.

## Zielgruppen

Die Hauptzielgruppe von Edible sind Allergiker die durch ihre Allergien gezwungen sind auf ihre Ernährung zu achten. Weiter kann die Anwendung von ernährungsbewussten Personen genutzt werden, die gewisse Inhaltsstoffe aus anderen Gründen meiden als Allergiker. Potenziell sind auch Personen auf Diäten eine Zielgruppe für die Anwendung, denn auch sie können durch die Anwendung gewisse Inhaltsstoffe meiden.

# Komponenten

Edible ist als Anwendung für HMDs verfügbar und wurde für eine Benutzung auf der Vuzix M100 optimiert. Für die Benutzung ist außerdem ein Account auf der Edible Website von Nöten. In Zukunft ist es vorstellbar, dass es eine Anwendung für Android-Smartphones gibt, welche die Funktionalität des HMDs und der Website übernimmt und erweitert.

# Anforderungen

Die angestrebten funktionalen Anforderungen wurden in 3 Bereiche eingeteilt „Must have“ (1), „Should have“ (2), „Nice to have“ (3). Wie in der folgenden Tabelle zu sehen ist, sind alle „Must have“ Anforderungen umgesetzt oder angepasst umgesetzt. Bis auf eine „Should have“ Anforderung wurden alle umgesetzt. Die Android-Smartphone Application wurde als „Nice to have“ formuliert und wurde zum größten Teil durch die Website in responsive Layout umgesetzt.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Funktionalität | | Priorität | Ergebnis |
| F10 | HMD App |  | |  |
| F10.1 | Das Zielsystem der HMD App ist die Vuzix M100. Die App muss ohne Einschränkungen auf der Vuzix M100 lauffähig sein. | 1 | | Umgesetzt |
| F10.2 | Um sein mobiles Endgerät mit einem Account zu verbinden, scannt der User einen Barcode, der ihm im Webbrowser angezeigt wird. Der User muss sich dafür bereits auf der Website registriert haben. | 2 | | Umgesetzt |
| F10.3 | Die App verfügt sowohl über den Standard-Sprachbefehl "go home" als auch den spezifischen Sprachbefehl "scan" zum Starten des Produktscanners. | 1 | | Angepasst |
| F10.4 | Nach dem Start der App wird der Homescreen angezeigt. Hier wird auf den Sprachbefehl zum Scannen gewartet. | 1 | | Umgesetzt |
| F10.5 | Mit dem Sprachbefehl "scan" aktiviert der User im Homescreen den Scanner. Danach kann er über die Kamera den Barcode eines Produkts scannen. | 1 | | Umgesetzt |
| F10.6 | Wurde ein Produkt eingescannt, wird dem User auf dem Infoscreen das Ergebnis angezeigt. Nach 10 Sekunden im Infoscreen kehrt die App automatisch in den Homescreen zurück. | 1 | | Angepasst |
| F20 | Accountsystem |  | |  |
| F20.1 | Der User kann einen Account erstellen, der persistent im System gespeichert wird. (siehe F30.1) | 1 | | Umgesetzt |
| F20.2 | Der User kann beliebig viele mobile Endgeräte mit seinem Account verbinden. | 2 | | Umgesetzt |
| F30 | Webapplikation |  | |  |
| F30.1 | Der User kann einen neuen Account über ein Registrierungsformular erstellen. (siehe F20.1) | 1 | | Umgesetzt |
| F30.2 | Der User kann sich mit seiner Email-Adresse und seinem Passwort in einem Login Formular anmelden. | 1 | | Umgesetzt |
| F30.3 | Der User kann Einträge aus der Inhaltsstoffdatenbank zu seiner Blacklist sowohl hinzufügen als auch wieder entfernen. | 1 | | Umgesetzt |
| F30.4 | Der User kann HMDs koppeln und entkoppeln. | 2 | | Umgesetzt |
| F30.5 | Für die Verbindung mit der HMD-App wird ein Barcode im Web-Browser angezeigt. | 2 | | Umgesetzt |
| F30.6 | Der User kann seine Email-Adresse und sein Passwort ändern. | 2 | | Umgesetzt |
| F30.7 | Registriert sich ein User oder ändert er seine Emailadresse, muss er diese bestätigen. Zum Bestätigen wird eine Bestätigungslink an die Emailadresse versandt, die durch den User bestätigt werden muss. | 2 | | Nicht umgesetzt |
| F40 | Android-Smartphone App (durch responsive Layout realisiert) | | | |
| F40.1 | Der User kann HMDs koppeln und entkoppeln. | 3 | | Teilweise umgesetzt |
| F40.2 | Der User kann sich für die Nutzung des Dienstes in der Android-Anwendung registrieren. | 3 | | Teilweise umgesetzt |
| F40.3 | Der User muss sich zur Nutzung des Dienstes in der Android-Anwendung anmelden. | 3 | | Teilweise umgesetzt |
| F40.4 | Der User muss seine Email-Adresse und sein Passwort ändern können. | 3 | | Teilweise umgesetzt |
| F40.5 | Der User kann Einträge aus der Inhaltsstoffdatenbank zu seiner Blacklist hinzufügen und wieder entfernen. | 3 | | Teilweise umgesetzt |
| F40.6 | Der User scannt einen Produktbarcode und erhält eine Information über die Verträglichkeit. | 3 | | Teilweise umgesetzt |
| F40.7 | Der User kann sich alle Inhaltsstoffe des Produktes anzeigen lassen, sowie eine Websuche nach dem Produkt starten. | 3 | | Nicht umgesetzt |

Tabelle 1: Funktionale Anforderungen

## Nicht funktionale Anforderungen

|  |  |
| --- | --- |
| ID | Funktionalität |
| N10 | Fachliche Mengen |
| N10.1 | Bei der Anzahl der Nutzer wird zu Beginn von einer Nutzeranzahl von unter 1000 Nutzern ausgegangen. Die Anwendung sollte später auch für größere Nutzeranzahlen auslegbar sein. |
| N10.2 | Jeder Nutzer hat durchschnittlich 10 Inhaltsstoffe in seiner Blacklist. |
| N20 | Usability |
| N20.1 | Die Oberfläche muss selbsterklärend und einfach sein, damit es auch von Einsteigern ohne Einweisung verwendet werden kann. |
| N20.2 | Die Funktionen der Oberfläche werden in einem Handbuch dokumentiert. |
| N20.3 | Insbesondere auf dem niedrig auflösendem Bildschirm der Vuzix M100 muss auf eine gute Lesbarkeit der Inhalte geachtet werden. Auch auf allen anderen Plattformen muss eine Lesbarkeit gegeben sein. |
| N30 | Erscheinungsbild |
| N30.1 | Die Erscheinung der Oberfläche soll in Design und Bedienungskonzept einheitlich sein. |
| N40 | Performanz |
| N40.1 | Der Server soll für die Bearbeitung einer Anfrage maximal 100ms brauchen. |
| N40.2 | Es wird von maximal 60 Anfragen pro Minute unter Volllast ausgegangen. |
| N40.3 | Da die App vorerst nicht kommerziell betrieben wird, ist eine hohe Verfügbarkeit nicht wichtig. Eine Verfügbarkeit des Webservices von 98% im Jahreszeitraum ist damit ausreichend. |
| N50 | Wartbarkeit |
| N50.1 | Für die Qualität und Einheitlichkeit des Codes wird ein Styleguide definiert, dem der produzierte Code entsprechen muss. |
| N50.2 | Der Code muss mit ausreichend Kommentaren ausgestattet sein. Für eine Automatische Dokumentationserstellung werden spezielle Kommentare erstellt, die automatisch verarbeitet werden. Diese Kommentare werden im Styleguide festgehalten. |
| N60 | Tests |
| N60.1 | Anforderungen an die Testbarkeit werden in einem separaten Testkonzept festgelegt. |

# Architektur

## Backend

### Datenmodell

### Klassendiagramme

### Libraries und Framworks

## Backend

### Klassendiagramme

### Libraries und Frameworks

## HMD Applikation

## Klassendiagramme

## Libraries, Frameworks und Intents

# Bewertung der Ergebnisse

Das Projekt ist sehr gut und besser als alle anderen

# Beschreibung des Projektablaufs

## Vorgehensmodell

Das Projekt wurde agil nach Scrum entwickelt.

## Rollen

Hendrik war für alles verantwortlich

## Testkonzeption

Mit agilem Testen wurde die Qualität des Produkts sichergestellt.

# Fazit und Ausblick