

Systemadministration Projekt
Pflichtenheft
Codename: Lila Dunst

Daniel Schneiderbeck und Ali Rasul

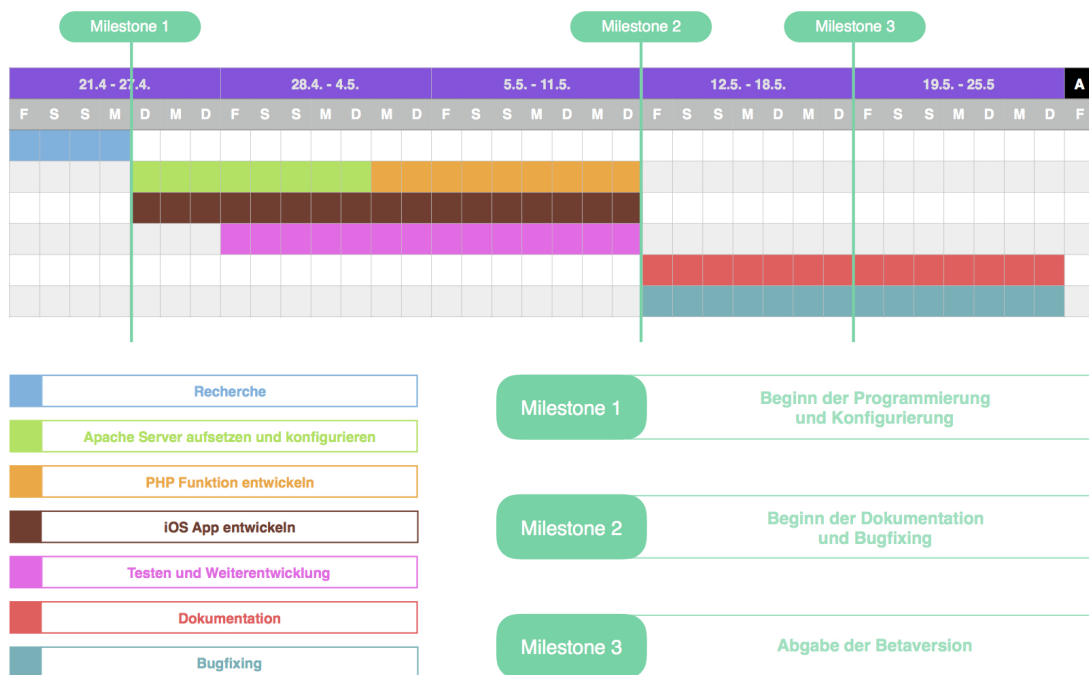
6. April 2017

1 Einleitung und Problemstellung

Mit diesem Projekt soll Nutzern die Arbeit abgenommen werden, standardisierbare Checklisten für generelle Lebensbereiche bei jedem Nutzen immer wieder neu anlegen zu müssen und sich dabei immer neu zu fragen, was diese beinhalten muss. Eine weitere Problematik für die Nutzer derzeit ist es, dass für verschiedene Listenvorlagen, verschiedene Webseiten oder Blogs besucht werden müssen.

2 Zielbestimmung

Durch dieses Projekt soll erreicht werden, dass allgemein bekannte Listen - z.B. Packlisten für Urlaube, Listen von bekannten Serien und Filmen, Einkaufslisten uvm. - zentral auf einem Server angelegt werden und von jedem Nutzer heruntergeladen und lokal benutzt werden können.



Gantt-Diagramm

2.1 Musskriterien

- Datenbank mit Templates von Listen
- Server der anfragen von App entgegennimmt und Template aus Datenbank lädt
- IOS App die Anfragen an den Server schickt und die Antwort (Template) verarbeitet
- Graphische Aufbereitung der Templates
- Abhaken der Listen
- Listen werden lokal gespeichert, somit offline bearbeitbar
- Mehrere Listen können parallel vorhanden sein
- Listenelemente sind mit sichtbarem Zeitstempel versehen
- Ausschließlich Admin kann Templates anlegen

2.2 Wunschkriterien

- User kann eigene Listen anlegen und diese als Template auf dem Server speichern
- Login für verschiedene User
- Listen können vom User, für ihn lokal, durch eigene Listenelemente erweitert werden
- Eigene Listen können lokal angelegt werden

2.3 Abgrenzungskriterien

- keine Weboberfläche, die auf Listen zugreifen lässt
- keine Teilen Funktion von Listen
- keine App für andere Plattformen
- kein Panorama Layout bei der iOS Applikation
- keine Deadline für Listenelemente

3 Produkteinsatz

Das Produkt soll bei iOS Usern im Task Management Bereich eingesetzt werden, indem allgemein bekannte Listenelemente als Listenvorlagen zur Verfügung gestellt werden.

3.1 Anwendungsbereiche

Der Nutzer soll in einer App verschiedene Listen Templates zur Auswahl angezeigt bekommen, welche als Kopie von einem Server geladen werden. Dadurch erspart er sich das zeitaufwendige Neuanlegen von Listen. Die Anwendung ist nicht für Nutzer ohne iOS Gerät. Des Weiteren ist keine Anbindung der Anwendung an eine Weboberfläche geplant und die User können Ihre Listen nicht mit anderen teilen.

3.2 Zielgruppen

Die Zielgruppe für die Anwendung sind iOS Nutzer im Alter von 12-99 Jahren. Nutzer von anderen Plattformen werden derzeit nicht berücksichtigt. Die Applikation ist somit nicht plattformunabhängig.

4 Entwicklungsumgebung

Die iOS Applikation wird in der Entwicklungsumgebung Xcode entwickelt. Hierzu wird die Programmiersprache Swift verwendet.

Es wird ein Apache-Server aufgesetzt der eine Anbindung an eine MongoDB Datenbank hat. Die serverseitige Entwicklung findet in Eclipse statt, dazu wird PHP verwendet. Die Vorteile von PHP gegenüber zu node.js sind, dass kein Neustart bei Code Änderung notwendig ist. Außerdem ist es einfach zu benutzen - keine großen Installationen nötig und es ist für einfache Aufrufe geeignet. Da wir nur einfache Anfragen serverseitig verarbeiten, ist PHP zu bevorzugen. PHP wird benutzt um die Anfragen entgegenzunehmen, eine Datenbank Query auszuführen und das Ergebnis zurückzusenden.

Eine Verschlüsselung ist hier nicht notwendig da die Listeninformationen ohnehin öffentlich zugänglich sind. Die Kommunikation zwischen Server und App findet mittels HTTP Requests statt.

5 Produktfunktionen

Hier erfolgt eine Auflistung der Funktionen aus Benutzersicht, die das gewünschte Produkt erfüllen soll.

5.1 Funktion 1

Die iOS Applikation soll ermöglichen aus einer Liste von vorhandenen Templates eine auszuwählen. Diese wird zu den ggf. lokal bereits vorhandenen Listen hinzugefügt.

5.2 Funktion 2

Die lokal in einer Liste vorhandenen Listenpunkte sollen durch visuelle Veränderungen als erledigt markiert werden können.

6 Benutzeroberfläche

Die Benutzeroberfläche ist noch im Hochformat funktionsfähig und kann nicht auf Panorama erweitert werden. Die Templates der Listen werden kategorisiert angezeigt. Innerhalb der Listen wird das als "erledigt" markieren einzelner Listenpunkte durch das Berühren dieser, mittels ausgrauen und durchstreichen des Textes, dargestellt.

7 Globale Testszenarien und Testfälle

Hier werden die Testfälle pro Funktion aus den Funktionsanforderungen abgeleitet und aufgelistet. Diese Testfälle sind für den Abnahmetest zu verwenden.

7.1 Testfall 1

Es werden auf dem Server mehrere Templates hinterlegt, diese müssen korrekt kategorisiert werden und auswählbar sein. Daraufhin wird der Inhalt des Templates auf der App angezeigt.

7.2 Testfall 2

Einzelne Listenpunkte können durch antippen als erledigt markiert werden. Durch erneutes antippen wird diese Aktion wieder rückgängig gemacht.