

Arbeitsplan

mok17

16. Januar 2017

Inhaltsverzeichnis

1	Projektvision	1
2	Voraussetzungen	1
2.1	Infrastruktur des Projektes	1
2.2	Kommunikation im Projektteam	2
2.3	Daten vom Kunden	2
3	Designübersicht und Funktionalität	2
3.1	Funktionalität	2
3.1.1	Web-Applikation	2
3.1.2	Mobile App	3
3.2	Designübersicht	3
3.2.1	Back-End	4
3.2.2	Mobile App	4
3.3	Rollen	4
3.3.1	Benutzer	4
3.3.2	Administrator	4
4	Arbeitspakete	5
4.1	Einarbeitung in die wichtigsten Tools	5
4.2	Vorprojekt	5
4.3	Erweiterung	5
4.4	Auslagerung auf den Server	5
4.5	Implementierung letzter Funktionen	6
4.6	Fertigstellung	6
5	Vorprojekt	6
6	Glossar	7
6.1	Back-End	7
6.2	Dashboard	7
6.3	Kanban	7
6.4	Scrum	7

1 Projektvision

Bei Logistik-Prozessen gibt es oft Abweichungen vom geplanten Ablauf. Um dem Disponenten und der Geschäftsleitung einen besseren Überblick zu ermöglichen, sollen eine mobile App und eine Webapplikation zur Tourüberwachung und zur Anzeige von kritischen Events entwickelt werden. Falls bestimmte, mit dem Kunden zusammen definierte, Grenzwerte von Kennzahlen überschritten werden, dann alarmiert die Anwendung den Disponenten. Desweiteren werden Tourdetails übersichtlich dargestellt. Die Realisierung erfolgt hierbei mit Drools, da so die Logik der Anwendung von der Implementierung des Systems getrennt ist. Der Administrator kann auf Nachfrage der Benutzer neue Regeln definieren und einbinden. Die mobile App soll sowohl auf iOS, als auch auf Android laufen, daher verwenden wir zur Entwicklung der mobilen App das Ionic Framework, welches es uns ermöglicht die App auf beiden Betriebssystemen anzupassen. Diese wird dann über Apache Cordova verteilt. Als Webserver und Webcontainer nutzen wir Tomcat. Dies bietet die Möglichkeit Webanwendungen auf Servlet- oder JSP-Basis auszuführen. Die verschiedenen Clients können über diesen Server dann auch mit dem Administrator kommunizieren. Außerdem generieren Tourevents auch Informationen über den aktuellen Tourstatus, welche dem Disponenten die Möglichkeit geben nachzusteuern.

2 Voraussetzungen

Zur Umsetzung der Projektvision ist es nötig verschiedene Voraussetzungen zu schaffen. Im Folgenden werden diese Punkte genannt und näher erläutert.

2.1 Infrastruktur des Projektes

Grundvoraussetzung für das Gelingen des Projektes ist der Aufbau einer guten Projektinfrastruktur. Dazu gehören die Auswahl einer leistungsfähigen Versionsverwaltung (git, Repository im GitLab des Fachbereichs Informatik der Uni Leipzig), die Einrichtung des Projektserver und der Datenbank (MariaDB) sowie die Installation von genutzten Frameworks und Paketen (Vaadin, Ionic, JUnit, node.js, Drools). Weiterhin setzt Vaadin die Installation eines Webserver voraus. Wir verwenden hierfür Apache Tomcat. Da im Projekt vor allem die Programmiersprache Java verwendet wird, muss auch sichergestellt werden, dass eine genügend aktuelle Version des Java SDK installiert ist, bzw. eventuell die Spezifikation Java EE. Außerdem sollten alle Teammitglieder Remotezugriff per SSH auf den Projektserver haben. Dieser ist bereits vorhanden.

2.2 Kommunikation im Projektteam

Entscheidend für den Projekterfolg ist auch die Kommunikation im Team. Um eine konsistente Historie über alle Vorschläge und Abstimmungen zu haben, wird die Chat-App Telegram verwendet. Hier ist ein Gruppenchat angelegt, welcher vor allem der Abstimmung von Präsenzterminen, aber auch der Diskussion zu aktuellen Anliegen dient. Weiterhin gibt es feste Termine zu denen das gesamte Projektteam zusammenkommt und die wichtigsten Punkte diskutiert werden. Ein gutes Aufgabentracking ist eine ebenso entscheidende Voraussetzung. Hierfür nutzt das Team das Issue-Board aus dem GitLab, welches ähnlich einem Scrum-Board oder einem Kanban-Board funktioniert. Änderungen von einzelnen Teammitgliedern werden in den Commit-Nachrichten im Git-Repository für alle anderen Teammitglieder ersichtlich dokumentiert.

2.3 Daten vom Kunden

Wichtig ist weiterhin ein Datensatz mit Beispieldaten, welchen das Team bereits erhalten hat. Auf Basis dieser Daten erfolgt die Entwicklung der Anwendung. Es wird sich im weiteren Projektverlauf zeigen, ob das Team noch zusätzliche Beispieldatensätze benötigt. Mit diesen Daten muss das Team den kontinuierlichen Datenfluss zur Anwendung simulieren, damit die Anwendung auf Änderungen der Kennzahlen anhand der Regeln im Drools reagieren kann.

3 Designübersicht und Funktionalität

Das Projekt beinhaltet eine Web-Applikation und eine mobile App. Die Web-Applikation wird als Verwaltungszentrale von einem Administrator/Benutzer übernommen. In der mobilen App werden Nachrichten für den Benutzer angezeigt. Falls ein Störfall eintritt, kann der Nutzer über die App benachrichtigt werden.

3.1 Funktionalität

Im folgenden Abschnitt werden kurz die vereinbarten Funktionalitäten erläutert.

3.1.1 Web-Applikation

Es muss Benutzern möglich sein, sich in dieser Web-Applikation anzumelden, um dort verschiedene Einstellungen vornehmen zu können. In der Benutzerverwaltung müssen folgende Funktionen implementiert werden:

1. Anmeldung über E-Mail und Passwort

2. Administratoren können Nutzer anlegen, löschen und bearbeiten
3. Nutzer können in Gruppen eingeteilt werden
4. Struktur- und Entwurfsprinzipien einzelner Pakete
5. Nutzer bestehen aus: Name, Gruppe, Rolle, E-Mail, Telefon, verbundene Smartphones

Desweiteren muss eine Prozessauswertung vorhanden sein. In einem Logistikprozess werden ständig Daten gesammelt und diese müssen verarbeitet werden. In diesem Projekt wird die Auswertung mittels Regeln realisiert. In der Web-Applikation sollen Kennzahlen definiert werden. Diese Kennzahlen können z.B. Anzahl abgeschlossener Touren oder vorraussichtlicher Gewinn sein. Weiterhin wird der Tour-Status in Form einer Ampel angezeigt. Im kompletten Logistikprozess kann es natürlich auch zu Störungen kommen. Diese Störfälle müssen klar ersichtlich dem Nutzer mit allen benötigten Informationen (Absendername, TourID, Transportgut, etc.) angezeigt werden. Dem Benutzer der Web-Applikation muss es außerdem noch möglich sein, Nachrichten zu senden/empfangen.

3.1.2 Mobile App

Der Benutzer der mobilen App soll Informationen zum Logistikprozess angezeigt bekommen. Im Falle einer Störung muss der Benutzer durch weitere Signale (Push-Benachrichtigung etc.) benachrichtigt werden, um sofort reagieren zu können. Ähnlich wie bei der Web-Applikation muss der Benutzer sich mit E-Mail und Passwort anmelden, aber vorher im Back-End registriert worden sein. Auf der mobilen App müssen folgende Informationen angezeigt werden:

1. Kennzahlen
2. Liste mit Touren und deren Status
3. Tour-Details
4. Liste von Störfällen
5. Liste von Nachrichten Smartphones

3.2 Designübersicht

Die Darstellung der Webseite, wie auch der mobilen App, wird in Form eines Dashboards realisiert. Hierbei wird ein konsistentes Layout verwendet, um den Nutzern einen einfachen Wechsel zwischen beiden Plattformen zu bieten und ein angenehmes Arbeiten zu gewährleisten.

3.2.1 Back-End

Beim Öffnen der Seite gelangt man zuerst auf einen Anmeldebildschirm. Nach erfolgreicher Anmeldung werden alle Informationen, welche für den jeweiligen Benutzer vorgesehen sind, angezeigt. Die Funktionen werden, wie im Punkt Funktionsübersicht angegeben, implementiert.

3.2.2 Mobile App

Bei der mobilen App wird auch zuerst ein Anmeldefenster erscheinen. Nach erfolgreicher Anmeldung wird hier ein Informationsfenster, mit allen vorher beschriebenen Informationen (siehe Funktionsübersicht), angezeigt. Dabei sind verschiedene Ansichten vorgesehen, wie beispielsweise Störfälle und Tour-Details. Des Weiteren wird bei wichtigen Nachrichten eine besondere Mitteilung verlangt, d.h. ein anderer Signalton, Push-Benachrichtigung, etc.

3.3 Rollen

Die Hauptunterscheidung zwischen Benutzern geschieht mit Hilfe der Gruppen. Diese Gruppen legen fest, wie die Darstellung in der Web-Applikation und der mobilen App gestaltet wird. Dennoch gibt es zwei klassische Rollen.

3.3.1 Benutzer

Back-End:

1. Anmeldung
2. Regelverwaltung
3. Einstellung zur Datenanbindung an das Logistik-System
4. Überwachung des Systems
5. Prozessauswertung

Mobile App:

1. Anmeldung
2. Anzeigen von Informationen
3. Benachrichtigung bei Störfall (Push-Benachrichtigung/Alarmfunktion)

3.3.2 Administrator

Der Administrator kann zusätzlich neue Benutzer anlegen, löschen und bearbeiten.

4 Arbeitspakete

4.1 Einarbeitung in die wichtigsten Tools

Jedes Teammitglied arbeitet sich selbstständig in alle wichtigen Tools ein, sodass alle die selben technischen Grundvoraussetzungen besitzen und diese auch grundlegend verstehen. Dazu gehört das Git (wichtige Versions-/änderungsbefehle), Apache Tomcat, Vaadin, Ionic und das Wissen, wie man auf den SQL-Server zugreift. Wenn das geschehen ist, sind alle Grundvoraussetzungen geklärt und man kann mit dem eigentlichen Projekt beginnen.

Arbeitsaufwand: 5%

4.2 Vorprojekt

Das Vorprojekt soll einen ersten Eindruck vom Projekt vermitteln. Dazu wird ein Aspekt des Projektes ausgewählt, in dem wichtige Teile der Funktionen des Endprojektes bereits integriert sind.

Der Kern des Vorprojektes ist die GUI. Alle Funktionen, die später einmal im abgeschlossenen Projekt vorhanden sein sollen, müssen im Vorprojekt bereits optisch erkennbar gemacht werden. Falls bestimmte Funktionen Daten aus der Datenbank brauchen, so werden diese simuliert, da zu diesem Zeitpunkt noch keine Datenbankanbindung vorhanden ist.

Arbeitsaufwand: 20%

4.3 Erweiterung

Dieser Abschnitt beschäftigt sich mit dem Ausbau des Vorprojektes. Weitere Funktionen werden implementiert und auf dem Server werden alle nötigen Komponenten etc. installiert, sodass für den nächsten Schritt der Server voll funktionsfähig ist.

Arbeitsaufwand: 20%

4.4 Auslagerung auf den Server

In diesem Abschnitt soll der Code auf dem Server getestet werden und alle nötigen Anbindungen (vorallem Datenbank und App) hergestellt werden. Damit müssen in diesem Abschnitt auch alle Funktionen von der Simulation der Daten auf die echte Anbindung an die Datenbank/App verändert werden.

Arbeitsaufwand: 15%

4.5 Implementierung letzter Funktionen

In diesem Arbeitspaket werden alle letzten noch fehlenden Funktionen eingebaut, grobe Fehler behoben und das Projekt zum Laufen gebracht. D.h. hier müssen bereits alle Funktionen das gewünschte Ergebnis liefern.

Arbeitsaufwand: 25%

4.6 Fertigstellung

In diesem Arbeitsschritt erhält das Projekt den letzten Feinschliff. Alle noch offenen bekannten Fehler werden behoben. Die Grundfunktionen werden dadurch aber nicht mehr verändert. Am Ende wird das Projekt in einen Zustand versetzt, indem es ausgeliefert werden kann.

Arbeitsaufwand: 15%

5 Vorprojekt

Im Vorprojekt liegt das Hauptaugenmerk auf der Erstellung einer Benutzeroberfläche (GUI). Es wird zwei verschiedene Oberflächen geben. Die erste Oberfläche wird die Admin-Gui. Diese hat die Aufgabe der Benutzerverwaltung, Regeldefinition und der Regelung des Informationsflusses. Die zweite Oberfläche wird die mobile App sein. Hierbei handelt es sich um eine Oberfläche, welche alle wichtigen Fakten übersichtlich für den jeweiligen Benutzer darstellen soll. Außerdem wird es hier auch möglich sein Nachrichten zu empfangen. Da im Vorprojekt nur die grafische Seite unseres Projektes erledigt wird, können wir noch nicht auf Echtzeitdaten/Prozesse aus der Datenbank zugreifen. Aus diesem Grund werden wir unsere Anwendung mit Testdaten befüllen, um somit die Benutzeroberflächen zu testen bzw. zu präsentieren.

6 Glossar

6.1 Back-End

Die Begriffe Front-End und Back-End (von englisch für Vor- bzw. Über- und Unterbau, wörtlich vorderes und hinteres Ende) werden in der Informationstechnik an verschiedenen Stellen in Verbindung mit einer Schichteneinteilung verwendet. Dabei ist typischerweise das Front-End näher am Benutzer, das Back-End näher am System.

6.2 Dashboard

Auf einem Dashboard lassen sich Informationen übersichtlich darstellen.

6.3 Kanban

Kanban ist ein Vorgehensmodell zur Softwareentwicklung, bei dem die Anzahl paralleler Arbeiten, der Work in Progress, begrenzt und somit kürzere Durchlaufzeiten erreicht und Probleme – insbesondere Engpässe – schnell sichtbar gemacht werden sollen.

6.4 Scrum

Scrum ist die Bezeichnung für ein Vorgehensmodell des Projekt- und Produktmanagements, insbesondere zur agilen Softwareentwicklung.