|  |
| --- |
| Elmer Lukas, Heidt Christina, Steiner Diego, Treichler Delia, Waltenspül Remo  17. März 2011 |

|  |
| --- |
| SE2 Projekt MRT |
| Anforderungsspezifikation |
|  |

****

# Dokumentinformationen

## Änderungsgeschichte

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Version | Änderung | Autor |
| 16.03.2011 | 1.0 | Erste Version des Dokuments | SD |

## Inhaltsverzeichnis

[1 Dokumentinformationen 1](#_Toc288125761)

[1.1 Änderungsgeschichte 1](#_Toc288125762)

[1.2 Inhaltsverzeichnis 1](#_Toc288125763)

[1.3 Einführung 1](#_Toc288125764)

[1.4 Allgemeine Beschreibung 1](#_Toc288125765)

[1.5 Spezifische Anforderungen 1](#_Toc288125766)

[1.5.1 Funktionalität 1](#_Toc288125767)

[1.5.2 Zuverlässigkeit 2](#_Toc288125768)

[1.5.3 Benutzbarkeit 2](#_Toc288125769)

[1.5.4 Effizienz 2](#_Toc288125770)

[1.5.5 Änderbarkeit 2](#_Toc288125771)

[1.5.6 Übertragbarkeit 2](#_Toc288125772)

[1.5.7 Schnittstellen 4](#_Toc288125773)

[1.5.8 Lizenzen 4](#_Toc288125774)

[1.5.9 Standarts 4](#_Toc288125775)

## Einführung

## Allgemeine Beschreibung

## Spezifische Anforderungen

### Funktionalität

#### Use Cases

### Zuverlässigkeit

Da es sich beim Rapport der Arbeitsstunden um einen Businesskritischen Prozess handelt, sind die Benutzer darauf angewiesen, dass die Applikation stets funktioniert.

#### Erreichbarkeit des Servers

Mit der Plazierung des Serverteils bei einem professionellen Hoster können wir eine Erreichbarkeit von über 99% sicherstellen. Sollte dieser Fall dann trotzdem mal der Fall sein, wird auf dem Android-Client ein lokales Caching eingesetzt, damit garantiert alle rapportierten Stunden aufgezeichnet werden.

### Benutzbarkeit

Wie in der Vision beschrieben ist die einfache Bedienung einses unserer Hauptziele. Die Aussendienstmitarbeiter (Wir gehen von grobmotorischen Handwerkern mit beschränktem Technikverständnis aus) sollen ihre Stunden mit möglichst wenig Aufwand rapportieren können.

Der Android-Client wird wie eine Stoppuhr bedienbar sein. Beim Aufnehmen der Arbeit drückt der Mitarbeiter den Start-Knopf, beim beenden von eben Diesen betätigt er wiederum den Stopp-Knopf und wählt abschliessend noch den Kunden aus, für den er die Arbeit verrichtet hat.

Der Server- und Reportingteil wird übersichtlich gestaltet und haltet sich an die gängigen Webkonventionen der Bedienbarkeit.

### Effizienz

#### Ladezeiten

Bitte Zeiten überprüfen

Auf dem Client sollten keine spürbaren Ladezeiten auftreten, da alles sehr simpel und Lokal gemacht wird. Die Übermittlung eines abgeschlossenen Rapports sollte nicht mehr als 10 Sekundenk in Anspruch nehmen. Der Serverteil sollte in einem modernen Browser nicht länger als 2 Sekunden laden. Bei der Generierung eines Reports müssen jedoch Wartezeiten bis 10 Sekunden in Anspruch genommen werden.

### Änderbarkeit

Wie im Projektplan schon festgelegt, wird MRT mit dem Abschluss dieses Projektes noch nicht fertig sein. Darum legen wir die Software so aus, dass sie leicht Erweitert werden kann.

Da MRT auch in echten Unternehmen zum Einsatz kommen soll, sorgen wir dafür dass der Code übersichtlich, wartbar und gut dokumentiert ist.

### Übertragbarkeit

#### Plattformen

Der Serverteil wird auf jedem Server mit einer korrekten Rails-Setup (inkl. Version) lauffähig sein und ist insofern Plattform unabhängig.

Für den Client beschränken wir uns auf die Androidversion 2.2. Derzeit sind keine Versionen für iOS (iPhone) oder WebOS (Palm) geplant.

#### Internationalisierung

Derzeit sind keine weiteren Sprachen geplant.

### Schnittstellen

##### Userschnittstellen

Der Benutzer bedient die Clientapplikation mit den Fingern über den Touchscreen und die Serverapptlikation mit Maus und Keyboard über den Browser.

#### Netzwerkschnittstellen

Luke?!?

Der Server stellt eine Netwerkschnittstelle über eine RESTFUL API zur Verfügung.

### Lizenzen

Es werden keine Lizenen benötigt.

### Standarts

* HTTP
* Android API Version 10?!?