Antrag zum Abschlussprojekt

1. Projektbezeichnung

Das Projekt wird unter dem Namen: „Verbesserung des Error-Monitor-Systems“ geführt.

* 1. Kurzform der Aufgabenerstellung

Mit dem Monitor-System kann man die Status von mehreren Servern ansehen, mit den Main-System(Kontakt-System) verbunden und die Daten ausgetauscht werden. Außerdem verwalten die auf dem Server aktuell laufenden Programme effizient, indem die Logdateien dieser Programme ausgelesen und kontrolliert werden. Damit der Benutzer mit dem System effizienter arbeiten kann, muss das System in Funktionalität und Gestaltung verbessert werden.

* 1. Ist-Analyse

Derzeit sind alle Funktionen des Monitor-Systems, die dem Benutzer zur Verfügung gestellt werden, nur in einem Screen unübersichtlich angeordnet, so dass dieses Design bei dem Benutzer Verwirrung verursachen muss. Außerdem werden in dem System mehrere Anfragen pro Sekunde an den Server geschickt, um die Daten zu holen, die im System genutzt werden.

1. Zielsetzung entwickeln/ Soll-Konzept
   1. Was soll am Ende des Projektes erreicht sein?

Mit dem Monitor-System soll dem Benutzer klarer werden, welcher Server oder welches Programm momentan ein Problem hat. Dafür muss das Design verbessern und ein Technologieupdate vorgenommen werden, indem ein Websocket verwendet werden, damit die Daten von dem Server in Echtzeit und mit geringerer Systemlast geholt werden können.

Nach der Implementierung aller Anforderungen müssen es weiterhin überprüft werden, ob sowohl das Design als auch die Funktionen ohne Probleme aufgebaut wurde.

* 1. Welche Anforderungen müssen erfüllt sein?

• Verringern des Datenverkehrs und damit weniger Last

• Verbesserung des Designs

1. Projektstrukturplan entwickeln
   1. Aufgaben auslisten

• Analyse

* die Struktur des Monitor-Systems
* Kommunikation zwischen dem System und RESTful Web-services(Mule)
* Fehleranalyse

• Entwurf

* Entwurf der Schnittstelle zum Monitor-System
* Erstellung eines Komponentendiagramms

• Implementierung

* Implementierung der Monitor-System-Schnittstelle
* Implementierung des Websocket-Servers
* Implementierung der Methoden von RESTful Web-services(Mule)

• Test

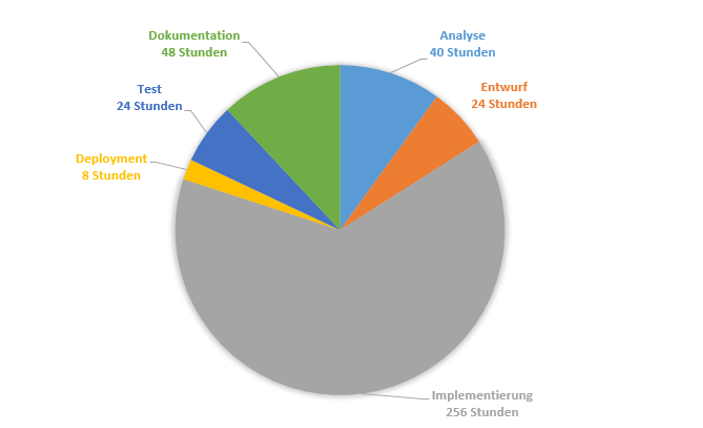
* Komponententest
* Integrationstest
* Systemtest

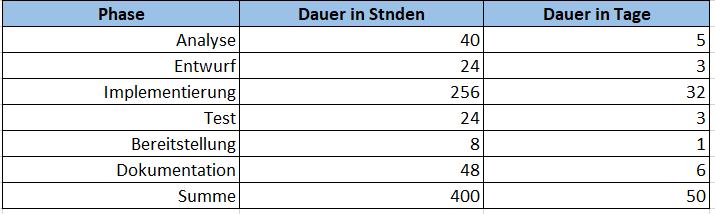
• Bereitstellung

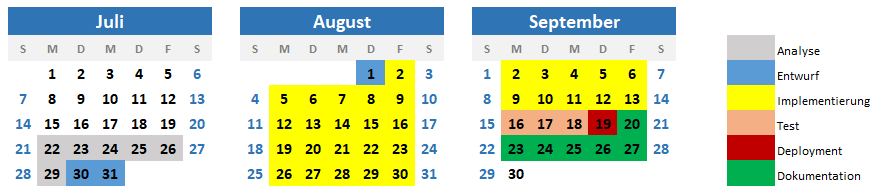
* Hochladen der Quellcode auf Git-Repository
* Installieren von Angular 7/ Node.js

• Dokumentation

* Erstellung der Projektdokumentation
* Erstellung der Entwicklerdokumentation
  1. Grafische und tabellarische Darstellung







1. Projektphasen mit Zeitplanung in Stunden

|  |  |
| --- | --- |
| **Analyse** | **40h** |
| • die Struktur des Monitor-Systems | 30h |
| • Kommunikation zwischen dem System und RESTful Web-services(Mule) | 5h |
| • Fehlerbehandlung | 5h |
| **Entwurf** | **24h** |
| • Schnittstelle zum Monitor-System entwerfen | 20h |
| • Erstellung eines Komponentendiagramms | 4h |
| **Implementierung** | **256h** |
| • Implementierung der Schnittstelle für Monitor-System | 80h |
| • Implementierung von Websocket Server | 80h |
| • Erstellen der Methoden für Websocket in RESTful Web-services(Mule) | 50h |
| • Integration der Schnittstelle in Kontaktsystem | 46h |
| **Test** | **24h** |
| • Komponententest | 8h |
| -Komponenten individuell und unabhängig prüfen |  |
| • Integrationstest | 8h |
| -Schnittstellen der Komponenten prüfen | (4h) |
| -Zusammenspiel der Komponenten überprüfen | (4h) |
| •Systemtest | 8h |
| -Prüfen gegen funktionale und nicht funktionale Benutzer-Anforderungen |  |
| **Bereitstellung** | **8h** |
| • Hochladen der Quellcode auf Git-Repository | 7h |
| • Installieren von Angular7, Node.js und Websocket usw. | 1h |
| **Erstellen der Dokumentation** | **48h** |
| • Erstellung der Projektdokumentation | 24h |
| • Erstellung der Entwicklerdokumentation | 24h |