**FH JOANNEUM - University of Applied Sciences**

**EBUSA**

**AIM18**

**Projektarbeit**

**Im Rahmen der Lehrveranstaltung EBUSA**

**SS 2019**

**Autor:**

**AIM18**

**Graz, 2019**



Inhaltsverzeichnis

[1. Projekt Ausgangssituation 1](#_Toc11122873)

[1.1. Aufgabenstellungen 1](#_Toc11122874)

[1.2. Projektziele 2](#_Toc11122875)

[1.3. Projektteam 3](#_Toc11122876)

[1.4. Meilensteinplan 3](#_Toc11122877)

[1.5. Erwartetet Projektstand AIM17 4](#_Toc11122878)

[2. Projektdurchführung 5](#_Toc11122879)

[2.1. Adaptiertes Projektziel 5](#_Toc11122880)

[2.2. Geplante Tasks 5](#_Toc11122881)

[2.2.1. Team Design 5](#_Toc11122882)

[2.2.2. Team Dev 5](#_Toc11122883)

[2.3. Erledigte Tasks 6](#_Toc11122884)

[2.3.1. Team Design 6](#_Toc11122885)

[2.3.2. Team Dev 7](#_Toc11122886)

[3. Projektendstatus 8](#_Toc11122887)

# Projekt Ausgangssituation

In diesem Kapitel befindet sich die Ausgangsituation mit den Aufgabenstellungen des gesamten Projektes. Weiters wird die Teamzusammenstellung und die Ziele aufgelistet.

Grundsätzliche Aufgabe war es S-BPM Prozesse möglichst simpel und servicebasiert ausführen zu können. Hier wurde das Projekt IPPR2016 ins Leben gerufen und es wurde eine Modelling und Execution Plattform für S-BPM Modelle entwickelt. Darauf aufbauend sollte mit dem EBUSA Projekt des Jahrganges AIM17 ein Prozessmarketplace entstehen, in dem User konkrete Instanzen der Prozesse kaufen oder verkaufen können. Ab diesen Zeitpunkt haben wir das Projekt fortgesetzt und weiterentwickelt. Das Projekt von AIM17 ist auf Microservices aufgebaut. Diese Architektur soll in die Cloud transferiert werden, um dynamische Ressourcen zu nützen und Prozessinstanzen noch besser zugänglich zu machen.

## Aufgabenstellungen

Die Aufgabenstellungen die wir abzuarbeiten bekommen haben, sind hier noch einmal abgebildet. Die Aufgabenstellungen bestehen aus zwei Bereichen, wie unten angeführt:

**Aufgabenstellung 1:**

Design & Weiterentwicklung eines auf Microservices basierenden verteilten WfMS im Kontext einer e-Business Plattform (BPMaaS)

* Analyse der bestehenden Anwendung (basierend auf Microservices) und Korrektur der funktionalen Defizite; Anpassung der Ausführungsengine an geänderte Inputformate (basierend auf einer S-BPM Ontologie); Verbesserung der Usability aller Komponenten (Ease of Use)
* Design einer Referenzarchitektur für WfMS nach den Vorschlägen der WfMC; alle Architekturmodelle müssen als Archimate Modelle vorliegen (Signavio); analog, Dokumentation der IST-Architektur der vorliegenden Microservice Architektur; Unterschiede sind aufzuzeigen

**Aufgabenstellung 2:**

Entwicklung eines Architekturkonzeptes für ein WfMS basierend auf Cloud Design Patterns (neue Referenzarchitektur) für z.B. verteilte Ausführung, Elastizität und Skalierung (twelve-factor app design principles; https://12factor.net/); Vergleich mit dem auf Microservices basierenden WfMS und Entwicklung einer Roadmap für eine Umsetzung

Umsetzung: Prozesschoreographie (Prozess A und B); Prozess A läuft auf Cloud 1, Prozess B läuft auf Cloud 2

Entwicklung von Anwendungsszenarien und möglichen weiteren Fragestellungen (IPPR und Masterarbeit)

z.B. IoT, Industrie 4.0, Cyber Physical Systems, Subjekt = smart, Netzwerke von Subjekten = Lernen und Optimieren (Machine Learning)

Architekturemodelle als Ontologie: was kann man damit machen?

## Projektziele

Unter diesem Punkt befinden sich die Projektziele, die in dem Projektantrag festgehalten wurden. Die Projektziele wurden im ersten Zusammentreffen (siehe Meilenstein) besprochen und mit den Auftraggebern festgelegt.

|  |  |
| --- | --- |
| Analyse des Projektes von AIM17 | * Dokumentierte Interfaces / Klassen / Problemen und Verbesserungspotentiale |
| Finden / Lösen von Fehlern und funktionalen Defiziten | * Fehler und Lösungsdokumentation |
| Lösen von Security Issues in AIM17 | * Keine Vulnerability issues in Github |
| Anpassung der Ausführungsengine an neue Ontologie Fileformate | * Engine funktioniert mit neuem Ontologie Format |
| Design einer Referenzarchitektur der WfmC und der Architektur von AIM17 | * Abgenommene Architekturen |
| Vergleich der Architekturen von WfmC und AIM17 | * Dokumentierte Unterschiede |
| Entwickeln von Lösungsideen für „Business Process Management in the Cloud“ basierend auf den Cloud Patterns mit den Schwerpunkten Skalierbarkeit, simultane Ausführung | * Abgenommene Lösungsideen mit Priorisierung |
| Evaluierung der unterschiedlichen Cloud Anbieter bezüglich der möglichen Umsetzung | * Dokumentierte Entscheidung, informieren der Auftraggeber über Kosten |
| Entwicklung einer Roadmap sowie die Umsetzung der erstellten Ideen | * Finalisierte Roadmap |
| Entwicklung von Konzeptideen für mögliche Anschlussprojekte und mögliche weitere Fragestellungen | * Abschlusspräsentation mit entwickelten Ideen |

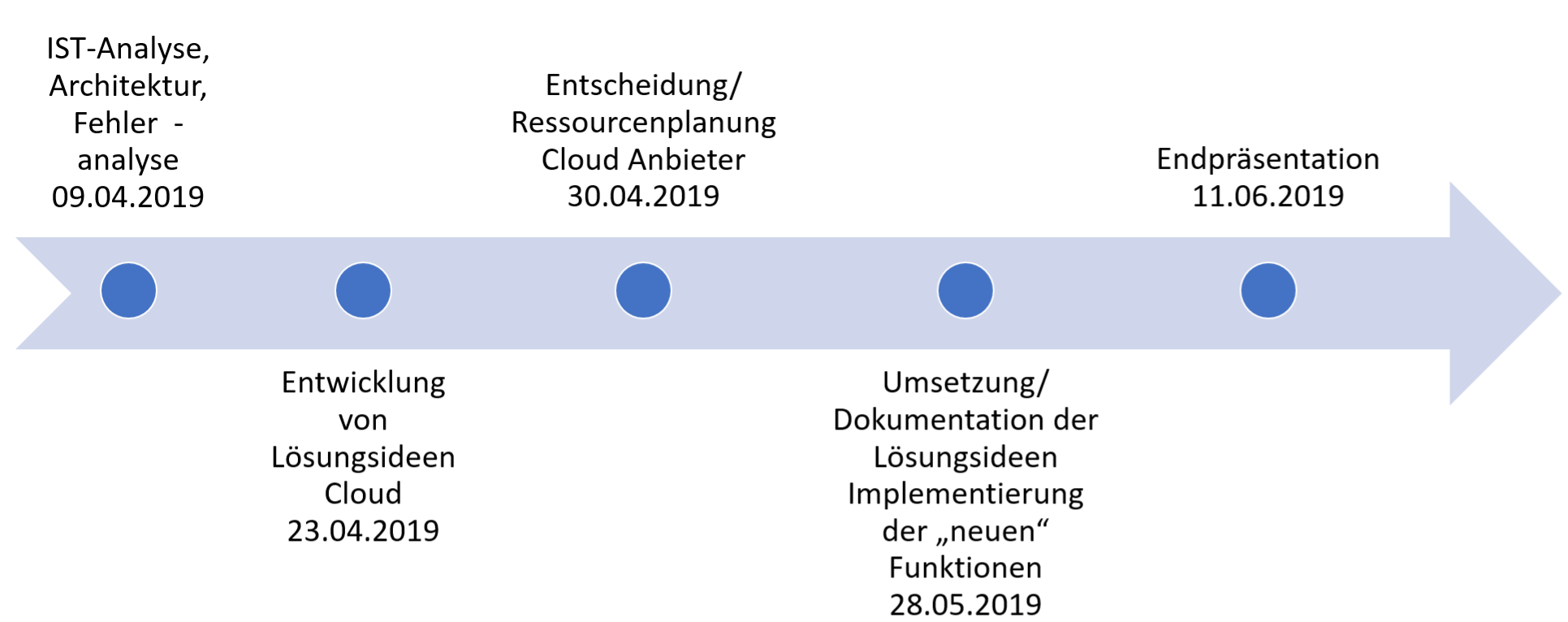
## Projektteam

Das Projektteam besteht aus dem Jahrgang AIM18. Hier haben sich drei Teams entwickelt. In der Projektleitung haben sich Einfalt Armin und Hribernig Nikolaus die Arbeiten aufgeteilt. Für die Design Themen und Developer Themen haben sich zwei Teams gebildet und die Teamleiter Reinprecht Florian (Design) und Melcher Domenik (Dev).

|  |  |
| --- | --- |
| **Projektleitung** | **Projektleitung** |
| Einfalt Armin | Hribernig Nikolaus |
|  |  |
| **TEAM DESIGN** | **TEAM DEV** |
| TL: Florian Reinprecht | TL: Domenik Melcher |
| Labinot Jashanica | Alexander Fink |
| Yannick Collasius | Cornelia Lahnsteiner |
| Kristian Ndou | Claudia Perus |
| Elisabeth Fellner | Agnesa Haxha |
|  | Andreas Krejan |

## Meilensteinplan

Der Meilensteinplan wurde zu Projektbeginn entwickelt, um die Zwischenschritte und die Endpräsentation einhalten zu können. Im Zuge der IST Analyse wurde der Meilensteinplan mit den Auftraggebern besprochen und bestätigt.



## Erwartetet Projektstand AIM17

Bei Übernahme des Projektes wurde sofort mit einer IST- Analyse angefangen. Dabei sind wir auf einen Projektstatus gestoßen, den wir so nicht erwartet haben.

Aufgrund von Gesprächen mit den Vortragenden, sind wir war die Annahme, dass in dem System S-BPM Prozesse als OWL Files hochgeladen & angezeigt werden können und eine funktionierende Frontendlösung für Mikroservices gegeben ist. Diese Funktionen wurden noch nicht bis zur Hälfte implementiert (siehe Abbildung 1).

Andere Punkte wurden nur andeutungsweise implementiert. Dazu zählen das Rollenkonzepte laut Businessplan von AIM17, implementierte Organisationskonzepte, Verkäufe von Prozessen, die in der Datenbank gespeichert​ werden und die Umsetzung der kompletten Anwendung in Docker Containern​.

Die Prozessausführung war überhaupt nicht möglich. Zwar waren die nötigen Komponenten vorhanden, allerdings nicht mit dem Frontend verbunden. Die komplette Anbindung der Process Engine an den von AIM17 erstellten Process store war notwendig.

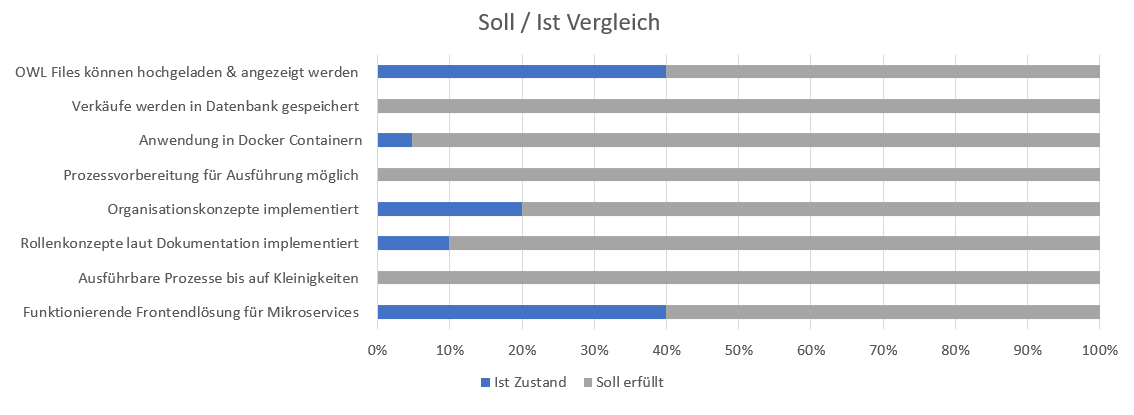


Abbildung 1: Soll / Ist Vergleich nach Ist Analyse des Projektes

# Projektdurchführung

Aufgrund der in Kapitel 1.5 beschriebenen Problemstellungen mussten die Projektziele in Abstimmung mit den Stakeholdern abgeändert werden. Da sich die Meilensteine an sich nicht geändert haben, aber die darin enthaltenen Funktionalitäten wurde der Meilensteinplan beibehalten.

## Adaptiertes Projektziel

Die in Punkt 1.2 beschriebenen Projektziele wurden verworfen und ein neues Ziel definiert. Dieses Ziel teilt sich auf in die Hauptpunkte Prozessausführung, Architektur und Dokumentation. Es wurde definiert das:

* Ein S-BPM Beispielprozess wie Urlaubsbeantragung in der Prozess Engine ausführbar sein muss
* Der User muss in der Prozessausführung Input eingeben können, wobei nicht beliebig viele Inputs nötig sind, sondern nur Standard Textinput
* Die Darstellung der Architektur mit IST und SOLL Analyse in ArchiMate
* Dokumentation über die Vorgehensweise bei der ArchiMate Modellierung
* Die Ausführung der Applikation in einer Cloud Umgebung

Bis Ende der IST Analyse konnte die Process Engine nicht getestet werden. Deshalb wurde die Vereinbarung getroffen, dass die Ziele nur erfüllt werden können, wenn die Process Engine wie geplant funktioniert.

## Geplante Tasks

Die Benötigten Tasks wurden auf die jeweiligen Teams aufgeteilt.

### Team Design

* Erstellung der Dokumentation Ist Stand
* Erstellung der Dokumentation Soll Stand
* Dokumentation Archimate
* Ist – Analyse Archimate
* Dockerize Ist-Systeme
* Cloud-Analyse: Azure, Google, AWS, Digital Ocean
* Entscheidungsfindung Cloud Plattform

### Team Dev

Die Tasks des Entwicklungsteam wurden in User Stories unterteilt. Wie in der Softwareentwicklung üblich, wurden diese in englischer Sprache formuliert. Für Dokumentationszwecke wurden diese User Stories geclustert.

Process Execution User Stories:

* As a user that has the right to configure processes, I want to link the appropriate user groups to a purchased process.
* As a user that interacts with a process, I want to have all necessary interfaces to do so.
* As a process owner, I would like to have a functionality that accounts the costs of a workflow, during execution
* As a User I want to have ProcessRoles which are connected to a Subject

Process Store User Stories:

* As an admin user, I want to get the possiblity to create and modify roles from the side menu/dashboard.
* As a user that approves the workflows, I would like to have the possibility to send the process owner instructions for changes in case it needs to be reworked before approval
* As a user, I want to see a preview picture of a process.
* As process owner, I would like to release my workflow to a specific admin user to be approved
* As a user I would like to delete processes
* As an admin user, I would like that the approval list always shows a list of unapproved processes
* As a logged in user, I want to have the possibility to view my organisation from the side menu/dashboard
* As an admin user, I want to add roles to an organisation

Administrative User Stories

* As a developer I want to move the changes from the GUI-dev to the GUI in order to build jar files

## Erledigte Tasks

Aufgrund der Zeiteinschränkungen konnten nicht alle Tasks durchgeführt werden. Die Tasks wurden in gemeinschaftlichen Diskussionen priorisiert. Dieser Punkt beschreibt die erledigten Tasks.

### Team Design

Als Cloud Plattform wurde Azure gewählt. Die Cloud Analyse wurde durch geführt und Vor- und Nachteile jedes Cloud Anbieters sowie die Kosten berechnet. Als möglicher Cloud Anbieter wurde Digital Ocean gestrichen. Alle anderen geplanten Tasks wurden durchgeführt. Die Interpretation der ArchiMate Modelle, die Cloud Analyse und die Beschreibung der Architekturen sind jeweils in anderen Dokumenten zu finden.

### Team Dev

Die implementierten User Stories für das Team Dev waren:

* As a user that has the right to configure processes, I want to link the appropriate user groups to a purchased processes.
  + Mit der Einschränkung, dass das konfigurieren nicht auf Rechte begrenzt ist
* As a user that interacts with a process, I want to have all necessary interfaces to do so.
* As a process owner, I would like to have a functionality that accounts the costs of a workflow, during execution
* As a User I want to have ProcessRoles which are connected to a Subject
* As an admin user, I want to get the possiblity to create and modify roles from the side menu/dashboard.
* As a user that approves the workflows, I would like to have the possibility to send the process owner instructions for changes in case it needs to be reworked before approval
* As process owner, I would like to release my workflow to a specific admin user to be approved
* As an admin user, I would like that the approval list always shows a list of unapproved processes
* As a logged in user, I want to have the possibility to view my organisation from the side menu/dashboard
* As an admin user, I want to add roles to an organisation
  + Nicht auf Admin user beschränkt

Diese User Stories wurden implementiert, und von den Teammitgliedern grob abgetestet. Da testing an sich nicht Teil der Zielsetzung war, wurde auch kein besonderer Wert daraufgelegt. Testing fand durch verwenden der Applikation statt.

# Projektendstatus

Die im Punkt 2.3 beschriebenen Punkte wurden implementiert und sind funktionstüchtig. Aufgrund von fehlenden Ressourcen konnte aber nicht alles umgesetzt werden. Deswegen wurde eine Liste angefertigt mit Funktionalitäten, die noch implementiert werden müssen beziehungsweise bekannte Fehler, die nicht mehr gelöst wurden.

Wichtige Punkte die im Rahmen des Projektes nicht gelöst werden konnten:

* Dependencies: Die Dependencies sind teilweise sehr alt und stellen zum Teil auch ein Sicherheitsrisiko dar. Dies wird in Zukunft zu Problemen führen. Diese Dependencies sollten dringend erneuert werden.
* Security: Auch auf Security konnte kein spezieller Fokus gelegt werden.
* Error Handling / Success Handling: Diese Punkte müssen erst implementiert werden. Nur beim Login/Logout gibt es Fehlermeldungen.
* Configuration der Prozesse nicht Rollenbasiert
* Konvertieren der Processversion in Process store Schema von „long to“ zu „int“um das If in ImportProcessModel.component.ts zu vermeiden
* Reload des Dashboards ist teilweise notwendig, damit Änderungen sichtbar werden
* Aufgrund der unterschiedlichen Vorprojekten gibt es Deutsche und englische Teile

Folgende User Stories konnten nicht fertig gestellt werden:

* As a user I want to see the correct preview picture of the process in the process details
* As a user I want to delete a process
* As a Process Creator, I want to set the pay as you go ratio
  + Wert bis jetzt hardcoded im Frontend

Auch eine Testing Strategie sollte angedacht werden. Eine Möglichkeit dies zu tun wäre karmam, protractor und jasmin.configs, diese liegen in der Dev Gui, daher könnte man automatisierte Tests über Browserstack sogar andenken – im ersten Schritt wären jedoch Junit tests in Spring zu empfehlen.