**Telemetric Gateway Monitoring IBM CIC**

Maciej Dzialoszynski, Sebastian Grünewald, David Jovanovic, Jordi Rieder



Sprint 1

Sprintdokumentation

06.03. – 3.04.2019

Inhaltsverzeichnis

[Änderungsverzeichnis 2](#_Toc4775924)

[1 Sprintbericht 3](#_Toc4775925)

[1.1 4390 IBM Cloud 3](#_Toc4775926)

[1.2 4391 Frontend – Wireframes 3](#_Toc4775927)

[1.3 4392 Backend – Setup 3](#_Toc4775928)

[1.4 4393 Backend – GW – Cloud 3](#_Toc4775929)

[1.5 4413 Frontend - Setup 3](#_Toc4775930)

[1.6 Sprint Burndown Chart 7](#_Toc4775931)

[1.7 Product Burndown Chart 7](#_Toc4775932)

[1.8 Produktivität 7](#_Toc4775933)

# Änderungsverzeichnis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Autor | QS | Datum | Status |
| 1.0 | David Jovanovic |  | 29.03.2019 | erledigt |

# Sprintbericht

Wir haben unser Projekt in grobe Userstorys eingeteilt, welche dann aus mehreren Subitems aufgebaut sind.

## 4390 IBM Cloud

Hier sollen grundlegende Konfigurationen auf der IBM-Cloud durchgeführt werden, welche notwendig sind, um das Projekt zu realisieren. Es wurde mit dem Auftraggeber eine Softwarearchitektur vereinbart und es werden die Verknüpfungen zwischen den Microservices hergestellt.

## 4391 Frontend – Wireframes

Es sollen Wireframes erstellt werden, welche die GUI der Software gut darstellen soll. Die Bedienung soll intuitiv möglich sein.

## 4392 Backend – Setup

Hier müssen grundlegende Strukturen aufgebaut werden. Es müssen alle notwendigen Module installiert werden. Weiters soll es durch diese Userstory, den Entwicklern ermöglicht werden, direkt mit der Implementierung zu beginnen.

## 4393 Backend – GW – Cloud

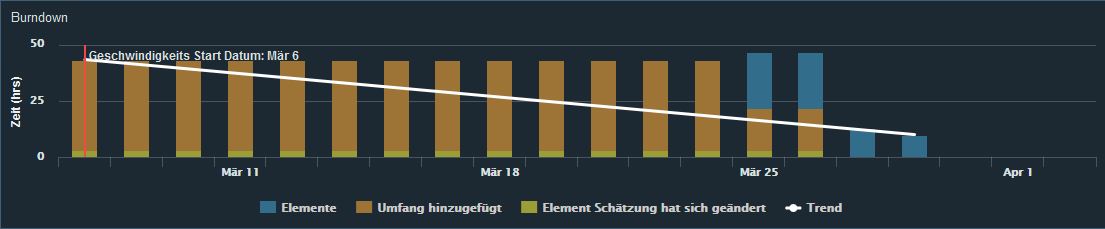
Es soll möglich sein Daten vom Gateway auf die Cloud zu übertragen, um dann in weiteren Schritten damit arbeiten zu können.

## 4413 Frontend – Setup

Hier müssen grundlegende Strukturen aufgebaut werden. Es müssen alle notwendigen Module installiert werden. Weiters soll es durch diese Userstory, den Entwicklern ermöglicht werden, direkt mit der Implementierung zu beginnen.

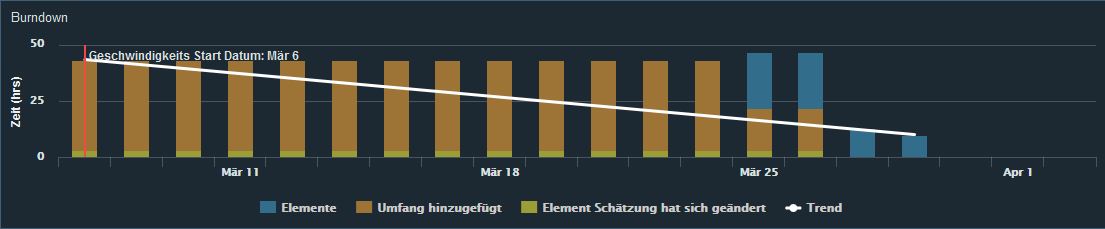
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Name** | **Zugewiesen** | **Schätzung** | **Status** | **Akzeptanzkriterium** |
| **4390** | **IBM Cloud** |  | **3.5 hrs** | **In Progress** |  |
| 4394 | Anlegen einer IoT-Plattform | Maciej Dzialoszynski | 0.5 hrs | **Completed** | Die IoT-Plattform ist über die IBM-Cloud-Accounts der Teammitglieder zugänglich. |
| 4395 | Einrichten einer Frontend-Cloud-Foundary-App | Maciej Dzialoszynski | 0.5 hrs | **Completed** | Die Sample-App ist lauffähig. |
| 4396 | Einrichten einer Backend-Cloud-Foundary-App | David Jovanovic | 0.5 hrs | **Completed** | Die Sample-App ist lauffähig. |
| 4397 | Anlegen von Git-Repository für Frontend | Maciej Dzialoszynski | 0.5 hrs | **Completed** | Git-Repository ist klonbar. |
| 4398 | Anlegen von Git-Repository für Backend | Jordi Rieder | 0.5 hrs | **Completed** | Git-Repository ist klonbar. |
| 4399 | Kommunikation zwischen Backend und Datenbank (Cloudant) | Jordi Rieder | 0.5 hrs | **Open** | Verbindung ist hergestellt |
| 4400 | Kommunikation zwischen Backend und Frontend | Jordi Rieder | 0.5 hrs | **Open** | Verbindung ist hergestellt |
| **4391** | **Frontend - Wireframes** |  | **10 hrs** | **Completed** |  |
| 4405 | Login/Registrierung | Maciej Dzialoszynski | 1 hrs | **Completed** | Das Wireframe wurde von Papier auf das Mockup-Tool Balsamiq übertragen. |
| 4406 | Gerät-Registrierung | Sebastian Grünewald | 3 hrs | **Completed** | Das Wireframe wurde von Papier auf das Mockup-Tool Balsamiq übertragen. |
| 4407 | Gerät-Liste | Sebastian Grünewald | 3 hrs | **Completed** | Das Wireframe wurde von Papier auf das Mockup-Tool Balsamiq übertragen. |
| 4408 | Gerät-Details | Maciej Dzialoszynski | 3 hrs | **Completed** | Das Wireframe wurde von Papier auf das Mockup-Tool Balsamiq übertragen |
| **4392** | **Backend - Setup** |  | **3 hrs** | **In Progress** |  |
| 4409 | Ordnerstruktur | David Jovanovic | 1 hrs | **Completed** | Jedes Modul und jede zusätzliche Technologie hat einen eigenen Ordner. |
| 4410 | Node-Arbeitsumgebung aufsetzen | David Jovanovic | 2 hrs | **Completed** | Die Mitglieder haben Zugriff auf die Node.js App auf der Cloud und können ihre Änderungen pushen. Die neuste Version sollte dann immer verfügbar sein über den Link auf der Cloud |
| **4393** | **Backend - GW-Cloud** |  | **10 hrs** | **In Progress** |  |
| 4411 | GW-Datenempfang | David Jovanovic | 5 hrs | **Open** | Die korrekten Daten werden empfangen. |
| 4412 | Datenfilterung | Jordi Rieder | 5 hrs | **In Progress** | Man erhält nur die gewünschten Daten |
| **4413** | **Frontend - Setup** |  | **7 hrs** | **In Progress** |  |
| 4414 | Ordnerstruktur | Sebastian Grünewald | 1 hrs | **Completed** | Jede Komponente und jede zusätzliche Technologie hat einen eigenen Ordner. |
| 4415 | React-App mit der Node-App verbinden | Sebastian Grünewald | 1 hrs | **Completed** | Datenaustausch zwischen jenen Apps möglich. |
| 4416 | Routen festlegen | Maciej Dzialoszynski | 4 hrs | **In Progress** | Navigation zwischen Login/Reg-Seite und Hauptseite funktioniert und die Navigation zwischen einzelnen Komponenten funktioniert. |
| 4417 | Sass aufsetzen | Maciej Dzialoszynski | 1 hrs | **Completed** | Sass in Projekt eingebunden und funktionstüchtig. |

## Sprint Burndown Chart



## Product Burndown Chart

Da es bisher nur einen Sprint gab ist das Product-Burndownchart nicht aussagekräftig.



## Produktivität

Die Produktivität ist leider nicht optimal, da es noch offene Userstorys gibt, welche noch vom Auftraggeber abhängen. Damit ist gemeint, dass der Betreuer, welcher zuständig für die Entwicklungsumgebung auf der Cloud ist, erst wieder zu Beginn vom zweiten Sprint zurück vom Urlaub ist.

