\*Product-Backlog of Corona Update

1.Rentner Story:

Als Rentner möchte ich immer die aktuellen Corona Fälle anschauen können, um meine Neugierde zu befriedigen.

2.Bestatter Story:

Als Bestatter möchte ich immer die aktuellen Zahlen der Corona Tode zur Verfügung haben, um mein Bestattungsunternehmen gut koordinieren zu können.

3.Studenten Story:

Als Student möchte ich einen Überblick über die Entwicklung der Corona Fällen erhalten, um einschätzen zu können, wann ich mich auf den Start der Präsenz-Veranstaltungen einstellen kann.

4.Einzelhändler Story:

Als Einzelhändler möchte ich die Fälle in Abhängigkeit der Bevölkerung überblicken, um die finanziell notwendigen Schritte abschätzen zu können.

5.Unternehmer Story:

Als Unternehmer brauche ich eine Übersicht der bekannten Corona Fälle, um den richtigen Zeitpunkt für die Umstellung von Kurzarbeit auf den normalen Betrieb zu finden.

6.Arzt Story:

Als Allgemeinmediziner, Inhaber einer Arzt-Praxis, wäre es vorteilhaft einen Überblick über die Entwicklung der Corona Fälle zu erhalten, um meine Praxis und Personal optimal auf die derzeitige Situation einzustellen.

7.Gamer Story:

Als ein Gamer möchte ich aktuelle Zahlen der Corona Pandemie erhalten, um die Entwicklung der Server-Kapazitäten in Bezug auf weniger Überlassungen abschätzen zu können.

\* Minimalistische Implementierung

Bestatter Story, Unternehmer Story

\* Zerlegen Sie die User-Stories in Tasks, schätzen Sie den Aufwand

Daten abfragen und aufbereiten: 14 Stunden

-Daten Verarbeitung in Node-Red

-> Durch getRequest an Corona Rest\_API

-> Benötigte Informationen der erhaltenen Json filter und in Datenbank abspeichern

-Daten von Node-Red auf Java Server abfragen

-> GetRequest an Node-Red Server auf Raspberry\_Pi

Geeignete Visualisierung der Daten erstellen :10 Stunden

- Leeres Diagramm mit geeigneter Library erstellen

- Daten aus GetRequest an Raspberry\_Pi in Diagramm laden

Bereitstellung der Visualisierten Daten durch Webserver: 2 Minuten

-Code Compilen und Webserver hochfahren

\* Sprints Retrospektive

+ Positiv gelaufen

-> Kommunikation in Gruppe trotz Corona auf sehr stabilen Niveau und das Arbeiten mit Raspberry (Server) war nicht einseitig, nicht nur von einer Person Programmiert, sondern in Online Calls (Screen sharing und Fern\_Zugriff )

->Programmieren in Node\_Red ergab keine größeren Probleme

* Negativ gelaufen

->Diagramm als Visualisierung der Corona Lage zwar als eindeutiges Ziel, aber keine vernünftige Umsetzung im Focus(In Java Programmiert oder andere ) erste Ansatz : mit Java Fx Oberfläche ( Diagramm ) gestalten und über Web\_Servlet ausgeben (Keine Lösung für diesen Ansatz gefunden ) zweiter Ansatz : Diagramm mit Java Script Library (Chart Js )erstellen und Html mit diesem JavaScript Code als String im Web\_Servlet zurückgeben ( Response-String )