



1. Software Tools

- **IDEs:** Visual Studio Code, IntelliJ IDEA
- **Versionierung:** GitHub
- **Entwicklungs-Tools:** Postman, Google Spreadsheets, draw.io
- **Projektmanagement:** Jira
- **Docs:** Python, MkDocs

2. Plattformen

- **Hosting:** Docker
- **Frameworks:** Vue.js (JavaScript), Spring Boot (Java), Docker Compose

3. Techniken / Architektur:

- **Programmierparadigmen:** Objektorientiert, reaktiv.
- **Entwicklungsmethoden:** Agile, Scrum, Kanban
- **Spezielle Patterns:** MVC (Backend), MVVM (Frontend)

4. Datenbank:





5. Testing

- Testing mit JUnit, worauf der Fokus auf dem Backend liegt
 - Interagiert mit separater Test-DB und/oder verwendet Mockdaten
 - Service Klassen der Datenbank auf Logik getestet
 - Controller Klassen auf HTTP Status getestet
- Code Coverage wird mittels JaCoCo errechnet (Manuell & automatisch in CI/CD Pipeline)
- Projekt Test Coverage = **75%**

6. Metriken

- Attribute Hiding Factor (AHF) = **97,73%**
- Coupling Factor (CF) = **11,67%**
- Average Operation Complexity (AOC) = **1,3**

7. CI/CD

Frontend

build-and-deploy

> **Push** auf **main**

- build
- artifact speichern

Backend

build-and-test

> **Push** auf **main**

- build
- tests
- artifact speichern

calc_coverage

> **PR** auf branch

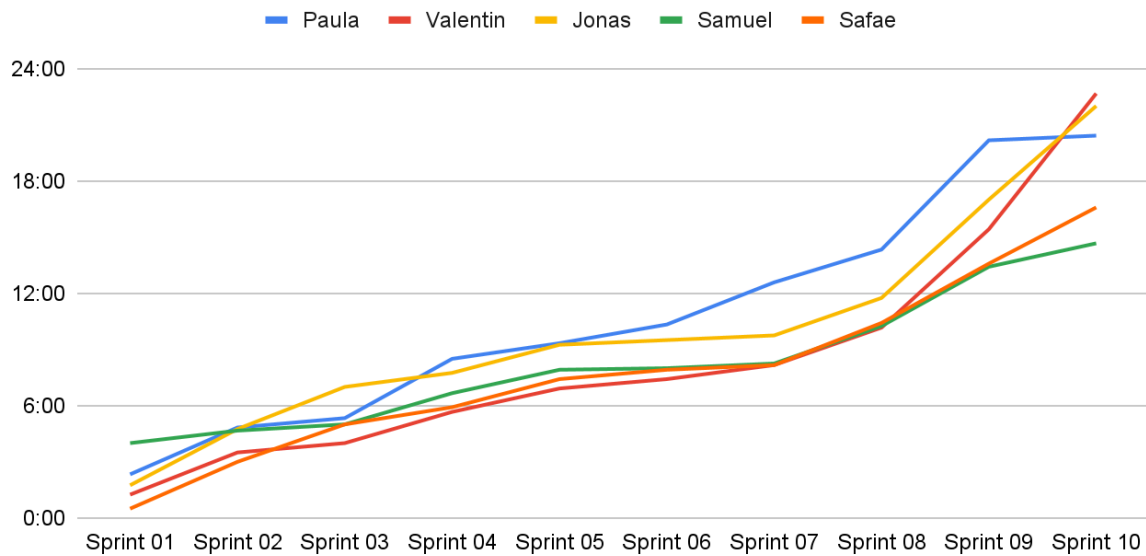
- testcoverage an PR anfügen



4. Zeitaufwand:

Stunden pro Person

Zeitganglinie pro Person



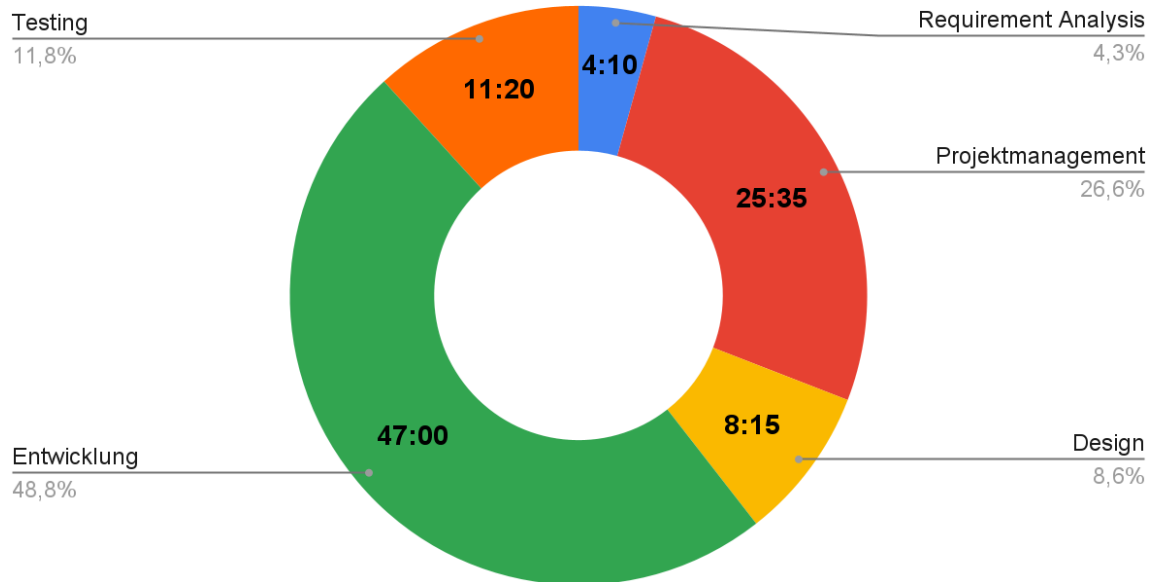
Hauptbeitrag pro Person

Person	Hauptbeitrag
Safae K.	Backend endpoints and JUnit Tests
Jonas S.	Frontend & Projektmanagement
Valentin W.	Openstreetmap Backend Anbindung & JUnit Tests
Paula K.	Fronted Sidebar Overview
Samuel B.	Blogeinträge & Dokumentation & Metriken

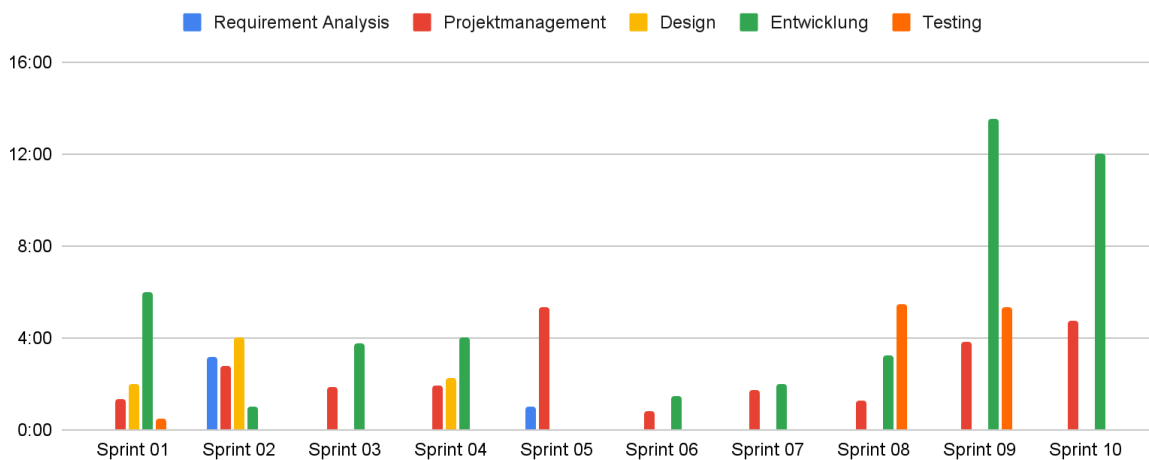


Stunden pro Workflow

Zeitaufwand pro Tätigkeitskategorie (gesamt)



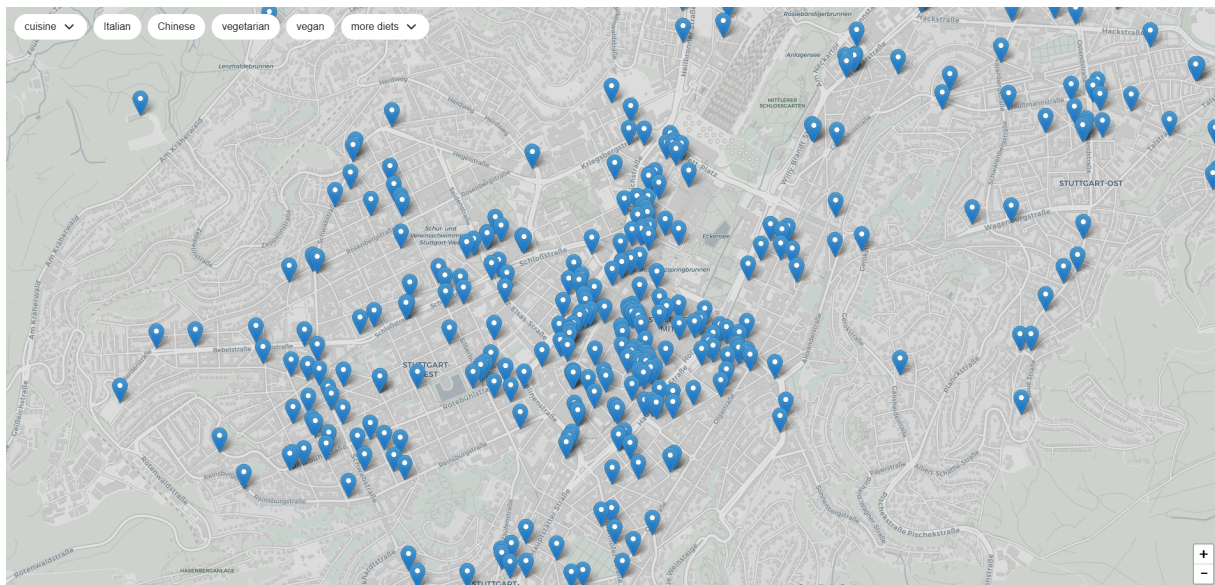
Zeitaufwand pro Tätigkeitskategorie (pro Sprint)



Inception	Elaboration	Construction	Transition
Sprint 1	Sprint 2 - 5	Sprint 6 - 10	Sprint 11



Highlights der Demo:



Beschreibung des Bildes:

Dies ist ein Screenshot der Hauptseite des Projekts. Zu sehen ist eine Karte mit den über OpenStreetMap geladenen, blau markierten Restaurants. Durch das Anklicken der blauen Marker wird die Seitenleiste geöffnet, die auf dem nächsten Screenshot zu sehen ist.

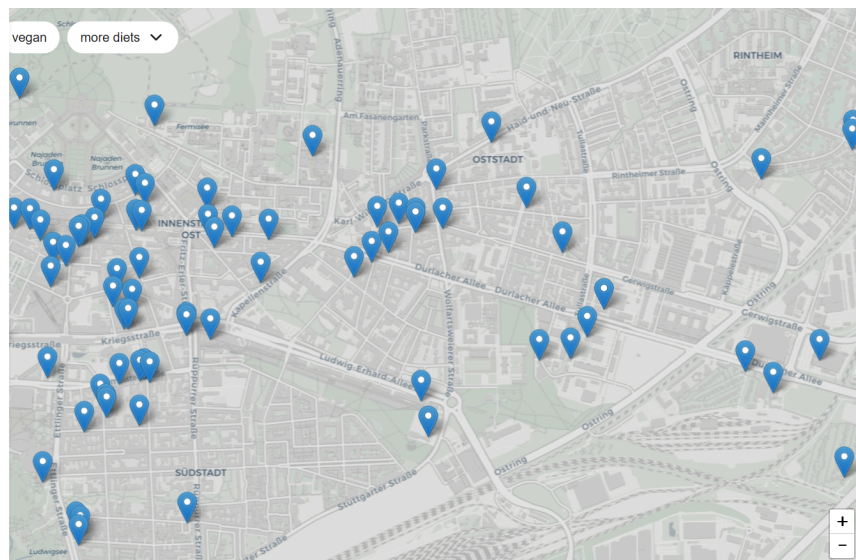
Hoepfner Burghof

★★★★★ (0)

Overview Reviews Info

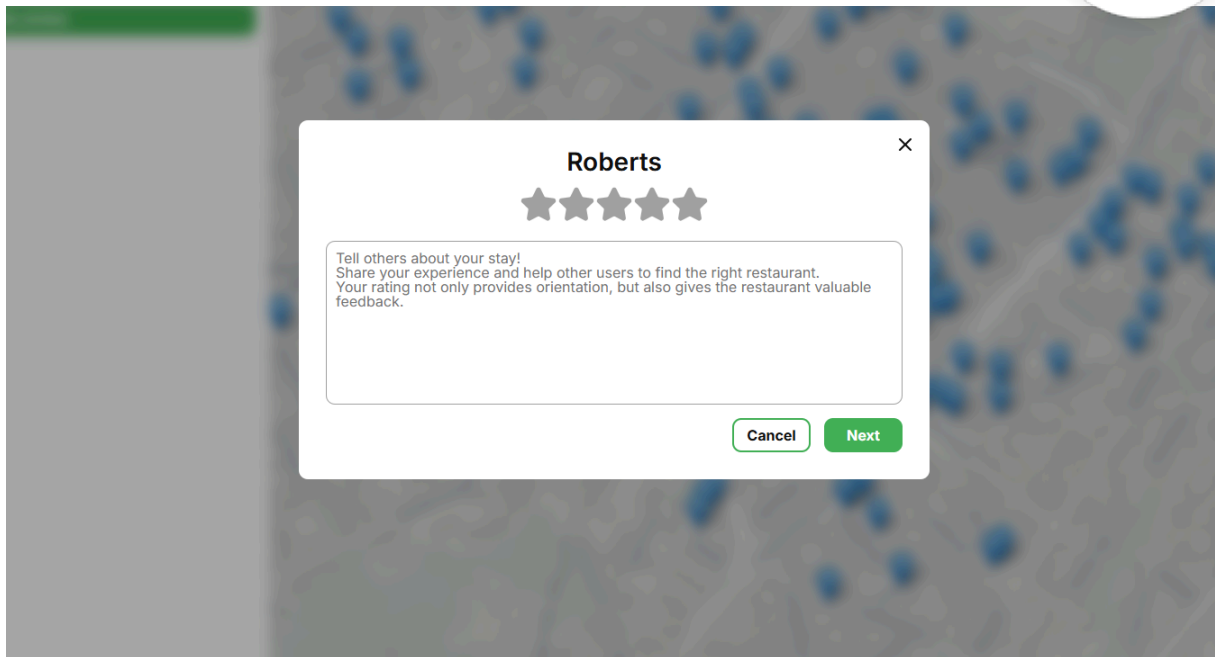
📍 Haid-und-Neu-Straße 18, 76131 Karlsruhe
🕒 Closed, opens at 11:30
🌐 <https://www.hoepfner-burghof.de/>
☎️ +49 721 622644

write review



Beschreibung des Bildes:

Dieser Screenshot zeigt die Seitenleiste des ausgewählten Restaurants. Hier werden die verfügbaren Informationen wie Adresse, Öffnungszeiten oder Telefonnummer angezeigt. Über die Seitenleiste können auch Bewertungen oder weitere Informationen, wie beispielsweise Angaben zur Barrierefreiheit oder zu Liefermöglichkeiten, angezeigt werden.



Beschreibung des Bildes:

In diesem Screenshot ist das Fenster zum Verfassen einer Rezension zu sehen. Es kann über die Seitenleiste eines Restaurants geöffnet werden. Eine Bewertung beinhaltet eine Sternebewertung von 1 bis 5 Sternen sowie einen optionalen Bewertungstext.

Weitere Highlights des Projekts sind:

- Ein weiteres Highlight sind die Docs, die über ein Package in Python erstellt werden. Dadurch wird ein einheitlicher und übersichtlicher Ort für unsere Dokumentation geschaffen.
- Die Verwendung von OpenStreetMap ist ein weiteres Highlight. Dadurch können wir Restaurants anzeigen, die auf der ganzen Welt verteilt sind.



- Handout

Das Handout muss folgendes enthalten:

- Projektnamen
- Aufwandsstatistiken
 - Arbeitsstunden pro Person und Hauptbeitrag pro Person
 - Arbeitsstunden pro Workflow (Requirement Analysis, Project Management, ...)
 - Arbeitsstunden pro Phase (Inception, Elaboration, ...)
- Highlights eurer Demo (Beschreibung + Screenshots)
- Highlights eures Projekts:
 - Architektur
 - Software Tools/Plattform/Technik/Libraries
 - Datenbank Design
 - Testing
 - Metriken
 - CI/CD
 - Und auf was ihr sonst noch sehr stolz seid