

TAU INF202 Software Engineering
Individuelles Projekt
Pflichtenheft

Projektdokumentation

Version: v1.0

Status: freigegeben

TADYA

Verantwortliche/r:

Berkay Biçer, e190503046@stud.tau.edu.tr

Yiğit Alıcıkuş, e200503026@stud.tau.edu.tr

Stakeholder: DI. Ömer Karacan, omer.karacan@tau.edu.tr

Fulya Yenilmez, fulya.yenilmez@tau.edu.tr

Dokumentenverwaltung

Dokument-Historie

Version	Status *)	Datum	Verantwortlicher	Änderungsgrund
v0.1	Entwurf	01.04.2023	Berkay Biçer	Ergänzungen zu Meilensteinen, Referenzen, Einleitung und Ausgangssituationen und Ziele.
v0.1	Entwurf	02.04.2023	Yiğit Alicıküş	Einschränkungen, Ziele(Lösung).
v0.1	Entwurf	07.04.2023	Yiğit Alicıküş	Gesamtarchitektur.
v0.1	Entwurf	07.04.2023	Berkay Biçer	Meilensteine, Referenzen, Design.
v0.1	Entwurf	14.04.2023	Berkay Biçer	Einleitung, Meilensteine, Design, Ausgangssituationen und Ziele.
v0.1	Entwurf	16.04.2023	Yiğit Alicıküş	Gesamtarchitektur.
v1.0	freigegeben	17.04.2023	Berkay Biçer	Nichtfunktionale Anforderungen. Gesamtkontrolle des Dokuments.
v1.0	freigegeben	17.04.2023	Yiğit Alicıküş	Funktionale Anforderungen. Gesamtkontrolle des Dokuments.

Dokument-Status: Entwurf / in Review / freigegeben (abgegeben)

freigegeben

Dokument wurde mit folgenden Tools erstellt:

Microsoft Office Word

Google Docs

QGIS

Inhaltsverzeichnis

1. Einleitung	4
2. Ausgangssituation und Ziele	5
3. Gesamtarchitektur	6
4. Funktionale Anforderungen	7
5. Nichtfunktionale Anforderungen	8
6. Abnahmekriterien	9
7. Projekt Meilensteine	10
8. Referenzen	11

1. Einleitung

Dieses Dokument dient dazu, die verbindlichen Anforderungen an den Prototyp einer interaktiven Kartenschnittstelle zu definieren und vollständig und einheitlich zu erläutern.

Die Use Cases und Anforderungen sind aus der Sicht des Stakeholders beschrieben.

Die von den Erdbebenopfern zu ergreifenden Maßnahmen und die grafische Oberfläche, auf die sie stoßen werden, werden aus der Sicht der Benutzer erläutert.

Kapitel 2 „Ausgangslage und Ziele“ zeigt anschaulich die Ausgangslage und die Gründe für die Wahl dieser interaktiven Kartenschnittstelle.

Im Kapitel 3 „Gesamtarchitektur“ sind die physikalische und die konzeptionelle Architektur des Systems, und die wichtigsten Subsysteme (Komponenten), die Anwender und die notwendigen Kommunikationsschnittstellen dargestellt. Hier sind auch zusätzliche Anforderungen an die Architektur oder Komponenten zu finden.

Im Kapitel 4 „Funktionale Anforderungen“ beinhaltet die Beschreibung der funktionalen Anforderungen durch die Ablaufbeschreibungen (User Stories), die Anwendungsfällen (Use Cases), und technischen und fachlichen Anforderungen (Requirements). Alle betriebsrelevanten Daten werden durch die Datenmodellen definiert.

Im Kapitel 5 „Nichtfunktionale Anforderungen“ sind die funktionalen Anforderungen durch diejenigen Anforderungen erweitert, die keine funktionalen Anforderungen sind.

Im Kapitel 6 „Abnahmekriterien“ sind die Abgabeartefakten festgelegt, die ohne Abstimmung des Stakeholders nicht zu manipulieren sind.

Im Kapitel 7 „Projekt Meilensteine“ sind die wichtigsten Termine aufgelistet, die den Fortschritt die Teilergebnisse des Projektes definieren.

Im Kapitel 8 „Referenzen“ sind die wichtigsten Referenzen aufgelistet.

2. Ausgangssituation und Ziele

Einleitung

In diesem Kapitel teilen wir mit Ihnen, was uns dazu inspiriert hat, dieses Projekt zu starten, was wir mit diesem Projekt erreichen wollen und was wir uns mit diesem Projekt vorstellen.

Nachdem das Erdbeben in der Türkei am 06.02.2023 stattgefunden wurde und die darauf folgenden Such- und Rettungsbemühungen sowie die Hilferufe der Menschen über soziale Medien gesehen hatten, beschlossen wir, dieses Projekt zu starten. Unser Ziel in diesem Projekt ist es, die Zusammenarbeit und den Informationsaustausch nach einem verheerenden Erdbeben zu erleichtern.

Problemstellung (Funktionalität)

Dieses Projekt ermöglicht, dass überlebende Betroffenen auf einfache Weise Hilfe anfordern und beschädigte Gebäude schnell anmelden.

Stakeholder (Anwender):

Alle Bürger, die von einem verheerenden Erdbeben betroffen sind, sowie Katastrophenschutzbeauftragte von Regierungen oder privaten Institutionen können dieses System nutzen.

Systemumfeld (Einsatzumgebung)

Benutzer können mit einem Mobiltelefon oder einem Computer mit **Internetverbindung** auf das System zugreifen. Eine leicht zugängliche und leicht verständliche Website wird Ihnen dienen.

Rahmenbedingung (Einschränkungen)

Es gibt einige Einschränkungen wegen der externen APIs:

- Auf der Benutzeroberfläche wird Bing Maps V8 Web Control APIs benutzt, deswegen unterliegen das Programm die Beschränkungen der API
- Die ganze Welt wird auf der Karte angezeigt, aber die Anzahl der Polygone, die der Karte interaktiv hinzugefügt werden können, ist begrenzt, vorbehaltlich der Beschränkungen der Bing-API-Schlüssel. Aus diesem Grund können nur die Gebäude in Istanbul bearbeitet werden. (Es besteht die Möglichkeit, dass der Bing-API-Schlüssel nicht die Polygone aller Häuser in einer Großstadt wie Istanbul unterstützt. In diesem Fall kann eine interaktive Schnittstelle nur für Gebäude im Stadtteil Beykoz vorbereitet werden, indem das Projekt verkleinert wird.)
- Das System kann man nur mit einem Gerät, das **Internetverbindung** herstellen kann, zugreifen kann.

Ziele (Lösung)

Mit diesem Projekt wird so schnell wie möglich eine Schadensanalyse des Erdbebengebiets durchgeführt. Mit den Beiträgen der Nutzer und der Zustimmung der Behörden können schnellstmöglich die Zahl der beschädigten und zerstörten Gebäude sowie die Zahl der Menschen unter den Trümmern ermittelt werden. Gleichzeitig können Menschen unter den Trümmern und ihre Angehörigen den Standort der Hilfeanforderungen in Echtzeit verfolgen und so die Angst vor fehlenden Informationen vermeiden.

Damit die Benutzer Hilfe anfordern und die Autorisierte Zustand des Gebäudes und Betroffene aktualisieren können, wird sich auf der Benutzeroberfläche eine **interaktive Karte** befinden.

Diese Karte wird mit **Bing Maps API** erbracht und Interaktivität auf der Karte(einfügen Pushpin, Gebäude auswählen, navigieren usw.) verschaffen.

Anforderungen und Informationen der Gebäude wird sich in der Datenbank befinden und Abfrage und Abspeichern Operationen Dank Webservices durchgeführt

Detaillierte Einzelheiten zu den folgenden Teilen berücksichtigen.

3. Gesamtarchitektur

Einleitung

Dieses Kapitel beinhaltet die Gesamtsystemarchitektur der Darstellung von dem System und deren Komponenten und die Einzelheiten von diesen Komponenten, damit die allgemeine Funktionsweise des Systems erklärt wird.

Gesamtarchitektur

In Gesamtarchitektur wird Einzelheiten der folgende Komponenten beschreibt:

- Benutzeroberfläche(UI)
- Webservices(API)
- Datenbank
- Externe APIs und Daten

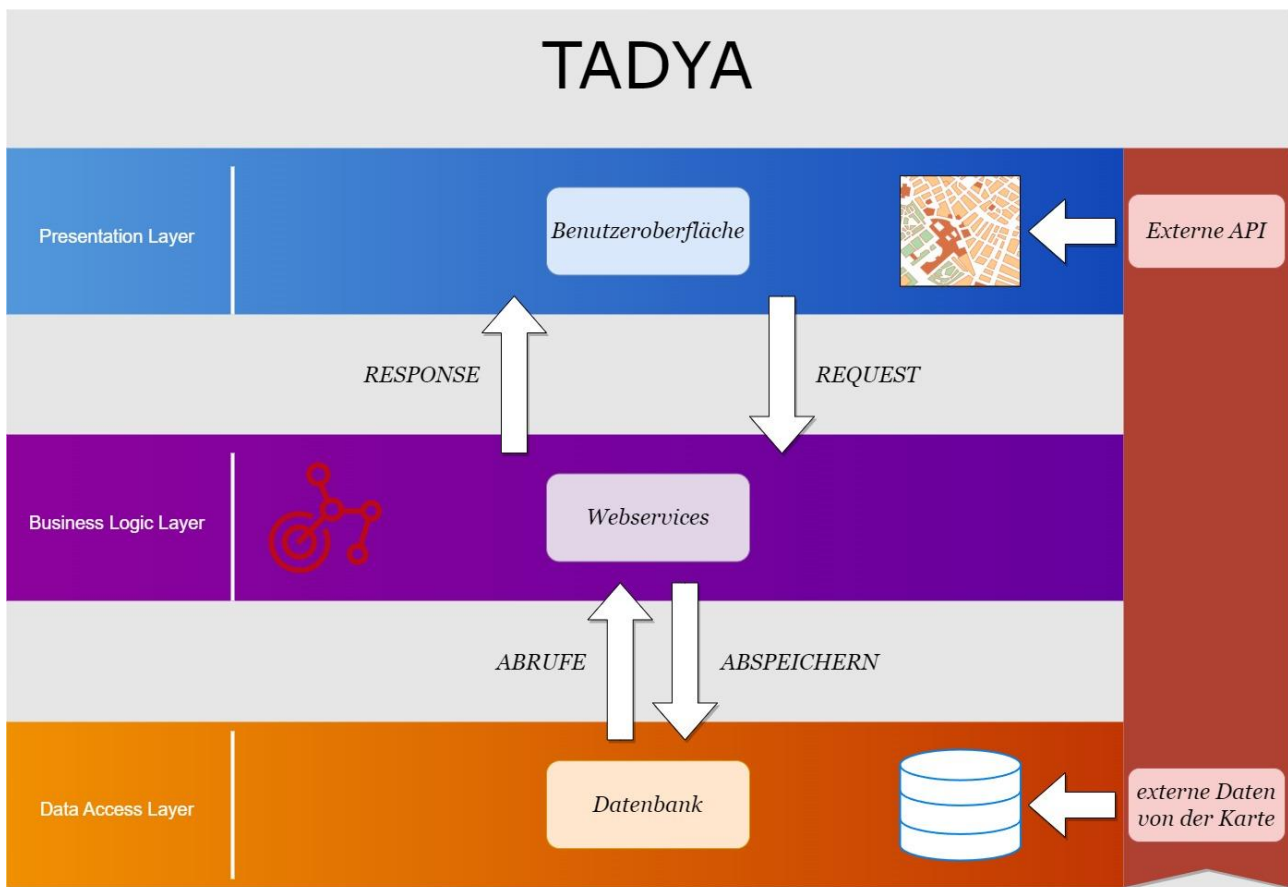


Abbildung 1: Architekturmuster und Komponenten

Mit dem Diagramm in **Abbildung 1** werden die Arbeitsbeziehungen zwischen den Systemkomponenten in der Systemumgebung beschrieben und ein Überblick über die allgemeine Struktur des Systems und Architekturmuster gegeben.

Komponente Benutzeroberfläche

Die Benutzer können auf das System über eine Benutzeroberfläche zugreifen und interagieren.

Bisher gemeldete Hilfeanforderungen werden von dem Benutzer in der Schnittstelle angezeigt und neue Anforderungen gemeldet können.

Die Autorisierten können Informationen(Bestätigung und Bericht nach der Rettung) eines Gebäudes aktualisieren.

Die Benutzeroberfläche dient als Brücke zwischen dem Benutzer und dem System, indem sie Benutzeranfragen an das System weiterleitet die Antworten anzeigt.

Komponente Webservices

Die Webservices befinden sich zwischen der Präsentationsschicht (der Benutzeroberfläche) und der Datenzugriffsschicht (der Datenbank).

Diese Schicht behandelt die Geschäftslogik, Geschäftsregeln sowie Abrufe und Abspeichern der Daten.

Es kommuniziert mit der Datenzugriffsschicht, um die Anforderung und Informationen eines Gebäudes abzurufen und abzuspeichern.

Durchführung von Berechnungen und Transformationen von Daten, die durch die Geschäftslogik erforderlich sind.

Komponente Datenbank

Es steht zwischen der Business-Logic-Schicht und dem Datenbank und bietet eine Abstraktionsschicht, die ermöglicht, dass dem Datenbank interagieren kann.

Zum Abrufen und Speichern von Daten werden die SQL-Abfragen oder anderen Datenzugriffsbefehlen generieren und ausführen.

Externe Schnittstellen

Bing Maps SDK APIs

Die Benutzer können die Karte in der Benutzeroberfläche interagieren und anzeigen, durch den Bing Maps APIs.

Die folgenden Interaktionen auf der Karte werden mit der APIs zur Verfügung gestellt:

- Auswahl die Gebäude auf der Karte
- Anzeigen der Anforderungen und Informationen der Gebäude auf der Karte
- Navigieren auf der Karte

Externe Daten

Externe Daten wurden verwendet, um die Interaktivität auf der Karte zu gewährleisten.

Wenn die Benutzer ein bestimmtes Gebäude auswählen möchten, wird es zur Verfügung gestellt, durch externe Polygon-Daten der Gebäude.

Diese Polygon-Schicht(Polygon-Daten der Gebäude) werden auf der Karte hinzufügen.



Abbildung 3

4. Funktionale Anforderungen

Einleitung

In diesem Kapitel sind Anforderungen (inklusive User Stories und Use Cases) an Gesamtsystem, aber auch an die einzelnen Systemkomponenten definiert.

UI Use Cases

/UI-1/ Ansicht der Karte aus der Sicht des non-authorized/authorized Benutzers wird in der **Abbildung 4** und **Abbildung 5** dargestellt

/UI-2/ In dieser konzeptionell Oberfläche werden die ausgewählten Gebäude mit Türkis markiert.

/UI-3/ Die gebäude(nicht gemeldete, gemeldete(nicht bestätigt), gemeldete(bestätigt)) werden auf der Karte auf verschiedene Weise markiert, sowie in **Abbildung 5**

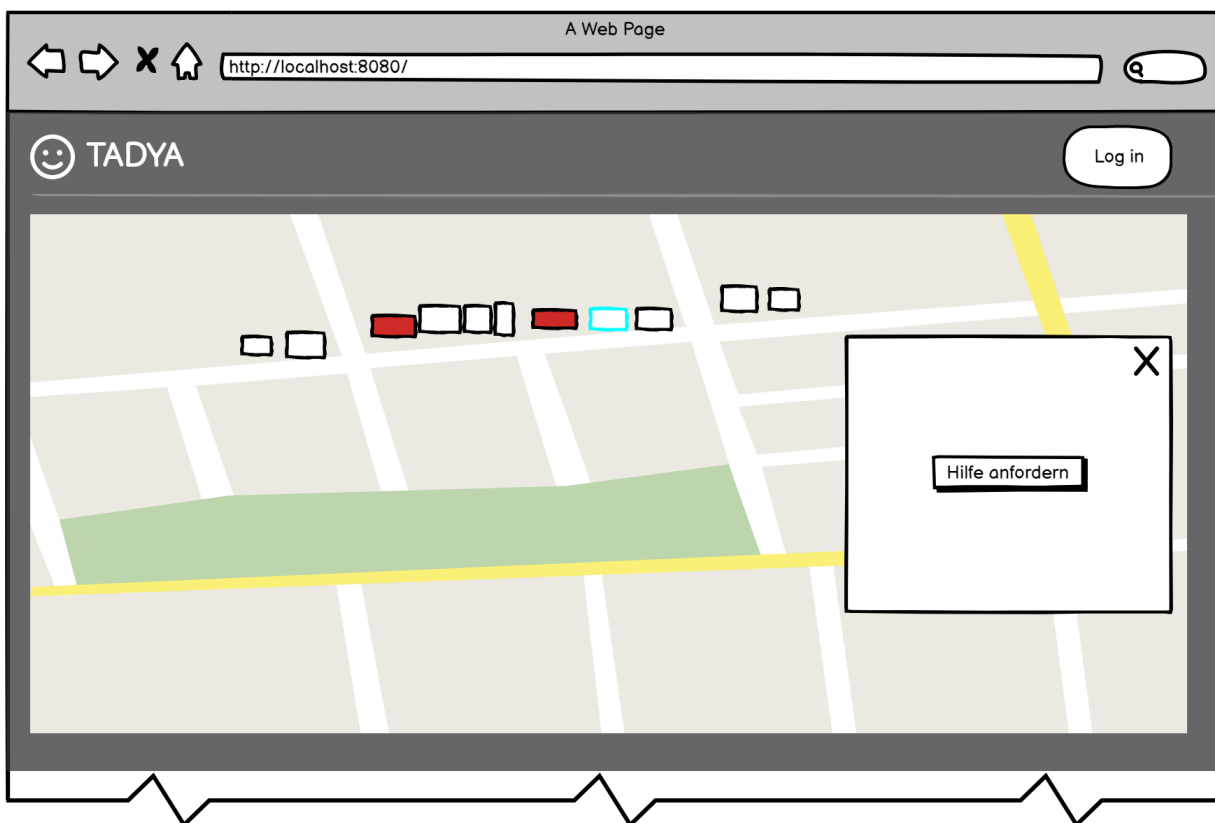


Abbildung 4: Benutzeroberfläche aus der Sicht des non-authorized Benutzers

/UI-4/ Non-authorized Benutzer können Hilfe für die Gebäude, für die schon nicht angefordert wird, anfordern.

/UI-5/ Sie können auch die schon gemeldete Anforderungen betrachten lassen, wenn man auf eine Gebäude klicken, sowie in **Abbildung 5**

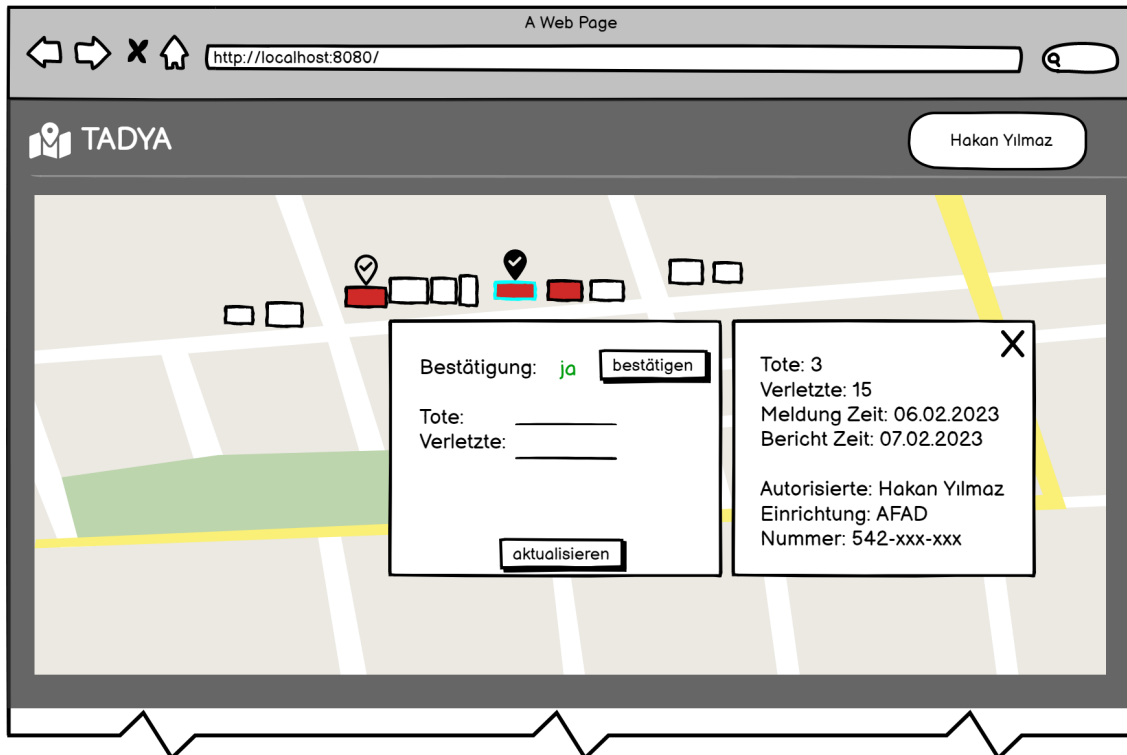


Abbildung 5: Benutzeroberfläche aus der Sicht der Autorisierte

/UI-6/ Zum Beispielfall wird die bereits gemeldete Gebäude(bestätigen und Informationen eingeben) dargestellt.

/UI-7/ Die Autorisierte können Anforderungen für ein Gebäude bestätigen und den Zustand des Gebäudes berichten und diesen Bericht aktualisieren.

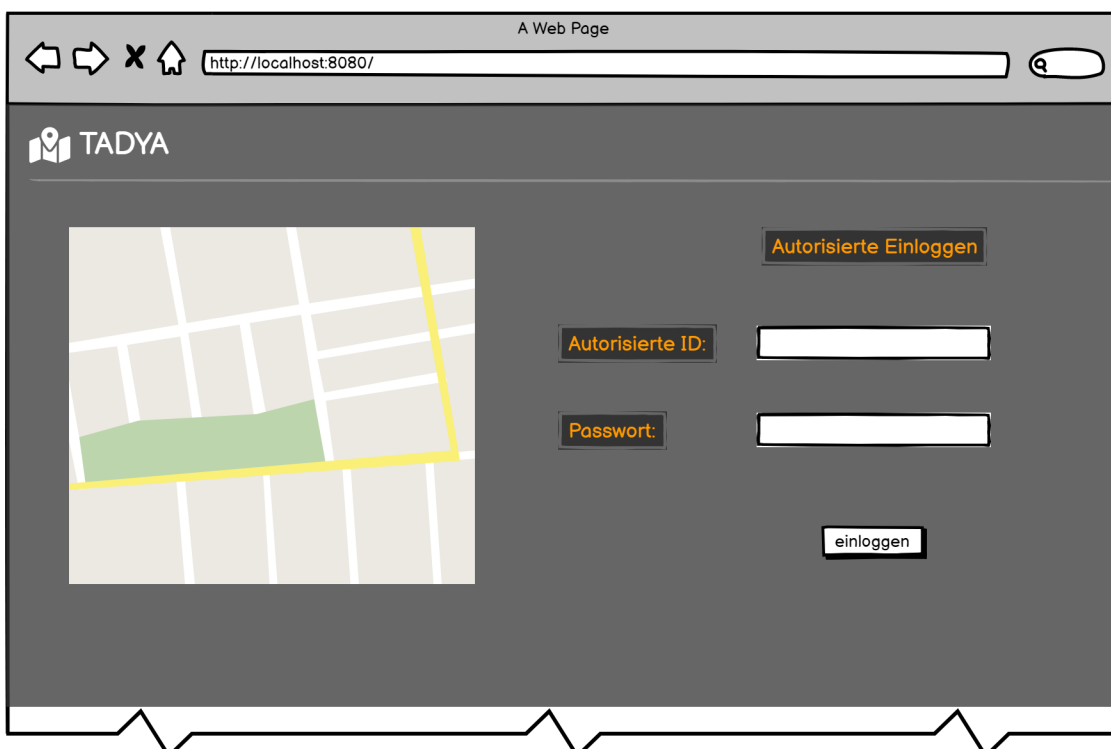


Abbildung 6: Login-Bildschirm für Autorisierte

/UI-8/ Wenn Autorisierte auf der autorisierten Schnittstelle erreichen möchten, er/sie kann einloggen, durch die Log in Button klicken.

API Use Cases

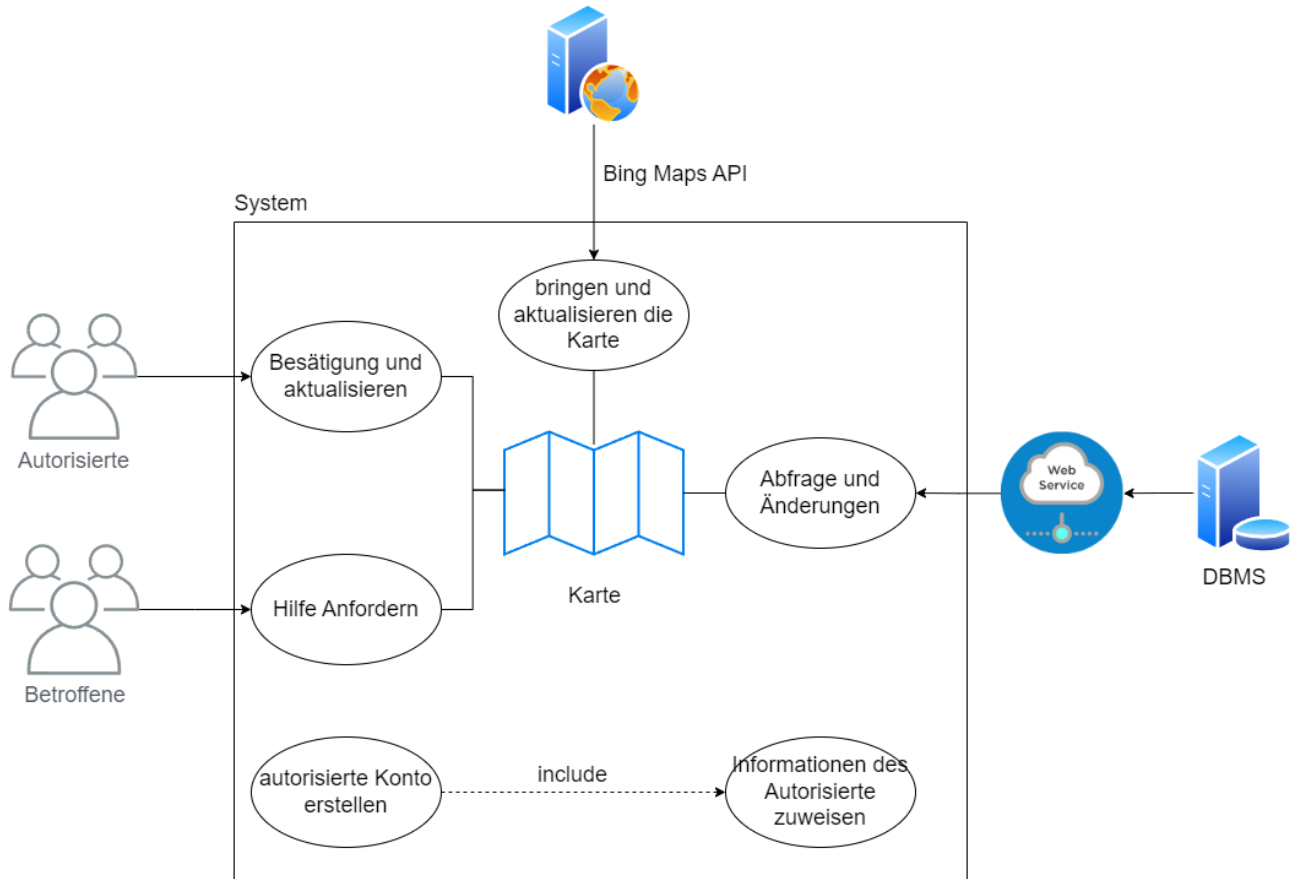


Abbildung 6: Use-Case diagramm

/UC-1/ Die Use-Cases, die auf der Benutzeroberfläche gewährleistet werden, werden in der Teil "UI Use Cases" erklärt.

/UC-2/ Autorisierte-Konten werden von Webservices erstellt und auf der Datenbank abgespeichert.

/UC-3/ Wenn ein Benutzer Hilfeanforderung für ein Gebäude melden möchte, die bereits nicht gemeldet, wird es von der Webservice in der Datenbank abgespeichert.

/UC-4/ Die Anforderungen und Berichte werden, die bereits in der Datenbank abgespeichert, in die Präsentations-Schicht geschickt, damit sie auf der Benutzeroberfläche erscheinen.

Technischen und fachliche Anforderungen

Bing Maps APIs:

/BMA-1/ Mithilfe der APIs wird die interaktive Karte auf der Benutzeroberfläche erbracht.

/BMA-2/ Es dient dazu, dass die Anforderungen und Informationen, die von Webservices API erhalten, dynamisch anzeigen.

/BMA-2/ Externe Polygon-Daten wird von der APIs auf dem Karte als Schicht hinzugefügt.

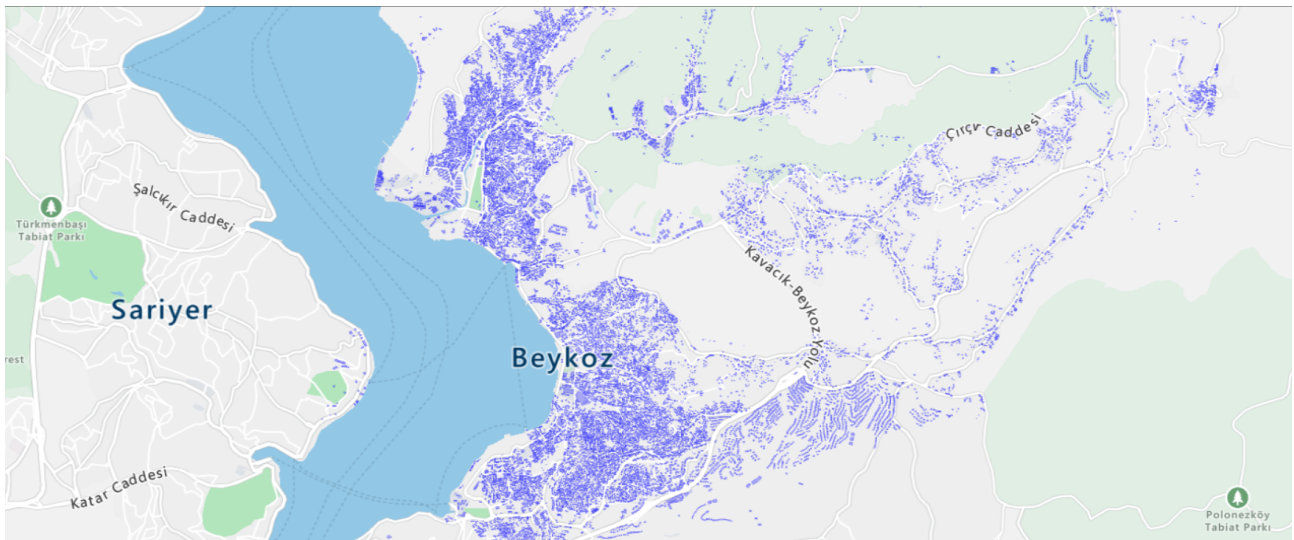
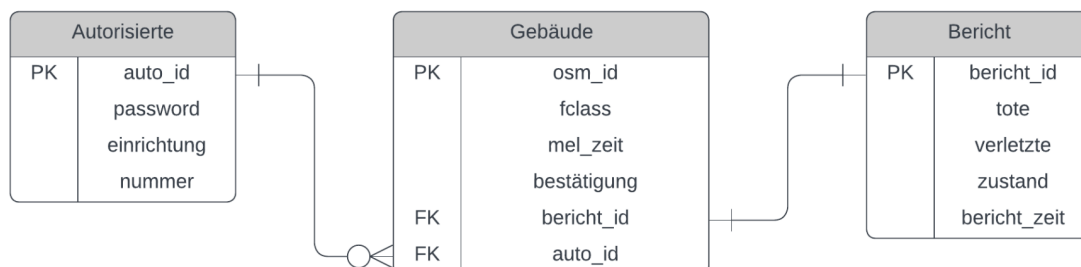


Abbildung 7: Karte mit der Polygon-Daten

Datenmodell



5. Nichtfunktionale Anforderungen

Einleitung

In diesem Kapitel sind die nicht-funktionalen Anforderungen ans Gesamtsystem aber auch an die einzelnen Systemkomponenten definiert. Es wird besonders auf die Software Qualität Wert gelegt (Testing).

Nicht-funktionale Anforderungen an die Systemarchitektur (Architekturmuster, Deployment)

- **/SYS-1/** Funktionalitäten(Abrufe und Abspeichern) werden in verschiedenen Klassen gewährleistet und implementiert.
- **/SYS-2/** Das ganze System wird auf dem lokalen Host laufen(Datenbankserver, Webseite, usw.)

Nicht-funktionale Anforderungen an die Entwicklungsumgebung

- **/DEV-1/** Die Entwicklungsumgebung ist frei wählbar!

Nicht-funktionale Anforderungen an die Entwicklungswerkzeuge (Sprache, IDE, Frameworks)

- **/TOL-1/** Verwendete Computersprachen im Projekt sind Java, html, css und javascript.
- **/TOL-2/** Verwendete Framework im Projekt ist Spring Boot.
- **/TOL-3/** Verwendete Database im Projekt ist MySQL Server.
- **/TOL-4/** IDE's, die bei der Vorbereitung des Projekts verwendet werden, sind IntelliJ und Visual Studio Code.

Nicht-funktionale Anforderungen an die Teststrategie (Qualitätssicherung)

- **/TEST-1/** Es sollte getestet werden, ob es sich mit dem Api-Schlüssel verbinden kann und ob die Karte auf der Webseite angezeigt werden kann.
- **/TEST-2/** Es sollte getestet werden, ob beim Anklicken der Gebäude auf der interaktiven Karte die Gebäudeinformationen in der Infobox angezeigt werden.
- **/TEST-3/** Alle Use Cases, User Stories und Anforderungen sollen getestet und berichtet werden.
- **/TEST-4/** Es sollte getestet werden, ob sich die vom Benutzer in der Präsentationsschicht geänderten Informationen auch in der Datenbank geändert haben.
- **/TEST-5/** Es sollte getestet werden, ob die geänderten Informationen in der Datenbank für andere Benutzer sichtbar sind.

6. Abnahmekriterien

Das Projekt wird mit den folgenden Artefakten abgegeben:

- Dokumentation:
 - Pflichtenheft: INF202-tadya-pflichtenheft-2023.docx
- Software
 - Link zu GitHub Projekt: <https://github.com/Gruppe1-Fulya/Tadya>
- Evidenz:
 - System/Software-Demo via Videoclip:
<https://drive.google.com/file/d/1gc6mGyECKLwm-yISTZom8DY2zZdMkgxM/view?usp=sharing>

7. Projekt Meilensteine

- Meilenstein #1: Wir recherchierten, wie wir das Projekt vorbereiten sollten, und diskutierten, welche Sprachen wir verwenden könnten, welche Kartenanwendung besser zu verwenden wäre und welche Art von Datenbank wir vorbereiten sollten.
- Meilenstein #2: Wir haben ein Musterdesign für die Webseite erstellt und am Frontend gearbeitet.
- Meilenstein #3: Wir haben das Lastenheft erstellt und die Genehmigung für das Projekt erhalten. Gleichzeitig erhielten wir Feedback von der Studiengangsleitung, was gut und was schlecht an unserem Projekt ist.
- Meilenstein #4: Um die Karte interaktiv zu gestalten, haben wir als Experiment eine Polygonkarte der Gebäude in Beykoz erstellt.
- Meilenstein #5: Wir haben den Polygon-Layer von Beykoz zu unserer Karte hinzugefügt und den Polygon-Layer von ganz Istanbul erstellt.
- Meilenstein #6: Wir haben GUI-Designs, Anforderungen und Anwendungsfälle erstellt und das Pflichtenheft fertiggestellt.

8. Referenzen

- <https://learn.microsoft.com/en-us/bingmaps/v8-web-control/>
- <https://samples.bingmapsportal.com/>
- https://tudelft3d.github.io/3dfier/building_footprints_from_openstreetmap.html#download-osm-data
- <https://geojson.io/>
- <http://download.geofabrik.de/>
- <https://www.qgis.org/tr/site/>
- <https://balsamiq.com/>
- <https://app.diagrams.net/>