



universität  
wien

# KONZEPT MICROSERVICE PLACES STATIC

Verfasser

Antonio Gasparevic (01506462), Nemanja Srdanovic (01576891),

Kevin Çobaj (01635087)

Wien, 2021

15.05.2021

Studienkennzahl lt. Studienblatt:

UA 033 526

Fachrichtung:

Informatik - Wirtschaftsinformatik

Betreuerin / Betreuer:

o. Univ.-Prof. Dr. techn. Dimitris Karagiannis; Dipl. Ing.,  
Mag. Dr. Wilfrid Utz

## **Abstract**

Wir haben es uns zum Ziel gesetzt eine API zu entwickeln die mit dem Microservice Framework OLIVE auf dem DKE Server für alle Mitstudierenden und Kollegen vom DKE zur Verfügung gestellt werden. Der Microservice soll einen Ort als Input bekommen und die Kategorien der in der Umgebung liegenden Orte zurückliefern. Danach kann nach der Kategorie gesucht werden, die dann die genauen Orte ausgibt und dem User bereitstellt.

# 1 Einleitung

Mit einer exponentiell wachsenden Smartphone-Technologie werden viele Dienste angeboten, an die vor 10 Jahren noch nicht einmal gedacht wurde. Viele dieser Dienste enthalten auch standortbezogene Optionen, z. B. die Freigabe Ihres aktuellen Standorts auf WhatsApp. Unser Microservice sollte „Points of Interest“ basierend auf Ihrer aktuellen Adresse und einer ausgewählten Kategorie zurückgeben. Das Potenzial für eine Kombination mit anderen Diensten wie Geolokalisierung oder Wettervorhersage ist enorm. Dieser Service kann auch als Plugin für verschiedene Anwendungen angeboten werden.

## 2 Potenzielle Anwendungen

### 2.1 Touristen-App

Dieser Service kann einer Touristen-App angeboten werden. Finden Sie je nach aktuellem Standort die für Touristen am wichtigsten benötigten Interessenpunkte. Ein Beispiel könnten Touristeninformationszentren, Sehenswürdigkeiten, öffentliche Verkehrsmittel sein.

### 2.2 Gastronomie-App

Dieser Service kann Restaurants und Bars angeboten werden. Ein aktuelles Beispiel wäre eine Person, die sich in der Innenstadt befindet und ein Gastronomieunternehmen besuchen möchte. Sie konnten sehen, wie viele Personen zu einem bestimmten Zeitpunkt auf jedes Gastronomieunternehmen geklickt haben oder wer das am häufigsten angeklickten Restaurant ist.

### 2.3 Statistik-App

Dieser Service kann der österreichischen Statistikbehörde angeboten werden. Zu jedem bestimmten Zeitpunkt konnten Sie sehen, welche Kategorien am häufigsten gesucht wurden. Beispiel: Ein Benutzer sucht zur Mittagszeit nach Restaurants, abends jedoch nach Bars.

## 2.4 Taxi-App

Dieser Service kann einer Taxi-App angeboten werden. Eine Kategorie im Service könnte darin bestehen, Taxis oder private Transportmittel wie Bolt in der Nähe zu finden. Dies kann auch hervorragend mit dem Geolocation Service kombiniert werden.

## 2.5 Freizeitpark-App

Dieser Service kann einem Spielpark angeboten werden. In großen Themenparks wie Praterpark (es gibt viele Parks, die viel größer sind) konnte der Benutzer seinen aktuellen Standort und die verschiedenen Spiele in seiner Nähe und im gesamten Park durch interaktives Zoomen sehen.

# 3 Potenzielle Kombinationen mit andere Microservices

## 3.1 People (Gruppe 10)

Andere Services könnten auf unser Service zugreifen, indem sie herausfinden, wieso die Menschenmasse gerade in diesem Bereich sich bewegen. Zuerst können wir nach der Adresse und dann nach der Kategorie wie z.B. Restaurants oder Bars filtern.

## 3.2 Geolocation Service (Gruppe 00)

Wir könnten ein Geolocation Microservice verwenden, um einen genaueren Standort festlegen zu können, mit Hilfe der erfragten Koordinaten und diese an den Benutzer weitersenden.

## 3.3 Charging Infrastructure Service (Gruppe 01)

Mit Enterprise Ressource Planung könnten wir nach E-Tankstellen kategorisieren, indem wir diese API verwenden und die Ergebnisse dem User ausgeben.

### 3.4 Weather (Gruppe 03)

Eine Kombination mit ein Wettervorhersage Service wäre auch möglich, wo der Service dem User zeigt, wie das Wetter in den angegebenen Standort sein wird. Es könnte sein, dass es am Standort eine Veranstaltung in einem offenen Bereich gibt, aber während der Veranstaltung wird es zum Regnen anfangen. Alert: Regenschirm mitbringen!

## 4 Realisierung der Services

Um unser Projekt zu realisieren, musste zuerst eine gewisse Planung stattfinden, in der wir besprochen haben, welche APIs wir nutzen werden. Hierbei haben sich die APIs Geoapify und Openstreetmap, als Favoriten herauskristallisiert. Diese beiden APIs werden hierfür benötigt, um Abfragen an diese zu senden und so einen Strom an Daten zu erlangen, den wir danach mittels unseres selbst erstellten Programms bearbeiten.

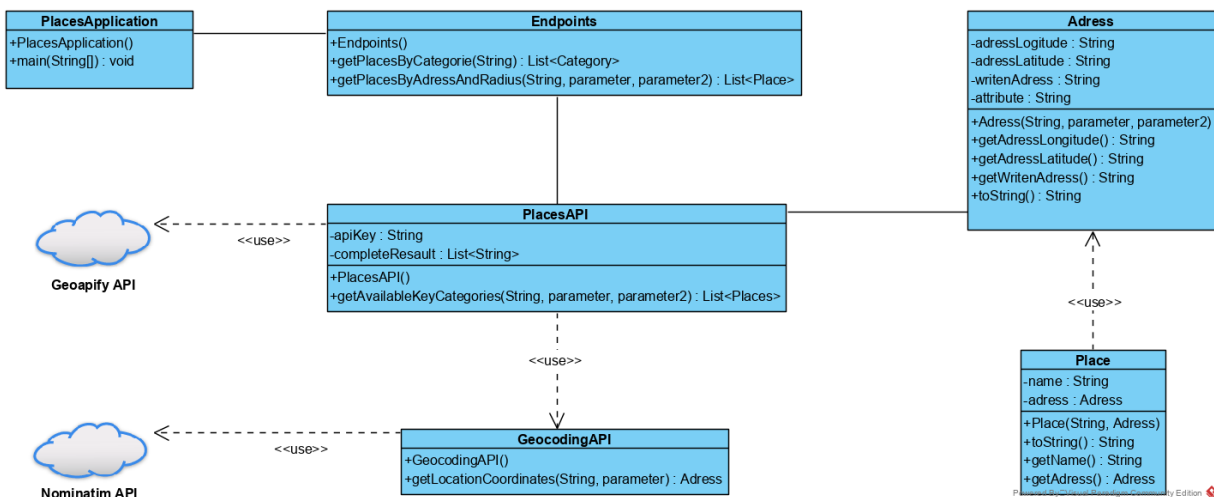


Abb. 1 UML-Places-Klassendiagramm

#### 4.1 places/byAdress/{streetName}/{houseNumber}/{radius}

Der Nutzer gibt den Straßennamen, Hausnummer und den Radius in dem gesucht werden soll an. Dabei übernimmt der Microservice die angegebenen Variablen und sendet sie an die Openstreetmap API, welche die genauen Koordinaten zurücksendet. Danach werden die Koordinaten verarbeitet und an die Geoapify API weitergesendet, welche uns dann eine Liste an Daten generiert mit der wir unsere benötigten Informationen erhalten. Hierbei werden dann die Kategorien, der in der Umgebung befindlichen Orte bereitgestellt.

#### 4.2 places/byCategorie/{streetName}/{houseNumber}/{radius}/{categorie}

Der Benutzer sendet eine, der bereits erhaltenen Kategorien von der vorherigen API, Kategorie an diesen Endpoint mit den vorherigen Daten und erhält alle Orte, die dieser Kategorie entsprechen in einer Liste zurück.

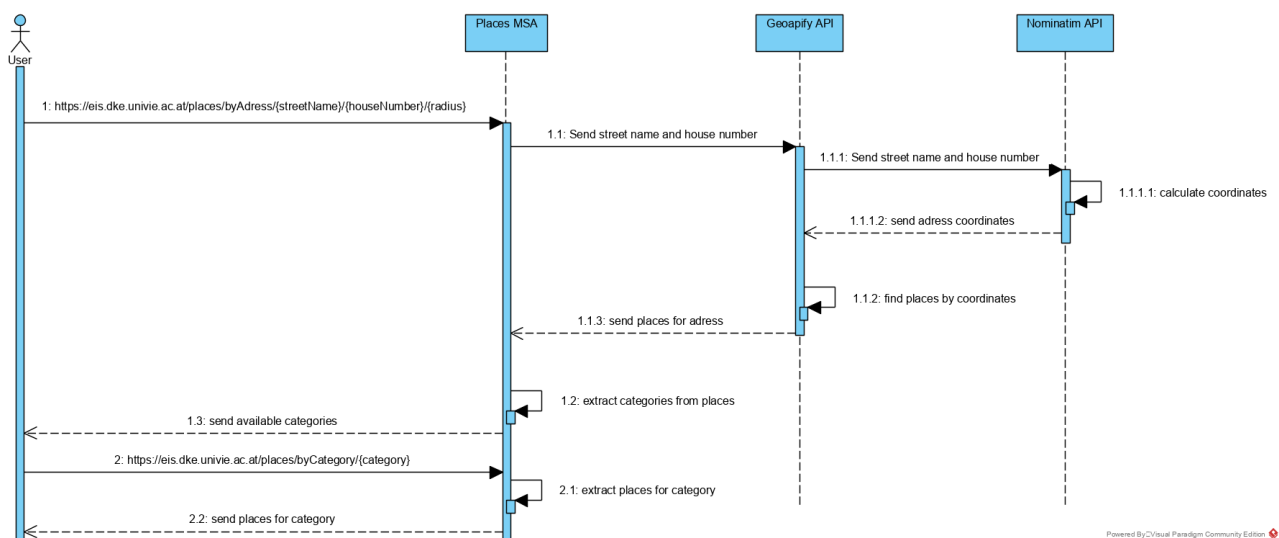


Abb. 2 UML-Places-Sequenzdiagramm

## 5 Arbeitsaufteilung

<b>Antonio Gasparevic (01506462)</b>	4,5,6
<b>Nemanja Srdanovic (01576891)</b>	4,5,6
<b>Kevin Çobaj (01635087)</b>	1,2,3

## 6 Conclusio

Der zur zeitige Implementationstand ist noch nicht fertiggestellt. Das oben beschriebene Design zu diesem Microservice repräsentiert unsere Vorstellung, wie dieses Service eingesetzt und implementiert werden könnte. Das bedeutet das im laufe der Implementierung noch weitere Features hinzukommen können oder aber auch weggelassen werden, da dieses Design noch nicht final ist.

## 7 Literaturverzeichnis

<https://nominatim.openstreetmap.org/>

<https://www.geoapify.com/>

## 8 Abbildungsverzeichnis

Abb. 1 UML-Places-Klassendiagramm .....	5
Abb. 2 UML-Places-Sequenzdiagramm .....	6