

# SoNah Parken API

(vorläufiger Masterarbeit Proposal)

## Introduction

SoNah ist ein Parkplatz Informationssystem, das Fahrern dazu hilft, einen Parkplatz schnell und unkompliziert in der Stadt zu finden. SoNah wertet Daten von an den Parkplätzen montierten Sensoren aus, um zunächst den Ist- Zustand der Parkplätze abzubilden. Ihre Stärke liegt darin, Anfragen auf verfügbare Parkplätze zu beantworten, die sich auf die Zukunft beziehen.

Der typische Anwendungsfall ist:

*Mirko fährt von A nach B. Er ist nicht nur daran interessiert schnell und sicher, geleitet von seinem Navi in B anzukommen, sondern auch in B unkompliziert zu parken. Mirko prüft den Stand der verfügbaren Plätze in B bereits beim Abfahrt, jedoch abhängig vom Reisedauer kann sich der Parksituation um 180° verändern. Deshalb braucht Mirko ein zuverlässiges und pfiffiges System, das ihm auf einen freien Parkplatz hinweist. Mirko probiert SoNah und stellt dann fest, dass SoNah diese Anforderungen völlig erfüllt.*

## Ziel der Masterarbeit

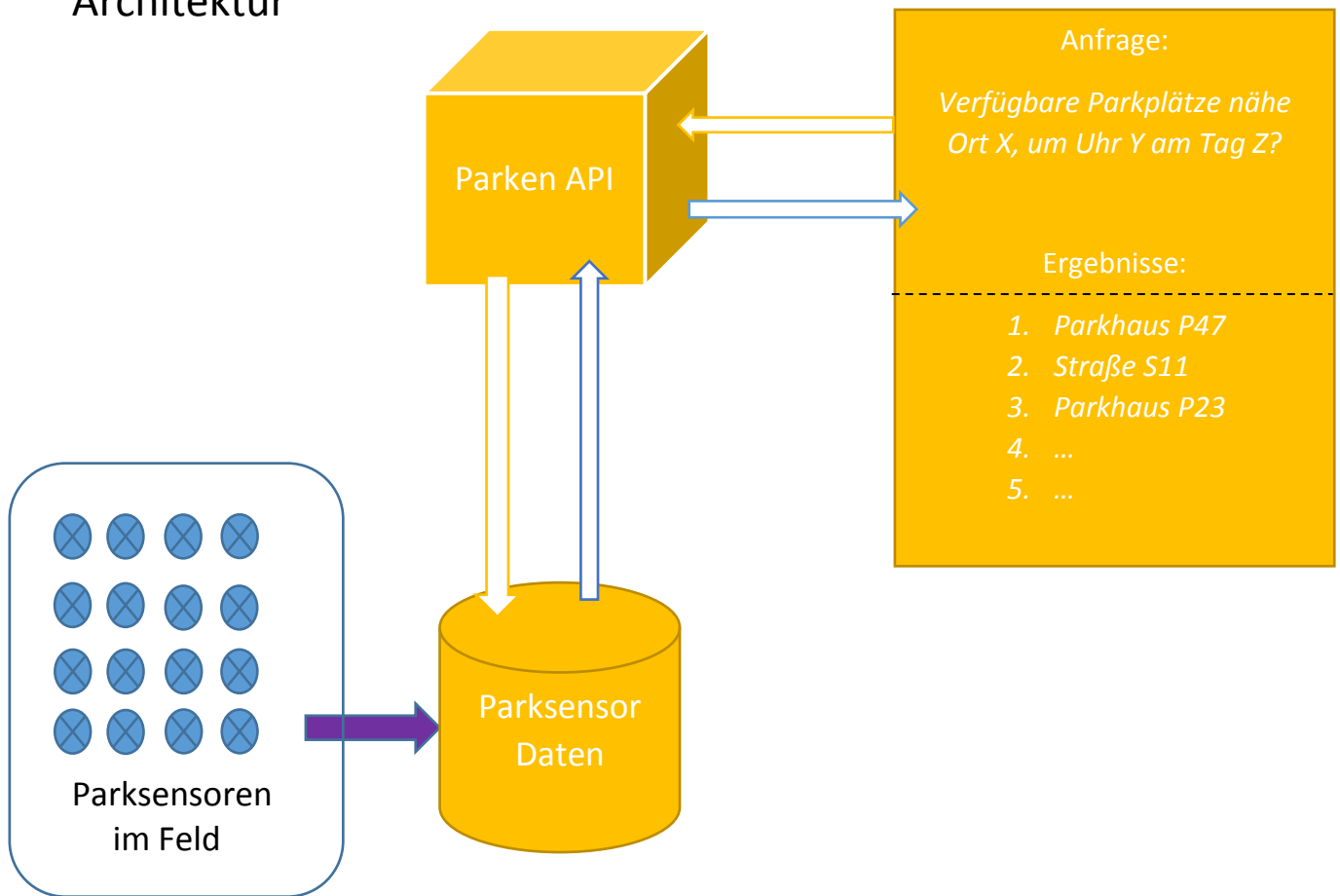
Ein System entwickeln, das Daten von Park-Sensoren so auswertet, dass Anfragen auf freie Parkplätze mit Bezug auf aktuellen und künftigen Zustand richtig und möglichst effizient beantwortet.

	Anfragen
<b>Parksensor</b>	<ul style="list-style-type: none"><li>Ist der Parkplatz X zurzeit frei?</li><li>Wie viele Plätze im Parkort* X sind frei?</li></ul>
<b>API</b>	<ul style="list-style-type: none"><li>Berechnet die Wahrscheinlichkeiten, dass ein Parkort mindestens einen Parkplatz frei hat in der Zukunft</li><li>Weist einen Parkplatz im Parkort X einem Auto zu, sodass die restlichen freien Parkplätze gleichmäßig verteilt bleiben</li></ul>
<b>User</b>	<ul style="list-style-type: none"><li>Wird mindestens ein Parkplatz im Parkort X in T Minuten frei?</li><li>Wie viele Parkplätze sind frei im Parkort X am Tag Y um die Uhrzeit Z?</li><li>Falls der Parkort aktuell mindestens ein Parkplatz verfügbar hat, bitte mir einen Parkplatz zuweisen</li></ul>

Figur 1: Funktionalität auf Ebenen

\*Parkort = Parkhaus, Parkplätze am Straßenrand, etc.

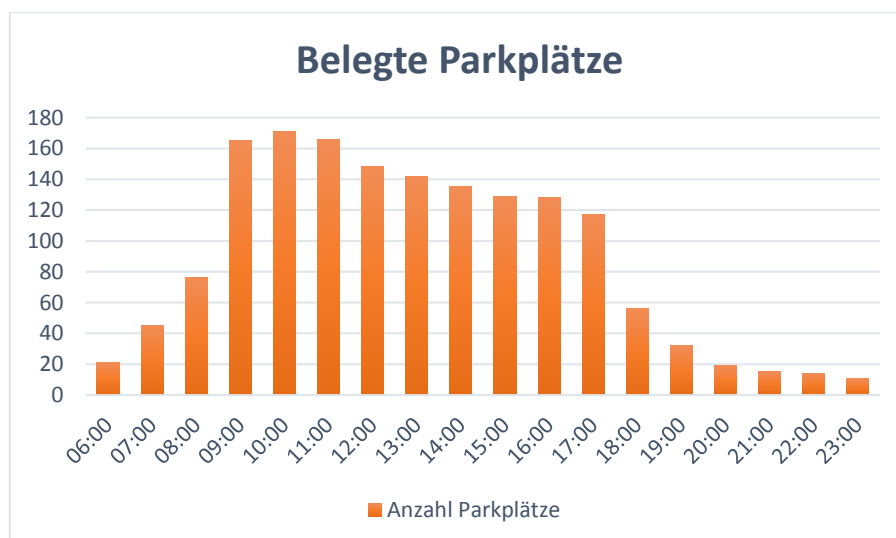
## Architektur



Figur 2: Systemkomponente und ihr Interagieren miteinander

## Stoßzeiten

Neben Antworten pro Anfrage wird das System auch Statistiken über bestimmte Parkorten liefern. Je nach Tag (Werktag, Wochenende, Feiertag) wird das System imstande sein eine komplette Vorhersage bereitzustellen.



Figur 3: Stoßzeiten-Diagramm für ein Parkhaus

## Zeitplan (vorläufig)

1. - 2. Woche	Proposal erstellen / Forschungsfrage formulieren
3. - 6. Woche	Literaturrecherche
6. - 9. Woche	Konzept aufbereiten
10. - 17. Woche	Implementierung
18. - 21. Woche	Evaluation
22. - 25. Woche	Abschließen

## Voraussichtlicher Konzept

### Theorie

Die Wahrscheinlichkeiten zu freie Parkplätze lassen sich als Markov Ketten modellieren. Lediglich mit berechneten Werten wie die bisherigen durchschnittlichen Einfahrttakt und Parkdauer lässt sich eine Matrix bauen, die der Kern der Lösung ist.

*Matrix  $Q$  ist  $(m+1) \times (m+1)$  groß, wobei  $m$  die Anzahl der Parkplätze am jeweiligen Parkort ist*

*$Q(s1, s2)$  = die Wahrscheinlichkeit dass gleich sich die Anzahl von  $s1$  freie Parkplätze auf  $s2$  freie Parkplätze ändert*

Durch die Einführung einer Zeit Variabel lassen sich dann die Wahrscheinlichkeiten berechnen, die beliebigen Zeitpunkten in der Zukunft entsprechen.

### Praxis

Das Modul wird voraussichtlich im Python oder Java implementiert. Steht es noch infrage, ob es als einfache Bibliothek oder als MVC Anwendung (in Django) umgesetzt wird.

## Evaluation

Anhand der angesammelten Parksensordaten werden erste Predictions geliefert. Das Testen wird auf einmal nicht die gesamte Zeitspanne der Daten benutzen, damit die Ergebnisse mit den später abgespeicherten Parkdaten verglichen werden können, e.g. die früheren 70% der Daten für Testen, die späteren 30% für die Evaluation.

Alternativ, bei unzureichenden Sensordaten lässt sich das Testen durch unabhängiges Simulieren einer kontinuierlichen Parksituation durchführen.

# Literaturrecherche

- [1] Predicting Parking Lot Occupancy in Vehicular Ad Hoc Networks, Murat Caliskan, Andreas Barthels, Björn Scheuermann, and Martin Mauve, 2007
- [2] Finding Available Parking Spaces Made Easy, Klappenecker, Andreas and Lee, Hyunyoung and Welch, Jennifer L., Ad Hoc Netw., 2014
- [3] Intelligent parking systems, D Teodorovic, P Lucic, European J. Operational Research, 2006
- [4] Learning the Relevance of Parking Information in VANETs, Piotr Szczurek, Bo Xu, Ouri Wolfson, Jie Lin, Naphtali Rishe, 2010
- [5] ParkNet: Drive-by Sensing of Road-Side Parking Statistics, Suhas Mathur, Tong Jin, Nikhil Kasturirangan, Janani Chandrashekharan, Wenzhi Xue, Marco Gruteser, Wade Trappe, 2010
- [6] PocketParker: Pocketsourcing Parking Lot Availability, Anandatirtha Nandugudi, Taeyeon Ki, Carl Nuessle, and Geoffrey Challen, 2014
- [7] Smart Parking: an Application of optical Wireless Sensor Network, Jatuporn Chinrungrueng, Udomporn Sunantachaikul, Satien Triamlumlerd, 2007
- [8] SPARK: A New VANET-Based Smart Parking Scheme for Large Parking Lots, R Lu, X Lin, H Zhu, X Shen, INFOCOM 2009
- [9] VANET-based approach for parking space availability, R Panayappan, J M Trivedi, A Studer, A Perrig, 2007