

Railwatch

Für bessere Ansicht auf [Github](#) ansehen.

Funktionsbeschreibung

Wir wollen eine Terminal-App realisieren, die mithilfe einer API Zugverbindungen auflisten soll.

<https://transport.opendata.ch/>

Das ist die API, die wir benutzen werden. Ein User muss die Startstation und die Zielstation und freiwillig noch die Uhrzeit angeben, danach werden ihm die nächsten Verbindungen angezeigt. Wir werden diese App in Java entwickeln.

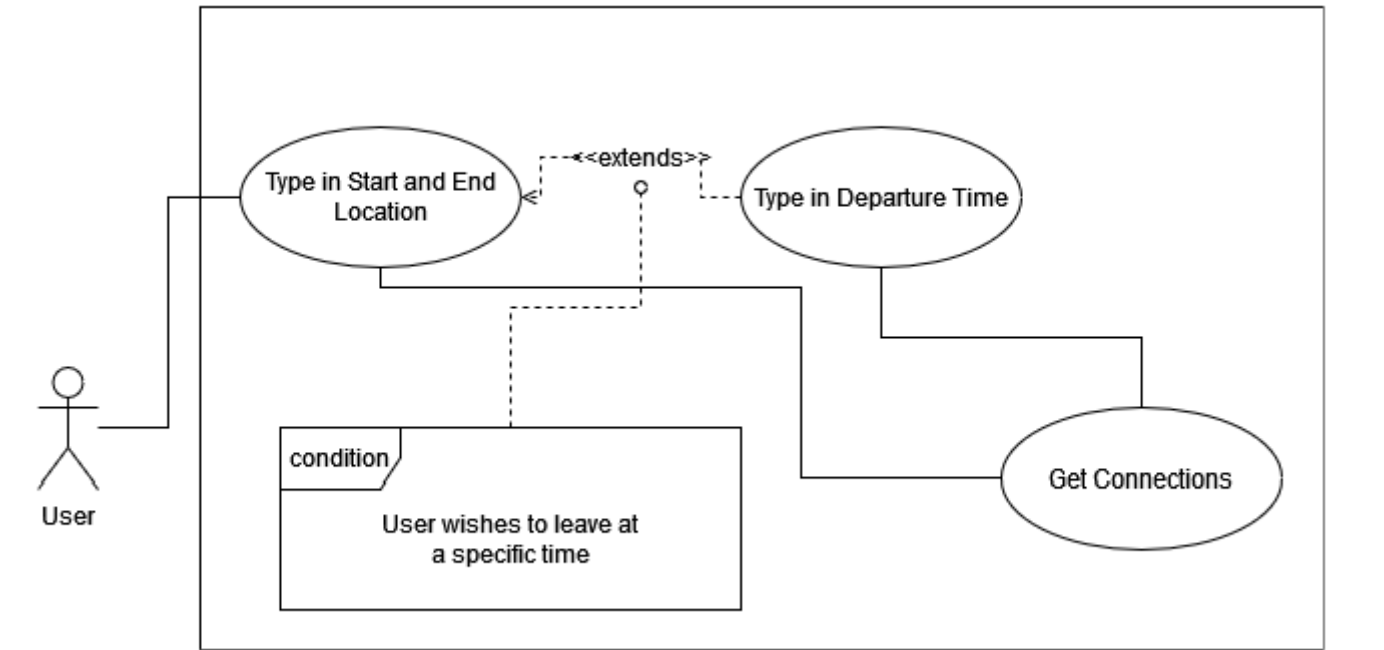
Schlussendlich soll die App so aussehen:

```
-----Railwatch-----
Type in the Departure Station:
Zürich HB
Type in the Station you're heading to:
Bonstetten
Type in the Time you want to depart:
12:00

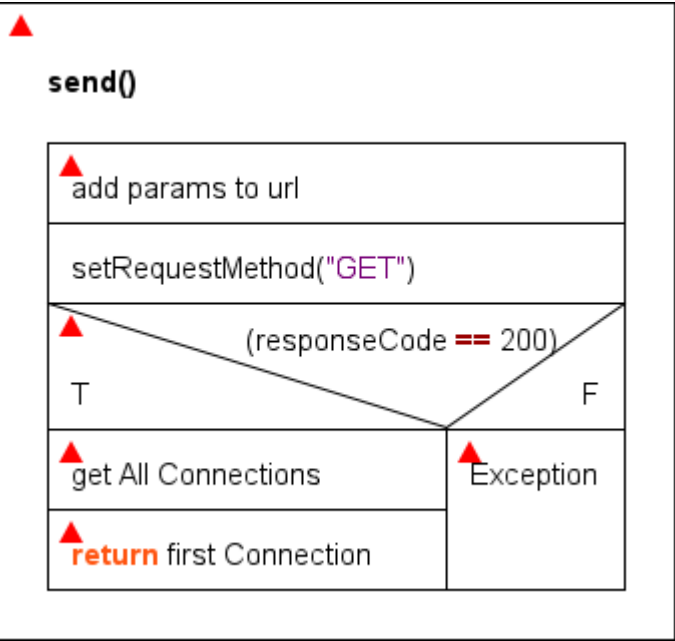
-----
Connections
-----
Departure Time ---- Journey Length ---- Platform
12:02      ----      0:21      ----      8
12:09      ----      0:21      ----     32
-----
```

Diagramme

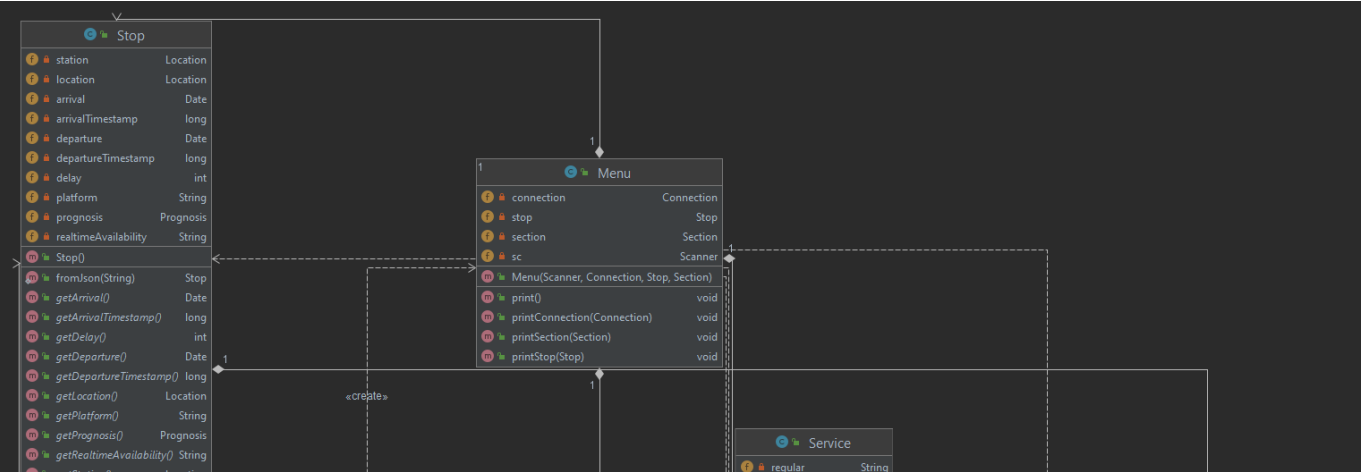
Use Case

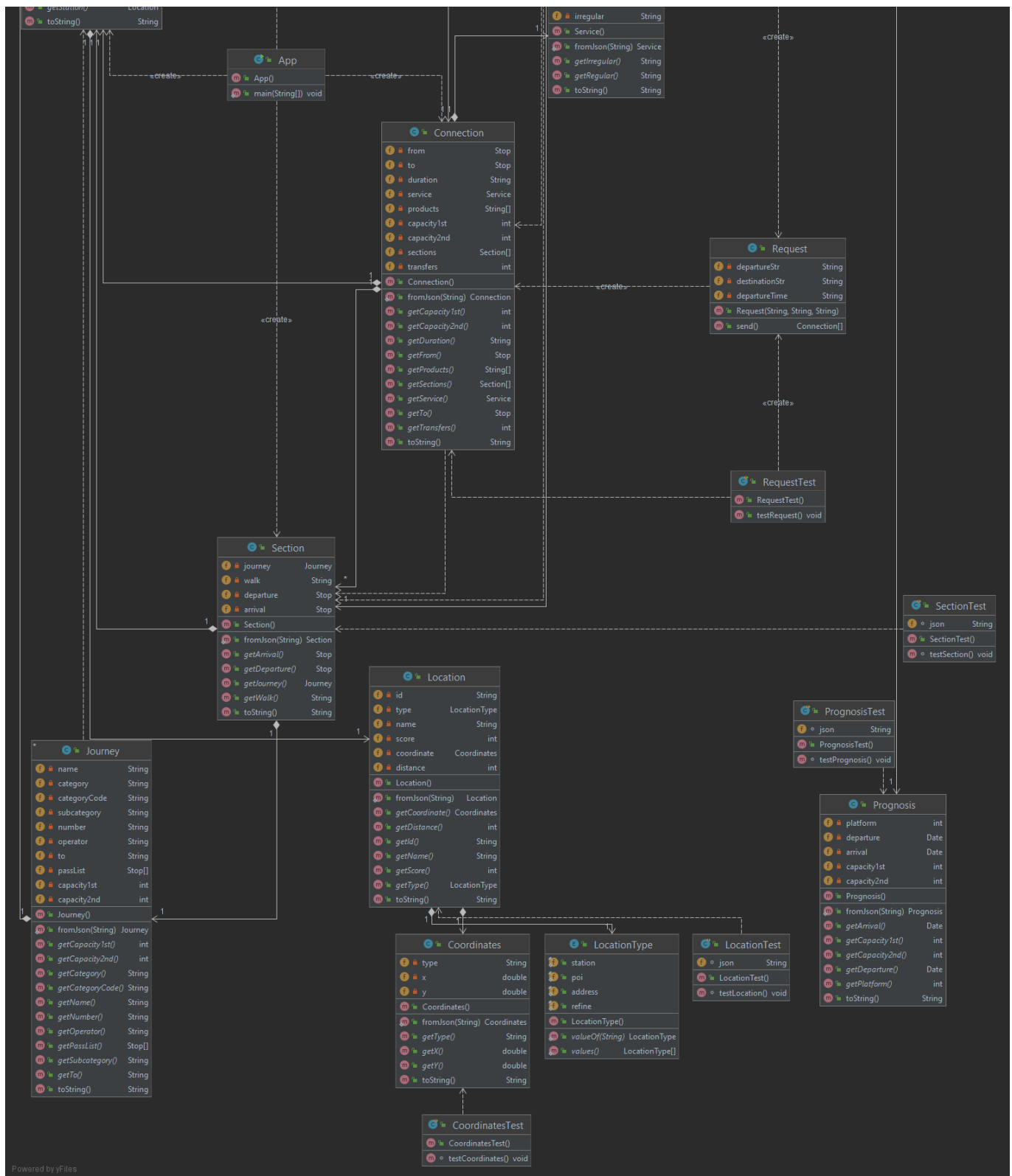


Struktogramm

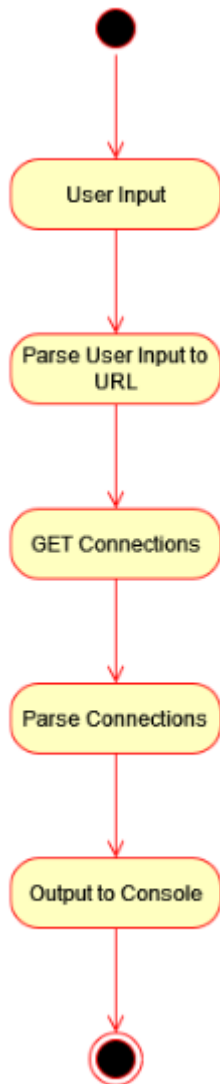


Klassendiagramm





Aktivitätsdiagramm



Testprotokoll

Funktionstests

Case	Eingabe	Erwartet	Tatsächlich	OK/NOK
Zeigt Menü	Programmstart	Zeigt Menü	Zeigt Menü	OK
Liest Startpunkt	Zürich	Zürich HB als Startpunkt	Zürich HB als Startpunkt	OK
Liest Ziel	Bern	Bern als Ziel	Bern als Ziel	OK
Liest Zeit	11:55	11:55 als Startzeit	11:55 als Startzeit	OK
Liest keine Zeit		Aktuelle Zeit als Startzeit	Aktuelle Zeit als Startzeit	OK
Keine Ergebnisse	Beliebiger inexistenter Ort	Keine Ergebnisse Meldung	Keine Ergebnisse Meldung	OK
Tab wird gefiltert	(TAB)	Keine gültige Eingabe	Keine gültige Eingabe	OK

Unit Tests

Coordinates: AssertEquals mit Getter überprüfen

JSON Attribut	Eingabe	Erwartet	Tatsächlich	OK/NOK
id	WGS84	Passed	Passed	OK
Coordinate X	47.391361	Passed	Passed	OK
Coordinate Y	8.051284	Passed	Passed	OK

Location: AssertEquals mit Getter überprüfen

JSON Attribut	Eingabe	Erwartet	Tatsächlich	OK/NOK
id	8502113	Passed	Passed	OK
name	Aarau	Passed	Passed	OK
score	null	Passed	Passed	OK
Coordinate Type	WGS84	Passed	Passed	OK
Coordinate X	47.391361	Passed	Passed	OK
Coordinate Y	8.051284	Passed	Passed	OK
distance	null	Passed	Passed	OK

Location: AssertEquals mit Getter überprüfen

JSON Attribut	Eingabe	Erwartet	Tatsächlich	OK/NOK
platform	10	Passed	Passed	OK
arrival	2022-03-31T08:58:00+02:00	Passed	Passed	OK
departure	2022-03-31T09:00:00+02:00	Passed	Passed	OK
Capacity1st	0	Passed	Passed	OK
Capacity2nd	-1	Passed	Passed	OK

Section: AssertNull mit Getter überprüfen

JSON Attribut	Eingabe	Erwartet	Tatsächlich	OK/NOK
walk	null	Passed	Passed	OK

Nutzung

Bei diesem Projekt wurde Git benutzt. Das Projekt kann mittels **git checkout** jederzeit an einem beliebigen Zeitpunkt angesehen werden.

Voraussetzungen

- Maven 3.8
- Java 17
- Internetverbindung

Applikation ausführen

```
mvn clean  
mvn package  
mvn exec:java
```

Tests ausführen

```
mvn test
```

Maven-Projektübersicht generieren

Generiert eine Projektübersicht, inklusive Apidocs.

```
mvn site
```

Die Übersicht ist zu finden unter <target/site/index.html>, die Apidocs befinden sich in <target/site/apidocs/index.html>.