



ÜK 318

# OBJEKTBASIERT PROGRAMMIEREN

Buchs Noah

22.04.22



## Table of Contents

<b>Einleitung .....</b>	<b>2</b>
<b>Zweck .....</b>	<b>2</b>
<b>Userstories.....</b>	<b>2</b>
<b>Userstory 1: Start- und Endstation eingeben -&gt; 4 Verbindungen .....</b>	<b>2</b>
Beschreibung .....	2
Abnahmekriterien .....	2
Priorität.....	2
<b>Userstory 2: Stationssuche .....</b>	<b>2</b>
Beschreibung .....	2
Abnahmekriterien .....	2
Priorität.....	2
<b>Userstory 3: Auto-Completion .....</b>	<b>2</b>
Beschreibung .....	2
Abnahmekriterien .....	2
Priorität.....	2
<b>Userstory 4: Abfahrtstafel .....</b>	<b>3</b>
Beschreibung .....	3
Abnahmekriterien .....	3
Priorität.....	3
<b>Userstory 5: Zukünftige Verbindungen .....</b>	<b>3</b>
Beschreibung .....	3
Abnahmekriterien .....	3
Priorität.....	3
<b>Userstory 6: Standort der Station .....</b>	<b>3</b>
Beschreibung .....	3
Abnahmekriterien .....	3
Priorität.....	3
<b>Userstory 7: Stationen in der Nähe .....</b>	<b>3</b>
Beschreibung .....	3
Abnahmekriterien .....	3
Priorität.....	3
<b>Userstory 8: Zugverbindungen per Mail verschicken.....</b>	<b>4</b>
Beschreibung .....	4
Abnahmekriterien .....	4
Priorität.....	4
<b>Userstory 9: Anzeige der Resultate .....</b>	<b>4</b>
Beschreibung .....	4
Abnahmekriterien .....	4
Priorität.....	4
<b>Aktivitätsdiagramm .....</b>	<b>4</b>
<b>Mock-Ups .....</b>	<b>5</b>
<b>Testfälle.....</b>	<b>6</b>
<b>Testplan .....</b>	<b>6</b>
Testfall 1: Verbindungen.....	6
Testfall 2: Stationssuche .....	6
Testfall 3: Abfahrtstafel .....	6
Testfall 4: Auto-Completion.....	7
Testfall 5: Zukünftige Verbindung.....	7
Testfall 6: Verbindung über Mail verschicken .....	8
Testfall 7: Standort der Station.....	8
<b>Testprotokoll .....</b>	<b>9</b>
Testfall 1: Verbindungen.....	9
Testfall 2: Stationssuche .....	9
Testfall 3: Abfahrtstafel .....	10
Testfall 4: Auto-Completion.....	10
Testfall 5: Zukünftige Verbindung.....	11
Testfall 6: Verbindung über Mail verschicken .....	11
Testfall 7: Standort der Station.....	12
<b>Installationsanleitung .....</b>	<b>12</b>
1. <b>Möglichkeit .....</b>	<b>12</b>
2. <b>Möglichkeit .....</b>	<b>13</b>
<b>Anderes .....</b>	<b>13</b>

## Einleitung

In diesem ÜK habe ich eine JavaFx-Applikation programmiert, welche den Nutzen hat, Verbindungen und Stationen des öffentlichen Verkehrs anzuzeigen. Es gibt auch weitere Möglichkeiten, wie die Station auf Google Maps zu sehen, Verbindungen um eine gewisse Zeit sehen und die Verbindung via Mail zu versenden.

## Zweck

Der Zweck der Dokumentation ist es, eine Übersicht über den Prozess und die dazugehörigen Anforderungen zu sehen. Auch werden Testfälle definiert und eine Installationsanleitung.

## Userstories

### Userstory 1: Start- und Endstation eingeben -> 4 Verbindungen

#### Beschreibung

Als Nutzer möchte ich eine Start- und Endstation eingeben können, um die vier nächstmöglichen Verbindungen für die Strecke zu bekommen.

#### Abnahmekriterien

- > Der Nutzer kann eine Startstation eingeben
- > Der Nutzer kann eine Endstation eingeben
- > Es werden die vier Verbindungen ausgegeben
- > Es wird eine Auflistung der vier nächsten Verbindungen angezeigt

#### Priorität

1 >>> must

### Userstory 2: Stationssuche

#### Beschreibung

Als Nutzer möchte ich in eine Eingabebox eine Station «anfangen» einzugeben und dann alle Stationen welche gleich anfangen ausgeben

#### Abnahmekriterien

- > Es werden alle Stationen angezeigt, welche gleich anfangen

#### Priorität

1 >>> must

### Userstory 3: Auto-Completion

#### Beschreibung

Als Nutzer möchte ich Vorschläge zu meiner Eingabe bekommen, falls ich den ganzen Namen der Station nicht kenne.

#### Abnahmekriterien

- > Es sollte entsprechend der Eingabe passende Vorschläge angezeigt werden
- > Es sollten mehrere Vorschläge kommen

#### Priorität

2 >>> should

## Userstory 4: Abfahrtstafel

### Beschreibung

Als Nutzer möchte ich eine Station eingeben können, um von dieser alle Verbindungen zu bekommen

### Abnahmekriterien

- > Es sollte eine Liste aller Verbindung der Station angezeigt werden

### Priorität

1 >>> must

## Userstory 5: Zukünftige Verbindungen

### Beschreibung

Als Nutzer möchte ich ein Datum und eine Uhrzeit angeben können, um auch zukünftige Verbindungen anzusehen.

### Abnahmekriterien

- > Es sollte ein Datum und eine Uhrzeit eingegeben werden
- > Entsprechend der angegebenen Zeit sollten die Verbindungen angezeigt werden

### Priorität

2 >>> should

## Userstory 6: Standort der Station

### Beschreibung

Als Nutzer möchte ich auf einer Karte die gewählte Station sehen, um zu wissen, wo diese liegt

### Abnahmekriterien

- > Die Station sollte eingegeben werden und dann auf der Karte angezeigt werden
- > Es sollte neben der angezeigten Station ein Knopf geben, bei welcher man es auf der Karte suchen kann

### Priorität

3 >>> nice to have

→ Dieses Feature ist nur bei der Locations-Seite vorhanden

## Userstory 7: Stationen in der Nähe

### Beschreibung

Als Nutzer möchte ich meinen aktuellen Standort und alle Stationen in der Nähe sehen, um die praktischste dann zu wählen.

### Abnahmekriterien

- > Der aktuelle Standort des Nutzers sollte angezeigt werden
- > Stationen im Umfeld von etwa einem Kilometer oder die 5 nächsten Stationen sollten angezeigt werden
- > Die Stationen sollten auch angeklickt werden können

### Priorität

3 >>> nice to have

→ Diese Userstory wurde nicht umgesetzt

## Userstory 8: Zugverbindungen per Mail verschicken

### Beschreibung

Als Nutzer möchte ich meine gewählte Verbindung jemanden via Mail weiterleiten könnte, um die andere Person darüber zu informieren

### Abnahmekriterien

- > Der Nutzer sollte seine Eingabe in ein Mail, via Knopfdruck, exportieren können

### Priorität

3 >>> nice to have

## Userstory 9: Anzeige der Resultate

### Beschreibung

Als Nutzer möchte ich eine deutliche Anzeige meiner Such-Resultaten, um Informationen aus dieser deutlich zu lesen können.

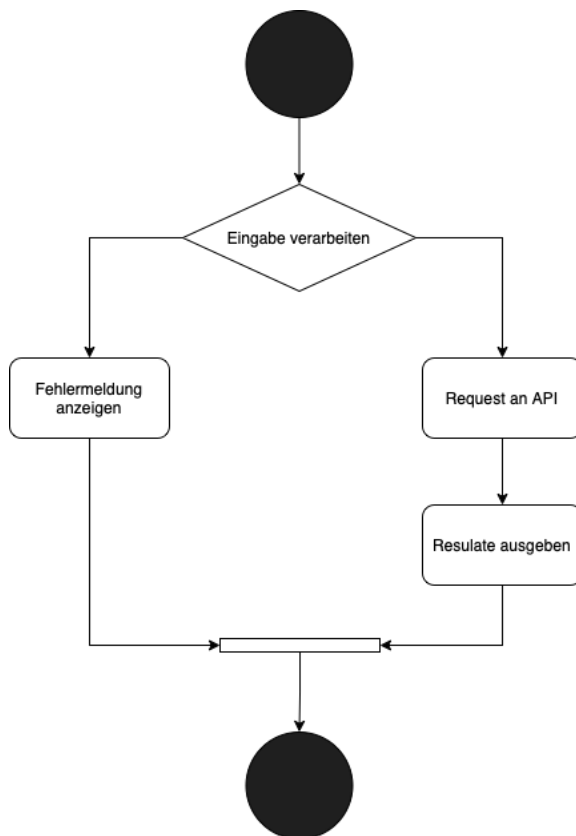
### Abnahmekriterien

- > Wenn vorhanden → Zug / Bus / Tram sollte angezeigt werden
- > Jedes Resultat hat einen bestimmten Aufbau (Grid)

### Priorität

3 >>> nice to have

## Aktivitätsdiagramm

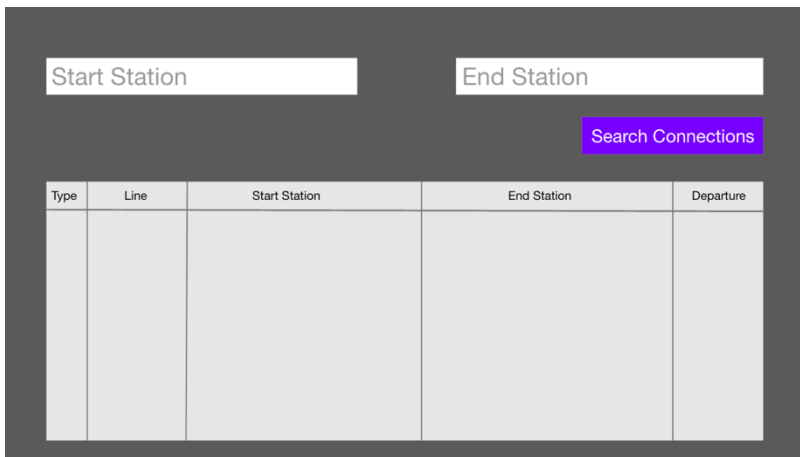


### Aktivitätsdiagramm der Userstory 2

Der User gibt etwas in das Eingabefeld ein. Beim Knopfdruck auf Station Suchen wird die Eingabe verwaltet.

Falls die Eingabe leer ist, wird eine Fehlermeldung angezeigt. Wenn nicht, dann wird eine Request an die API geschickt und die Response wird dann in einer Liste ausgegeben.

## Mock-Ups



Start Station

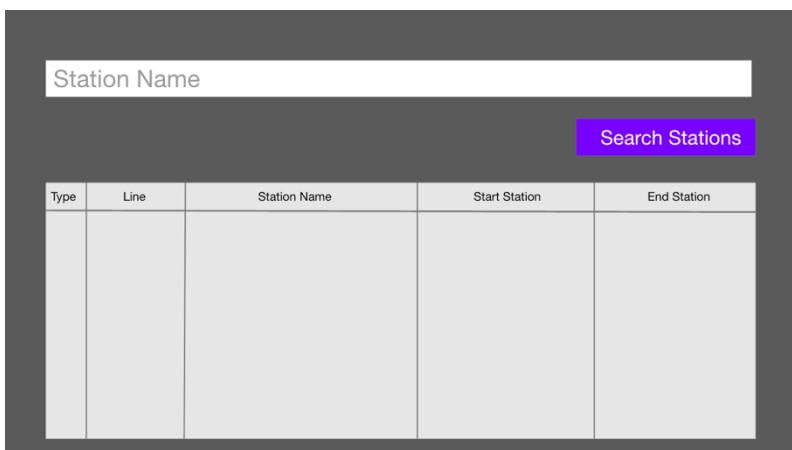
End Station

Search Connections

Type	Line	Start Station	End Station	Departure

Hier sieht man ein Mock-Up der Connections-Seite.

Es gibt zwei Input Felder für die Start und Stop Station, einen Knopf für die Suche und eine Liste mit den Verbindungen



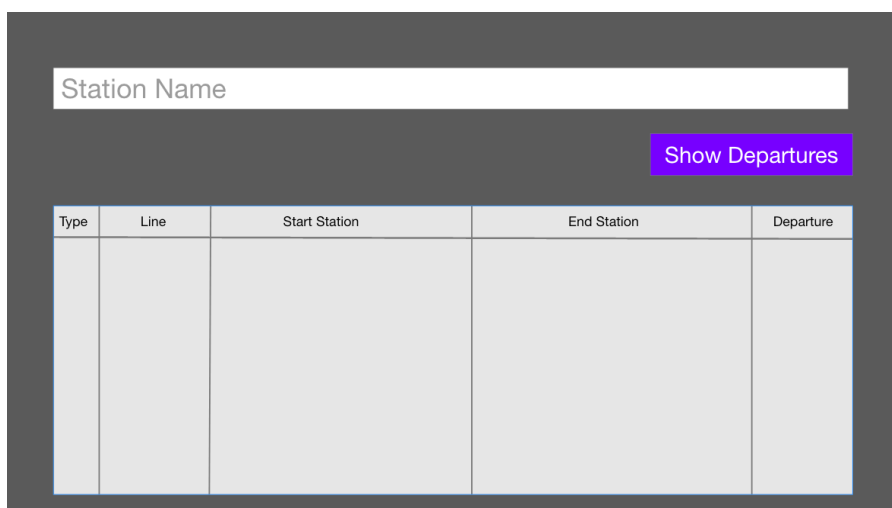
Station Name

Search Stations

Type	Line	Station Name	Start Station	End Station

Hier sieht man ein Mock-Up der Locations-Seite.

Es gibt ein Input Feld für die gesuchte Station, einen Knopf für die Suche und eine Liste mit den Stationen



Station Name

Show Departures

Type	Line	Start Station	End Station	Departure

Hier sieht man ein Mock-Up der StationBoard-Seite.

Es gibt ein Input Feld für den Station-Namen, einen Knopf für die Suche und eine Liste mit den Verbindungen, welche dort halten.

## Testfälle

### Testplan

#### Testfall 1: Verbindungen

Vorbedingungen: Programm offen, Connections-Page offen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Erfüllt
1	In "From"-Feld wird Luzern eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	
2	In "To"-Feld wird Basel eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	
3	Der "Search"-Knopf wird gedrückt	Nach kurzem warten wird eine Liste mit den nächsten vier Verbindungen angezeigt.	

#### Testfall 2: Stationssuche

Vorbedingungen: Programm offen, Locations-Page offen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Erfüllt
1	In "Station Name"-Feld wird Luzern eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	
3	Der "Search"-Knopf wird gedrückt	Nach kurzem warten wird eine Liste mit allen Stationen angezeigt, welche den Input-Text in sich haben	

#### Testfall 3: Abfahrtstafel

Vorbedingungen: Programm offen, StationBoard-Page offen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Erfüllt
1	In "Station Name"-Feld wird Luzern eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	
3	Der "Search"-Knopf wird gedrückt	Nach kurzem warten wird eine Liste mit allen Verbindungen angezeigt, welche an der bestimmten Station halten.	

#### Testfall 4: Auto-Completion

Vorbedingungen: Programm offen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Erfüllt
1	In ein Input-Feld wird «Bas» eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt. Einer der Vorschläge sollte Basel sein.	
3	Es wird auf den Vorschlag «Basel» gedrückt	Der gedrückte Vorschlag sollte dann in das entsprechende Input-Feld übernommen werden und die Liste sollte verschwinden	

#### Testfall 5: Zukünftige Verbindung

Vorbedingungen: Programm offen, Connections-Page offen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Erfüllt
1	In "From"-Feld wird Luzern eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	
2	In "To"-Feld wird Basel eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	
3	Der "Advanced"-Knopf wird gedrückt	Es werden zwei weitere Felder angezeigt, in welchen man die Zeit eingeben kann	
4	Es wird die Zeit «15:00» eingegeben	Die eingegebene Zeit wird im entsprechenden Feld übernommen und angezeigt.	
5	Es wird das Datum «28.05.22» eingegeben	Das eingegebene Datum wird im entsprechenden Feld übernommen und angezeigt.	
6	Der «Search»-Knopf wird gedrückt	Es wird eine Liste mit den nächsten vier Verbindungen im angegebenen Zeitraum angezeigt	



#### Testfall 6: Verbindung über Mail verschicken

Vorbedingungen: Programm offen, Connections-Page offen, angezeigte Verbindungen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Erfüllt
1	Es wird der «Send via Mail»-Knopf gedrückt	Es öffnet sich ein PopUp, bei welchem man die E-Mail-Adresse eingeben kann	
2	Es wird «test@test.com» eingegeben	Es steht genau das, was eingegeben wurde	
3	Der Knopf «Send Mail» wird gedrückt	Es öffnet sich ein Mail, bei welchem alle Informationen der Verbindung vorhanden sind. Der User muss nur noch senden drücken.	

#### Testfall 7: Standort der Station

Vorbedingungen: Programm offen, Locations-Page offen, angezeigte Stationen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Erfüllt
1	Es wird der Knopf «View on Maps» bei einer Station gedrückt	Der Nutzer wird auf Google Maps weitergeleitet, wo er dann den Standort der gewählten Station sieht	

## Testprotokoll

Dokumentversion: 1.0

Datum: 22.04.22

Testperson: Noah Buchs

App-Version: v1.0

Installation: siehe Kapitel *Installationsanleitung*

### Testfall 1: Verbindungen

Vorbedingungen: Programm offen, Connections-Page offen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Resultat	Erfüllt
1	In "From"-Feld wird Luzern eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	=	Ja
2	In "To"-Feld wird Basel eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	=	Ja
3	Der "Search"-Knopf wird gedrückt	Nach kurzem warten wird eine Liste mit den nächsten vier Verbindungen angezeigt.	=	Ja

### Testfall 2: Stationssuche

Vorbedingungen: Programm offen, Locations-Page offen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Resultat	Erfüllt
1	In "Station Name"-Feld wird Luzern eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	=	Ja
3	Der "Search"-Knopf wird gedrückt	Nach kurzem warten wird eine Liste mit allen Stationen angezeigt, welche den Input-Text in sich haben	=	Ja

### Testfall 3: Abfahrtstafel

Vorbedingungen: Programm offen, StationBoard-Page offen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Resultat	Erfüllt
1	In "Station Name"-Feld wird Luzern eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	=	Ja
3	Der "Search"-Knopf wird gedrückt	Nach kurzem warten wird eine Liste mit allen Verbindungen angezeigt, welche an der bestimmten Station halten.	=	Ja

### Testfall 4: Auto-Completion

Vorbedingungen: Programm offen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Resultat	Erfüllt
1	In ein Input-Feld wird «Bas» eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt. Einer der Vorschläge sollte Basel sein.	=	Ja
3	Es wird auf den Vorschlag «Basel» gedrückt	Der gedrückte Vorschlag sollte dann in das entsprechende Input-Feld übernommen werden und die Liste sollte verschwinden	=	Ja

### Testfall 5: Zukünftige Verbindung

Vorbedingungen: Programm offen, Connections-Page offen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Resultat	Erfüllt
1	In "From"-Feld wird Luzern eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	=	Ja
2	In "To"-Feld wird Basel eingegeben	Während dem Eingeben wird eine Liste angezeigt, welche Stations-Vorschläge angepasst an den Input anzeigt.	=	Ja
3	Der "Advanced"-Knopf wird gedrückt	Es werden zwei weitere Felder angezeigt, in welchen man die Zeit eingeben kann	=	Ja
4	Es wird die Zeit «15:00» eingegeben	Die eingegebene Zeit wird im entsprechenden Feld übernommen und angezeigt.	=	Ja
5	Es wird das Datum «28.05.22» eingegeben	Das eingegebene Datum wird im entsprechenden Feld übernommen und angezeigt.	=	Ja
6	Der «Search»-Knopf wird gedrückt	Es wird eine Liste mit den nächsten vier Verbindungen im angegebenen Zeitraum angezeigt	=	Ja

### Testfall 6: Verbindung über Mail verschicken

Vorbedingungen: Programm offen, Connections-Page offen, angezeigte Verbindungen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Resultat	Erfüllt
1	Es wird der «Send via Mail»-Knopf gedrückt	Es öffnet sich ein PopUp, bei welchem man die E-Mail-Adresse eingeben kann	=	Ja
2	Es wird «test@test.com» eingegeben	Es steht genau das, was eingegeben wurde	=	Ja
3	Der Knopf «Send Mail» wird gedrückt	Es öffnet sich ein Mail, bei welchem alle Informationen der Verbindung vorhanden sind. Der User muss nur noch senden drücken.	=	Ja

## Testfall 7: Standort der Station

Vorbedingungen: Programm offen, Locations-Page offen, angezeigte Stationen, Internetverbindung

Schritt	Aktivität	Erwartetes Resultat	Resultat	Erfüllt
1	Es wird der Knopf «View on Maps» bei einer Station gedrückt	Der Nutzer wird auf Google Maps weitergeleitet, wo er dann den Standort der gewählten Station sieht.	=	Ja

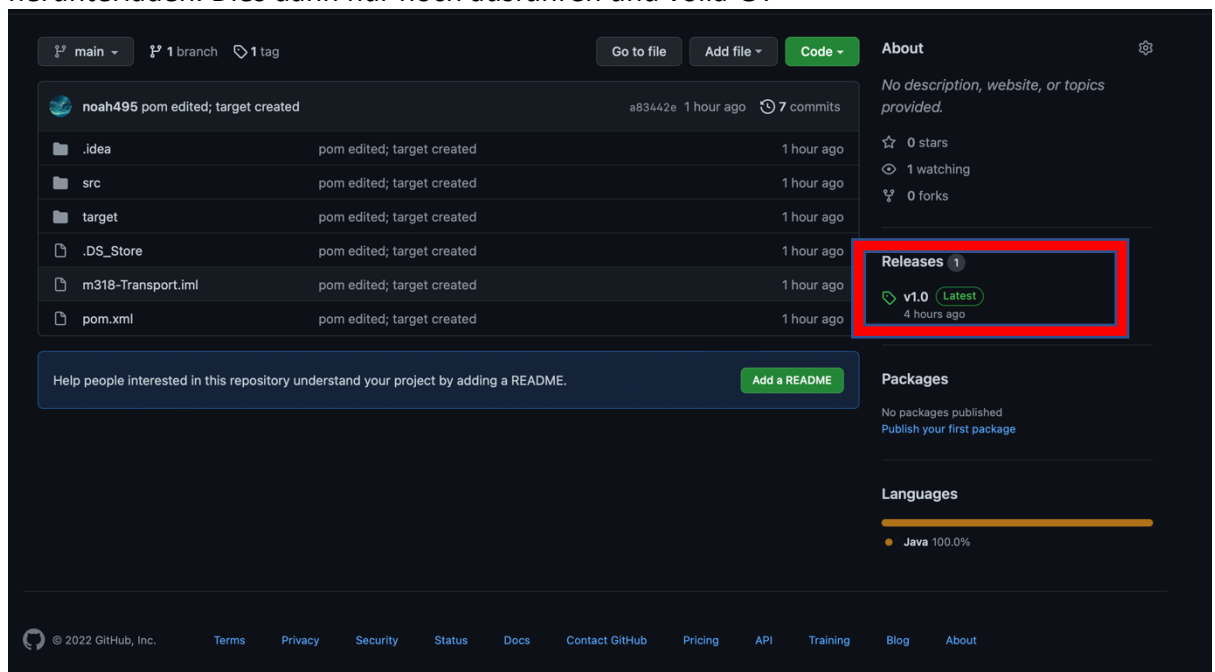
## Installationsanleitung

Es ist sehr wichtig, dass die neuste empfohlene Version von Java heruntergeladen ist. Wenn dies nicht zustimmt, kann es via diesen Link installiert werden → [Download](#)

Als nächstes kann man auf zwei verschiedene Arten das .jar «holen». Für beide muss man auf GitHub gehen und auf mein Projekt (Link: <https://github.com/noah495/m318-UeK> )

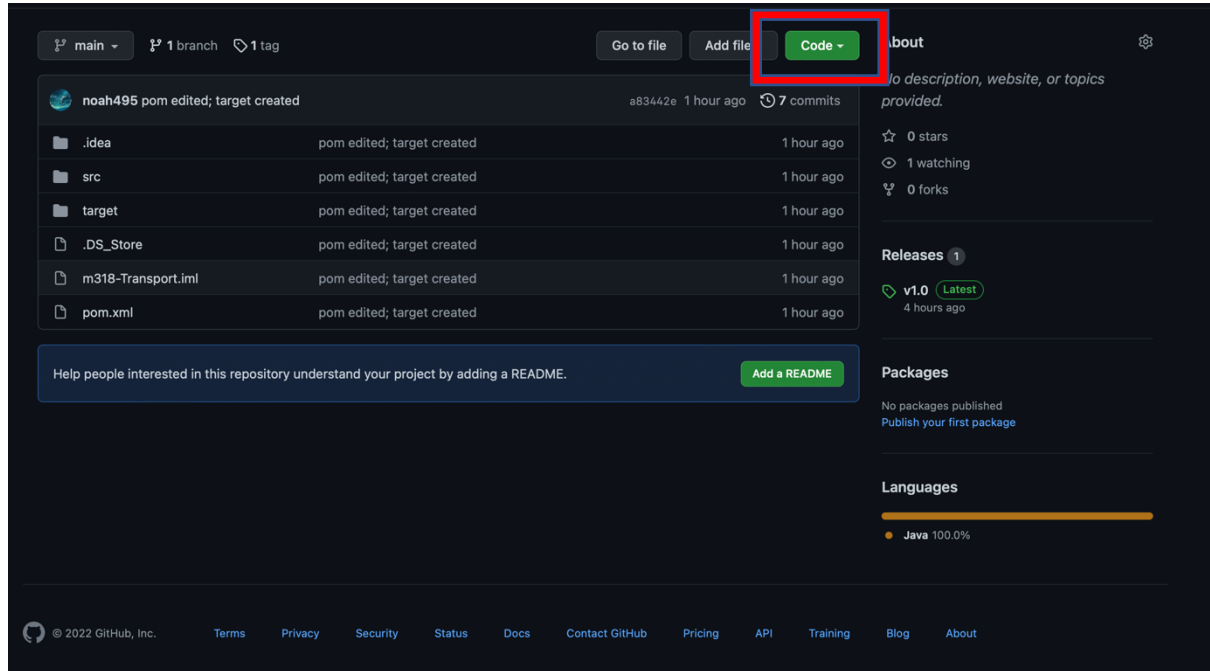
### 1. Möglichkeit

Auf den Link drücken und dann auf Releases gehen und das dort hochgeladene .jar herunterladen. Dies dann nur noch ausführen und voila 😊.



## 2. Möglichkeit

Auf den Link drücken und dort den Code klonen (entweder via Cmd oder als .zip downloaden).



Das Projekt dann in dem gewünschten Editor öffnen, in meinem Fall IntelliJ.  
Dort dann im Terminal «maven clean install» oder «mvn clean install».  
Falls Maven nicht vorhanden ist, kann es hier heruntergeladen werden  
→ <https://maven.apache.org/download.cgi>

Sobald man «maven clean install» ausgeführt, sollte man einen target Ordner finden.  
In diesem sieht man dann zwei .jar-Files. Es ist wichtig, dass man das File «...-with-dependencies» ausführt.

Viel Glück :)

## Anderes

Das Maps-Feature ist nur bei Locations vorhanden, da ich es dort als am nötigsten gehalten habe.