**Project**: Business Case demo shared mobility

**Version:** 1.0

This document will entail all the necessary information to successfully implement the given project.

It is divided in two parts; the functional documentation which describes what it is our team is building and the technical documentation, which describes how our program will be build.

Contents

[1 Document Properties 2](#_Toc628857)

[1 Functional documentation 3](#_Toc628858)

[1.1 Project Scope 3](#_Toc628859)

[1.2 Real World 3](#_Toc628860)

[1.3 Entity Relationship model (ER) 4](#_Toc628861)

[1.4 Entitty Relationship diagram (ERD) 5](#_Toc628862)

[2 Technical documentation 5](#_Toc628863)

# Document Properties

#### Properties at the start of the document

|  |  |
| --- | --- |
| Authors | Jorden D’Hulster |
| Date | 26 feb 2019 |

#### History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Author | Description |
| 26 februari 2019 | 1.0 | Jorden D’Hulster | Shared Mobility |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

#### Actual status

|  |  |
| --- | --- |
| Status | Draft |

# Functional documentation

This section describes what the software will do we are building.

## Project Scope

The project scope aims for the fact that we want to make it easier for the students to travel to several events. we want to write a program that allows students to see who goes to an event how and that it is possible to travel together in this way.

### Project justification

This project facilitateds co-traveling for students who have to go to the same location (event). This will make going to events easier.

### project scope

The project scope aims for the fact that we want to make it easier for the students (this is only for howest students) to travel to several events. we want to make a database to safe to store data. This allows students to see who goes to an event how and that it is possible to travel together in this way.

### Not in scope

The solution is made for students in particular. This may be not ideal to use in company’s or in high school. Because the largest group of students does not have a car yet and the school rules everything for them. Teachers are also able to use the system.

### project success

* The program is easy and clear to use, the use of the program can’t be timerobbing.
* The project shows who travels how to the event and if there are possibilities for co-traveling.
* This database publishes rapports to participants can see how wa can travel to the event.

### project deliverables

On regular base this Specification ‘lastenboek” will be updated on Github:

* Progress reports
* Issue reports
* Meeting notes

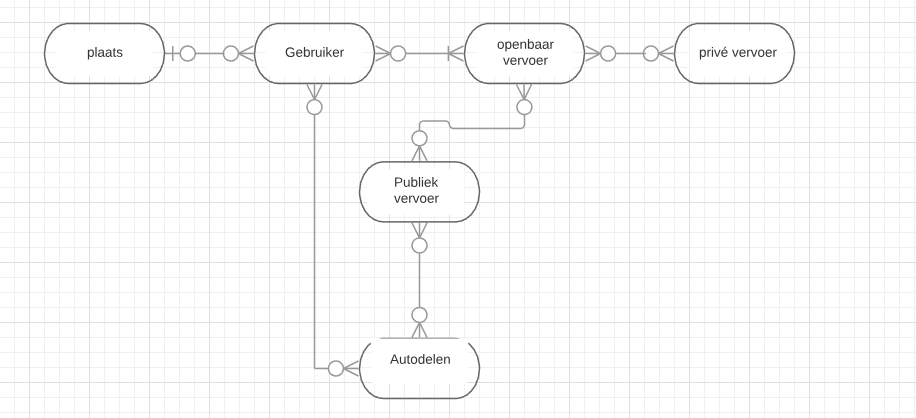
## Real World

Shared mobility is a program that ensures that students can see who is going to a particular event in which way. In a simple database the most important data of the students and means of transport are kept. With this data we can make it possible for students to see who is going which way and if there is a possibility to go there together. How will this work?

* At each event the student has to input how he will be going there. This is stored in a database and shown to the other students.
* If there are several students travelling to the event by the same means of transport, it is suggested with whom they can travel together.
* Everything is visualized who goes to the event how and the students can see with whom they can travel together.
* An extension to this may be that the students can enter how much everything will cost to get there.

## Entity Relationship model (ER)

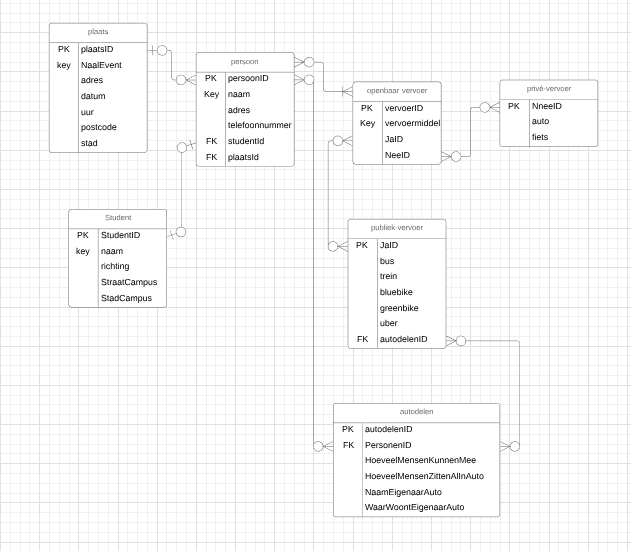
### the model



### Description of the ER

* Een plaats kan 0 personen hebben, maar ook meerdere personen.
* Een persoon kan gelinkt worden aan 0 plaatsen, ofwel aan 1.
* Een persoon kan gelinkt worden aan minstens 1 openbaar vervoer ofwel kan je ook meerdere vervoeren.
* En op een openbaar vervoer kan je ofwel 0 personen hebben, maar ook meerdere.
* Een openbaar vervoer kan gelinkt worden aan 0 of meerdere privé-vervoeren, maar een privévervoer kan gelinkt worden aan 0 of meerdere openbare vervoeren.
* Een openbaar vervoer kan gelinkt worden aan 0 of meerdere publiek vervoer, en publiek vervoer kan gelinkt worden aan 0 of meerdere openbare vervoeren.
* Een publiek vervoer kan gelinkt worden aan 0 of meerdere autodelen, en autdelen kan gelinkt worden aan 0 of meerdere publieke vervoeren.
* Autodelen kan gelinkt worden aan 0 of meerdere personen, en personen kan gelinkt worden aan 0 of meerdere autodelen.

## Entitty Relationship diagram (ERD)



# Technical documentation

## core components vs core logic, low level analysis

Core components, marked in red, are the technical blocks of code that perform a certain action. Typically, this is the input/output.  
Core logic, marked in yellow, is the logical process, the algorithm, that is executed on your input. Typically, this is the processing.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Input | Program | Output | Error handling |
| Input | Always: how the student is going to the event  1e time:  data student |  |  |  |
| Processing |  | Saves how the student is going to the event  Saves the data of the users in a database |  |  |
| Output |  |  | Shows how the students are going to the event | There will be a multiple choice, so an error isn’t possible unless there is no internet |

## DUMMY DATA

This section contains dummy data so that the development of the project can start without having the need of the working data-retrieval component.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tijdstempel | #1 | #2 | #3 |  |
| 27/02/2019 | possibile ways of traveling | Event adress | Data from the person who wants to the event |  |

## CORE LOGIC

The user must say how he is going to the event place. This info will be send to a database. (the first time a person uses this program the user has to give his data address, name etc..) (for each event, the address and other data must be adjusted). users will be able to see who is going to the event how and if there are possibilities to travel together

### VALIDATE DATA

The user has to select how he will be going to the event. Without the data the program can’t run and the user can’t see who is going how to the event.

### PROCESS DATA

Input = the way of travelling

The program sends the data to a database and tells the user who is going how to the event and if there are possibility’s for co-traveling.

### GENERATE OUTPUT

Generate an array in following structure so that it can be used in later output:

The output in this program will show the user who is going how to the event and is there are possibility’s to travel together.

|  |  |  |
| --- | --- | --- |
| Choice | Student | Student |
| Shared Mobility | Jorden | Amber |