Bürgerbus mConcAppt Interaction Concept Documentation

**Authors:**

**Main Editors**

Server: Ricarda Rosemann

BürgerApp: Marcel Müller

BusDriveApp: Sascha Müller

**Project Managers**

Server: Muhammad Baniasad

BürgerApp: Dominik Skalník

BusDriveApp: Charel Irrthum

# Table of Contents

[Table of Contents 1](#_Toc456715291)

[List of Tables 2](#_Toc456715292)

[1 Introduction 5](#_Toc456715293)

[2 Usage Context 5](#_Toc456715294)

[3 Stakeholder Description 6](#_Toc456715295)

[3.1 Stakeholders and Goals 6](#_Toc456715296)

[3.2 User Personas 7](#_Toc456715297)

[4 As-is Situation 9](#_Toc456715298)

[4.1 As-is Situation Scenarios 9](#_Toc456715299)

[4.2 Problems in As-is Situation 10](#_Toc456715300)

[5 To-be Situation 11](#_Toc456715301)

[5.1 To-be Situation Scenarios 11](#_Toc456715302)

[5.2 Example Scenario for the To-Be-Situation 13](#_Toc456715303)

[6 Non-Functional Requirements 14](#_Toc456715304)

[7 Solution 14](#_Toc456715305)

[7.1 Assumptions 14](#_Toc456715306)

[7.2 Key Solution Concepts 14](#_Toc456715307)

[7.3 Traceability between System Functions 15](#_Toc456715308)

[7.4 System Functions for the BusDriveApp 16](#_Toc456715309)

[7.5 System Functions for the BürgerApp 26](#_Toc456715310)

[7.6 System Functions for the server 32](#_Toc456715311)

[8 App Functionality 42](#_Toc456715312)

[8.1 Interaction Cases BürgerApp 42](#_Toc456715313)

[8.2 Interaction Cases BusDriveApp 45](#_Toc456715314)

[8.3 Screenflow BürgerApp 52](#_Toc456715315)

[8.4 BusDriveApp Screens (Screenflow) 53](#_Toc456715316)

[8.5 BuergerApp Screens 57](#_Toc456715317)

[9 Additional implemented elements of BusDriveApp at current state 64](#_Toc456715318)

[10 Glossary 65](#_Toc456715319)

# List of Tables

[Table Stakeholders and Goals 6](#_Toc456715320)

[Table #P01: Persona Bus Driver 7](#_Toc456715321)

[Table #P02: Persona Citizen 8](#_Toc456715322)

[Table #AIS\_01: Picking up at bus stations 9](#_Toc456715323)

[Table #AIS\_02: Picking up at home 9](#_Toc456715324)

[Table #TBS\_01: Picking up at bus stations 11](#_Toc456715325)

[Table #TBS\_02: Picking up Citizens at their homes 11](#_Toc456715326)

[Table #SF\_Traceability 15](#_Toc456715327)

[Table #SF\_B: Overview 16](#_Toc456715328)

[Table #SF\_B0.1: Request bus list 16](#_Toc456715329)

[Table #SF\_B0.2: Request line list 17](#_Toc456715330)

[Table #SF\_B0.3: Request route and stops 18](#_Toc456715331)

[Table #SF\_B1: Select Bus 18](#_Toc456715332)

[Table #SF\_B2: Select Busline 19](#_Toc456715333)

[Table #SF\_B3.1: Send Bus Status Data to server 19](#_Toc456715334)

[Table #SF\_B3.2: Send Real Time Data to server 20](#_Toc456715335)

[Table #SF\_B3.3: Send Custom Stop status update to server 21](#_Toc456715336)

[Table #SF\_B4: Change number of taken seats 21](#_Toc456715337)

[Table #SF\_B5: Receive Custom Stop Request 22](#_Toc456715338)

[Table #SF\_B6: Show map of current position 23](#_Toc456715339)

[Table #SF\_B7: Show Line Stops 23](#_Toc456715340)

[Table #SF\_B8: Cycle through Line Stops 24](#_Toc456715341)

[Table #SF\_B9: Respond to Custom Stop Request 24](#_Toc456715342)

[Table #SF\_B10: Complete Custom Stop 25](#_Toc456715343)

[Table #SF\_C: Overview 26](#_Toc456715344)

[Table #SF\_C1: get current bus position 26](#_Toc456715345)

[Table #SF\_C2: Get estimated arrival time for stop / current position 27](#_Toc456715346)

[Table #SF\_C3: Get information about the bus (line/color/bus photo) 27](#_Toc456715347)

[Table #SF\_C4: Request a stop at a stop /iteration 2 28](#_Toc456715348)

[Table #SF\_C5: Notify the user about changes in the schedule /iteration 2 28](#_Toc456715349)

[Table #SF\_C6: Show number of free seats 29](#_Toc456715350)

[Table #SF\_C7: Request help for shopping /iteration x 30](#_Toc456715351)

[Table #SF\_C8: Schedule wayback /iteration x 30](#_Toc456715352)

[Table #SF\_C9: Show schedule for current stop 31](#_Toc456715353)

[Table #SF\_C10: Show List of Stops 31](#_Toc456715354)

[Table #SF\_C11: Request a stop at a given location /iteration 2 32](#_Toc456715355)

[Table #SF\_S1: Send list of lines 32](#_Toc456715356)

[Table #SF\_S2: Send list of busses 33](#_Toc456715357)

[Table #SF\_S3: Send List of stops 34](#_Toc456715358)

[Table #SF\_S4: Send List of routes 34](#_Toc456715359)

[Table #SF\_S6: Send GPS-Data of Bus 35](#_Toc456715360)

[Table #SF\_S7: Store bus and line 36](#_Toc456715361)

[Table #SF\_S8: Store GPS data 36](#_Toc456715362)

[Table #SF\_S9: Send latest timestamp 37](#_Toc456715363)

[Table #SF\_S10: Notification of custom stop request 38](#_Toc456715364)

[Table #SF\_S11: Response to custom stop request 39](#_Toc456715365)

[Table #SF\_S12: Transferring received help request 39](#_Toc456715366)

[Table #SF\_S13: Send help request response 39](#_Toc456715367)

[Table #SF\_S14: Send number of available seats 40](#_Toc456715368)

[Table #SF\_S15: Store number of available seats 40](#_Toc456715369)

[Table #SF\_S16: Send status of Custom Stop request 41](#_Toc456715370)

[Table IC\_C1 42](#_Toc456715371)

[Table IC\_C2 43](#_Toc456715372)

[Table IC\_C3 44](#_Toc456715373)

[Table IC\_C4 44](#_Toc456715374)

[Table #IC\_B1: Setting up for driving 45](#_Toc456715375)

[Table #IC\_B2: Document number of Citizens 46](#_Toc456715376)

[Table #IC\_B3: See route 47](#_Toc456715377)

[Table #IC\_B4: See stops 48](#_Toc456715378)

[Table #IC\_B5: Receiving a Custom Stop Request 49](#_Toc456715379)

[Table #IC\_B6: Completing a Custom Stop 51](#_Toc456715380)

[Table Additional implemented elements 64](#_Toc456715381)

[Table Glossary 65](#_Toc456715382)

# Introduction

This document includes the results from the requirements workshop with the customer and the following updates on requirements about the development of a mobile system for the local Bürgerbus on the 22.04.2016. The Bürgerbus system consists of 2 mobile applications, the BusDriveApp for the Bus Driver and the BürgerApp for the citizens. It is intended, that the system will be used within the project Digitale Dörfer (for further information see [www.digitale-doerfer.de](http://www.digitale-doerfer.de)) in cooperation with the Fraunhofer IESE.

The chapters are structured as follows. The chapters 2 to 4 are covering the before stage, which means they describe the usage context, the people who will be influenced by the project and the current as-is situations with scenarios for a better understanding of how the Bürgerbus works today. And then in chapter 5 possible corresponding to-be situations are explained and a detailed example for a future scenario is described. From Chapter 6 to 8 this document explains how the finished product should function seen in the detailed system functions (chapter 7), interaction cases and screens of the finished apps (chapter 8). In the very end you find a glossary with some used vocabulary.

# Usage Context

Weilerbach is a small city near Kaiserslautern. It has a very good infrastructure, but the distance between points of interest for the normal citizens is normally very far. Let us imagine Emma Meier (see in persona section 3.2). She has been living in Weilerbach all her life. She is living at the beginning of Weilerbach and wants to go shopping at the local supermarket. Mrs. Meier cannot walk properly and has no car instead she uses the Bürgerbus to get to the supermarket, but has problems with the time schedule of the bus and sometimes she needs to get picked up at her house. The Bürgerbus system wants to ease her life and make her a more satisfied stakeholder.

# Stakeholder Description

## 3.1 Stakeholders and Goals

### Table Stakeholders and Goals

|  |  |  |
| --- | --- | --- |
| Stakeholder Name | Stakeholder Role | Stakeholder Main Goals |
| Bus Driver | The Bus Driver drives a given bus route and stops at bus stations where passengers are waiting. | His main goal is to get people to their destination and be on time according to the schedule. |
| Citizen | Main User | Mobility, possibility to transport the goods, socialization (meet other people).  /\*See where the bus actually is to get on.\*/ |
| Developer | Developing the BürgerApp, the BusDriveApp or the backend | Developing the system  good time effort  high quality results. |
| Customer | Wants the System to be developed | High quality system  all use cases covered  reliability. |
| Administrational Bus App Community Members | Evaluates the system  Distributes the app | Continuous improvement of the system and service, paths optimization.  /\*See statistics.\*/ |

**Author:** Marcel Müller, Erik Gruener

**Status:** Complete

**Reviewer:** Dominik Skalnik

**Review status:** Complete

## 3.2 User Personas

### Table #P01: Persona Bus Driver

|  |
| --- |
| **Helmut Schmidt**  tag-der-aelteren-generation.jpg   * Age: 55 * Role: Bus Driver for the community (drives 4-5 times a month) * Family background: married, 2 children   Main characteristics   * He has lived in Weilerbach all his life. * He works from home for an IT Company and knows how to use a smartphone. * He is also involved in other community project.   Main Goals   * to drive the route through Weilerbach and pick up people. * be on time so the passengers do not have to wait for long time. * Have an easy system that does not distract him too much.   Typical Challenges   * There is a schedule but as of now it does not work perfectly. * If he has to pick up a person from their home his schedule will change.   Prospective concrete usage context:   * The system should be running during the time the bus is driving around. * The Bus Driver can interact with the device during breaks or at stops. |

**Author:** Erik Gruener

**Status:** Complete

**Reviewer:** Sascha Müller

**Review status:** Complete

### Table #P02: Persona Citizen

|  |
| --- |
| **Emma Meier**  herzlichen-glueckwunsch-zum-100-geburtstag-liebe-frau-coelln-3840ee8f-2e4c-4139-954f-d96621eacbd8.jpg   * Age: 70 * Role: Citizen (uses the Bürgerbus 1-2 times a week) * Family background: married, children, grandchildren   Main characteristics   * She has lived in Weilerbach all her life. * Manages the household, is in charge of the groceries purchases. * Owns a (low end) smartphone that confuses her most of the time. * Is involved in other senior oriented communities.   Main Goals   * Uses the Bürgerbus for trips to and from the grocery store and the community center * Does not want to wait too long on the street for a ride. * Sometimes needs to get picked up from her house. * Have a well readable, simple system that provides her the information   Typical Challenges   * There is a schedule but as of now it does not work perfectly. * She may needs assistance with the purchases (From the store to the bus and from the bus to my home) which changes the schedule of the Bürgerbus   Prospective concrete usage context:   * The citizen app will be used inside and outside of buildings mostly with mobile low bandwidth data |

**Author:** Steffen Holzer

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:**Complete

# As-is Situation

## 4.1 As-is Situation Scenarios

### Table #AIS\_01: Picking up at bus stations

|  |  |
| --- | --- |
| Item | Description |
| Context | The Bus Driver is driving around town and picking up people at the bus stations. |
| Precondition | There are free seats in the bus. There is a person at the bus stop. If no person is at the bus station the bus will continue driving. |
| Step 1 | The Bus Driver gets close to a bus stop on the route. |
| Step 2 | A Citizen signals that he wants to get picked up. |
| Step 3 | The Bus Driver pulls over and stops. |
| Step 4 | The Citizen enters bus. |
| Step 5 | The Bus Driver continues route. |
| Postcondition | There is one less seat available on the bus. |

**Author:** Dominik Skalnik, Erik Gruener

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table #AIS\_02: Picking up at home

|  |  |
| --- | --- |
| Item | Description |
| Context | The Office in Weilerbach receives a call from a Citizen that wants to get picked up at home due to health-problems etc. |
| Precondition | There are free seats in the bus. The Citizen is at home and ready at a specific time. |
| Step 1 | The Bus Driver gets informed by phone call to pick up a new person at a certain place. |
| Step 2 | The Bus Driver takes a detour and arrives at the Citizen's door |
| Step 3 | The Bus Driver honks to tell the person that he is there. |
| Step 4 | The Citizen comes out the door and enters the bus OR decides not to join the bus. |
| Step 5 | The Bus Driver continues to his old route. |
| Postcondition | The Citizen is picked up from home and bus is on his usual route.There is one less seat available on the bus. |

**Author:** Dominik Skalnik, Erik Gruener

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

## 4.2 Problems in As-is Situation

* Sometimes it happens that a bus is full with Citizens and no more can fit in the bus. The Citizens waited only to see that they cannot take this bus but have to wait for the next one. When this scenario occurs it also possible for the Bus Driver to drive the Citizens to their destination, if they all want to go to the same place, and then return to pick up the waiting Citizens to minimize their waiting time.
  + How it is addressed: The Bus Driver documents the number of Citizens riding the bus (#IC\_B2: Document number of Citizens) and then the Citizens are able to see how much seats are free in the bus they are waiting for (#IC\_C2 next iteration). So they can see if it may not be worth to wait for the bus.
* A Problem with a Citizen calling to get picked up is that the office has to call the Bus Driver that day or the next day to tell him where to go at what time which disturbs the schedule. It is also possible that the Citizen is not at home and the Bus Driver’s time would get wasted. Also for now there is no confirmation that the Bus Driver is informed and the Citizens will be picked up.
  + Solution: With the new Custom Stop Request (#IC\_C4) the requests of the Citizens can immediately be propagated to the Bus Drivers and they can directly answer if they will include the stop on their route (#IC\_B5: Receiving a Custom Stop Request) and the Citizens will be notified
* Of course it is possible that due to obstacles or detours the set schedule for the bus to arrive at a station cannot be met. In this Situation the Citizen does not know when the bus is coming or if it has already past.
  + Solution: With the BusDriveApp sending the GPS position the Citizens can check the status of the bus and its estimated arrival time (#IC\_C2)

**Author:** Erik Gruener, Sascha Müller

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

# To-be Situation

## 5.1 To-be Situation Scenarios

### Table #TBS\_01: Picking up at bus stations

|  |  |
| --- | --- |
| Item | Description |
| Context | The bus is on its route around the town and sends information about its current status. |
| Precondition | The Bus Driver’s phone and the BusDriveApp are running. The phone is connected to the internet. Citizens are waiting at the bus stations. |
| Step 1 | The Bus Driver drives on his route from stop to stop. |
| Step 2 | The BusDriveApp sends the GPS location to the server. |
| Step 3 | The Citizens at the bus stations can see the status of the bus with the BürgerApp (#IC\_C2, #IC\_C3) |
| Step 4 | The bus stops, Citizens leave and enter the bus, the Bus Driver sets the number of available seats on the BusDriveApp (#IC\_B2: Document number of Citizens) |
| Step 5 | Bus continues its route. |
| Postcondition | Citizens are satisfied because they know the current location of the bus and when it will arrive. |

**Author:** Erik Gruener, Sascha Müller

**Status:** Complete

**Reviewer:** Hafiz Ahsan Raza

**Review status:** Complete

### Table #TBS\_02: Picking up Citizens at their homes

|  |  |
| --- | --- |
| Item | Description |
| Context | The bus drives around town and on request it takes a detour to pick a person up at their house. The arrival time at the next stops will change. |
| Precondition | The BusDriveApp is running and is connected to the internet. Citizens are waiting at bus stations and custom stops (their homes). |
| Step 1 | The Bus Driver receives information about a Custom Stop (#IC\_B5: Receiving a Custom Stop Request) |
| Step 2 | The Bus Driver decides that he will include the stop on his tour (#IC\_B5: Receiving a Custom Stop Request) |
| Step 3 | The bus drives to the requested stop to pick up the Citizen |
| Step 4 | The Bus Driver marks the stop request as completed (#IC\_B6: Completing a Custom Stop) |
| Step 5 | Bus continues driving its regular route. |
| Step 6 | The App sends information about current status and the Citizens waiting for bus can see that the arrival time has changed (#IC\_C2) |
| Postcondition | Citizens are satisfied because: they can be picked up at their house and other Citizens know where the bus currently is and when it will arrive. |
| Refined in system functions | #SF\_B5, #SF\_B9, #SF\_B10, #SF\_S6, #SF\_S7; #SF\_S8, SF\_S9 (later iteration) |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Hafiz Ahsan Raza, Erik Gruener

**Review status:** Complete

## 5.2 Example Scenario for the To-Be-Situation

It’s Monday morning and Emma Meier (#P02: Persona Citizen), a 70-year-old Citizen of Weilerbach, who has lived there all her live with her husband and raised her children there, wants to go to the supermarket today to buy the groceries for the week.

In another part of the city Helmut Schmidt (#P01: Persona Bus Driver), a 55-year-old IT worker is on his way to the Bürgerbus Central in Weilerbach to start his community service as a Bus Driver today. He does this about four times a month to contribute to his community Weilerbach, where he also has been living all his live.

So when he reaches the central he shortly visits the office to inform the people working there that he is here to start his tour. He is handed the keys to one of the busses in the nearby parking lot and after that Helmut is ready to start.

He enters the bus and turns on the stationary smartphone, which is attached to the windshield and connected to the cigarette lighter. He taps the BusDriveApp on the Homescreen of the device and after start up the Start Screen (#SC\_B0) of the BusDriveApp is shown. Because Helmut doesn’t need to change any major settings he taps the ‘START TOUR’ button to set up the app for driving (#IC\_B1: Setting up for driving). The BusDriveApp is ready, Helmut can start driving now, but before that he wants to get a quick overview of all the stops on his route (#IC\_B4: See stops) and how it looks on the map (#IC\_B3: See route). With that being done, he starts the motor and drives towards the first stop, where an older man is waiting. The man enters the bus – he needs to get to the next stop and can’t walk that fast so he decided to wait for the bus – and Helmut can increase the displayed number of Citizens riding the bus (#IC\_B2: Document number of Citizens).

Back to Emma Meier: Of course she will be also using the Bürgerbus to get to the supermarket today. She used to drive on her own but now she doesn’t feel up to the task due to her age. So she takes out her smartphone, which her oldest daughter gave her last year, and starts the BürgerApp. In the Citizen Application Emma can see all available bus stops on a map and she can see when the bus arrives where (#IC\_C1). Unfortunately the bus route is very far away from her position. Nevertheless she has the possibility to request a custom stop for the bus next tour right at her position (#IC\_C4). She decides to request a stop right where she is.

A Notification on Helmut Schmidt’s Bus Driver phone appears, which displays that new Custom Stop Requests are pending (#IC\_B5: Receiving a Custom Stop Request). He is standing at a traffic light so taps the notification and sees that Emma Meier wants to be picked up at her home at 11am. It’s close to his tour and he can pick her up later so he decides to accept the request. Again he checks the map to have an overview of where the Custom Stop exactly is (#IC\_B3: See route).

It’s 11am and Helmut arrives in front of Emma’s home, she looks out of the window and recognizes the bus, so she leaves her house and enters the bus. The Custom Stop is reached and the Citizen entered the bus, so Helmut denotes the stop as ‘completed’ on his list (#IC\_B6: Completing a Custom Stop) and the tour goes on.

After a rather long way the supermarket is reached and Emma Meier leaves the bus. She is at her destination and she is happy that everything worked fine and that she is able to start shopping now. Helmut Schmidt continues his tour and after her purchase Emma Meier will be using the Bürgerbus again.

Everyone in Weilerbach is satisfied that it is easy to participate in community services and that the Bürgerbus is a way of transportation you can count on.

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

# Non-Functional Requirements

Easy to use: The apps should be easy to use. No instruction or help manual should be needed to use the functionality of the applications.

# Solution

## 7.1 Assumptions

The device running the BusDriveApp is stationary in all busses.

## 7.2 Key Solution Concepts

The Bus Driver uses BusDriveApp to communicate with server about current tour status and updates. Citizens use BürgerApp to check on the busses and request Custom Stops.

**Author:** Sascha Müller

**Status:** Complete (BusDriveApp)

**Reviewer:** Charel Irrthum

**Review status:** complete

## 7.3 Traceability between System Functions

### Table #SF\_Traceability

|  |  |  |
| --- | --- | --- |
| Bus Driver App | Server | Citizen App |
| #SF\_B0.2: Request line list | #SF\_S1: Send List of Lines | #SF\_C9 : Show schedule for current stop (partially) |
| #SF\_B0.1: Request bus list | #SF\_S2: Send List of Busses | #SF\_C3: Get information about the Bus  #SF\_C9 |
| #SF\_B0.3: Request route and stops | #SF\_S3: Send list of Stops | #SF\_C10: Show list of Stops |
| #SF\_S4: Send List of Routes |
| - | #SF\_S6: Send GPS-Data of Bus | #SF\_C1: Get current Bus position |
| #SF\_B3.1: Send Bus Status Data to server | #SF\_S7: Store bus and line | - |
| #SF\_B3.2: Send Real Time Data to Server | #SF\_S8: Store GPS-Data | - |
|  | #SF\_S9: Send latest timestamps | Implicit in most system functions |
| #SF\_B5: Receive Custom Stop Request | #SF\_S10: Notification of custom stop request | #SF\_C11: Request a stop at a given location |
| #SF\_B3.3: Send Custom Stop status update to server | #SF\_S11: Response to custom stop request |
| - | #SF\_S14: Send number of available seats | #SF\_C6: Show number of free seats |
| #SF\_B3.2: Send Real Time Data to server | #SF\_S15: Store number of available seats | - |
| - | #SF\_S16: Send status of custom request | #SF\_C11 |

## 7.4 System Functions for the BusDriveApp

### Table #SF\_B: Overview

|  |  |
| --- | --- |
| Identifier | Name |
| #SF\_B0.1 | Request bus list |
| #SF\_B0.2 | Request line list |
| #SF\_B0.3 | Request route and stops |
| #SF\_B1 | Select Bus |
| #SF\_B2 | Select Busline |
| #SF\_B3.1 | Send Bus Status Data to server |
| #SF\_B3.2 | Send Real Time Data to server |
| #SF\_B3.3 | Send Custom Stop status update to server |
| #SF\_B4 | Change number of taken seats |
| #SF\_B5 | Receive Custom Stop Request |
| #SF\_B6 | Show map of current position |
| #SF\_B7 | Show Line Stops |
| #SF\_B8 | Cycle through Line Stops |
| #SF\_B9 | Respond to Custom Stop Request |
| #SF\_B10 | Complete Custom Stop |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Oliver Säger

**Review status:** Complete

### Table #SF\_B0.1: Request bus list

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B0.1 |
| Name | Request bus list |
| Input Data | - |
| Precondition | The Bus Driver starts the BusDriveApp. |
| Description | The current list of all busses is requested from the server (#SF\_S2: Send list of busses) |
| Exception | No connection to server; |
| Business rules | - |
| Quality Management | - |
| Output Data | request |
| Postcondition(s) | After receipt of data (#SF\_S2: Send list of busses) the Bus Driver is able to select the bus he will drive (#SF\_B1: Select bus) |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Oliver Säger

**Review status:** Complete

### Table #SF\_B0.2: Request line list

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B0.2 |
| Name | Request line list |
| Input Data | - |
| Precondition | The Bus Driver selects bus |
| Description | The current list of all lines is requested from the server (#SF\_S1: Send list of lines) |
| Exception | No connection to server; |
| Business rules | - |
| Quality Management | - |
| Output Data | request |
| Postcondition(s) | After receipt of data (#SF\_S1: Send list of lines) the Bus Driver is able to select line (#SF\_B2: Select bus line) |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Oliver Säger

**Review status:** Complete

### Table #SF\_B0.3: Request route and stops

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B0.3 |
| Name | Request route and stops |
| Input Data | - |
| Precondition | The Bus Driver selects line |
| Description | The according route and stops are requested from the server (#SF\_S3: Send list of stops, #SF\_S4: Send list of routes) |
| Exception | No connection to server; |
| Business rules | - |
| Quality Management | - |
| Output Data | request |
| Postcondition(s) | After receipt of data (#SF\_S3: Send list of stops, #SF\_S4: Send list of routes) the Bus Driver is able to see his route and stops (#SF\_B6: Show map of current position, #SF\_B7: Show Line Stops) |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Oliver Säger

**Review status:** Complete

### Table #SF\_B1: Select Bus

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B1 |
| Name | Select bus |
| Input Data | List of busses |
| Precondition | Successful receipt of data (#SF\_S2: Send list of busses) |
| Description | The Bus Driver selects the bus he will be driving that day. |
| Exception | - |
| Business rules | - |
| Quality Management | - |
| Output Data | Selected bus |
| Postcondition(s) | Bus was selected by the Bus Driver, information will be sent to the server (#SF\_B3.1: Send Bus Status Data to server and #SF\_B3.2 Send Bus Status Data to server) |

**Author:** Erik Grüner, Sascha Müller

**Status:** Complete

**Reviewer:** Charel Irrthum

**Review status:** Complete

### Table #SF\_B2: Select Busline

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B2 |
| Name | Select Busline |
| Input Data | List of lines |
| Precondition | Bus was selected (#SF\_B1: Select bus), successful receipt of data (#SF\_B2: Send list of busses) |
| Description | The Bus Driver selects the line that he will be driving that day. |
| Exception | - |
| Business rules | - |
| Quality Management | - |
| Output Data | Selected line |
| Postcondition(s) | Line was selected, Information will be sent to the server (#SF\_B3.1: Send Bus Status Data to server and #SF\_B3.2 Send Bus Status Data to server) |

**Author:** Erik Grüner, Sascha Müller

**Status:** Complete

**Reviewer:** Charel Irrthum

**Review status:** complete

### Table #SF\_B3.1: Send Bus Status Data to server

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B3.1 |
| Name | Send Bus Status Data to server |
| Input Data | Bus and Line Information |
| Precondition | connection to the server, Bus and Line selected |
| Description | After the Bus Driver has selected bus and line the app sends this information (and that he started driving, later iteration) to the server |
| Exception | no server connection |
| Business rules | - |
| Quality Management | - |
| Output Data | bus and line |
| Postcondition(s) | data is sent to server and stored (#SF\_S7: Store bus and line), the tour starts |

**Author:** Sascha Müller,

**Status:** Complete

**Reviewer:** Charel Irrthum

**Review status:** complete

### Table #SF\_B3.2: Send Real Time Data to server

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B3.2 |
| Name | Send Real Time Data to server |
| Input Data | Bus, GPS coordinates, current time |
| Precondition | connection to the server, GPS connection, the tour has started |
| Description | While driving the app sends bus, current GPS coordinates and a timestamp to the server (later iteration: also the number of seats taken) |
| Exception | no server connection, no GPS signal |
| Business rules | Data is sent every 60 seconds or if the GPS position has changed by at least 75 metres |
| Quality Management | - |
| Output Data | Bus, GPS coordinates, timestamp |
| Postcondition(s) | data is sent to server and stored (#SF\_S8: Store GPS data) |

**Author:** Sascha Müller,

**Status:** Complete

**Reviewer:** Charel Irrthum

**Review status:** complete

### Table #SF\_B3.3: Send Custom Stop status update to server

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B3.3 |
| Name | Send Custom Stop status update to server |
| Input Data | New custom stop status (accepted, declined, completed) |
| Precondition | connection to the server, the tour has started, receipt of Custom Stop Requests |
| Description | When the Bus Driver accepts or declines a Custom Stop Request or when a custom stop request has been completed, the app sends a status update of this Custom Stop Request to the server |
| Exception | no server connection |
| Business rules | - |
| Quality Management | - |
| Output Data | Custom Stop ID, status, timestamp |
| Postcondition(s) | data is sent to server (#SF\_S11: Response to custom stop request) |

**Author:** Sascha Müller,

**Status:** Complete

**Reviewer:** Patrick Pschorn

**Review status:** Complete

### Table #SF\_B4: Change number of taken seats

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B4 |
| Name | Change number of taken seats |
| Input Data | - |
| Precondition | The tour has started |
| Description | A Citizen enters (leaves) the bus and the Bus Driver increases (decreases) the displayed number of passengers on the bus. |
| Exception | - |
| Business rules | - |
| Quality Management | The numbers go from 0 to max. number of seats according to used bus. |
| Output Data | The number of seats available |
| Postcondition(s) | The number of seats has changed to current situation, (later iteration: information will be sent to server in #SF\_B3.2,) driver continues tour. |

**Author:** Sascha Müller, Erik Grüner

**Status:** Complete

**Reviewer:** Patrick Pschorn

**Review status:** Complete

### Table #SF\_B5: Receive Custom Stop Request

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B5 |
| Name | Receive Custom Stop Request |
| Input Data | Data of the location where the Citizen wants to be picked up plus additional information (#SF\_S10: Notification of custom stop request) |
| Precondition | connection to the server, Bus Driver selected the line (#SF\_B2: Select bus line) |
| Description | The BusDriveApp receives a request to pick up a Citizen at a stop that is not on the stops list and shows a notification to the Bus Driver. |
| Exception | no connection to server |
| Business rules | The notification will disappear after a certain time |
| Quality Management | Only the requests relevant to the Bus Driver are displayed |
| Output Data | - |
| Postcondition(s) | The Bus Driver will be able to accept or decline the request (#SF\_B9: Respond to Custom Stop Request) |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Patrick Pschorn

**Review status:** Complete

### Table #SF\_B6: Show map of current position

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B6 |
| Name | Show map of current position |
| Input Data | Current Position, Custom Stops, Line Stops |
| Precondition | GPS signal, internet connection |
| Description | The Bus Driver can see his position on a map, markers for the Custom Stops and a line for his route including markers for the Line Stops. |
| Exception | Lost internet connection and/or GPS signal |
| Business rules | - |
| Quality Management | The position should be realtime |
| Output Data | - |
| Postcondition(s) | - |

**Author:** Erik Grüner, Sascha Müller

**Status:** Complete

**Reviewer:** Charel Irrthum, Oliver Säger

**Review status:** Complete

### Table #SF\_B7: Show Line Stops

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B7 |
| Name | Show Line Stops |
| Input Data | List of Line Stops provided by the server |
| Precondition | Successful receipt of data (#SF\_0.3: Request route and stops) |
| Description | A list of all stops of the chosen line is shown |
| Exception | - |
| Business rules | The arrival time for each Line Stop is shown according to schedule (possibly, in a later iteration it will be updated when the bus is delayed) |
| Quality Management | stops ordered |
| Output Data | - |
| Postcondition(s) | - |

**Author:** Erik Grüner, Patrick Pschorn, Sascha Müller

**Status:** Complete

**Reviewer:** Charel Irrthum, Patrick Pschorn

**Review status:** Complete

### Table #SF\_B8: Cycle through Line Stops

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B8 |
| Name | Cycle through Line Stops |
| Input Data | - |
| Precondition | The tour has started |
| Description | When passing a stop the Bus Driver presses a button so the app shows the next Line Stop or another button to go back to the previous Line Stop. |
| Exception | - |
| Business rules |  |
| Quality Management |  |
| Output Data | next (previous) Line Stop |
| Postcondition(s) | The next (previous) Line Stop is shown on the Drive Screen |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Patrick Pschorn

**Review status:** Complete

### Table #SF\_B9: Respond to Custom Stop Request

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B9 |
| Name | Respond to Custom Stop Request |
| Input Data | - |
| Precondition | Receipt of a Custom Stop Request (#SF\_B5: Receive Custom Stop Request) |
| Description | The Bus Driver chooses to accept or decline Custom Stop Requests. If he accepts, the Custom Stop will be shown on the #SC\_B3.1: Drive Screen - Main Tab. If he declines it will disappear. |
| Exception | - |
| Business rules | The Custom Stops are ordered by requested pickup time |
| Quality Management | - |
| Output Data | Custom Stop marked as ‘accepted’ or ‘declined’ |
| Postcondition(s) | The response will be sent to server (#SF\_B3.3: Send Custom Stop status update to server) |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Patrick Pschorn

**Review status:** Complete

### Table #SF\_B10: Complete Custom Stop

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_B10 |
| Name | Complete Custom Stop |
| Input Data | - |
| Precondition | Tour has started and contains Custom Stops |
| Description | When a Custom Stop has been reached the Bus Driver selects it and denotes it as ‘completed’ and it will disappear from the list. The Custom Stop can also be marked as ‘not shown up’ if the Citizen is not there. |
| Exception | - |
| Business rules | The Custom Stops are ordered by requested pickup time |
| Quality Management | - |
| Output Data | Custom Stop marked as ‘completed’ or ‘not shown up’ |
| Postcondition(s) | The updated status of the Custom Stop will be sent to server in (#SF\_B3.3: Send Custom Stop status update to server) |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Patrick Pschorn, Oliver Säger

**Review status:** Complete

## 7.5 System Functions for the BürgerApp

### Table #SF\_C: Overview

|  |  |
| --- | --- |
| Identifier | Name |
| #SF\_C1 | Get current bus position |
| #SF\_C2 | Get estimated arrival time |
| #SF\_C3 | Get information about the bus |
| #SF\_C4 | Request a stop |
| #SF\_C5 | Notify the user about changes in the schedule |
| #SF\_C6 | Show number of free seats |
| #SF\_C7 | Request help for shopping |
| #SF\_C8 | Schedule way back |
| #SF\_C9 | Show schedule for current stop |
| #SF\_C10 | Show List of Stops |
| #SF\_C11 | Request a stop at a given location |

**Author:** Steffen Holzer

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** complete

### Table #SF\_C1: get current bus position

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C1 |
| Name | get current bus position |
| Input Data | - |
| Precondition | the bus is driving on its route |
| Description | shows the citizen where the bus actually is on a map |
| Exception | no connection to the internet |
| Business rules | the position is transmitted once it is requested |
| Quality Management | the location should be accurate (delta 20 m) |
| Output Data | the current location of the bus |
| Postcondition(s) | the location is transmitted to the citizen |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table #SF\_C2: Get estimated arrival time for stop / current position

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C2 |
| Name | Get estimated arrival time for stop / current position |
| Input Data | Bus Position, bus station |
| Precondition | #SF\_C1 is successful |
| Description | Estimates the time of arrival of the bus at the selected bus station |
| Exception | - |
| Business rules |  |
| Quality Management | Time to arrival in h:mm,  mean estimation error <= 20% |
| Output Data | Time of arrival at bus station, time to arrival |
| Postcondition(s) | Estimated t.o.a. Is presented to the citizen |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table #SF\_C3: Get information about the bus (line/color/bus photo)

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C3 |
| Name | Get information about the bus (line/color/bus photo) |
| Input Data | - |
| Precondition | Bus is driving on the route |
| Description | Provides detailed information about the bus on the route |
| Exception | Bus information doesn’t exists (e.g. a bus that is not in the database has to be used because of reasons) |
| Business rules | One successful transmission per bus line |
| Quality Management | Information should match the used bus (color matches the bus photo, bus photo shows the used bus) |
| Output Data | Information about the bus |
| Postcondition(s) | Citizen knows how the bus looks like |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table #SF\_C4: Request a stop at a stop /iteration 2

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C4 |
| Name | Request a stop at a stop /iteration 2 |
| Input Data | Bus station, time of departure |
| Precondition | Bus is scheduled |
| Description | Requests a stop of the bus at the selected bus station |
| Exception | Bus is on it’s last round and already passed the bus station |
| Business rules | Retry transmission every 5-10s |
| Quality Management | Transmission state is visible,  notify user about the planned schedule (15 min before t.o.d.) |
| Output Data | Answer to the request (accepted, declined) |
| Postcondition(s) | Citizen know if the bus stops at her station |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table #SF\_C5: Notify the user about changes in the schedule /iteration 2

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C5 |
| Name | Notify the user about changes in the schedule /iteration 2 |
| Input Data | Schedule changes |
| Precondition | App is running on the citizen device |
| Description | Inform the citizen about changes in the schedule |
| Exception | - |
| Business rules | Check at least once per (minimal driving time between two successive bus stations) for changes or push notifications |
| Quality Management | Display changes in an easy to understandable form (e.g.   * The bus will be xy minutes too late * The bus will skip station xy) |
| Output Data | Notification with schedule changes |
| Postcondition(s) | The citizen knows about the changes in the schedule |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table #SF\_C6: Show number of free seats

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C6 |
| Name | Show number of free seats /iteration x |
| Input Data | - |
| Precondition | Bus drives on the route |
| Description | Shows the number of free seats in the bus |
| Exception | - |
| Business rules | Check no more than the minimal driving time between two successive bus stations |
| Quality Management | Different text design for 100-50%, 49-25% and 24-0% free seats |
| Output Data | Number of free seats in the bus |
| Postcondition(s) | Citizen knows if the bus currently has a free seat for her |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table #SF\_C7: Request help for shopping /iteration x

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C7 |
| Name | Request help for shopping /iteration x |
| Input Data | - |
| Precondition | Bus is driving on the route, #FC04 is successful |
| Description | Requests help for shopping |
| Exception | - |
| Business rules | See #SF\_C4 |
| Quality Management | See #SF\_C4 |
| Output Data | See #SF\_C4 |
| Postcondition(s) | Citizen knows if she will have help with the shopping trip |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table #SF\_C8: Schedule wayback /iteration x

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C8 |
| Name | Schedule wayback /iteration x |
| Input Data | Time of departure |
| Precondition | Bus is on the route |
| Description | Schedules the return of the citizen from the current destination to the start destination |
| Exception | T.o.d. is in the working time of the Bürgerbus |
| Business rules | Manage via predated #FC04 |
| Quality Management | See #SF\_C4 |
| Output Data | See #SF\_C4 |
| Postcondition(s) | Citizen planned her way back |

### 

### Table #SF\_C9: Show schedule for current stop

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C9 |
| Name | Show schedule for current stop |
| Input Data | Stop identifier |
| Precondition | none |
| Description | Shows the schedule for the current stop |
| Exception | No data |
| Business rules |  |
| Quality Management |  |
| Output Data | Schedule data for the current stop |
| Postcondition(s) | Citizen knows when a bus arrives at the current stop |

**Author:** Steffen Holzer,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table #SF\_C10: Show List of Stops

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C10 |
| Name | Show List of Stops |
| Input Data | none |
| Precondition | none |
| Description | Shows all available stops |
| Exception | No data |
| Business rules |  |
| Quality Management |  |
| Output Data | List of all available stops |
| Postcondition(s) | Citizen knows where the bus stops |

### 

### Table #SF\_C11: Request a stop at a given location /iteration 2

|  |  |
| --- | --- |
| Item | Description |
| ID | #SF\_C11 |
| Name | Request a stop at a given location /iteration 2 |
| Input Data | Current location, time of departure |
| Precondition | Bus is scheduled |
| Description | Requests a stop of the bus at the selected location |
| Exception | Bus is too far away from the user, user is going to be picked up during the next round. |
| Business rules | Retry transmission every 5-10s |
| Quality Management | Transmission state is visible,  notify user about the planned schedule (15 min before t.o.d.) |
| Output Data | Answer to the request (accepted, declined) |
| Postcondition(s) | Citizen know if the bus stops at its position |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Steffen Holzer

**Review status:** Complete

### 

Table 25:System Function for BürgerApp#10

**Author:** Steffen Holzer,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:**Complete

## 7.6 System Functions for the server

### Table #SF\_S1: Send list of lines

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S1** |
| **Name** | Send List of lines |
| **Input Data** | http-Get-Request |
| **Precondition** | *CitizenApp* or *DriverApp* sends http-GET-Request |
| **Description** | **Server:**   1. Queries the database for all lines contained in it 2. Sends list of all lines to requesting App |
| **Exception** | No data available |
| **Business rules** |  |
| **Quality Management** | Response time <=5s |
| **Output Data** | List of all lines with their properties |
| **Postcondition(s)** | The client app has access to all line data |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Mohammad Baniasad

**Review status:**Complete

### Table #SF\_S2: Send list of busses

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S2** |
| **Name** | Send List of busses |
| **Input Data** | http-GET-Request |
| **Precondition** | *CitizenApp* or *DriverApp* sends http-GET-Request |
| **Description** | **Server:**   1. Queries the database for all busses contained in it 2. Sends list of all busses to requesting App |
| **Exception** | No data available |
| **Business rules** |  |
| **Quality Management** | Response time <= 5s |
| **Output Data** | List of all busses with their properties |
| **Postcondition(s)** | The client app has access to all bus data |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Mohammad Baniasad

**Review status:**Complete

### Table #SF\_S3: Send List of stops

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S3** |
| **Name** | Send List of stops |
| **Input Data** | http-GET-Request |
| **Precondition** | *CitizenApp* or *DriverApp* sends http-GET-Request |
| **Description** | **Server:**   1. Queries the database for all stops contained in it 2. Sends list of all stops to requesting App |
| **Exception** | No data available |
| **Business rules** |  |
| **Quality Management** | Response time <= 5s |
| **Output Data** | List of all stops with their properties This include schedules of a stop |
| **Postcondition(s)** | The client app has access to all stop data |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Mohammad Baniasad

**Review status:**Complete

### Table #SF\_S4: Send List of routes

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S4** |
| **Name** | Send List of routes |
| **Input Data** | http-GET-Request |
| **Precondition** | *CitizenApp* or *DriverApp* sends http-GET-Request |
| **Description** | **Server:**   1. Queries the database for all routes contained in it 2. Sends list of all routes to requesting App |
| **Exception** | No data available |
| **Business rules** |  |
| **Quality Management** | Response time <= 5s |
| **Output Data** | List of all routes with their properties |
| **Postcondition(s)** | The client app has access to all route data |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:**Mohammad Baniasad

**Review status:**Complete

**Table #SF\_S5: Send properties of bus: *Deprecated***

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Mohammad Baniasad

**Review status:** Complete

### Table #SF\_S6: Send GPS-Data of Bus

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S6** |
| **Name** | Send GPS-Data of Bus |
| **Input Data** | http-GET-Request including bus reference |
| **Precondition** | *CitizenApp* or *DriverApp* sends http-GET-Request |
| **Description** | **Server:**   1. Queries the database for the GPS position of the referenced bus 2. Sends GPS position along with timestamp |
| **Exception** | No data available, referenced bus does not exist |
| **Business rules** |  |
| **Quality Management** | Response time <= 5s |
| **Output Data** | GPS position of bus with timestamp |
| **Postcondition(s)** | The client has access to bus position and can trace it to time |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Mohammad Baniasad

**Review status:** Complete

### Table #SF\_S7: Store bus and line

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S7** |
| **Name** | Store bus and line |
| **Input Data** | Received http-POST with parameters:  **lineId:** integer  **busId:** integer |
| **Precondition** | *DriverApplication* sends bus and line information to server |
| **Description** | **Server:**   1. Receives bus and line data 2. Stores lineId in bus table in column of bus with id busId |
| **Exception** | Referenced bus or line do not exist |
| **Business rules** |  |
| **Quality Management** |  |
| **Output Data** | - |
| **Postcondition(s)** | Line and bus information are stored |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Mohammad Baniasad

**Review status:** Complete

### Table #SF\_S8: Store GPS data

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S8** |
| **Name** | Store Location data of the Bus |
| **Input Data** | Received http-POST with parameters:  **busId:** integer  **GPS data:** [double, double] |
| **Precondition** | *DriverApp* sends GPS data to server |
| **Description** | **Server:**   1. Receives GPS data 2. Stores the GPS data in the data base for the given bus. |
| **Exception** | Bus does not exist |
| **Business rules** |  |
| **Quality Management** |  |
| **Output Data** | - |
| **Postcondition(s)** | GPS data is stored |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Hafiz

**Review status: Complete**

### Table #SF\_S9: Send latest timestamp

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S9** |
| **Name** | Send latest timestamp |
| **Input Data** | Http-GET-request |
| **Precondition** | *CitizenApp* or *DriverApp* sends http-GET-request |
| **Description** | **Server:**   1. Retrieves the time of latest updates of bus, route, line and stop information from database 2. Sends latest update times to the client app. |
| **Exception** | No data available |
| **Business rules** |  |
| **Quality Management** | Response time<=5s |
| **Output Data** | Last timestamps for bus, route, line and stop |
| **Postcondition(s)** | Client app has access to latest timestamps |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:**

**Review status:**

### Table #SF\_S10: Notification of custom stop request

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S10** - Iteration 3 |
| **Name** | Notification of custom stop request |
| **Input Data** | Received http post request with following parameters:  **Position:** JSON object  **Departure Time:** Timestamp |
| **Precondition** | *CitizenApplication* sends a stop request |
| **Description** | **Server:**   1. Receives stop request from *CitizenApplication.* 2. Saves the position, departure time, user name, number of passengers to the database. 3. Sends request ID as confirmation to *CitizenApplication.* 4. Sends notification to the *DriverApplication*, notifying the driver serving the corresponding line. |
| **Exception** | No driver available to check requests. |
| **Business rules** |  |
| **Quality Management** |  |
| **Output Data** | *DriverApplication* receives a notification with location and departure time.  **Position:** JSON object  **Departure Time:** Timestamp |
| **Postcondition(s)** | Request sent to Bus Driver. |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Sriram kumar Srinivasan

**Review status:** Complete

### Table #SF\_S11: Response to custom stop request

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S11** - Iteration 3 |
| **Name** | Response to custom stop request |
| **Input Data** | Received http POST request with following parameters:  **requestId:** integer  **reqResponse**: integer |
| **Precondition** | Custom Stop request has been sent to the Bus Driver, *DriverApp* sends response to custom stop |
| **Description** | **Server:**   1. Receive response for the custom stop request from *DriverApp*. 2. Update response (Accepted/Declined) in Database. |
| **Exception** |  |
| **Business rules** |  |
| **Quality Management** | Response should be given within 30 min |
| **Output Data** | - |
| **Postcondition(s)** | Driver’s response is saved in the Database. |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Sriram kumar Srinivasan

**Review status:** Complete

### Table #SF\_S12: Transferring received help request

***Included in #SF\_S10: Notification of custom stop request***

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Mohammad Baniasad

**Review status:** TODO

### Table #SF\_S13: Send help request response

***Included in #SF\_S11: Response to custom stop request***

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Mohammad Baniasad

**Review status:** TODO

### Table #SF\_S14: Send number of available seats

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S14** - Iteration 4 |
| **Name** | Send number of available seats |
| **Input Data** | Http-GET-request with parameter:  **busId:** integer |
| **Precondition** | Number of available seats has been provided by the *Driver*App |
| **Description** | **Server:**   1. Queries database number of available seats of bus against busId 2. Sends Number of available seats of the referenced bus to the Client app |
| **Exception** | No seat number available |
| **Business rules** |  |
| **Quality Management** | Response time <=5s |
| **Output Data** | Number of available seats on bus |
| **Postcondition(s)** | Client received driver’s response |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Hafiz

**Review status:** Complete

**Remarks:** I’m not sure about the QM aspect

### Table #SF\_S15: Store number of available seats

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S15** - Iteration 4 |
| **Name** | Store number of available seats |
| **Input Data** | Http-POST-request with parameters:  **busId:** integer  **Number of seats:** integer |
| **Precondition** | *DriverApp* sends number of available seats to server |
| **Description** | **Server:**   1. Receives number of available seats for bus with busId 2. Number of available seats on the referenced bus is stored in the database |
| **Exception** | Bus does not exist |
| **Business rules** |  |
| **Quality Management** | Response time <=5s |
| **Output Data** | - |
| **Postcondition(s)** | Number of available seats is stored in database |

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:** Hafiz

**Review status** Complete

### Table #SF\_S16: Send status of Custom Stop request

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | #**SF\_S16** - Iteration 3 |
| **Name** | Send status of Custom Stop request |
| **Input Data** | Receive http get request with following parameters:  **requestID**: Integer |
| **Precondition** | Custom Stop request has been sent to Bus Driver, *CitizenApplication* sends a http-GET-request |
| **Description** | **Server:**   1. Queries database for custom Stop with id requestId 2. Sends the status of the request to the client app. 3. A default value is sent if there is no status (eg. “Status Unknown”, “Status Pending”) |
| **Exception** | No response was received from the driver |
| **Business rules** |  |
| **Quality Management** |  |
| **Output Data** | Custom Stop request’s status is sent to the *CitizenApplication*  **reqResponse**: String |
| **Postcondition(s)** | *CitizenApp*  received the status of his Custom Stop request. |

**Author:** Sriram kumar Srinivasan

**Status:** Complete

**Reviewer:** Ricarda Rosemann

**Review status:** Complete

FUT\_S1: Communicate schedule changes

**Author:** Ricarda Rosemann

**Status:** Complete

**Reviewer:**

**Review status:**

# App Functionality

## 8.1 Interaction Cases BürgerApp

### Table IC\_C1

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | **IC\_C1** |
| **Usage Context** | **The user wants to check when the bus arrives** |
| **Screen Arrangement 1** | **The screen shows:**   * The select stop screen shows the available stops   **The screen includes:**   * A scroll view with the available bus stops * A search input box * A button to set the current location as the bus stop |
| **Human Action 1** | **The user starts the application**  **Usage type: single tap** |
| **System Action 1** | **The application is started,**   1. **the system fetches the bus stop data (SF\_S3: Send list of stops)** 2. **shows the “select stop” screen (SC\_bbv)** |
| **Human Action 2** | **The user selects the stop or current location** |
| **System Action 2** | **The system fetches the detail data about the location / stop and shows the “stop detail” screen** |
| **Post conditions** | **The system shows the “stop detail” screen** |
| **Reference to Systemfunction** | **SF\_S3: Send list of stops** |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table IC\_C2

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | **IC\_C2** |
| **Usage Context** | **The user wants to see more information about the bus** |
| **Screen Arrangement 2** | **The screen shows:**   * Detailed information about the bus   **The screen includes:**   * Estimated arrival of the bus * Bus line number / color * Map with the position of the bus (fixed, without zoom or pan option) (#SF\_S6: Send GPS-Data) |
| Pre conditions | **User is at the “stop detail” screen** |
| **Human Action 1** | **User clicks at the show bus details button** |
| **System Action 1** | **System switches the view to the “bus information” screen** |
| **Post conditions** | **The system shows the “bus information” screen** |
| **Reference to System function** | **#SF\_S6: Send GPS-Data, #SF\_S14: Send number of available seats** |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table IC\_C3

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | **IC\_C3** |
| **Usage Context** | **The user wants to get detailed information about the position of the bus** |
| **Screen Arrangement 1** | **The screen shows:**   * Detailed information about the position of the bus   **The screen includes:**   * A full feature native mapview with option to pan and zoom |
| **Pre conditions** | **User is now at the “bus information” screen** |
| **Human Action 1** | **User taps on the map** |
| **System Action 1** | **System switches the view to the “bus position” screen** |
| **Post conditions** | **The system shows the “bus position” screen** |
| **Reference to System function** | **#SF\_S6: Send GPS-Data, #SF\_S14: Send number of available seats** |

**Author:** Dominik Skalnik,

**Status:** Complete

**Reviewer:** Marcel Müller

**Review status:** Complete

### Table IC\_C4

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | **IC\_C4** |
| **Usage Context** | **The user request a stop.** |
| **Screen Arrangement 2** | **The screen shows:**   * Request stop wish   **The screen includes:**   * Place where to get picked up * Line * Time when to be picked up * Amount of persons to be picked up * Users Name * Users Adress * Options for required help (baggage, wheelchair, support at shopping tasks) |
| Pre conditions | **Request a stop wish clicked** |
| **Human Action 1** | **The Citizen User customizes his request and presses accept.** |
| **System Action 1** | **The system shows the Start screen** |
| **Post conditions** | **-** |
| **Reference to System function** | **#SF\_S10: Notification of custom stop request** |

## 

## 8.2 Interaction Cases BusDriveApp

### Table #IC\_B1: Setting up for driving

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | **#IC\_B1** |
| **Usage Context** | **The Bus Driver wants to set up for driving** |
| **Pre conditions** | **Application is installed on device** |
| **Human Action 1** | **The Bus Driver starts the BusDriveApp** |
| **System Action 1** | **The system shows the #SC\_B0: Start Screen** |
| **Screen Arrangement 1** | **The screen shows:**   * #SC\_B0: Start Screen   **The screen includes**   * A ‘START TOUR’ button |
| **Human Action 2** | **The Bus Driver tabs the ‘START TOUR’ button** |
| **System Action 2** | **The system receives a list of busses provided by the server (#SF\_B0.1: Request bus list, #SF\_S2: Send list of busses)** |
| **Screen Arrangement 2** | **The screen shows:**   * #SC\_B1: Select Bus Screen   **The screen includes:**   * A list of available busses |
| **Human Action 3** | **The Bus Driver selects a bus** |
| **System Action 3** | **The system remembers the choice of the Bus Driver and shows the #SC\_B2: Select Line Screen (SF\_B1: Select bus)** |
| **System Action 4** | **The system receives a list of line provided by the server (#SF\_B0.2: Request line list, #SF\_S1: Send list of lines)** |
| **Screen Arrangement 3** | **The screen shows:**   * #SC\_B2: Select Line Screen   **The screen includes:**   * A list of available lines |
| **Human Action 4** | **The Bus Driver selects a line** |
| **System Action 5** | **The system remembers the choice of the Bus Driver, shows the #SC\_B3.1: Drive Screen - Main Tab (#SF\_B2: Select Busline)** |
| **System Action 6** | **The system sends the selected bus and line to the server (#SF\_B3.1: Send Bus Status Data to server)** |
| **Screen Arrangement 4** | **The screen shows:**   * #SC\_B3.1: Drive Screen - Main Tab   **The screen includes:**   * Next Stop * Number of passengers |
| **Post conditions** | **The system is set up for driving and shows the #SC\_B3.1: Drive Screen - Main Tab** |
| **System Functions** | **#SF\_B0.1: Request bus list, #SF\_B0.2: Request line list, #SF\_B1: Select bus, #SF\_B2: Select Busline, #SF\_S1: Send list of lines, #SF\_S2: Send list of busses, #SF\_B3.1: Send Bus Status Data to server** |

**Author:** Oliver Säger, Sascha Müller

**Status:** Complete

**Reviewer:** Charel Irrthum, Patrick Pschorn

**Review status:** Complete

### Table #IC\_B2: Document number of Citizens

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | **#IC\_B2** |
| **Usage Context** | **The Bus Driver wants to document the number of Citizens riding the bus** |
| **Screen Arrangement** | **The screen shows:**   * #SC\_B3.1: Drive Screen - Main Tab   **The screen includes:**   * Next Stop * Number of passengers * Options to increase and decrease the number of people riding the bus * Options to cycle through displayed Stops |
| **Pre conditions** | **The Bus Driver sets up the BusDriveApp and is shown the #SC\_B3.1: Drive Screen - Main Tab** |
| **Human Action 1** | 1. **The Bus Driver taps the “Increase” button** 2. **The Bus Driver taps the “Decrease” button** |
| **System Action 1.1** | **The displayed number of passengers is increased by one (#SF\_B4: Change number of taken seats)** |
| **System Action 1.2** | **The displayed number of passengers is decreased by one (#SF\_B4: Change number of taken seats)** |
| **Post conditions** | **The number of Citizens inside the bus is displayed on the screen and the next time real time data is sent to the server it includes the updated number (#SF\_B3.2: Send Real Time Data to server)** |
| **System Functions** | **#SF\_B3.2: Send Real Time Data to server, #SF\_B4: Change number of taken seats** |

**Author:** Oliver Säger, Sascha Müller

**Status:** Complete

**Reviewer:** Patrick Pschorn

**Review status:** Complete

### Table #IC\_B3: See route

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | **#IC\_B3** |
| **Usage Context** | **The Bus Driver wants to see the route of the line he selected** |
| **Pre conditions** | **The Bus Driver sets up the BusDriveApp and is shown the #SC\_B3.1: Drive Screen - Main Tab** |
| **Screen Arrangement 1** | **The screen shows:**   * #SC\_B3.1: Drive Screen - Main Tab   **The screen includes:**   * Next Stop * Number of passengers |
| **Human Action 1** | **The Bus Driver selects the Map tab** |
| **System Action 1** | **The BusDriveApp shows the #SC\_B5: Map Screen (#SF\_B6: Show map of current position)** |
| **Screen Arrangement 2** | **The screen shows:**   * #SC\_B5: Map Screen   **The screen includes:**   * Map that displays   + The position of the bus   + Markers for each Line Stop   + Markers for each Custom Stop   + A route that connects the Line Stops |
| **Post conditions** | **The BusDriveApp shows the #SC\_B5: Map Screen where the route is displayed on the map** |
| **System Functions** | **#SF\_B6: Show map of current position** |

**Author:** Oliver Säger, Sascha Müller

**Status:** Complete

**Reviewer:** Patrick Pschorn

**Review status:** Complete

### Table #IC\_B4: See stops

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | **#IC\_B4** |
| **Usage Context** | **The Bus Driver wants to see an overview of the line** |
| **Pre conditions** | **The Bus Driver sets up the BusDriveApp and is shown the #SC\_B3.1: Drive Screen - Main Tab** |
| **Screen Arrangement 1** | **The screen shows:**   * #SC\_B3.1: Drive Screen - Main Tab   **The screen includes:**   * Next Stop * Number of passengers |
| **Human Action 1** | **The Bus Driver taps the Stops tab** |
| **System Action 1** | **The BusDriveApp shows the #SC\_B4: Stops screen (#SF\_B7: Show Line Stops)** |
| **Screen Arrangement 2** | **The screen shows:**   * #SC\_B4: Stops Screen   **The screen includes:**   * A List of all stops of the line (including arrival time) |
| **Post conditions** | **The BusDriveApp shows the #SC\_B4: Stops Screen where a list of all stops is displayed** |
| **System Functions** | **#SF\_B7: Show Line Stops** |

**Author:** Oliver Säger, Sascha Müller

**Status:** Complete

**Reviewer:** Patrick Pschorn

**Review status:** Complete

### Table #IC\_B5: Receiving a Custom Stop Request

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | **#IC\_B5** |
| **Usage Context** | **Receiving a Custom Stop Request** |
| **Pre conditions** | **The Bus Driver sets up the BusDriveApp and is shown the #SC\_B3.1: Drive Screen** |
| **Screen Arrangement 1** | **The screen shows:**   * #SC\_B3.1: Drive Screen - Main Tab   **The screen includes:**   * Next Stop * Number of passengers |
| **System Action 1** | **The BusDriveApp receives a Custom Stop Request and shows a notification that new Custom Stop Requests are pending (#SF\_B5: Receive Custom Stop Request)** |
| **Human Action 1** | **The Bus Driver taps the notification or the #SC\_B3.2: Drive Screen - Custom Stops Tab** |
| **System Action 2** | **The BusDriveApp shows the #SC\_3.2: Drive Screen - Custom Stops Tab** |
| **Screen Arrangement 2** | **The screen shows:**   * #SC\_3.2: Drive Screen - Custom Stops Tab   **The screen includes:**   * A List of all pending Custom Stop Requests (if you swipe an element there are buttons that say ‘accept’ and ‘decline’) |
| **Human Action 2** | **The Bus Driver swipes an element of the list and taps the option ‘accept’** |
| **System Action 3** | **The element of the list will disappear and will be moved to the ‘SC\_B3.1: Drive Screen - Main Tab (#SF\_B9: Respond to Custom Stop Request)** |
| **System Action 4** | **The BusDriveApp sends the status update of the Custom Stop (‘accepted’) to the server (#SF\_B3.3: Send Custom Stop status update to server)** |
| **Human Action 3** | **The Bus Driver taps the Drive Screen - Main Tab** |
| **System Action 5** | **The BusDriveApp shows the #SC\_B3.1: Drive Screen - Main Tab** |
| **Screen Arrangement 3** | **The screen shows:**   * #SC\_B3.1: Drive Screen - Main Tab   **The screen NOW includes:**   * Next Stop * Number of passengers * A List of all accepted Custom Stops |
| **Post conditions** | **The Custom Stop is displayed on the Drive Screen - Main Tab and the server received the updated status of the Custom Stop (#SF\_S11: Response to custom stop request)** |
| **System Functions** | **#SF\_B5: Receive Custom Stop Request, #SF\_B9: Respond to Custom Stop Request, #SF\_B3.3; Send Custom Stop status update to server, #SF\_S11: Response to custom stop request** |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Patrick Pschorn

**Review status:** Complete

### Table #IC\_B6: Completing a Custom Stop

|  |  |
| --- | --- |
| **Item** | **Description** |
| **ID** | **#IC\_B6** |
| **Usage Context** | **Completing a Custom Stop** |
| **Pre conditions** | **The Bus Driver accepted at least one Custom Stop Request and the app shows the #SC\_B3.1: Drive Screen - Main Tab** |
| **Screen Arrangement 1** | **The screen shows:**   * #SC\_B3.1: Drive Screen - Main Tab   **The screen includes:**   * Next Stop * Number of passengers * List of all accepted Custom Stops |
| **Human Action 1** | **(After the Citizen enters the bus) The Bus Driver swipes the corresponding element of the Custom Stop list and taps the option ‘completed’** |
| **System Action 1** | **The element of the list disappears (#SF\_B10: Complete Custom Stop)** |
| **System Action 2** | **The BusDriveApp sends the status update of the Custom Stop (‘completed’) to the server (#SF\_B3.3: Send Custom Stop status update to server)** |
| **Post conditions** | **The Custom Stop is completed, the Bus Driver continues his tour and the server received the updated status of the Custom Stop (#SF\_S11: Response to custom stop request)** |
| **System Functions** | **#SF\_B10: Complete Custom Stop Request, #SF\_B3.3: Send Custom Stop status update to server, #SF\_S11: Response to custom stop request** |

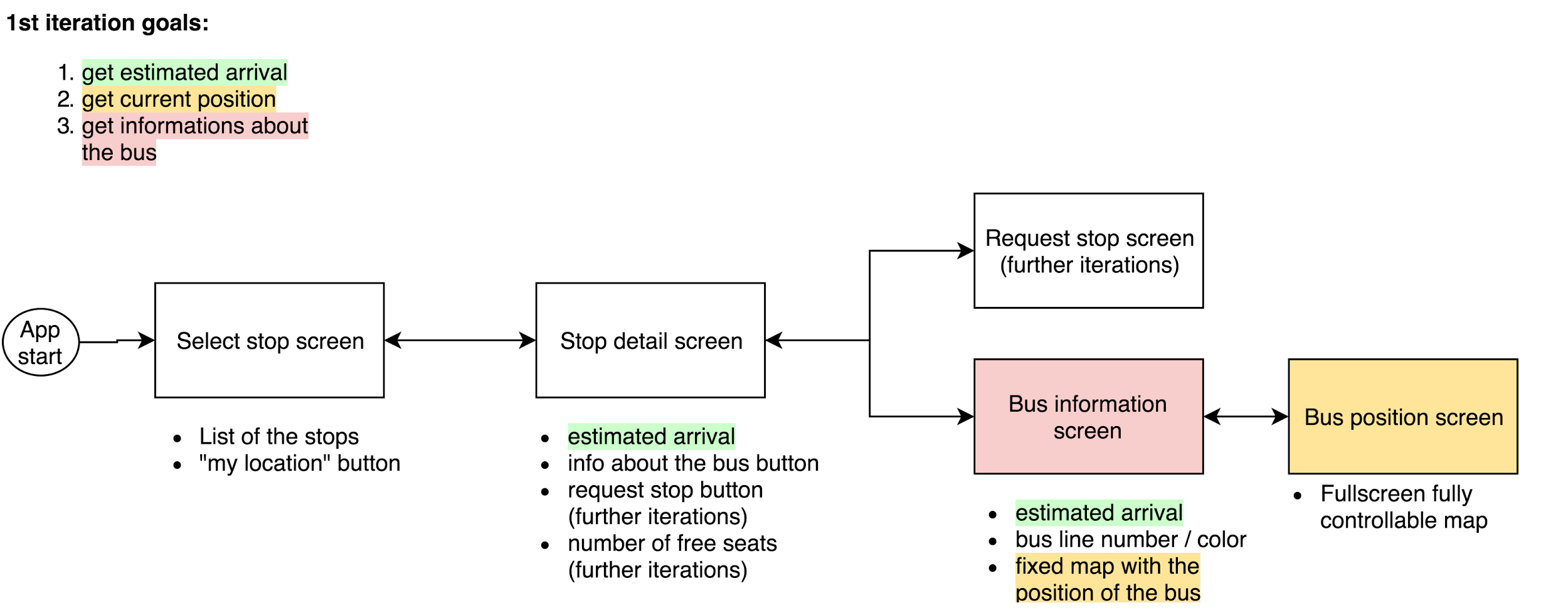
**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Oliver Säger

**Review status:** Complete

## 8.3 Screenflow BürgerApp



**Author:** Dominik Skalnik,

**Status:** Complete

Figure xx: Screenflow of the citizen application

## 8.4 BusDriveApp Screens [(Screenflow)](https://github.com/GSE-Project/SS2016-group3/blob/develop/Doc/Architecture/GUI%20-%20Flow.pdf)

|  |  |
| --- | --- |
| #SC\_B0: Start Screen This is the screen which is displayed when the Bus Driver starts the app. He can press the button ‘START TOUR’ to be forwarded to the Select Bus Screen.  On the symbol on the upper left corner he can switch to the Settings Screen (or the About Screen). |  |
| #SC\_B1: Select Bus Screen The screen shows a list of the busses available including number plate and a picture. The bus is chosen by tapping the element and the app changes to the Select Line Screen. |  |
| #SC\_B2: Select Line Screen The screen shows a list of the lines. A Line is chosen by tapping the element and the app changes to the Drive Screen. |  |
| #SC\_B3.1: Drive Screen - Main Tab On this Tab of the Drive Screen screen the Bus Driver can adjust the number of passengers on his bus by pressing the ‘+’/’-’ buttons. On the top of the screen he can go back to the Start Screen by pressing the ‘<’ button.  The next Line Stop of his route is shown and the Bus Driver can cycle through the Line Stop by pressing the ‘<-’/’->’ buttons.  On the bottom of the screen all accepted Custom Stops are displayed. The Bus Driver can swipe the elements to mark them as ‘completed’.  Via the lower tab bar the screen can be changed to the Drive Screen - New Custom Stops Tab.  Via the upper tab bar the screen can be changed to the Map Screen and Stops Screen. |  |
| #SC\_B3.2: Drive Screen - New Custom Stops Tab The screen shows a list of all pending Custom Stop Requests relevant to the Bus Driver. If he swipes an element he can choose to accept or decline the request.  Via the lower tab bar the screen can be changed to the Drive Screen - Main Tab.  Via tab bar the screen can be changed to the Map Screen and Drive Screen. |  |
| #SC\_B4: Stops Screen On this screen the Bus Driver can see the list of all Line Stops on his current route.  Via tab bar the screen can be changed to the Map Screen and Drive Screen. |  |
| #SC\_B5: Map Screen The screen shows the map of the area of the chosen route including connected markers for the Line Stops, Custom Stops and the Bus Driver’s current position.  Via tab bar the screen can be changed to the Stops Screen and Drive Screen. |  |
| #SC\_B6: Settings Screen The screen shows several options to configure the app so the Bus Driver can:   * Change the language * Set the server address * Change additional settings |  |
| (#SC\_BX: About Screen) The screen shows several addresses and by tapping them the browser will open and show the corresponding web address. Also includes Imprint, Privacy Policy and License |  |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Patrick Pschorn, Oliver Säger

**Review status:** Complete

## 8.5 BuergerApp Screens

|  |  |
| --- | --- |
| #SC\_C0: Start Screen | Bildschirmfoto 2016-06-27 um 20.53.52.png  Screenshot_20160627-202414.png |
| #SC\_C1: Request Stop | Bildschirmfoto 2016-06-27 um 20.54.16.png  Screenshot_20160627-202420.png |
| #SC\_C2.1: Show Bus Information | Bildschirmfoto 2016-06-27 um 20.53.47.png  Screenshot_20160627-202037.png |
| #SC\_C2.2: Show Bus Position | Screenshot_20160627-202031.png |
| #SC\_C3: Show Stops | Bildschirmfoto 2016-06-27 um 20.53.56.png  Screenshot_20160627-202045.png |
| #SC\_C3.1: Show Routes | Bildschirmfoto 2016-06-27 um 20.59.04.png  Screenshot_20160627-202041.png |
| #SC\_C4: Show Map | Screenshot_20160627-202007.pngScreenshot_20160627-202014.png |

**Author:** Marcel Müller

**Status:** incomplete

**Reviewer:**

**Review status:**

# Additional implemented elements of BusDriveApp at current state

### Table Additional implemented elements

|  |  |
| --- | --- |
| Name | Description |
| BusDriveApp: Change Language | For later localisation of the BusDriveApp, you can now change the language of the application on the Settings Screen |
| BusDriveApp:  Set Server Address | On the Settings Screen you can choose a server address and also add new server addresses. |
| BusDriveApp:  Route to Custom Stop | If you tap on the marker for an accepted Custom Stop or if you tap on the address of the Custom Stop on the #SC\_B3: Drive Screen the route from your position to the Custom Stop is displayed |

**Author:** Sascha Müller, Charel Irrthum

**Status:** Complete

**Reviewer:** Patrick Pschorn, Oliver Säger

**Review status:** Complete

# Glossary

### Table Glossary

|  |  |
| --- | --- |
| Name | Description |
| Bürgerbus system | Consists of BusDriveApp, BürgerApp, server backend |
| BusDriveApp | Mobile application to be used by the Bus Driver |
| BürgerApp | Mobile application to be used by the Citizens |
| Line Stop | A stop that is constantly part of one or more Lines, with estimated arrival time within the corresponding schedule |
| Custom Stop | A stop that is not in the schedule of any Line, the position can be off the normal Lines, a Citizen requests it, the server propagates it to an appropriate Bus Driver and the Bus Driver decides if he includes it to his tour and tries to be there on time |

**Author:** Sascha Müller

**Status:** Complete

**Reviewer:** Charel Irrthum, Oliver Säger

**Review status:** Complete