Deliverable #1 Template

SE 3A04: Software Design II – Large System Design

1 Introduction

The following documents is dedicated to showcasing the requirements of the taxi carpool app requested by a local taxi company. The app is dedicated to make rides cheaper and more convenient for their riders. This app is also meant to be a long-term project for the company, meaning this SRS is a living document which has the possibility of changing in the future. This document will explain the important stakeholders involved, requirements and viewpoints of each while also explaining the main functions of the app

1.1 Purpose

This SRS is meant for developers, stakeholders and direct players in the planning and development in the app and should be understandable for all those involved. This SRS will further break down the functionalities, constraints, use cases and overall design of the project in further detail.

1.2 Scope

To develop this project the production of the main taxi request and offer app called TaxiCarPool, the Conversation Prompt Generator and a variety of databases will have to be produced. There will not be a need to produce a payment system into the app as this will be handled by the taxi drivers. The project will only project the cost of the ride but will not handle the payments as it is not part of the app's main functionality.

The TaxiCarPool app will perform the main functions of the app. This will include allowing users to request a carpool taxi, receive a list of possible rides to select and allow current rides to allow carpooling in their car. This product's objective is to perform the main functions of match making users for a carpool to save money.

The Conversation Prompt Generator will randomly generate a conversation starter for the riders to use based on their profile. Their profile will specify if they want to use this feature and if so, what they are interested in. The generator will not be forced upon any of the user and can be switched off if desired. The goal of this product is to create a friendly environment when carpooling with others that one may not know. This function is mostly dedicated to benefit those who are carpooling to a further distance.

The creation of a database will hold all the necessary information for the app to function, making it easily accessible and manageable. The information being stored can range from user information; name, email, phone number, to riding history, to preferences of taxi times, models, or other passengers.

1.3 Definitions, Acronyms, and Abbreviations

BE: Business Event is something that occurs between the client/stakeholder and the system.

SRS: Software Requirements Specification is a document that explains the retirements of the software being developed.

VP: Viewpoint, often in reference to a stakeholder/client interested in the system and how they view the business event. should be in alphabetical order.

1.4 References

- Provide a complete list of all documents referenced elsewhere in the SRS
- Identify each document by title, report number (if applicable), date, and publishing organization
- Specify the sources from which the references can be obtained
- Order this list in some sensible manner (alphabetical, or something else that makes more sense)

1.5 Overview

This SRS is structured to first give background information about the app. This will include where the project will stand in the taxi company along with what are the general functionalities and assumptions are being made. Mainly describing the background information needed to understand the problem. It will then follow with showcasing one of the main business events and its related use cases. This will explain how the main function will work across different scenarios and stakeholders in a visual manner. This will later be further expanded upon in more detail in the document. It will include more business events with a variety of viewpoints and scenarios for both the function shown in the use case diagram and other main function of the project. This document will conclude by explaining the non-functional requirements involving presentation, performance, usability, security, and maintainability notes of the project.

2 Overall Description

- This section of the SRS should describe the general factors that affect the product and its requirements.
- It does not state specific requirements.
- It provides a *background* for those requirements and makes them easier to understand.

2.1 Product Perspective

- Put the product into perspective with other related products, i.e., context
- If the product is independent and totally self-contained, it should be stated here
- If the SRS defines a product that is a component of a larger system, then this subsection should relate the requirements of that larger system to the functionality of the software being developed. Identify interfaces between that larger system and the software to be developed.
- A block diagram showing the major components of the larger system, interconnections, and external interfaces can be helpful

2.2 Product Functions

- Provide a summary of the major functions that the software will perform.
  - **Example**: An SRS for an accounting program may use this part to address customer account maintenance, customer statement, and invoice preparation without mentioning the vast amount of detail that each of those functions requires.
- Functions should be organized in a way that makes the list of functions understandable to the customer or to anyone else reading the document for the first time
- Present the functions in a list format - each item should be one function, with a brief description of it
- Textual or graphical methods can be used to show the different functions and their relationships
  - Such a diagram is not intended to show a design of a product, but simply shows the logical relationships among variables

2.3 User Characteristics

- Describe those general characteristics of the intended users of the product including educational level, experience, and technical expertise
- Since there will be many users, you may wish to divide into different user types or personas

2.4 Constraints

- Provide a general description of any constraints that will limit the developer's options

2.5 Assumptions and Dependencies

- List any assumptions you made in interpreting what the software being developed is aiming to achieve
- List any other assumptions you made that, if it fails to hold, could require you to change the requirements
  - **Example**: An assumption may be that a specific operating system will be available on the hardware designated for the software product. If, in fact, the operating system is not available, the SRS would then have to change accordingly.

2.6 Apportioning of Requirements

- Identify requirements that may be delayed until future versions of the system

3 Use Case Diagram

- Provide *one* use case diagram for the most important Business Event.
- The text of all use cases will be specified under "Highlights of Functional Requirements"

4 Highlights of Functional Requirements

- Specify the "use cases" organized by Business Event. (The Global Scenario is what you might think of as a use case). Be sure to consider Business Events that aren't just triggered by users with goals (e.g. something happens in the environment that your system needs to respond to)
- Your focus should be on what the system needs to do, not how to do it. Specify it in enough detail that it clearly specifies what needs to be accomplished, but not so detailed that you start programming or making design decisions.
- Keep the length of each use case (Global Scenario) manageable. If it's getting too long, you need to condense your steps and give a name to what's accomplished by that sequence of steps. (e.g. "Authenticate user" in one line, instead of a list of steps of how to; that's a design decision anyways)
- You are *not* specifying a complete and consistent set of functional requirements here. (i.e. you are providing them in the form of use cases/global scenarios, not a refined list). For the purpose of this project, you do not need to reduce them to a list; the global scenarios format is all you need.

Below, we organize by Business Event.

BE1. Business Event name

VP1.1 Viewpoint name

- S_1**:** Initial response of the system to the Business Event
- E_1**:** Reaction of the environment to S_1
- S_2**:** Response of the system to E_1
- E_2**:** Reaction of the environment to S_2

  ...
- S_n**:** Response of the system to $E_{(n-1)}$
- E_n**:** Reaction of the environment to $E_{(n-1)}$
- $S_{(n+1)}$**:** Final response of the system concluding its function regarding the Business Event

VP1.2 Viewpoint name

- S_1**:** Initial response of the system to the Business Event
- E_1**:** Reaction of the environment to S_1
- S_2**:** Response of the system to E_1
- E_2**:** Reaction of the environment to S_2

  ...
- S_k**:** Response of the system to $E_{(k-1)}$
- E_k**:** Reaction of the environment to $E_{(k-1)}$
- $S_{(k+1)}$**:** Final response of the system concluding its function regarding the Business Event

VP1.3 ...

VP1.4 ...

VP1.5 ...

...

Global Scenario of *Business Event Name***:** It is the scenario corresponding to the integration of all the above scenarios from the different Viewpoints of the Business Event BE1.

- S_1**:** Initial response of the system to the Business Event
- E_1**:** Reaction of the environment to S_1
- S_2**:** Response of the system to E_1
- E_2**:** Reaction of the environment to S_2

  ...
- S_m**:** Response of the system to $E_{(m-1)}$
- E_m**:** Reaction of the environment to $E_{(m-1)}$
- $S_{(m+1)}$**:** Final response of the system concluding its function regarding the Business Event

BE2. Business Event name

VP1.1 Viewpoint name

- S_1**:** Initial response of the system to the Business Event
- E_1**:** Reaction of the environment to S_1
- S_2**:** Response of the system to E_1
- E_2**:** Reaction of the environment to S_2

  ...
- $S_{n'}$**:** Response of the system to $E_{(n'-1)}$
- $E_{n'}$**:** Reaction of the environment to $E_{(n'-1)}$
- $S_{(n'+1)}$**:** Final response of the system concluding its function regarding the Business Event

VP1.2 Viewpoint name

- S_1**:** Initial response of the system to the Business Event
- E_1**:** Reaction of the environment to S_1
- S_2**:** Response of the system to E_1
- E_2**:** Reaction of the environment to S_2

  ...
- $S_{k'}$**:** Response of the system to $E_{(k'-1)}$
- $E_{k'}$**:** Reaction of the environment to $E_{(k'-1)}$
- $S_{(k'+1)}$**:** Final response of the system concluding its function regarding the Business Event

VP1.3 ...

VP1.4 ...

VP1.5 ...

...

Global Scenario of *Business Event Name***:** It is the scenario corresponding to the integration of all the above scenarios from the different Viewpoints of the Business Event BE2.

- S_1**:** Initial response of the system to the Business Event
- E_1**:** Reaction of the environment to S_1
- S_2**:** Response of the system to E_1
- E_2**:** Reaction of the environment to S_2

  ...
- $S_{m'}$**:** Response of the system to $E_{(m'-1)}$
- $E_{m'}$**:** Reaction of the environment to $E_{(m'-1)}$
- $S_{(m'+1)}$**:** Final response of the system concluding its function regarding the Business Event

5 Non-Functional Requirements

- For each non-functional requirement, provide a justification/rationale for it.
  Example:
  SC1. *The device should not explode in a customer's pocket.*
  Rationale: Other companies have had issues with the batteries they used in their phones randomly

exploding [insert citation]. This causes a safety issue, as the phone is often carried in a person's hand or pocket.

- If you're making a guess because you couldn't really talk to stakeholders, you can say "We imagined stakeholders would want...because..."
- Each requirement should have a unique label/number for it.

5.1 Look and Feel Requirements

5.1.1 Appearance Requirements

LF-A1.

5.1.2 Style Requirements

LF-S1.

5.2 Usability and Humanity Requirements

5.2.1 Ease of Use Requirements

UH-EOU1.

5.2.2 Personalization and Internationalization Requirements

UH-PI1.

5.2.3 Learning Requirements

UH-L1.

5.2.4 Understandability and Politeness Requirements

UH-UP1.

5.2.5 Accessibility Requirements

UH-A1.

5.3 Performance Requirements

5.3.1 Speed and Latency Requirements

PR-SL1.

5.3.2 Safety-Critical Requirements

PR-SC1.

5.3.3 Precision or Accuracy Requirements

PR-PA1.

5.3.4 Reliability and Availability Requirements

PR-RA1.

5.3.5 Robustness or Fault-Tolerance Requirements

PR-RFT1.

5.3.6 Capacity Requirements

PR-C1.

5.3.7 Scalability or Extensibility Requirements

PR-SE1.

5.3.8 Longevity Requirements

PR-L1.

5.4 Operational and Environmental Requirements

5.4.1 Expected Physical Environment

OE-EPE1.

5.4.2 Requirements for Interfacing with Adjacent Systems

OE-IA1.

5.4.3 Productization Requirements

OE-P1.

5.4.4 Release Requirements

OE-R1.

5.5 Maintainability and Support Requirements

5.5.1 Maintenance Requirements

MS-M1.

5.5.2 Supportability Requirements

MS-S1.

5.5.3 Adaptability Requirements

MS-A1.

5.6 Security Requirements

5.6.1 Access Requirements

SR-AC1.

5.6.2 Integrity Requirements

SR-INT1.

5.6.3 Privacy Requirements

SR-P1.

5.6.4 Audit Requirements

SR-AU1.

5.6.5 Immunity Requirements

SR-IM1.

5.7 Cultural and Political Requirements

5.7.1 Cultural Requirements

CP-C1.

5.7.2 Political Requirements

CP-P1.

5.8 Legal Requirements

5.8.1 Compliance Requirements

LR-COMP1.

5.8.2 Standards Requirements

LR-STD1.

A Division of Labour

Include a Division of Labour sheet which indicates the contributions of each team member. This sheet must be signed by all team members.

IMPORTANT NOTES

- Be sure to include all sections of the template in your document regardless whether you have something to write for each or not
  - If you do not have anything to write in a section, indicate this by the *N/A*, *void*, *none*, etc.
- Uniquely number each of your requirements for easy identification and cross-referencing
- Highlight terms that are defined in Section 1.3 (**Definitions, Acronyms, and Abbreviations**) with **bold**, *italic* or underline
- For Deliverable 1, please highlight, in some fashion, all (you may have more than one) creative and innovative features. Your creative and innovative features will generally be described in Section 2.2 (**Product Functions**), but it will depend on the type of creative or innovative features you are including.