Software Requirements Specification

for

BusyBus

**Version 1.0**

**Prepared by**

|  |  |  |
| --- | --- | --- |
| **Dominic Olukoga** | **500000697** | **dominicolukoga@gmail.com** |
| **Kyle King** | **500000902** | **KingKyleTech@outlook.com** |
| **<name>** | **<ID #>** | **<e-mail>** |
| **<name>** | **<ID #>** | **<e-mail>** |
| **<name>** | **<ID #>** | **<e-mail>** |
| **<name>** | **<ID #>** | **<e-mail>** |
|  |  |  |

|  |  |
| --- | --- |
| **Course Instructor:** | **Dr Ilenius Ildephonce** |
| **Course:** | **COMP2225 – Software Engineering** |
| **Date:** | **10/31/22** |

TABLE of CONTENTS

**Revisions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| Draft Type and Number | Full Name | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | 00/00/00 |

# 

# *<In this template you will find text bounded by the “<>” symbols. This text appears in italics and is intended to guide you through the template and provide explanations regarding the different sections in this document. Please, make sure to delete all of the comments before submitting the document.*

# *The explanations provided below, do not cover all of the material, but merely, the general nature of the information you would usually find in SRS documents. It is based on the IEEE requirements and was adapted specifically for the needs of COMP2140. Most of the sections in this template are required sections, i.e. you must include them in your version of the document. Failure to do so will result in marks deduction. Optional sections will be explicitly marked as optional. “TO DO” are specific instructions on what should be done in each section.*

# Overall Description

## <items 1.1 – 1.3 in this section should be primarily based on the problem statement developed and presented earlier in the course>

## Product Context and Need

*<Describe the context and origin of the product being specified in this SRS.*

*TO DO: Give an overview of the organization or context for the system. Identify the purpose. This should serve as a background to the project/product>*

## Product Functionality

*<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 2, so only a high-level summary is needed here. Organize the functions to make them understandable to any reader of the SRS.*

*TO DO: Provide a list of all the major functions of the system at a very high level, not detailed>*

## Stakeholders and Users Characteristics

*<Identify the various users that you anticipate will use this product. Users may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience.*

*TO DO:*

1. *Identify the key stakeholders of this system.*
2. *Describe the pertinent characteristics of each user. Certain requirements may pertain only to certain users.*

*3. Distinguish the most important users for this product from those who are less important to satisfy.>*

## Operating Environment

*<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.*

*Optional: Include a simple diagram that shows the major components of the overall system, subsystem interconnections, and external interface*

*TO DO: As stated above, in at least one paragraph, describe the environment your system will have to operate in. Make sure to include the minimum platform requirements for your system. >*

## Design and Implementation Constraints

*<Describe any items or issues that will limit the options available to the developers. These might include: hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software).*

*TO DO: In this section, you need to consider all of the information you gathered so far, analyze it and correctly identify at least 3 constraints.>*

## Assumptions and Dependencies

*<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project.*

*TO DO: Provide a short list of some major assumptions that might significantly affect your design. For example, you can assume that your client will have 1, 2 or at most 50 Automated Banking Machines. Every number has a significant effect on the design of your system. >*

# Specific Requirements

## External Interface Requirements

### Hardware Interfaces

*<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware.*

*TO DO: Please provide a short description of the different hardware interfaces. If you will be using some special libraries to communicate with your software mention them here. In case you have more than one hardware interface divide this section into subsections.>*

### Software Interfaces

*<Describe the connections between this product and other specific software components (name and version), including databases, operating systems (Windows? Linux? Etc…), tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.*

*TO DO: You are only required to describe the specific interface with the operating system.>*

### Communications Interfaces

*<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.*

*TO DO: Do not go into too much detail, but provide 1-2 paragraphs outlining the major communication standards. For example, if you decide to use encryption there is no need to specify the exact encryption standards, but rather, specify the fact that the data will be encrypted and name what standards you consider using. >*

## Functional Requirements

*<*

*Functional requirements capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions the system is required to perform. Here, you should list in detail the different product functions with specific explanations regarding every function. Each requirement should be numbered. The following template is to be used to specify each functional requirement:*

***Requirement #: <val****> <****Title****>*

**Use Case:** *<name or number of use case that this requirement relates to>*

**Rationale:** *<Why is this requirement necessary>*

**User Requirement:** *<example - The system shall allow the user to enter user data>*

***System Requirements*:** *<examples - 1. The system shall accept date, time, name and address*

*2. The user shall be allowed to validate input and edit inputs before submission*

*3. The system shall allow saving partial data by the user before log-out>*

**Acceptance Criteria:** *<1. The user is able to save partial data having entered at least one completed field 100% of the time>*

**Relates to/Dependencies:** *<any requirement relates to / None>*

**Priority:** *<High/Low/Medium >*

**Team Owner:** *<Name of Team member responsible for specifying this requirement>*

*TO DO: use the above template to specify each feature of the system. Provide at least 6 functional requirements.*

*>*

## Behaviour Requirements

### Use Case View

*<A use case defines a goal-oriented set of interactions between external actors and the system under consideration.*

*TO DO: Provide a use case diagram which will encapsulate the entire system and all possible actors. Include use case narrative with: Name, Description, Precondition, Basic Flow and Post-condition>*

# Other Non-functional Requirements

## Performance Requirements

*<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.*

*TODO: Provide at least 3 different performance requirements based on the information you collected from the client. For example you can say “1. Any transaction will not take more than 10 seconds, etc…>*

## Safety and Security Requirements

*<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied. Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements.*

*TODO:*

* *Provide at least 2 different safety requirements based on your interview with the client*
* *Describe briefly what level of security is expected from this product by your client and provide a bulleted (or numbered) list of the major security requirements.>*

## Software Quality Attributes

*<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.*

*TODO: Make sure, that you do not just write “This software shall be maintainable…” Indicate how you plan to achieve it, & etc…Do not forget to include such attributes as the design for change. Please note that you need to include at least 2 quality attributes>*

# Other Requirements

*<This section is* ***Optional.*** *Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>*

**Appendix**

*<Please include here a description of what was done to elicit requirements, include dates of interview and who was interviewed, questionnaires, interview logs, observation activities, client responses etc.*

*Essentially, this section should provide proof of activities that resulted in the compilation of this document>*

*Failure to include supporting evidence here may be construed as evidence that requirements elicitation and analysis activities were not done!!*