Software Requirements Specification

for

<Augmented Reality Application to view bakery’s Items>

**Version 1.0 approved**

**Prepared by <Group No. 1>**

**<Section No. 4>**

**<14th october 2021>**

**Table of Contents**

**Table of Contents [ii](#_heading=h.gjdgxs)**

**Revision History [ii](#_heading=h.30j0zll)**

**1. Introduction [1](#_heading=h.1fob9te)**

1.1 Purpose [1](#_heading=h.3znysh7)

1.2 Document Conventions [1](#_heading=h.2et92p0)

1.3 Intended Audience and Reading Suggestions [1](#_heading=h.3dy6vkm)

1.4 Product Scope [1](#_heading=h.1t3h5sf)

1.5 References [1](#_heading=h.4d34og8)

**2. Overall Description [2](#_heading=h.2s8eyo1)**

2.1 Product Perspective [2](#_heading=h.17dp8vu)

2.2 Product Functions [2](#_heading=h.3rdcrjn)

2.3 User Classes and Characteristics [2](#_heading=h.26in1rg)

2.4 Operating Environment [2](#_heading=h.lnxbz9)

2.5 Design and Implementation Constraints [2](#_heading=h.35nkun2)

2.6 User Documentation [2](#_heading=h.1ksv4uv)

2.7 Assumptions and Dependencies [3](#_heading=h.44sinio)

**3. External Interface Requirements [3](#_heading=h.2jxsxqh)**

3.1 User Interfaces [3](#_heading=h.z337ya)

3.2 Hardware Interfaces [3](#_heading=h.3j2qqm3)

3.3 Software Interfaces [3](#_heading=h.1y810tw)

3.4 Communications Interfaces [3](#_heading=h.4i7ojhp)

**4. System Features [4](#_heading=h.2xcytpi)**

4.1 System Feature 1 [4](#_heading=h.1ci93xb)

4.2 System Feature 2 (and so on) [4](#_heading=h.3whwml4)

**5. Other Nonfunctional Requirements [4](#_heading=h.2bn6wsx)**

5.1 Performance Requirements [4](#_heading=h.qsh70q)

5.2 Safety Requirements [5](#_heading=h.3as4poj)

5.3 Security Requirements [5](#_heading=h.1pxezwc)

5.4 Software Quality Attributes [5](#_heading=h.49x2ik5)

5.5 Business Rules [5](#_heading=h.2p2csry)

**6. Other Requirements [5](#_heading=h.147n2zr)**

**Appendix A: Glossary [5](#_heading=h.3o7alnk)**

**Appendix B: Analysis Models [5](#_heading=h.23ckvvd)**

**Appendix C: To Be Determined List [6](#_heading=h.ihv636)**

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

The aim of the document is to describe the detailed functions of an Augmented Reality based Bakery Purchasing System. With this document, users, developers, testers, managers, clients and contract managers can know what services and functions this Augmented Reality based software shall provide and its operational and development constraints.

## Document Conventions

* Convention for main title

* Font face: Times
* Font style: Bold
* Font size: 18
* Convention for sub title

* Font face: TImes
* Font style: Bold
* Font size: 14
* Convention for body

* Font face: Arial
* Font style: Normal
* Font size: 11

## Intended Audience and Reading Suggestions

Customers:

* Select the quantity.
* Add details to the product
* At checkout, delivery time and total amount is displayed.

Owner of the bakery:

* Know the quantity of each product ordered by the customer.
* How many items of each product are left in the stock..

## Product Scope

We're working on an augmented reality based application for one of our projects. It will benefit clients who shop at bakeries or department stores owned by business owners. A user scans items in this application, and the details of the scanned item are printed on the screen. This app can assist bakery or store owners in reducing the number of employees working as cashiers and controlling inventory in their stores.

Augmented reality is a highly visual, interactive method of presenting relevant digital information in the context of the physical environment—connecting employees and improving business outcomes.

## 1.5 References

Unity 3D:

Unity is a cross-platform game engine with a built-in IDE developed by Unity Technologies. It is used to develop video games for web plugins, desktop platforms, consoles and mobile devices.Since our project requires us to have items in augmented reality we will be using unity.

Adobe Aero:

It optimizes a wide array of assets, including OBJ, GLB, and glTF files, for AR, so we can visualize them in real time.

Android Studio:

Since the requirement is to make a mobile application we will be using android studio.

Stack Overflow, Github, Youtube:

We will be using these sites to learn the new techniques and how to implement them.

As of right now we are still in the planning phase and still looking at different projects on github (so cant give a link on which projects are we gonna be referencing to)

Previous Projects:

We all have made projects in our OOP & DS courses so we will be reusing some of the code of the algorithms and methods we used in them

# Overall Description

## Product Perspective

This augmented reality based system aims at advancing the user experience for both the customers and the vendors. The application should show all the items in 3D which can be placed in mixed reality. Their costs, their designs, their quantity and their sizes. This will allow the customer to have a better experience while shopping and the vendors will have an in-depth understanding of their product. The application will also allow the customer to place an order.

The application will be able to run on the android operating system.

## Product Functions

**Functions for vendor:**

* See the items in 3D
* Add new item
* Update items such as their price, quantity
* See the details of every item such as their price, size, quality, quantity etc
* Accept or reject orders.

**Functions for customer:**

* See the items in 3D
* Browse through the shop
* See details of every item such as their price, size, quality
* Place order
* Cancel order (within a specific period of time after the order is placed)
* Give reviews

## User Classes and Characteristics

* *Users should be familiar with using the internet.*
* *User entering their details will be stored in database specified in system*
* *Users will have access to see their order, register themselves if they want to order something through our app.*
* *Through the image processing function of our app, users can scan the item they need, they will get details about the item.*

## Operating Environment

The software will operate on Android OS. Any hardware that can support Android version 7.0 or later will be able to run the software. The minimum SDK will be API 24 (Android (7.0) NOUGAT) and the software will be able to run on any device supporting android 7 or higher.

## Design and Implementation Constraints

This augmented reality is a virtual bakery shop where customers place the item in front of the camera. He can add details to it and then the item is added into the shopping cart. At checkout time, the items in the cart will be displayed as an order along with its price and time required to prepare it. The customer cannot change the time nor can bargain the price. The customer enters the details and chooses to pay either by cash or card.

## User Documentation

With the application we will be providing three documents

1. Description Document

Will give a detailed overview of the product with all the services offered by the product.

1. Installation and Setup

Will have detailed information on how to install and setup the software.

1. Product / User Manual

It gives the normal functions of the product with illustrated examples. It has all the ‘How-to’ information regarding the regular use of the product and some FAQs.

1. Demo Video

A video showing how the application works.

## Assumptions and Dependencies

*App will be used in an operating system that will hold enough space for all the documentation specified in the app. The app will require good quality internet connection and a strong operating system that will make the app run its functionalities.*

# External Interface Requirements

## User Interfaces

*<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>*

## 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, and communication protocols to be used.>*

## Software Interfaces

*<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, 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. Refer to documents that describe detailed application programming interface protocols. 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.>*

## 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.>*

# System Features

## Account Management

*Calculates the total amount payable by the customer.*

4.1.1 Description and Priority

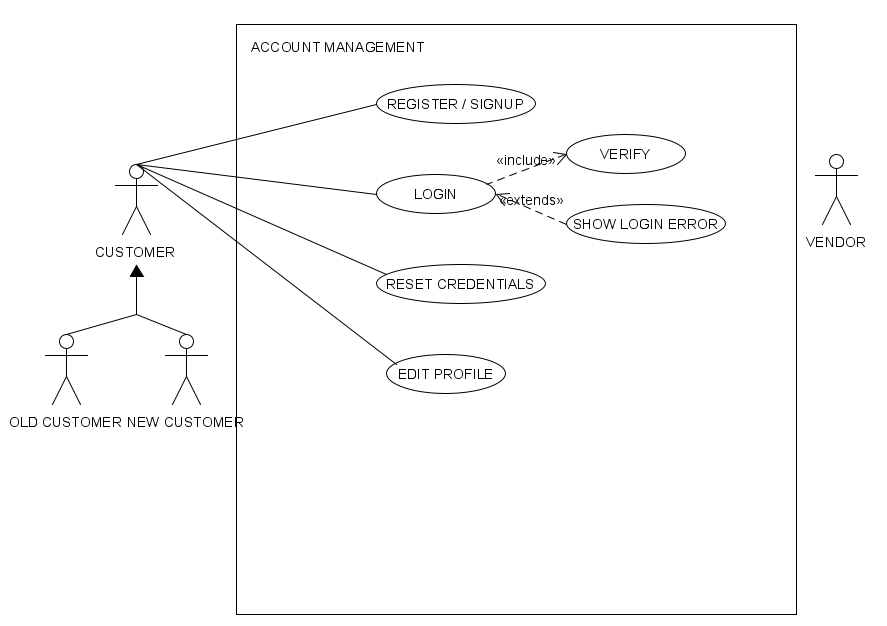
*It’s a high priority feature as all the financial records are managed with it. Any error in this feature can cause huge loss to the bakery. The main aim of the owner is to maximize its profit so this feature allows the owner to see the amount of each product sold and the profit earned on it. Also he can check the percentage of customers paying through cash or through bank.*

4.1.2 Stimulus/Response Sequences

*Customers:*

*Both old and new customers can:*

* *Register/ Sign up.*
* *Login: The system verifies it and if the customer enters wrong details, it gives an error.*
* *Reset Credentials*
* *Edit their profile*

**

4.1.3 Functional Requirements

*Register/ Sign up: If a new customer comes, he has to first signup/ register in the bakery management system by providing his required details.*

*Login: Both the old and new customers can enter the bakery system by providing their login details. The system verifies these details. If correct, it allows access else it pops an error notification “The login credentials are incorrect” and takes it back to the login page.*

*Reset credentials: If a customer wants to change its login credentials for safety purposes, he can change his password by providing the old and new password along with its username.*

*Edit Profile: If a user wants to edit his profile for example if he wants to change his address or bank card number, he can login to his account and make changes to his profile.*

*Some features are mandatory without which the customer’s account will not be registered. These are his username, password and phone number. Other features i.e. address and bank account details can later be added therefore, temporarily they will be recorded as TBD.*

## Inventory Management

*Inventory management feature allows the owner to keep track of inventory. He can check the number of each product sold and the number of items remaining in stock.*

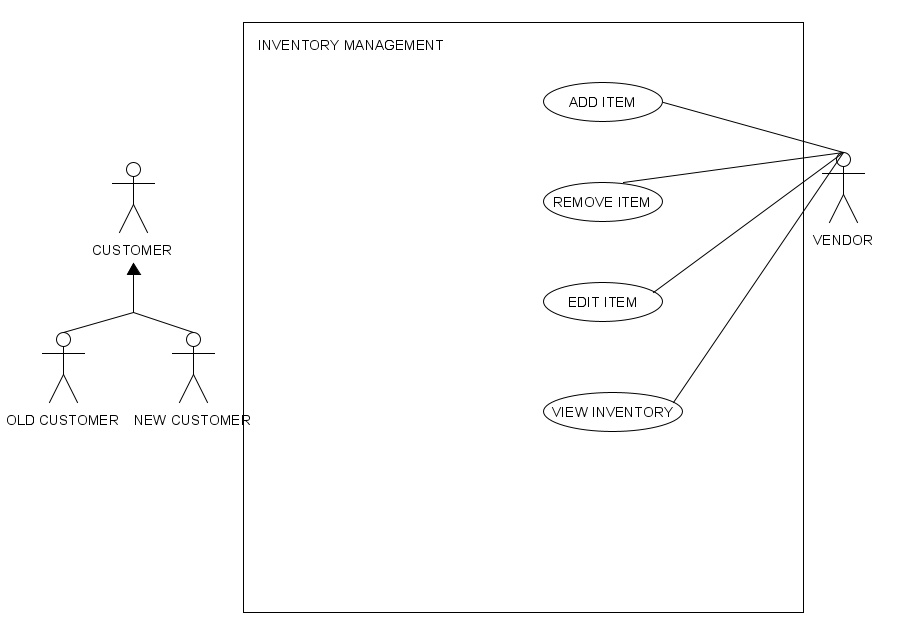
4.2.1 Description and Priority

*Inventory management is an important feature especially in the case of a bakery where the products are perishable. The owner will not want to say no to a customer if he wants to buy a certain product nor does he want his products to expire and be wasted. The aim of the owner will be to provide fresh products to its customer to increase the customer satisfaction.*

4.2.2 Stimulus/Response Sequences

*Inventory is managed by the bakery owner by using following options:*

* *Add item*
* *Remove item*
* *Edit item*
* *View inventory*

**

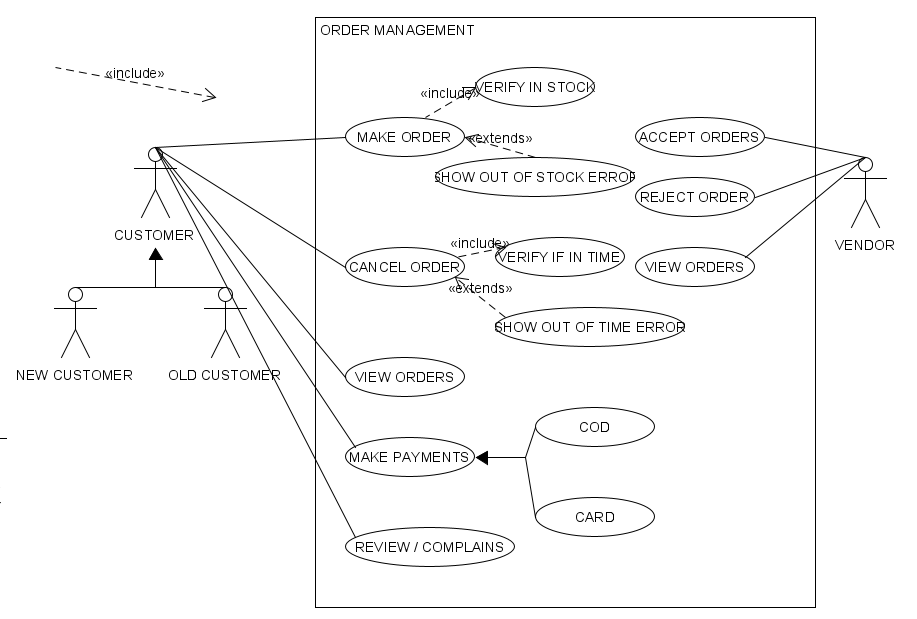
4.2.3 Functional Requirements

*Add item: The owner can introduce new products in stock. This option is only used when entering a certain product for the first time.*

*Remove item: If an item is not being bought by the customers and the owner wishes to stop selling that particular product, he can delete that product permanently from the inventory.*

*Edit: If the owner wants to increase or decrease the quantity of the existing product, he can use this option.*

*View: Enables the owner to check the quantity of each product in stock. This allows the owner to know which product is remaining less in stock so he can make more of that product and which product is not being sold.*

**

## Order Management

4.3.1 Description and Priority

*It’s a high priority feature as all the orders are managed with it. Any error in this feature can cause problems for both the customer and bakery owner. This is the most important feature as the customers can place their orders and the bakery owner has an option to accept or reject them. It will also allow the owner to see all the past orders and current orders.*

4.3.2 Stimulus/Response Sequences

*Customers:*

* *Make order*
* *Cancel order*
* *View order*
* *Make payment*
* *Review/ Complains*

*Bakery owner:*

* *Accept order*
* *Reject order*
* *View order*

4.3.3 Functional Requirements

*Customer:*

*Make Order: Customers can make orders thorugh it (An error will be given if item is out of stock)*

*Cancel Order: Customers can cancel order if it hasnt exceeded a certain time*

*View Orders: Customer can view all the past order they have made through that account*

*Make Payments: Customers have option of COD or by card*

*Review/Complains: An option to review their last order*

*Vendor:*

*Accept order: Option to accept a order if the vendor wants to*

*Reject Order: Option to reject the order if the vendor cant complete it*

*View Orders: Option to view order history.*

# Other Nonfunctional Requirements

## Performance Requirements

The system needs to be interactive, with minimal delays. As a result, there are no immediate delays in any of the system's actions.There is a delay of less than 2 seconds when opening windows forms, displaying error messages, or storing settings or sessions. There are no delays when opening databases, sorting questions, or evaluating them, and the operation takes less than 2 seconds for more than 95% of the files. The system should be simple to use. While using the app, the user should feel at ease. In every device where it is installed, the app should utilise minimal storage. If an interruption in internet access occurs while transmitting data to the server, the data can be sent again for verification.

## Safety Requirements

The fundamental document used by designers and developers to construct and manage apps is the safety requirement. The Safety Requirements must explicitly specify the Hazard that each of the Safety Instrumented Functions (SIF) is designed to prevent, as well as the functions that the SIFs must accomplish. The Software Requirement must set testing and verification requirements for the SIFs. The Software Requirements should specify which procedures are required, such as:

Procedures & Policies

Validation Phase 1

Procedures for Testing on a Regular Basis

Procedures for Bypassing

Records of Performance Data

Evaluations are conducted on a regular basis.

## Security Requirements

Users' accounts are the most vulnerable to hacking, hence adequate login mechanisms should be utilised to prevent this. The tablet id registration is a means to prevent spam and improve security. As a result, security is given against the unintended usage of recognition software.

The system provides the necessary tools for debate and problem solutions, it must be ensured that it is trustworthy in its operations and secures sensitive information.

## Software Quality Attributes

* The project is not open source
* The interface of the software will be neat and very user friendly for all users
* The software can be downloaded easily through google play store
* The software will be highly adaptive and regular updates will be provided

## Business Rules

## A business rule is anything that captures and implements business policies and practices. Each user should agree to terms and conditions specified by the admin of the app, neither admin should maliciously use user information. Business rules should provide promo code to customers on special occasions. Admin can make use of the database specified in the app in analysing business.

# Other Requirements

* A chat option where customer and the vendor can have a one on one conversation if there is a query.
* Promo codes / discount offers for customers if they shop above a certain price

**Appendix A: Glossary**

*Bakery owner*

*Customer*

*Inventory*

*Product*

*Item*

*Add*

*Remove*

*Edit*

*Login*

*Payment*

*Order*

*Complain*

*Accept*

*Reject*

*Verify*

*Cash on delivery*

*Card*

*Reset credentials*

*Error*

**Appendix B: Analysis Models**

*<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams*.>

**Appendix C: To Be Determined List**

* *Product will not use its functionalities in a way that will produce disastrous outputs or threaten users identity.*
* *Product should be displaying error messages within a delay of 30secs.*
* *Product will use a database that will maintain integrity of data for the admin and protect data from malicious users.*
* *Product should be designed in a way that will be easily functional in many different operating systems.*
* *If any advancement is made in the field of augmented reality product will be updated too.*