

**SOFTWARE REQUIREMENT SPECIFICATION**

**TOPIC**

**FOOD DELIVERY WEBSITE**

**PREPARED BY**

NASHRINBANU N 211520205095

NIVEDHA M 211520205099

RABIYATHUL 211520205108

HASNA M.I.K

**TABLE OF CONTENTS**

**1. Introduction**

1.1 Purpose

1.2 Document Conventions

1.3 Intended Audience and Reading Suggestions

1.4 Project Scope

**2. Overall Description**

2.1 Product Perspective

2.2 Product Features

2.3 User Classes and Characteristics

2.4 Operating Environment

2.5 Design and Implementation Constraints

2.6 User Documentation

2.7 Assumptions and Dependencies

**3. System Features**

3.1 System Feature 1

3.2 System Feature 2

**4. External Interface Requirements**

4.1 User Interfaces

4.2 Hardware Interfaces

4.3 Software Interfaces

4.4 Communications Interfaces

**5. Other NonFunctional Requirements**

5.1 Performance Requirements

5.2 Safety Requirements

5.3 Security Requirements

5.4 Software Quality Attributes

**6.Models**

**1.Introduction**

**1.1 Purpose**

The purpose of this SRS is to outline both the functional and non-functional requirements of the subject Restaurant food ordering system. In addition to said requirements, the document also provides a detailed profile of the external interfaces, performance considerations and design constraints imposed on the subsequent implementation. The document should act as a foundation for efficient and well-managed project completion and further serve as an accurate reference in the future.

**1.2 Document Conventions**

RFOS- restaurant Food Ordering System

SRS - Software Requirements Specification

DBMS - Database Management System

LAN - Local Area Network

IP - Internet Protocol

TCP - Transmission Control Protocol

UDP - User Datagram Protocol

IEEE 802.11 Wireless Local Area Network Standard

**1.3 Intended Audience and Reading Suggestions**

The primary audience of this SRS document will be the development team employed to implement the specified Restaurant food ordering system. It will not only provide an extensive capacity for project planning and progress assessment but it will further assist with stakeholder interactions. The secondary document audience comprises the stakeholders of the project, that is, restaurateurs and associated staff. To this audience group, this SRS should convey and confirm the required functionality and represent a contractual agreement between the involved parties.

**1.4 Project Scope**

In current formal dining environments, some form of physical static menu is utilised to convey the available food and beverage choices to customers. Said menus are generally paper based and hence impose restrictions on the textual real estate available and the ability a restaurateur has to update them. The related concepts are encompassed by the general scope of the Restaurant food Ordering System. It is to the replacement of paper-based menus using an electronic format.

**2. Overall Description**

**2.1 Product Perspective**

The software described in this SRS is the software for a complete Restaurant food ordering system. The system merges various hardware and software elements and further interfaces with external systems. it relies on a number of external interfaces for persistence and unhandled tasks, as well as physically interfacing with humans.

**2.2 Product Features**

The Restaurant food ordering system interfaces with an existing payment system, including a cash register and software accessible credit system, in order to quickly and easily handle customer billing. The payment system should be operable such that it can return information to the RFOS system as to whether payment was successful or failed.

**2.3 User Classes and Characteristics**

There are three separate user interfaces used by the RFOS software, each related to an interfaced physical hardware device. These three user interfaces are the Surface Computer UI, Tablet UI and Display UI

**2.4 Operating Environment**

The Surface Computer UI is the interface used by restaurant customers. This interface uses the surface computer paradigm - users interact with the system by dragging 'objects' around on the ﬂatscreen touch-sensitive display. The Tablet UI is designed to run on a small, wireless-enabled touch-screen tablet PC, to be used by waiters to accommodate customer needs. The Display UI provides kitchen staﬀ with simple functionality related to ordered items.

**2.5 Design and Implementation Constraints**

The RFOS should be written in an object-oriented language with strong GUI links and a simple, accessible network API. The primary candidate tool chains are Java/Swing, C++/Qt and Python/Qt. The system must provide a capacity for parallel operation and system design should not introduce scalability issues with regard to the number of surface computers, tablets or displays connected at any one time. The system must be reliable enough to run crash and glitch free more or less indeﬁnitely, or facilitate error recovery strong enough such that glitches are never revealed to its end-users.

**2.6 User Documentation**

The end-users of the RFOS fall into three primary categories, unskilled, partly skilled and highly skilled.

**2.7 Assumptions and Dependencies**

The SRS assumes that none of the constituent system components will be implemented as embedded applications. It is further assumed that tablet PCs of sufficient processing capability and battery life will be utilised.

**3. System Features**

Functional requirements are listed first, according to their relationship to the overall system, customers, waiters, chefs and supervisors.

**Customer**

A customer shall be able to engage their menu by double tapping the activated surface computer in their table.

A customer shall be able to add an item to a pending order by dragging the item from the engaged menu onto the order.

A customer shall be able to remove an item from a pending order by dragging the item oﬀ the order.

A customer shall be able to add a special dietary requirement to an order by dragging the requirement from the engaged menu onto the order.

When in billing mode, a surface computer shall display a representation of a bankcard payment foreach customer.

**Waiter**

A waiter assigned to a table shall be alerted via their wireless tablet when: An order is placed from that table. An item ordered by that table is rejected by the kitchen. An item ordered by that table is ready to be served. The table has requested waiter assistance A tablet shall allow a waiter to accept ,reject and modify an order placed by a customer through a surface computer. A tablet shall allow a waiter to process a payment using cash or a bankcard.

**Chef**

A chef shall be able to accept or reject a customer’s order item through a display.

A chef shall be able to indicate that a customer’s order item is ready to be served through a display.

**Supervisor**

A supervisor shall be able to abort/purge a table's account/meals from the active system with no expectation of payment.

A supervisor shall be able to issue a refund for one or more items to a customer.

**4. External Interface Requirements**

**4.1 User Interfaces**

This interface uses the surface computer paradigm - users interact with the system by dragging 'objects' around on the ﬂatscreen touch-sensitive display. For the RFOS, users can manipulate objects such as items of food, dietary requirements, tips and menus on the surface of their table. Such objects can be moved into static objects such as meals and payments to perform various functions.

**4.2 Hardware Interfaces**

These devices are the surface computers, the wireless tablets and the touch displays. All three devices must be physically robust and immune to liquid damage and stains. The devices (with the possible exception of displays) must also have good industrial design aesthetics, as they are to be used in place of normal restaurant tables and notepads and will be in direct contact with customers.

**4.3 Software Interfaces**

The RFOS will interface with a Database Management System (DBMS) that stores the information necessary for the RMOS to operate. The DBMS must be able to provide, on request and with low latency, data concerning the restaurant's menu, employees (and their passwords) and available dietary requirements.

**4.4 Communication Interfaces**

The RFOS will interface with a Local Area Network (LAN) to maintain communication with all its devices. It should use a reliable-type IP protocol such as TCP/IP or reliable-UDP/IP for maximum compatibility and stability. All devices it will interface with should contain standard Ethernet compatible, software accessible LAN cards to maintain communication between the server and the surface computers, tablets, displays and the external payment system

**5. Other Non-Functional Requirements**

This subsection presents the identiﬁed non-functional requirements for the subject RFOS. The subcategories of non-functional requirements given are performance, safety, security requirements.

**5.1 Performance Requirements**

The server shall be capable of supporting an arbitrary number of surface computers, tablets and displays, that is, it shall provide no limit on how many devices are in the system. The server shall be capable of supporting an arbitrary number of active customer payments, that is, no payments shall be lost under any circumstances.

**5.2 Safety Requirements**

The system shall log every state and state change of every surface computer, tablet and display to provision recovery from system failure.

The system shall be capable of restoring itself to its previous state in the event of failure (e.g. a system crash or power loss).

The system shall be able to display a menu at all times to facilitate manual order taking should the need arise.

The system shall utilise periodic 30-second keep-alive messages between tablets and the server to monitor tablet operational status.

**5.3 Security Requirements**

A waiter password used for tablet login must have a bit-strength of at least 64 bits.

A waiter password used for tablet login must be changed every three months.

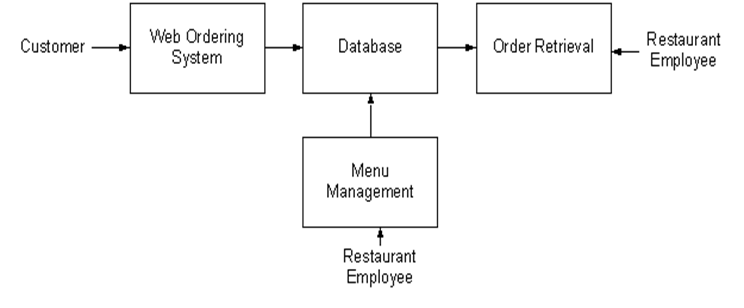
A waiter shall only be able to log into one tablet at any given instance of time.

A waiter that attempts to log into a second tablet while already logged into

**5.4 Software Quality Attributes**

The software shall be capable of supporting an arbitrary number of surface computers, tablets and displays, that is, it shall provide no limit on how many devices are in the system. It shall be capable of supporting an arbitrary number of active meals/orders, that is, no meals/orders shall be lost under any circumstances.

**6.Model**



The purpose of this SRS is to outline both the functional and non-functional requirements of

the subject Restaurant food ordering system. In addition to said requirements, the document also

provides a detailed profile of the external interfaces, performance considerations and design

constraints imposed on the subsequent implementation. The document should act as a foundation

for efficient and well-managed project completion and further serve as an accurate reference in the

future.

The purpose of this SRS is to outline both the functional and non-functional requirements of

the subject Restaurant food ordering system. In addition to said requirements, the document also

provides a detailed profile of the external interfaces, performance considerations and design

constraints imposed on the subsequent implementation. The document should act as a foundation

for efficient and well-managed project completion and further serve as an accurate reference in the

future.