# Software Requirements Specification

## for

# Pizza Ordering System, Release 1.0

**\*As Per IEEE-830 standards**

**Version 1.0 approved**

**Prepared by Vrishank Gupta, Shivam Gupta**

**April 2, 2018**

# Table of Contents

# 1.Introduction

## 1.1.Purpose

This SRS describes the software functional and nonfunctional requirements for release 1.0 of the Pizza Ordering System (POS). This document is intended to be used by the members of the project team that will implement and verify the correct functioning of the system. Unless otherwise noted, all requirements specified here are high priority and committed for release 1.0.

## 1.2.Project Scope and Product Features

The Pizza Ordering System will permit Bitsplease employees to order meals from the company pizzeria on-line to be delivered to specified campus locations. A detailed project description is available in the *Pizzeria Ordering System Vision and Scope Document* [1]. The section in that document titled "Scope of Initial and Subsequent Releases" lists the features that are scheduled for full or partial implementation in this release.

## 1.3.References

1.  Vrishank, Shivam. *Pizzeria Ordering System Vision and Scope Document*, www.bitsplease.com/ projects/POS/POS_vision_and_scope.doc
2.  Vrishank, Shivam. *Bitsplease Intranet Development Standard, Version 1.3*, www.bitsplease.com/ corporate/standards/PI_intranet_dev_std.doc
3.  Vrishank, Shivam. *Bitsplease Business Rules Catalog*, www.bitsplease.com/corporate/policies/ PI_business_rules.doc
4.  Vrishank, Shivam. *Bitsplease Internet Application User Interface Standard, Version 2.0*, www.bitsplease.com/corporate/standards/PI_internet_ui_std.doc
5.  *IEEE 830* SRS Quality Assurance Specification, https://ieeexplore.ieee.org/document/7321195/.
6.  Ranjana Rajnish et. al. / Indian Journal of Computer Science and Engineering, WRITING QUALITY REQUIREMENTS (SRS): AN APPROACH TO MANAGE REQUIREMENTS VOLATILITY, http://www.ijcse.com/docs/IJCSE10-01-01-06.pdf

# 2.Overall Description

## 2.1.Product Perspective

The Pizza Ordering System is a new system that replaces the current manual and telephone processes for ordering and picking up lunches in the Bitsplease pizzeria.The system is expected to evolve over several releases, ultimately connecting to the Internet ordering services for several local restaurants and to credit and debit card authorisation services.

## 2.2.User Classes and Characteristics

Patron (favored)    A Patron is a Bitsplease employee at the corporate campus in Delhi, India, who wishes to order meals to be delivered from the company pizzeria. There are about 600 potential Patrons, of which an estimated 400 are expected to use the Pizza Ordering System an average of 4 times per week each (source: current pizzeria usage data). Patrons will sometimes order multiple meals for group events or guests. An estimated 90 percent of orders will be placed using the corporate Intranet, with 10 percent of orders being placed from home. All Patrons have Intranet access from their offices. Some Patrons will wish to set up meal subscriptions, either to have the same meal to be delivered every day or to have the day's meal special delivered automatically. A Patron must be able to override a subscription for a specific day.

Pizzeria Staff      The Bitsplease pizzeria currently employs about 20 Pizzeria Staff, who will receive orders from the Pizza Ordering System, prepare meals, package them for delivery, print delivery instructions, and request delivery. Most of the Pizzeria Staff will need to be trained in the use of the computer, the Web browser, and the Pizza Ordering System.

Menu Manager        The Menu Manager is a pizzeria employee, perhaps the pizzeria manager, who is responsible for establishing and maintaining daily menus of the food items available from the pizzeria and the times of day that each item is available. Some menu items may not be available for delivery. The Menu Manager will also define the pizzeria's daily specials. The Menu Manager will need to edit the menus periodically to reflect planned food items that are not available or price changes.

Meal Deliverer      As the Pizzeria Staff prepare orders for delivery, they will print delivery instructions and issue delivery requests to the Meal Deliverer, who is either another pizzeria employee or a contractor. The Meal Deliverer will pick up the food and delivery instructions for each meal and deliver it to the Patron. The Meal Deliverers' primary interactions with the system will be to reprint the delivery instructions on occasion and to confirm that a meal was (or was not) delivered.

## 2.3. Operating Environment

OE-1:    The Pizza Ordering System shall operate with the following Web browsers: Microsoft Internet Explorer ,Google Chrome and Apple Safari, including their mobile versions too.

OE-2:    The Pizza Ordering System shall operate on a server running the current corporate approved versions of Red Hat Linux and Apache WebServer.

OE-3:    The Pizza Ordering System shall permit user access from the corporate Intranet and, if a user is authorised for outside access through the corporate firewall, from an Internet connection at the user's home.

OE-4:    The Pizza Ordering System shall operate with all the modern Android Devices with a dedicated Android App and on IOS Devices with a dedicated IOS App.

## 2.4. Design and Implementation Constraints

CO-1:    The system's design, code, and maintenance documentation shall conform to the *Bitsplease Intranet Development Standard, Version 1.3* [2].

CO-2: The system shall use the current corporate standard Oracle database engine.
CO-3: All HTML code shall conform to the HTML 4.0 standard.
CO-4: All scripts shall be written in Perl.
CO-5: All code for Android will be written in **Kotlin**(including data classes) with the usage of Anko Library to minimise the usage of XML code in the app, Retrofit Library to fetch data from Web Client API.
CO-6: All IOS code will be strictly written in Swift.

## 2.5.User Documentation

UD-1: The system shall provide an online hierarchical and cross-linked help system in HTML/Kotlin that describes and illustrates all system functions.
UD-2: The first time a new user accesses the system and on user demand thereafter, the system shall provide an online tutorial to allow users to practice ordering meals using a static tutorial menu. The system shall not store meals ordered using this template in the database or place orders for such meals with the pizzeria.

## 2.6.Assumptions and Dependencies

AS-1: The pizzeria is open for breakfast, lunch, and dinner every company business day in which employees are expected to be on site.
DE-1: The operation of the POS depends on changes being made in the Payroll System to accept payment requests for meals ordered with the POS.
DE-2: The operation of the POS depends on changes being made in the Pizzeria Inventory System to update the availability of food items as POS orders are accepted.

# 3.System Features

## 3.1.Order Meals

### 3.1.1 Description and Priority

A pizzeria Patron whose identity has been verified may order meals either to be delivered to a specified company location or to be picked up in the pizzeria. A Patron may cancel or change a meal order if it has not yet been prepared. **Priority = High**.

### 3.1.2 Stimulus/Response Sequences

Stimulus: Patron requests to place an order for one or more meals.
Response: System queries Patron for details of meal(s), payment, and delivery instructions.
Stimulus: Patron requests to change a meal order.
Response: If status is "Accepted," system allows user to edit a previous meal order.
Stimulus: Patron requests to cancel a meal order.
Response: If status is "Accepted, "system cancels a meal order.

### 3.1.3 Functional Requirements

| | |
|---|---|
| Order.Place: | The system shall let a Patron who is logged into the Pizza Ordering System place an order for one or more meals. |
| Order.Place.Register: | The system shall confirm that the Patron is registered for payroll deduction to place an order. |
| Order.Place.Register.No | If the Patron is not registered for payroll deduction, the system shall give the Patron options to register now and continue placing an order, to place an order for pickup in the pizzeria (not for delivery), or to exit from the POS. |
| Order.Place.Date: | The system shall prompt the Patron for the meal date. |
| Order.Place.Date.Cutoff: | If the meal date is the current date and the current time is after the order cutoff time, the system shall inform the patron that it's too late to place an order for today. The Patron may either change the meal date or cancel the order. |
| Order.Deliver.Select: | The Patron shall specify whether the order is to be picked up or delivered. |
| Order.Deliver.Location: | If the order is to be delivered and there are still available delivery times for the meal date, the Patron shall provide a valid delivery location. |
| Order.Deliver.Notimes: | The system shall notify the Patron if there are no available delivery times for the meal date. The Patron shall either cancel the order or indicate that the Patron will pick up the order in the pizzeria. |
| Order.Deliver.Times: | The system shall display the remaining available delivery times for the meal date. The system shall allow the Patron to request one of the delivery times shown, to change the order to be picked up in the pizzeria, or to cancel the order. |
| Order.Menu.Date: | The system shall display a menu for the specified date. |
| Order.Menu.Available: | The menu for the current date shall display only those food items for which at least one unit is available in the pizzeria's inventory. |
| Order.Units.Food: | The system shall allow the Patron to indicate the number of units of each menu item that he wishes to order. |
| Order.Units.Multiple: | The system shall permit the user to order multiple identical meals, up to the fewest available units of any menu item in the order. |
| Order.Units.TooMany: | If the Patron orders more units of a menu item than are presently in the pizzeria's inventory, the system shall inform the Patron of the maximum number of units of that food item that he can order. |
| Order.Units.Change: | If the available inventory cannot fulfil the number of units ordered, the Patron may change the number of units ordered, change the number of identical meals being ordered, or cancel the meal order. |

| | |
|---|---|
| Order.Confirm.Display: | When the Patron indicates that he does not wish to order any more food items, the system shall display the food items ordered, the individual food item prices, and the payment amount. |
| Order.Confirm.Prompt: | The system shall prompt the Patron to confirm the meal order. |
| Order.Confirm.Not: | If the Patron does not confirm the meal order, the Patron may either edit or cancel the order. |
| Order.Confirm.More: | The system shall let the Patron order additional meals for the same or for different date. |
| Order.Pay.Method: | When the Patron indicates that he is done placing orders, the system shall ask the user to select a payment method. |
| Order.Pay.Deliver: | Patron can choose the pizza to be delivered at the required address. |
| Order.Pay.Pickup: | If the meal is to be picked up in the pizzeria, the system shall let the Patron choose to pay by payroll deduction or by paying cash at the time of pickup. |
| Order.Pay.Details: | The system shall display the food items ordered, payment amount, payment method, and delivery instructions. |
| Order.Pay.Confirm: | The Patron shall either confirm the order, request to edit the order, or request to cancel the order. |
| Order.Pay.Confirm.Deduct: | If the Patron confirmed the order and selected payment by payroll deduction, the system shall issue a payment request to the Payroll System. |
| Order.Pay.Confirm.OK: | If the payment request is accepted, the system shall display a message confirming acceptance of the order with the payroll deduction transaction number. |
| Order.Pay.Confirm.NG: | If the payment request is rejected, the system shall display a message with the reason for the rejection. The Patron shall either cancel the order, or change the payment method to cash and request to pick up the order at the pizzeria. |

| | |
|---|---|
| Order.Done: | When the Patron has confirmed the order, the system shall do the following as a single transaction: |
| Order.Done.Store | Assign the next available meal order number to the meal and store the meal order with an initial status of "Accepted." |
| Order.Done.Inventory: | Send a message to the Pizzeria Inventory System with the number of units of each food item in the order. |
| Order.Done.Menu: | Update the menu for the current order's order date to reflect any items that are now out of stock in the pizzeria inventory. |
| Order.Done.Times: | Update the remaining available delivery times for the date of this order. |
| Order.Done.Patron: | Send an e-mail message to the Patron with the meal order and meal payment information. |
| Order.Done.Pizzeria: | Send an e-mail message to the Pizzeria Staff with the meal order information. |
| Order.Done.Failure: | If any step of Order.Done fails, the system shall roll back the transaction and notify the user that the order was unsuccessful, along with the reason for failure. |
| Order.Previous.Period: | The system shall permit the Patron to view any meals he has ordered within the previous six months. [Priority = Medium] |
| Order.Previous.Reorder: | The Patron may reorder any meal he had ordered within the previous six months, provided that all food items in that order are available on the menu for the meal date. [Priority = Medium] |
| *[functional requirements for changing and canceling meal orders are not provided in this example]* | |

# 4.External Interface Requirements

## 4.1.User Interfaces

UI-1:    The Pizza Ordering System screen displays shall conform to the *Bitsplease Internet Application User Interface Standard, Version 2.0* [4].

UI-2:    The system shall provide a help link from each displayed HTML page to explain how to use that page.

UI-3:    The Web pages shall permit complete navigation and food item selection using the keyboard alone, in addition to using mouse and keyboard combinations.

## 4.2.Hardware Interfaces

No hardware interfaces have been identified.

## 4.3.Software Interfaces

SI-1:    Pizzeria Inventory System

SI-1.1:    The POS shall transmit the quantities of food items ordered to the Pizzeria Inventory System through a programmatic interface.

SI-1.2:   The POS shall poll the Pizzeria Inventory System to determine whether a requested food item is available.

SI-1.3:   When the Pizzeria Inventory System notifies the POS that a specific food item is no longer available, the POS shall remove that food item from the menu for the current date.

SI-2:   Payroll System
The POS shall communicate with the Payroll System through a programmatic interface for the following operations:

SI-2.1:   To allow a Patron to register for payroll deduction.

SI-2.2:   To allow a Patron to unregister for payroll deduction.

SI-2.3:   To check whether a patron is registered for payroll deduction.

SI-2.4:   To submit a payment request for a purchased meal.

SI-2.5:   To reverse all or part of a previous charge because a patron rejected a meal or wasn't satisfied with it, or because the meal was not delivered per the confirmed delivery instructions.

## 4.4. Communications Interfaces

CI-1:   The Pizza Ordering System shall send an e-mail message to the Patron to confirm acceptance of an order, price, and delivery instructions.

CI-2:   The Pizza Ordering System shall send an e-mail message to the Patron to report any problems with the meal order or delivery after the order is accepted.

# 5. Other Nonfunctional Requirements

## 5.1. Performance Requirements

PE-1:   The system shall accommodate 400 users during the peak usage time window of 8:00am to 10:00am local time, with an estimated average session duration of 8 minutes.

PE-2:   All Web pages and server data via API for mobile Apps generated by the system shall be fully downloadable in no more than 10 seconds over a 40KBps modem connection.

PE-3:   Responses to queries shall take no longer than 7 seconds to load onto the screen after the user submits the query.

PE-4:   The system shall display confirmation messages to users within 4 seconds after the user submits information to the system.

## 5.2. Safety Requirements

No safety requirements have been identified.

## 5.3. Security Requirements

SE-1:   All network transactions that involve financial information or personally identifiable information shall be encrypted.

SE-2:   Users shall be required to log in to the Pizza Ordering System for all operations except viewing a menu.

SE-3:   Patrons shall log in according to the restricted computer system access policy.

SE-4:   The system shall permit only pizzeria staff members who are on the list of authorised Menu Managers to create or edit menus.

SE-5: Only users who have been authorised for home access to the corporate Intranet may use the POS from non-company locations.

SE-6: The system shall permit Patrons to view only their own previously placed orders, not orders placed by other Patrons.

## 5.4.Software Quality Attributes

Availability-1: The Pizza Ordering System shall be available to users on the corporate Intranet and to dial-in users 99.9% of the time between 5:00am and midnight local time and 95% of the time between midnight and 5:00am local time.

Robustness-1: If the connection between the user and the system is broken prior to an order being either confirmed or canceled, the Pizza Ordering System shall enable the user to recover an incomplete order.

# Appendix A: Data Dictionary and Data Model

delivery instruction    =   patron name  
                               +   patron phone number  
                               +   meal date  
                               +   delivery location  
                               +   delivery time window

delivery location        =   * building and room to which an ordered meal is to be delivered *

delivery time          =   * 15-minute range during which an ordered meal is to be delivered;  
window                         must begin and end on quarter-hour intervals *

employee ID          =   * company ID number of the employee who placed a meal order; 6-  
                                 character numeric string *

food item description  =   * text description of a food item on a menu; maximum 100 characters *

food item price       =   * pre-tax cost of a single unit of a menu food item, in Rupees*

meal date            =   * the date the meal is to be delivered or picked up; format MM/DD/  
                               YYYY; default = current date if the current time is before the order  
                               cutoff time, else the next day; may not be prior to the current date *

meal order          =   meal order number  
                               +   order date  
                               +   meal date  
                               +   1:m{ordered food item}  
                               +   delivery instruction  
                               +   meal order status

meal order number   =   * a unique, sequential integer that the system assigns to each accepted  
                               meal order; initial value is 1 *

meal order status     =   [ incomplete | accepted | prepared | pending delivery | delivered |  
                               canceled ]

meal payment          =   payment amount
                      +   payment method
                      +   (payroll deduction transaction number)


menu                  =   menu date
                      +   1:m{menu food item}
                      +   0:1{special}


menu date             =   * the date for which a specific menu of food items is available; format
                          MM/DD/YYYY *


menu food item        =   food item description
                      +   food item price


order cutoff time     =   * the time of day before which all orders for that date must be placed *


order date            =   * the date on which a patron placed a meal order; format MM/DD/
                          YYYY *


ordered food item     =   menu food item
                      +   quantity ordered


patron                =   patron name
                      +   employee ID
                      +   patron phone number
                      +   patron location
                      +   patron e-mail

patron e-mail         =   * e-mail address of the employee who placed a meal order; 50
                          character alphanumeric *

patron location       =   * building and room numbers of the employee who placed a meal
                          order; 50 character alphanumeric *

patron name           =   * name of the employee who placed a meal order; 30 character
                          alphanumeric *

patron phone number   =   * telephone number of the employee who placed a meal order; format
                          AAA-EEE-NNNN xXXXX for area code, exchange, number, and
                          extension *

payment amount          =    * total price of an order in Rupees*

payment method          =    [ payroll deduction | cash ] * others to be added beginning with release 2 *

payroll deduction       =    * 8-digit sequential integer number that the Payroll System assigns to
transaction number           each payroll deduction transaction that it accepts *

quantity ordered        =    * the number of units of each food item that the Patron is ordering;
                             default = 1; maximum = quantity presently in inventory *

special                 =    special description
                        +    special price
                             * the Menu Manager may define one or more special meals for each
                             menu, with a particular combination of food items at a reduced price *

special description     =    * text description of a daily special meal; maximum 100 characters *

special price           =    * cost of a single unit of a daily special meal, in Rupees *

# Appendix B: Analysis Models