PES UNIVERSITY, BANGALORE
Department of Computer Science and Engineering

Software Requirements Specification

Online Food Delivery System

Version 1.1 approved

Prepared by Ram Selvaraj | PES2UG20CS265

Riya Hurtis | PES2UG20CS277

Sagarikha M | PES2UG20CS292

S Kameel | PES2UG20CS320

PES UNIVERSITY, BANGALORE

Date:07-09-2022

Revision History

Name	Date	Reason For Changes	Version
Rev1	7/09/2022	Minor updates and changes	1.1

Table of Contents

Introduction

Purpose

The software is supposed to enable the user to order food online and have it delivered to their house.

Intended Audience

Users above the age of 13, those who do not have time to cook food at home or visit

Product Scope

The online food delivery platform enables users to choose from a wide variety of cuisines. It has an Admin Dash which provides information about the pending orders, cancelled orders, on-delivery orders, update orders etc. It also has a User Dash which shows details about present and past orders, re-order etc. It also shows recommended food items

References

Swiggy

Zomato

Uber Eats

Amazon Food

Overall Description

Product Functions

- The software provides checkboxes and textboxes to collect data from the user
- The interface is easy to use
- Keeps track of current and past orders
- Generates the total cost including tax and delivery fee based on distance and weather conditions

User Classes and Characteristics

Admin Class – Administrators of the website, that have access to the backend of the website, can view and edit the food items available to purchase. They can also view current and past orders of all users.

User Class - Users who wish to order food online and have it delivered to their place of residence. They can view their own current and past orders.

Operating Environment

The software will operate on an online browser based platform. It is cross-platform available on PC and Mobile. The phone app has the UPI API hook which enables easy payments

Design and Implementation

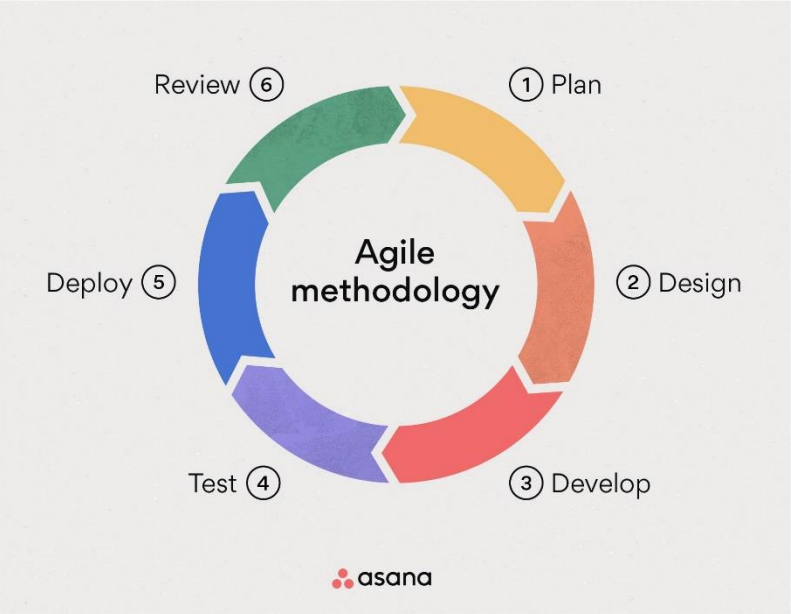
Implementation of payment systems relies on integration with the UPI payment service, which is a major dependency. The software is only in English as of now although a large portion of userbase might only speak local languages. We will be using a database that consists of all the Food items, user ids etc.

ConstraintsDesign

When planning a project, it is important to use a systems development life cycle (SDLC) to successfully take a project through the various stages of development there are twomain SDLC methodologies to plane and design the project based on the available of resources and other constrains namely; waterfall and agile.

Agile Methodology:

The agile methodology is used when the requirements may change during the development of a project. Projects using agile development are broken up into sprints, each with their own analysis, design, implementation and testing phases. These sprints begin with a meeting, detailing the work to be done in the sprint with daily meetings to keep the project on target. Essentially, this breaks the project into smaller segments witheach segment having its own development phase, then at the end, the segments are combined to form the finished product. Figure 2.2 shows the agile method as a series of sprints with their own development life cycles.

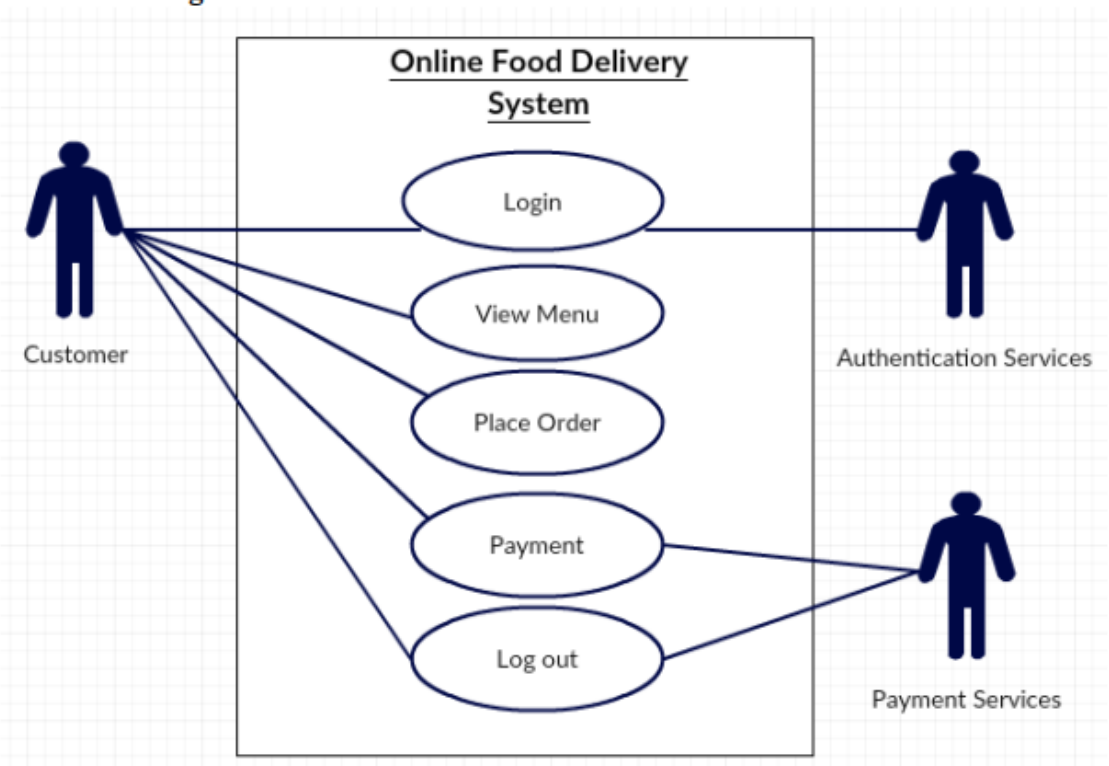


For this project, the agile methodology will be utilised. As there is project team, meetings on each sprint is being held, and the method of breaking down the project into smaller, more manageable segments will be used. This will allow the project to be able to adapt toany changes in the strategies of work. As this project consists of a web crawler and a webapplication, both the agile methodology and waterfall method are better suited to handlingmultiple deliverables than the single method.

System Architecture

There are 3 layers in this program, online web ordering layer, menu management layer and order retrieval system layer. These 3 work in tandem and simultaneously to make the process of ordering food easier.

User-case Diagram



Use Case Diagrams

Implementation Constraints

Moral regulations along with sensitive data to be protected are the only constraints which we will encounter during this time

Assumptions and Dependencies

Assumptions include that the User is literate in English and has basic technological acumen. We also assume that the UPI payments service is up and running at all times so the User never experiences failed transactions. All assets and components will be well documented with troubleshooting steps

NAME	DEPENDANCY
XAMPP	MySQL, HTML, PHP
Payment API	Python

External Interface Requirements

User Interfaces

We are using HTML,CSS and PHP to design the website. It is a fully interactive, intuitive and multi platform app. We also add some PHP animations to make the UI more appealing to the user.

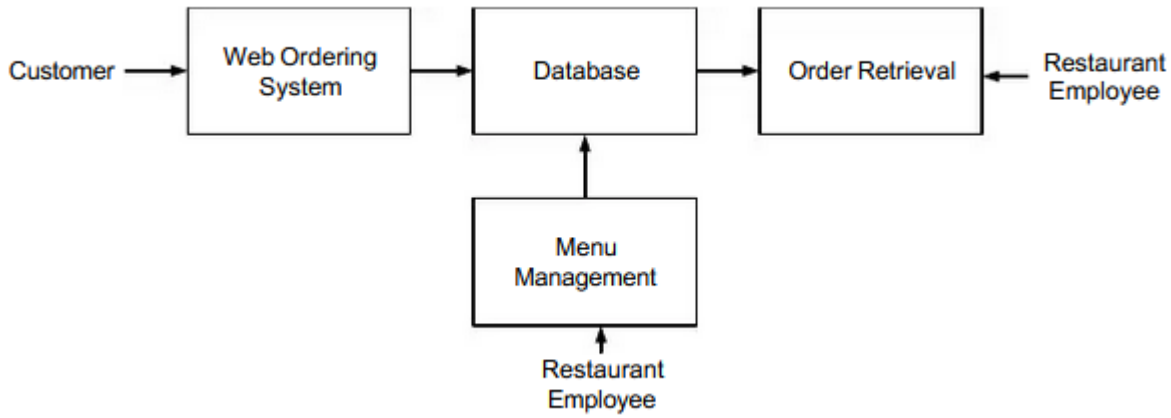
Software Interfaces

- Windows 10/Linux/ubuntu
- Visual studio code/Notepad++
- Browsers i.e., chrome, Bing
- Python 3.8 and above
- HTML, CSS, JS

Communications Interfaces

When the user adds items, we temporarily store the items in a cart, then we calculate the total bill after retrieving the prices from our Food database, and once the user confirms the purchase, add the bill number and details to our Orders database. The user can then view his current and past orders from the dashboard

Analysis Models



System Features

The online food delivery platform enables users to choose from a wide variety of international cuisines. It has an Admin Dash which provides information about pending orders, cancelled orders, on-delivery orders etc. It also has a User Dash which shows details about present and past orders

A Easy to use UI.

7.1.1 Description and Priority

We provide High level of priority because we are the main connection point between the restaurant and end user. We handle the payments on the user's end and payment to the restaurants This is very convenient for people who don't have quick access to food.

7.1.2 Stimulus/Response Sequences

When the user opens the website, it will list the food items available for purchase, along with pictures and price details, the user can then select and add the desired food items to his/her cart and confirm the purchase. He/She can then view the current order and past orders from the dashboard

7.1.3 Functional Requirements

Web Ordering System –

- Create an account.
- Manage their account.
- Log in to the system.
- Navigate the restaurant's menu.
- Select an item from the menu.
- Customize options for a selected item.
- Add an item to their current order.
- Review their current order.
- Remove an item/remove all items from their current order.
- Provide delivery and payment details.
- Place an order.
- Receive confirmation in the form of an order number.

Menu Management System-

- Add a new/update/delete vendor to/from the menu.
- Add a new/update/delete food category to/from the menu.
- Add a new/update/delete food item to/from the menu.
- Add a new/update/delete option for a given food item.
- Update price for a given food item.
- Update default options for a given food item.
- Update additional information (description, photo, etc.) for a given food item.

Order Retrieval System -

- Retrieve new orders from the database.
- Display the orders in an easily readable, graphical way.
- Mark an order as having been processed and remove it from the list of active orders.

REQ-

WOS101 – Web Ordering System
MMS145 – Menu Management System
ORS200 – Order Retrieval System

Other Non-functional Requirements

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 circumstance

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.

Security Requirements

- A user password used for Account login must have a bit-strength of at least 64 bits.
- A user password used for Account login must be changed once a year.
- A user shall only be able to place orders on one system at a time.
- OTP will be generated each time the user wants to log in.

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

Business Rules

- The customer must supply a local telephone number number for web orders.
- A valid credit/debit card must be available for web order confirmation.
- The web order must be confirmed by the head waitress via return phone call.
- Preparation of the web order is not started until after the confirmation call.
- The credit/debit card number is held for web orders if the customer chooses to pay with cash or personal check.
- Payment for a web order is not processed until (failure of) customer pick-up.

Appendix B: Field Layouts

An Excel sheet containing field layouts and properties/attributes and report requirements.

Sample sheet with information required to register the customer

Field	Length	Data Type	Description	Is Mandatory
Account Number	16	Numeric		Y
ISFC code	11	Alphanumeric		Y
Card Amount	20	Numeric		Y
Mandate Start Date	8	Date	Date of Mandate Registration	N
Mandate End Date	8	Date	Date of Mandate Expiry	N
Status	25	Alphanumeric	Status of Registration	Y
Customer Name	60	String		Y
Reject Reason Code	4	String	Reject Reason code in casemandate is rejected	N

Sample Report Requirements: Include the fields to be included in the report

Registration Report	Transaction Report
Bank Account Number	Transaction Reference Number
ISFC Code	Bank Account Number
Bank Name	IFSC Code
Account Status	Bank Name
Account Type	Customer Name
Customer Name	Card Number
Card Number	Debit Transaction Amount
SI Start Date	Transaction Date
Status	Status
Remarks	Debit Attempt Number
	Remarks

Appendix C: Requirement Traceability Matrix

Sl. No	Requirement ID	Brief Description of Requirement	Architecture Reference	Design Reference	Code File Reference	Test Case ID	System Test Case ID
1	WOS101	Interface for users to place their orders and view and track orders	agile	Swiggy & Zomato	-	TC-1	STC-10
2	MMS145	Interface for Admins to view and manage the menu visible to the users.	agile		-	TC-2	STC-20
3	ORS200	Show the orders in a graphical and easily readable manner	agile	Swiggy & Zomato	-	TC-3	STC-30