ONLINE FOOD DELIVERY SYSTEM

SOFTWARE ENGINEERING PROJECT

SEMESTER- 5

SECTION – F

V RAGHAV LOKNATH – PES2UG20CS375

VISHAL C E – PES2UG20CS386

VISHNUDEEP M R – PES2UG20CS388

VISHWAS M – PES2UG20CS390

Problem Statement -
Develop an efficient software to order food online.

Objective –
The main objective of the system is to develop a customer friendly software where they can order the food easily by adding the items they wish to have in a cart, make the payment and have their food within a few steps.

Abstract –
When the customer first opens the software, he will be led to the recommendations page where the best restaurants will be displayed along with their bestsellers. The customer can search for the food in the search bar also. When a customer enters the food he wishes to have, all the restaurants which provide that particular food at that specific time will be displayed in an order such that it goes from the ones which are nearby to his house to ones which are far-away. The items which they wish to order will be added to a cart. Finally when they are done selecting all the items, they can view the cart and then update it. They can increase the quantity of the items, they can also remove the items and can also go back to ordering more.

Once they have finished selecting all the food items, they will be led to a page where they will enter basic information such as Name, Delivery Address, Contact No., Email ID. The customer can then review his order, the entire bill, his details and can edit

and confirm them. Finally in the checkout page they will be provided with multiple options to pay such as COD, UPI, Credit/Debit card, gift vouchers. After the payment is processed the customer can track his order and the ETA.

Software Requirements Specification

Introduction

Purpose

This SRS's goal is to list the functional and non-functional needs for the aforementioned restaurant food ordering system. The document also includes a full profile of the external interfaces, performance concerns, and design limitations placed on the ensuing implementation in addition to the above mentioned requirements. The document should serve as the basis for an effective and well-managed project completion as well as a reliable source of information in the future.

Intended Audience

The primary audience of SRS is mainly conveyed to the development team, testing team, project manager, clients and stakeholders.

Product Scope

The main purpose of the system is to provide efficient delivery within the system.
The goal of the system is to deliver the food quickly to the hungry customers.

Benefits:

1. No need for the customer to visit the restaurant and order food
2. They can order food from multiple restaurants at the same time
3. The food from the different restaurants gets delivered at the same time
4. Multiple orders can be placed with pitstop delivery.

References

1. https://www.studocu.com/in/document/shanmugha-arts-science-technology-and-research-academy/software-engineering/srs-food-order-system/24303017

2. https://www.gloriafood.com/online-food-ordering-and-delivery-system
3. https://www.swiggy.com

Overall Description

Product Perspective

The software for a whole restaurant food ordering system is the software described in this SRS. The system combines numerous pieces of hardware and software and also connects to external systems. For persistence and unhandled activities, it depends on a number of external interfaces, in addition to physically interacting with people.

Product Functions

In order to rapidly and conveniently handle customer invoicing, the online food delivery system interacts with an existing payment system, comprising a cash register and software accessible credit system. The payment system must be functional so that it can inform the RFO system as to whether a transaction was successful or unsuccessful.

User Classes and Characteristics

Three levels of user classes:

1. User
2. Developer
3. Physical and hardware tester.

Includes UI for all different views:

1. Mobile
2. Desktop.

Operating Environment

Customers will interact with the desktop Computer UI. Users interact with the system by dragging things about on the flat-screen touch-sensitive display, which is how this interface employs the surface computer paradigm. The Mobile UI is created to function on a small, wireless-enabled PC that servers can use to meet client requests. The kitchen crew has access to basic functionality regarding ordered goods thanks to the Display UI.

Design and Implementation Constraints

The RFOS should be created using an object-oriented programming language, with solid GUI connections and an easy-to-use network API. The three main contender tool chains are Python, C++ Java. The system must support parallel operation, hence must be deployed on cloud using any of the services such as AWS, Azure, GCP and its design must avoid scaling problems related to the number of connected tablets, surface computers, or screens at any given time. The system must be trustworthy enough to function more or less without errors or with strong enough error recovery capabilities to prevent users from ever discovering errors.

2.6 Assumptions and Dependencies

Consistent system without any hardware dependencies.

All the PC and desktops will provide fully functional scalability without any embedded dependency.

No fall through in cloud deployment.

External Interface Requirements

User Interfaces

This includes sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions that will appear on every screen, keyboard shortcuts, error message display standards.

Software Interfaces

NoSQL is used for maintaining the databases.
For high streaming online analytical processing we will be using Hive and HBase.

Communications Interfaces

Communications functions required by this product, includes e-mail, web browser, network server communications protocols, electronic forms. Some standards that will be used are FTP or HTTP. Synchronization mechanisms using 2FA.

Analysis Models

In Analysis Modeling, information, behavior, and functions of the system are defined and translated into the architecture, component, and interface level design in the design modeling.

1. Data Dictionary
2. Entity Relationship Diagram
3. Data Flow Diagram
4. State Transition Diagram
5. Process Specification
6. Control Specification
7. Data Object description

System Features

The major service provided by the product is online food ordering.

Mode of operation is management through online.

Three levels of user classes:

1. User
2. Developer
3. Physical and hardware tester.

System UI

5.1.1 Description and Priority

This includes sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions that will appear on every screen, keyboard shortcuts, error message display standards.

Users interact with the system by dragging things about on the flat-screen touch-sensitive display, which is how the surface computer paradigm is used in this interface. Users of the RFOS can move about items like food, dietary needs, tips, and menus on their table's surface. These things can be transferred into static objects, like meals and payments, to carry out a variety of tasks. In addition to this paradigm for object handling, a constrained system menu is required.

5.1.2 Stimulus/Response Sequences

Using a simple touch gesture, a double-tap on the touch surface, users will call up their restaurant menu, which is coupled with a system/command menu, and dismiss it using a similar gesture or by hitting a close button.

5.1.3 Functional Requirements

Weather prediction models are enabled as in case of rain or any discrepancies, the user will be notified in advance if his user will get delayed or he will be suggested with nearby restaurants so that his order will be delivered soon.

In case of invalid address inputs, the user's current location is detected and then a dialog box appears which asks the user whether the order needs to be delivered to his current location or the address entered.

Payment Gateway

5.2.1 Description and Priority

The payment gateway authorizes a user's transfer of funds between financial institutions to sellers without direct delivery of either bank or credit card account information. Payment gateways include PayPal, Google Wallet, Apple Pay, PhonePe,Whatsapp Pay and Amazon Pay.

5.2.2 Stimulus/Response Sequences

An integrated payment gateway wherein no third parties involved at the payment checkout stage of. Companies using integrated gateways obtain PCI DSS compliance, which means they're in charge of storing, securing, and conducting initial verification for each transaction.

Payment gateways also authorize payments for card-not-present transactions. To accept credit cards online, our system needs a payment gateway, as funds cannot be sent from one bank to another for security reasons.

In essence, the payment processor enables the transaction, whereas the payment gateway communicates the approval or the decline of the transaction between the merchant and the buyer. If it seems a bit complicated so far, be patient.

5.2.3 Functional Requirements

1. We use the RESTful API design convention for the payment service.
2. The payment service accepts payment events from users and coor
3. dinates the payment process. The first thing it usually does is a risk check, assessing for compliance with regulations such as AML/CFT , and for evidence of criminal activity such as money laundering or financing of terrorism. The payment service only processes payments that pass this risk check. Usually, the risk check service uses a third-party provider because it is very complicated and highly specialized.
4. PSP protocols are implemented to provide a safe transaction.

Other Nonfunctional Requirements

Performance Requirements

1. The server must be able to sustain an unlimited amount of active consumer payments; in other words, payments cannot ever be lost.
2. The server must have no upper restriction on the number of surface computers, tablets, and displays that can be connected to the system. No user's experience must be neglected / compromised due to many others using it at the same time.

Safety Requirements

1. The system must save every state and any changes that take place on all devices. This is done so that in a case where the app closes abruptly due to any reason, the previous state is remembered and is restored so that no order data is lost.
2. In a situation when the online order cannot be processed or facilitated, the menu of the restaurant should be displayed to the user so that he/she can order manually.
3. The database which stores the required data is protected using multi-layered protection to keep the data safe from attackers/hackers.
4. The stored data, such as payment details, locations and other information, will also be encrypted so that in a situation where the database does get attacked, data breach or loss is minimal.

Security Requirements

1. All users which includes the waiter/hotel representative, the customer, the delivery personnel will use 2 factor authentication to login to the application or website.
2. The waiter is allotted a password from the hotel which they login using and this password is different for every waiter and must be changed periodically. A waiter can only log into one system at a time.
3. Customers and delivery personnel only need to set and login using password and 2 factor authentication the first time they login on any device. For this they need to enable the remember me option that is displayed in the login page of the application or the website.
4. The password should be set according to security requirements such as at least one uppercase letter, one lowercase letter, one number, one special character and it should be longer than 8 characters.

Software Quality Attributes

1. The software must be highly versatile, i.e., it should work on the majority of devices with various form factors and hardware specification, without any issues.
2. The software must be user-friendly such that even a first time user wouldn't feel very uncomfortable using it.
3. The software must be integrated with Google Maps so that the customer can track the location of the delivery person and the customer’s food.
4. All the features of the application/ website should be available simultaneously on multiple devices.

Business Rules

An invoice processing decision-making system where only particular supervisors can approve bills reaching a certain amount.

Other Requirements

1. Android Version 7 and above
2. iOS version 12 and above
3. Windows 7 and above
4. If using the website, use the most up-to-date Google Chrome browser or Mozilla Firefox Browser.