Software Requirements Specifications

for

**Online Food Delivery System**

Prepared by: (Team 1)

1) Aditya More

2) Akash Goyal

3) Tulasi Krishna

4) Basit Fayaz

5) Kshitij Nishant

**Contents**

ABSTRACT………………………………………………………………….….3

Existing System………………………………………………………......3

Proposed System………………………….…………………………..….3

SOCIO TECHNICAL SYSTEM…………………………………………..........3

MODEL………………………………………………………………………...4

RUP Model (Rational Unified Process Model)……………………...…..4

Advantages of RUP Software Development…………………………….4

Other Models…………………….……………………….…………..….4

Waterfall model................................................................................4

Incremental model…………………………………………………4

RAD model……………………………………………………......4

Prototype model………………………………...…………………4

Spiral model…………………………………………………….....4

FUNCTIONAL & NON-FUNCTIONAL REQUIREMENTS………………………………………………..……………..5

Functional Requirements………………………………...……………….5

Non-Functional Requirements…………………………………...…….…7

UML DIAGRAMS……………………………………………………………...9

Sequence Diagram…………………………………………….…………9

Activity Diagram…………………………………………………………9

Class Diagram……………………………………………...…………...10

User-case Diagram……………………………………………...………10

TESTING………………………………………………………………….…..11

Unit Testing……………………………………………………….........11

MODULES………………………………………….……………..….………12

SCREENSHOTS………………………………………………………………13

**ABSTRACT**

Online Food Delivery System is a system which will help restaurant to optimized and control over their restaurants. For the waters, it is making easier because they don't have to go kitchen and give the orders to chef easily. For the management point of view, the manager will able to control the restaurant by having all the reports to hand and able to see the records of each employees and orders. This application helps the restaurants to do all functionalities more accurately and enhances the spend of all the tasks Online Food Delivery System reduces manual works and improves efficiency of restaurant.

The online food delivery system set up menu online and the customers easily places the order with a simple mouse click. Also with a food menu online you can easily track the orders, maintain customer's database and improve your food delivery service. This system allows the user to select the desired food items from the displayed menu. The user orders the food items. The payment can be made online or pay-on-delivery system. The user's details are maintained confidential because t maintains a separate account for each user. An id and password is provided for each user. Therefore, it provides a more secured ordering.

**Existing System:**

In existing system for giving any orders users should visa hotels or restaurants to know about food items and then give order and pay advance. In this method time and manual work is required. Maintaining critical information in the files and manuals is full of risk and a tedious process.

**Proposed System:**

This online application enables the end users to register online, select the food from the e-menu card, read the E-menu card and order food online. By just selecting the food that the user wants to have. The results after selecting the food from the E-menu card will directly appear in the screen near the Chef who is going to cook the food for you. By using this system, the work of the Water is reduced and we can also say that the work is notified. The benefit of this is that if there is rush in the Restaurant then there will be chances that the waiters will be unavailable and the users can directly order the food to the chef online by using this system. The user will be given a username and a password to login.

**SOCIO TECHNICAL SYSTEM**

This system is a socio-technical system as the system has such properties:

• Involves operational processes

• Include people who use and interact with the technical system.

So, we can conclude that the system is Social technical system because it involves operational processes (like place, order, online payment) and include people.

**MODEL**

**RUP Model (Rational Unified Process Model)**

For the implementation of the system the RUP Model (Rational Unified Process Model) is used. Its goal is to ensure the production of the high-quality software that meets the needs of its end users within a predictable schedule. It is easily customized and useful for web-based implementations.

RUP methodology has a highly flexible development path. It uses the some of the industry's best practices. These are known as six best practices of RUP methodology. The reasons why we are using RUP model for developing this system are also these best practices.

**Develop Iteratively:** The software requirements specification (SRS) keeps on evolving throughout the development process and loops are created to add them without affecting the cost of development.

**Manage Requirements:** The business requirements documentation and project management requirements need to be gathered properly from the user in order to reach the targeted goal.

**Use Components:** The components of large project which are already tested and are in use can be conveniently used in other projects. This reuse of components reduces the production time.

**Model Visually:** Use of Unified modelling language (UML) facilitates the analysis and design of various components. Diagrams and models are used to represent various components and their interactions.

**Verify Quality:** Testing and implementing effective project quality management should be a major part of each and every phase of the project from initiation to delivery (aka the project management life cycle).

**Control Changes:** Synchronization of various parts of the system becomes more challenging when the parts are being developed by various teams working from different geographic locations on different development platforms. Hence special care should be taken in this direction so that the changes can be controlled.

**Advantages of RUP Software Development**

1. This is a complete methodology with an emphasis on accurate documentation.

2. It is proactively able to resolve the project risks associated with the client's evolving requirements requiring careful change request management

3. Less time is required for integration as the process of integration goes on throughout

the software development life cycle.

4. The development time required is less due to reuse of components.

**Disadvantage of RUP software Development**

1. The team members need to be expert in their field to develop a software under this methodology.
2. The development process is too complex and disorganized.
3. On cutting edge projects which utilise new technology, the reuse of components will not be possible. Hence the time saving one could have made will be impossible to fulfil.
4. Integration throughout the process of software development, in theory sounds a good thing. But on particularly big projects with multiple development streams, it will only add to the confusion and cause more issues during the stages of testing.

**FUNCTIONAL & NON-FUNCTIONAL REQUIREMENTS**

1. *Functional Requirements*
2. **Registration**

If customer wants to order the food then, the user, should first register, or else unregistered users cannot go ahead and order.

1. **Login**

The customer should login to the system by entering the valid credentials he/she provided during registration, like user ID and password.

1. **Display the menu**

Menu page gets loaded after logging in. The page contains all the items with their rates.

1. **Modify Menu**

Any addition or removal of items are made on this page. To ensure what the customer has selected.

1. **Select Food item(s)**

Customers are free to select from the various range of food items we have to provide them with.

1. **Changes to order**

Before final review, customer can view the items they select before finally moving forward with payment.

1. **Review the order before submitting**

Before completion of order, customer is asked to confirm the name, phone number, location(address) and placed order accordingly. Then they should submit the order after finalizing.

1. **Payment**

There are many modes of payment that the customer can select to pay us from. Security of our customer’s payment details is one of our top priorities.

1. **Provide delivery and payment details**

On this page, bill is generated. Order number and payment is given and delivery confirmation is done.

1. **Logout**

Customer will be asked to either continue with ordering more food or logging out of the system. Inactive users will be logged out of the system automatically to maintain server load every month.

1. *Non-Functional Requirements*
2. **Portability**

The system will be able to switch from one platform to another without issues.

1. **Reliability**

Consistent service in a user-acceptable manner while operating within the environment for which the system was intended.

1. **Availability**

The system must be available at all times so that users can access it, and should be restricted only by downtime of the server on which the system is running.

1. **Maintenance**

Maintaining a database is done and the application server takes care of the site.

1. **Security**

Securing customer’s data is a must. It is always kept in mind while designing/updating the web application.

1. **User Friendly**

Easy to use and understand by the user.

1. **Performance**

The page should be perform effectively.

1. **Efficient**

It should work effectively without getting hanged even in heavy traffic of orders placed.

1. **Safety**

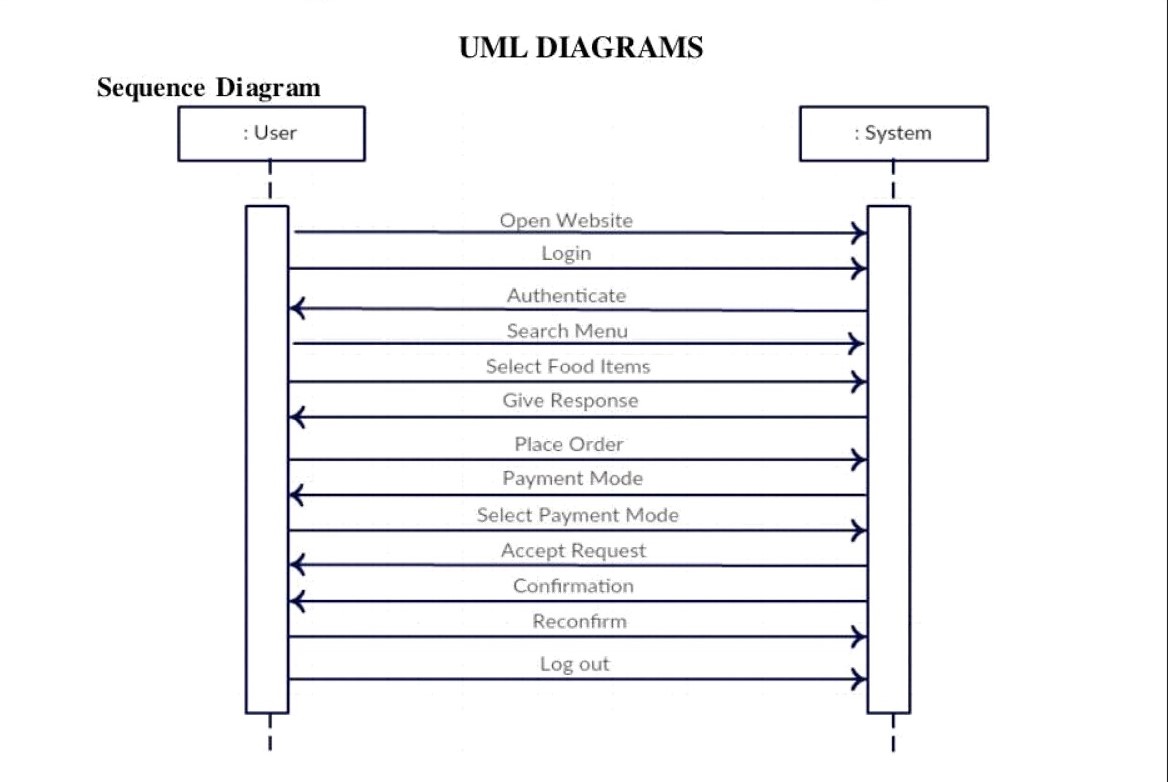
No loss or damage of user data from database of system.

1. **Privacy**

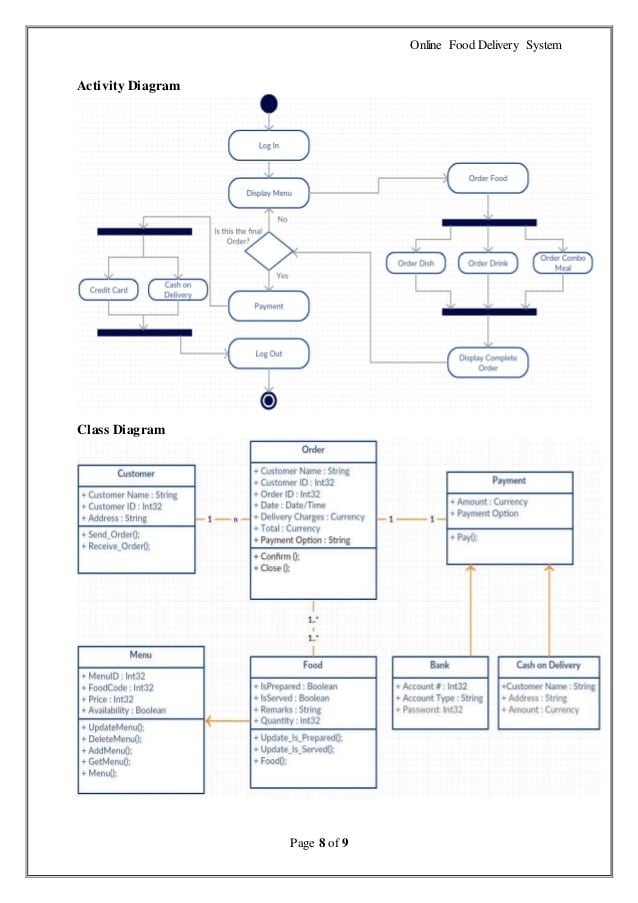
Personal data in the system should not be disclosed to anyone.

**UML Diagrams**

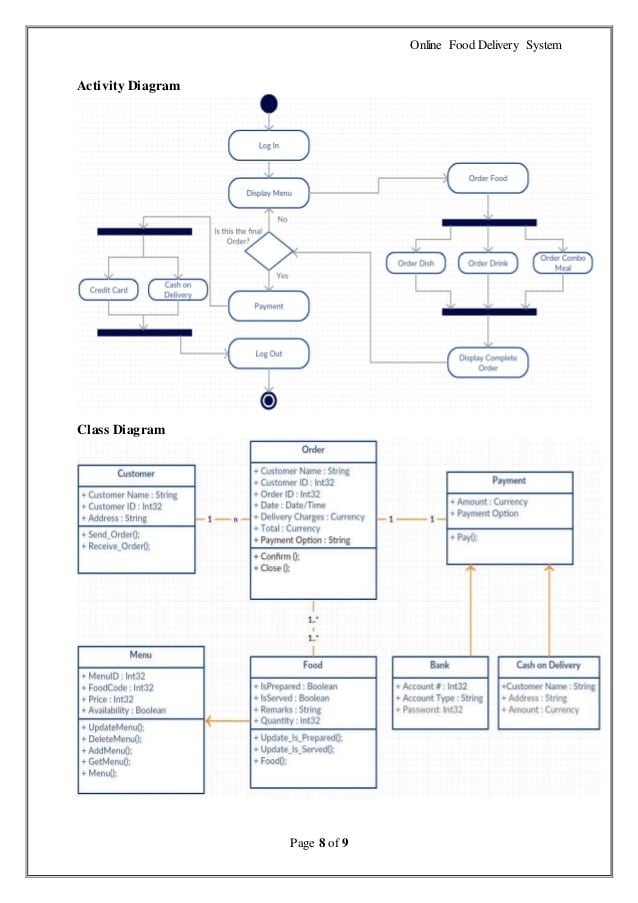
*1.Sequence Diagram*

****

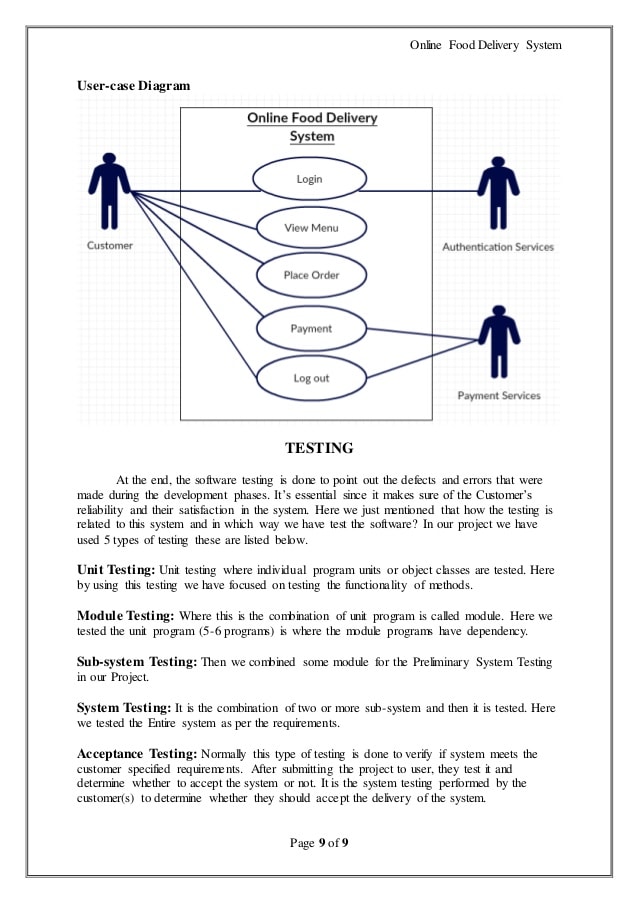
*2.Activity Diagram*

**

*3.Class Diagram*

**

*4.User-Case diagram*

**

**TESTING**

At the end, the testing of system software is done to figure out the defects and errors that were made during the development phase. It’s crucial since it makes sure of the reliance of the Customer’s trust and their satisfaction for user engagement. Here we mention how testing is related to this delivery system and in which way we have tested the software. These are listed as follows:

1. *Unit Testing*

Discrete program object classes are tested by focusing on functionality of methods.

**MODULES**

Modules of the Online Food Delivery System:

These are the main modules of the project:

* **Order Module:** We can create, read, update and delete Order from this module
* **Food Item Module**: All the operations related to Food Item, is managed by this module.
* **Sales Module:** Sales Module is used to manage the Sales
* **Delivery Module**: It has been developed for managing the Delivery
* **Payment Module:** It manages the Payment
* **Customer Module:** Customer operations will be managed by Customer module

SALES

FOOD ITEM

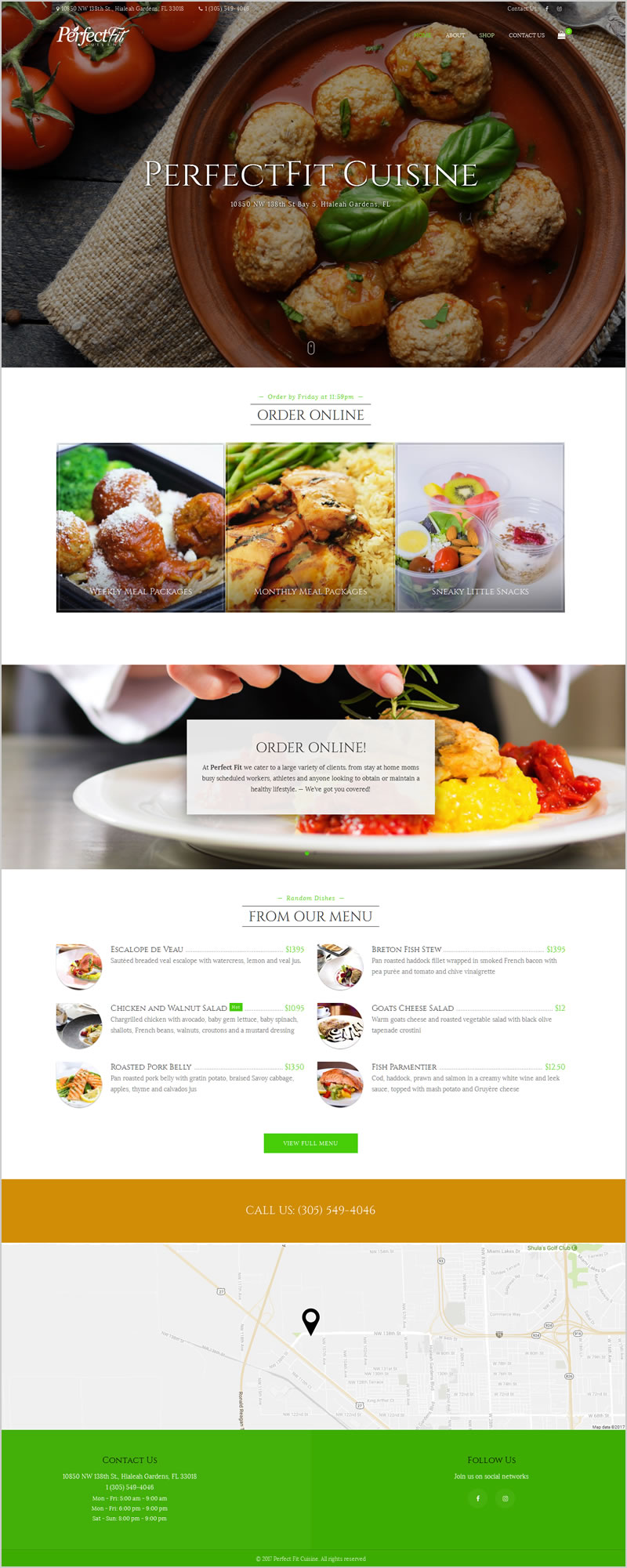
CUSTOMER

PAYMENT

DELIVERY

ORDER

**SCREENSHOTS**

****

**Graphical user interface, application

Description automatically generated**

**Graphical user interface, website

Description automatically generated**