ASSIGNMENT
SRS of UIU Eatery

SYSTEM ANALYSIS & DESIGN

Submitted by:
Md Sazzad Mazumder
ID: 011201285
CSE 3411(A)

SRS of UIU Eatery

1. INTRODUCTION

1.1 PURPOSE

The purpose of this document is to build an online food ordering system to manage the orders and food delivery system for students and faculty in the university cafeteria in order to tackle the problem of overcrowding the cafeteria and knowing the available food items beforehand.

1.2 DOCUMENT CONVENTIONS

This document uses the following conventions.

- Headings and subheadings to organize the document
- Numbered or bulleted lists to present information in a clear and concise manner
- Consistent use of fonts, font sizes, and formatting styles
- Use of tables, graphs, and other visual aids to present complex information
- Consistent use of terminology and abbreviations
- Use of citations and references for external sources and materials

1.3 INTENDED AUDIENCE AND READING SUGGESTIONS

This project is a prototype for the food ordering system and it is open for all personals for education purpose or business purpose. This project is useful for any type of organization that contains cafeteria.

1.4 PROJECT SCOPE

The purpose of the online food ordering system is to provide a user-friendly platform for students and faculty to place food orders and have them delivered to their location on campus. The system will have a database of available food items and their prices, and users will be able to view menus, place orders, and make payments through the platform. The system will also provide real-time updates on the status of orders and delivery times. The scope of the project is limited to the university campus and the participating food vendors.

1.5 REFERENCES

- https://pathao.com/bn/
- https://hungrynaki.com/

2. OVERALL DESCRIPTION

2.1 PRODUCT PERSPECTIVE

The food ordering system will consist of the following components:

● Menu and food item details:

It will include the list of food items available for ordering, along with their descriptions, prices, and images.

● Customer details:

It will include customer information such as name, email, phone number, and address, which will be used for delivery purposes.

● Order details:

It will include information about the customer's order, such as the food items selected, the quantity of each item, the total price, and the delivery time and location. This information will be stored for order tracking and management purposes.

2.2 PRODUCT FEATURES

- Search & choose item
- Cart
- Reg. for homemade food by students with admin approval.
- Room delivery for faculty and homemade food.
- Payment options (Bkash, Nagad, UCAM and COD)
- Review & Rating
- Stock availability check
- **Order history**
- **Complaint box**
- **Admin order monitoring**
- **Notification**
- **Add & Update menu items**
- **Show current crowd status of cafeteria**

2.3 USER CLASS and CHARACTERISTICS

The users of the food ordering system include students and faculty members of the university who wish to order food online. They are tech-savvy and comfortable with using online applications. They are expected to have basic knowledge of how to use a computer, a web browser, and a smartphone. They should also have the capability to pay online using mobile banking, e-wallets, and cash on delivery. The admin will have access to the order monitoring and management functions, and should have the necessary technical skills to manage the system. Students & faculty should be able to use all the features available other than room delivery.

Faculties should have following extra features functionalities:

● FACULTY FUNCTIONS.

• Room delivery

● RESTAURANT OWNER
• Add/remove food item
• Accept or reject pending orders
• All orders pending/completed
• View Complaints
● ADMINISTRATIVE
• Accept/reject restaurant
• Block restaurants or students
• View Complaints

2.4 OPERATING ENVIRONMENT

Operating environment for the airline management system is as listed below.
● Web-based application
● Operating system: Windows
● Web server: Xampp
● Database: MySQL
● Programming languages: HTML, CSS, JavaScript, PHP

2.5 DESIGN and IMPLEMENTATION CONSTRAINTS

1. The database schema should be designed to accommodate all the necessary information related to the food items, customers, orders, payments, etc.

2. The user interface should be designed in a way that is easy to use and navigate for the customers.

3. The payment gateway should be secure and reliable to ensure that customer data is protected.

4. The system should be able to handle a large number of orders and customers simultaneously without any lag or downtime.

5. The system should have a mechanism to check the availability of food items in real-time and update the stock levels accordingly.

6. The system should have a feature to generate various reports such as order history, sales, and customer feedback.

7. The implementation should be done using modern web development frameworks and libraries to ensure scalability and maintainability of the system.

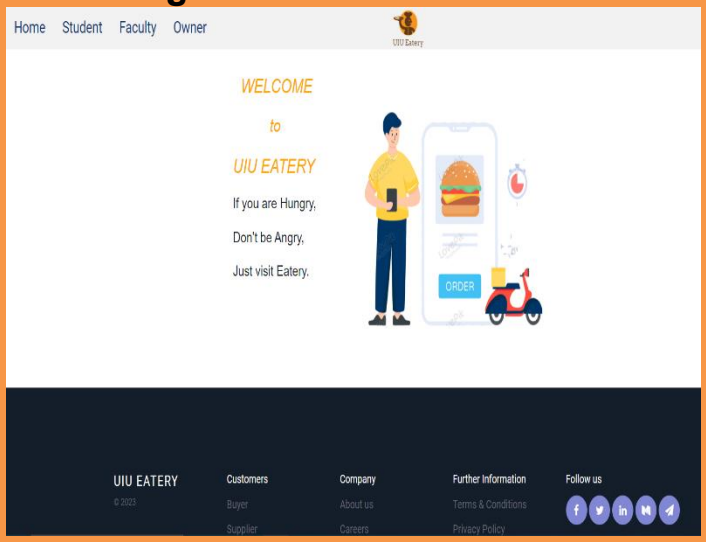
2.6 ASSUMPTION DEPENDENCIES

Let us assume that this is a full-fledged food ordering system and it is used in the following application:

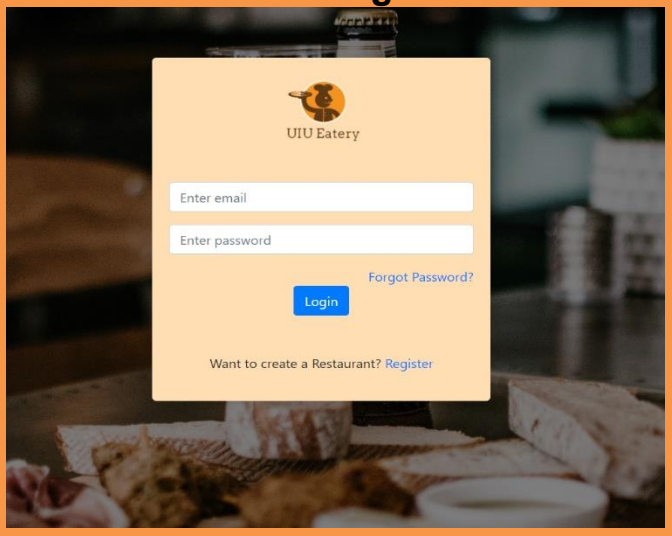
- The availability of food items is updated in real-time.
- Users have a stable internet connection to access the system.
- The payment gateway and other third-party services integrated with the system are working correctly.
- The delivery personnel can access all areas of the university campus.
- The homemade food submitted by students for approval meets the quality standards set by the administration.
- The administration can monitor and manage the system efficiently to ensure smooth operations.

2.7 UI OF THE SYSTEM

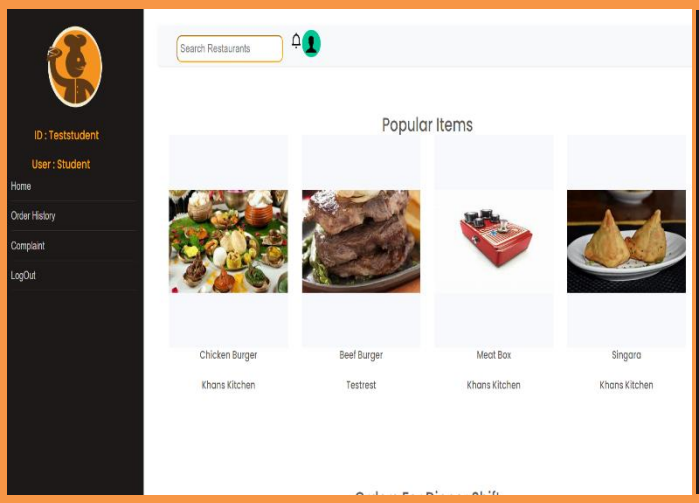
Home Page



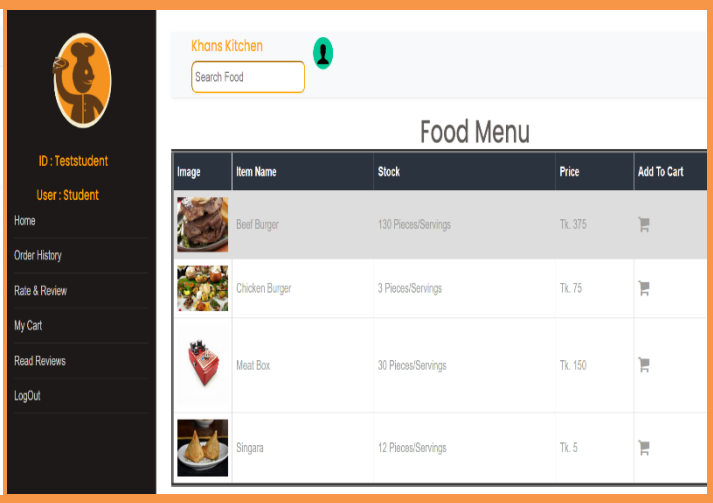
Restaurant owner login



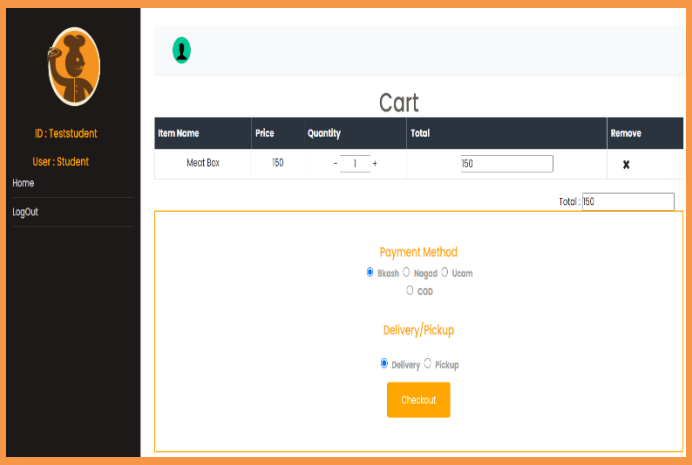
Restaurants page



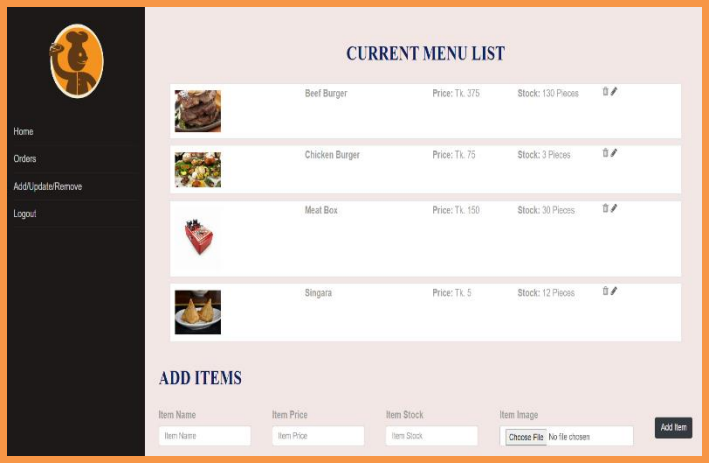
Food Menu



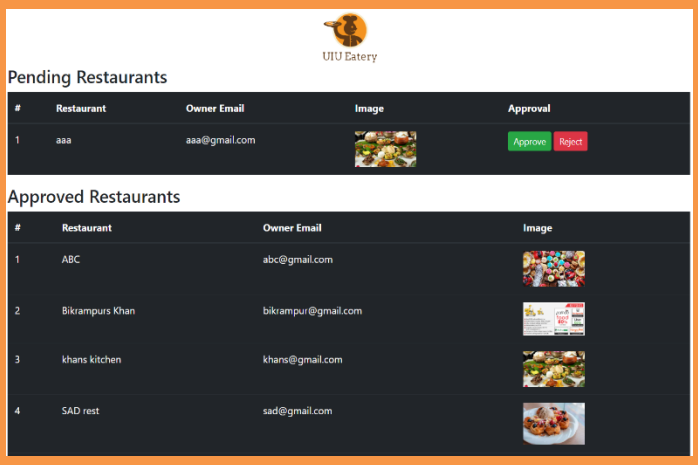
Cart & Order



Update menu list by owner



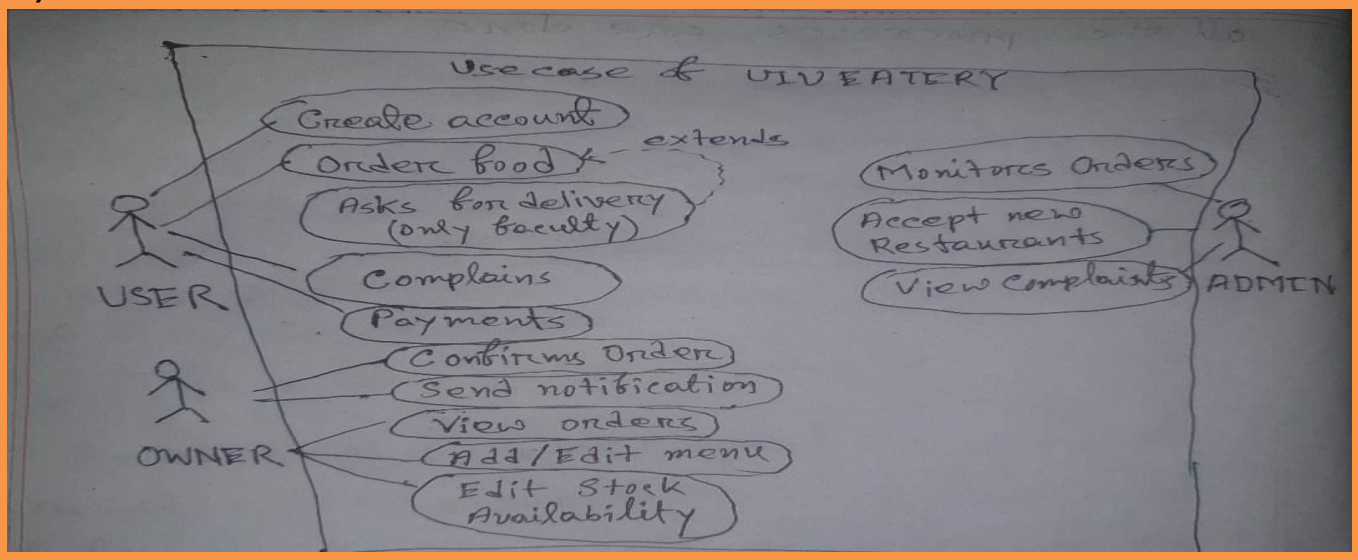
Admin Aproval



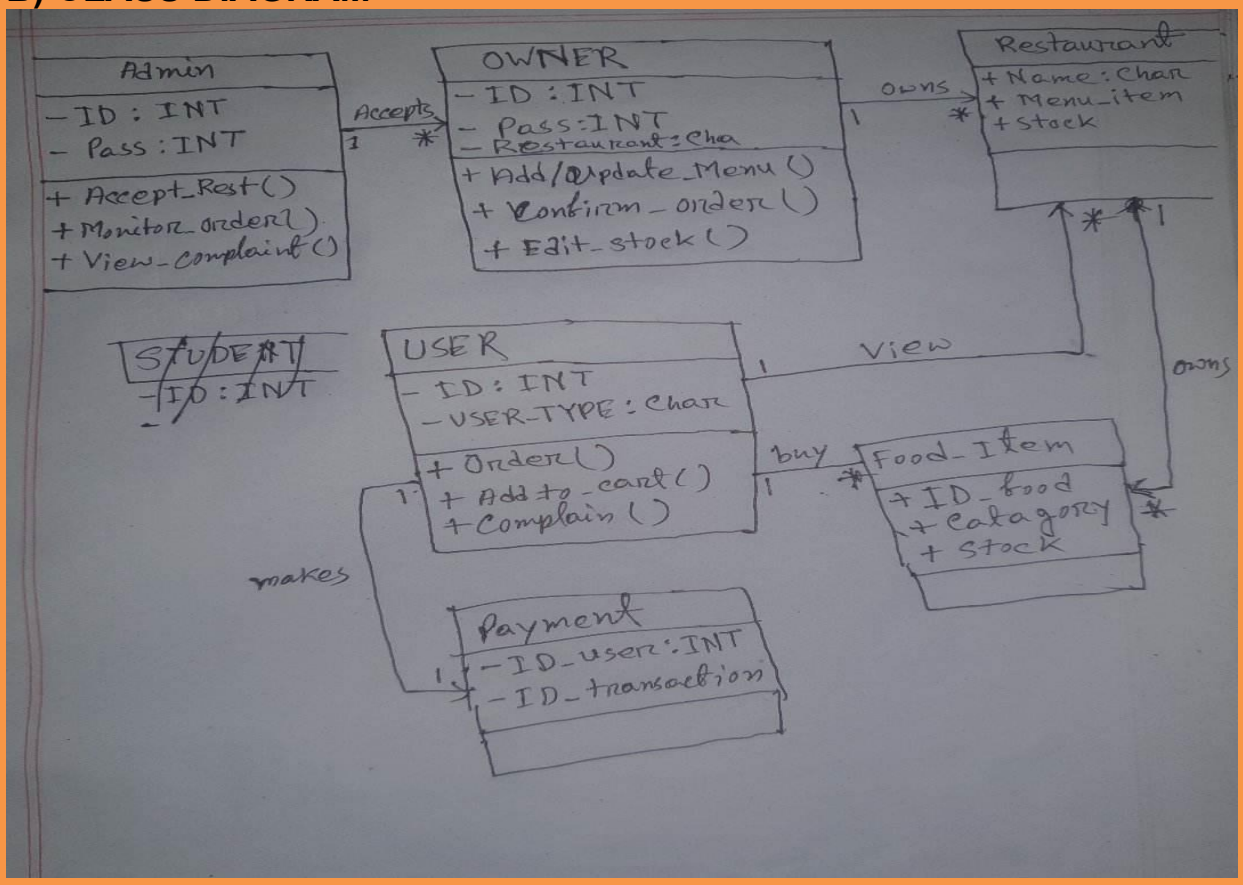
Admin Complaint Review

2.8 GRAPH & DIAGRAMS

A) USE CASE



B) CLASS DIAGRAM



3. SYSTEM FEATURES

● DESCRIPTION and PRIORITY

The food ordering system maintains information on users,
personal preferences, prices, and order history. Of course, this project has a medium
priority because it is helping an existing real system to maintain it virtually and efficiently.

● STIMULUS/RESPONSE SEQUENCES

○ Search for food items
○ Add items to cart
○ Review cart
○ Place an order
○ Rate & Review
○ View order history

● **FUNCTIONAL REQUIREMENTS**

Other system features include:

- Register as a student to provide homemade food with admin approval
- Check the availability of stock
- Provide a complaint box for customers
- Allow admin to monitor orders
- Send notifications to customers regarding the status of their order
- Show the current crowd status of the cafeteria

4. EXTERNAL INTERFACE REQUIREMENTS

4.1 USER INTERFACES

● Front-end software: HTML, JavaScript
● Back-end software: SQL, PHP

4.2 HARDWARE INTERFACES

● Windows.
● A browser which supports CGI, HTML & Javascript.

4.3 SOFTWARE INTERFACES

Following are the software used for the system application.

Software used Description

Operating system
We have chosen Windows operating system for its best support and user-friendliness.
Database
To save the order & user records records we have chosen SQL database.
PHP
To implement the project we have chosen PHP language for its more interactive support.

4.4 COMMUNICATION INTERFACES

This project supports all types of web browsers. We are using simple electronic forms for the orders, searching etc.

5. NONFUNCTIONAL REQUIREMENTS

5.1 PERFORMANCE REQUIREMENTS

The system should be able to handle a high volume of orders and transactions during peak hours without any delays or errors. The response time for each action should be quick to provide a seamless user experience. The system

should also be able to handle a large database of menus, customers, and orders efficiently. The system's performance should be monitored regularly to identify any bottlenecks and optimize the system for better performance.

5.2 SAFETY REQUIREMENTS

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure.

5.3 SECURITY REQUIREMENTS

Security systems need database storage just like many other applications. However, the special requirements of the security market mean that vendors must choose their database partner carefully.

5.4 SOFTWARE QUALITY ATTRIBUTES

● RELIABILITY: The system should reliably process and fulfill orders without errors or delays, ensuring that customers receive their food on time and in the correct quantity and quality.

● SCALABILITY: The system should be able to handle a large number of users and orders without slowing down or crashing.

● SECURITY: The system should be secure and protect sensitive customer information, such as payment details and personal information.

● USABILITY: The system should be user-friendly and easy to navigate for both customers and administrators.

● EFFICIENCY: The system should efficiently process orders and payments, minimize wait times for customers, and optimize delivery routes for delivery personnel.

● MAINTAINABILITY: The system should be easy to maintain and update, allowing for quick fixes and improvements to be implemented.