NAME   : NIKITA GANAPATI PATIL

EMAIL ID   : NIKSPAT@UMD.EDU

Table of Contents:

Figures:

1.0. Introduction

1.1. Purpose

The purpose of this document is to present a detailed description of the Dining Order System. It will explain the purpose and features of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both, the stakeholders and the developers of the system. It will be proposed to the University Dining Hall Management for its approval.

1.2. Scope of Project

This software system is for ordering hassle free meals from the dining hall. This software system will be used by a student to order meal and by a dining hall manager to prepare the requested meal. This will help the manager to manage the orders efficiently and minimize the food wastage. The list of food items available will be displayed on the interface. The students can login and order his/her meal two weeks in advance and latest by two days before the actual delivery date. The update in food items, timings of dining hall or holidays will be displayed on the notice section of the interface. The system will be using a database from the university which has a unique ID for each student. For any unexpected input used to login and password, system will display an error message.

1.3. Glossary

Term	Definition
Database	Central university database with student information
Food Items	Dishes available with their name, calorie and content.
Manager	Person who can add, edit and delete the food items in list.
Student	Person placing an order for the food.
Unique ID	Each student having an university unique identification number.

2.0. Overall Description

2.1   System Architecture
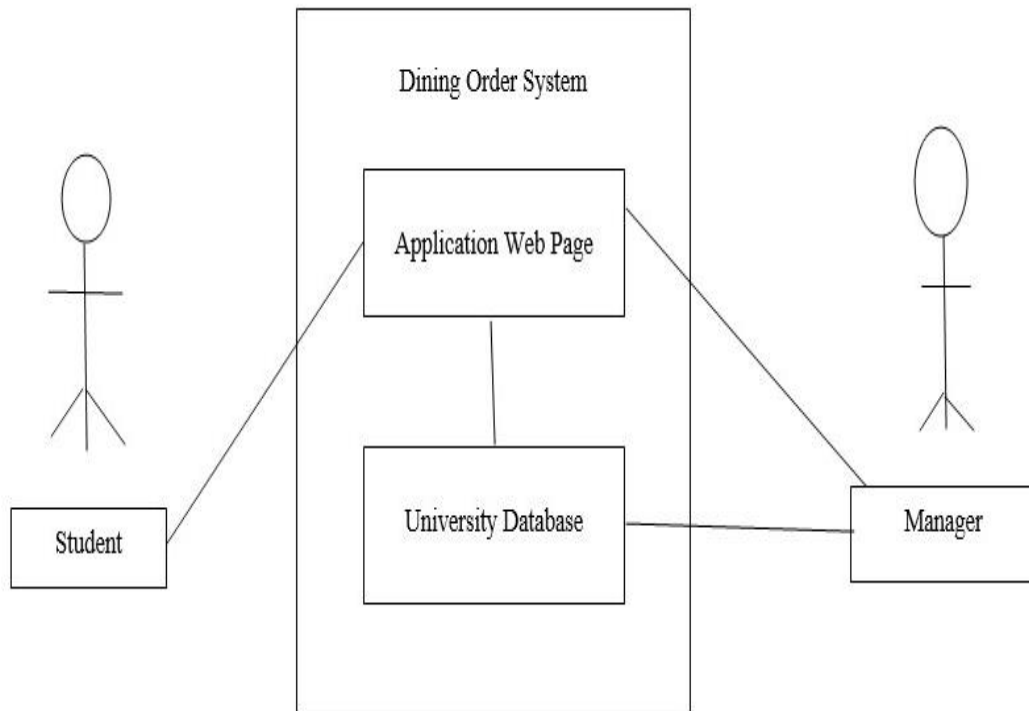
Figure 1 - System Environment

The Dining Order System has two active actors namely the Student and the Manager. The student can access the Dining Order System through a website using the Internet. Any Student can view and order the meal, view notices and edit the meal using the same web page. The Manager accesses the entire system directly. The Manager can add, edit and delete the food items in the list as the menu changes in the dining hall.

The student needs to login to be able to order the meal from the system. The manager is required to login to be able to perform any operations on the database.

2.1.1 Entity Relationship Diagram:

The logical structure of the data to be stored in the Dining Order System database is given below.
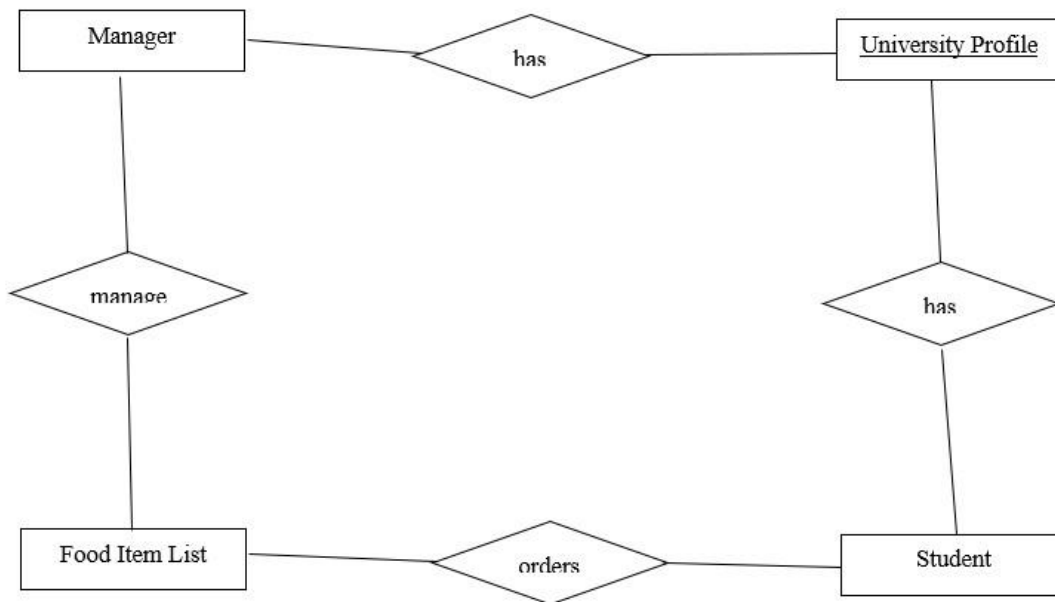
Figure 2 – Entity Relationship of the Dining Order System

The data descriptions of each of these data entities is as follows:

Food Item List

Data Item	Type	Description	Comment
Name	Text	Name of dish (food item)	Primary Key, Unique

Calorie	Integer	Number of calories in the dish	
Contents	Text	Contents of the food item	

Notice Item List

Data Item	Type	Description	Comment
NoticeTitle	Text	Title of the notice	Primary key, Unique
NoticeContent	Text	Description of the notice	
NoticeDate	Text	Date when manager publishes the notice	

Food Order List

Data Item	Type	Description	Comment
OrderID	Text	Order ID of the placed order	Primary key, Unique
DeliveryDate	Text	Delivery date of the food	
OrderedItems	Text	List of ordered items	
StudentName	Text	Name of the student	
StudentID	Text	ID of the student	

Asp Net User Roles

Data Item	Type	Description	Comment
UserID	Text	User ID of the student	Primary key, Foreign key reference from 'Id' of AspNetUsers table.
RoleID	Text	Name of the role	Primary Key, Foreign key reference from 'Id' of AspNetRoles table.

Asp Net Roles

Data Item	Type	Description	Comment
Id	Text	ID of the user	Primary key, Unique
Name	Text	Role of the user	
NormalizedName	Text	Normalization of name for ease and security of use	
ConcurrencyStamp	Text	Phone number	

Asp Net Users

Data Item	Type	Description	Comment
Id	Text	User ID of the student	Primary key, Unique
AccessFailedCount	Integer	Number of failed login attempts	
Email	Text	Email of the user	
ConcurrencyStamp	Text	A random value that must change when a user is persisted to store	
EmailConfirmed	Integer	Email verification to check if the email exists	
LockOutEnabled	Integer	Flag to lockout the account after three failed login attempts	
LockoutEnd	Text	Lockout reset time to unlock the account after twenty-four hours	
Name	Text	Name of the user	
NormalizedEmail	Text	Normalized email for ease and secure use	
NormalizedUserName	Text	Normalized username for secure and ease of use	

PasswordHash	Text	Store the hashed password	
PhoneNumber	Text	Phone number of user	
PhoneNumberConfirmed	Integer	Verification of the phone number	
SecurityStamp	Text	A random value that changes whenever the user credentials change	
TwoFactorEnabled	Integer	Flag to check if two factor authentication is enabled	
Username	Text	Username of the user	

4.0. Security

The server on which the Dining Order Services resides will have its own security to prevent unauthorized write/delete/edit access. There is no restriction on read access. The server will handle limited number of users to prevent the denial of service. The manager will have access to the web application only from restricted physical machines making it difficult to be misused.

The client machines from where the students will login, will have restricted authorization and authentication to prevent from attacks. The students enrolled in university will only have access to this site to order the food items.

The students and the manager both have to enable 2FA to order the food, which enhances the security of the application. The session management takes care of the cookies and each cookie expires after twenty minutes of inactivity. All the input fields are sanitized as per requirements and

conditions mentioned above and new user registration is out of scope of the application as mentioned.

The student has different privilege and the manager has different privilege. Based on their role-based access they can perform different activities. The password is hashed with PBKDF2 algorithm and stored as a hashed value. There is a minimum limit of 8 characters length on password out of which 1 must be uppercase letter, 1 lowercase letter, 1 special character and 1 digit. The registration is not defined in the application however, all the passwords satisfy the above condition. The user gets locked out after 3 failed login attempts and gets reset after 24 hours from the time of lockout. The password reset and forget password are out of scope of the application.

4.0. References

www.google.com
https://learn.microsoft.com/