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

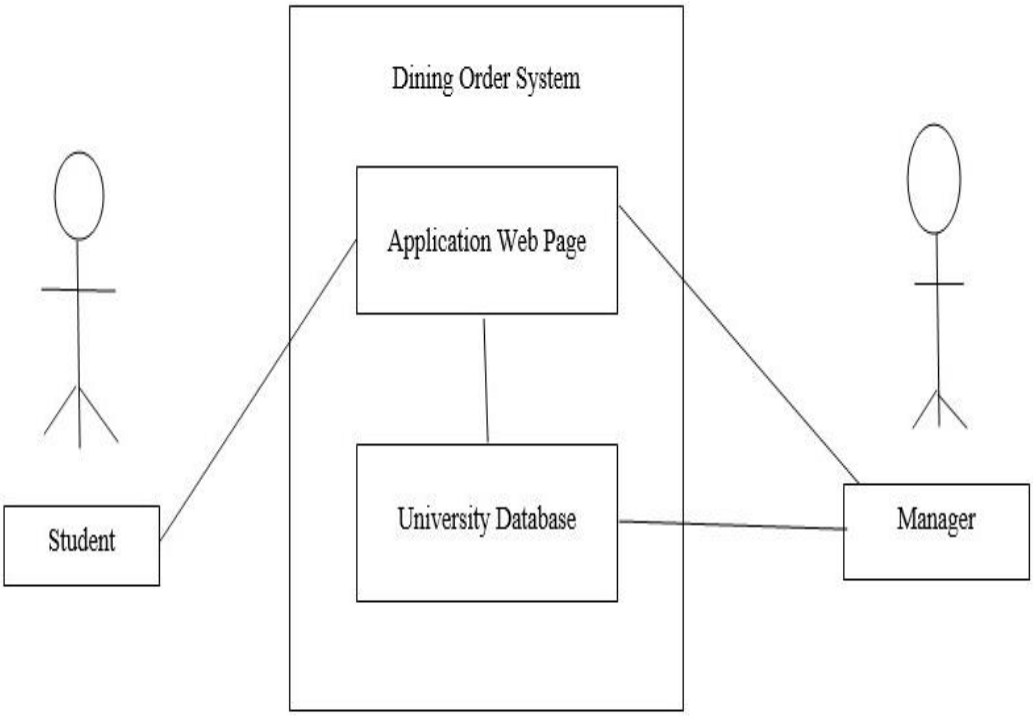

Figure 1 - System Environment

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.

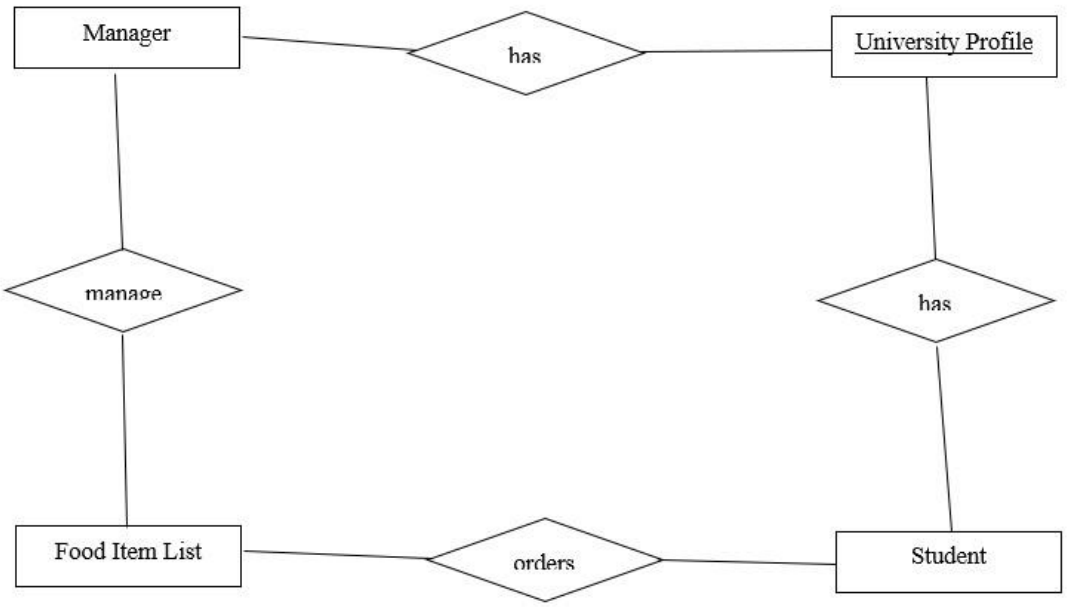

Figure 2 – Entity Relationship of the Dining Order System

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

University Profile

Data Item	Type	Description	Comment
Name	Text	Name of the university enrolled student/manage	
University ID	Text	Unique university ID	Unique
Email Address	Text	Internet address	
Phone number	Text	Phone number	

3.0 Threat Modeling and its fixes:

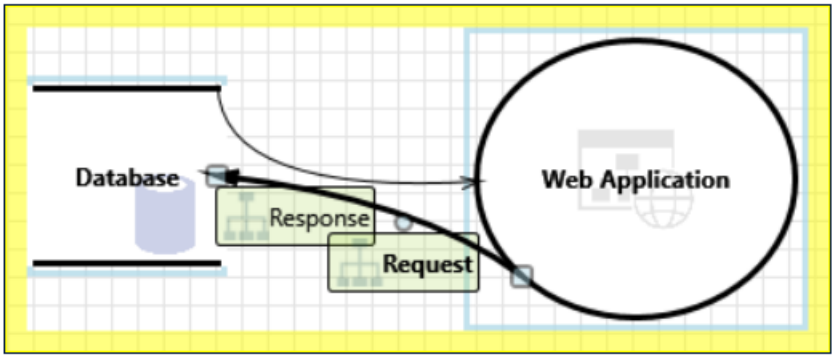

Figure 3: Threat model

An adversary can deny actions on database due to lack of auditing [State: Mitigation Implemented] [Priority: Medium]

Category:	Repudiation
Description:	Proper logging of all security events and user actions builds traceability in a system and denies any possible repudiation issues. In the absence of proper auditing and logging controls, it would become impossible to implement any accountability in a system.
Justification:	The database auditing and logging is enabled and user activites in web application are logged.
Possible Mitigation(s):	Ensure that login auditing is enabled on SQL Server. Refer: https://aka.ms/tmtauditlog#identify-sensitive-entities
SDL Phase:	Implementation

An adversary can gain access to sensitive data by sniffing traffic to database [State: Mitigation Implemented] [Priority: High]

Category:	Information Disclosure
Description:	An adversary can eaves drop on communication between application server and Database server, due to clear text communication protocol usage.
Justification:	The communication between application server and database server is encrypted.
Possible Mitigation(s):	Ensure SQL server connection encryption and certificate validation. Refer: https://aka.ms/tmtcommsec#sqlserver-validation Force Encrypted communication to SQL server. Refer: https://aka.ms/tmtcommsec#encrypted-sqlserver
SDL Phase:	Implementation

An adversary can gain unauthorized access to database due to lack of network access protection [State: Mitigation Implemented] [Priority: High]

Category:	Elevation of Privileges
Description:	If there is no restriction at network or host firewall level, to access the database then anyone can attempt to connect to the database from an unauthorized location
Justification:	Database is only accessible through the web application server and is not exposed to the internet.
Possible Mitigation(s):	Configure a Windows Firewall for Database Engine Access. Refer: https://aka.ms/tmtconfigmgmt#firewall-db
SDL Phase:	Implementation

An adversary may leverage the lack of monitoring systems and trigger anomalous traffic to database [State: Mitigation Implemented] [Priority: High]

Category:	Tampering
Description:	An adversary may leverage the lack of intrusion detection and prevention of anomalous database activities and trigger anomalous traffic to database
Justification:	Database server has proper authorization and authentication to prevent any malicious attacker, to perform activites. Also the database logs are used to detect anamolous traffic to the database.
Possible Mitigation(s):	Enable Threat detection on Azure SQL database. Refer: https://aka.ms/tmtauditlog#threat-detection
SDL Phase:	Design

3.3.2 Security

The server on which the Dining Order Services resides will have its own security to prevent the 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.