NAME : NIKITA GANAPATI PATIL

EMAIL ID : NIKSPAT@UMD.EDU

Table of Contents:

1.0 Features
2.0 User Characteristics
3.0 Non-Functional Requirements
4.0 Functional Requirements
    4.1 Student Use Cases
    4.2 Manager Use Cases

1.0 Features

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.

2.0 User Characteristics

The student and manager are expected to be Internet literate and be able to use a search engine. The main screen of the Dining Order System will have the search function, list of food items available, notice and login in option.

The student and manager are expected to be enrolled in the University and have their username and password with them. The manager is expected to be basic input form, Windows literate and to be able to use button, pull-down menus, and similar tools.

3.0 Non-Functional Requirements

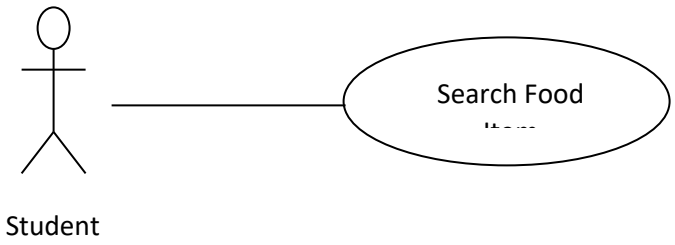
The Dining Order System will be on a server with high speed Internet capability. The physical machine to be used will be determined by the University Dining Hall. The software developed here assumes that the user is English language literate and be able to login and enter information on the website. The speed of the student's connection will depend on the hardware used rather than characteristics of this system.

4.0 Functional Requirements Specification

This section outlines the use cases for each of the active readers separately. The student and the manager have multiple use cases in this system.

4.1 Student Use Cases

Use case: *Search Food Item*
Diagram:



Brief Description

The student accesses the Dining Order System web page, searches for a food item and adds it to the cart.

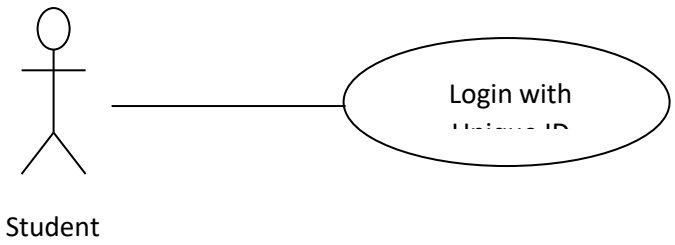
Initial Step-By-Step Description

Before this use case can be initiated, the student has already accessed the Dining Order System Website.

1. The student chooses to search dish name.
2. The system displays the choices to the student.
3. The student selects the desired food item.
4. The system presents the details of the food item to the student.

5. The student chooses to add the food item in his/her cart.
6. The system provides the login page.

Use case: *Login*
Diagram:



Brief Description

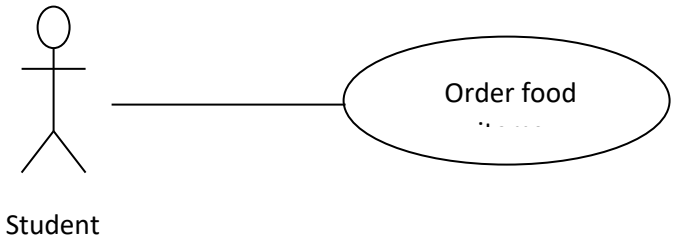
The student logs into the Dining Order System web page to order the food item added in the cart.

Initial Step-By-Step Description

Before this use case can be initiated, the student has already added the food item in his/her cart.

1. The student chooses to enter the username and password on the login page.
2. The student logs in if the correct user credentials are provided.
3. The system checks for the format of the username.
4. The system has handled the extreme scenarios of the input provided for username and password.
5. The system provides the order checkout page.

Use case: *Order food item*
Diagram:



Brief Description

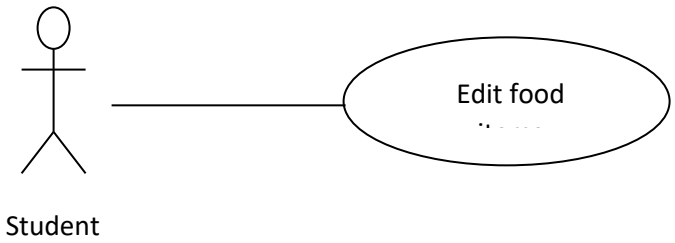
The student orders his/her food items after successful login into the Dining Order System web page.

Initial Step-By-Step Description

Before this use case can be initiated, the student has already successfully logged into the Dining Order System webpage.

1. The student has successfully logged into the Dining Order System webpage.
2. The student chooses to check out and order the food items in the cart.
3. The student confirms the order and can see the ordered food items.
4. The student can order a meal 2 weeks in advance and make changes to it till 48 hours before the pickup date.
5. The system provides the order edit page.

Use case: *Edit ordered food item*
Diagram:



Brief Description

The student can edit his/her ordered food items after successful login into the Dining Order System web page.

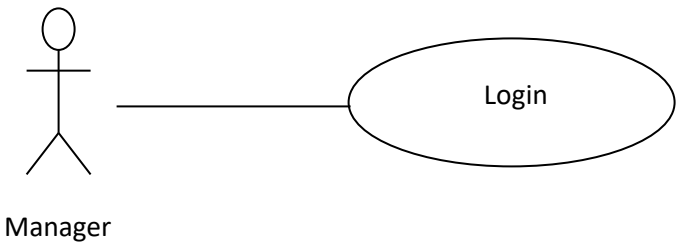
Initial Step-By-Step Description

Before this use case can be initiated, the student has already successfully logged into the Dining Order System webpage and ordered food items.

1. The student has successfully logged into the Dining Order System webpage.
2. The student can view the ordered food items list.
3. The edit option will only be provided till 48 hours before the pickup date.
4. The student chooses the changes to the food item and confirms the changes.
5. The system provides the order confirmation page.

4.2 Manager Use Cases

Use case: *Login*
Diagram:



Brief Description

The manager can add ordered food items after successful login into the Dining Order System web page.

Initial Step-By-Step Description

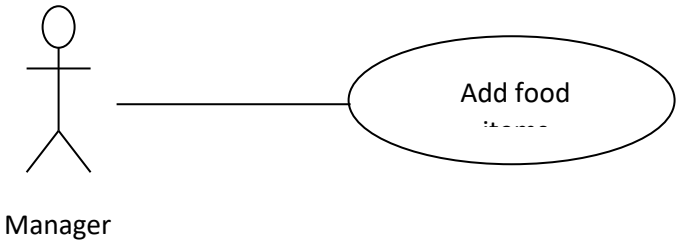
Before this use case can be initiated, the manager has already successfully logged into the Dining Order System webpage.

1. The manager enters the username and password on the login page.

2. The manager logs in if the correct user credentials are provided.
3. The system checks for the format of the username.
4. The system has handled the extreme scenarios of the input provided for username and password.
5. The system provides operations option or logout option.

Use case: *Add food items*

Diagram:



Brief Description

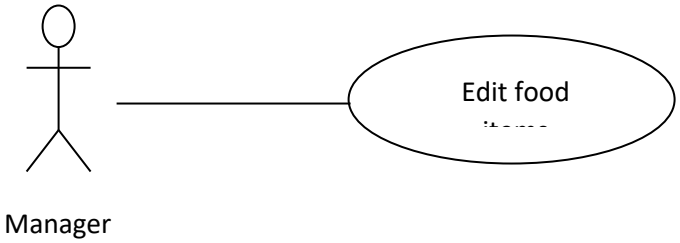
The manager can add ordered food items after successful login into the Dining Order System web page.

Initial Step-By-Step Description

Before this use case can be initiated, the manager has already successfully logged into the Dining Order System webpage.

1. The manager selects the add food items in the list on operations page.
2. The manager enters all the required details of the food item like dish name, calories, contents and adds it to the list.
3. The manager saves the added food items in the list.
4. The system has to validate for the inputs of the food items and then run the query on database to reflect the changes.
5. The system provides operations option or logout option.

Use case: *Edit food items*
Diagram:



Brief Description

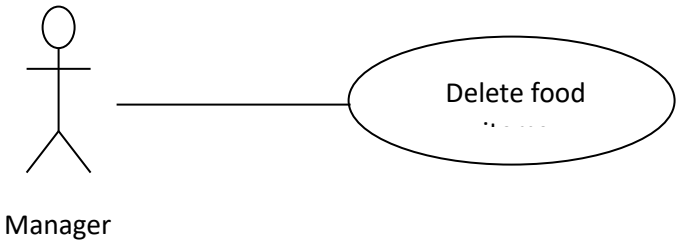
The manager can edit the food items after successful login into the Dining Order System web page.

Initial Step-By-Step Description

Before this use case can be initiated, the manager has already successfully logged into the Dining Order System webpage.

1. The manager selects the add food items in the list on operations page.
2. The manager enters all the required changes for the details of food items like calories, contents and saves it.
3. The manager cannot edit the name of food item as it is unique.
4. The system has to disable the option to edit the food item name.
5. The system validates the data and then runs the query to reflect it on the webpage.
6. The system provides operations option or logout option.

Use case: *Delete food items*
Diagram:



Brief Description

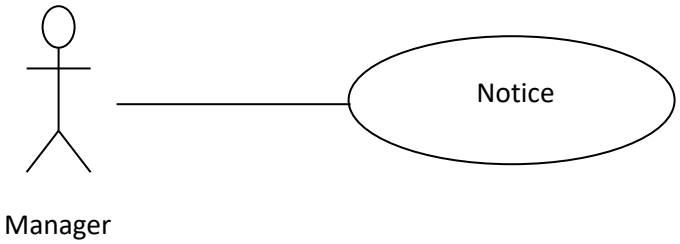
The manager can delete the food items after successful login into the Dining Order System web page.

Initial Step-By-Step Description

Before this use case can be initiated, the manager has already successfully logged into the Dining Order System webpage.

1. The manager selects the food items in the list on operations page and deletes it.
2. The manager enters the name of the food item manually to confirm the deletion operation.
3. The system verifies if the food item is available in the list and then runs the query to delete the food item.
4. The system validates the entered data and then runs the query to reflect it on the webpage.
5. The system provides operations option or logout option.

Use case: *Notice*
Diagram:



Brief Description

The manager can publish notice about the change in operation hours, changes in the food items, vacation etc. after successful login into the Dining Order System web page.

Initial Step-By-Step Description

Before this use case can be initiated, the manager has already successfully logged into the Dining Order System webpage.

1. The manager selects the publish Notice option on operations page.
2. The manager enters the notice and saves the notice.
3. The system has to verify the entered text and run the query to reflect on the webpage.
4. The system provides operations option or logout option.

1.4 Security requirements:

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.

These are required to prevent the Tampering, Elevation of Privileges, Information Disclosure and Repudiation from the STRIDE model.