

---

# **Software Requirements Specification**

**for**  
**Lower Wismer Mobile**

**Version 1.0 approved**

**Prepared by Andrew Kelly, Justin Casey, Jon Berman**

**CS-275, The Three Amigos**

**November 16<sup>th</sup>, 2020**

## **1.0. Introduction**

### ***1.1 Purpose***

This would be a very useful application for the ursinus students that use Lower Wismer. As one of those students, I can honestly say that the worst part about Lower Wismer is the waiting aspect. There are often lines at the ordering kiosk and then there is more waiting for the worker to shout out your order number. This can become very inconvenient around busy hours. Another aspect of our system is that it would limit the congestion of Lower Wismer. Especially during a global pandemic, any way that you can limit close contact with others is encouraged and beneficial.

### ***1.2 Team Dynamic***

Our group will have weekly meetings on Tuesday around 5. At these meetings, we will begin by deciding which person will work on what aspects of the project. After this, we will use the meeting time to discuss the progress we have individually made on our respective portions of the project in addition to bringing up any concerns that may arise. We will also always have a line of communication open by utilizing a GroupMe which was created by Andrew Kelly. This GroupMe will be used to quickly bring up any concerns other group members are having, call emergency meetings, or simply ask questions to clear up any ambiguities. Each member will be responsible for their own schedule. As stated previously, we will all agree on what our weekly goal should be at the meeting and it is expected that there will be some progress towards that goal by the following weeks meeting.

### ***1.3 Intended Audience***

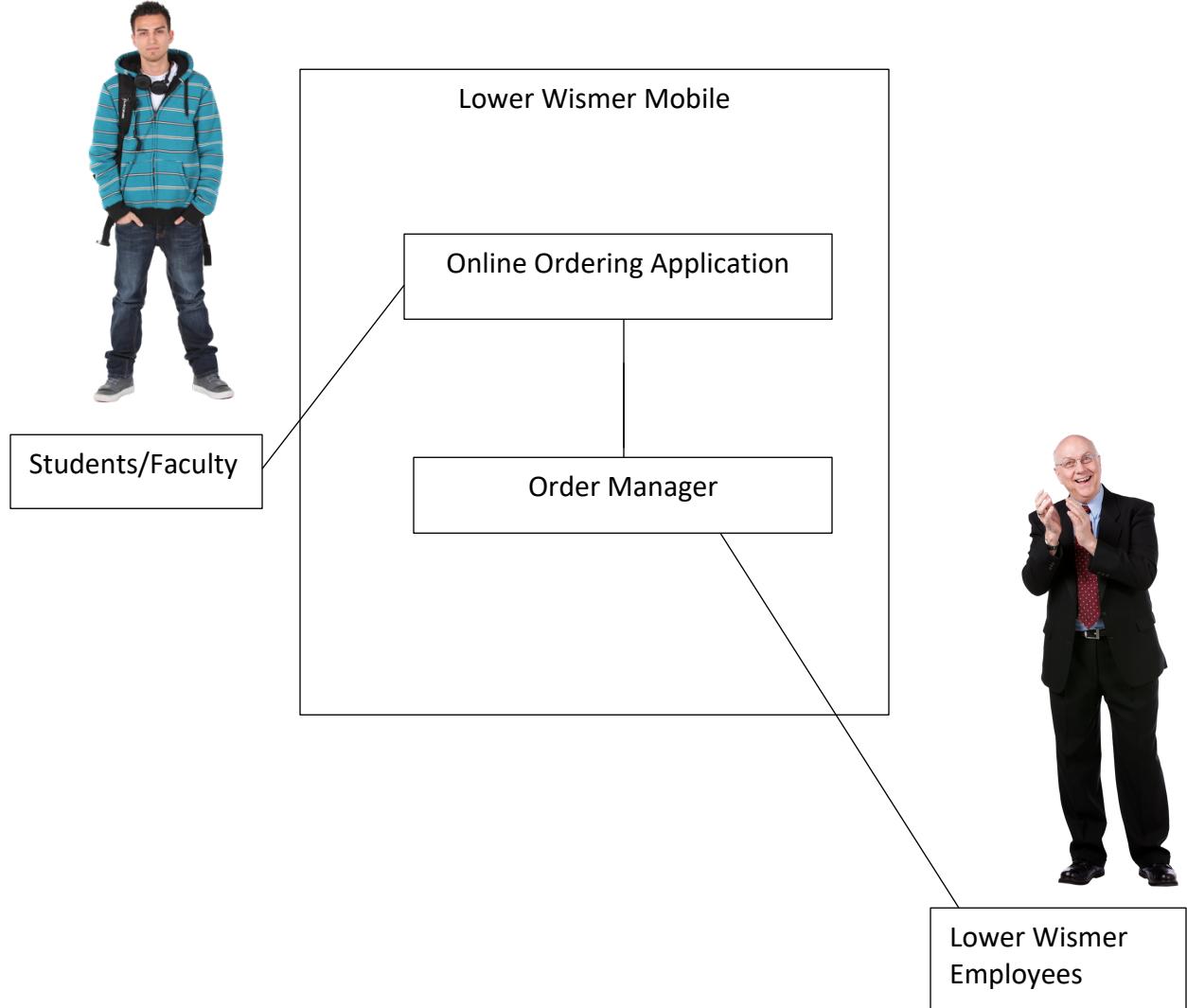
Since this project is for Lower Wismer our audience will be almost exclusively for Ursinus. students and faculty. Especially those who are in a time crunch and normally would not have time to stop at lower during its busy hours. The system will only accept dining dollars and swipes so anyone who is not a student or faculty at ursinus would not be able to order food using our system.

#### ***1.4 Product Scope***

As stated in section 2, the project will be a mobile ordering system for Lower Wismer. The program will have a web-based interface. Upon entering the application, the user will be asked to log in using their Ursinus email address. After signing in, the program will then have access to all the relevant data pertaining to this student, such as meal swipes, dining dollars, favorite meals, etc., which it reads in from a text file. The student then begins ordering the meal of their choosing. This data is then stored by the program and is sent to Lower Wismer upon the student's confirmation that the order is complete. The program retains this information within a text file and if a given meal is ordered 3 or more times by the same student, the meal will be added to a favorites list so the student can quickly order it in the future. Each time a meal is ordered, the system will charge the student an appropriate amount of dining dollars or meal swipes. The system will adjust the student's balance accordingly and then save this information to a text file which will be read the next time the student orders a meal, with meal swipes resetting to 3 at midnight every day.

## 2.0. Overall Description

### 2.1 System Environment



**Figure 1 - System Environment**

The Lower Wismer Mobile system will have 2 active users and 1 system that provides a link between the two. Students / faculty will sign in using their Ursinus credentials and use the Lower Wismer Mobile system to place their order from anywhere on the Ursinus College campus. This order information will be saved and used to determine this specific student's favorite order after repeated orders. The order will then be sent to the order manager of the respective Lower Wismer shop which the food was ordered from, where it will be displayed on a monitor to be viewed and created by the worker.

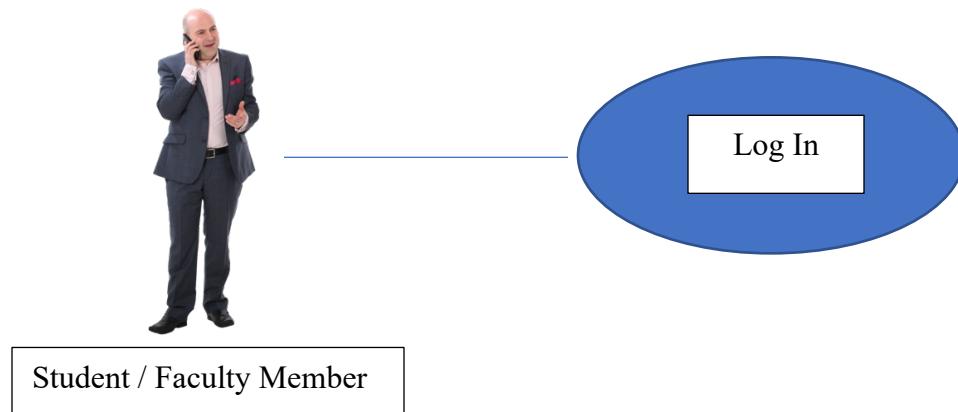
## ***2.2 Functional Requirements Specification***

This section outlines the use cases for students/faculty and employees separately. The students and faculty have the majority of the use cases.

### ***2.2.1 Student / Faculty Use Cases***

#### **Use Case 2: Log In**

**Diagram:**



#### **Brief Description**

The student or faculty access the online application, either through their laptop or smart phone, and are prompted to log in with their Ursinus credentials. This allows us to properly track the balance of each student or faculty member.

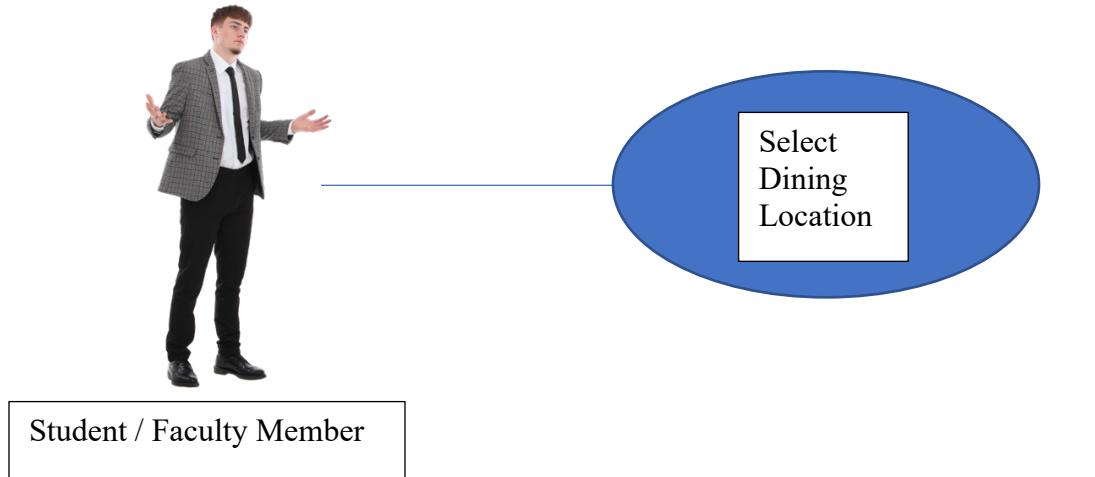
#### **Initial Step-By-Step Description**

- 1.) The student or faculty member launches the Lower Wismer Mobile application on their computer, smart phone, or other mobile device.

- 2.) The student or faculty member then inputs their Ursinus email address and password.

### Use Case 2: Select Dining Location

**Diagram:**



#### **Brief Description**

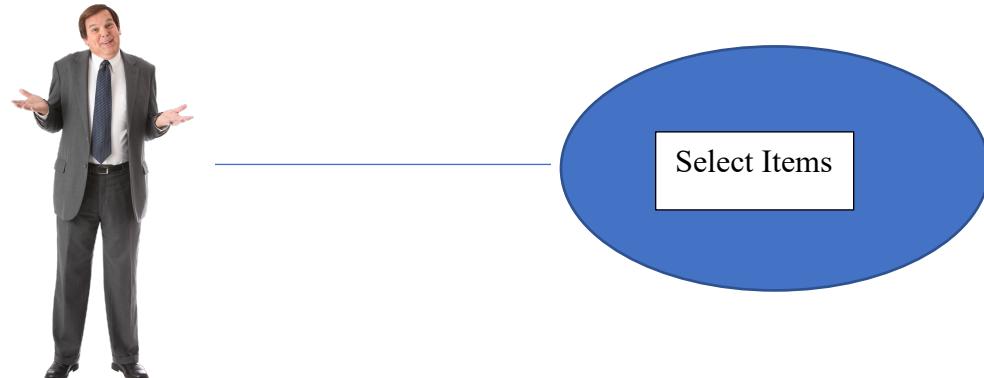
The student or faculty member now chooses which one of the four Lower Wismer dining options they wish to get their food from.

#### **Initial Step-By-Step Description**

- 1.) The user logs into their Ursinus account on the application.
- 2.) They are then presented with an interface to choose which Lower Wismer dining option they would like to purchase their food from.
- 3.) They select their dining location.

### Use Case 3: Select Items

**Diagram:**



Student / Faculty Member

### Brief Description

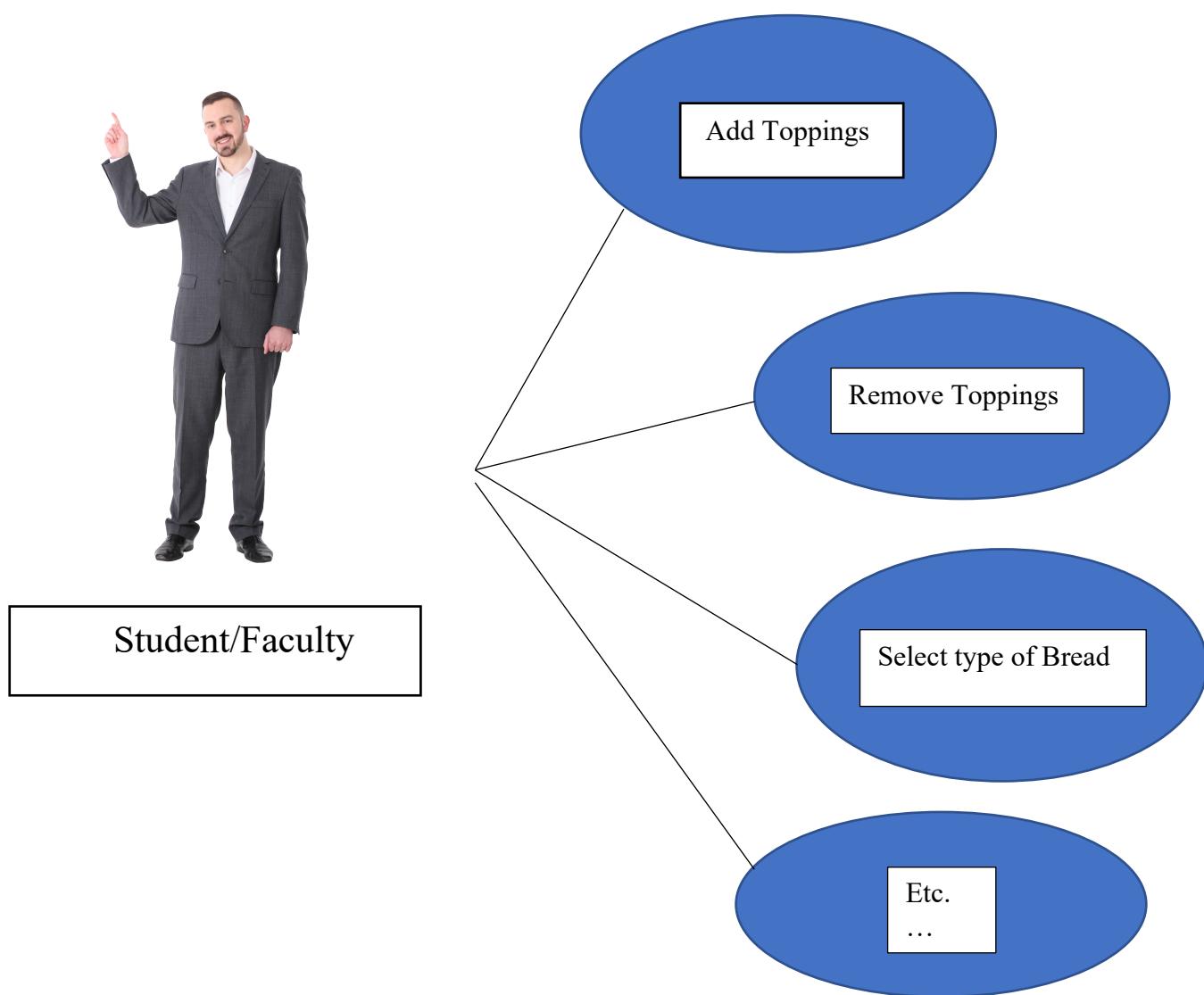
The user selects which food and / or drink options they wish to purchase from their dining option of choice.

### Initial Step-By-Step Description

- 1.) The user logs into their Lower Wismer Mobile account.
- 2.) The user selects which Lower Wismer dining option they wish to order their meal from.
- 3.) The user selects their food or drink item, such as a sandwich, burrito, coffee, etc. This item will then be able to be customized later.

### Use Case 4: Specifying Order

#### Diagram:



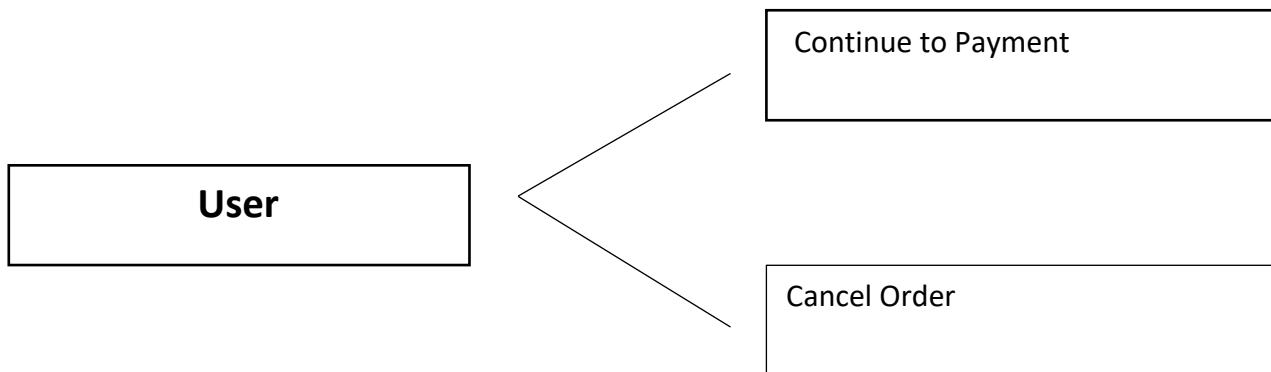
## **Brief Description**

Here the user will specify their selected item to their liking by changing the toppings and other aspects of the item. This could be type of bread if they are ordering a sandwich, or if they are getting a medium or large drink.

## **Initial Step-By-Step Introduction**

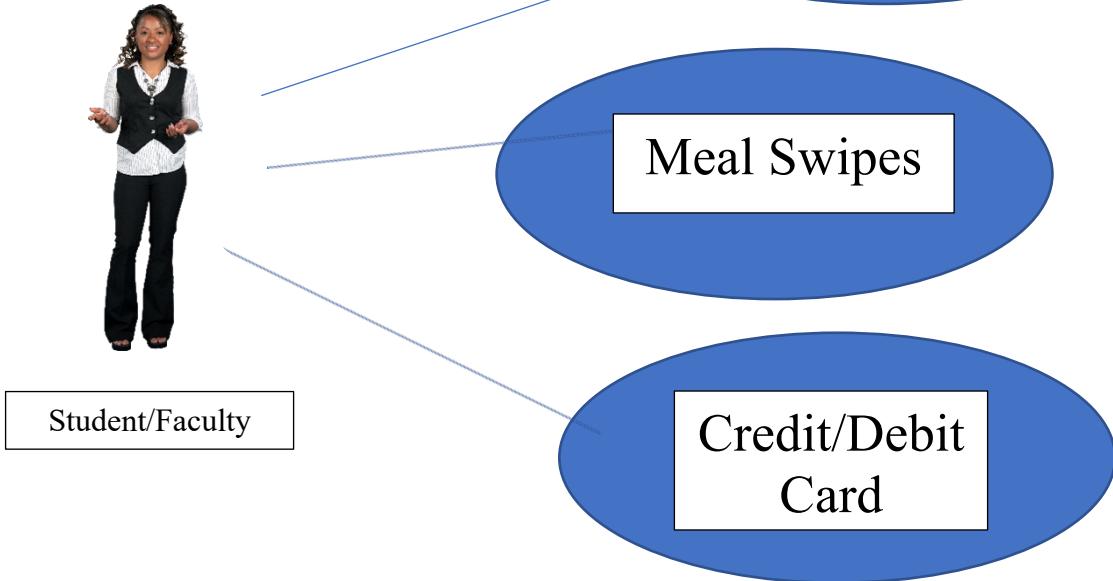
After selecting their item they will be directing to a screen where they can add or remove toppings specific to that item. Then they will be directed to another screen where they can make other choices to specify their order

## **Use Case 5: Complete Order**



**Brief Description:** After Specifying their order the user will now be able to finalize and complete the order and continue to pay or they can choose to cancel their order.

**Use Case 6: Select Payment Type**  
**Diagram:**



**Brief Description**

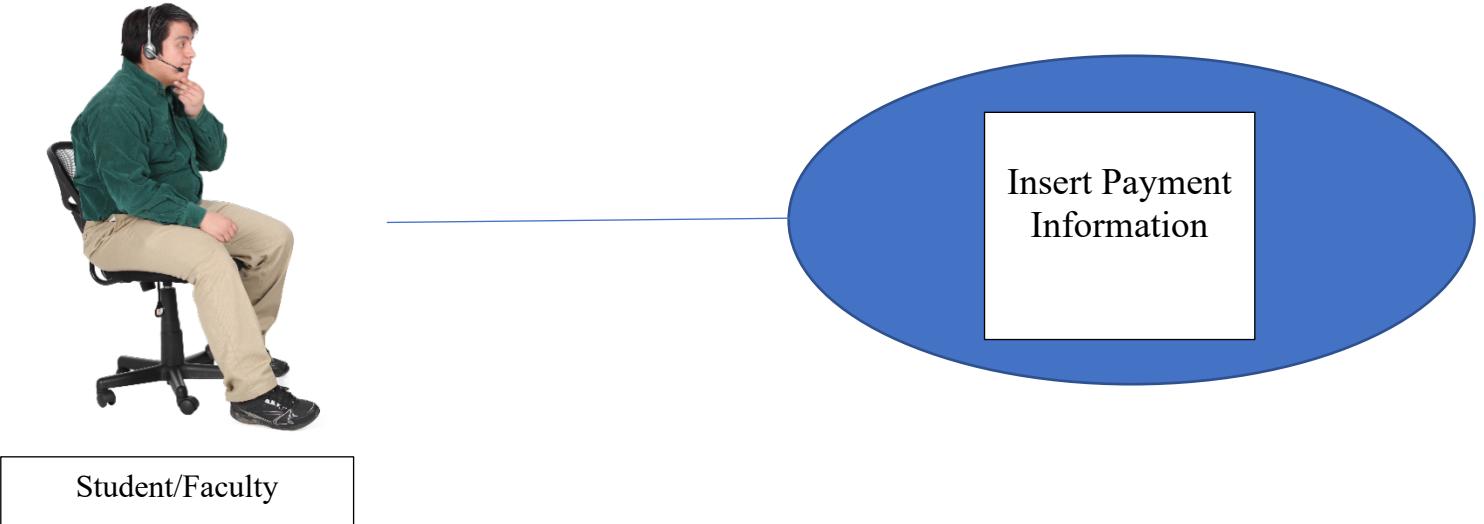
The user will be prompted with selecting their payment type upon completing their order. The options will be Dining Dollars, Meal Swipes, or some other form of electronic payment such as credit/debit card. The user selects their preferred payment type.

**Initial Step-by-step Description:**

- 1.) The user is prompted with a choice of three different payment types.
- 2.) The user selects a form of payment.

**Use Case 7: Give Payment Information**

**Diagram:**



### **Brief Description**

The user inputs their payment information into the user interface. This will either be inputting student/faculty identification or credit/debit card numbers.

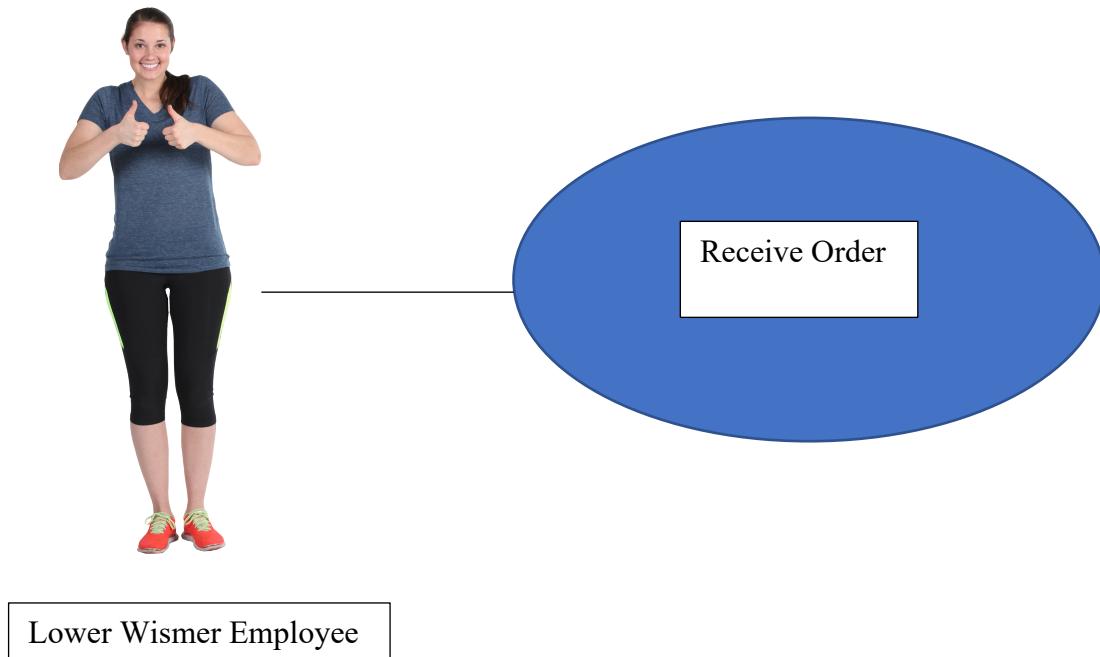
### **Initial Step-By-Step Description**

- 1.) The user is prompted to fill in information regarding their payment type.
- 2.) If the payment type is connected to the user's meal plan, they will input their student/faculty id number.
- 3.) If the payment type is credit/debit card, the user will input the card number, expiration date, and CVV.

#### ***2.2.2 Lower Wismer Employee Use Case***

##### **Use Case 8: Receive Order**

**Diagram:**



## **Brief Introduction**

Upon completion of a student or faculty member's order, it will be sent to the order manager, which will then display the order on a monitor to be viewed by a Lower Wismer employee so they can prepare the order.

## **Initial Step-By-Step Introduction**

- 1.) Student or Faculty member utilizes the online ordering application to complete their order.
- 2.) The order is sent to the order manager.
- 3.) The order manager displays the order on a monitor so it can be viewed by employees.

### ***2.3 User Classes***

The user is expected to have both access and literacy in either a computer or a smart phone. The application will be able to be run on either a smart phone app or a web browser. The user will be expected to have access to the Ursinus College wireless internet. The application requires that the user will be on or around the campus for an order to be placed. The user will be expected to have some form of digital payment (i.e. dining dollars, meal swipes, credit card, etc.). The user will be expected to understand how to interact with a basic user interface. The interface will be extremely clear to guide the user in the direction of their desired order. Once the order has been placed, it will be sent to the respective lower wismer food service where the lower wismer employee will be expected to translate it and construct the proper order. Considering lower wismer already has a similar ordering system, the translation of the employee should not be any different from the way that they read the current orders. The most important users will be

the Ursinus students as they will be most frequently using this application. Since they are college students, it can be anticipated that they are accepting of change and welcome to our new ordering application.

#### ***2.4 Operating Environment***

This application will be accessible on both apple and android smart phones. In addition to smart phones, there will be a website that can be run on any web browser. The application will run in most environments, but it will not be able to be used unless connection with an Ursinus College wireless network is secured.

#### ***2.5 User Documentation***

No user documentation is required for our system to function properly.

#### ***2.6 Assumptions and Dependencies***

This program is building off of and working in tandem with the current Lower Wismer ordering system. Thus, we are assuming that that system is working correctly for our system to operate. In order for the system to work, we are depending on the information from the users order to reach the employees of Lower Wismer, so they are capable of fulfilling the order.