Design Review Report

Name/description of Infrastructure System:	**Project Title:**
Design Review Report	Online Shopping System

What was reviewed:

The team designed multiple diagrams to understand and visualize the set steps that the team must take to build the restaurant's website successfully. Class, sequence, and state diagrams were the three top diagrams made. The team designed two class diagrams. Both class diagrams were different variations of the same conceptual models. Next, the group organized and reviewed the specific use cases that form the website's functionality in the conceptual diagrams. The group agreed that four prominent use cases help the website function: product, shopping cart, payment details, and payment. The product's use case aggregates into the shopping cart, and the payment is composed of the payment details and the shopping cart. Then the group reviewed and designed two sequence diagrams, a shopping cart diagram and a complete website diagram. Next, we discussed the following objects we needed to interact with our end-user(actor) for the website to operate successfully. The five objects we added were the following: system, product, shopping cart payment details, and payment. The sequence would start with the actor interacting with the system, adding item(s), increasing item quantity(s), decreasing item quantity(s), or removing the item(s). Next, the system interacts with the product; the product is updated into the shopping cart, the shopping cart is displayed to the system, and the system displays the shopping cart to the actor. Once the actor interacts with the shopping cart through the system and finalizes the items within the shopping cart, the system communicates with the shopping cart, and the shopping cart is added to the payment object. Only then does the payment object create a payment details object. The payment object then sends the payment information template to the system; the system displays the template to the end-user, and the end-user fills in and completes the payment info on the system. Finally, the payment details object receives the completed payment information from the system. The payment details class then verifies the information, updates the payment class and the payment objects export all the information to the database. The last diagram reviewed and constructed by the team was the state diagram of the online shopping cart, another version of visualizing the site functionality. By studying and making the three main designs, the team understood the steps needed to be taken to successfully build the online website and the possible errors that can occur when creating the website.

#	**Findings**	**Impact**
1	End-users payment method is not valid or has Insufficient funds.	The payment detail object will validate the payment information before updating the payment. The payment method will not be processed if it is not valid or has insufficient funds. Then the payment detail object will reject the payment method and send an error to the system, and the system will display the error to the end-user.

2	End user finalizes cart with nothing added to it	If the end-user tries to submit the shopping cart, displayed by the system, to the payment class, the system will display an error to the end-user.
3	End-user submits payment details without adding all required payment information	The payment details object will validate the imputed information before submitting the form to the payment object. If the end-user attempts to submit the payment details without filling in all the required information to finalize the report successfully, then the payment details object will send an error to the system, and the system will display the error to the end-user.

Conclusion:
To summarize, contriving and reviewing the following designs above allowed the group to understand the possible outcomes and potential errors better when the end-users operate the restaurants' website. The group discovered three possible errors that can occur when using the website. Understanding the following errors allowed the team to program a safety net around the found errors that will allow for the website's functionality to remain efficient.

Reviewers
Okiki Ojo
John Howe
Johnathon Tannous