**COM-430(Software Engineering) Project-Group 11- Fuz Inc**

Brianna Bent, Jefferson Levausser, Rashad Almasoud

COM430-CA01

Saint Leo University

Dr. Joshua Adams

May 3, 2022.

https://github.com/briannabent/Group11-Fuz.git

Table Of Contents

Introduction 2

Production Features 3

User Cases 5

Scope and Development Plan 6

Github Configurations, Commits and Branches 7

Project Stack and Structure 7

Software Architecture and Design 8

Client Server Architecture 9

Food Inventory Architecture 11

Appendix 13

# 1.0 - Introduction

The goal of this project is to make a software system for a local catering company. It is a food ordering system for employees and management to keep track of the inventory of the business.

1.1 - Group Details

Team Name: Team Eleven

Team Members: Jefferson LeVasseur, Rashid Almasoud, Brianna Bent

## 1.2 - Initial Scenario

A local catering company that would like to keep better track of its inventory. At such a small-scale restaurant that does not want to pay the fees for inventory software that is for the level of business that it operates at. This company may also prefer a lightweight program that specifically meets their business needs and can help in maintaining stock count.

# 2.0 - Product Features

## 2.1 Initial Features

The initial system created is made to fulfill the basic features of the applications.

The system will be able to keep stock of all products and their availability by total count. If a product reaches a stock count of “0”, then the product will be unavailable for purchase. Additionally, it will be able to automatically update low stock products by setting up “Restock Days”. Low-stock products can be preemptively updated and prompt administrative users to confirm or deny the option to reorder products necessary to keep in stock and will update if a product delivery has arrived. An additional feature planned to be in The System will be able to monitor which employee completes or takes an order in the system, this allows the restaurant to know who was in charge of which order and if they have any problems later down the line, they will know how to track the entry.

Admin Features:

1. Access stock inventory to keep track of supplies
2. View and edit inventory count for the company
3. Ability to make stock orders from the suppliers and cancel orders if necessary

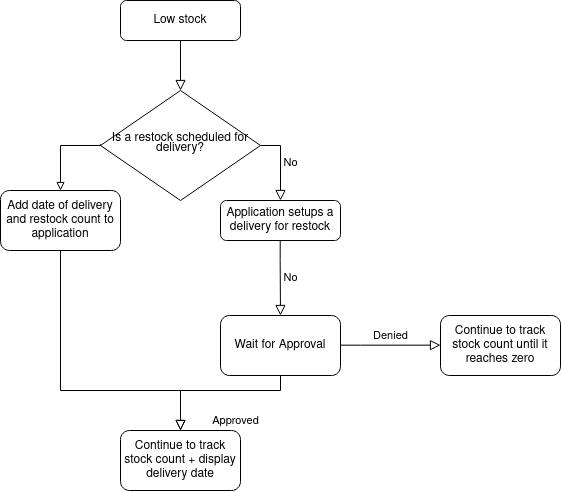
## 2.2 - User Stories

When a customer would like to make an order, if their order is out of stock, they do not receive a notification, instead, they get a phone call shortly after telling them their choice is not available and they need to pick something else. An employee is managing the number of items that are available for purchase but is unable to know how much will be available when a new shipment arrives. When an employee completes an order for the business, and something goes wrong, not in the exact moment but a few days/weeks/months later it will be important to have a software system that monitors who messed up. For example, if a business has a limit of sandwiches to sell per customer to help inventory changes, if an employee sells more than that limit that is established it is important to track who did it and how many sandwiches were sold over the limit, in order to place an order for the inventory for the business.

User Tasks

1. User places an order
2. User waits for order confirmation
3. Print invoice
4. Sales Updates
5. Inventory Updates

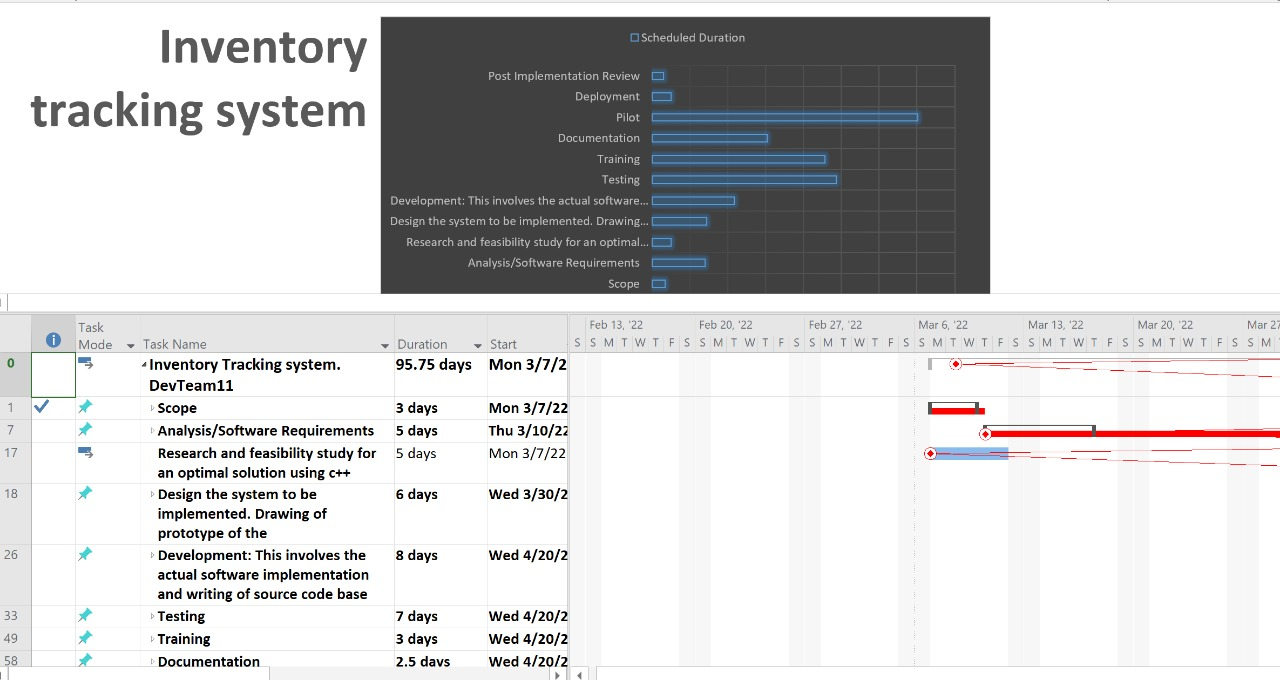
## 2.3 - User Cases



This program will be used for employees and managers of the business to have a system which provides stock inventory for their materials to make the food. It is important for a food business to have a proper software that tracks inventory of what is available and what needs to be ordered for future consumption. If a business is running low on a certain item, it is important to be alerted by the system to prevent losing the sale because they did not have the inventory to make the food.

# 3.0 Scope and Development Plan

## 3.1 Project Schedule



## 3.2 Github Configuration, Commits and Branches

GitHub will be used for collaboration amongst the team members. Each member will handle a given task implementation and collaborate with others by pushing their source code to the repository. GitHub will also be used as a version control during implementation. We will adopt a feature branch workflow where features will be developed in a dedicated staging branch where multiple developers will work on a feature without disrupting the main codebase. Each commit should focus on implementing effective changes to the program and each message should describe what was improved upon in the program. Commits comments will be structured with a subject and a body separated by a new line. The body should explain how and why a commitment was implemented. The subject should be limited to a maximum of 50 characters and briefly describe the implementation made. In order for all members to adhere to the rules above a brief training session will be conducted to ensure uniformity in our work.

## 3.3 Project Stack and Structure (Code Rules, Security, License)

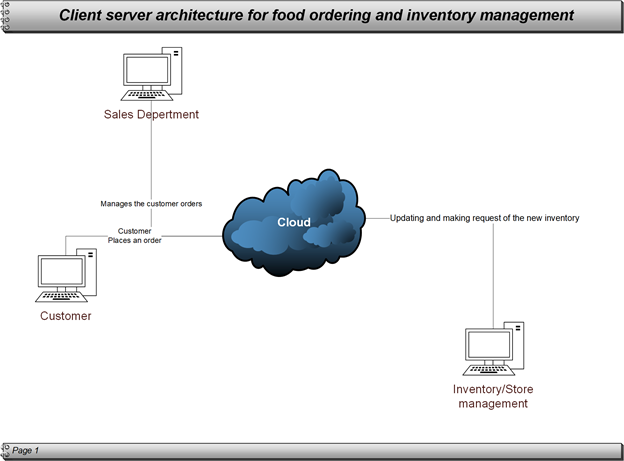
The System will be structured as a standard web application, with a database at the lowest end, a program written in PHP to communicate between the database and the web page, and a front-end web page for a user to interact with. However, this may breed major concerns of security. The database and the program that will interact with the database will most likely need the most security, as databases hold the most value in the information that they carry. Ensuring that unauthorized users cannot access vital information will be part of the process

However, additional research on the optimal implementation of these features will take up most of the time developing the system. We will set up specific goals for our system to achieve as we build upon our product. These align with the weekly goals we have set up in Section 3.1. While not new problems may appear as time goes on, documenting what works and what needs to be fixed will keep us up to date on our goals.

# 4.0 Software Architecture and Design

The architecture depicts the high-level interrelationship of various components in the inventory system to achieve a specific function forming the entire blueprint of the food ordering and inventory management system. At the architectural level, we are focusing on the main tradeoffs that largely affect the performance, quality, and maintainability of the inventory system.

On the other hand, the design focuses on the actual conceptualization of the inventory system requirements that will be used to create a software design specification document. In our team's Inventory tracking system, we will focus on the high-level architectural design of the system to clearly map out the components and their functional relationships.



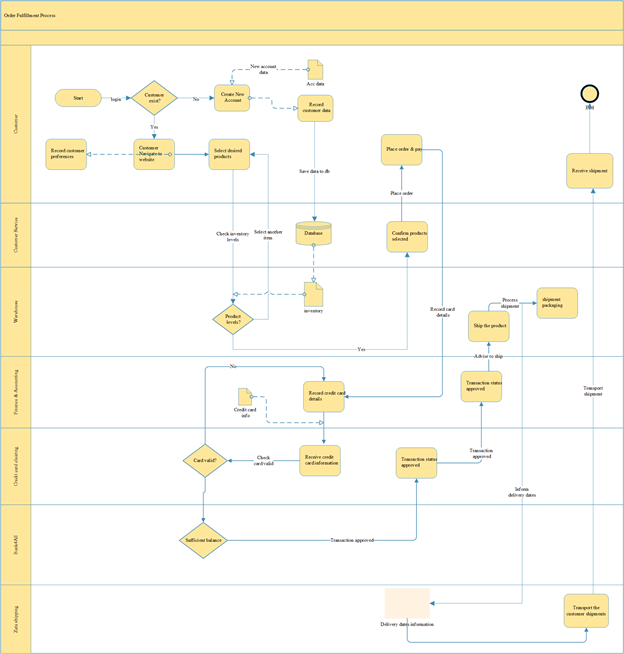
## 4.1 Client-server architecture for the food ordering and inventory management system

From the diagram above, we have 3 main users of the system who will assist the catering company to achieve the set objectives by implementing the food ordering and inventory tracking system. The process involves customers accessing the ordering system via the computer devices which are connected to the cloud. From the customer application system, they will be able to raise the orders and send them to the cafeteria company.

The Sales agents and managers associated with the cafeteria company will receive and review orders for fulfillment. Once an order is reviewed and approved, it is then sent to the production for processing and delivery.

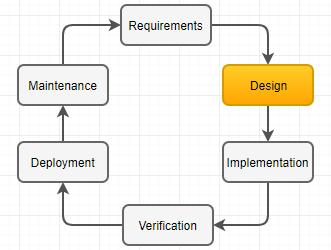
Finally, we have the inventory and store managers who are able to access the orders raised at the production department for fulfillment.

An in-depth overview of the food ordering and inventory tracking fulfillments is outlined through a use case diagram.

From the diagram above, we can follow the entire process from the customer order to payment and finally shipment of the entire order. This highlights all the users and their functional needs to operate the system.

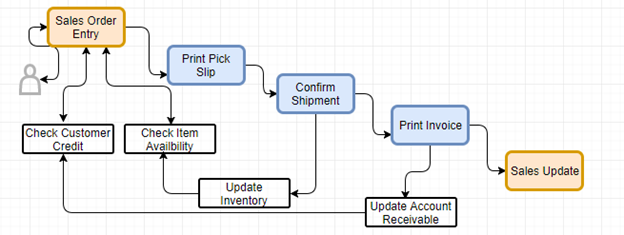
## 4.2 Food ordering and inventory tracking software design.

To actualize this software implementation, we have adopted the SDLC life cycle where design is crucial and comes as the third stage in the SDLC as shown below.



This stage has assisted us in converting the software requirement document into a design specification document that will be used in the next phase of software development.

This project aims to meet the gathered requirements in the area of sales management, real-time inventory updates, and sales order fulfillment via the system.



The above is the high-level design overview of various components and stages of fulfilling an order from a customer by design.

# 

# Appendix

Brianna Bent completed the PowerPoint presentation and the Github repository. I assisted in the creation of the User stories, tasks, and features.

Rashid Almasoud completed the scope project schedule, Software architecture, and design, and the source code.

Jefferson LeVasseur created the structure for the report and refined both the report and PowerPoint.