CS3343 Software Design

Project Plan

Group 12

Name	Student ID	Title
CHENG Yin		Project Manager
SONG Tao		Assistant Project Manager
LUO Peiyuan		Programmer
ZHOU Junchen		Programmer
DU Wenxi		Tester
FENG Yong		Tester

Tables of content

1. Project Description

1.1 Abstract

Since the COVID-19 pandemic is alleviating, people would go out to dine more often. Therefore, the project is to develop a food court management system named GoGo Eat for Food Court with many dining windows to cater to customers with different flavors and dining options, including dine-in, queue, and reserve, and to satisfy the need for merchants and admin to manage their business.

1.2 Objectives

This project is designated for restaurant merchants and admins who want to manage the food court's operations and customers who wish to reserve a table or dine in and finally order food and make payment through the system.

The Restaurant Management System GoGo Eat consists of an online queuing service to allow customers to estimate the waiting time and queue in the application. Since the customers might feel tired and bored while queuing physically and get worried that they might miss their turns if not having attention to the queue number written on a small piece of paper, the online queuing service would tackle the problem of getting worried about missing turns and prevent potential crowd gathering outside the restaurant.

Moreover, GoGo Eat is an online ordering service for customers to dine in. Customers can pre-order to save the ordering time and the waiting time before the food arrives.

As the demand for non-cash payments dramatically increases with the emergence of technology. GoGo Eat consists of cash and non-cash payment methods, reducing social contacts for dine-in customers or the need for remote payment for customers.

1.3 Scope

The Restaurant Management System consists of 4 main modules:

1. Customer Module
    - Login & Register
    - Dine-in (including Online Queuing)
    - Reservation
    - Food Ordering
2. Payment
    - Pay by Cash
    - Pay by Online Payment Methods
3. Merchant Module
    - Login & Register
    - Edit Restaurant Information
    - Edit Dish Information
    - Check Customers' Orders
    - Cash Payment for Customer
4. Admin Module
    - Set Food Court's opening hours
    - Check Customers' orders
    - Check Customers' reservation status
    - Add / Remove Restaurants
    - Add / Remove Tables
5. Helper Functions:
    - Tables Management
    - Time Management

Each Module is in charge of dealing with the specific actors who interact with the system.

1.4 Constraints

- Scope

It is a deliverable food court management system including a customer module, payment module, merchant module, and admin module that is the project scope, aiming to provide a platform for customer booking and ordering, merchants offering dishes and setting prices, and an admin managing the whole food court and serving customers. There are login and register, online queue, and payment functions in the customer module; login and register, food ordering in the merchant module; login and register, and checkout functions in the admin module.

- Time

The deadline for the project is November 18, the last time to achieve a deliverable project. The project would take about ten weeks, one for team building, one for use case diagram, one for class diagram, one for project plan documentation, two for programming, three for testing and debugging, and finally, one for documentation and flexibility. A time management plan is set to track time spent on each task and process in the project.

- Cost

The overall budget is $30,000, which is lower than the market rates of around $39,000, excluding the cost of developing tools' license, which the client provides. Comparing the historical data with updated, relevant data, developing a system in java is becoming more convenient and lower cost. The hourly cost should be controlled at around $9,000.

1.5 Stakeholders

Related stakeholders of the software:

Stakeholders	Description
Customers	The customer is the primary user of GoGo Eat, operations ranging from booking seats and ordering dishes through the customer module and paying the bill through the payment module.
Restaurant Merchants	Merchants are one the users of the system. They use the merchant module in the deliverable product system to edit their dishes and price in the system and use the payment module to get paid from customers.
Restaurant Admins	The restaurant admin is the admin of the food court for seat allocation and customer service, and they are also users of the system.
Project Manager	The project manager is an internal and the most important stakeholder, managing the project, balancing the project scope, cost, and time, and aiming to achieve a successful project.
Development Team	The development team is the team of developers of the whole restaurant management system, including the "programmer" for writing program code and "develop and test" for test case writing and program development.

2. Project Team Organization
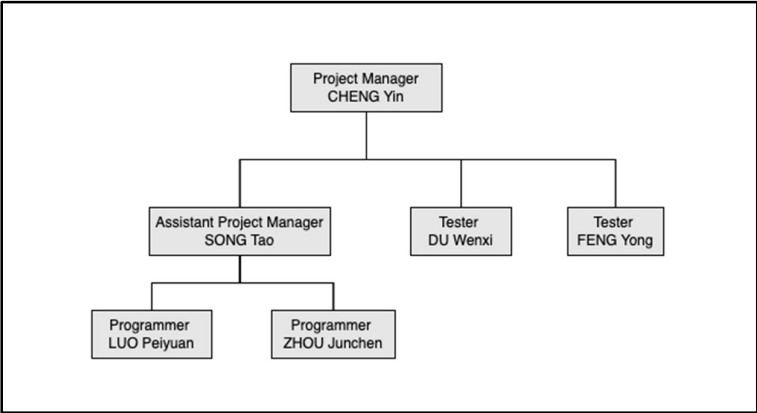
Name	Title	Job Description
CHENG Yin	Project Manager	Documentation, Management, Design, Development
SONG Tao	Assistant Project Manager	Documentation, Management, Design, Development
LUO Peiyuan	Programmer	Documentation, Design, Development
ZHOU Junchen	Programmer	Documentation, Design, Development
DU Wenxi	Development and Test	Documentation, Design, Testing
FENG Yong	Development and Test	Documentation, Design, Testing

3. Methodology

We adopted the agile development method in our project. Agile development is an iterative software-development methodology. In the Agile development method, the construction of a software project can be divided into multiple subparts, which are related to each other but can also be run independently and completed separately.
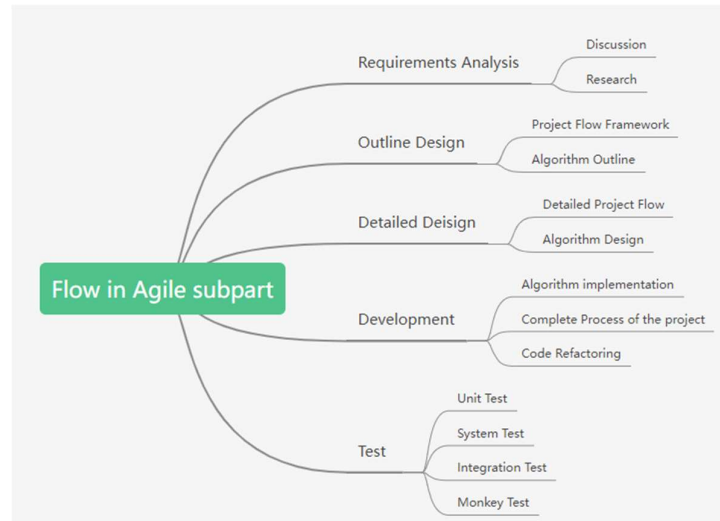
In the Requirements Analysis, we discussed selecting a topic and did some research to determine the requirements of the topic.

After obtaining the requirements, we divided the whole part into two main parts. One is project flow, and the other is algorithms. We did the framework of project flow and algorithm at first and further improved the details of project flow and algorithm design, which played an important role in the programming.

Then we moved into the development phase. In this stage, we used Java code to implement various functions and detailed algorithms. In this process, if we found that there was duplicate code or two over-coupled classes, we would do code refactoring to make the software more comprehensible.

And then, we would hand over the code to testers to detect bugs in the software. Our testers would write test code to implement the unit tests, system tests, and so on, allowing bugs in the software to be found. And our programmers will debug the code to ensure the smooth functioning of the software. Above is the subpart of the agile development method, which was iterated multiple times throughout the project production process. With the Agile development method, we make the software development go smoothly.

4. **Development Tools**
   - Visual Paradigm
     - Class Diagram
     - Use Case Diagram
   - Process on
     - Class Diagram
   - Visual Studio Code
     - Programming
   - Eclipse IDE
     - Testing
     - Debugging
     - Generating Javadoc
   - Google Sheet
     - Activity Log
   - Google Doc
     - Project Plan
     - Test Report
     - Bug Report
   - Google Slides
     - Powerpoint for presentation
   - Microsoft Visio
     - Work Breakdown Structure
   - Miro
     - Work Breakdown Structure
   - TeamGantt
     - Gantt Chart
   - Dropbox Paper
     - Bug Fix Recording

5. Configuration Management

5.1 Change Control

For program flow change, it will not be considered upon project execution. If there are any program flow changes before project execution, a full change report with adequate reasons for the change must be submitted to all stakeholders for review.

For bug fixes, all the change requests will be documented in the "Bug Fix Record" document. Under full evaluation, if the bug is approved for change, the record will be logged to Bug Report, and the programming team will start to work on it.

5.2 Version Control

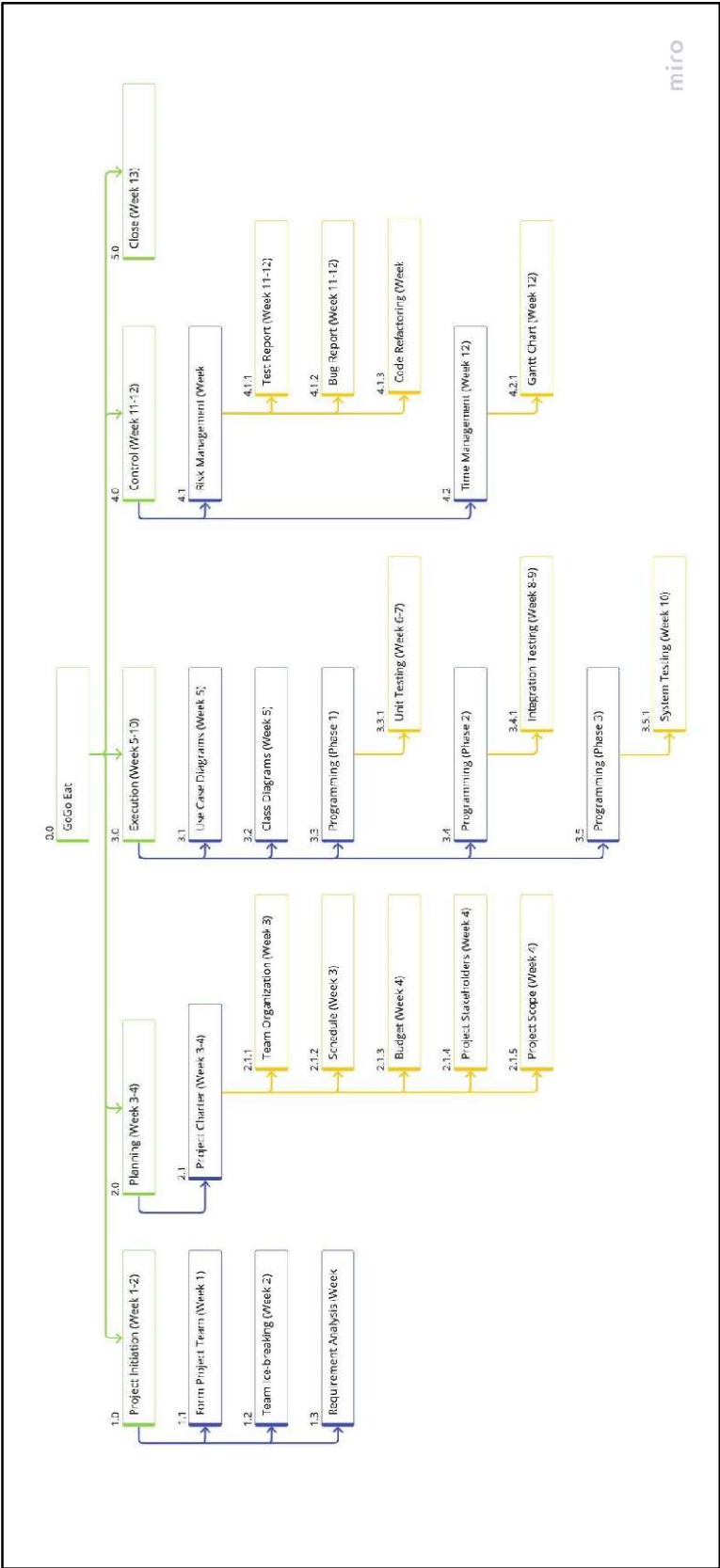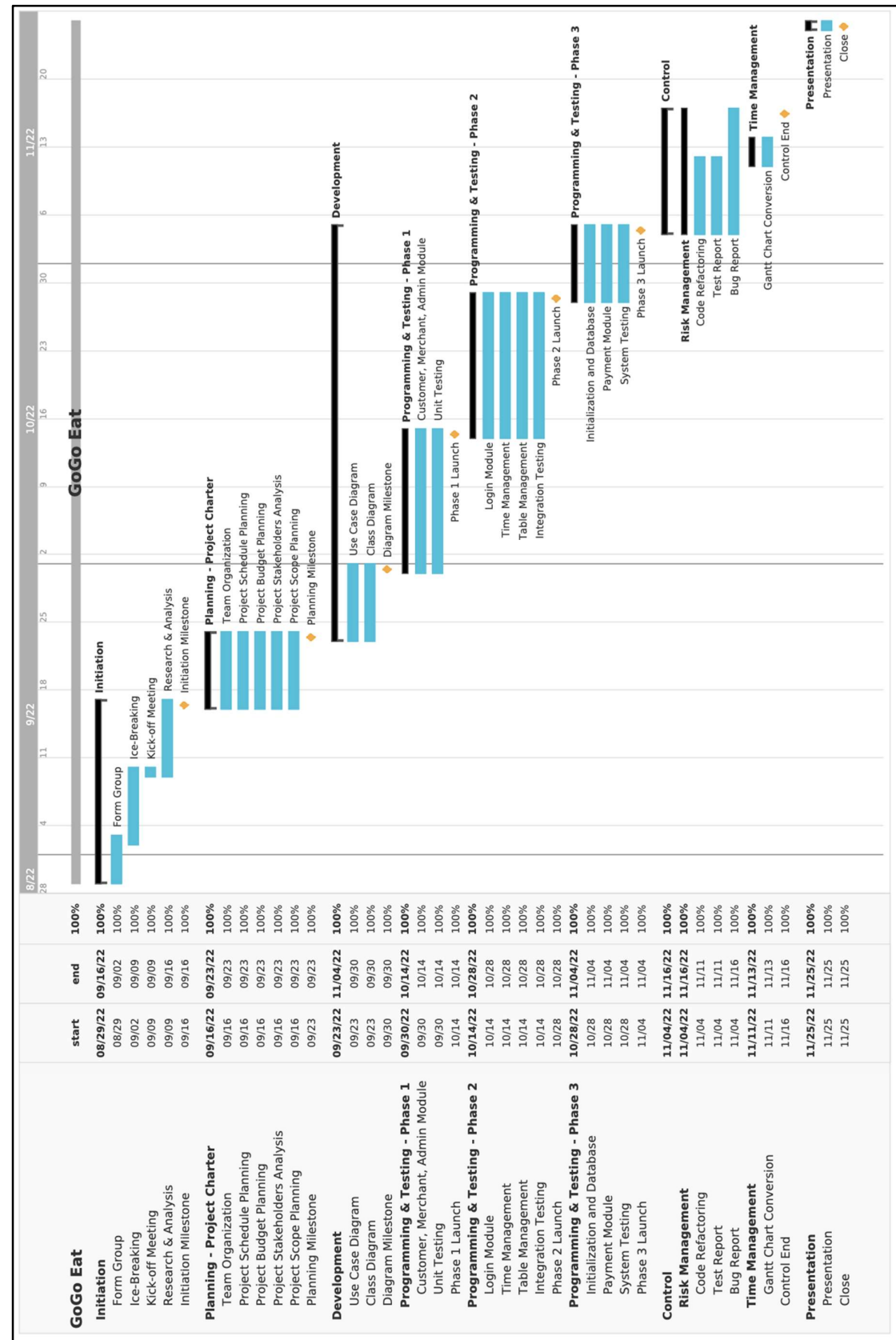The below repositories are established in GitHub:
- LoginModuleNotIncluded
- Include-Login
- Merge-All
- Refactor
- FinalChange

Each programmer works on their assigned classes. If there are common classes, pull and push should be done every time a method is modified, and a time-stamped log message should be made. Working partners should also be notified of the change.

6. Work Breakdown Structure

7. Project Team Gantt Chart

GoGo Eat	start	end	100%
Initiation	**08/29/22**	**09/16/22**	**100%**
Form Group	08/29	09/02	100%
Ice-Breaking	09/02	09/09	100%
Kick-off Meeting	09/09	09/09	100%
Research & Analysis	09/09	09/16	100%
Initiation Milestone	09/16	09/16	100%
Planning - Project Charter	**09/16/22**	**09/23/22**	**100%**
Team Organization	09/16	09/23	100%
Project Schedule Planning	09/16	09/23	100%
Project Budget Planning	09/16	09/23	100%
Project Stakeholders Analysis	09/16	09/23	100%
Project Scope Planning	09/16	09/23	100%
Planning Milestone	09/23	09/23	100%
Development	**09/23/22**	**11/04/22**	**100%**
Use Case Diagram	09/23	09/30	100%
Class Diagram	09/23	09/30	100%
Diagram Milestone	09/30	09/30	100%
Programming & Testing - Phase 1	**09/30/22**	**10/14/22**	**100%**
Customer, Merchant, Admin Module	09/30	10/14	100%
Unit Testing	09/30	10/14	100%
Phase 1 Launch	10/14	10/14	100%
Programming & Testing - Phase 2	**10/14/22**	**10/28/22**	**100%**
Login Module	10/14	10/28	100%
Time Management	10/14	10/28	100%
Table Management	10/14	10/28	100%
Integration Testing	10/14	10/28	100%
Phase 2 Launch	10/28	10/28	100%
Programming & Testing - Phase 3	**10/28/22**	**11/04/22**	**100%**
Initialization and Database	10/28	11/04	100%
Payment Module	10/28	11/04	100%
System Testing	10/28	11/04	100%
Phase 3 Launch	11/04	11/04	100%
Control	**11/04/22**	**11/16/22**	**100%**
Risk Management	**11/04/22**	**11/16/22**	**100%**
Code Refactoring	11/04	11/11	100%
Test Report	11/04	11/11	100%
Bug Report	11/04	11/16	100%
Time Management	**11/11/22**	**11/13/22**	**100%**
Gantt Chart Conversion	11/11	11/13	100%
Control End	11/16	11/16	100%
Presentation	**11/25/22**	**11/25/22**	**100%**
Presentation	11/25	11/25	100%
Close	11/25	11/25	100%

8. Future Planning

In the future, we would like to implement functions to enhance the application:

1. Implement Graphical User Interface
   Instead of a command line interface, we plan to implement the graphical user
   interface for all modules, including the admin interface, merchant interface, and
   customer interface, for better visualization and user experience

2. Enhancing the algorithms to improve running time
   The current table management and reservation may not support many tables and may
   even cause lag during usage.

3. Implement Multi-threaded design
   Due to the triple constraints of time, scope, and cost, we could not implement a multi-
   threaded design for our program. Therefore, we plan to enhance our program to meet
   the realistic needs