NANYANG TECHNOLOGICAL UNIVERSITY

CZ3003 - Software System Analysis & Design

Lab 3 Deliverables

Project Name: Food Wars

Group Name: Team 1

Lab group: TDDP3

Date of Submission: 17 October 2021

Group Member	Matric No.
David Tay Ang Peng	U1910603L
Grace Ong Yong Han	U1721575H
Jordon Kho Junyang	U1920297F
Lim Wei Rong	U1921791D
Ryan Tan Yu Xiang	U1922774F
Joy Cheng Zhaoyi	U1922716L
Tang Hoong Jing	U1721417E
Guo Wan Yao	U1822530E
Ng Wee Hau, Zaphyr	U1822044D
Lee Kai Jie, John	U1921862J
Chio Ting Kiat	U1720465K

Tables Of Content

Static Model – Component Diagram

For Foodwars, the team has decided to adopt the **Client-Server Architecture** after carefully deliberating the trade-offs between the various models. The team has then broken down the game into smaller subsystems via **decompositional design** and derived the **various homogeneous, single-responsibility components**. The following is the static model of our system, shown using the **Component Diagram** in Figure 1.

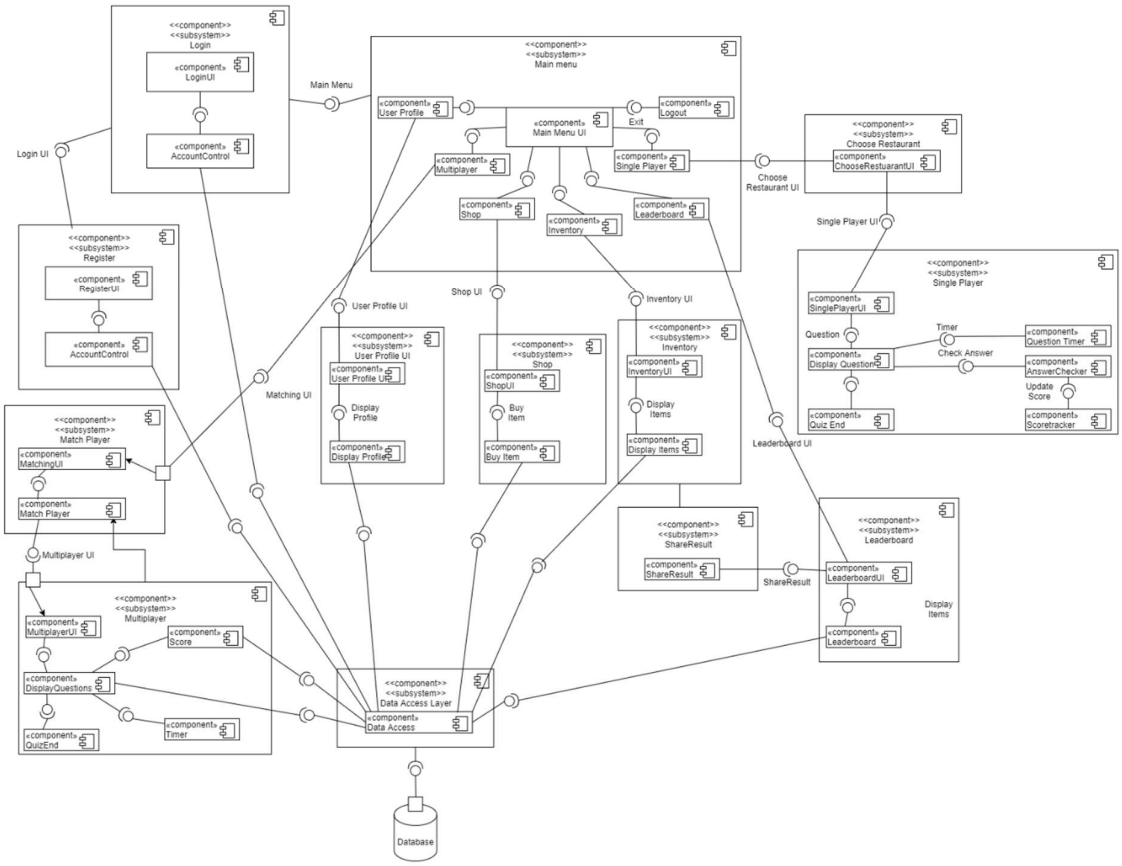

Figure 1 - Component Diagram

Dynamic Models – Communication Diagrams

To showcase the communications between the various subsystem components, the team will be using **Communication Diagrams** to illustrate the process for some of the **most important use cases for Foodwars**, which are:

- Starting a new level after finishing the current level
- User visits the shop and buys something

Communication Diagram 1

For the following diagram, the communications between the relevant subsystem components when a **user starts a new level after finishing the current level** are illustrated.

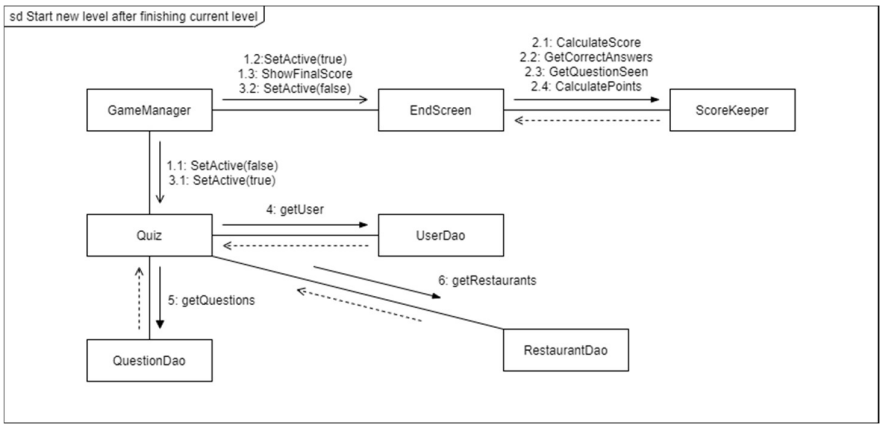

Figure 2 - Communication Diagram 1 (Starting New Level)

Communication Diagram 2

For the following diagram, the communications between the relevant subsystem components when a user visits the shop and buys an item are illustrated.

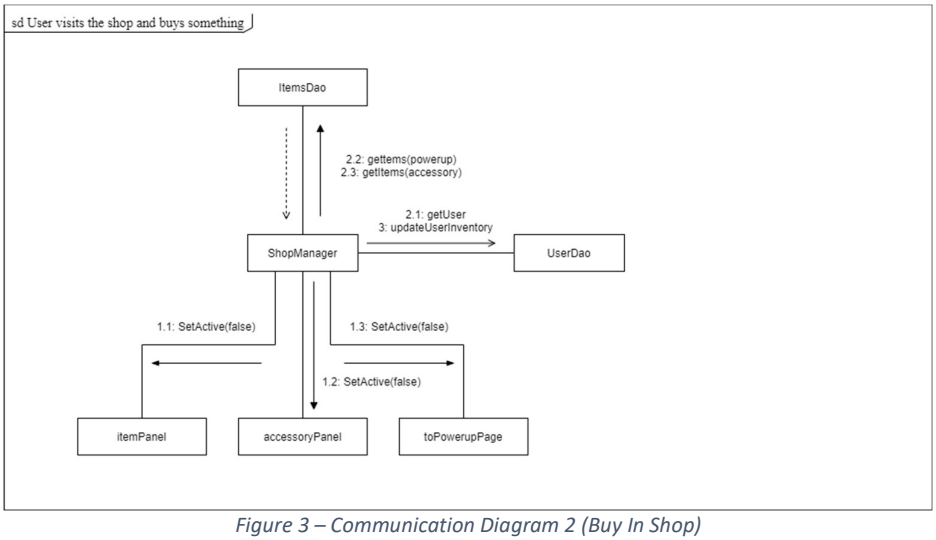

Figure 3 – Communication Diagram 2 (Buy In Shop)

Data Persistence Design Model

To demonstrate the Data Persistence Design in Foodwars, the team will be using an **Entity-Relationship Diagram**, or ER Diagram, which is shown in Figure 4.

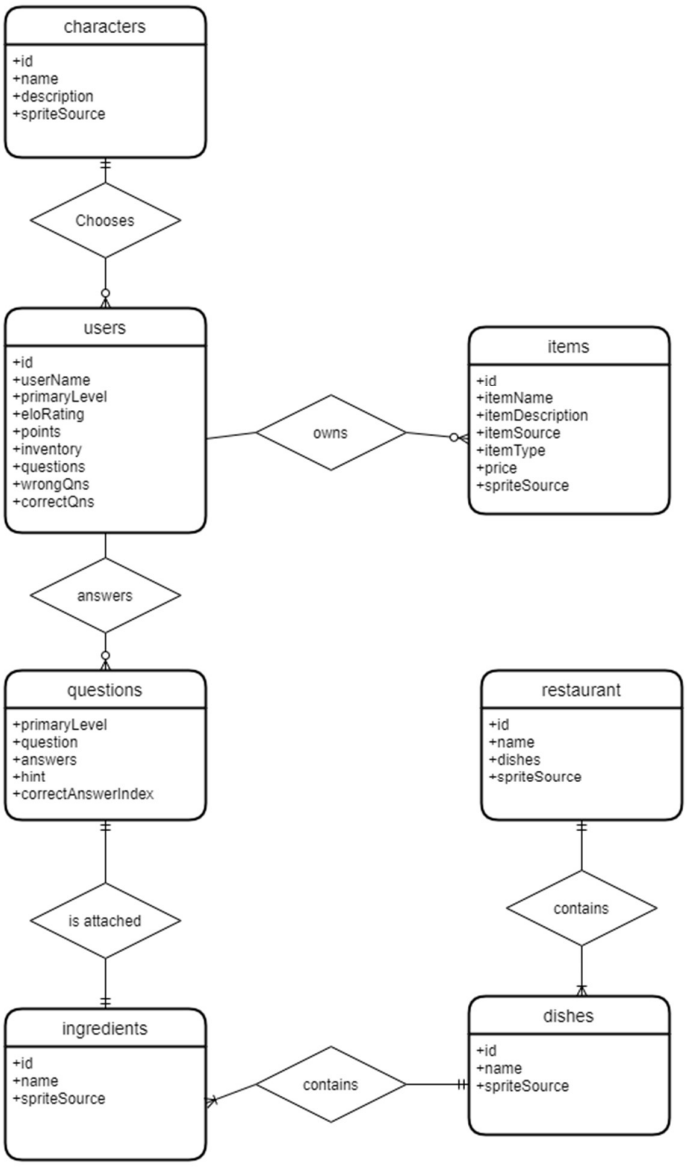

Figure 4 - Entity-Relationship Diagram