Deliverable 05

Green Team

Vanier College

Faculty of Science & Technology

System Development

420-436-VA

Prepared for Alex, and Client

Created by

Green Team

Ariel Wazana

Yaman Alhammy

Kevin Judal

Pledge of Certification

We, the Green Team, certify that this assignment is our own work.

I, Ariel Wazana, SID #2181476, certify that I have contributed to this deliverable, A.W.

I, Yaman Alhammy, SID #2195286, certify that I have contributed to this deliverable, Y.A.

I, Kevin Judal, SID #1995333, certify that I have contributed to this deliverable, K.J.

Table of Contents

Executive Overview

This project will be about the development of the prototype user interface for our application for Lumia Residence in Pointe Claire. Additionally, it will also contain the manager, Afi, and her comments about the prototype UI and the changes implemented due to her comments. This system will solve all of the business's problems. Let's dive into the executive overview.

In this Executive Overview, we dive into the critical components of our project's Deliverable 6 which is dedicated to the development of the prototype UI for the web application for Lumia Residence's restaurant in Pointe-Claire, under the guidance of our client, Afsaneh Hojabri (Afi) and her comment. This deliverable, aka deliverable #6, focuses on the creation of the prototype UI for the new system. This prototype UI will allow us to have a visualization of the application and provide Afi with a base model to provide changes and comments about.

Summary description of client

Our client's name is Afsaneh Hojabri, also known as Afi. Lumia Residence is an elderly residence that also has a restaurant where the elderly can go eat delicious food and have some drinks. Afi is the manager of the restaurant in Lumia Residence. She handles the recruitment and supervision of employees, overseeing operations in the restaurant, handling the resident's complaints, and generating financial and restaurant records.

Business Problem

The restaurant faces several challenges or obstacles that affect its overall efficiency and the quality of service it provides to the residents. These challenges revolve around the restaurant’s current operational processes, the pen-and-paper system and the impact on order management, menu printing and the overall customer experience.

Order Management: The restaurant relies on a manual pen-and-paper system for order management. Waitstaff and kitchen staff use handwritten notes to communicate the resident's orders. While this traditional approach may have worked in the past, it has become increasingly inconvenient and error-prone.

Menu Printing: The restaurant relies on the frequent printing of menus, including weekly menus, daily menus, server’s personal menus, and notepads for various purposes.

Customer Experience: The inefficiencies in order management and menu printing directly impact the customer experience.

Addressing these business problems is essential to improve operational efficiency, reduce costs, enhance customer experience, and align the restaurant with more sustainable practices. The development of the web application aims to provide innovative solutions to these challenges, ultimately benefiting both the restaurant and its valued residents.

Narrative Description (Database Design)

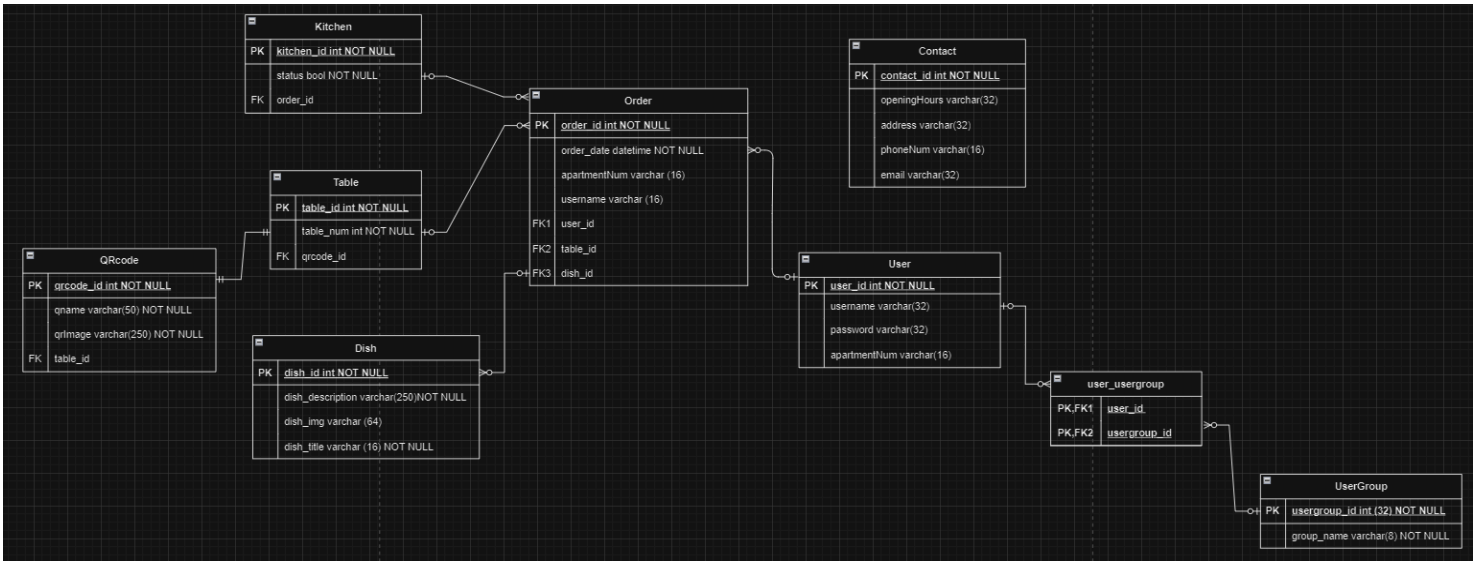
Our system requires three actors, the Resident, the Admin(Manager/Staff), and the Kitchen/Cook. These actors have different roles designed to maintain clear distinctions in permissions and responsibilities. The admin user will have full access to the web app where they can view the menu and be able to add, update, or delete dishes or contact sections. The kitchen user can only see the coming orders. The resident can only view the menu and order dishes.

Appendix 1

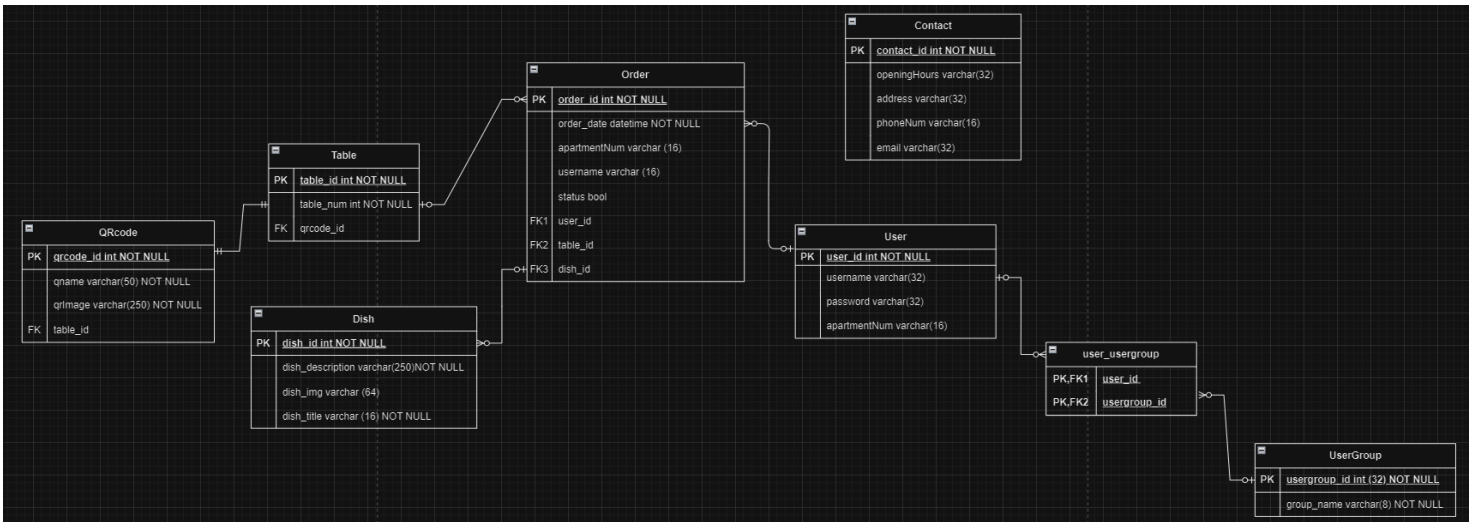
Table	Column	Data Type	Format /relations	Field Size	Nullable	Description
UserGroup	usergroup_id	int	PK		Not Null	Id for group name
UserGroup	group_name	varchar		8	Not Null	Every group going to have name (permissions)
user_user group	user_id	int	Pk/Fk		Not Null	
user_user group	usergroup_id	int	PK/FK		Not Null	
User	user_id	int	PK		Not Null	Id for the users
User	username	varchar		32		staff/admin username when logged in / sign up
User	password	varchar	hashed	32		staff/admin password when logged in / sign up
User	apartmentNum	varchar		16		Resident’s apartment number when accessing to Home Page
Order	order_id	int	PK		Not Null	Id for each submitted order
Order	apartmentNum	varchar		16		
Order	user_id	int	Fk1			User id to access username if admin/staff or apartment number to know who ordered

Order	table_id	int	Fk2			Table id to know which table to serve
Order	dish_id	int	Fk3			Dish id to know what did the user order and t access dish details
Kitchen	kitchen_id	int	PK	32	Not Null	Id for the kitchen
Kitchen	status	bool			Not Null	Boolean to check weather the order is ready or not
Kitchen	order_id	int	Fk			Order id to specify if this specific order is ready or not
Dish	dish_id	int	Pk		Not Null	Id for every dish
Dish	dish_description	varchar		250	Not Null	Dish’s description
Dish	dish_title	varchar		16	Not Null	Dish’s name
Dish	dish_img	varchar		64		Dish image (image path)
Table	table_id	int	PK		Not Null	Id for every table
Table	table_num	int		32	Not Null	Every table will be assigned a number
Table	qrcode_id	int	FK			Every table will be assigned to a specific qrcode
QRcode	qrcode_id	int	PK		Not Null	Id for every qr code
QRcode	qname	varchar		50	Not Null	For providing a name for the qr code
QRcode	qrImage	varchar		250	Not Null	For storing the qrcode image
QRcode	table_id	int	Fk			Table id to assign it to a specific qrcode

Appendix 2



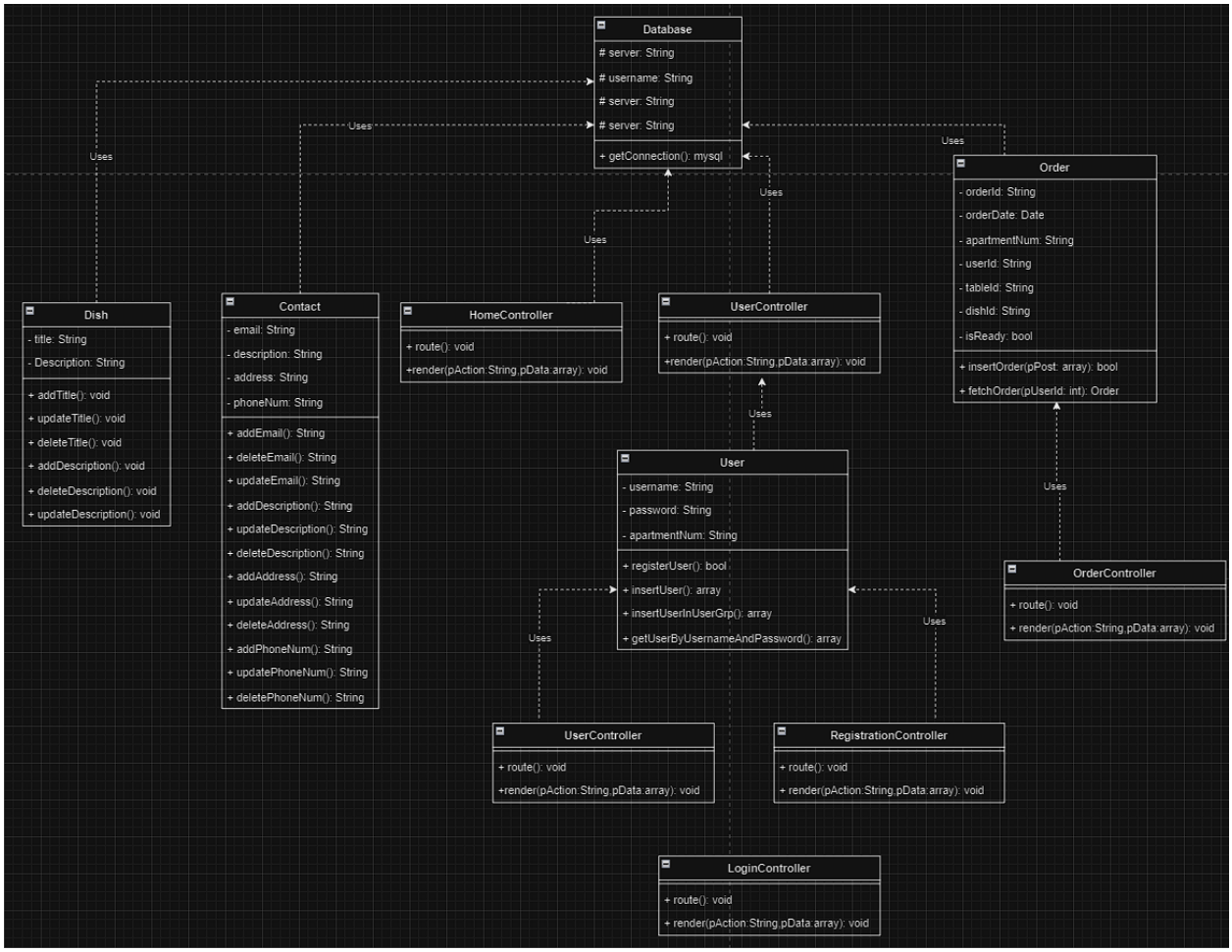
OR



The class diagram and the ERD are different because after designing the ERD we noticed that there's some classes that weren't needed like the Account in some ways and similar in another. They are similar in attributes/fields name and

quantity because all the data in the UML have to be saved with the same name and field number in the database. They are different where the database doesn't save the functionalities/methods that are in the UML

After Editing (New version of UML):



Appendix 3

For the query it has to be ofcourse optimized in every way possible. From storing the data to fetching them and displaying it to the user. There are multiple ways to do that:

- Avoid SELECT * and retrieve only necessary columns.
- Avoid redundant or unnecessary data retrieval.
- Normalize database tables
- Nulls Use NULL when non-existence of data needs to be modeled in a meaningful fashion. Do not use made-up data to fill in NOT NULL columns, such as “1/1/1900” for dates, “-1” for integers, “00:00:00” for times, or “N/A” for strings.
- Old Data Whenever data is created, ask the question, “How long should it exist for?”. Forever is a long time and most data does not need to live forever.

Appendix 4

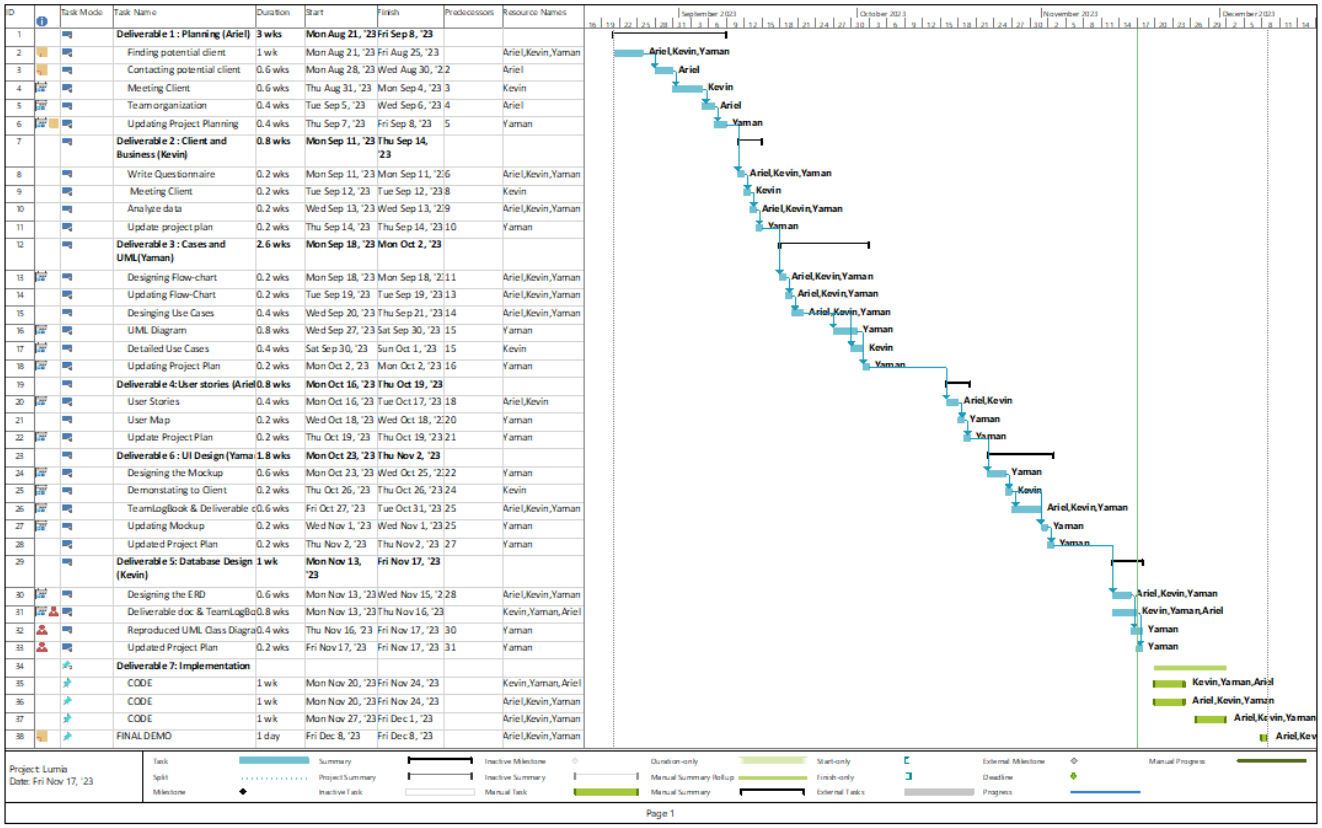
As for speed, the **kitchen**'s view page needs to access the order database every time a resident submits an order so they can start to prepare it. The restaurant needs to serve its customers as fast as possible, which means the responsive time for this page needs to be extremely brisk.

After every order the resident does, the order's table should be directly updated so the kitchen/chef can prepare/see the dish accordingly.

Loading the homepage should also be done in seconds so the residents will be able to start ordering.

Project Plan

The Project Plan will be updated every deliverable so by the end it is super detailed and perfect.



ID	Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1		**Deliverable 1 : Planning (Ariel)**	**3 wks**	**Mon Aug 21, '23**	**Fri Sep 8, '23**		
2		Finding potential client	1 wk	Mon Aug 21, '23	Fri Aug 25, '23		Ariel,Kevin,Yaman
3		Contacting potential client	0.6 wks	Mon Aug 28, '23	Wed Aug 30, '23	2	Ariel
4		Meeting Client	0.6 wks	Thu Aug 31, '23	Mon Sep 4, '23	3	Kevin
5		Team organization	0.4 wks	Tue Sep 5, '23	Wed Sep 6, '23	4	Ariel
6		Updating Project Planning	0.4 wks	Thu Sep 7, '23	Fri Sep 8, '23	5	Yaman
7		**Deliverable 2 : Client and Business (Kevin)**	**0.8 wks**	**Mon Sep 11, '23**	**Thu Sep 14, '23**		
8		Write Questionnaire	0.2 wks	Mon Sep 11, '23	Mon Sep 11, '23	6	Ariel,Kevin,Yaman
9		Meeting Client	0.2 wks	Tue Sep 12, '23	Tue Sep 12, '23	8	Kevin
10		Analyze data	0.2 wks	Wed Sep 13, '23	Wed Sep 13, '23	9	Ariel,Kevin,Yaman
11		Update project plan	0.2 wks	Thu Sep 14, '23	Thu Sep 14, '23	10	Yaman
12		**Deliverable 3 : Cases and UML(Yaman)**	**2.6 wks**	**Mon Sep 18, '23**	**Mon Oct 2, '23**		
13		Designing Flow-chart	0.2 wks	Mon Sep 18, '23	Mon Sep 18, '23	11	Ariel,Kevin,Yaman
14		Updating Flow-Chart	0.2 wks	Tue Sep 19, '23	Tue Sep 19, '23	13	Ariel,Kevin,Yaman
15		Designing Use Cases	0.4 wks	Wed Sep 20, '23	Thu Sep 21, '23	14	Ariel,Kevin,Yaman
16		UML Diagram	0.8 wks	Wed Sep 27, '23	Sat Sep 30, '23	15	Yaman
17		Detailed Use Cases	0.4 wks	Sat Sep 30, '23	Sun Oct 1, '23	15	Kevin
18		Updating Project Plan	0.2 wks	Mon Oct 2, '23	Mon Oct 2, '23	16	Yaman
19		**Deliverable 4: User stories (Ariel**	**0.8 wks**	**Mon Oct 16, '23**	**Thu Oct 19, '23**		
20		User Stories	0.4 wks	Mon Oct 16, '23	Tue Oct 17, '23	18	Ariel,Kevin
21		User Map	0.2 wks	Wed Oct 18, '23	Wed Oct 18, '23	20	Yaman
22		Update Project Plan	0.2 wks	Thu Oct 19, '23	Thu Oct 19, '23	21	Yaman
23		**Deliverable 6 : UI Design (Yama**	**1.8 wks**	**Mon Oct 23, '23**	**Thu Nov 2, '23**		
24		Designing the Mockup	0.6 wks	Mon Oct 23, '23	Wed Oct 25, '23	22	Yaman
25		Demonstating to Client	0.2 wks	Thu Oct 26, '23	Thu Oct 26, '23	24	Kevin
26		TeamLogBook & Deliverable c	0.6 wks	Fri Oct 27, '23	Tue Oct 31, '23	25	Ariel,Kevin,Yaman
27		Updating Mockup	0.2 wks	Wed Nov 1, '23	Wed Nov 1, '23	25	Yaman
28		Updated Project Plan	0.2 wks	Thu Nov 2, '23	Thu Nov 2, '23	27	Yaman
29		**Deliverable 5: Database Design (Kevin)**	**1 wk**	**Mon Nov 13, '23**	**Fri Nov 17, '23**		
30		Designing the ERD	0.6 wks	Mon Nov 13, '23	Wed Nov 15, '23	28	Ariel,Kevin,Yaman
31		Deliverable doc & TeamLogBo	0.8 wks	Mon Nov 13, '23	Thu Nov 16, '23		Kevin,Yaman,Ariel
32		Reproduced UML Class Diagra	0.4 wks	Thu Nov 16, '23	Fri Nov 17, '23	30	Yaman
33		Updated Project Plan	0.2 wks	Fri Nov 17, '23	Fri Nov 17, '23	31	Yaman
34		**Deliverable 7: Implementation**					
35		CODE	1 wk	Mon Nov 20, '23	Fri Nov 24, '23		Kevin,Yaman,Ariel
36		CODE	1 wk	Mon Nov 20, '23	Fri Nov 24, '23		Ariel,Kevin,Yaman
37		CODE	1 wk	Mon Nov 27, '23	Fri Dec 1, '23		Ariel,Kevin,Yaman
38		FINAL DEMO	1 day	Fri Dec 8, '23	Fri Dec 8, '23		Ariel,Kevin,Yaman