Software Requirement Specification Document for Loukie's catering

Ahmed Hesham, Amr Hegazi, Hagar Atalla, Mohamed Agroody
Supervised by: Dr. Essam Eliwa, Eng. Nada Ayman

May 13, 2021

Table 1: Document version history

Version	Date	Reason for Change
1.0	25-May-2021	SRS First version's specifications are defined.
1.1	2-May-2021	Added use case for User and Admin. Identified remaining functional requirements. Identified hardware constraints
1.3	5-May-2021	Non-Functional Requirements updated. Removed additional unnecessary user interface design constraints.

GitHub: https://github.com/ahmedhesham-1/catering.git

Contents

Abstract

Loukie owns a catering page on Facebook. She offers a huge variety of food and She can organize whole events' buffet. The main problem is that the client isn't able to communicate with all her customers properly due to only having a Facebook page. We have decided to engage this problem by designing a web application that is easy to manage for our client because of her main problem. We are developing a web application to help our client to manage her time and make the ordering process easier. Increase sales and be well-organized content, user-friendly site, effective calls to action, increase conversion rate.

1 Introduction

1.1 Document Purpose

This document purpose is to create a professional web application for Homemade food catering service (Loukie's Catering), It will explain the purpose and features of the web application, the interfaces, and what will do.

1.2 Document Scope

This web application will make it easier for admin and user to do the process, users will be able to see the menu categories and prices also description and photos of products which will let them choose wisely what they want, and will make it easy for admin to review orders or add new products, pictures, and offers.

1.3 Business Context

3) The program will improve the income and the process , the income will increase because it will be able to target more people and review orders easily for admin and the process because the user will be able to access it and order anywhere at anytime.

2 Similar Systems

2.1 Academic

According to CATERING MANAGEMENT
[1]

1. The main problem is always the communication between the caterer and the customer sometimes the customers after calling the caterer and making a reservation with specific food they may call them again to change some food or add more food before the reservation which the caterer won't be able to comply.

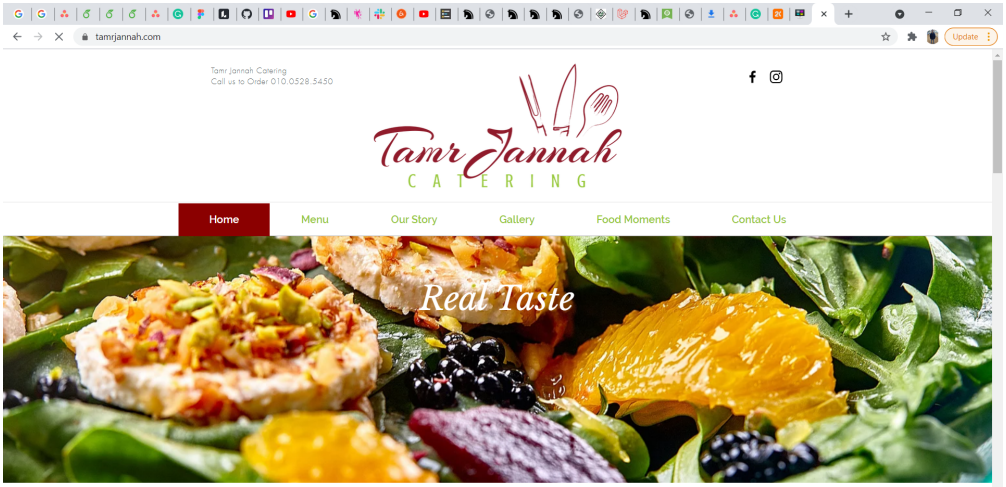
2. The researchers made a web application that is user-friendly, and improves efficiency for caterers by saving time, reduces human errors and provides customer feedback.

3. The researchers used some surveys where the customers can give their reviews on their service or the food. Customers can also gave the researchers suggestions to improve their service.

4. The main result that the researchers reached is gathering all the reservations in one web application for customer satisfaction.

2.2 Business Applications

https://www.tamrjannah.com/ Allows the user to view the menu of all the products and some photos of them and order by the phone number listed in the contact us, also there is a page showing the people feedback.



3zouma Menu

3 System Description

3.1 Problem Statement

- It is a hassle for the owner to reply to each comment or message asking for the price, ingredients, or availability and due to that the customers always complain about a late reply.
- It's more organized for both the customers and owner to have a categorized menu than random photos in Facebook posts.
- The owner always forgets her upcoming orders especially if it's ordered more than a week ago, so calendar with her upcoming orders will solve this problem and save her from this kind of embarrassing situations.
- The owner can't separate between her personal and work life because she uses the same platform (Facebook) which is very stressful.

- The customer always asks for item's feedback in the comments on her page, so it's more organized and time-saving for the customer to read previous customer's feedback under each item.
- There is an option to make categories and menus which is not able on Facebook.
- The owner always gets confused in calculating the bill, as for example, she may forget an item or calculate the wrong amount.

3.2 System Overview

We plan for a well-organized, professional web application. That will meet owner and user requirements, needs, and expectations. This app will make every step easier whether for the owner when uploading a new item or edit in it or for the user while making an order or adding feedback.

3.3 System Scope

- The web application will target people in Cairo and Giza all other location are out of scope but you can order from anywhere in Egypt to locations inside Cairo and Giza.
- The expected outcome from this project is to let people reach us more easier, and know more info about the products and order at anytime. Also we expect that people get satisfied finding the ordering process more easier for them.

3.4 System Context

The project is a web application that our customers can access it anywhere his device like phones and laptops are connected to the internet and has a browser.

3.5 Objectives

- To make the connection between user and client in the ordering process easier.
- To reduce the number of clicks it takes for a user to reach the highest traffic page that the majority of our website users regularly visit from any point on the site to 2 clicks or less.
- Increase sales and be well-organized content, user-friendly site, effective calls to action, increase conversion rate.
- To build a web application able to scale up to accommodate the increasing amount of orders.

3.6 User Characteristics

- Age: 25-60
- Some users will have little to average knowledge of how to use the web app.
- However admins must be experienced with common database concepts, SQL and database.

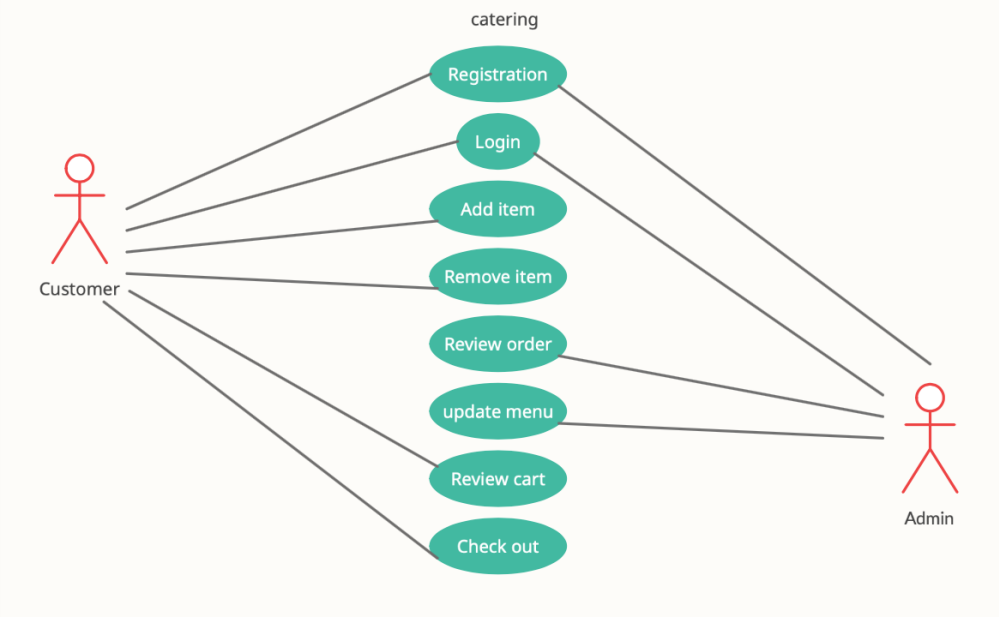
3.7 User Stories

- Our web app is for any uses will all ages to use but our target users will be between the age of 25-60.
- What the user expects from our system is and easy user-friendly interface for them to easily navigate through in order to achieve their goal whether it's for seeing the menu only or ordering as well.
- Why it is important ?
-

Examples:
As a manager, I need to keep track of the reserved orders.
As a manager, I need to keep customers' information confidential.
As a customer, I want to know what the available items are and see every season's special food.
As a customer, I want to see the feedbacks of items I wish to order.

4 Functional Requirements

4.1 System Functions



1. The Admin will be capable to add new admin.
2. The Admin will be capable to update the menu.
3. The Admin will be able to review orders.
4. The user can register.

5. The user can view available products.
6. The User can remove and add from the cart.
7. The User can make an order .

4.2 Detailed Functional Specification

Show the details of all functions shown in section 4.1. Describe each function in the following structure.

Login Function Description

Name	Login Fun
Code	FUN01
Priority	Extreme
Critical	It's the main step user need to place an order
Description	This Function allow all users to login into the system using their accounts.
Input	Email and Password.
Output	Boolean acceptance login.
Pre-condition	Sign Up
Post-condition	Will be able to order anytime
Dependency	FUN01
Risk	Don't read data from database

Check Validation Function Description

Name	Check Validation
Code	FUN02
Priority	Extreme
Critical	Checks if the data entered in a real exist and if it is valid in format.
Description	This Function allow all users to login into the system using their accounts.
Input	data
Output	Boolean acceptance valid.
Pre-condition	Enter data needs validation
Post-condition	Data entered is valid
Dependency	FUN01
Risk	Data pass validation and enter wrong

Check Phone Number Function Description

Name	Check Phone Number
Code	FUN03
Priority	Extreme
Critical	Checks if the phone number's format is valid. If so it returns true.
Description	Check if the phone number of the user is valid or not.
Input	Phone number
Output	Boolean if phone number is valid
Pre-condition	Ask user to enter phone number
Post-condition	Phone number is valid
Dependency	FUN01
Risk	Phone number is wrong

Update Menu Function Description

Name	Update Menu
Code	FUN04
Priority	Extreme
Critical	It's the main component of the catering application system, as it's the final thing people order
Description	Add and edit prices and products to the menu
Input	The product description and prices
Output	Boolean if product is updated
Pre-condition	Add new products data
Post-condition	List new products in menu
Dependency	N/A
Risk	Data base don't get updated

View Order Function Description

Name	View Order
Code	FUN05
Priority	Extreme
Critical	This is where we know the new orders that people have made so could start working on.
Description	Shows all order data so you know everything such as time, location, etc.
Input	The order number
Output	Boolean if order is found
Pre-condition	User must have done an order
Post-condition	Redirect to view order details.
Dependency	FUN06
Risk	User ordered don't get saved

Make Order Function Description

Name	Make Order
Code	FUN06
Priority	High
Critical	This is where user place order.
Description	send all order data so it could get be prepared
Input	The name, number, date, address.
Output	Boolean if order is done successfully
Pre-condition	User must have done an order
Post-condition	Redirect to view order details.
Dependency	N/A
Risk	User don't do orders

Add and Remove Function Description

Name	Add and Remove
Code	FUN07
Priority	High
Critical	This is where user choose from products and remove them from cart.
Description	User is able to add product from menu and remove them in the cart if he don't want them.
Input	The product details.
Output	Boolean
Pre-condition	user must visit menu and choose product
Post-condition	Items are added or removed successfully
Dependency	N/A
Risk	Data don't get saved

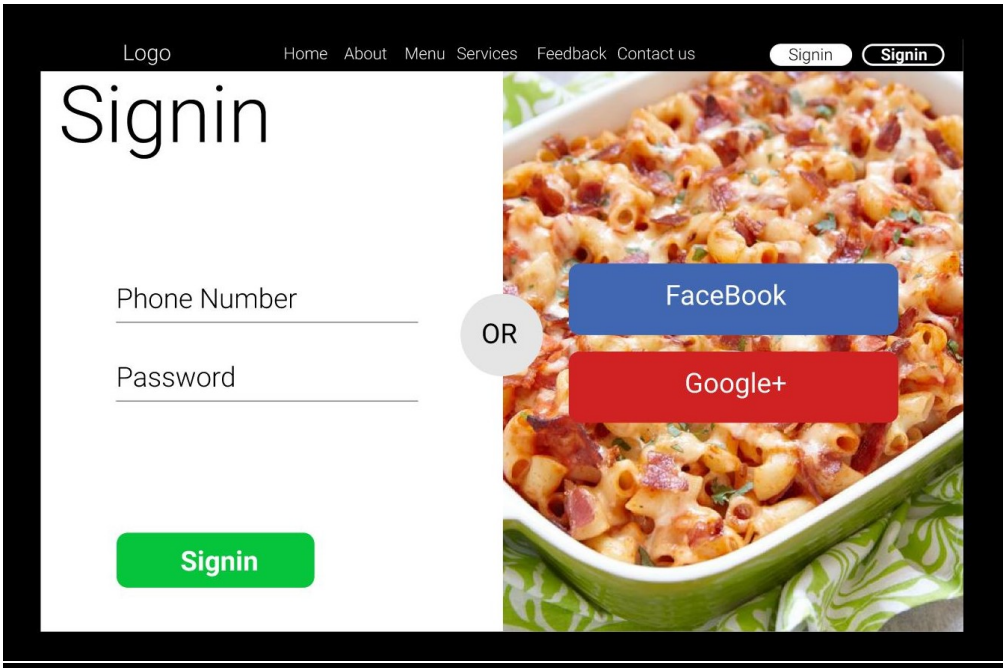
5 Interface Requirements

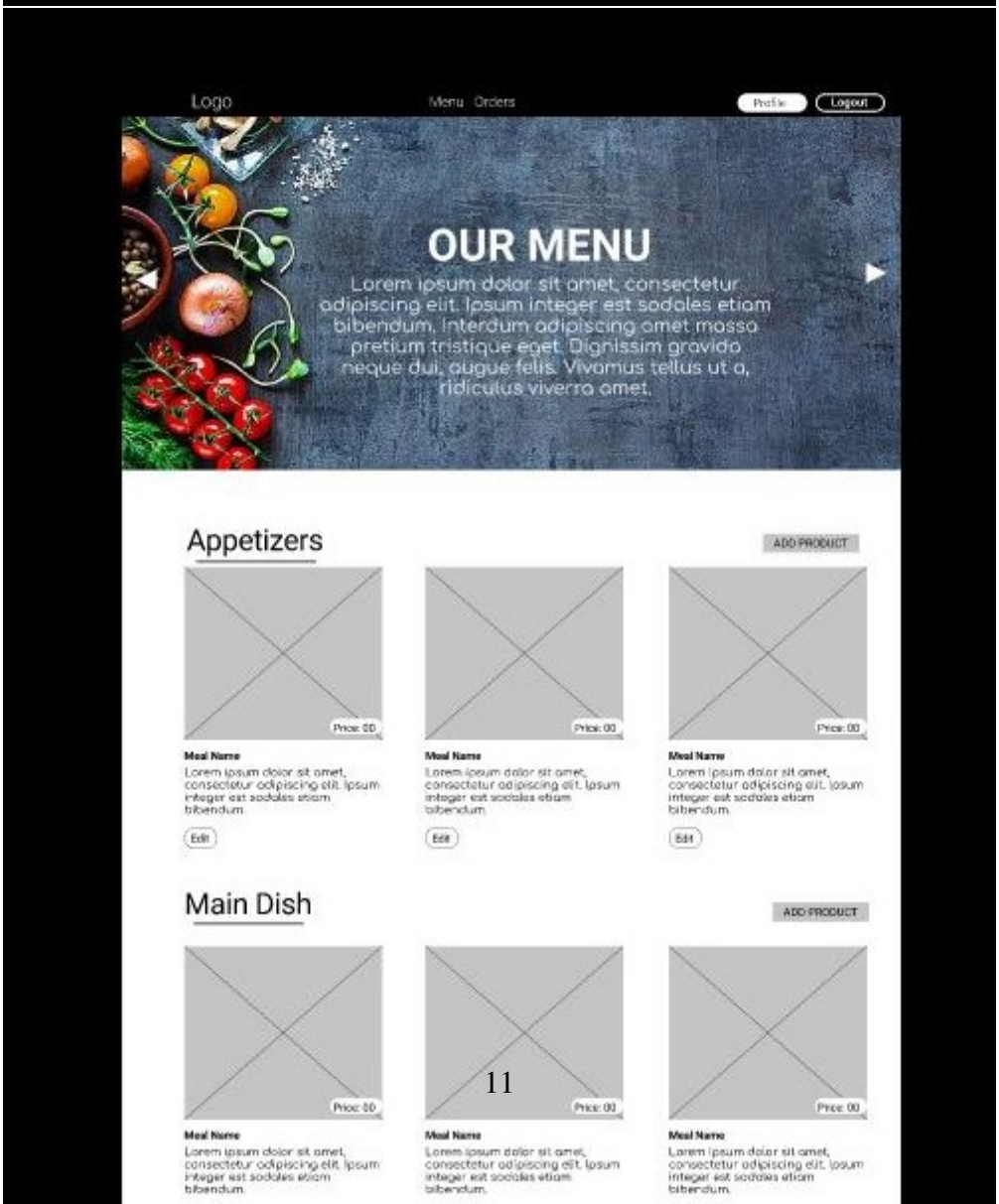
This section describes how the software interfaces with other software products or users for input or output. Examples of such interfaces include library routines, token streams, shared memory, data streams, and so forth.

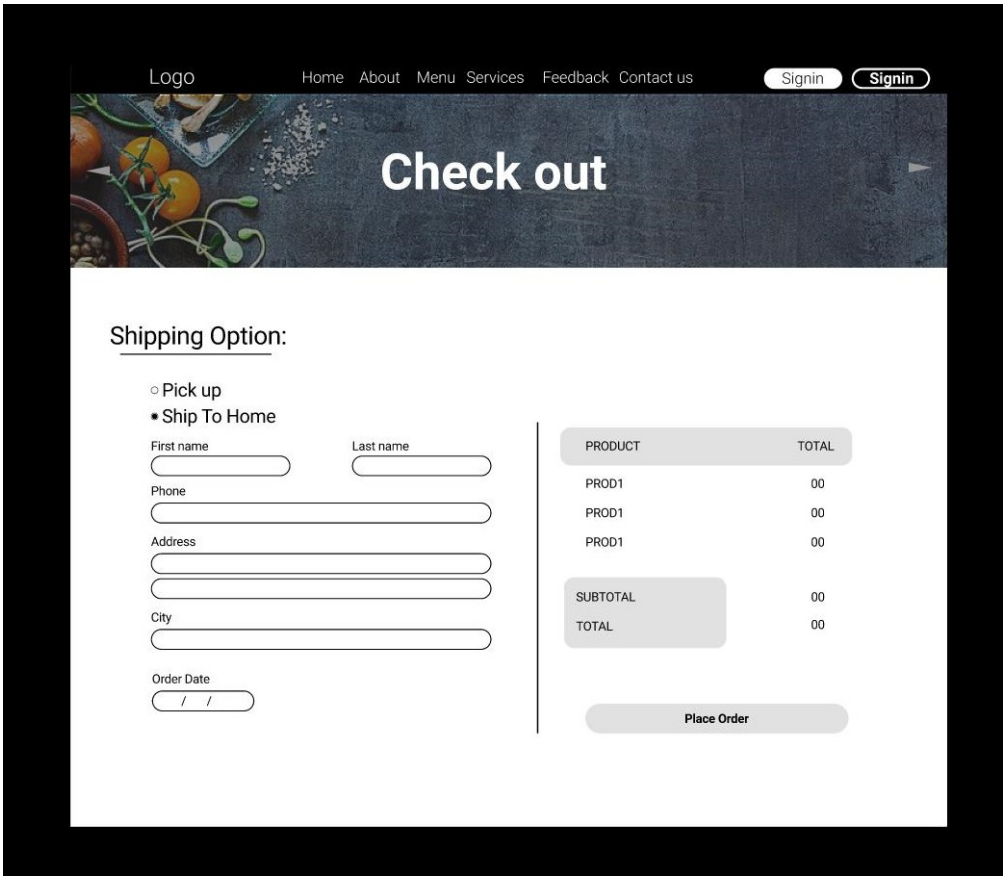
5.1 User Interfaces

Made with Figma.

5.1.1 GUI


Logo
Home
About
Menu
Services
Feedback
Contact us
Signin
Signin
Signin
Phone Number
Password
Signin
OR
FaceBook
Google+


Logo
Menu
Orders
Profile
Logout
OUR MENU
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ipsum integer est sodales etiam bibendum. Interdum adipiscing amet massa pretium tristique eget. Dignissim gravido neque dui, augue felis. Vivamus tellus ut a, ridiculus viverra amet.
Appetizers
ADD PRODUCT
Price: 00
Meal Name
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ipsum integer est sodales etiam bibendum.
Edit
Main Dish
ADD PRODUCT


Logo
Home
About
Menu
Services
Feedback
Contact us
Signin
Signin
Check out
Shipping Option:
Pick up
Ship To Home
First name
Last name
Phone
Address
City
Order Date
/ /
PRODUCT
TOTAL
PROD1
00
PROD1
00
PROD1
00
SUBTOTAL
00
TOTAL
00
Place Order

5.1.2 CLI

5.2 Hardware Interfaces

5.3 Communications Interfaces

5.4 API

5.4.1 External APIs

5.4.2 External Libraries

6 Design Constraints

6.1 Standards Compliance

6.2 Hardware Limitations

6.3 Other Constraints as appropriate

7 Non-functional Requirements

Specifies any particular non-functional attributes required by the system. non-functional requirements are constraints upon the systems behaviour.They describe how the system works, while functional requirements describe what the system should do. **Non-functional Requirements Example:**

7.1 Security

- The software must be available and behave reliably even under DOS attacks.
- The software must be available and behave reliably even under SQL injection attacks.
- The software must ensure the integrity of the customer account information.
- The server must not return a restricted web page to any browser that it cannot authenticate.
- The server must not return a restricted web page to a user who is not authorized to access it.
- The software must not accept overlong input data.
- The application must not accept invalid URLs.
- Users shall be forced to change their passwords if the "password expiration duration" has passed.
- the payroll system shall ensure that the salary data can be viewed only by the authorized users.

- the payroll system shall ensure that the salary data can be modified only by the head of the department.

7.2 Accessibility

- All course lessons will provide a text alternative to audio content.
- All web pages should be Keyboard-Friendly (can work without the use of a mouse)
- All images must have an alt text
- Forms shall be designed carefully so that each field is clearly labeled and clear validation messages are provided.

7.3 Availability

- Users can depend on the system to be up .
- Users can access the system 99% of the time without failure during working days (Sunday to Thursday)
- Users can access the system 80% of the time without failure during (Friday and Saturday) as routine system maintenance should be scheduled over the weekends.
- Any scheduled system maintenance should not take more than three hours.

7.4 Performance

- Our web application's quality must be very good in order not to give the user a negative experience.
- The front-page load must not exceed 3 second while loading.
- The system must not be overloaded.

7.5 Usability

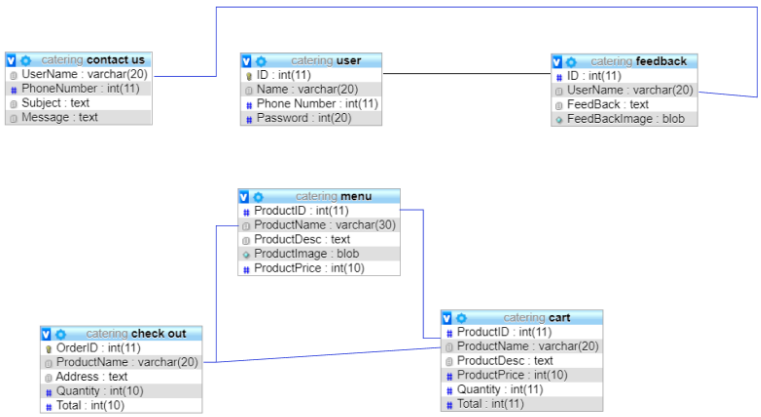
- The average time it takes for non experienced users without help to accomplish what they want to do in a web application takes about 1 to 2 minutes.
- Our pages will be easy for the user to understand in order not to take much of his time.

8 Data Design

8.1 Data Description

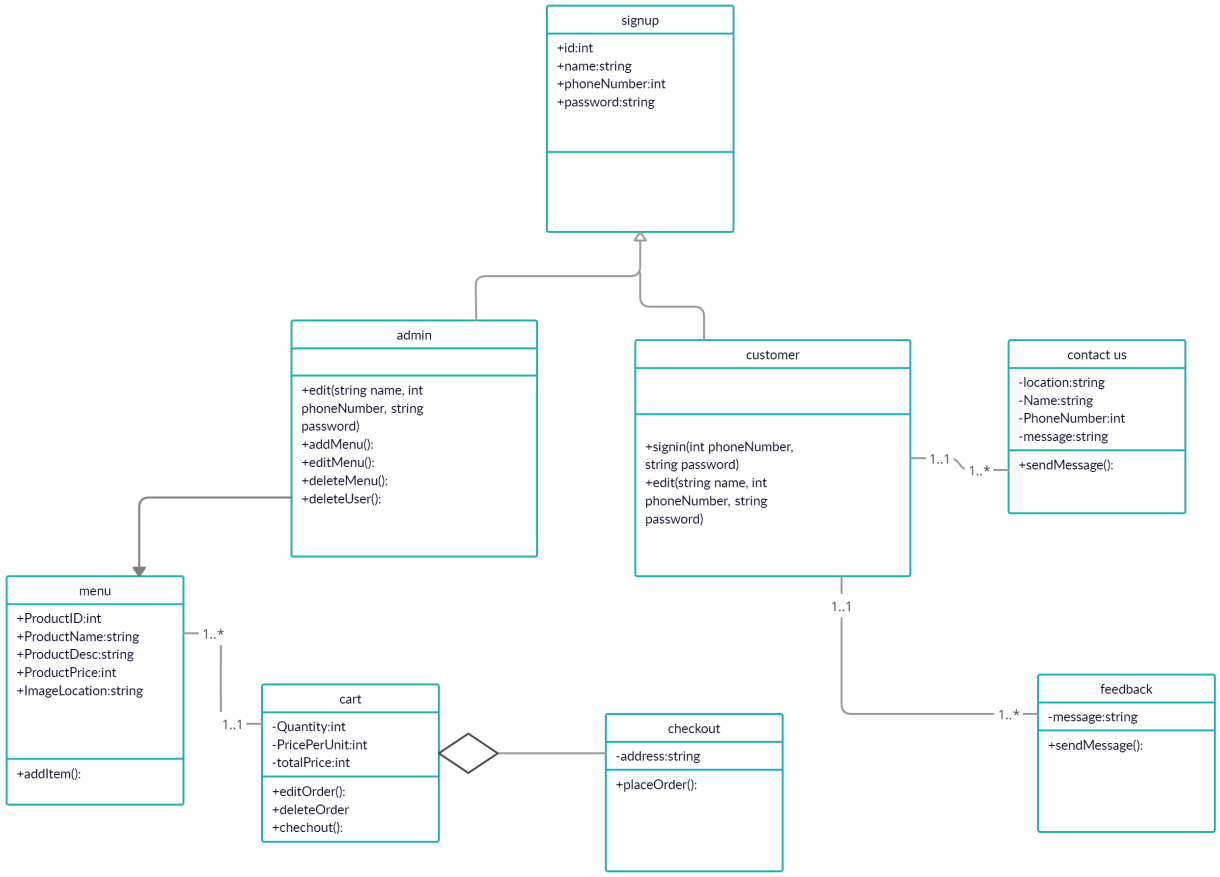
- The Original format of the data is the chats on Facebook where our client interacts with her customers.
- Our main method of capturing data in our system will be web pages formats.
- Our database is expected to be between 20 to 50 rows and 40 to 90 columns.
- Our expected number of customers are between 20 to 50.
- Each customer will have an ID.
- Yes our data contain date and time, they will be used when the customer make a huge order which will take several days to be cooked so they will reserve a date and time for the order to be picked up or delivered.
- We will use SQL database.
- The most important query which will be issued is how many people will be present and will eat when taking an order for a wedding or a party.

8.2 Database design description



9 Preliminary Object-Oriented Domain Analysis

9.1 Inheritance Relationships



9.2 Class descriptions

Table 2: Class Name - Menu

Abstract or Concrete:	Concrete.
List of Superclasses	Menu.
List of Subclasses	none
Purpose	shows all the product in the menu.
Collaborations	Cart, Admin, Customer.
Attributes	id:int,Name:string,Price:int,Description:string,Image:string.
Operations	Lists each operation AddItem(), EditItem()
Constraints	Admin could add, remove, edit and user choose only product they want.

Table 3: Class Name - Signup

Abstract or Concrete:	concrete.
List of Superclasses	Menu.
List of Subclasses	admin and customer.
Purpose	Let the user and admin create account to use the website
Collaborations	Customer,admin
Attributes	id:int, name:string, phoneNumber:int, password:string
Operations	None
Constraints	User and admin could crate accounts

Table 4: Class Name - Checkout

Abstract or Concrete:	concrete.
List of Superclasses	Menu.
List of Subclasses	none
Purpose	Is used to confirm the order and be placed
Collaborations	Customer,cart.
Attributes	address:string
Operations	placeOrder()
Constraints	Customer place order after seeing the subtotal.

Table 5: Class Name - Cart

Abstract or Concrete:	concrete.
List of Superclasses	Menu.
List of Subclasses	none
Purpose	Let the user see the products he bought and the total price.
Collaborations	customer, checkout
Attributes	Quantity:int, PricePerUnit:int, totalPrice:int
Operations	removeItem():, Increase quantity(), chechout():
Constraints	User is able to checkout after viewing the cart.

Table 6: Class Name - admin

Abstract or Concrete:	concrete.
List of Superclasses	Menu
List of Subclasses	none.
Purpose	Allows the admin to control the web app either by adding menus or deleting them.Also deleting customers if needed.
Collaborations	Names each class with which this class must interact in order to accomplish its purpose, and how.
Attributes	none
Operations	edit(string name, int phoneNumber, string password) , addMenu(): , editMenu(): , deleteMenu(): , deleteUser():
Constraints	admin able to add and delete items as he wishes.

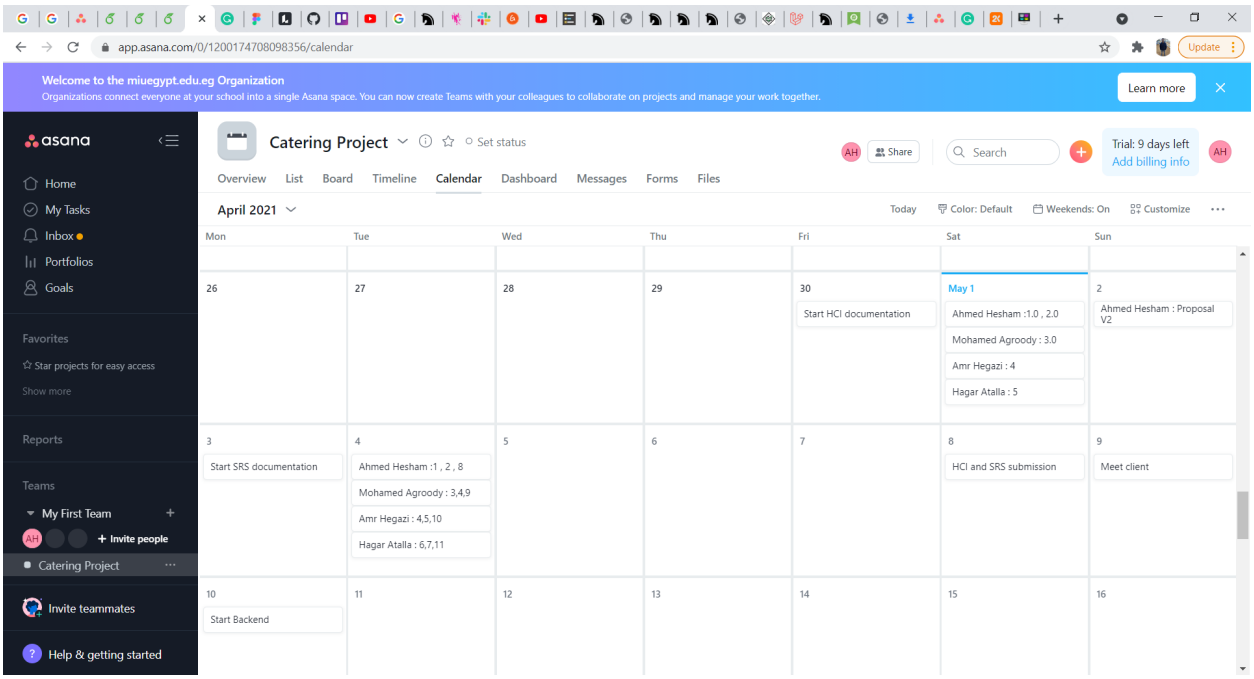
Table 7: Class Name - Contact us

Abstract or Concrete:	concrete.
List of Superclasses	Menu.
List of Subclasses	none
Purpose	It help user send message easier to the admin
Collaborations	Customer.
Attributes	location:string, Name:string, PhoneNumber:int, message:string
Operations	sendMessage():
Constraints	Customer send message to inquire about anything.

Table 8: Class Name - customer

Abstract or Concrete:	concrete.
List of Superclasses	Menu.
List of Subclasses	none
Purpose	The customers being able to sign up and order from the web app
Collaborations	signup and feedback and contact us.
Attributes	none.
Operations	signin(int phoneNumber, string password) , edit(string name, int phoneNumber, string password)
Constraints	customer can order any item.

10 Project Plan



11 Appendices

11.1 Definitions, Acronyms, Abbreviations

1. DoS attack: is an attack meant to shut down a machine or network, making it inaccessible to its intended users.

2. SQL: is a domain-specific language used in programming and designed for managing data held in a relational database management system, or for stream processing in a relational data stream management system.

3. Web Application: s application software that runs on a web server, unlike computer-based software programs that are run locally on the operating system of the device. Web applications are accessed by the user through a web browser with an active network connection.

11.2 Supportive Documents

Our team had a meeting with the client in which we discussed everything needed for the project.

References

[1] Ayesha Shaikh, Anjali Singh, Gargee Walawalkar, et al. “CATERING MANAGEMENT”. In: (2019).