Software Design Specification

for

MenuWhiz

Prepared by

Zahin Ahmed

Nafisa Tafshir

Md. Jubaer Khan

28.08.2020

Table of Contents

1. Introduction

The purpose of this Software design Specification (SDS) is to provide a description of the design of MenuWhiz. It includes design components, high level component design, class diagrams and their descriptions, details about ER diagrams to proceed with an understanding of what is to be built and how it is implemented in our project.

1.1 Document Conventions

Main Section Titles

Font: Calibri

Face: Bold

Size: 18

Sub Section Titles

Font: Calibri

Face: Bold

Size: 14

Sub subsection Titles

Font: Calibri

Face: Bold

Size: 13

Subsub subsection Titles

Font: Calibri

Face: Bold

Size: 12

Paragraph text Explanations

Font: Calibri

Face: Normal

Size: 12

2. Description of Design Components

In MenuWhiz our basic components are the database, the server-side and API, and the client-side (Android application for customer and Web interface for restaurants).

2.1 Database

One database will be used to store all the information regarding customers, restaurants, menu and order information. MySQL, a Relational Database Management System (RDBMS), will be used for this purpose.

2.2 Server-side and API

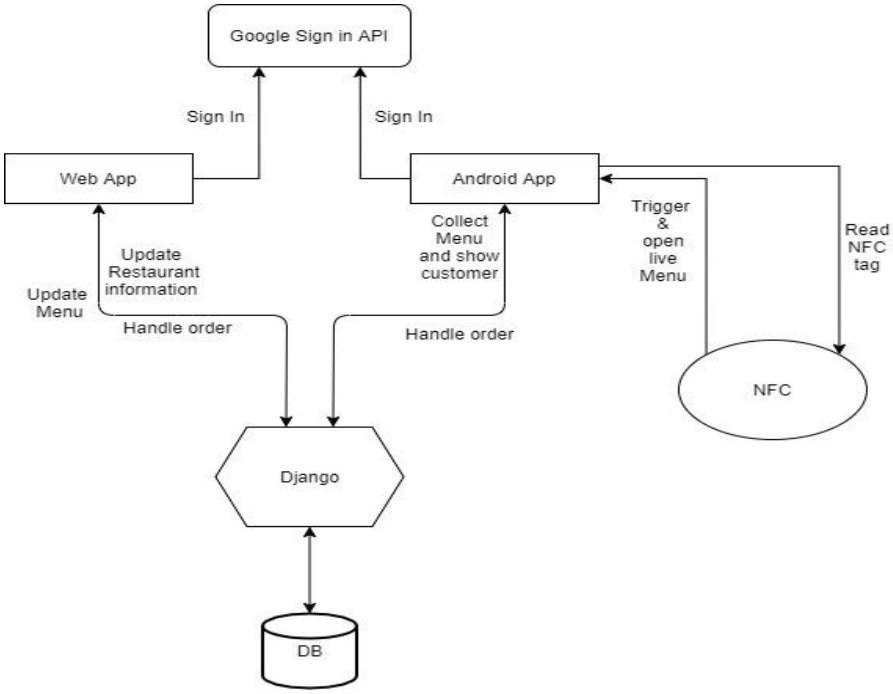
The backend or server-side of both of our android application and website will be the same. The backend will be designed using Django, a framework written using python. The backend will deal with activities such as user authentication, information retrieval from database, updating and writing information to database and processing information.

The android application and website interface will access the backend through an API. We will also use third party APIs such as Google Sign in, and NFC writer/reader to facilitate features of the product.

2.3 Client- Side

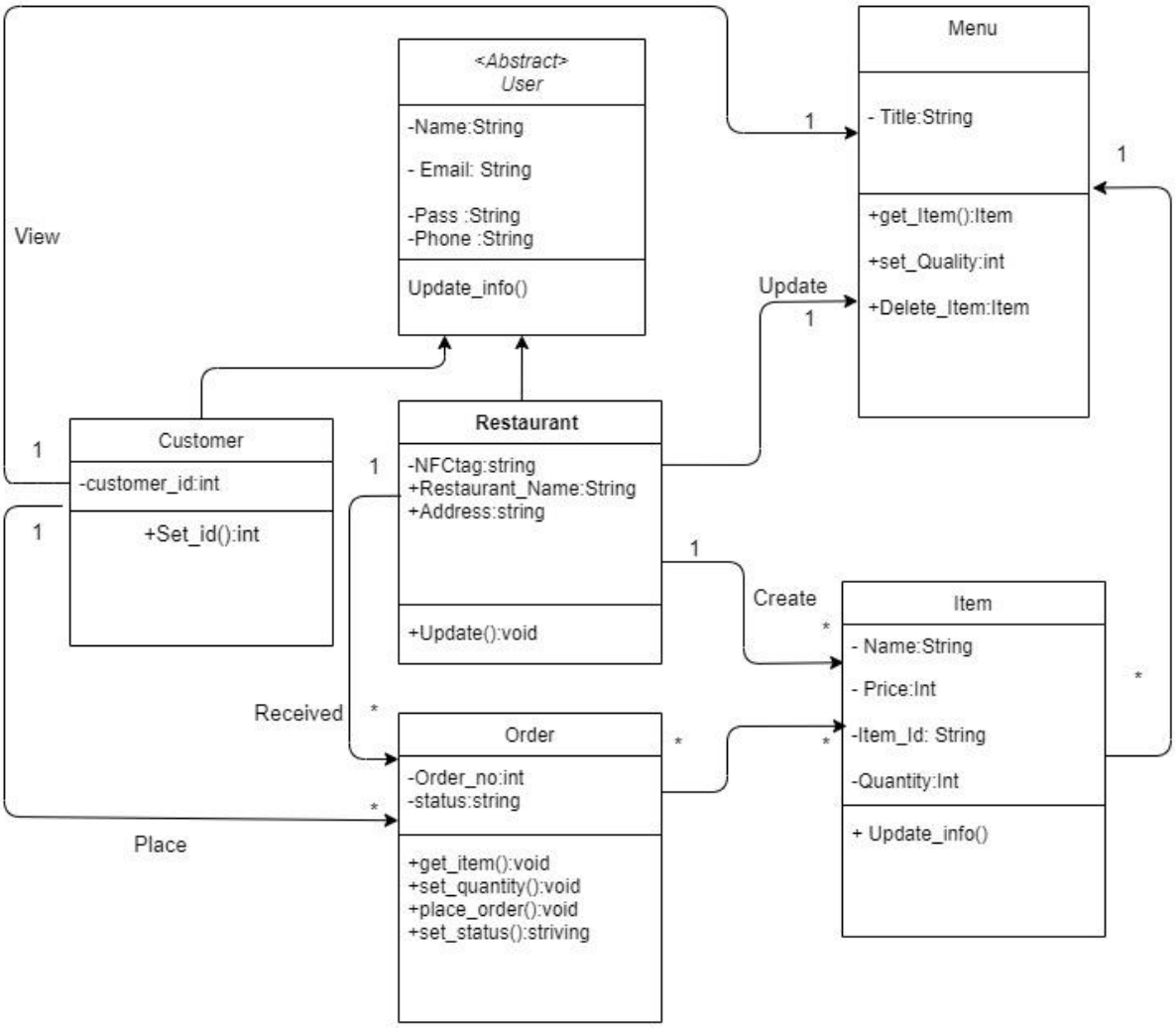
The client-side of our product has two interfaces, the android application and the web interface. The interfaces will be responsible for providing access to the features of our product, and executing the dynamic scripts which are variable from user to user.

3. High-Level Component Design



4. Class diagrams

4.1 Detailed Class Diagrams



4.2 Detailed Class Descriptions

A. **Class Name:** User
Purpose: Updates basic info about user
Public attributes: None
Public Method:
Name: Update_info()
Parameters: None
Return Type: void
Purpose and description: Every information of the class will be updated by invoking the method.

B. **Class Name:** Menu

Purpose: This class will update every available item in the restaurant
Public attributes: None
Public Methods:

a. **Name:** get_item()
Parameters: None
Return Type: Item
Purpose and description: Any items can be added to the menu .

b. **Name:** set_Quantity()
Parameters: None
Return Type: Integer
Purpose and description: quantity of the available items will be updated.

c. **Name:** Delete_items()
Parameters: None
Return Type: Item
Purpose and description: Any items can be deleted from the menu.

C. **Class Name:** Item
Purpose: Updates the items information about the menu.
Public attributes: None
Public Methods:
Name: Update_info()
Parameters: None
Return Type: Void
Purpose and description: Updates every information of any items.

D. **Class Name:** Restaurant
Purpose: Identifiers of the restaurants will be updated by this class
Public attributes:

a. **Name:** Restaurant_Name
Type: String
Purpose: It updates the name of the restaurant

b. **Name:** Address
Type: String
Purpose: Address of the restaurant will be updated

Public Methods:
Name: Update()

Parameters: None
Return Type: Void
Purpose and description: Information of the restaurant will be updated.

E. **Class Name:** Order
Purpose: Order is placed through the class
Public attributes: None
Public Methods:

a. **Name:** get_item()
Parameters: None
Return Type: Void
Purpose and description: Items are updated when customers select items.

b. **Name:** set_quantity()
Parameters: None
Return Type: Void
Purpose and description: Number of every selected items are updated.

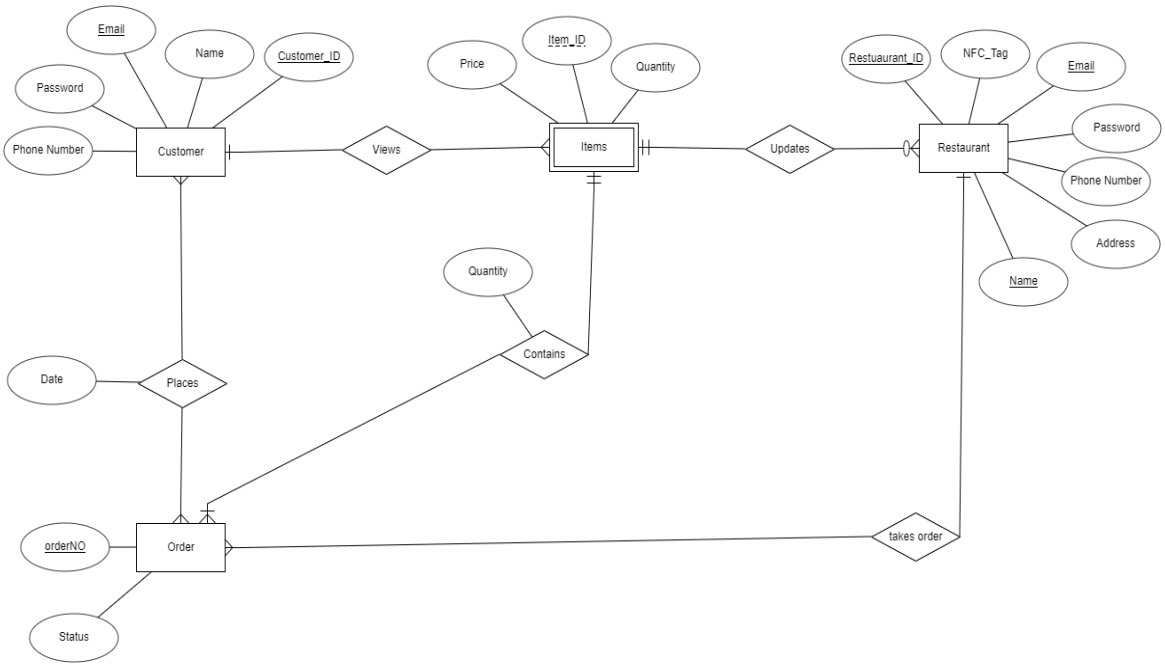
c. **Name:** place_order()
Parameters: None
Return Type: Void
Purpose and description: Orders is paced.

d. **Name:** set_status()
Parameters: None
Return Type: String
Purpose and description: The order status is updated.

F. **Class Name:** Customer
Purpose: People who want to view menu and order from this app.
Public attributes: None
Public Methods:
Name: Set_id()
Parameters: None
Return Type: Integer
Purpose and description: Automatic generates a id for customer.

5. Database Design

5.1 ER Diagram



5.2 Tables

1. Name: Customers
   Purpose: Storing the information of customers
   Attributes:
   A. Name: Customer_ID
      Type: Integer
      Unique: Yes
      Primary key: Yes
      Foreign key: No

   B. Name: Name
      Type: String
      Unique: No
      Primary key: No
      Foreign key: No

   C. Name: Email
      Type: String
      Unique: Yes
      Primary key: No
      Foreign key: No

D. Name: Phone number
   Type: Integer
   Unique: No
   Primary key: No
   Foreign key: No
E. Name: Password
   Type: String
   Unique: No
   Primary key: No
   Foreign key: No

2. Name: Restaurant
   Purpose: Storing the information of restaurants
   Attributes:
   A. Name: Restaurant_ID
   Type: Integer
   Unique: Yes
   Primary key: Yes
   Foreign key: No

   B. Name: Name
   Type: String
   Unique: Yes
   Primary key: No
   Foreign key: No

   C. Name: Email
   Type: String
   Unique: Yes
   Primary key: No
   Foreign key: No

   D. Name: Address
   Type: String
   Unique: No
   Primary key: No
   Foreign key: No

   E. Name: Phone number
   Type: Integer
   Unique: No
   Primary key: No
   Foreign key: No

F. Name: Password
Type: String
Unique: No
Primary key: No
Foreign key: No

G. Name: NFC_Tag
Type: String
Unique: Yes
Primary key: No
Foreign key: No

Dependent table: Items

A. Name: Item_ID
Type: Integer
Partially Unique: Yes
Partial Primary key: Yes
Foreign key: No

B. Name: Name
Type: String
Unique: Yes
Primary key: No
Foreign key: No

C. Name: Email
Type: Quantity
Unique: No
Primary key: No
Foreign key: No

D. Name: Price
Type: Float
Unique: No
Primary key: No
Foreign key: No

3. Name: Orders
Purpose: Storing the information of current order status
Attributes:
A. Name: orderNO
Type: Integer

Unique: Yes
Primary key: Yes
Foreign key: No

B. Name: Status
Type: String
Unique: No
Primary key: No
Foreign key: No

4. Name: Placed_Orders
   Purpose: Storing the information of who placed which orders when.
   Attributes:
   A. Name: Customer_ID
   Type: Integer
   Unique: Yes
   Primary key: Yes
   Foreign key: Yes

   B. Name: orderNo
   Type: Integer
   Unique: Yes
   Primary key: Yes
   Foreign key: Yes

   C. Name: Date
      Type: Date
      Unique: No
      Primary key: No
      Foreign key: No

5. Name: Order_Contents
   Purpose: Storing the information of which items each order contains
   Attributes:
   A. Name: orderNo
      Type: Integer
      Unique: Yes
      Primary key: Yes
      Foreign key: Yes
   B. Name: Item_ID
      Type: Integer
      Unique: Yes
      Primary key: Yes
      Foreign key: Yes

C. Name: Quantity
   Type: Integer
   Unique: No
   Primary key: No
   Foreign key: No

6. Implementation Plan

Target date	Landmark	Responsibility
1st September	Sign up, Sign in, Sign out interfaces of Android Application created	Md. Jubaer Khan
	Sign up (without NFC configuration), Sign in, View Menu and Sign out interfaces created	Nafisa Tafshir
	Backend of completed interfaces configured	Zahin Ahmed
5th September	NFC Configuration interface created	Nafisa Tafshir
	NFC Configuration enabled in backend	Zahin Ahmed and Md. Jubaer Khan
15th September	Completed interfaces of both Android App and Website	Zahin Ahmed, Md. Jubaer Khan and Nafisa Tafshir
30th September	Full backend configured and testing completed	Zahin Ahmed, Md. Jubaer Khan and Nafisa Tafshir