Sri Lanka Institute of Information Technology

Assignment 2

MLB_WD_04.02_03

Online Recipe Management System

Information Systems & Data Modeling – IT1090

B.Sc. (Hons) in Information Technology

Information Systems and Data Modeling IT1090

Assignment

Title: Online Recipe Management System

Batch Number: MLB_WD_04.02	Group Number: 03

Declaration:

We hold a copy of this assignment that we can produce if the original is lost or damaged.

We hereby certify that no part of this assignment has been copied from any other group's work or from any other source. No part of this assignment has been written / produced for our group by another person except where such collaboration has been authorized by the subject lecturer/tutor concerned.

Group Members:

IT23227804	N.V. RANDUNUGE	Randunuge
IT23227354	W.C.S.A LOWE	
IT23218758	T.R.S SURAWEERA	
IT23223912	L.H.R.C LAMAHEWA	
IT23228658	I.W.K HASARANGA	Hasaranga

Submitted on: 04/05/2024

Contents

Hypothetical scenario of Online Recipe Management System

“Foodify” recipe management system is a platform created by and for kitchen experts to organize and keep track of all recipes in one place. “Foodify” connects home cooks worldwide with their best source of inspiration – other home cooks. This website strives to be the best resource to learn cooking from scratch, and also for every home cook who loves to share their family recipes, create new recipes, and try out making tasty global recipes at home.

Anyone around the world can access the Foodify website and read recipes and articles. Both registered as well as non-registered users can view recipes, read articles on cooking tips and tricks, and explore our cookbooks. However, a registered user always get more facilities than a guest. A registered user of “Foodify” gets the opportunity to share their own recipes, create articles, comment on their experiences after trying out recipes, and gets to purchase our e-cookbooks to learn more about cooking. Also, registered users get to easily plan their meals for the whole week, by making a meal plan that we provide through our system.

If a user wants to share a recipe that's been passed down in the family or a new recipe they came up with, they will have to get registered to the system and become a part of the community first. The user will be directed to fill out a form and submit the recipe through it. Later the user-uploaded recipe will be published on the website, only if the recipe developing team approves the recipe after testing them. Each new recipe is run by a team of recipe testers, making sure the recipe meets the good standards of our system. Only after he team decides if the recipe is good enough to go on Foodify, the system admin will update the website and the database, while notifying the user via an email, that their recipe has been approved to be published on the website. If the recipe was rejected, the user will be informed through an email with the reasons for rejection.

If any user who uses this system came cross any issues, they can contact one of the Contact Support Agents who will be 24/7 available to help the users.

Requirement Analysis

Main Requirements Analysis

➢ Functional Requirements

Main functions of a website that take place between the users and the system is described by the Functional Requirements. Five users are using this online recipe management system. They are Guest user, Registered user, Admin, Recipe developer and Support Agent. They can use the website for their needs.

<u>Guests and Registered users.</u>

User requirements –

- Guest can view recipes without login to the system.
- Guest can register to the system by providing g required details for the registration.
- Guest can check FAQ.
- Registered User login to the system using by providing required user login credentials.
- Registered User can edit their own Account details.
- Registered User can upload their own recipes.
- Registered User can buy cookbooks.
- Registered User can contact website contact us page and get help from the support agent.

System requirements –

- System validates the credentials entered by the Guest.
- System should approve registration details and create a user account.
- System displays the feedback uploaded by the users.
- System should validate the login credentials entered by the registered user.
- System should save the recipe details uploaded by the user.
- System should send the deleting request and approving request to the admin.

Admin

User requirements –

- Admin signs into the website by providing required login credentials.
- Admin can add and remove accounts.
- Admin can activate and deactivate user account.
- Admin can update recipe details.
- Admin checks user feedback and reviews.
- Admin can approve or reject the recipe details uploaded by the user.

System requirements –

- System validates the admin login credentials.
- System should delete details of the recipes uploaded by the user.
- System should update the details of the user and modified accounts in the database.
- System stores the member feedback, recipe details and reviews and display them.

Recipe Developing Team

User requirements –

- Recipe developer signs into the website by providing required login credentials.
- Recipe developer views the recipes uploaded by the users.
- Recipe developer test the new uploaded recipes and give a report to the admin.

System requirements –

- System validates the recipe developer login credentials.
- System forwards the latest uploaded recipes to the recipe developer.

Support Agent

User Requirements –

- Support agent signs into website by providing required login credentials.
- Support agents check inquiry list.
- Support agents provide answers to the user’s questions.

System requirements –

- System validates the support agent login credentials.
- System displays users’ inquiries to the support admin.
- System displays the FAQs to the users.
- The system saves new inquiries and their answers to the database.

➢ Non-Functional Requirements

Non-functional requirements simply known as quality attributes. It explains the components of the system that aren't immediately relate to a except functionality. It's probable that at some instances non-functional requirements are more important than functional ones. if these objectives are not achieved the system could useless.

Speed

- The system needs to function fast.
- The system can access multiple users at same time without any errors.

Availability

- The system should be available in 24/7.

User friendly

- The system should be accessible to users with low It literacy.

Reliability

- The system should have ability to detect the invalid user credentials.

Security

- The system must have ability to prevent unauthorized access, misuse, forgery and secure user data.
- Also, by providing unique user ID and password, no one can access the system by using any other's user ID and password.

Scalability

- The system should be able to handle a higher workload on-demand.

Performance

- Admin can add, delete, update and edit.
- Multiple users can be able to access the system at the same time and the response of the system regarding to the user requests will be very high.

Data Requirements Analysis

- User
  - UserID
  - FirstName
  - LastName
  - Age
  - DOB
  - Gender
  - City
  - Country
  - Email
  - Password
- Admin
  - AdminID
  - AdminName
  - DOB
  - Email
  - Phone_no
  - Password
- Recipe
  - RecipeID
  - Title
  - Description
  - Instructions
  - Cooking Time
  - Preparation Time
  - Status
- Recipe Developer
  - DeveloperID
  - Name
  - DOB
  - Phone_no
  - Address
  - Email
  - Password
  - Qualification

- E-Cookbook
  - CookbookID
  - Title
  - Publication_date
  - Author
  - Tags
  - Description
  - Price

- Meal Plan
  - PlanID
  - Day_of_week
  - Breakfast
  - Lunch
  - Dinner
  - Snacks

- Sales Report
  - ReportID
  - ReportID
  - Date
  - Quantity_sold
  - Payment_Method
  - Revenue

- Support Agent
  - AgentID
  - Name
  - DOB
  - Phone_no
  - Email
  - Password

- Inquiry
  - InquiryID
  - Date
  - Message

- Sender_email

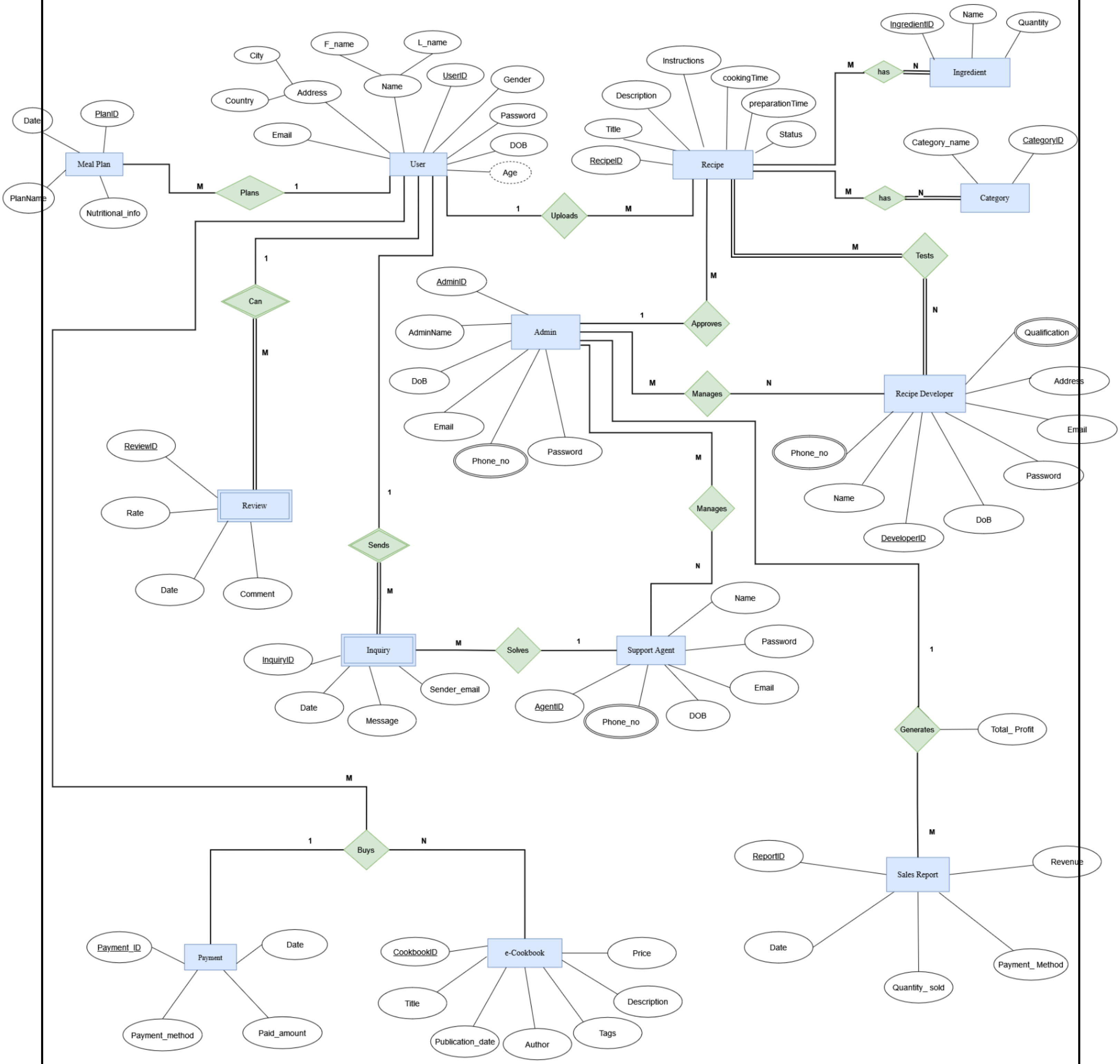
- Review
    - ReviewID
    - Date
    - Rate
    - Comment

- Payment
    - PaymentID
    - Date
    - Payment_method
    - Paid_amount

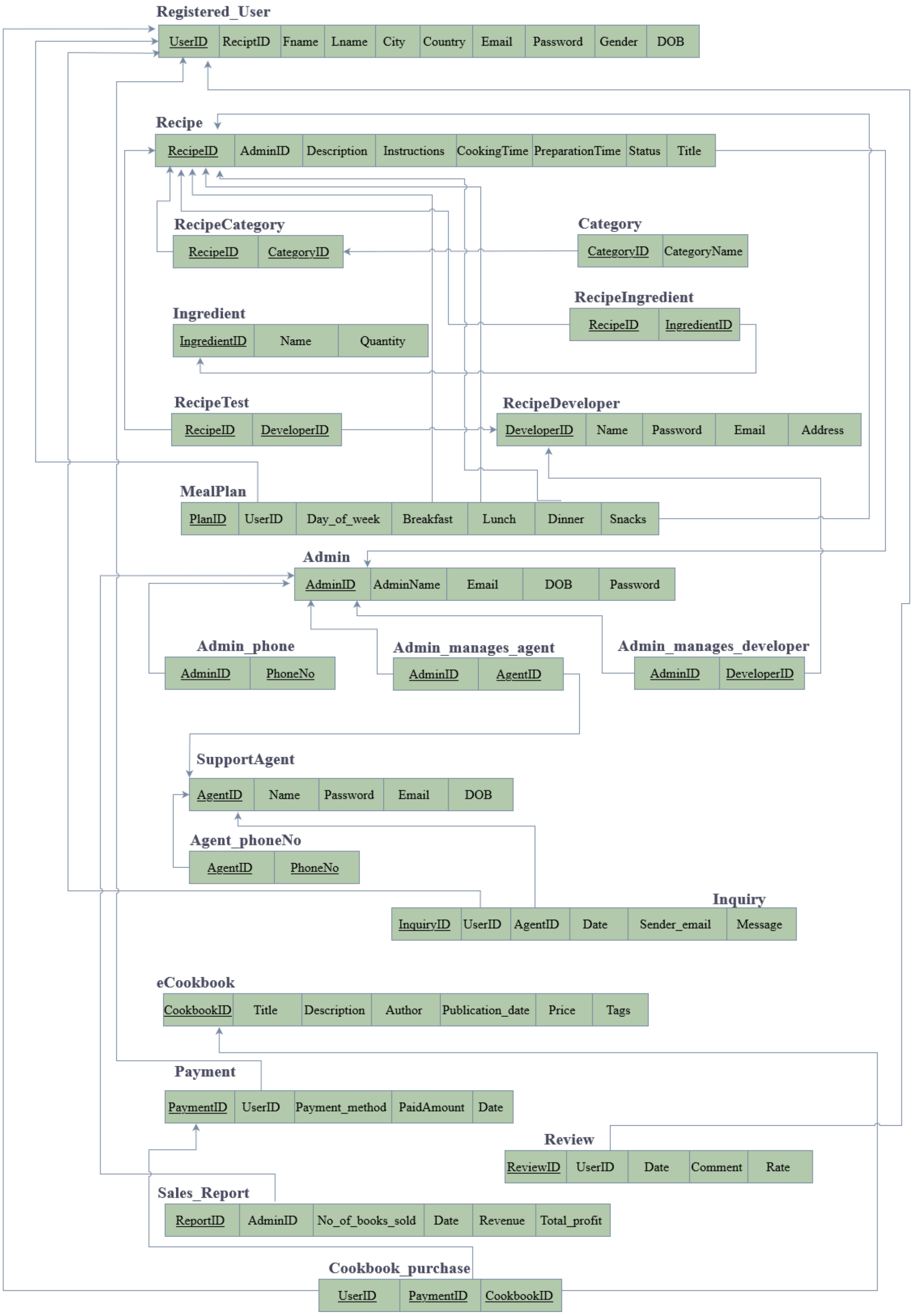
- Category
    - CategoryID
    - CategoryName

- Ingredient
    - IngredientID
    - IngredientName
    - Quantity

Entity Relationship (ER Diagram)



Relational Schema



Registered_User

UserID	ReciptID	Fname	Lname	City	Country	Email	Password	Gender	DOB

Recipe

RecipeID	AdminID	Description	Instructions	CookingTime	PreparationTime	Status	Title

RecipeCategory

RecipeID	CategoryID

Category

CategoryID	CategoryName

RecipeIngredient

RecipeID	IngredientID

Ingredient

IngredientID	Name	Quantity

RecipeTest

RecipeID	DeveloperID

RecipeDeveloper

DeveloperID	Name	Password	Email	Address

MealPlan

PlanID	UserID	Day_of_week	Breakfast	Lunch	Dinner	Snacks

Admin

AdminID	AdminName	Email	DOB	Password

Admin_phone

AdminID	PhoneNo

Admin_manages_agent

AdminID	AgentID

Admin_manages_developer

AdminID	DeveloperID

SupportAgent

AgentID	Name	Password	Email	DOB

Agent_phoneNo

AgentID	PhoneNo

Inquiry

InquiryID	UserID	AgentID	Date	Sender_email	Message

eCookbook

CookbookID	Title	Description	Author	Publication_date	Price	Tags

Payment

PaymentID	UserID	Payment_method	PaidAmount	Date

Review

ReviewID	UserID	Date	Comment	Rate

Sales_Report

ReportID	AdminID	No_of_books_sold	Date	Revenue	Total_profit

Cookbook_purchase

UserID	PaymentID	CookbookID

SQL queries

Database and Table creation

```
CREATE DATABASE Foodify;

/*Create Registered User table*/
CREATE TABLE Registered_User
(
   UserID varchar(15) not null,
   F_name varchar(20) not null,
   L_name varchar(20) not null,
   City varchar(10) not null,
   Country varchar(20) not null,
   Email varchar(40) not null,
   Passcode char(8) not null,
   Gender char(1),
   DOB date,
   RecipeID varchar (10),

   CONSTRAINT UserPK Primary Key (UserID)
);



/*Create Recipe table*/
CREATE TABLE Recipe
(
       RecipeID varchar(10) not null,
       Title varchar (50),
       Description varchar (300),
       Instructions varchar (1000),
       CookingTime varchar (6),
       PreparationTime varchar(6),
       Status varchar(8),
       AdminID varchar (10),

       CONSTRAINT RecipePK PRIMARY KEY (RecipeID),
);

/*Create Ingredient table*/
CREATE TABLE Ingredient
(
       IngredientID int not null PRIMARY KEY,
       Name varchar(30),
       Quantity varchar(8),

);
```

```sql
/*Create Recipe Developer's table*/
CREATE TABLE RecipeDeveloper
(
	DeveloperID varchar(5) PRIMARY KEY not null,
	Name varchar(20),
	password char(8)not null,
	Email varchar(40)not null,
	Address varchar(60)not null,

);

/*Create eCookbook table*/
CREATE TABLE eCookbook
(
	CookbookID varchar(6)not null PRIMARY KEY,
	Title varchar(50),
	Description varchar(500),
	Author varchar(60),
	Publication_date varchar(10)not null,
	Price varchar(5)not null,
	Tags varchar(4)
);

/*Create table for Payment details*/
CREATE TABLE Payment
(
	PaymentID int PRIMARY KEY,
	Payment_method varchar(15),
	Paid_amount int,
	Date varchar(10)not null,
	UserID varchar(15)not null,
);

/*Create table for Sales Report*/
CREATE TABLE Sales_Report
(
	ReportID varchar(10),
	Title varchar (90),
	Revenue varchar(10),
	No_of_books_sold varchar(3),
	Date varchar(10),
	AdminID varchar(10),
	Total_profit int,

	CONSTRAINT Sales_ReportPK PRIMARY KEY (ReportID),
);

/*Create table for Recipe Testing*/
CREATE TABLE RecipeTest
(
	RecipeID varchar (10),
	DeveloperID varchar (5),

	CONSTRAINT TestPK PRIMARY KEY (RecipeID, DeveloperID)
);
```

```
/*Create table named Review*/
CREATE TABLE Review
(
        ReviewID int IDENTITY(1,1) PRIMARY KEY not null,
        Rate varchar(5),
        ReviewDate date,
        Comment varchar (500),
        UserID varchar (15),
);


/*Create table named Administrator*/
CREATE TABLE Administrator
(
        AdminID varchar(10),
        AdminName varchar (50),
        Email varchar (50),
        DoB date,
        Passcode char (8),

        CONSTRAINT AdminPK PRIMARY KEY (AdminID)
);


/*Create table named Addmin_phone*/
CREATE TABLE Admin_phone
(
        AdminID varchar(10),
        PhoneNo int,

        CONSTRAINT AdminPhonePK PRIMARY KEY (AdminID,PhoneNo)
);

/*Create table for Cookbook Purchasings*/
CREATE TABLE Cookbook_Purchase
(
        UserID varchar(15),
        PaymentID int,
        CookbookID varchar(6),

        CONSTRAINT purchasePK PRIMARY KEY (UserID,PaymentID,CookbookID),

);

/*Create table for Recipe Ingredients*/
CREATE TABLE RecipeIngredient(
    RecipeID varchar(10) not null,
        IngredientID int not null,

        CONSTRAINT RecipeIngredient_PK PRIMARY KEY (RecipeID, IngredientID),
);
```

```
/*Create table for Inquiry*/
CREATE TABLE Inquiry(
    InquiryID varchar(10) not null,
       UserID varchar(15) not null,
       AgentID varchar(10) not null,
       InqDate date,
       Inquiry varchar(1000) not null,
       Sender_email varchar(20) not null,

  CONSTRAINT Inquiry_PK PRIMARY KEY (InquiryID)
);

/*Create table for Support Agent*/
CREATE TABLE SupportAgent(
        AgentID varchar(10) not null,
        AgentName char(40) not null,
        Passcode varchar(6) not null,
        Email varchar(20) not null,
        DOB date,

  CONSTRAINT SupportAgent_PK PRIMARY KEY (AgentID)
);

/*Create table for Support Agents' phone number*/
CREATE TABLE Agent_phoneNo(
    AgentID varchar(10) not null,
       PhoneNo int  not null,

  CONSTRAINT Agent_phoneNo_PK PRIMARY KEY (AgentID, PhoneNo)
);


/*Create table for Admin managing recipe developers*/
CREATE TABLE Admin_manage_developer(
    AdminID varchar(10) not null,
    DeveloperID varchar(5) not null,

  CONSTRAINT Admin_manage_developer_PK PRIMARY KEY (AdminID,DeveloperID)
);

/*Create table for Admin managing Support Agent*/
CREATE TABLE Admin_manage_Agent(
  AdminID varchar(10) not null,
  AgentID varchar(10) not null,

  CONSTRAINT Admin_manage_Agent_PK PRIMARY KEY (AdminID ,AgentID)
);

/*Create table for Categories*/
CREATE TABLE Category
(
       CategoryID int PRIMARY KEY not null,
       CategoryName varchar (20),
);
```

```
CREATE TABLE RecipeCategory
(
	CategoryID int,
	RecipeID varchar (10),

	CONSTRAINT RecipeCategoryPK PRIMARY KEY (CategoryID ,RecipeID)
);


/*Create table named MealPlan*/
CREATE TABLE MealPlan
(
	PlanID int IDENTITY(1,1) PRIMARY KEY,
	UserID varchar (15),
	Day_of_week TINYINT CHECK (Day_of_week between 1 and 7),
	Breakfast varchar (10),
	Lunch varchar (10),
	Dinner varchar (10),
	Snacks varchar (10)
);
```

Setting Foreign key constraints to create relationships between tables

```sql
/*Adding foreign keys to the relevant tables*/

ALTER TABLE Registered_User
ADD CONSTRAINT UserFK Foreign Key (RecipeID) References Recipe(RecipeID);

ALTER TABLE Recipe
ADD CONSTRAINT RecipeFK FOREIGN KEY (AdminID) REFERENCES Administrator(AdminID);

ALTER TABLE Payment
ADD CONSTRAINT PaymentFK FOREIGN KEY(UserID) REFERENCES Registered_User(UserID);

ALTER TABLE Sales_Report
ADD CONSTRAINT Sales_ReportFK FOREIGN KEY(AdminID) REFERENCES Administrator(AdminID);

ALTER TABLE RecipeTest
ADD CONSTRAINT TestFK1 FOREIGN KEY (RecipeID) REFERENCES Recipe(RecipeID),
CONSTRAINT TestFK2 FOREIGN KEY (DeveloperID) REFERENCES RecipeDeveloper(DeveloperID);

ALTER TABLE Review
ADD CONSTRAINT ReviewFK FOREIGN KEY (UserID) REFERENCES Registered_User(UserID);



ALTER TABLE Admin_phone
ADD CONSTRAINT AdminPhoneFK FOREIGN KEY (AdminID) REFERENCES Administrator(AdminID);

ALTER TABLE Cookbook_Purchase
ADD CONSTRAINT purchaseFK1 FOREIGN KEY (UserID) REFERENCES Registered_User(UserID),
CONSTRAINT purchaseFK2 FOREIGN KEY (PaymentID) REFERENCES Payment(PaymentID),
CONSTRAINT purchaseFK3 FOREIGN KEY (CookbookID) REFERENCES eCookbook(CookbookID);

ALTER TABLE RecipeIngredient
ADD CONSTRAINT RecipeIngredient_FK1 FOREIGN KEY (RecipeID) References Recipe(RecipeID),
CONSTRAINT RecipeIngredient_FK2 FOREIGN KEY (IngredientID) References
Ingredient(IngredientID);

ALTER TABLE Inquiry
ADD CONSTRAINT Inquiry_FK1 FOREIGN KEY (UserID) References Registered_User(UserID),
CONSTRAINT Inquiry_FK2 FOREIGN KEY (AgentID) References SupportAgent(AgentID);

ALTER TABLE Agent_phoneNo
ADD CONSTRAINT Agent_phoneNo_FK FOREIGN KEY (AgentID) References SupportAgent(AgentID);

ALTER TABLE Admin_manage_developer
ADD CONSTRAINT Admin_manage_developer_FK1 FOREIGN KEY (AdminID) References
Administrator(AdminID),
CONSTRAINT Admin_manage_developer_FK2 FOREIGN KEY (DeveloperID) References
RecipeDeveloper(DeveloperID);
```

```
ALTER TABLE Admin_manage_Agent
ADD CONSTRAINT Admin_manage_Agent_FK1 FOREIGN KEY (AdminID) References
Administrator(AdminID),
CONSTRAINT Admin_manage_Agent_FK2 FOREIGN KEY (AgentID) References SupportAgent(AgentID);

ALTER TABLE RecipeCategory
ADD CONSTRAINT RepCategoryFK1 FOREIGN KEY (CategoryID) REFERENCES Category(CategoryID),
CONSTRAINT RepCategoryFK2 FOREIGN KEY (RecipeID) REFERENCES Recipe(RecipeID);


ALTER TABLE MealPlan
ADD CONSTRAINT PlanFK1 FOREIGN KEY (UserID) REFERENCES Registered_User(UserID),
CONSTRAINT PlanFK2 FOREIGN KEY (Breakfast) REFERENCES Recipe(RecipeID),
CONSTRAINT PlanFK3 FOREIGN KEY (Lunch) REFERENCES Recipe(RecipeID),
CONSTRAINT PlanFK4 FOREIGN KEY (Dinner) REFERENCES Recipe(RecipeID),
CONSTRAINT PlanFK5 FOREIGN KEY (Snacks) REFERENCES Recipe(RecipeID);
```

Storing data in database

```
/*Data stored in Registered_User table*/

INSERT INTO Registered_User VALUES ('0001', 'Nimal', 'Fernando', 'Colombo', 'Sri Lanka',
'nimal12@gmail.com', '', 'Male', '1987-10-08', 'RE001');
INSERT INTO Registered_User VALUES ('0002', 'Sunil', 'Lowe', 'Negombo', 'Sri Lanka',
'sunill@gmail.com', '', 'Male', '1983-01-02', 'RE002');
INSERT INTO Registered_User VALUES ('0003', 'Nethmi', 'Kariyawasam', 'Gampaha', 'Sri
Lanka', 'nethmikkk@gmail.com', '', 'Female', '2002-09-13', 'RE003');
INSERT INTO Registered_User VALUES ('0004', 'Sumudu', 'Silva', 'Agunakolapelessa', 'Sri
Lanka', 'sumudupk@gmail.com', '', 'Male', '1999-10-19', 'RE004');
INSERT INTO Registered_User VALUES ('0005', 'Kamal', 'Perera', 'Trincomalee', 'Sri
Lanka', 'pererak2@gmail.com', '', 'Male', '1992-03-08', 'RE005');

/*Data stored in Recipe table*/

INSERT INTO Recipe VALUES ('R001', 'Meat curry', 'Rich meat curry with tender meta and
flavourful gravy', 'Brown the meat in a skilet', '1.5-2 hours', '20 minutes',
'Flavor:9/10', 'A0001');
INSERT INTO Recipe VALUES ('R002', 'Salami Sandwich','olive mixture poses as a dressing
for the lettuce and tomato', 'Marinate the fish', '30-40 minutes', '15 minutes',
'Comfort:7/10', 'A0002');
INSERT INTO Recipe VALUES ('R003', 'Chocolate pudding', 'Indulgent chocolate pudding with
a velvety-smooth texture', 'Whisk together cocoa powder,sugar,cornstarch and milk', '20-
25 minutes', '10 minutes', 'Flavor:9/10', 'A0003');
INSERT INTO Recipe VALUES ('R004', 'Creamy Garlic Pasta', 'Creamy and cheesy pasta
infused with garlic,onions and herbs', 'Saute minced garlic and onions in olive oil until
golden', '30-40 minutes', '10 minutes', 'Versatility:9/10', 'A0004');
INSERT INTO Recipe VALUES ('R005', 'Bread', 'Freshly baked bread with a golden crust',
'Knead together flour,water,yeast, and salt until smooth', '25-35 minutes', '15 minutes',
'Aroma:8/10', 'A0005');

/*Data stored in Ingredient table*/

INSERT INTO Ingredient VALUES ('I0001', 'Meat', '1kg');
INSERT INTO Ingredient VALUES ('I0002', 'Fish', '500g');
INSERT INTO Ingredient VALUES ('I0003', 'Chocolate', '250g');
INSERT INTO Ingredient VALUES('I0004', 'Tomato', '250g');
INSERT INTO Ingredient VALUES ('I0005', 'Flour', '1kg');

INSERT INTO Review VALUES ('001', '2.5', '2024-12-12', 'Great', 'R0001');
INSERT INTO Review VALUES ('002', '4.5', '2024-10-02', 'Neutral', 'R0002');
INSERT INTO Review VALUES ('003', '5.0', '2023-09-01', 'Good', 'R0003');
INSERT INTO Review VALUES ('004', '2.5', '2024-11-12', 'Good', 'R0004');
INSERT INTO Review VALUES ('005', '3.0', '2023-10-11', 'Neutral', 'R0005');
```

```
/*Data stored in RecipeDeveloper table*/

insert into RecipeDeveloper values('D001','Kinara de
Silva','12345','recipe12@gmail.com','no,123,colombo');
insert into RecipeDeveloper values('D002','Sachini
Wijesinghe','5678','sachi@gmail.com','no,88,badhulla');
insert into RecipeDeveloper values('D003','Nethmi
Kaluhediwela','88779','nethmi56@gmail.com','no,18,Gampaha');
insert into RecipeDeveloper values('D004','K Amanda
Helani','23456','amaa@gmail.com','no,89,kadawatha');
insert into RecipeDeveloper values('D005','Nithmi
Randunuge','0012','nith2003@gmail.com','no,36,malabe');


/*Data stored in RecipeTest table*/

INSERT INTO RecipeTest VALUES ('R001', 'D002');
INSERT INTO RecipeTest VALUES ('R001', 'D003');
INSERT INTO RecipeTest VALUES ('R001', 'D005');
INSERT INTO RecipeTest VALUES ('R002', 'D001');
INSERT INTO RecipeTest VALUES ('R002', 'D002');

/*Data stored in Inquiry table*/

INSERT INTO Inquiry VALUES ('I001', 'U003', 'AG004', '30-04-2024', 'Connection error',
'nethmikkk@gmail.com' );
INSERT INTO Inquiry VALUES ('I002', 'U005', 'AG002', '31-04-2024', 'User profile access
error', 'pererak2@gmail.com' );
INSERT INTO Inquiry VALUES ('I003', 'U005', 'AG004', '01-05-2024', 'Login error',
'pererak2@gmail.com' );
INSERT INTO Inquiry VALUES ('I004', 'U001', 'AG001', '28-03-2024', 'Connection error',
'nimal12@gmail.com' );
INSERT INTO Inquiry VALUES ('I005', 'U002', 'AG003', '18-02-2024', 'Inquire more
information about system', 'sunill@gmail.com' );

/*Data stored in SupportAgent table*/

INSERT INTO SupportAgent VALUES ('AG001', 'D.S.A Kumarasinghe', '1478asdf',
'daskumarasinghe@gmail.com', '11-04-1989');
INSERT INTO SupportAgent VALUES ('AG002', 'I.W.K Hasaranga', 'GYHN234',
'hasaranga@gmail.com', '28-11-1979');
INSERT INTO SupportAgent VALUES ('AG003', 'B.A.E Silva', '1452Asde',
'baesilva@gmail.com', '13-05-1986');
INSERT INTO SupportAgent VALUES ('AG004', 'Jenny de Dias', '89*12fg',
'jennydias@gmail.com', '01-09-1997');
INSERT INTO SupportAgent VALUES ('AG005', 'S.A.M Kumarasinghe', 'd5dfshd',
'samkumarasinghe@gmail.com', '25-12-1995');

/*Data stored in MealPlan table*/

INSERT INTO MealPlan VALUES ('0002','1', 'R005', 'R002', 'R004', ' ');
INSERT INTO MealPlan VALUES ('0002','1', 'R002', 'R003', 'R005', 'R003 ');
INSERT INTO MealPlan VALUES ('0002','1', 'R005', 'R001', 'R005', 'R002');
INSERT INTO MealPlan VALUES ('0002','1', 'R004', 'R002', 'R005', 'R003');
INSERT INTO MealPlan VALUES ('0002','1', 'R002', 'R003', 'R002', ' ');
```

```
/*Data stored in RecipeIngredient table*/


insert into RecipeIngredient values('R001','001');
insert into RecipeIngredient values('R002','002');
insert into RecipeIngredient values('R003','003');
insert into RecipeIngredient values('R004','004');
insert into RecipeIngredient values('R005','005');

/*Data stored in Category table*/

insert into Category values('C001','Vegan');
insert into Category values('C002','Non-Vegetarian');
insert into Category values('C003','Vegetarian');
insert into Category values('C004','Dessert');

/*Data stored in RecipeCategory table*/

insert into RecipeCategory values('C002','R001');
insert into RecipeCategory values('C002','R002');
insert into RecipeCategory values('C003','R003');
insert into RecipeCategory values('C003','R004');
insert into RecipeCategory values('C001','R005');
insert into RecipeCategory values('C003','R005');


/*Data stored in Agent_phoneNo table*/

insert into Agent_phoneNo values('AG002','0782345667');
insert into Agent_phoneNo values('AG001','0769845292');
insert into Agent_phoneNo values('AG001','0724890397');
insert into Agent_phoneNo values('AG004','0777468699');
insert into Agent_phoneNo values('AG005','0775547821');

/*Data stored in Administrator table*/

insert into Administrator values('AD001','Kamal Perera','kamalperera12@gmail.com','28-12-
1998','Perera123@');
insert into Administrator values('AD002','Supuni de Silva','supunisilva34@gmail.com','02-
07-2001','Perera123@');
insert into Administrator values('AD003','Dumindu Dias','dumindudias@gmail.com','17-01-
2000','Perera123@');
insert into Administrator values('AD004','Nimali
Gunathilake','nimaligunathilake@gmail.com','23-12-1994','Perera123@');
insert into Administrator values('AD005','Amanda
Mandakini','amandamandakini92@gmail.com','01-06-2002','Perera123@');

/*Data stored in Admin_phone table*/

insert into Admin_phone values('AD001','0741234522');
insert into Admin_phone values('AD002','0713397442');
insert into Admin_phone values('AD003','0710046330');
insert into Admin_phone values('AD004','0772284799');
insert into Admin_phone values('AD005','0784468737');
```

```
/*Data stored in Admin_manage_developer table*/

insert into Admin_manage_developer values('AD001','D001');
insert into Admin_manage_developer values('AD002','D002');
insert into Admin_manage_developer values('AD003','D003');
insert into Admin_manage_developer values('AD004','D004');
insert into Admin_manage_developer values('AD005','D005');


/*Data stored in Admin_manage_Agent table*/

INSERT INTO Admin_manage_Agent VALUES ('AD001', 'AG001');
INSERT INTO Admin_manage_Agent VALUES ('AD002', 'AG002');
INSERT INTO Admin_manage_Agent VALUES ('AD003', 'AG003');
INSERT INTO Admin_manage_Agent VALUES ('AD004', 'AG004');
INSERT INTO Admin_manage_Agent VALUES ('AD004', 'AG005');

/*Data stored in eCookbook table*/

INSERT INTO eCookbook VALUES ('CB001', 'Italian Cuisine', 'A collection of traditional
Italian recipes.', 'Maria Rossi', '2023-05-15', '15.99', 'ITAL');
INSERT INTO eCookbook VALUES ('CB002', 'Vegetarian Delights', 'Explore a variety of tasty
vegetarian dishes.', 'John Green', '2023-09-28', '12.50', 'VEGE');
INSERT INTO eCookbook VALUES ('CB003', 'Healthy Eating Habits', 'Learn how to incorporate
healthier choices into your diet.', 'Emily Smith', '2024-01-10', '10.99', 'HLTH');
INSERT INTO eCookbook VALUES ('CB004', 'Baking Basics', 'Master the fundamentals of
baking with these easy-to-follow recipes.', 'Michael Johnson', '2023-11-20', '18.75',
'BAKE');
INSERT INTO eCookbook VALUES ('CB005', 'Asian Fusion', 'Experience a fusion of flavors
from across Asia.', 'Li Chen', '2020-03-05', '14.25', 'ASIA');

/*Data stored in Cookbook_Purchase table*/

INSERT INTO Cookbook_Purchase VALUES ('0001', 1, 'CB001');
INSERT INTO Cookbook_Purchase VALUES ('0002', 2, 'CB003');
INSERT INTO Cookbook_Purchase VALUES ('0003', 3, 'CB002');
INSERT INTO Cookbook_Purchase VALUES ('0001', 4, 'CB004');
INSERT INTO Cookbook_Purchase VALUES ('0004', 5, 'CB001');
```

```
/*Data stored in Payment table*/

INSERT INTO Payment VALUES (1, 'Credit Card', 1599, '2024-04-01', '0001');
INSERT INTO Payment VALUES (2, 'PayPal', 1099, '2024-04-05', '0002');
INSERT INTO Payment VALUES (3, 'Credit Card', 1250, '2024-04-08', '0003');
INSERT INTO Payment VALUES (4, 'Debit Card', 1875, '2024-04-10', '0001');
INSERT INTO Payment VALUES (5, 'PayPal', 1599, '2024-04-15', '0004');

/*Data stored in Sales_Report table*/

INSERT INTO Sales_Report VALUES ('SR001', 'Italian Cuisine', 1599, 100, '2024-04-30',
'AD001', 800);
INSERT INTO Sales_Report VALUES ('SR002', 'Vegetarian Delights', 1099, 70, '2024-04-30',
'AD002', 600);
INSERT INTO Sales_Report VALUES ('SR003', 'Healthy Eating Habits', 1250, 80, '2024-04-
30', 'AD003', 700);
INSERT INTO Sales_Report VALUES ('SR004', 'Baking Basics', 1875, 120, '2024-04-30',
'AD001', 1200);
INSERT INTO Sales_Report VALUES ('SR005', 'Italian Cuisine', 1599, 100, '2024-04-30',

'AD004', 800);
```

Performance Requirements

Performance is another essential part of a system. It has some of the features expected of a good system.

Response Time

The recipe management system should provide search results within 2 seconds for efficient browsing of recipes.

Reservation Processing Time

The system should process recipe requests and generate cooking instructions in 5 seconds.

Availability Accuracy

Real-time accuracy of ingredient availability information is essential for meal preparation.

System Uptime

The system should have 99.9% uptime to ensure reliable access for meal planning and cooking.

Payment Processing Time

The system should process ingredient purchases securely and efficiently to provide a seamless shopping experience.

Data Retrieval Speed

The system should retrieve user recipe preferences and saved recipes in 2 seconds to provide personalized cooking recommendations.

Scalability

The system should be scalable to handle increased recipe and user data as it grows, ensuring consistent performance.

Concurrent User Handling

The system should be able to handle 100 simultaneous users without degradation to support multiple users accessing recipes concurrently.

Security Requirements

- Security is an important factor for an online recipe management system, especially as it relates to the customer's personal information such as sensitive information details.
- Users should be able to use secure login credentials to prevent unauthorized access to recipe collections and personal information.
- Role-based access control ensures users only have access to necessary features based on their role, such as chef, home cook, or guest.
- Sensitive data, including user profiles and saved recipes, should be encrypted to protect it from unauthorized access.
- Payment information for ingredient purchases should be securely handled and stored to prevent unauthorized access and ensure compliance with data protection regulations.
- The system should receive regular security updates to protect against potential threats and vulnerabilities.
- Strong password policies should be implemented to enforce secure user authentication and prevent unauthorized access to the system.
- The system should have regular backups of all recipe and user data to ensure data integrity and availability in case of system failures or data loss events.
- Prior to adding payment information to the system, verification by the payment processor must be conducted to ensure the validity and security of the information.