



UNIVERSITI TEKNOLOGI MALAYSIA
FACULTY OF COMPUTING, UTM
SEMESTER I, SESSION 2023/2024

PROJECT PHASE 2: DATABASE

CONCEPTUAL DESIGN

SECD2523 : DATABASE

SECTION 02

NAME :

- | | |
|--|-----------|
| 1. AINA NURAIN BINTI MOHD AROFF | B23CS0022 |
| 2. DANESH MUTHU KRISNAN | B23CS0034 |
| 3. MUHAMMAD FIRDAUS BIN MOHIDEEN ABDUL CADER | B23CS0056 |

COURSE : BACHELOR OF COMPUTER SCIENCE (COMPUTER NETWORKS AND SECURITY)

SUBMISSION DATE : 17 - DECEMBER - 2023

LECTURER'S NAME : DR. IZYAN IZZATI BINTI KAMSANI

TABLE OF CONTENTS

1.0	Introduction.....	2
2.0	Data Flow Diagram (DFD).....	3
2.1	Context Diagram.....	3
2.2	DFD Level-1.....	4
2.3	DFD Level-1 (Finder).....	5
3.0	Data and Transaction Requirement.....	6
3.1	Proposed Business Rules.....	6
3.2	Proposed Data and Transactional Rules.....	7
3.2.1	Data Requirement.....	7
3.2.2	Transaction Requirement.....	10
4.0	Database Conceptual Design.....	12
4.1	Conceptual ERD.....	12
4.2	Enhanced ERD (EERD).....	13
5.0	Data Dictionary.....	14
5.1	Description of Relationship.....	14
5.2	Description of Attributes in Entity.....	15
6.0	Summary.....	16

1.0 Introduction

When it comes to data management, one must also deal with the tedious process of designing a database system that fully satisfies the needs of the customer or organization. This is achieved through thorough planning across three phases, which include conceptual, logical and physical database design. For phase 2 of this project, we will be focusing on the first step in the database design process, which is the database conceptual design for our “Food Recipe Finder” system.

Conceptual database design is the process of identifying the essential data elements, relationships, and constraints in a data model, which represents a particular organization's business requirements. There are a few key objectives in creating a clear conceptual database design and that includes identifying the entities and their attributes, defining the relationships, and establishing the constraints. This will serve as a blueprint for the next 2 database design phases, and allow us to create a high-level data model that can reflect the business requirements needed to fulfill our “Food Recipe Finder” system. At this stage, the main focus will be solely on understanding the problem domain and defining the overall structure of the database without getting into technical implementation details. The process to create the conceptual database design encompasses a certain set of steps, which include requirements gathering, entity-relationship modeling, normalization and lastly review and feedback.

In a nutshell, this report will be focusing on understanding the problem domain and identifying entities and relationships in our “Food Recipe Finder” system. Phase 2 serves the purpose of further discussing the points proposed in our project proposal, as well as visualizing our initial database design in order to develop the most efficient and robust database for our system in the future phases. Each level of the database design process serves a unique purpose, and by following this structured approach, we can ensure that our final database meets the necessary data management needs effectively and efficiently.

2.0 Data Flow Diagram (DFD)

2.1 Context Diagram

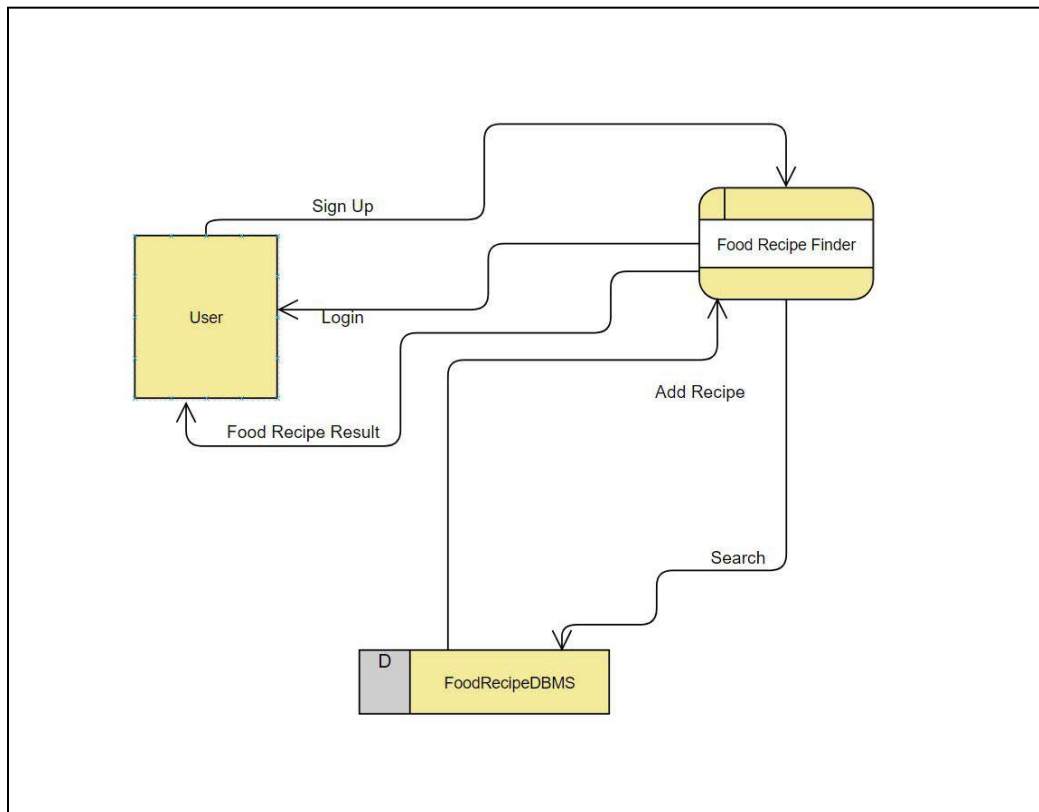


Figure 2.1.1: Context Diagram

2.2 DFD Level-1

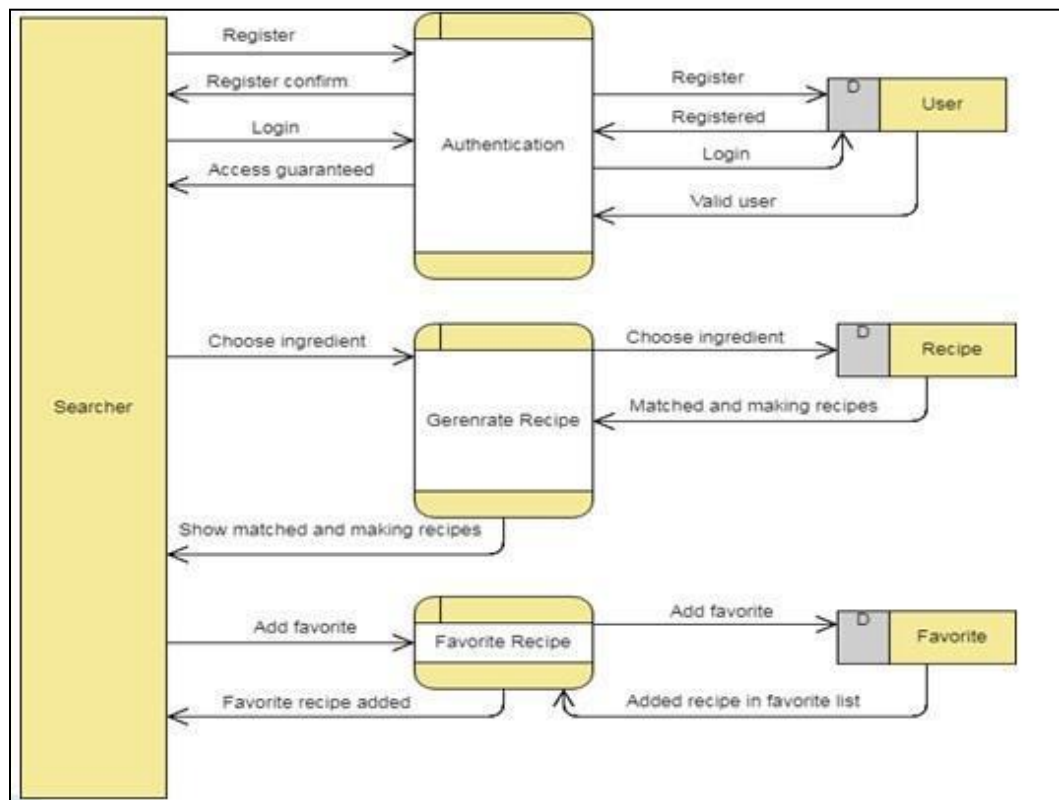


Figure 2.2.1: DFD Level-1

2.3 DFD Level-1 (Finder)

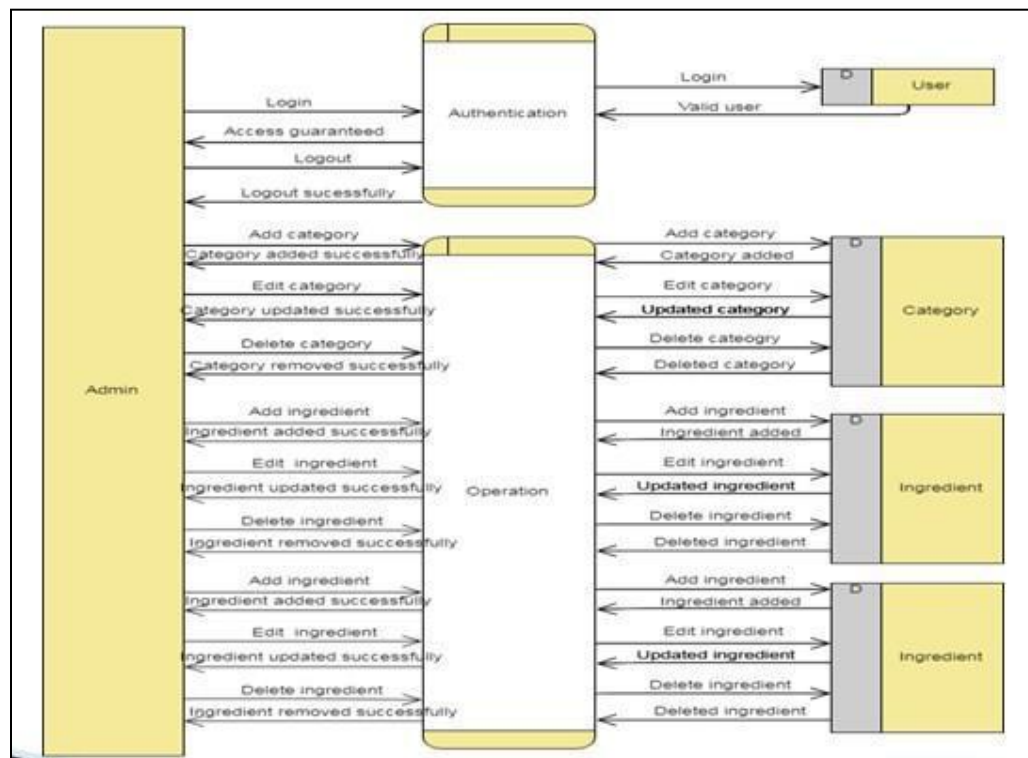


Figure 2.3.1: DFD Level-1 (Finder)

3.0 Data and Transaction Requirement

3.1 Proposed Business Rules

1. Only one recipe has one to many recipe courses.
2. Only one recipe has one to many recipe food categories.
3. One to many recipes has one to many cuisines.
4. One to many recipe courses have one to many courses.
5. One to many recipe food categories have one to many food categories.
6. One to many recipes have one to many levels.
7. One user searches for one recipe.
8. One to many recipes have one to many recipe nutritional information.
9. One to many recipes have one to many recipe ingredients.
10. One to many recipes have one to many menu recipes.
11. One to many nutritional information uses one to many measurements.
12. One to many recipe ingredients have one to many measurements.
13. One to many recipe ingredients have one to many ingredients.
14. One to many menu recipes have one to many menus.

3.2 Proposed Data and Transactional Rules

Data and transactional rules refers to data entry, data update/deletion, and data query operations in a database system. For our “Food Recipe Finder” system, there will be several data and transactional rules, such as:

3.2.1 Data Requirement

Users

Someone who uses the “Food Recipe Finder” system is considered a user. They are identified through a unique userID and will be required to fill in their first name, last name, password, phone number and email.

Recipe

Each recipe will be designated a unique recipeID. All the recipes will require information such as name, preparation time, cook time, duration, calories per serving and ratings. The level and cuisine for each recipe is also referred to through the foreign keys rLevelID and rCuisineID in their respective tables.

Recipe Course

The recipe course will be identified with its own rcRecipeCourseID, and will refer to the recipe and course tables through the rcRecipeID and rcCourseID foreign keys.

Course

Each course is represented by its own unique cCourseID and includes the type of course.

Recipe Food Categories

Recipe food categories will be identified with its unique ID called rfcID, and will refer to the recipe and food category tables through the rfcRecipeID and rfcFoodCategoryID foreign keys.

Food Categories

Each food category is represented by its own unique fcFoodCategoryID and includes information on the category of the food.

Cuisines

Each cuisine is represented by its own unique cCuisineID and includes information on the type of cuisine.

Levels

Each difficulty level is represented by its own unique lLevelID and includes the type of difficulty level.

Recipe Nutritional Information

Recipe nutritional information will be identified with its unique ID called rniID, and will refer to the recipe and nutritional information tables through the rniRecipeID and rniNutritionalInformationID foreign keys.

Nutritional Information

The nutritional information will be identified with its own niNutritionalInformationID, and will include information on each recipe's nutritions. It will also refer to the measurement table through the niMeasurementID foreign keys.

Recipe Ingredients

Recipe ingredients will be identified with its own riRecipeIngredientID, and will refer to the recipe, measurement and ingredient tables through the riRecipeID, riMeasurementID and riIngredientID foreign keys.

Ingredients

Each ingredient is represented by its own unique iIngredientID and includes the name of the ingredient.

Measurements

Each measurement is represented by its own unique mMeasurementID and includes information on the measurements and its abbreviations.

Menu Recipes

Menu recipes will be identified with its own ID called mrMenuRecipes, and will refer to the menu and recipe tables through the mrMenuID and mrRecipeID foreign keys.

Menus

Each menu is represented by its own unique mMenuID and includes the name of the menu.

3.2.2 Transaction Requirement

Data Entry:

1. Enter the email address of users.
2. Enter the password of users.
3. Enter the profile information of users.
4. Enter the submitted recipe description.
5. Enter the submitted recipe ingredients.
6. Enter the submitted recipe name.
7. Enter the submitted recipe category.
8. Enter the submitted recipe nutritional information.
9. Enter the submitted recipe cooking instructions.
10. Enter the submitted recipe measurements.
11. Enter the recipe ratings.
12. Enter bookmarked recipes.

Data Update/Delete:

1. Update/delete the email address of users.
2. Update/delete the password of users.
3. Update/delete the profile information of users.
4. Update/delete the submitted recipe description.
5. Update/delete the submitted recipe ingredients.
6. Update/delete the submitted recipe name.
7. Update/delete the submitted recipe category.
8. Update/delete the submitted recipe nutritional information.

9. Update/delete the submitted recipe cooking instructions.
10. Update/delete the submitted recipe measurements.
11. Update/delete the recipe ratings.
12. Update/delete bookmarked recipes.

Data Query:

1. List the details of users.
2. Display a list of all the recipes.
3. List the recipes based on category.
4. List the recipes based on ingredients.
5. List the recipes based on the recipe name.
6. List the details of each recipe.
7. Display list of bookmarked recipes.
8. Display list of submitted recipes.
9. Display list of recipe ratings.

4.0 Database Conceptual Design

4.1 Conceptual ERD

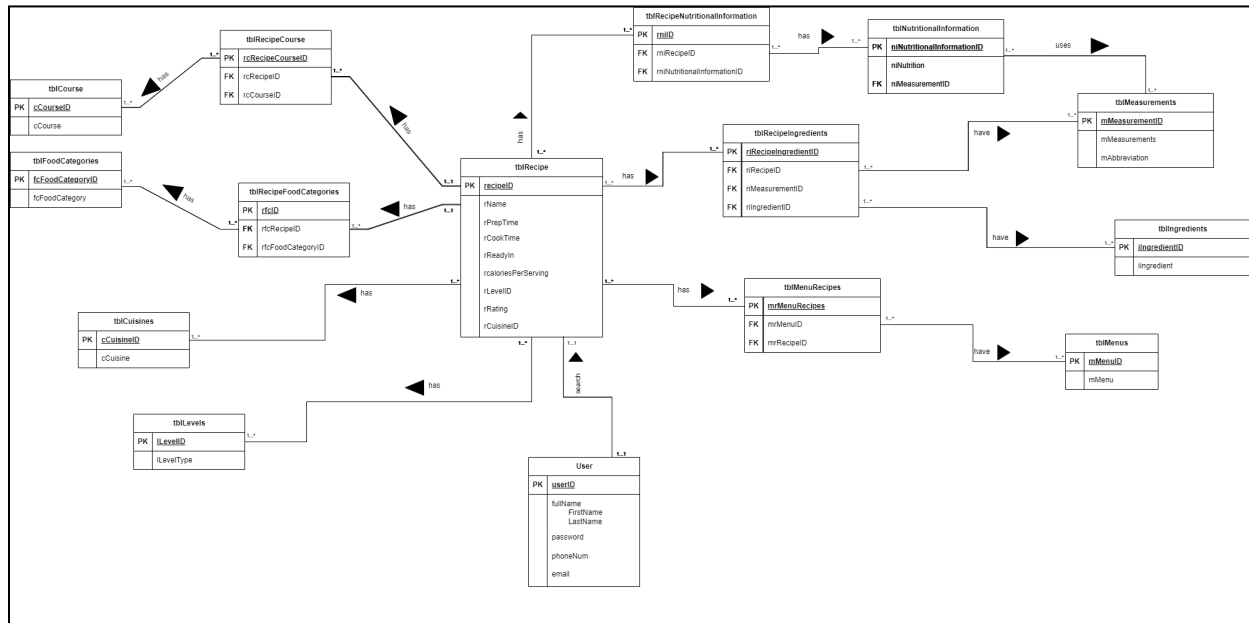


Figure 4.1.1: Conceptual ERD

4.2 Enhanced ERD (EERD)

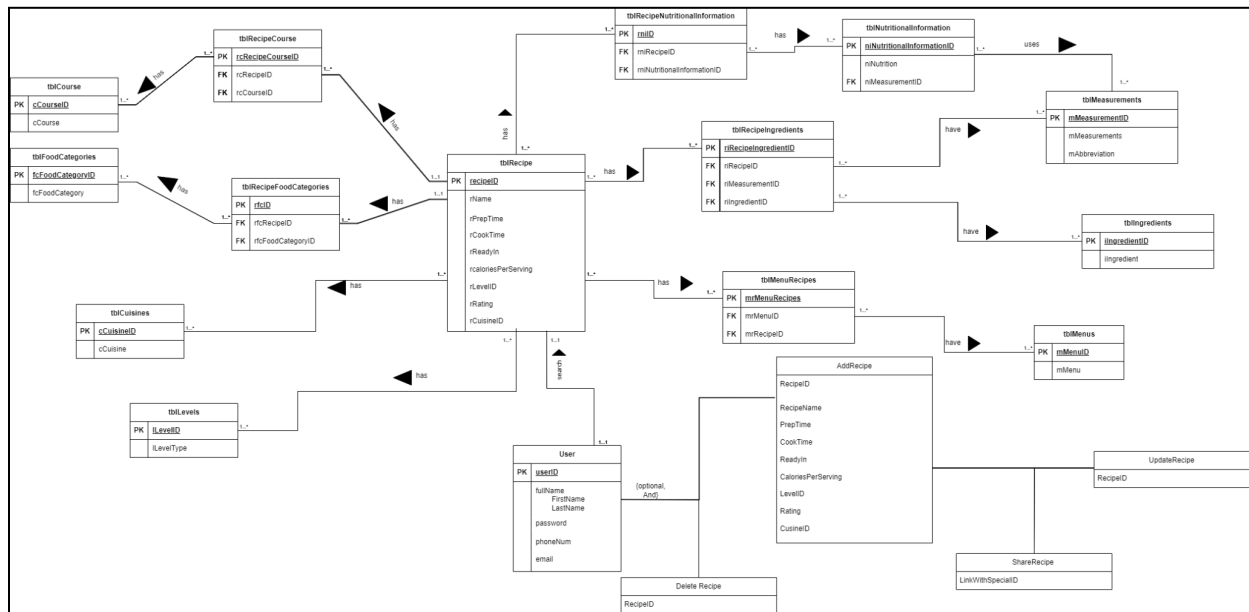


Figure 4.2.1: Enhanced ERD (EERD)

5.0 Data Dictionary

5.1 Description of Relationship

Relationship Description:					
#	Entity	Multiplicity	Relationship	Entity	Multiplicity
1	user	1...1	search	tblRecipe	1...1
2	tblRecipe	1...*	has	tblLevels	1...*
3	tblRecipe	1...*	has	tblCuisines	1...*
4	tblRecipe	1...1	has	tblRecipeFoodCategories	1...*
5	tblRecipe	1...1	has	tblRecipeCourse	1...*
6	tblRecipeFoodCategories	1...*	has	tblFoodCategories	1...*
7	tblRecipeCourse	1...*	has	tblCourse	1...*
8	tblRecipe	1...*	has	tblRecipeNutritionalInformation	1...*
9	tblNutritionalInformation	1...*	use	tblMeasurements	1...*
10	tblRecipe	1...*	has	tblRecipeIngredients	1...*
11	tblRecipeIngredients	1...*	have	tblMeasurements	1...*
12	tblRecipeIngredients	1...*	have	tblIngredients	1...*
13	tblRecipe	1...*	has	tblMenuRecipes	1...*
14	tblMenuRecipes	1...*	have	tblMenus	1...*

Figure 5.1.1: Relationship Description

5.2 Description of Attributes in Entity

Attributes Description:

#	Entity	Attribute	Data Type	Field Size	Key	Description
1	tblRecipe	recipeID	VARCHAR	50	PK	Recipe unique identification number
		rName	VARCHAR	50		Recipe Name
		rPrepTime	NUMBER			Preparation time in minutes
		rCookTime	NUMBER			Total time (preparation + cook) in minutes
		rReadyIn	NUMBER			Cooking time in minutes
		rcaloriesPerServing	NUMBER			Total calories per serving in grams
		rLevelID	NUMBER			Difficulty level unique identification number
		rRating	DECIMAL	(2,1)		Rating of recipe
		rcuisineID	VARCHAR	255		Cuisine types of unique identification number
2	User	userID	VARCHAR	50	PK	User unique Identification number
		fullName				
		FirstName	VARCHAR	50		First name of user
		LastName	VARCHAR	50		Last name of user
		password	VARCHAR	50		Password of user account
		phoneNum	NUMBER			User phone number
3	tblCuisines	cCuisineID	VARCHAR	50	PK	Cuisine type unique identification number
		cCuisine	VARCHAR	50		Cuisine type
4	tblLevels	lLevelID	VARCHAR	50	PK	Level type unique identification number
		lLevelType	VARCHAR	50		Difficulty level of the recipe
5	tblRecipeFoodCategories	rfcID	VARCHAR	50	PK	Recipe food categories unique identification number
		rfcRecipeID	VARCHAR	50	FK	Recipe Unique Identification number
		rfcFoodCategoryID	VARCHAR	50	FK	Food categories unique identification number
6	tblFoodCategories	fcFoodCategoryID	VARCHAR	50	PK	Food category unique identification number
		fcFoodCategory	VARCHAR	50		Food category list (Beverages, grilling, desserts)
7	tblRecipeCourse	rcRecipeCourseID	VARCHAR	50	PK	Recipe course unique identification number
		rcRecipeID	VARCHAR	50	FK	Recipe unique identification number
		rcCourseID	VARCHAR	50	FK	course unique identification number
8	tblCourse	cCourseID	VARCHAR	50	PK	Course unique identification number
		cCourse	VARCHAR			Course list (lunch, dinner)
9	tblMenuRecipes	mrMenuRecipe	VARCHAR	250	PK	The list of recipes in menu
		mrMenuID	VARCHAR	50	FK	Menu unique identification number
		mrRecipeID	VARCHAR	50	FK	Recipe unique identification number
10	tblMenus	mMenuID	VARCHAR	50	PK	Menu unique identification number
		mMenu	VARCHAR	250		Menu holds recipes
11	tblRecipeIngredients	riRecipeIngredientID	VARCHAR	50	PK	Recipe ingredient unique identification number
		riRecipeID	VARCHAR	50	FK	Recipe unique identification number
		riMeasurementID	VARCHAR	50	FK	Measurement unique identification number
		riIngredientID	VARCHAR	250	FK	Ingredient unique identification number
12	tblMeasurements	mMeasurementsID	VARCHAR	50	PK	Measurement unique identification number
		mMeasurements	VARCHAR	50		Measurment of ingredients and nutrition
		mAbbreviation	VARCHAR	50		Abbreviation such as grams and cup for ingredients and measurement
13	tblIngredients	iIngredientID	VARCHAR	50	PK	Ingredient unique identification number
		iingredient	VARCHAR	50		Ingredients list
14	tblRecipeNutritionalInformation	rniID	VARCHAR	50	PK	Recipe nutritional information unique identification number
		rniRecipeID	VARCHAR	50	FK	Recipe unique identification number
		rniNutritionalInformationID	VARCHAR	50	FK	Nutritional information unique identification number
15	tblNutritionalInformation	niNutritionalInformationID	VARCHAR	50	PK	Nutritional information unique identification number
		niNutrition	VARCHAR	50		List of nutrition
		niMeasurementID	VARCHAR	50	FK	Measurement unique identification number

Figure 5.2.1: Attributes Description

6.0 Summary

To preface, the “Food Recipe Finder” system is a versatile and intuitive platform designed to transform how cooking enthusiasts find, manage and exchange recipes. Using a sophisticated search feature, users can effortlessly explore a wide range of foods, filtering options by ingredients, recipe name, category, cuisine, etc. The system enables convenient storage and classification of recipes, ensuring easy retrieval for future culinary adventures. Additionally, its emphasis on community involvement provides an interactive space where users can share their favorite foods, creating a collaborative and diverse culinary environment. It offers a wide variety of tastes, food preferences and cooking styles. This program stands out as a must-have tool for cooking enthusiasts, professional and individual. As a conclusion, with our system proposal in mind, we managed to develop a robust conceptual database design that will be the building blocks for the next phases in this project which encompass the logical and physical database design. In this phase, we managed to identify the entities and their attributes, define the relationships, and establish the constraints present in the database design. This will hopefully be a sufficient blueprint to develop the final database in the future.