Design Specification Document

Group 14

Members:

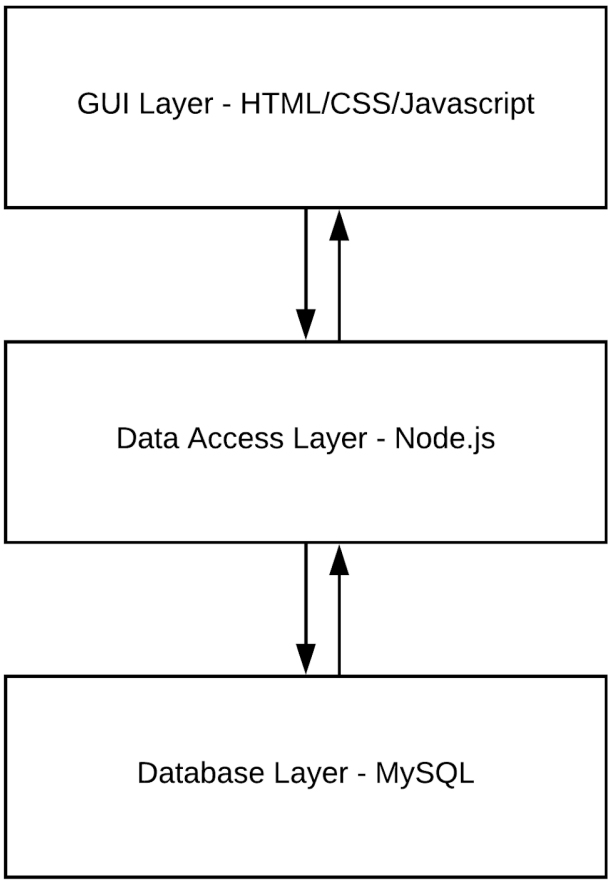
Marek Kracl, Mac Ozanne, Linlin Chen, Qiming Yuan, Crystal Warta

1 . Introduction

The purpose of this design document is to effectively record the architecture and design of the finished product, “Bacon Search”. Entity relation diagrams showing the relationship between different parts of the system, as well as the relationships between tables in the database will be given in the document. The audience of this document is software engineers and system architects who will be implementing and maintaining the described project.

2 Architecture

2.1 Introduction



The high level architectural design of the system will be a layered model. The lowest level of this model will be a MySQL database to store the permanent information such as user info, recipes, and ingredients. This data can be read and written to via the data access layer that will convert the data into a form that can be easily manipulated. The top layer is the GUI interface that users will use to interact with the system.

2.2 Modules

2.2.1 Database layer

The database layer is responsible for holding all the persistent data for the system and defining the relationships between this data. The system will be using a MySQL database that can easily be incorporated into the project via Node.js. The data relationships are described in more detail below in section 3.1.2.

2.2.2 Data Access Layer

This module is responsible for taking the user input and querying the database to return matching Recipes. This will be accomplished via Node.js which has built-in functionality for database querying.

2.2.3 GUI Layer

This is the primary layer that the user will interact with and is responsible for generating the user interface. It is in this layer that the user will be able to login to the system to access their saved recipes as well as search for new recipes containing selected ingredients.

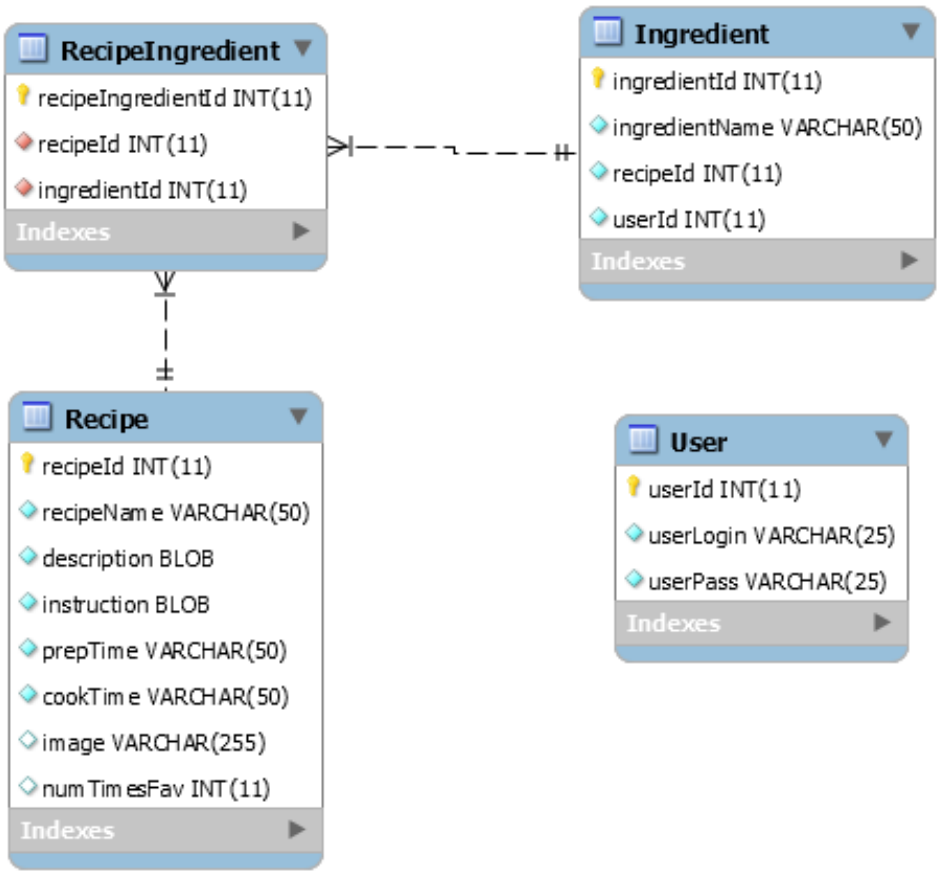
A combination of HTML, CSS, Javascript will be used in this layer to dynamically generate user web pages from the data and allow for seamless user interaction. Users will also be able to interact with the GUI layer to select ingredients to be queried, which will be sent back down the layers and returned to display relevant recipes to the user. In the future phases, the user will be able to view and populate a shopping list section and the user interface will be upgraded to be more visually appealing.

3 Class Diagrams

3.1 Data Table Classes

The system will be using a MySQL database to hold all the persistent data necessary. This includes information on users, ingredients, and recipes, as well as the relationships between them. The figure below shows each table in the database with its respective columns as well as the relationships between the different tables

3.1.1 Schema



3.1.2 Schema Information

The data in the database shall be organized in the following tables and columns

User: Holds the data for a single user of the system including information such as their username and password.

Ingredient: This table holds all relevant information regarding each individual ingredient. Mainly the name of said ingredient.

Recipe: This table will represent an individual recipe. It will contain the name of the recipe, the description, the cooking instructions, the prep time, the cook time, and an image if applicable. It will also contain a field to keep track of how many times that recipe has been favorited.

RecipeIngredient: This is a join-table between the recipe and any ingredients it may have.

3.2 Class Information

Classes will be implemented in the node.js data access layer of the system and used throughout the application. These classes will replicate the database schema above with each table corresponding to a different node.js class. For example, a user class will be implemented with variables for userLogin and userPass. The logic layer will take these classes and pass the information up to the GUI layer where it is displayed to the user.

3.3 GUI Layer

The top layer of the system consists of only four pages for the initial phase, a login page, a create account page, a search/selection page, and a results page. All will be implemented in HTML/CSS with Javascript for the necessary logic.

The login page will contain two fields; a username and password, and a submit button. If the user is not in the database, they will be redirected to a create-account page, where they will be able to input their desired information, and then redirected back to the login page to log in. After logging in, the user will be presented with a checklist of possible ingredients and a submit button. Upon submission, the user will be presented with recipes matching the ingredients checked by the user on the previous page.