**Design Document for KitchenMingle**

Group LG\_116

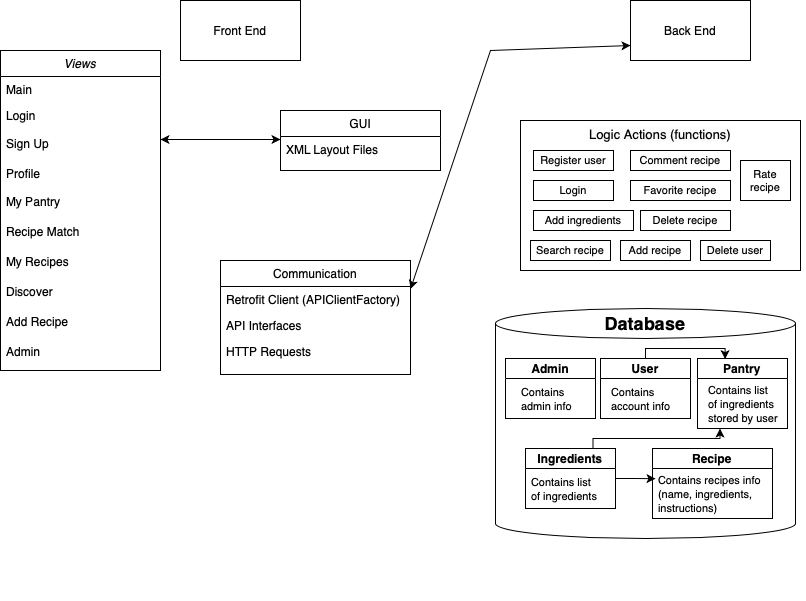
Christy: 25% contribution

Jesus Soto: 25% contribution

Kat: 25% contribution

Xuan Wen: 25% contribution

PUT THE BLOCK DIAGRAM PICTURE ON THIS PAGE! (Create the picture using pencil or drawIO)



Use this third page to describe complex parts of your design.

**Frontend** (currently implemented)

*activity\_main*

* Main has generates a page with the following elements:
  + TextView: KitchenMingle
  + Button: btnToSignUp
  + Button: btnToLogin
* Clicking either the btnToSignUp or btnToLogin takes you to either the sign up page or login page

*activity\_signup* (User)

* Has the user create an account, generates a page with the following elements:
  + EditText: confirmPasswordEditText
  + EditText: usernameEditText
  + EditText: passwordEditText
  + EditText: emailEditText
  + Button: btnToMain
  + Button: btnSignUp
* The sign up page takes user input of their username, password, email, and the password confirmation. It will take the user to the recipe creation page.

*activity\_login* (User)

* Login creates an page with the following elements:
  + Button: btnLogin
  + Button: btnToMain
  + EditText: passwordEditText
  + EditText: emailEditText
* The login page takes user input of their password and email. It will take the user to the recipe creation page.

*activity\_indredient*

* generates a page with the following elements:
  + EditText: recipeName
  + TextView: activity\_main\_textView1
  + Button: activity\_main\_post\_by\_body\_button
  + EditText: activity\_main\_recipename\_editText
  + TableLayout: tableLayout
  + TableRow: ingredientName1
  + TextView: quantityTextView
  + Button: increase1
  + Button: decrease1
  + TextView: textViewMyPantry
  + Button: button

**Backend**

*Communication*

Backend uses mappings to update the database based on the information sent to the given mappings’ URLs.

* GET: Request information that’re already stored in the database
* POST: Send information on an object(user, recipe, etc.) to be added to the database
* PUT: Send information to update a specific item in the database
* DELETE: Send an identifier to remove a specific item in the database

*Controllers*

Controllers contain the mappings for communication between frontend and the database.

* Users: Creates (register) users, contain one-to-one relationship with pantry
* Pantry: Stores ingredient based on user input, connected to the Users table
* Ingredient: To add list of ingredients to the database that will be used by Pantry and Recipe
* Recipe: Stores list of recipes, Users(Editor) are allowed to add new recipe that will be added to the database
* Admin: Has access to the database to perform changes such as delete users.

PUT THE TABLE RELATIONSHIPS DIAGRAM on this fourth page! (Create the picture using MySQLWorkbench)

A diagram of a recipe

Description automatically generated