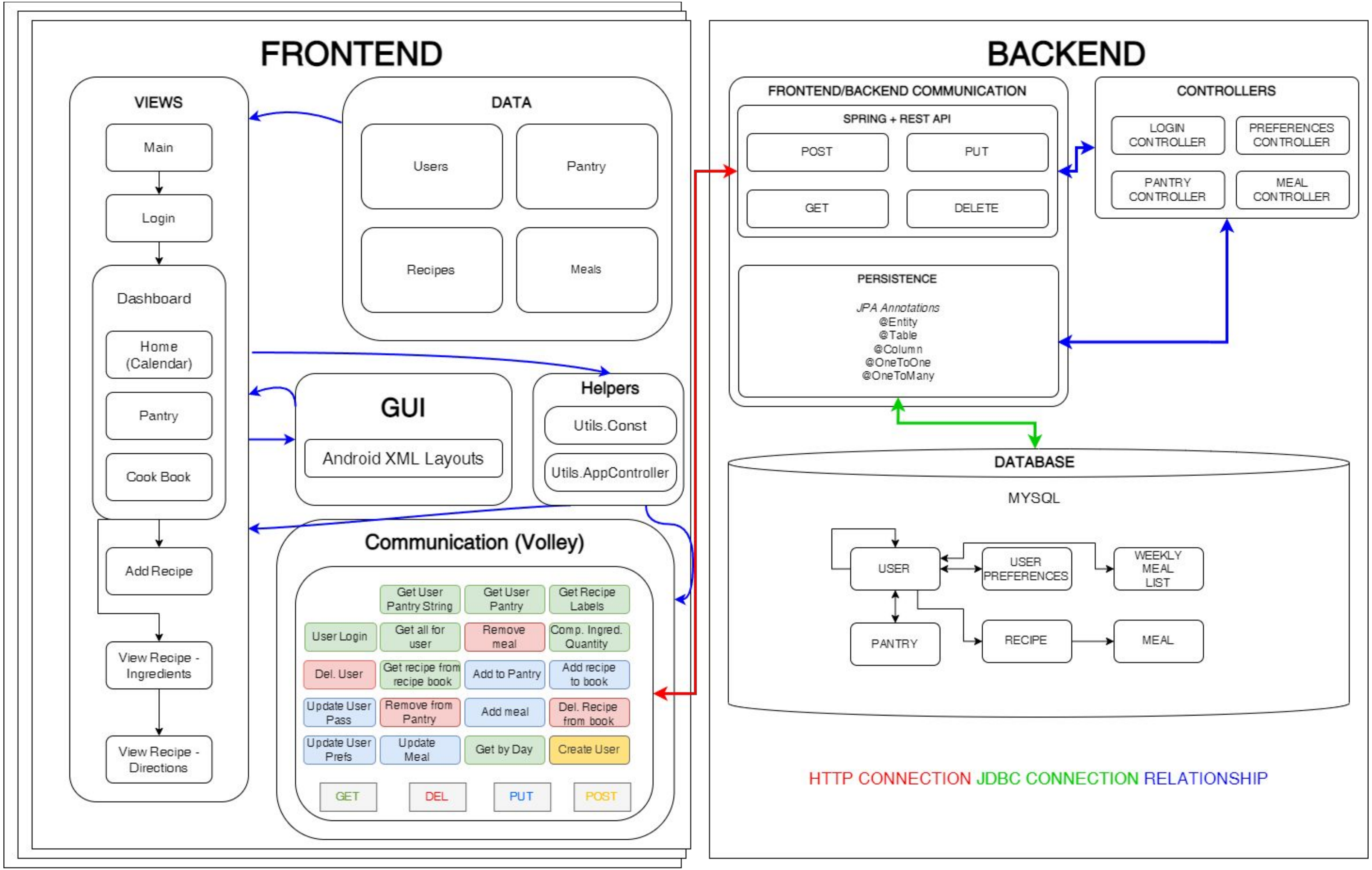
Design Document for FoodTime

Group 1_HB_6

Anthony Manschula - 25%
Blake Hardy - 25%
Henry Shires - 25%
Will Maahs - 25%



FRONTEND

VIEWS

- Main
- Login
- Dashboard
  - Home (Calendar)
  - Pantry
  - Cook Book
- Add Recipe
- View Recipe - Ingredients
- View Recipe - Directions

DATA

- Users
- Pantry
- Recipes
- Meals

GUI

- Android XML Layouts

Helpers

- Utils.Const
- Utils.AppController

Communication (Volley)

	Get User Pantry String	Get User Pantry	Get Recipe Labels
User Login	Get all for user	Remove meal	Comp. Ingred. Quantity
Del. User	Get recipe from recipe book	Add to Pantry	Add recipe to book
Update User Pass	Remove from Pantry	Add meal	Del. Recipe from book
Update User Prefs	Update Meal	Get by Day	Create User

GET DEL PUT POST

BACKEND

FRONTEND/BACKEND COMMUNICATION

SPRING + REST API

- POST
- PUT
- GET
- DELETE

PERSISTENCE

JPA Annotations
@Entity
@Table
@Column
@OneToOne
@OneToMany

CONTROLLERS

- LOGIN CONTROLLER
- PREFERENCES CONTROLLER
- PANTRY CONTROLLER
- MEAL CONTROLLER

DATABASE

MYSQL

HTTP CONNECTION JDBC CONNECTION RELATIONSHIP

Frontend (Currently Implemented)

Main (Child, Parent, Admin) - Welcome the user and provide a landing page

- Button: Get Started ➔ Send user to Login

Login (Child, Parent, Admin) - Login an existing user or create a new account. Takes input of email and password and login (if the user exists). For new accounts, an account role (Child, Parent, or Admin) and the username of the parent account is additionally required

- EditText: E-mail
- EditText: Password
- RadioButton: Child Role
- RadioButton: Parent Role
- RadioButton: Admin Role
- EditText: Parent Username
- Button: Create Account
- Button: Login

Calendar (Child, Parent, Admin) - Show the user their meal schedule at a week-by-week glance, with a detailed description for each day when clicked

- CalendarView: Weekly Calendar
- ListView: Meals list to show meals scheduled for the selected day
- Button: Schedule meals from the cookbook to a day of the week on the calendar

Frontend Cont. (Currently Implemented)

***Pantry (Child, Parent, Admin)* -** View a list of ingredients and groceries currently available to the user added by the user

- EditText: Add Ingredient Name
- EditText: Add Ingredient Quantity
- EditText: Add Quantity Type

***Cookbook (Child, Parent, Admin)* -** View a list of all recipes available for the user to schedule as a meal, added by the user. Each recipe also contains a list of required ingredients and directions that are compared with the available ingredients in the pantry

- CardView: View list of recipes as clickable cards with names and thumbnails
- Button: Add recipe to Cookbook (Parent/Admin only) → Navigates user to AddRecipeActivity

***AddRecipe (Parent, Admin)* -** Add a new recipe to the user’s cookbook, specifying ingredients and directions

- EditText: Ingredient names, quantities, quantity types
- EditText: Directions for cooking the recipe
- Button: Save and submit recipe to database

***ViewRecipe (Parent, Admin)* -** Not implemented yet

Backend

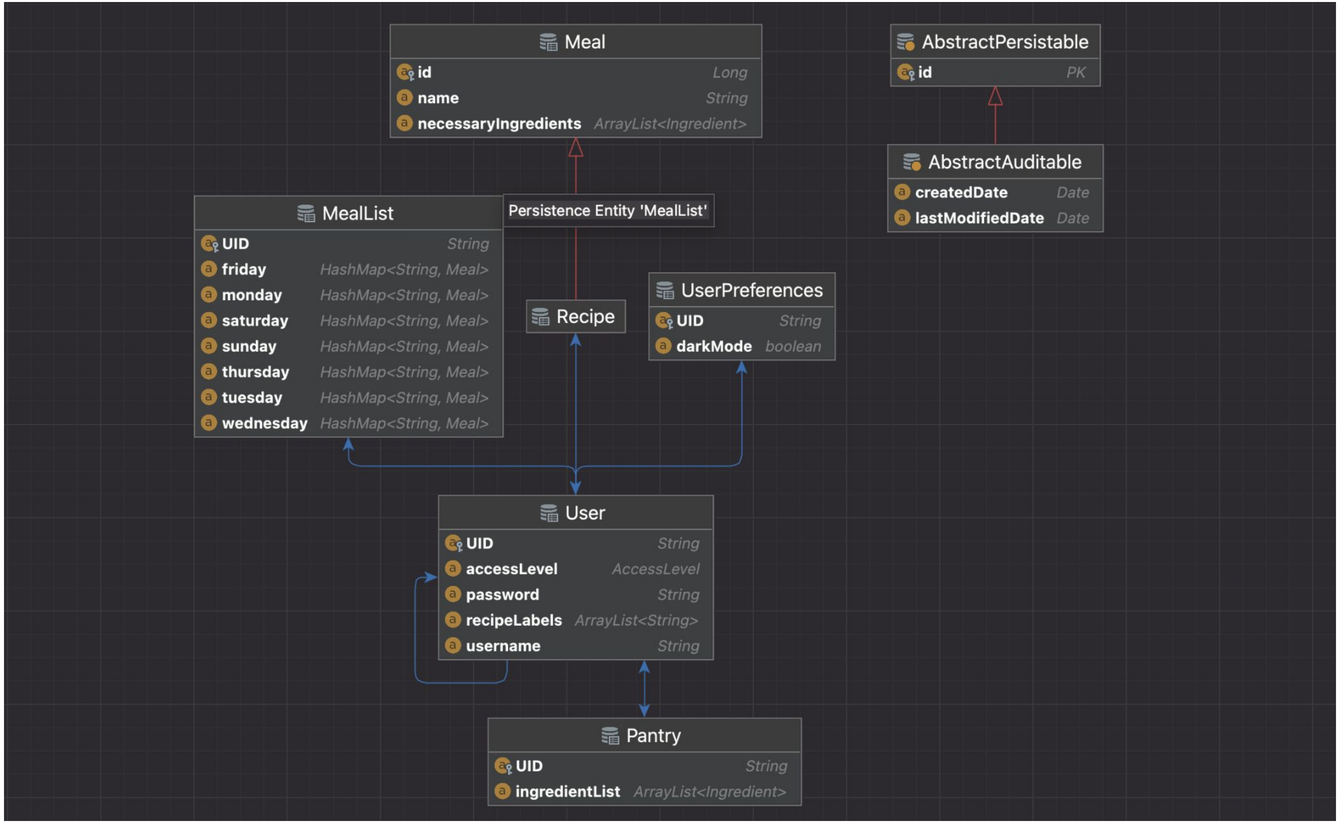
Communication:

- Backend uses standard HTTP *POST*, *GET*, *PUT*, and *DELETE* mappings for our REST controllers

Controllers:

- User: used for login and to associate relevant data. Contains one-to-one relationships to: userPreference, Pantry, MealList and one-to-many relationship with: RecipeBook
- Meal: used to store lists of ingredients and other data relevant to creating a meal
- Pantry: stores ingredients associated with user to be used in meals
- UserPreferences: stores data for user app preferences like dark mode and access levels. One-to-one with: User.

Table Relationship Diagram



Meal	
id	Long
name	String
necessaryIngredients	ArrayList<Ingredient>

AbstractPersistable	
id	PK

AbstractAuditable	
createdDate	Date
lastModifiedDate	Date

Persistence Entity 'MealList'

MealList	
UID	String
friday	HashMap<String, Meal>
monday	HashMap<String, Meal>
saturday	HashMap<String, Meal>
sunday	HashMap<String, Meal>
thursday	HashMap<String, Meal>
tuesday	HashMap<String, Meal>
wednesday	HashMap<String, Meal>

Recipe

UserPreferences	
UID	String
darkMode	boolean

User	
UID	String
accessLevel	AccessLevel
password	String
recipeLabels	ArrayList<String>
username	String

Pantry	
UID	String
ingredientList	ArrayList<Ingredient>