Final project FODMAP app documentation

Application is run from we/final\_project\_thymeleaf\_JDBC/FinalProjectThymeleafJdbcApplication.java, the rest of the application components are in their respective packages in we/final\_project\_thymeleaf\_JDBC.

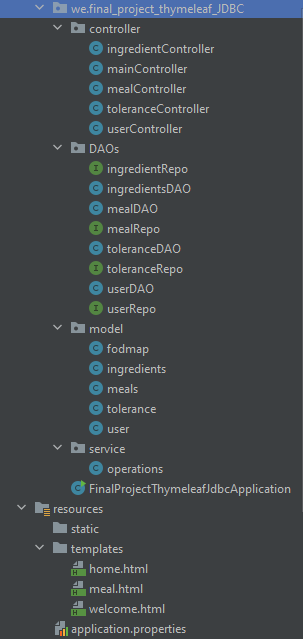
# Landing page:

http://localhost:8080/home

Features

* FODMAP calculator at http://localhost:8080/meal
  + Takes name of dish, list of ingredients and quantities in format “ingredient1: Xg, ingredient2: Yg, …" - there must be at least two ingredients
  + Separates ingredients from quantities in Service layer
  + The ingredients and quantities are linked in a hash map
  + Ingredients are searched in database
  + FODMAP content of each ingredient is calculated as: FODMAP\_content\_in\_database \* quantity\_in\_dish/quantity\_in\_databse
  + FODMAP content of dish is calculated by summing up the FODMAP content of all ingredients
  + FODMAP content of dish is displayed
* See all ingredients at localhost:8080/ingredient/ingredients
  + Displays all ingredients in the database
* Personalization
  + The program asks for the user’s name and displays it on the next page

MVC structure



Unit testing

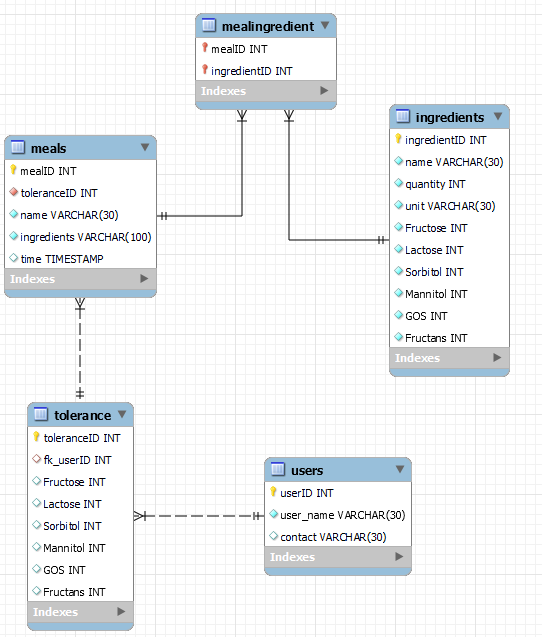
* Ingredient Repository and meal Repository
* Input validation for ingredient list

Version control system: none

Database

Can be created by script “Create fodmap database.sql”.

* ingredients- stores the amount of each foodmap for each **food** in the database
* Tolerance – the tolerance each **user** has to each fodmap
* meals– **food** journal logging meals eaten in the last x nr of hours
* Users – **user** info
* Meal ingredients – crossover table to avoid many to many relasionship



# Technologies used

Thymeleaf

JPA Repository - CRUD operations

JDBC template - custom queries

Spring Boot

Maven

Bugs

~~AddAttribute method not found~~ – fixed – the model imported automatically was not from springboot, but logback.core

~~JPARepository not working due to fodmap superClass–~~ fixed – used @MappedSuperclass on the fodmap class and made a default constructor (empty constructor for it. This allowed the child classes to have default constructors which JPA needs

~~Input validation process doesn’t recognize ingredients made of more than one word~~ – fixed – added a space in the ragex from [A-Za-z] to [A-Za-z ].

The get operations (find all and find by ID) of meals, tolerances and users don’t work properly, they all return an error message where about the SQL quarry where every word in the query has u1\_0] or t1\_0]. Operations like add work fine for meals, but not users. Custom queries work fine. The ingredient repository works well. - can be seen in testMealRepo().

Due to the bug above I set the currentUserID to 1, this can’t be changed by the user, so all meals are logged under the same userID.