Status Summary

Project Title: FlexFit

Names of all team members: Alam Bushara, Cole Bairley , David Ustyan

Work Done:
Alam (User): I developed the User component for the FlexFit application, creating a system that distinguishes between adult and child users. This included designing an abstract User class with specific subclasses for Adult and Child, each with age-appropriate validations. A key feature is the input method using Java's Scanner class to capture user data and a factory method, createUser(), that automates user type determination based on age. Additionally, I implemented a BMI calculation adjusted for imperial units and outlined a strategy for unit testing. This foundational work ensures the FlexFit app can efficiently manage diverse user data and adapt to future enhancements.

David (Nutrition): I developed the nutrition/food portion of the FlexFit application. The class stores essential dietary information such as the food's name, calories, and macronutrients (carbs, fats, proteins). I used private attributes for these details, a constructor to initialize them, and getter methods for access. Additionally, I included a toString method for easy output of the food's information. This design ensures data security and straightforward access, essential for maintaining a clean and efficient base.

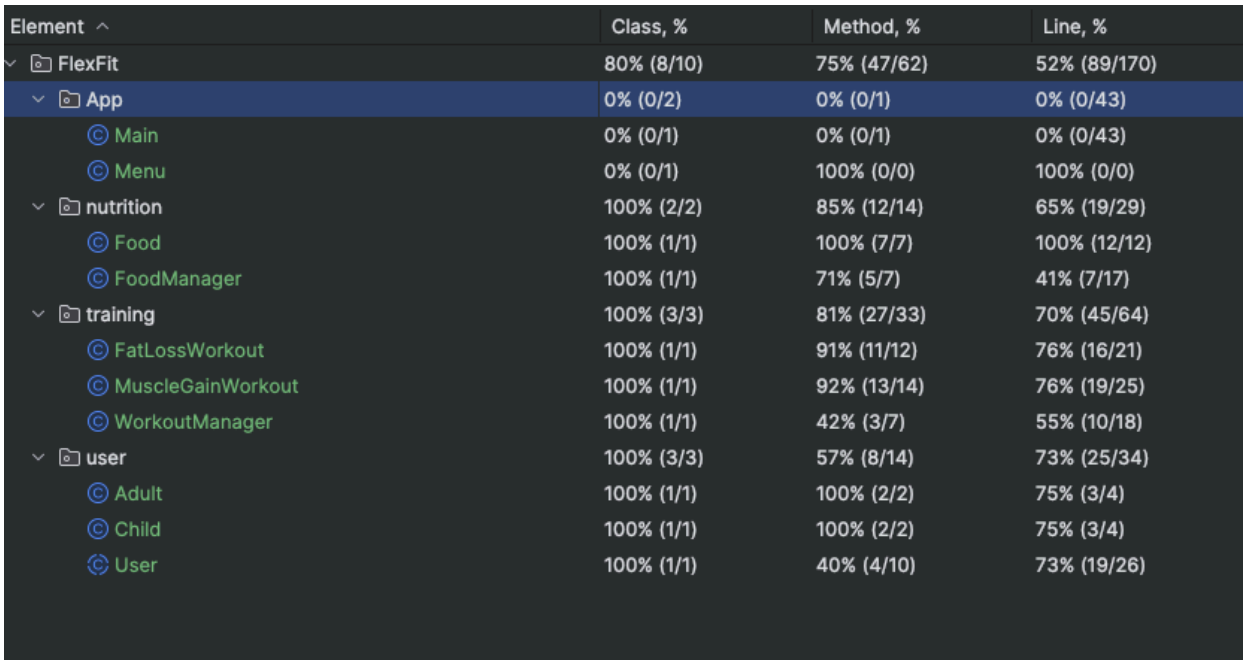
Cole (Training): I developed the training part of the FlexFlit application, which allows the user to either add workouts into their routine that they enjoy or choose from our curated list of workouts that are pre-programmed into the app. Additionally, the workouts are split into subsections one of them being workouts that focus on muscle growth and the other being workouts that focus on losing weight. This allows the user to have a more personalized workout program to reach their goals. To go more in-depth into this sector of flex fit each workout has attributes of time, weight used, calories burned, type of workout, etc., which allows the user to know exactly what workout is being performed.

Changes or Issues Encountered: After chatting with the course mentors during office hours, we agreed to focus on the Java parts, especially on using design patterns and keeping the logic straightforward. We've dropped Node.js, Docker, and React since they weren't needed for what we're aiming to do. We also decided to cut back on some of the extra features like Authentication and Social Interaction to keep things manageable.

Patterns: Now that you have more of your system implemented, please describe the use of design patterns so far in your prototype and how they are helping you or your design:
So far we have used the facade pattern in our WorkoutManager and FoodManager classes. These classes encapsulate the complexity of managing lists of multiple food and workout types

and allowing us to operate on them. Additionally, this makes it so that other parts of our application can interact with the food and workout types much more simplistically allowing for an overall better application.

Test coverage report showing coverage on at least 80% of the methods



Element	Class, %	Method, %	Line, %
FlexFit	80% (8/10)	75% (47/62)	52% (89/170)
App	0% (0/2)	0% (0/1)	0% (0/43)
Main	0% (0/1)	0% (0/1)	0% (0/43)
Menu	0% (0/1)	100% (0/0)	100% (0/0)
nutrition	100% (2/2)	85% (12/14)	65% (19/29)
Food	100% (1/1)	100% (7/7)	100% (12/12)
FoodManager	100% (1/1)	71% (5/7)	41% (7/17)
training	100% (3/3)	81% (27/33)	70% (45/64)
FatLossWorkout	100% (1/1)	91% (11/12)	76% (16/21)
MuscleGainWorkout	100% (1/1)	92% (13/14)	76% (19/25)
WorkoutManager	100% (1/1)	42% (3/7)	55% (10/18)
user	100% (3/3)	57% (8/14)	73% (25/34)
Adult	100% (1/1)	100% (2/2)	75% (3/4)
Child	100% (1/1)	100% (2/2)	75% (3/4)
User	100% (1/1)	40% (4/10)	73% (19/26)

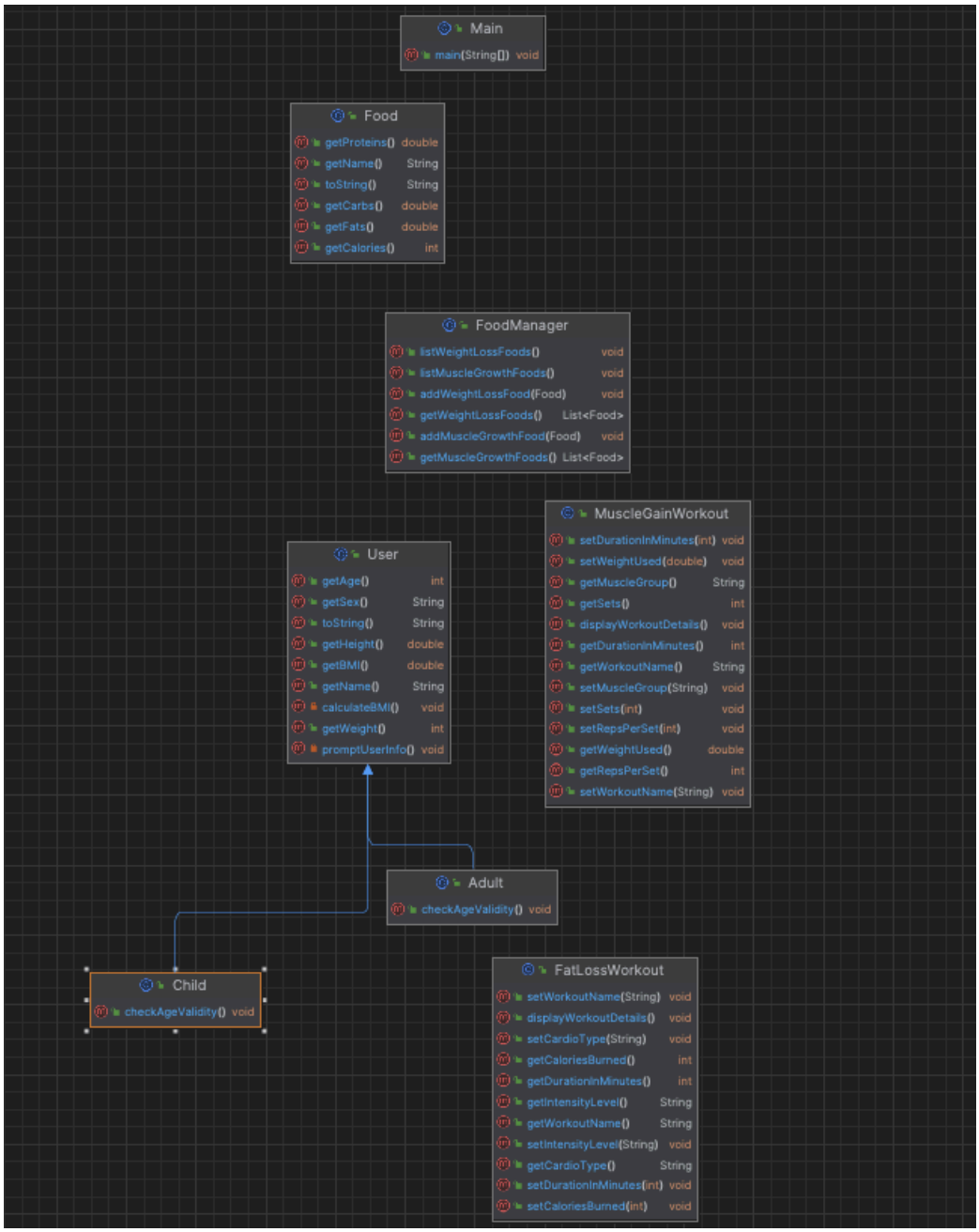
BDD Scenarios

For the nutrition section one use of it could be if someone is trying to go on a diet to lose weight they could ask Flex Flit what are some good foods to eat when trying to lose weight. This will then lead to flex flit outputting a list of foods that we chose are good choices when trying to lose weight.

For the training section if a user of Flex Fit is at the gym and is on a weight loss journey he could then ask Flex Flit what is a good workout to do to lose weight. Flex Fit would then give the user a suggested workout he could complete and would tell him how many calories this would burn.

The user section, which is crucial for the registration into Flex Fit, takes in the user's sex, height, weight, etc. to calibrate how we can best assist them on their workout journey. It will also automatically calculate the user's BMI when given their height and weight.

UML Class Diagram



Plan For Next Iteration/Completion

For the next iteration/completion of our FlexFit project, we aim to improve functionality, streamline user interactions, and increase test coverage. We'll expand integration tests to ensure seamless cooperation between user, nutrition, and training modules, and introduce more BDD scenarios. We plan to enhance user customization by allowing specific fitness goals and dietary preferences. We also intend to improve our nutritional recommendations with a wider range of food data and meal suggestions. Additionally, we will review and optimize our code

using design patterns to enhance scalability and maintainability, update our UML diagram, and document all new features. We will also do more performance tests to ensure our project runs smoothly as it grows. This plan is designed to enhance user experience and application quality, making FlexFit more effective for users.