



Norah Almubarak

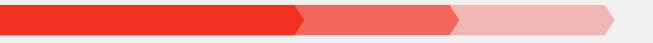
Aljawharah Alotaibi

Lujain Albattah

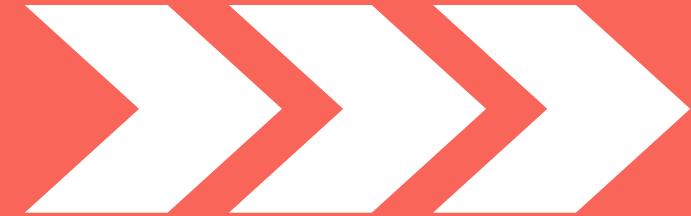
Raghad Alotibi

Yara Aljasir

Sarah Alshali



Outlines :

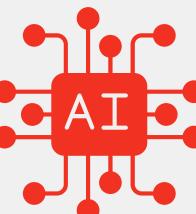
- Welcome to Uqla
 - Why Uqla
 - Mthodology
 - Uqla's Interfaces
 - Use case Diagram
 - Class Diagram
 - Component Diagram
 - Architecture Diagram
 - Sequence Diagrams
 - Deployement Diagram
 - Future Plans
- 

Welcome To Uqla

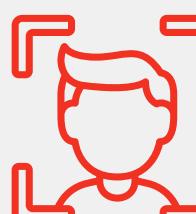
Nowadays, with health and fitness taking center stage in our lives, the demand for personalized and innovative solutions is more pronounced than ever.



Connects trainees with certified coaches



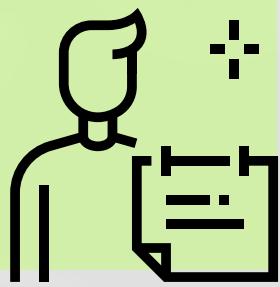
Utilizing AI analysis for detailed health insights



Integration of video recognition technology

Why Uqla?

Personalization of
Training Plans



Injury Prevention



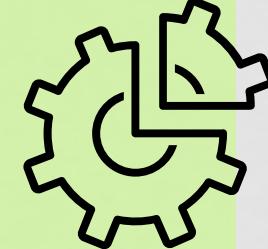
Data-Driven Coaching



Communication
Challenges



Integration and
Coordination



Feedback and
Interaction I





Methodology

Uqla adopts Agile Methodology.



Feedback after each sprint



continuous adaptation to market trends



continuous adaptation to user preferences.

Iteration number	Module/Subsystem	duration
1	Sign-Up, Log-In, Log-Out	18 days
2	Form recognition, External systems Integration	27 days
3	Finding a coach, Plans, Live sessions, Payment	49 days
4	Beta Testing	16 days

Vay API



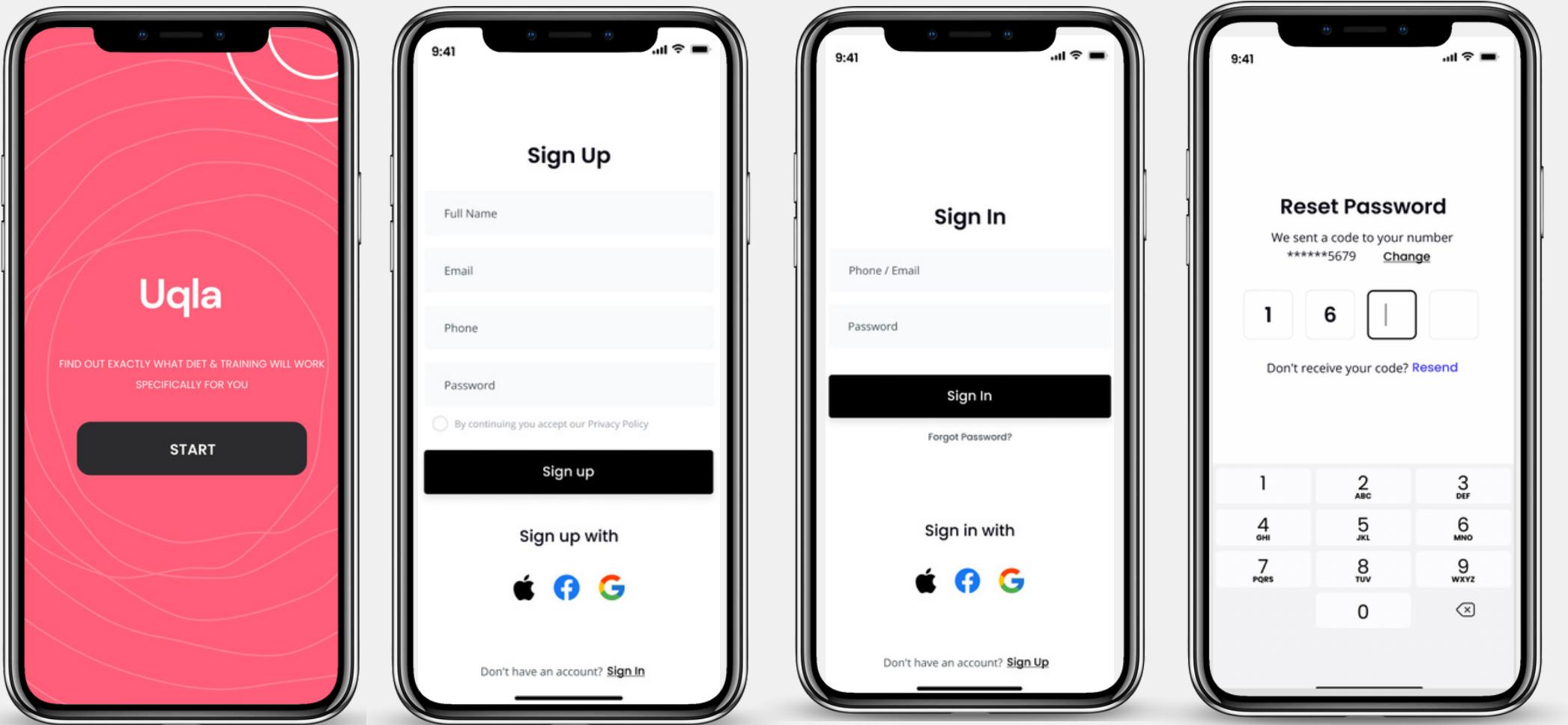
OpenAI API



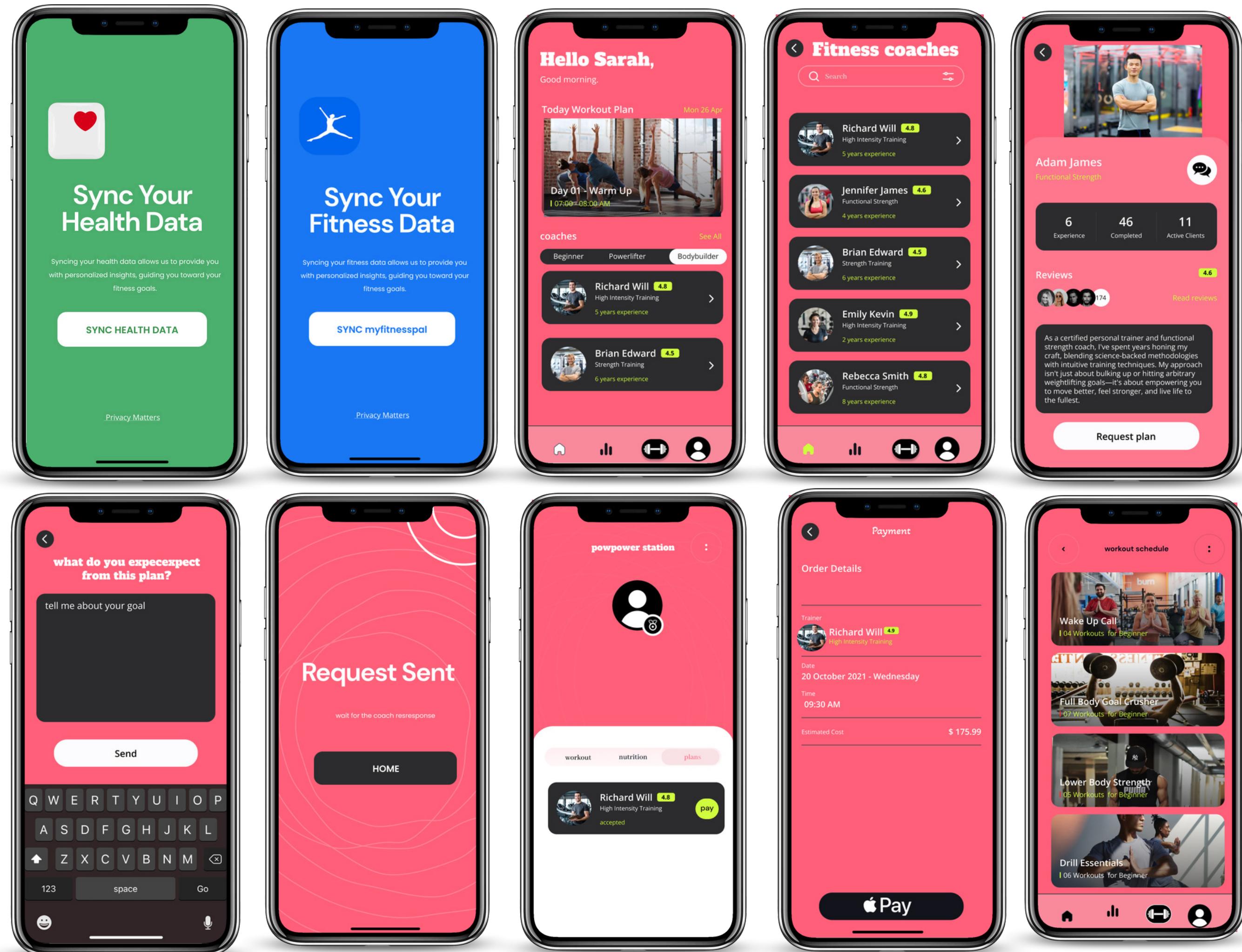
Idy
API

User Interfaces

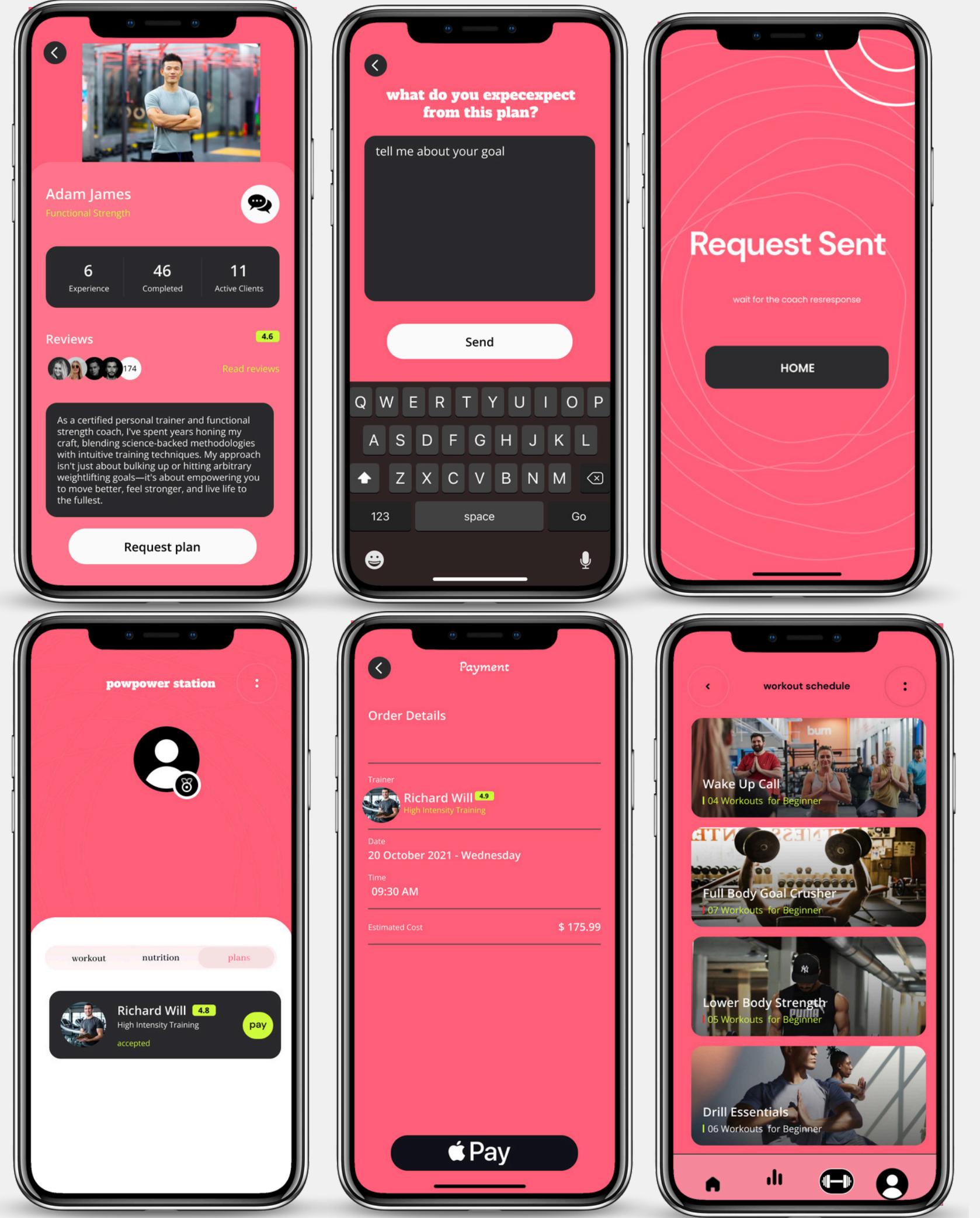
Our Solution



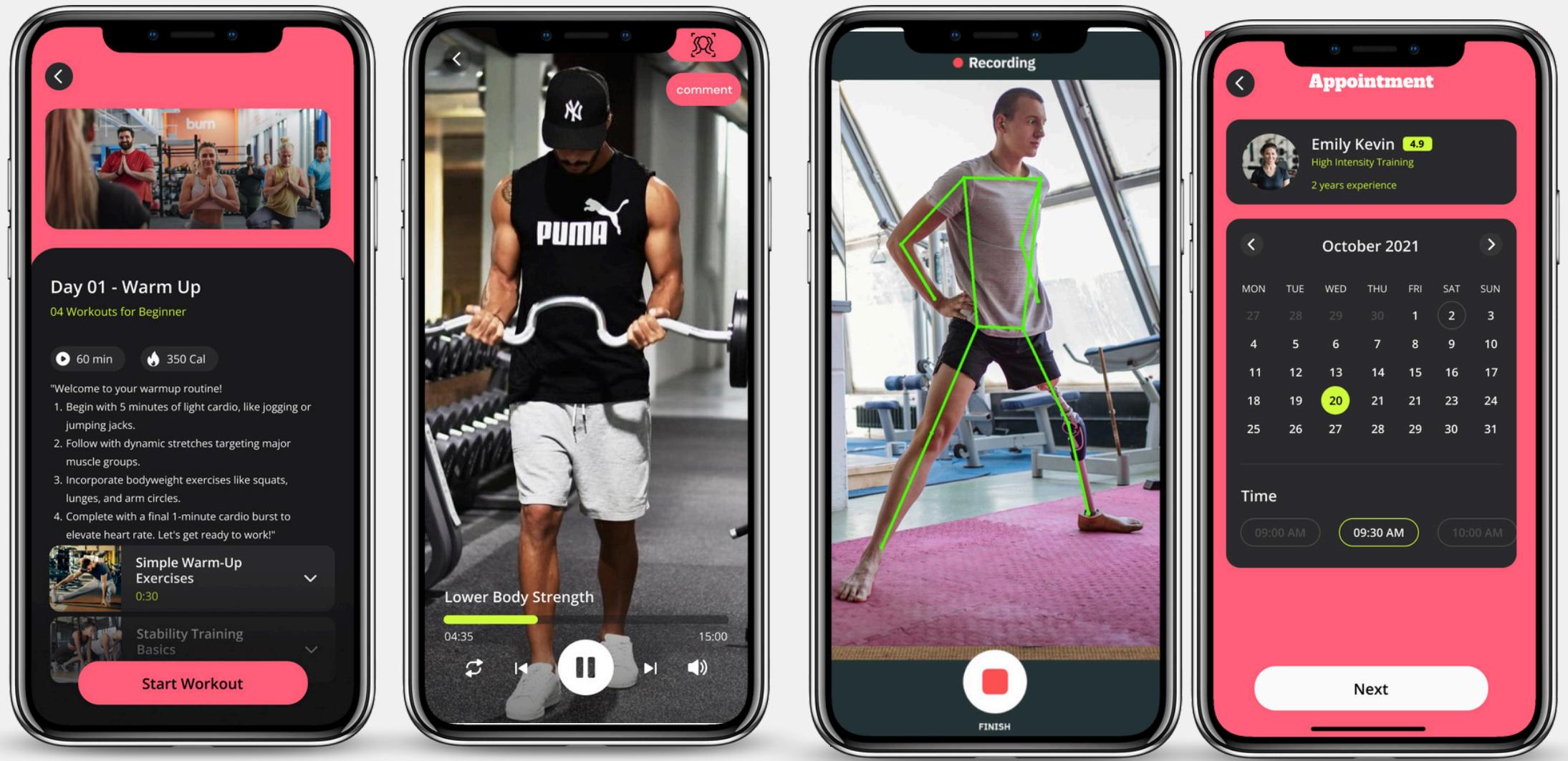
Trainee Interfaces



Trainee Interfaces



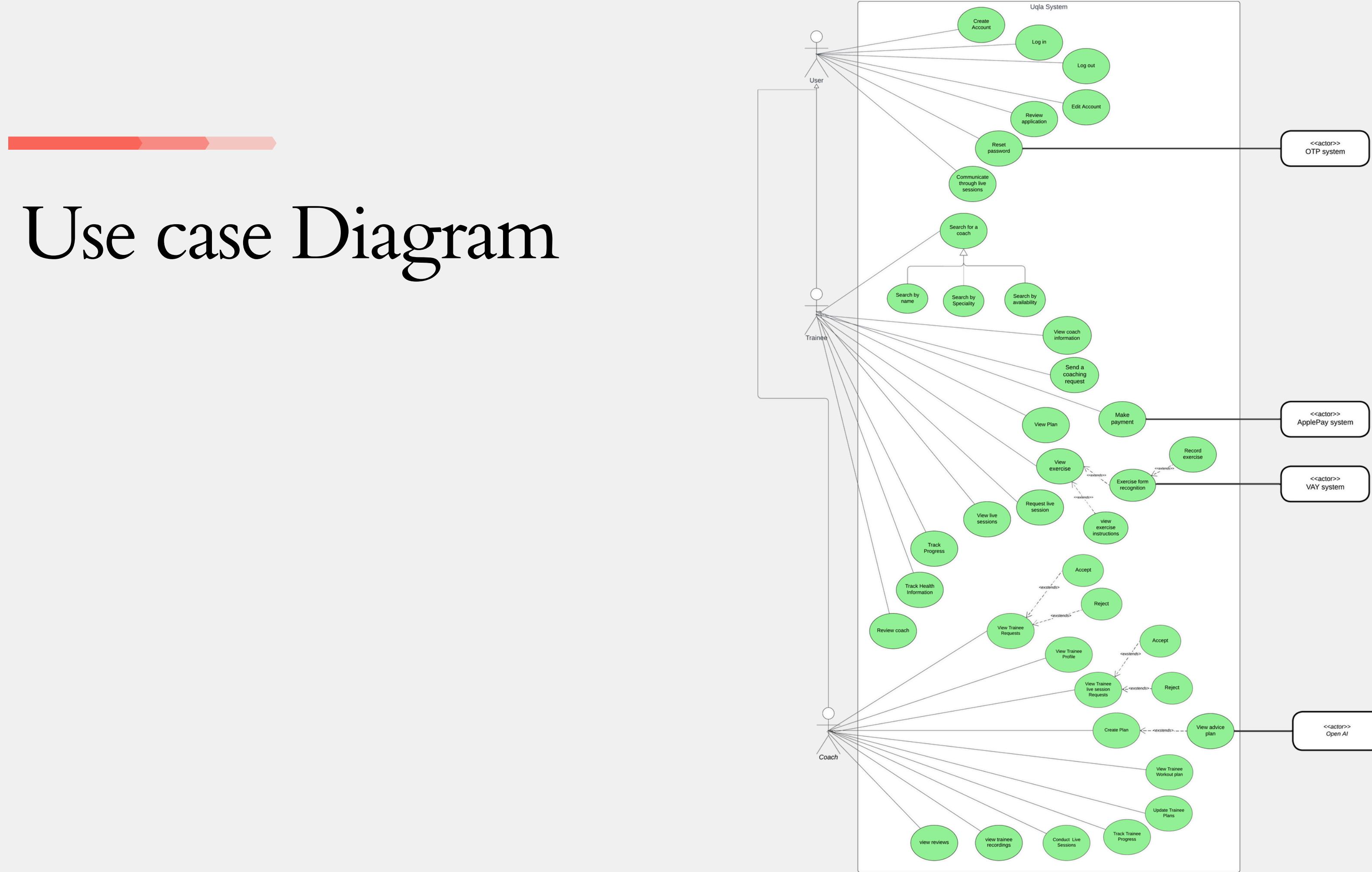
Trainee Interfaces



Coach Interfaces

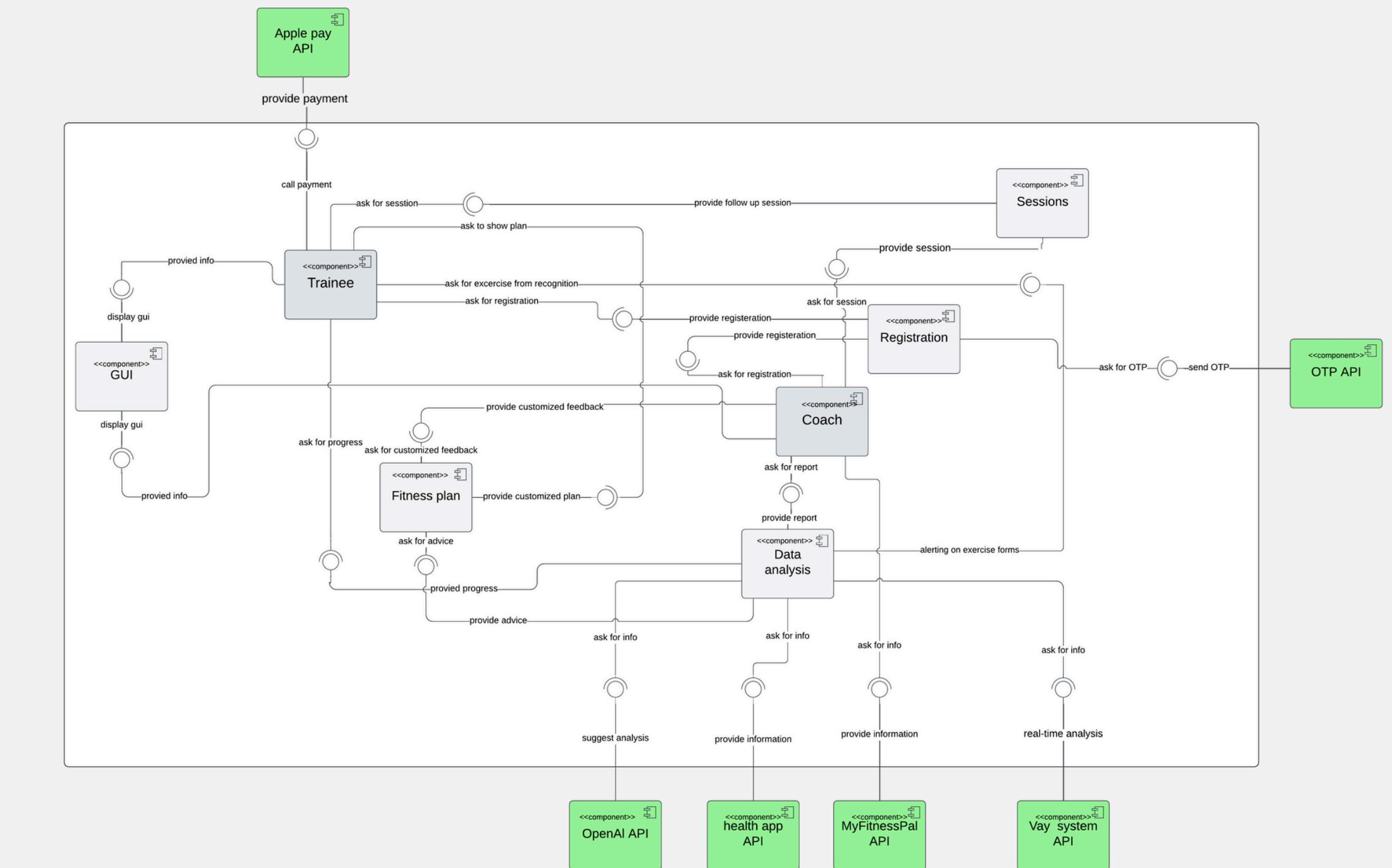


Use case Diagram





Component Diagram



Architecture style

we've chosen to adopt the Interaction Oriented Model-View-Controller (MVC) architectural style, which is an established design pattern that delineates an application into three principal logical components: Model, View, and Controller. This pattern offers several advantages that align well with the goals and requirements of Uqla.

Model:

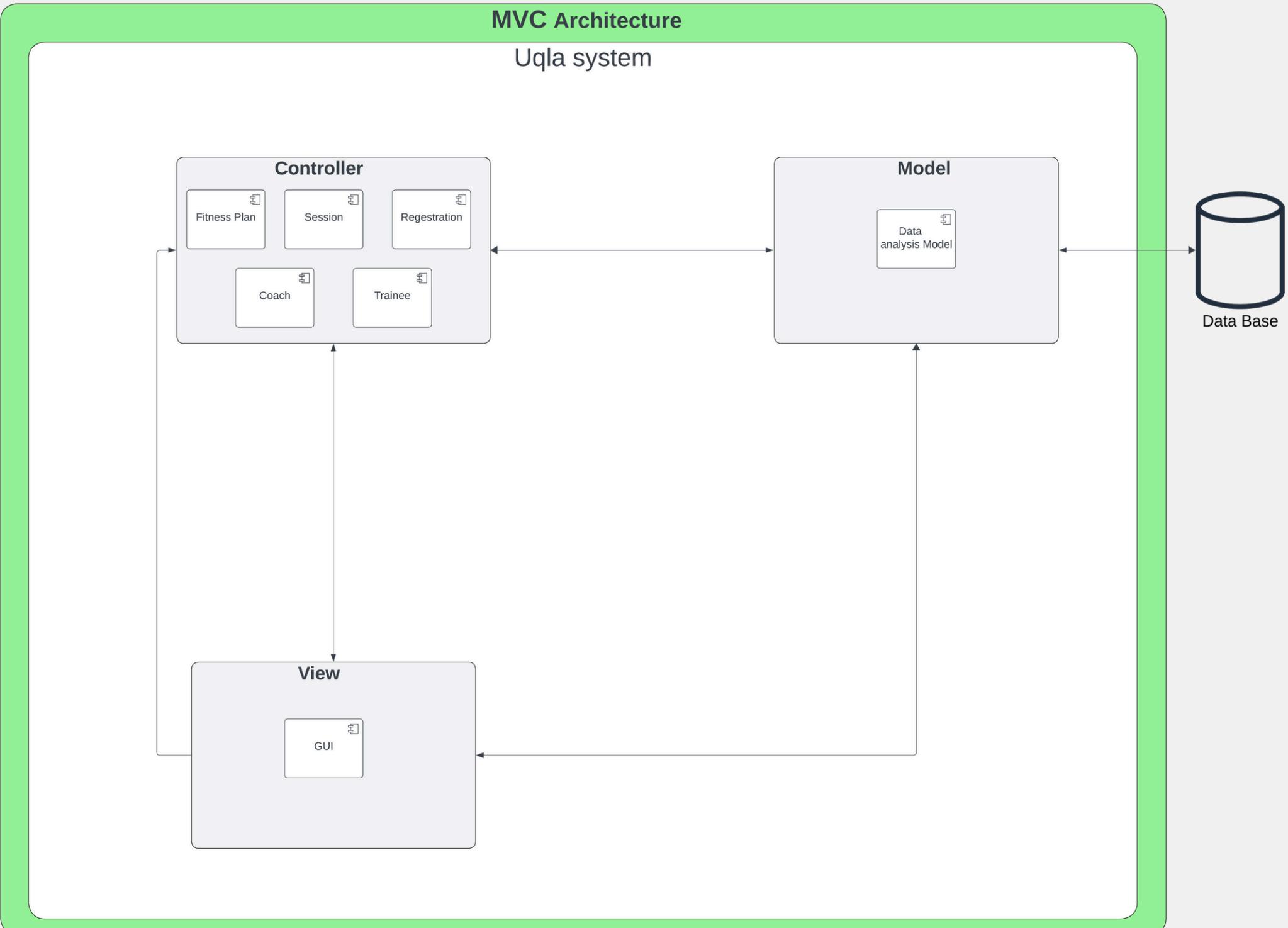
encapsulate the data and the business rules pertinent to the system. It will process data logic, and interact with the database for data storage and retrieval. This will form the backbone of Uqla's operations, handling the core functionalities and data manipulations.

View:

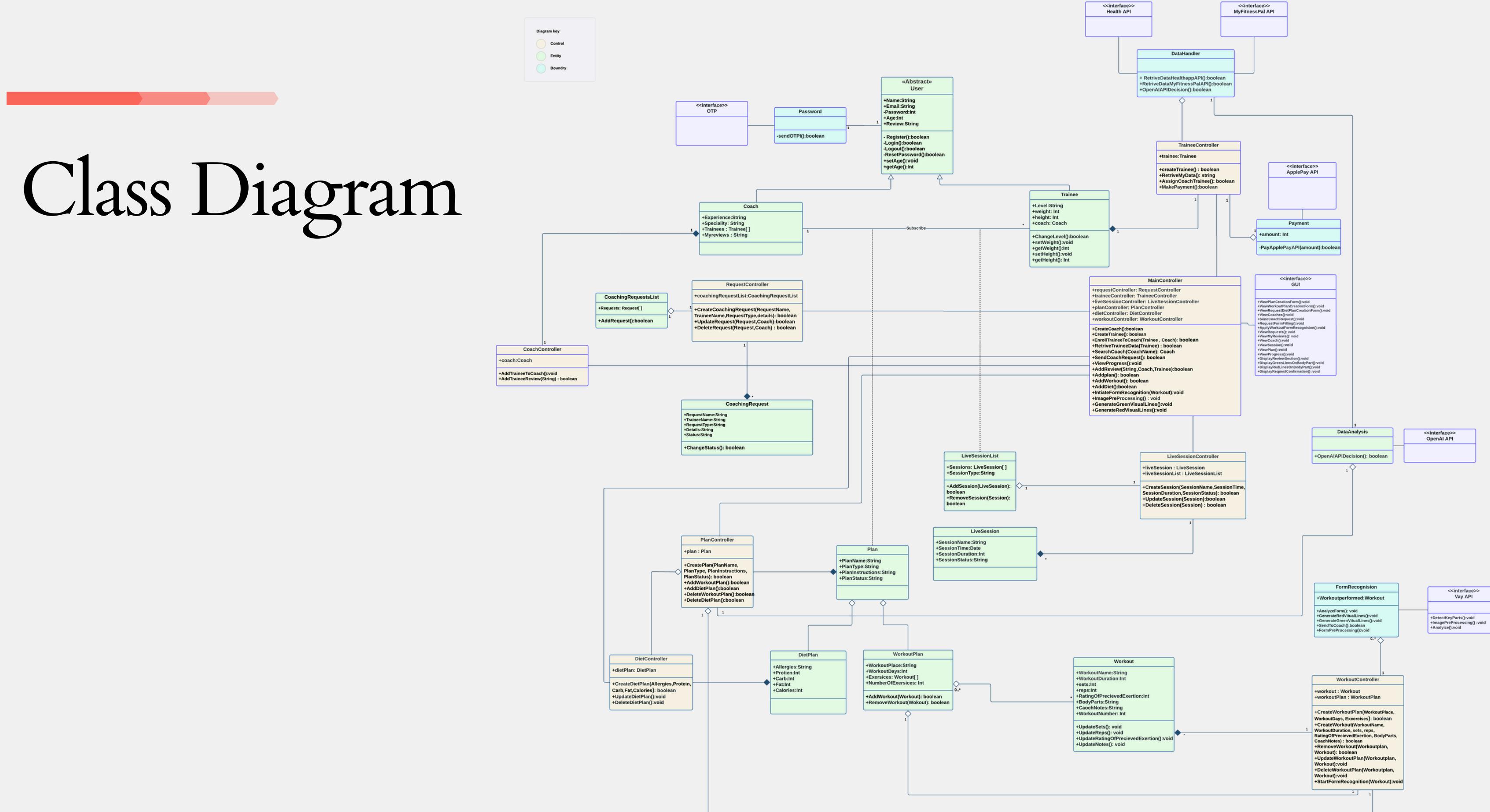
responsible for presenting data to the user in a structured and coherent manner. It will be the interface through which users interact with Uqla, ensuring that information is displayed effectively, and user input is captured accurately.

Controller:

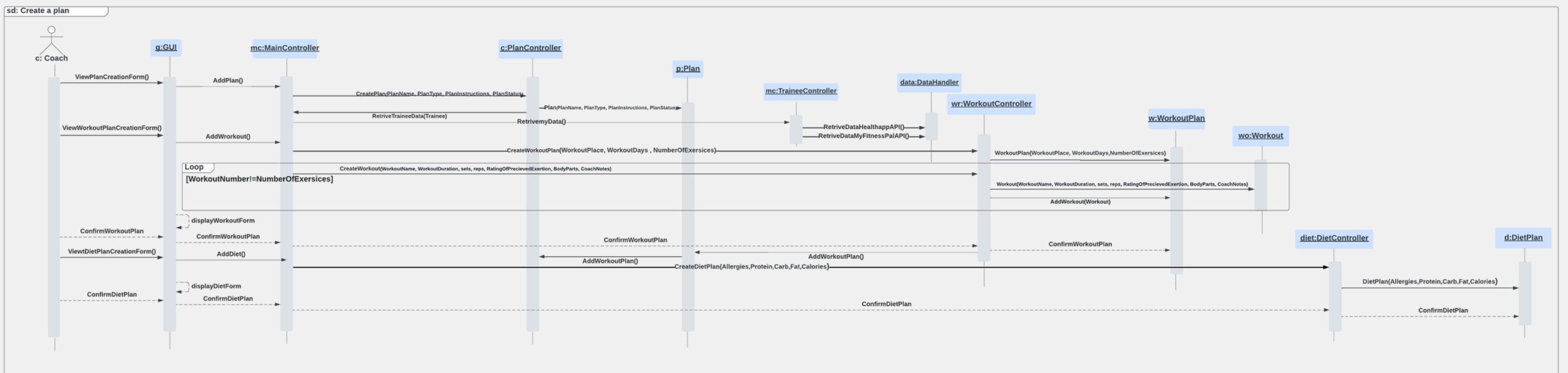
serve as the intermediary that connects the view and the model, facilitating communication between the user interface and data model. It will interpret user inputs and translate them into actions to be performed by the model.



Class Diagram

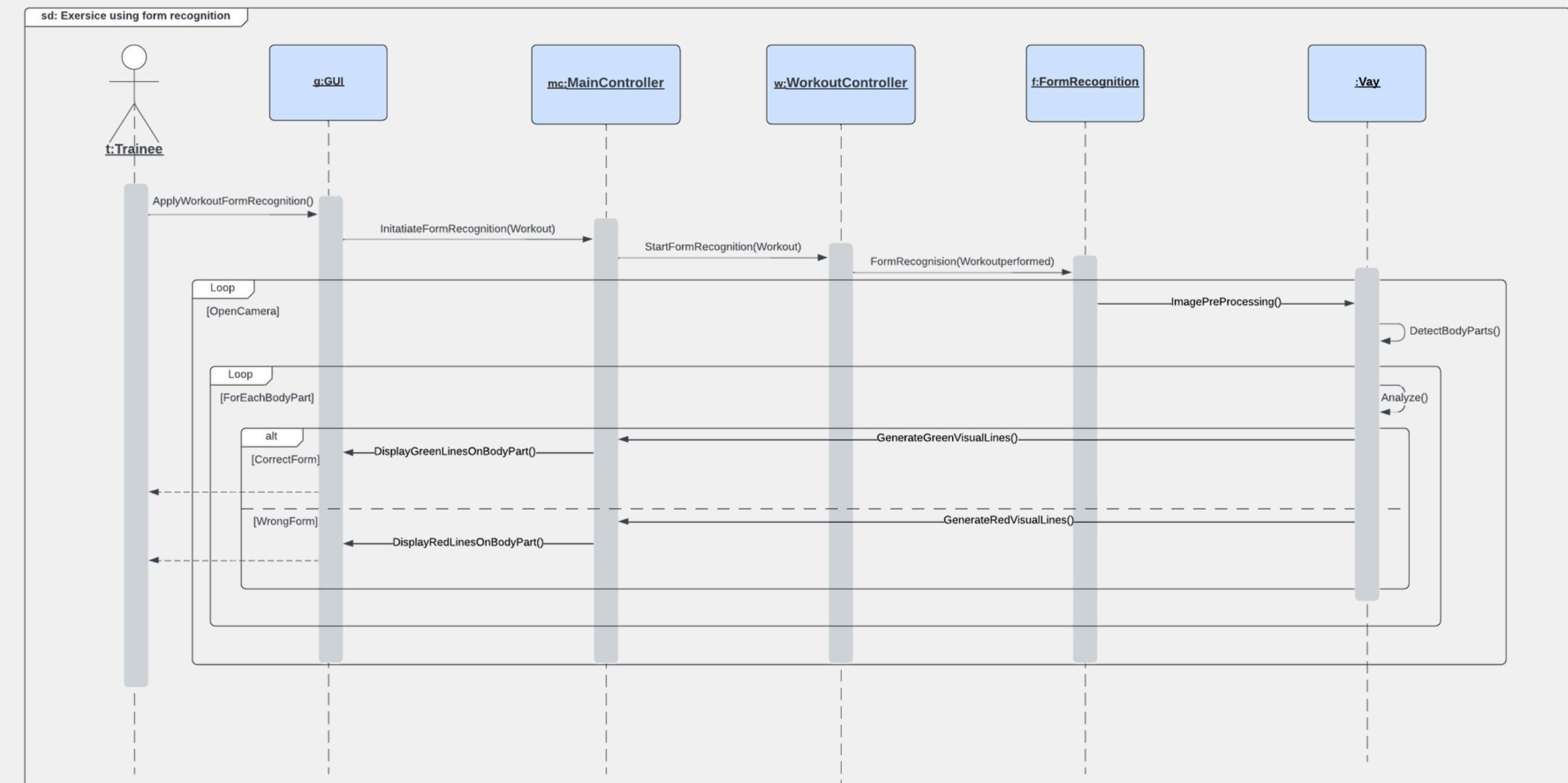


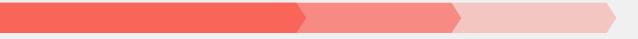
Sequence Diagrams



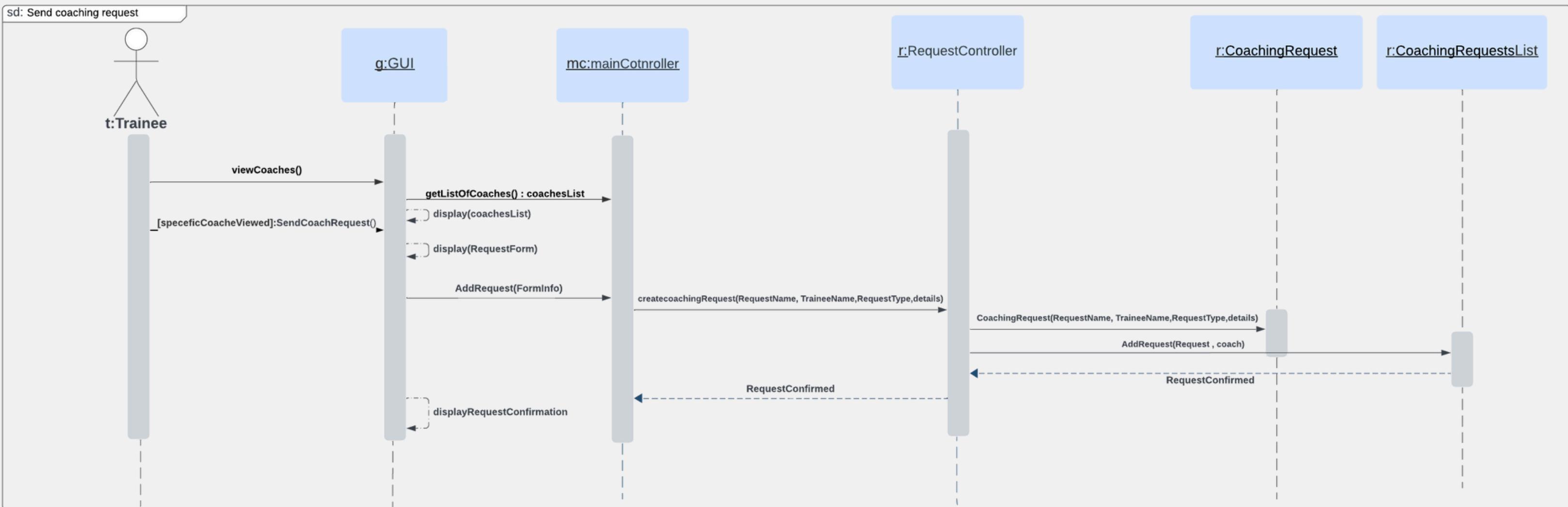


Sequence Diagrams



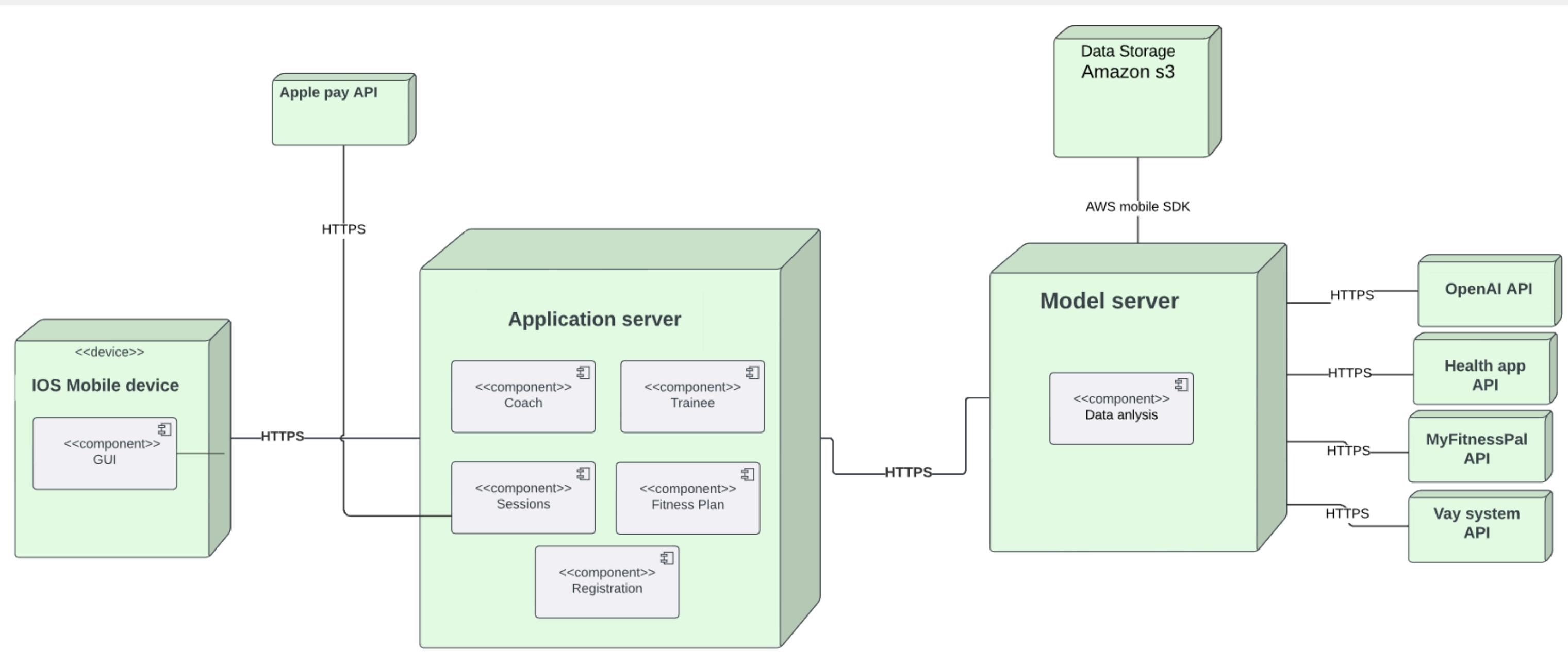


Sequence Diagrams





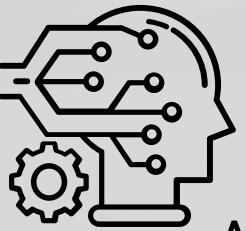
Deployment diagram



Future Work



External systems and
cloud computing



Adaptability in
Machine Learning



Proof of Concept (POC)
strategy to validate the
feasibility of integrating
external systems



Thank you for listening!