Project Overview

The AI Workout Companion is an application to help users break into fitness and bridge the gap between setting goals and executing them effectively and safely. The platform uses AI to either create a fitness plan or convert a previously prescribed plan into a structured workout schedule that users can easily follow and track through the app.

By providing form correction based on Computer Vision the system provides feedback on exercise form and monitors progress dynamically throughout each workout session. This enables users to maintain proper form, reduce the risk of injury, and build consistency in their routines. Overall, the project's objective is to create an accessible, data-driven fitness companion that empowers new gym-goers to achieve their goals safely and confidently.

Scope

<u>In Scope</u>

Our project will focus on creating an AI-driven fitness coaching system that leverages LLM API Interaction and CV-based form feedback to provide contextualized and generative insights during workouts. Users will be able to create accounts, track workout data, and receive semi-real-time, AI feedback based on their performance and form. A standard exercise form database will be integrated to enable dynamic comparison between the user's uploaded movements and ideal form models. The LLM will use both the user's past performance data and the form evaluation results to provide personalized feedback, encouragement and progress insights.

Core features include:

- Secure user account management and data persistence (individualized workout and form data)
- Contextualized feedback through generative AI, adapting tone and content based on user history
- Computer vision integration for post-upload analysis of movement form
- Comparison against a standard form database for accuracy scoring and targeted improvement suggestions
- Saved contextualization so the system learns and tailors feedback over time for each user

These components together will demonstrate how generative AI and CV can collaboratively enhance user fitness experiences through adaptive, data-informed coaching, even when form analysis occurs after video upload rather than in fully live conditions. Saved contextualization will be a combination of previous user inputs, user health metric data, and prior chats with their LLM "Fitness Coach." Some levels of this contextualization (especially previous LLM chats) will use context engineering methods and a RAG-based context retrieval.

<u>Under Evaluation</u>

If time and resources permit, we plan to evaluate integrating wearable health metric data (e.g., heart rate, step count, calories burned, etc.) from Apple Watch to enrich the AI's contextual understanding of user performance. This evaluation will be made via an evaluation of the ease of integration for the necessary software packages for Apple device accessibility and then also an evaluation of its "need" via user trial feedback. We are also considering a mobile iOS interface connected to the backend, allowing users to interact with the system directly through a clean and accessible app interface as opposed to a web browser. These capabilities will be implemented only if system integration and testing of the core features are completed early enough to allow expansion.

Out of Scope

Certain features have been intentionally excluded from this project to maintain focus on the technical integration of AI, computer vision, and user data systems:

- Nutrition tracking or coaching, including meal suggestions or calorie tracking
- Recipe generation or pantry inventory tracking
- Full integration with external fitness ecosystems (e.g., Apple HealthKit, Fitbit APIs) beyond experimental evaluation (experimental evaluation discussed in "Under Evaluation")
- Social media-type features such as leaderboards or friend activity feeds

These are beyond the intended scope of demonstrating semi-real-time AI and CV-driven feedback capabilities.

Assumptions

- Users are motivated to improve their fitness and are open to receiving feedback from an AI system
- Users will have access to a device with a camera capable of capturing sufficient-quality video form analysis
- Stable internet connectivity will be available to support communication between the app, the AI model, and backend servers
- The standard form database that the team integrates will be a reliable source for comparison purposes and a provider of healthy, standard repetitions of exercises.

Constraints

- The project must be completed within two academic semesters, limiting time for extended integration or large-scale testing
- Project funding is limited, so all components must be designed to operate with a minimal or uncertain budget, as provided by the course
- Computer vision model training and tuning will depend on available computational resources and data access
- The standard form database and LLM API usage may involve access or cost limitations

- Accessibility must be prioritized for users who are new to fitness and exercise tracking, ensuring that feedback is easy to understand, encouraging, and supportive regardless of technical or fitness experience
- Development and deployment are constrained by team expertise and available development tools

Requirements Elicitation & Prioritization

Through the MoSCoW method, the team learned how organized prioritization can make a complex project like our AI Workout Companion far more structured. To help map and prioritize our requirements into the MoSCoW method categories the team took on different roles (user and system) and walked through the expected flows relevant to each requirement. In doing so, the group was able to mass compile all the different potential defined requirements before differentiating and organizing them into the aforementioned MoSCoW categories. By differentiating *must-haves* such as fitness plan generation and AI-based contextualization from *should-haves* like computer vision form tracking and wearable data integration, we were able to focus on the core functionality that focuses on beginner users' success. The exercise also told us how helpful the process is for identifying alignment between technical and user-facing priorities—especially during early planning and scope definition. Although, we found that the *could-have* and *won't-have* sections were less actionable for a small team with a limited timeframe, since most features naturally fell into the must/should range. Overall, MoSCoW helped us define actionable goals, discuss trade-offs, and keep the project directed on its primary focus: assisting new gym-goers safely and regularly follow personalized fitness plans.

Requirements

Functional Requirements

1. **User Login Page**
    - **ID**: FR-1
    - **Statement**: The system shall allow a user login using a valid email address and password within 2 seconds and will present the opportunity to create a login if the user doesn't already have one.
    - **Rationale**: Secure authentication and timely access is critical for user access, especially for systems involving potential personal data.
    - **Test Method**: Perform 50 login attempts with valid credentials and measure the response time. 95% of attempts must succeed within 2 seconds. Also perform 50 attempts at loading the create a login page. All attempts should yield success.

- **Supporting Context**:Valid email address in this sense means common valid email addresses for any given application requiring an email for login (gmail, yahoo, .edu, etc.).
- **Trace**: Secure user account management and data persistence
- **Priority**: Must Have

2. **Contextualization Conversational Interface for Goals/Plan Setting**
    - **ID**: FR-2
    - **Statement**: The system shall provide an interface resembling a conversation with a chatbot/LLM upon first login. Through this interface the system will gain initial insights on the user to create an early fitness plan.
    - **Rationale**: To best model an optimal fitness plan the user must first give context to the LLM so that it can be contextualized to the user metrics, goals and more so that the resulting user-specific fitness plan used in the system is specialized per user.
    - **Test Method**: Upon creation of a new account, perform a high level text comparison between the summary of LLM responses and user input. The level of context overlap should be greater than 75% to indicate correctness of the “conversational” feel of the initial interface.
    - **Supporting Context**: Conversational interface means a back and forth between the LLM/AI and the user. There is an assumption that the user knows specific baseline metrics about themselves (height, age, bodyweight, etc.)
    - **Trace**: Contextualized feedback through generative AI, adapting tone and content based on user history
    - **Priority**: Must Have

3. **Fitness Dashboard**
    - **ID**: FR-3
    - **Statement**: The system shall provide a web page acting as a fitness dashboard upon user login that loads the progress visualizations, current plan, AI chatbot access, and workout snapshot within 2 seconds.
    - **Rationale**: Gathering and showing all information in one place is crucial to effective navigation and understanding of the system/app as well as best conveying progression to the user.
    - **Test Method**: Upon login for any existing user account, the first page directed to is the fitness dashboard page and that all necessary subcomponents show up within 2 seconds. Test for 10 different users with

a minimum 100% accuracy that the correct page is loaded and a minimum 90% accuracy all components load under 2 seconds.
- **Supporting Context**: User account is already existing. Workout snapshot refers to the current “next-up” workout in the user's plan.
- **Trace**:
    - Secure user account management and data persistence
    - Saved contextualization so the system learns and tailors feedback over time for each user
- **Priority**: Must Have

4. **AI Coach “Chatbot” Ability**
    - **ID**: FR-4
    - **Statement**: The system shall have a subsection on all pages beside login and conversational interface that provides user ability to interact with their contextualized coach and get prompt response within 5 seconds for prompts with maximum size 5k words. To best enable interface, before each LLM API call a summary of previous interactions will be provided as part of the prompt.
    - **Rationale**: To enable the “coach” feel of the AI/LLM, it must be accessible from any given place in the app and be adequately responsive to any user questions.
    - **Test Method**: For each page in the system, test chatbot interaction with prompts of size 500 words to 5k words in 500 word increments, response time should be under 5 seconds minimum 80% of the time.
    - **Supporting Context**: Subsection just refers to an element on page, in this case the element enabling prompting and prompt responses from an LLM that has been contextualized to the user goals.
    - **Trace**:
        - Contextualized feedback through generative AI, adapting tone and content based on user history
        - Saved contextualization so the system learns and tailors feedback over time for each user
    - **Priority**: Must Have

5. **Workout Input Ability**
    - **ID**: FR-5
    - **Statement**: The system shall provide the ability for the user to access the current workout via the workout snapshot in the fitness dashboard. The user will be provided the option to manually input data…if selected the

input ability will load instantaneously and any input data will be reflected in the snapshot after input.

- **Rationale**: The user should be given the opportunity to input data as the user may want to slightly change exercise, reps, weight for any given workout in their plan. This decision to change will likely come from a conversation with their AI coach and manual input will allow greater insight on user data.
- **Test Method**: With manual data input enabled, the user should be able to edit reps/weight of any exercise every time and after entry the updated value should be reflected in the workout snapshot. Test data entry for each exercise of a given workout snapshot for 5 different users, ensuring 100% data input ability and reflection.
- **Supporting Context**: The manually inputted data in this instance is just the reps/weight of any given exercise within the scope of that user's current workout snapshot.
- **Trace**:
    - Secure user account management and data persistence
    - Saved contextualization so the system learns and tailors feedback over time for each user
- **Priority**: Must Have

6. **Video File Upload for CV Feedback**
    - **ID**: FR-6
    - **Statement**: The system shall enable that, within the workout snapshot, the user will be able to select an exercise from the current workout and move into a form feedback page where a file of max size 100 MB can be uploaded in 5 seconds and feedback will be generated within 30 seconds as well as the user will be presented a link to the correct guided form and a list of common falls.
    - **Rationale**: The user should be able to get feedback on their form within a reasonable time with accuracy and tips to improve their form so that they can efficiently improve the safety and effectiveness of any given exercise in their workout.
    - **Test Method**: Test 25 video uploads of different size files. File upload time should be within 5 seconds 95% of the time and CV form feedback should output within 30 seconds 85% of the time with constant feedback to the user as upload progresses.
    - **Supporting Context**: Form feedback means a percentage comparison to a known "correct" form video as well as any detected common pitfalls of given exercise compiled into one "Form Score".

- **Trace**:
    - Computer vision integration for post-upload analysis of movement form
    - Comparison against a standard form database for accuracy scoring and targeted improvement suggestions
- **Priority**: Should Have

7. **Progress Visualization**
    - **ID**: FR-7
    - **Statement**: The system shall provide updated visuals on the fitness dashboard consisting of relevant and up to date data. The visuals will constantly be up to date with all data available upon any refresh after any data update and will update visualization within 3 seconds.
    - **Rationale**: Progress can be difficult to measure just from looking at plain data, by conveying to the users relationships between their data points motivation can increase and plan progression can be better seen.
    - **Test Method**: For each given visualized graph that contains data that can be changed, in either the user profile or manual data input via the workout snapshot, test 10 times for each graph that a change in the input data results in a visual change on the graph in 3 or less seconds with 90% accuracy. Accuracy will be measured via a comparison against the current stored data and the data being shown on said graph. These graphs will vary in type based on user selection.
    - **Supporting Context**: “Relevant” data means the selected data from the “Up to Date” data that actually can provide visualiztion that shows progression. There will be some data that will not be relevant to showing the user their progress through visualization.
    - **Trace**:
        - Secure user account management and data persistence
        - Saved contextualization so the system learns and tailors feedback over time for each user
    - **Priority**: Must Have

8. **Apple Watch Metric Tracking/Integration**
    - **ID**: FR-8
    - **Statement**: The system shall provide the option for apple watch metric integration while the user is within the workout snapshot, if selected the system will provide real time updates for relevant metrics in the snapshot within x seconds and will alert any dangerous or unusual metric activity with immediacy.

- **Rationale**: Fitness wearable metric integration enables access to user biometrics that, when monitored, can enable the system to track the intensity of a workout and alert any danger.
- **Test Method**: Test that, when enabled, the relevant metrics from the apple watch update within 3 seconds in a period of 60 seconds with 90% accuracy.
- **Supporting Context**: Alert would me a pop-up that a metric has gone above a standard healthy level during a workout.
- **Trace**:
    - Integrating wearable health metric data (Apple Watch) to enrich AI's contextual understanding of performance
- **Priority**: Should Have

9. Data Storage and Backend

- **ID**: FR-9
- **Statement**: The system shall adequately store all user account data, workouts, metrics, and AI interactions that can be accessed via a backend within 2 seconds from the relational databases.
- **Rationale**: The system is built on interactions of user data and summing it up into insights and plan progression for the user. For this to work there must be proper organization and accessibility of said data.
- **Test Method**: Test with 50 retrieval and write each respectively to the backend looking for all different types of data in the relational databases. Ensure that 95% of the retrievals return within 2 seconds and every single write successfully is reflected in the database.
- **Supporting Context**:In this context, the backend refers to the relational database infrastructure containing the user specific metrics and enabling access to it all.
- **Trace**:
    - Secure user account management and data persistence
    - Saved contextualization so the system learns and tailors feedback over time for each user
- **Priority**: Must Have

10. User Profile Management Page

- **ID**: FR-10
- **Statement**:The system shall have a user profile page showcasing profile metrics of the user that the user can change/edit and the change/edit will be reflected in 3 seconds where the reflection time will scale with the effect of the change.

- **Rationale**: Because the system is personalized per user, the user may wish to update their goals or metrics so that their personal interface is more aligned with where they want to be.
- **Test Method**: Perform 10 edits of different user profiles changing different user specific information for each user profile tested. The accuracy of updates being reflected needs to be at least 90% of tests to be reflected within 3 seconds. Additionally validate new results stay on refresh.
- **Supporting Context**: User profile metrics include all the different attributes the user initially sets (height, weight, age, goals) as well as their current plan and AI context.
- **Trace**:
    - Secure user account management and data persistence
    - Saved contextualization so the system learns and tailors feedback over time for each user
- **Priority**: Must Have

11. Data Export Capability

- **ID**: FR-11
- **Statement**: The system shall provide the opportunity for a data export file in the form of a CSV and/or a readable PDF that will compile all relevant user data and progress tracking that the user may export after download within 5 seconds off the system.
- **Rationale**: In the case that the user is utilizing the system as a means to progress through a professionally made plan before an eventual check-in with a personal fitness trainer, it is necessary that the user has the ability to summarize their progressions and data through the time they have used the system.
- **Test Method**: Test by doing 10 different data export requests, each from a different user account with different amounts of data. Validate that 95% of the export requests download within 5 seconds.
- **Supporting Context**: Relevant data includes all data that is visualized to show progression as well as any trends/insights in data that is behind the scenes.
- **Trace**:
    - Secure user account management and data persistence
    - Mobile iOS interface for exporting user data
- **Priority**: Should Have

12. User Plan Update

- **ID:** FR-12

- **Statement:** The system shall automatically suggest an update to the user's fitness plan every seven days by performing an evaluation of current stored metrics and determining whether or not the user is over or under progressing.
- **Rationale:** The core mission of the system is to help users execute their goals effectively alongside LLM and CV assistance. The system must have some level of dynamic change as to keep the interest of the user and continue to bridge the gap between setting goals and executing them.
- **Test Method:** Create 10 "fake" test users with differing input histories with 7 days worth of system usage. Verify that 100% of the time the users receive an updated version of the plan and are given the option to update their current one.
- **Supporting Context:** Stored metrics includes user inputted workout data, CV form accuracy, and LLM conversational insights.
- **Trace:**
    - Contextualized feedback through generative AI
    - Saved contextualization so the system can iteratively improve
    - CV comparison against standard form database
- **Priority:** Must Have

Non-Functional Requirements

1. **Usability**
    - **ID**: NFR-1
    - **Statement**: System shall have an intuitive and consistent user interface that is easy to navigate for any new user to complete the actions necessary to maximize the system without external help.
    - **Rationale**: The easier to use the system the better the feel of the system is and thus the better the engagement and therefore overall number of users is.
    - **Test Method**: Find 5-10 people with no context on the system and have them each create an account, login, and complete 3 core features. 80% of participants should face 0 issues completing the core features. These issues will be measured against a set baseline of quantitative measurements indicating intuition (number of clicks, time to access a specific feature, etc.). There will be consistent idea for these measurements of what number is indicative of an intuitive and easy to use system.

- **Supporting Context**: Core features include login, conversational contextualization via AI/LLM, fitness dashboard interaction, activity with workout snapshot, inputting/reporting data, and file uploading for CV form feedback.
- **Trace**:
    - Secure user account management and data persistence
    - Saved contextualization so the system learns and tailors feedback over time for each user
- **Priority**: Must Have

2. **Accuracy**
    - **ID**: NFR-2
    - **Statement**: The system shall have a Computer Vision system that produces accurate outputs with a minimum 85% correctness compared to any data via the standard form database used
    - **Rationale**: Accuracy creates user trust which directly affects the usability of the app.
    - **Test Method**: Setup test with someone trained in form correction (either a CoRec personal trainer or upper level Kinesiology student) and compare CV based "Form Scores" against that of the trained person so that the results overlap 85% percent of the time for 20 set exercises.
    - **Supporting Context**: Standard data will be reference videos identified from professional weightlifting sites as good form reps for that given exercise.
    - **Trace**:
        - Computer vision integration for post-upload analysis of movement form
        - Comparison against a standard form database for accuracy scoring and targeted improvement suggestions
    - **Priority**: Must Have
3. **Security**
    - **ID**: NFR-3
    - **Statement**: The system shall protect and secure all user data, personal information, stored metrics, and any prompting/context history from AI/LLM interactions via encryption and secure database storage with user authentication.
    - **Rationale**: The system will include personal data. It is crucial this data is protected to ensure privacy and user trust.
    - **Test Method**:
      Verify through security testing and configuration review that all user data and AI interaction history are encrypted at rest and in data transaction, and

that access requires authenticated requests. Perform attempted unauthorized access via injection techniques and confirm data is secure.
- **Supporting Context**: Security across all backend communication whether that is between the front-end or AI interactions with the relational databases through the backend.
- **Trace**:
  - Secure user account management and data persistence
- **Priority**: Must Have

Deliverables

Software Deliverables

- AI Workout Companion Mobile Application
  - Fully functional v1.0 of our software application that allows user authentication, plan generation, and real-time interaction with an AI coach
  - Core features include:
    - Personalized fitness plan generation using LLM integration
    - Form feedback through computer vision
    - Apple Healthkit wearable data synchronization
    - Progress and metric tracking dashboard
- Relevant Requirements: FR-1 (User Login Page), FR-2 (Contextualization Conversational Interface for Goals/Plan Setting), FR-3 (Fitness Dashboard), FR-4 (AI Coach "Chatbot" Ability), FR-5 (Workout Input Ability), NFR-1 (Usability), NFR-3 (Accuracy)

Backend Services

  - API endpoints for user data, workout logging, and LLM communication
  - AWS RDS for persistent data storage and Amazon S3 being used for video uploads and storage
  - AWS Lambda for asynchronous processes such as form scoring
  - LLM API serving contextual AI chat interface that user can access
- Relevant Requirements: FR-6 (Video File Upload for CV Feedback), FR-9 (Data Storage and Backend), FR-4 (AI Coach "Chatbot" Ability), NFR-3 (Security)

Documentation Package

- A comprehensive set of documentation to support users and developers, will include:
  - End-user quick start guide explaining how to upload videos and view form scores
  - Developer documentation which will cover setup, API endpoints, and module structure

    - OpenAPI documentation for backend endpoints
- Relevant Requirements: NFR-1(Usability),

Development Methodology

Our team chose Scrum as our development methodology due to its iterative structure, adaptability to changing requirements, and emphasis on continuous feedback. Scrum allows us to break development into short, manageable sprints, enabling regular inspection of progress and rapid incorporation of stakeholder feedback. This approach is especially effective for projects with evolving requirements, as it promotes transparency, accountability, and consistent delivery of incremental value.

We use Jira to support our Scrum workflow by managing the product backlog, sprint planning, task assignment, and progress tracking. Jira provides clear visibility into sprint goals, story status, and team velocity, which helps ensure alignment across the team and facilitates effective sprint reviews and retrospectives. Together, Scrum and Jira enable structured collaboration, efficient prioritization, and steady progress toward project objectives.

Verification and Validation Plan

https://github.com/lkay21/AI-Gym-Feedback-Companion/blob/main/doc/Verification%20and%20Validation.pdf

Gantt Chart

https://docs.google.com/spreadsheets/d/1jqt0bqbJDMeF0s4qPc-QQ24W8nhqlXgvxYtmo_-P1sQ/edit?gid=1709744959#gid=1709744959