Progress Report

- Increment 2 -
Group #4

1) Team Members

- Matthew Cegala, MLC22R, @mattprog/@mattprogfsu
- Amanda Orama, ao22h, @amandaorama
- Nicholas Holguin, NCH22A, @Nicholas87100
- Ashton Singpradith, AS23DI, @AshSingg
- Matthew Hummel, MBH22, @mhummel04

2) Project Title and Description

Smart Gym Tracker

Smart Gym Tracker is a web-based fitness tracking application that lets users log workouts, record biometric data, track milestones, and view trends. Smart Gym Tracker utilizes a React + Tailwind CSS frontend, C# backend, and a MySQL database to create a simple but powerful gym tracker app that allows users to simply enter their workout data and see trends.

3) Accomplishments and overall project status during this increment

Overall: Continued updating documentation and structure about how the different components of the application will be set up and communicated. Currently on schedule with our proposed releases for the database and front-end setups. The backend has fallen behind the release schedule and calls that will send data to and from both the frontend and database have not been setup. There are no changes in our features and communication at this time.

Frontend: Implemented and tested full user flows with mock data, including registration, login, logout, and forgotten password. Added admin functionality for managing users, workouts, and exercises. Improved UX with Back buttons and error handling. All pages functioned correctly using mock data since backend integration was still in progress.

Backend:

Ashton Singpradith: Developed and tested CRUD endpoints for Biometrics/workoutBiometrics API. These endpoints were successfully integrated with the database. The backend was also able to receive biometric data from the front-end, which ensured proper communication and success of storing data in the database. As of right now, the backend is fully functional with the front-end and database regarding biometric and workout biometric data.

Nicholas Holguin: I developed and tested the CRUD endpoints for the User/Admin API. The front end was able to receive data from the backend and the databases, which ensures a user can

input or get their data from the database. The integration of these endpoints also allows an admin to add, delete, or update a user.

Database: Setup database controller that will allow for the libraries created to access their respective tables in the database. All create, read, update, and delete functionality for individual tables has been created. It was tested with dummy code, and all function calls were working. The database is on pace to keep up with its components for the software delivery deadlines outlined.

Source Control: Continued management on GitHub with master branch and Increment branches, with any added branches being merged with Increment 2 upon completion. The final end product of this submission will be on the master branch.

4) Challenges, changes in the plan and scope of the project and things that went wrong during this increment

Frontend: A main challenge was creating realistic frontend functionality using mock data while waiting for backend integration. Ensuring consistent layouts across admin and user pages also required adjustments to styling and organization.

Backend:

Ashton Singpradith: A challenge I encountered during this increment was receiving data from the front-end. After going through all my options, I first checked the front-end code to see if the issue was from my backend. I noticed that the front-end code never called "postBiometrics" after saving the data. After I fixed this issue, my backend was finally receiving data from the front-end.

Nicholas Holguin: A challenge I encountered during this increment was the compatibility of the .NET version that I was using for the API and the other libraries that I had to source data from. Another major challenge was trying to connect all the controllers to the front end due to naming conflicts and how the metadata was handled by the program.

Database: A challenge for this increment was making sure that all the data was being properly synched from the database despite the backend not being fully operational. There were test programs created to supplement and ensure that calls to each of the create, read, update, and delete functions created for the database worked as intended.

5) Team Member Contribution for this increment

Matthew Cegala:

- Progress report –I wrote main parts to sections 6, 7, 8. I was also involved in writing and reviewing all sections. I was involved in database updates for sections 3 and 4.
- RD document – I created the designs and write ups for sections 4 and 5 which were the Use Case and Class Diagrams. I added and refined sections 1, 2, 3, and 7. I contributed to all parts in either the form of writing small sections of it or reviewing it for accuracy.

- IT document - I contributed to parts 1-2 with all database requirements. I wrote the test procedures followed for the database parts of parts 3 and 4. I wrote the non-execution review procedures for part 5.
- Source code – Created the database class library section that will be utilized to send and receive create, read, update, and delete data between the database and backend. I also updated the database schema, DDL, script, and instructions that can be used to set up that on each machine. I also created the GitHub repository and have been maintaining the tickets, pull requests, and other administrative features to ensure proper documentation.
- Video/presentation – Presented the database portion of the demo and wrote/presented the presentation aspect of the video.

Amanda Orama:

- Progress Report: Contributed to all sections regarding the front-end. Involved in reviewing sections and refining them. Started sections to make it easier to complete for the rest of the team.
- RD Document: I wrote section 1, contributed to sections 1, 2, 3, 6, 7
- IT Document: Filled out. Sections 1 and 2, Filled out section 3 & 4 regarding frontend for increment 2
- Source Code: Implemented all new frontend pages for Increment 2, including login, registration, forgot password, and logout flows. Updated admin pages for user management and workout/exercise management. Integrated mock data for testing and demonstrations. Improved overall UI consistency and added UX enhancements such as validation of messages, navigation buttons, and accessibility indicators.
- Video: Presented the front-end portion of the demo, showcasing user and admin workflows using mock data.

Ashton Singpradith

- Progress Report: I contributed to sections 3 (Accomplishments) and 4 (Challenges) regarding the backend. I also reviewed and updated backend information/grammar mistakes that appeared.
- RD Document: I did not have any contributions but did review the document and approved everything written by my groupmates.
- IT Document: I contributed to sections 3 (Functional) and 4 (Non-Functional) regarding the backend. I also reviewed and corrected any mistakes throughout the document.
- Source Code: Improved/Updated Biometric and workoutBiometric controllers, models, services, and related components. Implemented functional CRUD endpoints that were integrated with the MYSQL database and library models provided by Matthew Cegala (database). The source code was tested and confirmed to communicate successfully with the front-end and database.

- Video: I also did not make any contributions to the video but reviewed it and was satisfied with the outcome.

Nicholas Holguin

- Progress Report: I contributed to sections 3 and 4 of the Progress Report. I provided the accomplishments and challenges that I had with the backend of the project.
- RD Document: I reviewed the document and made corrections.
- IT Document: I reviewed the document and made corrections. But the document was great overall.
- Source Code: I greatly improved the user controllers, models, services, and clients. I managed to connect the user API with other libraries and gain access to the user database through a library that other members of the team built. I provided CRUD endpoints and endpoints for login, registration, and password reset, which are all inside the authentication controller.
- Video: I didn't contribute to the video, but I reviewed it and was overall pleased with how it turned out.

Matthew Hummel

- Progress Report: Reviewed this document and grammar and info
- RD Document: I reviewed the document and made corrections.
- IT Document: I reviewed the document and made corrections.
- Source Code: I completed the API support to the library of workouts and muscle groups we used, using Swagger in order to test the endpoint results successfully in my branch.
- Video: I reviewed the video and thought it looked great

6) Plans for the next increment

For the increment, we are planning on providing the following:

- Completing backend functionality
- Smart Tips and Suggestions
- Trends of Data
- Notifications

- Milestone and Progress Tracking

7) Stakeholder Communication

Subject: Smart Gym Tracker Progress and Current Status

Dear Stakeholders,

This email is intended to provide a brief update on our Smart Gym Trackers' progress and status.

Our teams working on the front end and database components of the application have been diligently working and met the expected requirements for the current release. These two teams are on track to meet the expectations outlined with this release and future releases. The backend team, however, has experienced some unintended setbacks while getting their controllers operational. These setbacks have resulted in some of their requirements to not have been met for this increment, but the backend team is working hard to resolve their issues and keep back on pace to meet their deadlines for the future.

In this release, we have provided you with all of the necessary components to have an operational frontend and database system with the majority of the user-related components for the backend setup. This release also contains updated documentation to reflect the updates to all class libraries and feature requirements.

The front-end team has created pages that revolve around limiting what users can see and update based on their privileges. Regular users now have the ability to fully insert workout and biometric data, while only administrators have the ability to access pages responsible for maintaining shared workout options and user management screens.

These libraires and pages are intended to be expanded to provide you with more refined documentation on our release plans, including what features will be provided and when.

The backend team has created some setup for getting the data from the database to the frontend, and vice versa, however there is still a lack of functionality for full communication channels to be set up.

The database team created calls that give the ability for user workout data and biometrics to be saved for future access. The database is now operational to create, read, update, and delete data from all tables outlined via these libraries.

Overall, the front end and database are on schedule for release and have met all expectations.

We sincerely apologize for the delay in our fully operation backend release and we are working hard to resolve this before our next release, where we intend to still provide the full features outlined in our delivery goals.

We hope this release has provided you with better insight into the current state of Smart Gym Tracker. Please reach out to any questions you may have.

Best Regards,

Smart Gym Tracker Development Team

8) Link to video

https://youtu.be/VtCAvOazOjc