Progress Report

- Increment 3 -
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: Further refined documentation and structure about how the different components of the application will be set up and communicated. Get notifications and goal tracking features setup for frontend and database. 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 fully setup. Database and front-end created all features and functionality as intended.

- Note: Backend has completed support between data flow of database to frontend, see details in Backend

Frontend: Further refined documentation and structure for frontend components. Implemented new pages and features for Increment 3, including milestone tracking, notifications, and data trends. All frontend components were wired to use mock data to simulate interactions with the backend. UI and UX were improved with consistent styling, validation, and navigation enhancements.

Backend: Tested User and Authentication endpoints using Postman and HeidiSQL to make sure the endpoints could connect to the database and could GET, POST, DELETE, etc. Reworked POST and PUT endpoints for User and Authentication to accept proper JSON data in the body. Also reworked the login controller in order to call the right database method and get it to output data.

Improved and tested CRUD endpoints for Biometrics/workoutBiometrics API. These endpoints were successfully integrated with the database/front-end.

Database: Further refined database for previous table updates and created new tables that will allow for trends, goals, and messages to be saved for persistence. Database controller libraries were updated

to allow for refined searches based on foreign key and user id linkages to reduce number of rows grabbed per request. All create, read, update, and delete functionality for new tables was added as well. It was tested with dummy code, and all function calls were working. The database is fully operational and tested.

Source Control: Continued management on GitHub with master branch and Increment branches, with any added branches being merged with Increment 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: Main challenge was working around the backend being incomplete. This required extensive use of mock data for testing pages and ensuring frontend flows like milestones, notifications, and trends worked as expected.

Backend: The main challenge was working through serialization issues. Also reworked POST and PUT endpoints for User and Authentication to accept proper JSON data in the body. Another challenge was reworking the login controller in order to call the right database method and get it to output data. Also had trouble connecting biometrics backend to front-end, resolved by calling missing "postBiometrics" function in front-end for biometrics.

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. This required extra work to create an additional test script from scratch that uses the database controller for libraries. 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. A DML script was also created to provide multiple entries of dummy data for each database table.

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 scripts, DML scripts, and instructions that can be used to set up that on

each machine. I also maintained 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.
- RD Document: I wrote section 1, contributed to sections 1, 2, 3, 6, 7
- IT Document: Filled out. Sections 1 and 2, Filled out sections 3 & 4 regarding frontend.
- Source Code: Implemented all new frontend pages for Increment 3, including milestone tracking, notifications, and trends pages. Improved overall UI consistency, validation, and navigation. Wired components to use mock data to simulate backend responses.
- Video: Presented the front-end portion of the demo, showcasing user and admin workflows using mock data.

Ashton Singpradith

- Progress Report: Reviewed and approved section 3/4 backend done by my classmates but also added some more information about my backend specifically in sections 3 and 4.
- RD Document: Reviewed this document and approved all sections done by my group members.
- IT Document: Reviewed and approved classmates' responses.
- 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/database (SmartGymTracker.Metrics.API).
- Video: I reviewed the Increment 3 video done by my classmates but also included my own video showing my biometric backend (Inc3BiometricsBackend).

Nicholas Holguin

- Progress Report: I contributed to the backend accomplishments and challenges in Section 3 and 4.
- RD Document: I reviewed the document and made corrections.
- IT Document: Contributed to Section 2 of the document and reviewed the completed document.
- Source Code: I reworked POST and PUT endpoints for User and Authentication to accept proper JSON data in the body. I reworked the login controller in order to call the right database method

and get it to output data. I tested all the User and authentication endpoints with Postman to make sure all the endpoints were connected to the database and could execute.

- Video: I reviewed the final video.

Matthew Hummel

- Progress Report: Contributed to backend completion, and added a note in section 3.
- RD Document: Reviewed this document.
- IT Document: Reviewed the document,
- Source Code: Added functionality of login, signup, profile editing, biometrics, and workout connectivity between frontend and database.
- Video: Added video titled “Inc3BackendShowcase”

6) Plans for the next increment

- This is Increment 3 submission for final release, no further increments planned at this time.

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 all the expected requirements for each release. These two teams have met all features and expectations outlined in time for our final release. 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 final release.

In this release, we have provided you with all of the necessary components to have an operational frontend and database system for every feature outlined in our proposal. We also have provided you with the user-related components and some other controllers for the backend setup. This release also contains finalized documentation to reflect the updates to all class libraries and feature requirements. These libraires and pages are intended to be expanded to provide you with a better understanding and documentation on our releases, including what features have been provided.

The front-end team has focused on the milestone, notification, and progress trend pages that were schedules for this release. Users now can view notifications intended for them as well as see trends based on their previously entered data.

The backend team has created some controller 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 and schema updates to allow for persistent notification and goal data, specialized queries to get more data searches, and refined some of the calls. 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 met all expectations and requirements listed.

We sincerely apologize for the delay in our fully operational backend release, due to the limited time crunch and unforeseen setback and challenges in getting the backend produced, we have been unable to provide you with a fully working app. In order to get the app fully operational we would like to request a meeting be scheduled to discuss funds and plans going forward.

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

inc3demo