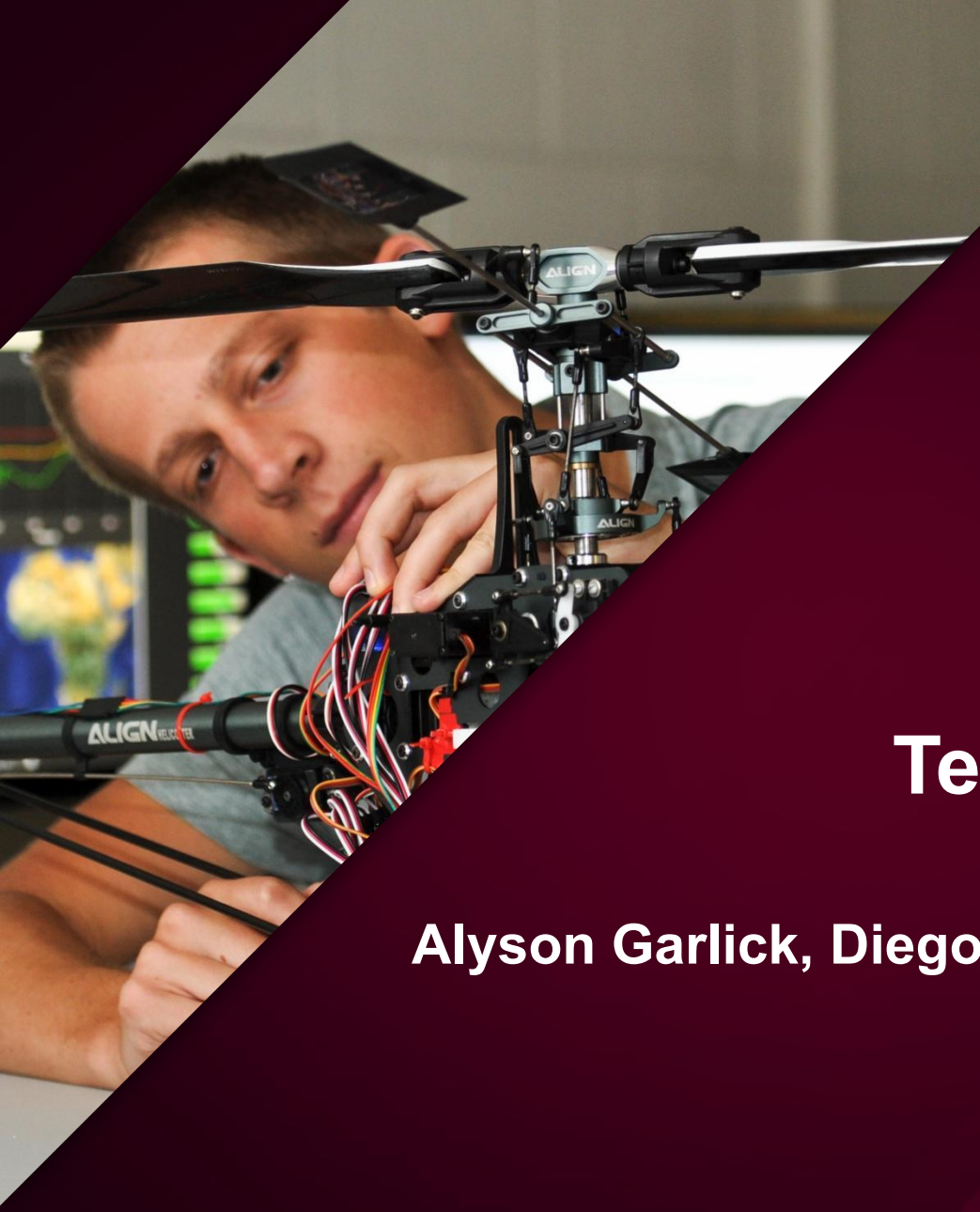


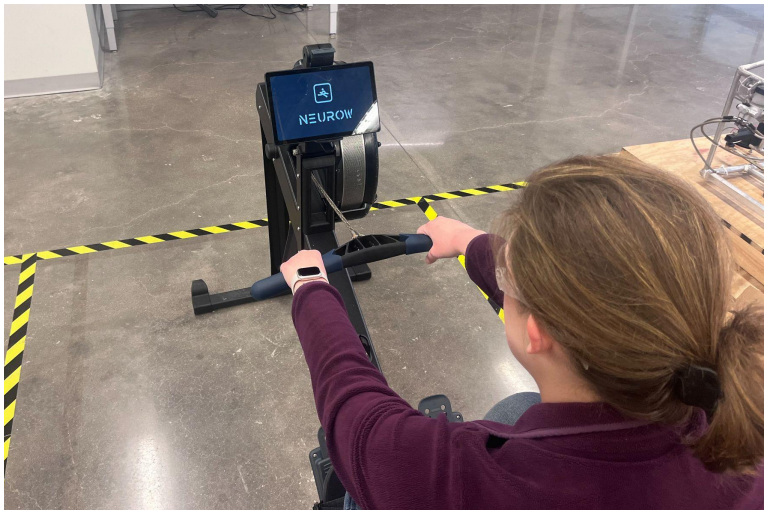
Dwight Look College of
ENGINEERING
TEXAS A&M UNIVERSITY

Team 45: TI Project #3
Bi-Weekly Update 4

Alyson Garlick, Diego Gumucio, Meredith McKean,
Nicholas McNamara
Sponsor: Matthew Krebs
TA: Pranav Dhulipala

Project Summary

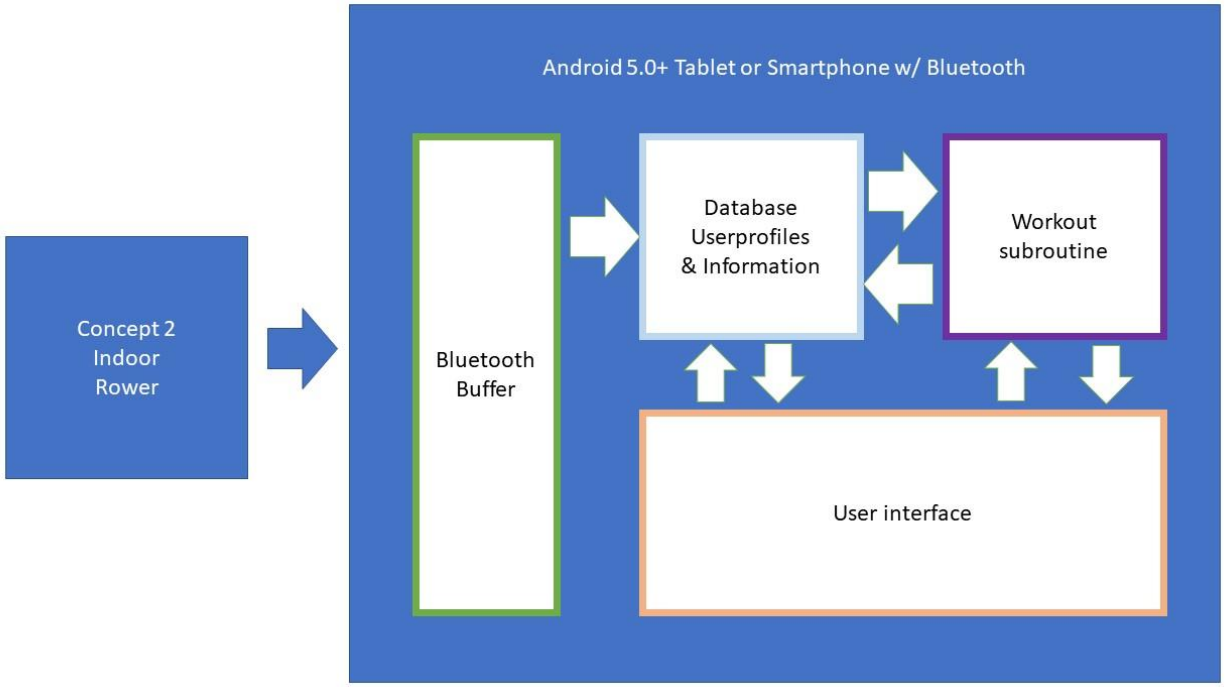
- A problem of older less expensive rower machines is that they lack features of higher end models and those from other types of equipment. These features keep the user engaged and focused on improving their overall fitness via Functional Threshold Power. FTP is a metric to maximize performance gains in the shortest span of time.



- To provide this experience, our subsystems will integrate on an Android application to connect to a Concept2 rower machine and provide:
  - A measurement of a user's FTP
  - Workouts consisting of seven different power zones
  - Recommendations based on their respective FTP
  - Storage for user's history local to the device.

Alyson Garlick
Diego Gumucio
Meredith McKean
Nicholas McNamara

System Overview



Data Communication Subsystem: Collects resistance, cadence, heart rate, and other data from Concept2 machine via Bluetooth

Database Subsystem: Manages and stores data for the Smart Rower, handles data transfer to subsystems.

Workout Subsystem: Responsible for creating the workout routines and calculating metrics to display through the user interface.

User Interface Subsystem: Facilitates user interaction and accessibility of data, ensuring an intuitive experience.

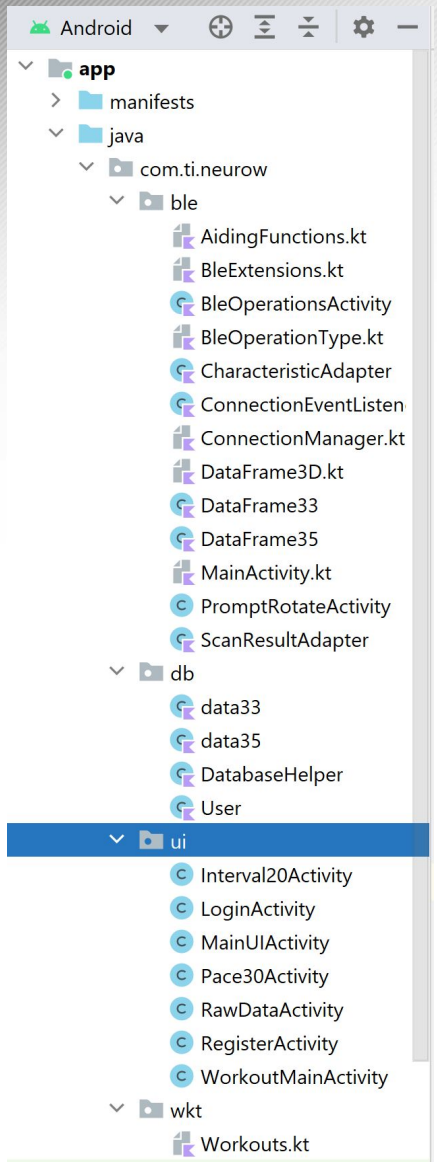
Project Timeline

Project Definition (completed 9/9)	Subsystem Designs and Testing (completed 12/2)	Integration of Database & Workouts and UI & Bluetooth (to complete by 3/1)	Integration (to complete by 3/5)	Front End Validation (to complete by 3/24)	Back End Validation (to complete by 4/12)	Demo and Report (to complete by 4/29)

Data Comm + UI

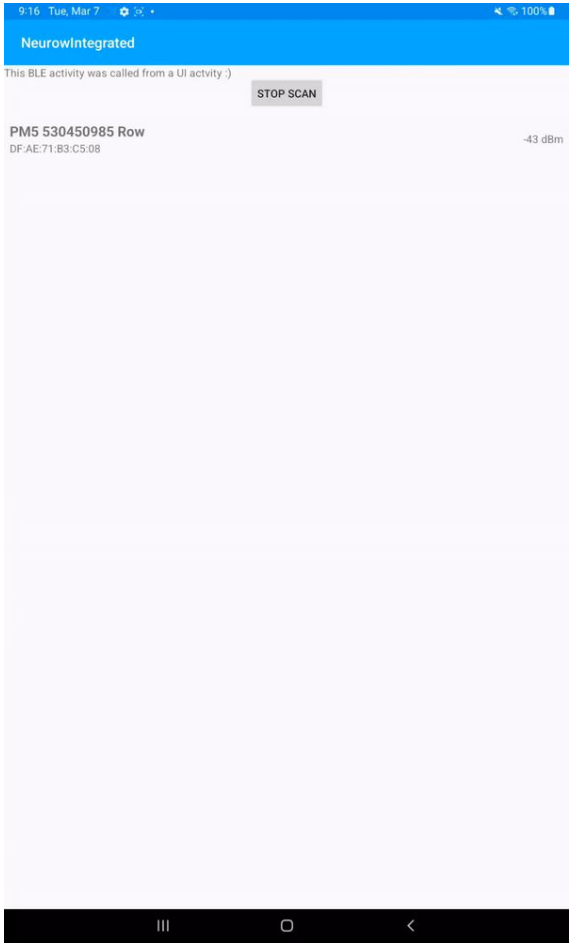
Diego Gumucio and Nick McNamara

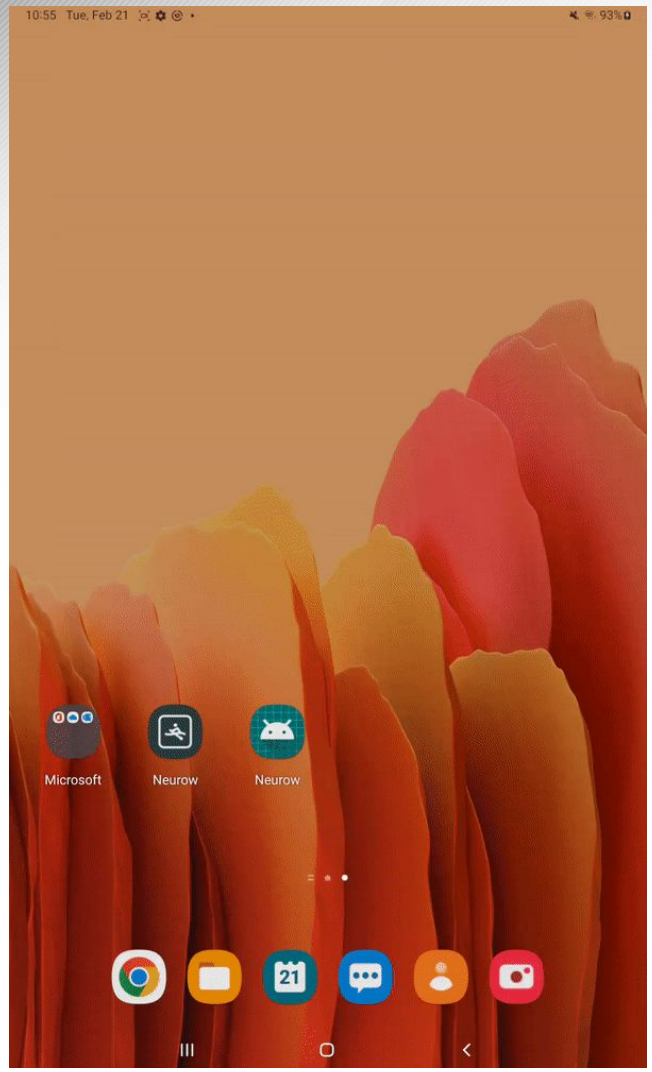
Accomplishments since last update 16 hrs of effort each	Ongoing progress/problems and plans until the next presentation
• BLE code compliant with new API level 33 • BLE + UI integration complete • Bug Fixes for UI elements in new API • All activities running correctly • Validation of BLE + UI through working BLE activity	• Ensure no locking occurs from BLE to DB queue • Ensuring concurrent processes in app • Begin integrating with database+workout engine



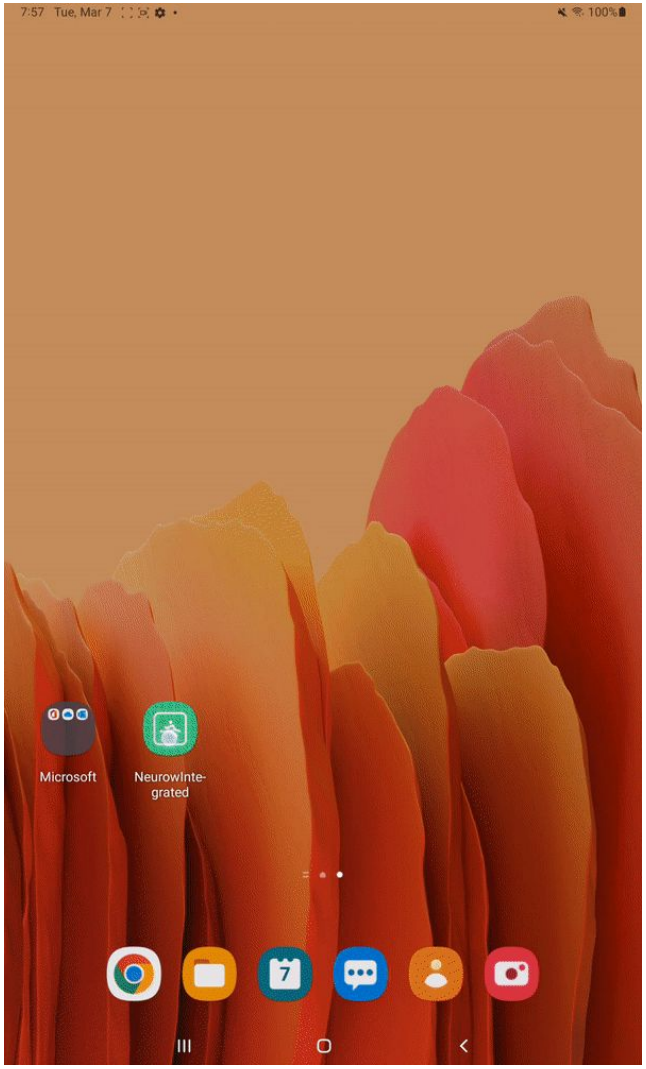
Data Comm + UI

- All subsystem source code in Packages
  - Only one set of app build files
- Fixed Additional BT permissions needed in API 33
  - Allowed for rest of BLE processes to run





Bluetooth Viewer
Crash Fixed!



Database + Workout Subsystem

Alyson Garlick and Meredith McKean

Accomplishments since last update 15 hrs of effort each	Ongoing progress/problems and plans until the next presentation
• Completed database method to return arraylist of power • Completed workout method for power predictions • Tested workout prediction method using database method output • Validated workout methods can write to database	• Integrate UI into workout methods • Exhaustive workout functionality validation will occur when integrated with BLE • Integrate database with BLE • Continue to modify/add database methods as integration continues • Start full system validation

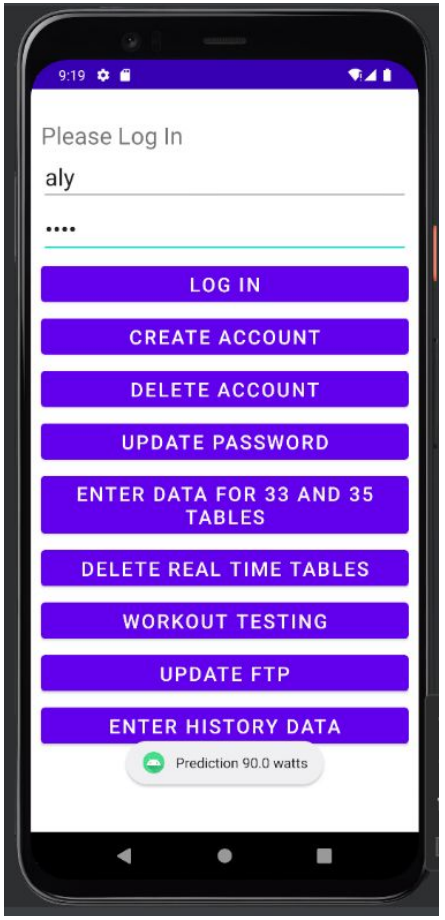
Database + Workout Subsystem

Alyson Garlick and Meredith McKean

9	aly	2023-03-0...	interval1	23	10.0
10	aly	2023-03-0...	interval1	23	20.0
11	aly	2023-03-0...	interval1	23	30.0
12	aly	2023-03-0...	interval1	23	40.0

```
val allpower = db.getAllPower( username: "aly",  workout_type: "interval1")
val predic = workouts.powerPredictor(allpower)
Toast.makeText( context: this@ActivityKotlin,  text: "Prediction $predic watts", Toast.LENGTH_SHORT).show()
```

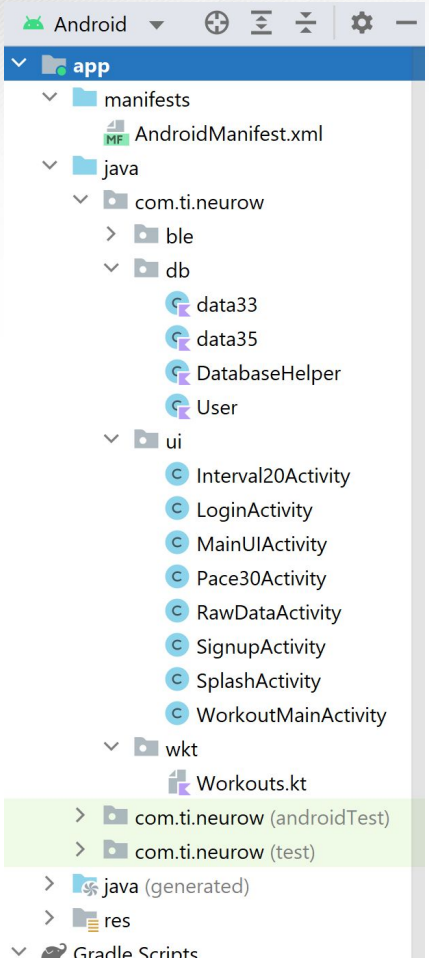
```
I/System.out: in predictor
I/System.out: power array [10.0, 20.0, 30.0, 40.0]
I/System.out: x sum 6
I/System.out: y sum 100.0
I/System.out: x mean 1.5
I/System.out: y mean 25.0
I/System.out: num sum 50.0
I/System.out: den sum 5.0
I/System.out: slope 10.0
I/System.out: y int 10.0
I/System.out: In 5 more workouts your power could be 90.0 watts
I/System.out: end predictor
```

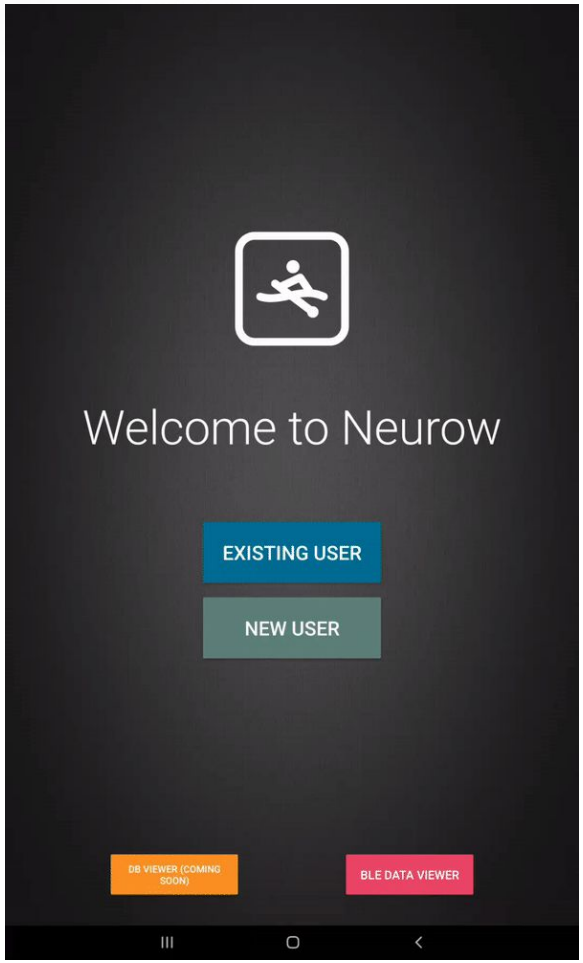


Integrated System

Accomplishments since last update 6 hrs of effort	Ongoing progress/problems and plans until the next presentation
• Integrated all subsystems into one application environment • Built and ran application successfully	• Validate user profile creation and login • Integrate UI into workout methods • Test exit of BLE driver application

Integrated System





- User login/creation authentication is working
- Feedback on username/password conventions
  - Have yet to add/call user to/from database
  - Temporary bypass login button for testing purposes

Execution Plan

	1/23/23	1/30/22	2/6/23	2/13/23	2/20/23	2/27/23	3/6/23	3/13/23	3/20/23	3/27/23	4/3/23	4/10/23	4/17/23	4/24/23	Date
Status Update 1															1/25/23
Github Reorganization															2/3/23
Bluetooth/User Interface Integration															2/10/23
Database/Workout Intergration															2/10/23
Status Update 2															2/8/23
Validation of Initial Integration															2/17/23
Dry Run Presentation with Sponsor															2/13/23
Create New Github of Kotlin App															2/19/23
Reacheive Functionality of Subsystems in Kotlin															2/22/23
Status Update 3															2/22/23
Workout and Database Integration															3/1/23
UI and BLE Integration															3/1/23
All Subsytems Integrate															3/5/23
Design Blitz Participation															3/5/23
Status Update 4															3/8/23
Integration Debug															3/24/23
Front End Validation															3/24/23
User Interface Polishing															3/31/23
Database Encryption															3/31/23
Status Update 5															3/29/23
Back End Validation															4/12/23
Corner Case Validation															4/12/23
Final Presentation															4/12/23
Demo and Poster Preperation															4/26/23
Final System Demo															4/26/23
Project Showcase															4/28/23
Final Report															4/29/23

Validation Plan

Paragraph #	Test Name	Success Criteria	Methodology	Status	Responsible Engineer(s)
N/A	BLE PM5 Connection	Device UUID on Application	Check BLE Callback object for successful connection	PASSED	Diego & Nick
N/A	Successful Characteristics subscription	Display Dataframe Units on Application	Call properties of Dataframe object for successful connection	PASSED	Diego & Nick
N/A	Time delay between HR data displayed (start to finish)	Heart Rate value displayed in UI falls within 1 second of value displayed on Bluetooth interface	Use a timer and visual judgement	UNTESTED	Diego & Nick
N/A	Database Storage	Storage of running bluetooth data in database	After connecting database to bluetooth, database will store all necessary data for calculation after a workout run	UNTESTED	Meredith & Nick
N/A	Calculations and Database Match	Workout subsystem runs with connected database subsystem	Connect the workouts with the database to test workouts function with gathering data from the database	PASSED	Alyson & Meredith
N/A	Workout Functionality with Bluetooth Data Stream	Workout functions and calculations are correct with real time data streamed into the database	Connect bluetooth to workouts and database and run workout methods to test functionality still performs	UNTESTED	Alyson & Nick
N/A	User Accounts connection with User Interface	Successfully able to create an account with app	Connect database subsystem with user interface subsystem	UNTESTED	Diego & Meredith
3.2.1.1	Display Five Workouts	All user interface elements are correctly displayed for each of the five workout options	All items are legible and selectable through the android debugging feature	UNTESTED	Alyson
3.2.1.2	Collect Data from Concept2	Data is collected from the Concept2 with every stroke pulled within 20 meters.	Connect the app to the Concept2 and begin data collection and steadily increase distance until 20m is met	PASSED	Nick
3.2.1.3	User Profile Storage	User profiles with corresponding workout history and FTP value shall be saved in the database	Create user, do FTP workout to get value, and complete two other workouts to see if the user, their FTP, and their history is saved correctly	UNTESTED	Meredith
3.2.2.3	Mounting	Smart Rower tablet is held up using the device holder.	Try and mount the tablet with the device holder and perform a workout to test the stability	UNTESTED	Nick
3.2.3.2.1	Data Output	Smart Rower displays four workout routines along with calculated FTP based power zones	Use the workouts on the app to test correct data display functionality and user experience	UNTESTED	Diego
3.2.4.1	Pressure (Altitude)	Smart Rower performs correctly in varying altitudes ranging from 0-12,000 ft above sea level	Use the app in different altitudes to verify correct functionality	UNTESTED	Full Team
3.2.4.2	Thermal	Smart Rower performs correctly in temperatures ranging from 0 to 35 degrees celsius	Setting ambient temperature to 0-32 degrees Celsius	UNTESTED	Full Team
3.2.4.3	Rain	Smart Rower runs while inside without wet conditions	Use the Smart Rower indoors in dry conditions	UNTESTED	Full Team
3.2.4.4	Humidity	Smart Rower runs while in humidity ranging from 30-50%	Use the Smart Rower indoors in humidity ranging from 30-50%	UNTESTED	Full Team
3.2.5.1	Failure Detection	Application displays generic failure flag	Introduction of a failure to connect flag from bluetooth	UNTESTED	Nick
3.2.5.2	Recovery	App displays error message and goes to home screen upon an incorrect user profile or invalid user input	Input an incorrect user profile and invalid user input to test whether app displays error message and goes to home screen	UNTESTED	Diego
N/A	Full System Demo	A user of the app is able to use all workouts and functionality of the app without issues or errors	A team member connects the tablet to the Concept2, creates a profile, and does all available workouts. They then connect/log back in to do more workouts and test user memory features.	UNTESTED	Full Team

Thank you for listening!