Sprint 2: February 16th, 2026 to March 1st, 2026

1. **System Overview**

   Mobile EEG-based self-experiment application that analyzes short-term focus

   stability across location, music, and time of day to identify optimal focusing

   conditions.

2. **Functional Requirements**

   User Story: As a user, I want the application to process and record my EEG data,

   convert it into a usable format, analyze its metrics, and display its findings on an

   Android app.

   2.13 - Create Algorithm to Map EEG Data to Focus Levels –

   - FR 2.13.1: Define the EEG input used by the algorithm (engagement proxy,

     SDK 'attention', etc.) based on the metric definitions.

   - FR 2.13.2: Implement a baseline "focus level" time-series computation with

     windowing & smoothing.

   - FR 2.13.3: Implement a session-level focus summary score derived from the

     time-series metric.

   - FR 2.13.4: Specify how poor signal-quality segments are handled (exclusion

     or down-weight).

   - FR 2.13.5: Implement handling of missing or invalid samples (skip

     windows).

   Progress Summary:

2.14 - Hard-code EEG data in Mobile Application to Test the Initial

Functionality –

- FR 2.14.1: Find / Create a demo dataset representing multiple sessions with timestamps and sample values.

- FR 2.14.2: Ensure the skeleton app can load the hard-coded dataset.

- FR 2.14.3: Verify the algorithm (Req 2.13) runs on the hard-coded dataset and produces expected outputs.

Progress Summary:

2.16 - Configure Environment: Create Backend Services for DB Services –

- FR 2.16.1: Select and initialize a backend stack and project structure.

- FR 2.16.2: Define a backend data model for session records.

- FR 2.16.3: Verify the backend runs.

- FR 2.16.4: Implement an endpoint to store session summaries or create a documentational equivalent.

- FR 2.16.5: Create and detail setup documentation so that anyone could read it and run the backend locally (readme basically).

Progress Summary:

2.21 - Set up EEG Interface (Middleman) –

- FR 2.21.1: Download and familiarize with EEG SDK.

- FR 2.21.2: Define a sample object with basic qualities (timestamp, values, signal quality, etc.).

- FR 2.21.3: Implement the output of a sample stream.

- FR 2.21.4: Implement connection state placeholders (disconnected, connecting, connected), even without EEG.

- FR 2.21.5: Document how device integration will work with the interface (SDK -> EEG Interface -> Session Manager).

Progress Summary:

2.25 - Create and Complete Sprint 2 Artifacts – Riley

- FR 2.25.1: Compile Requirements for Sprint 2 into formatted documentation.

- FR 2.25.2: Write functional requirements for each requirement, leaving space for progress summaries.

- FR 2.25.3: Verify artifact alignment with course rubric and project objectives.

- FR 1.20.4: Upload all Sprint 2 documents to Github.

Progress Summary: Compiled all Sprint 2 Requirements into the specified format of the Artifacts document. Broke each requirement down into its Functional Requirements, and gave a description for each. Uploaded to Github.

2.46 - Create Analysis of Perceived Focus vs. EEG Stability –

- FR 2.46.1: Implement a 'perceived focus' score in the app at the end of a session.

- FR 2.46.2: Ensure the algorithm produces an EEG stability score for the same session.

- FR 2.46.3: Implement a simple mismatch or comparison output (difference score).

Progress Summary:

2.52 - Implement Focus Algorithm Into App –

- FR 2.52.1: Implement algorithm (Req 2.13) into application.
- FR 2.52.2: Ensure the algorithm processes incoming (mock) data during the session.
- FR 2.52.3: Ensure outputs are passed into the UI layer.
- FR 2.52.4: Ensure no crashes during runtime integration.

Progress Summary:

2.53 - Research and Order EEG – Jackson

- FR 2.53.1: Research and compare at least 2 different EEG devices.
- FR 2.53.2: Select EEG for project use.
- FR 2.53.3: Place order through KU Shop.

Progress Summary:

2.54 - Implement UI in App for Features Implemented During Sprint 2 –

- FR 2.54.1: Add UI for focus algorithm.
- FR 2.54.2: Add UI for perceived focus input after session.
- FR 2.54.3: Display the comparison result for perceived focus in the UI (text summary and placeholder chart if possible).
- FR 2.54.4: Ensure UI integrates with hard-coded EEG data flow.
- FR 2.54.5: Ensure UI is functional and navigable without crashes.

Progress Summary: