EnerGreen Project - Iteration 2: Core App & Initial Insights

Sprint Period: July 22, 2025 – August 11, 2025 (As per Gantt Chart)

Overall Goal for Iteration 2:

To refine the IoT device firmware for accurate data, expand backend APIs to handle user and historical data, implement a core dashboard for data visualization, and begin initial data preparation for AI model integration.

Key Deliverables for Iteration 2:

- Stable and accurate IoT device firmware for data collection.
- Robust backend APIs for user management and historical data retrieval.
- Functional core dashboard displaying key energy metrics.
- Initial data pipeline and preparation for AI model training.
- Comprehensive internal testing report for all implemented features.

Detailed Task Breakdown and Preparation:

1. Task: S2.1 - Sprint 2 Planning & Requirements Refinement

- **Lead:** Project Team
- **Timeline:** July 22 – July 24, 2025
- **Description:** Kick-off meeting for Iteration 2. Review and finalize the detailed requirements for each task, ensuring clarity, feasibility, and alignment with overall project goals. Assign specific responsibilities within the team.
- **Preparation Steps:**
    - **Review Iteration 1 Completion:** Confirm all aspects of Sprint 1 (PZEM integration, basic data flow to Firestore) are stable and documented.
    - **Detailed API Specification:** For S2.3, define exact API endpoints, request/response formats, and authentication mechanisms for user and historical data.
    - **Dashboard Mockups/Wireframes:** For S2.4, create high-level mockups or wireframes for the core dashboard, focusing on what data will be displayed (real-time, historical trends) and how users will interact with it.

    - **AI Data Needs:** For S2.5, Cyril should outline the specific data format and volume required for initial AI model data preparation.
    - **Testing Strategy:** Outline the testing approach for this sprint, including types of tests and expected outcomes.

2. Task: S2.2 - Refine IoT Firmware for Data Accuracy

- **Lead:** Bryl Darel Gorgonio
- **Timeline:** July 24 – August 07, 2025
- **Description:** Optimize and fine-tune the ESP32 firmware for consistent and accurate data acquisition from the PZEM-004T. This includes addressing any remaining communication quirks, ensuring robust error handling, and optimizing data transmission.
- **Preparation Steps:**
    - **Deep Dive into PZEM Manual:** Bryl should re-read the PZEM-004T V3.0 manual [cite: 937e4689-7d09-40ed-848e-7bdda51a6cfb.jpg] to ensure all nuances of the Modbus protocol and register readings are correctly implemented.
    - **Error Logging:** Implement detailed logging within the firmware to capture communication errors, response anomalies, or power issues, aiding in debugging.
    - **Edge Case Testing:** Test the firmware under various conditions (e.g., no load, very low load, fluctuating load if possible) to ensure data accuracy and stability.
    - **Firmware Update Mechanism (Consideration):** While not a sprint deliverable, consider how future firmware updates might be deployed.
    - **Coordinate with Backend (Cyril):** Ensure the data format sent by the firmware aligns precisely with what the backend APIs expect.

3. Task: S2.3 - Expand Backend APIs (User, Historical Data)

- **Lead:** Cyril John Ypil
- **Timeline:** July 24 – August 07, 2025
- **Description:** Develop and expand the Cloud Function backend APIs to handle user-related operations (e.g., user profile update, potentially linking devices) and to retrieve historical energy data from Firestore and/or Cloud SQL for the dashboard.
- **Preparation Steps:**

  - **API Design:** Based on S2.1, define the RESTful endpoints for user profiles (GET, PUT) and historical data (GET with date ranges, device filters).
  - **Firestore/Cloud SQL Integration:** Cyril will implement the logic to query data from Firestore (for recent data) and Cloud SQL (for aggregated/long-term historical data). This involves writing efficient database queries.
  - **Authentication & Authorization:** Ensure all API endpoints are secured using Firebase Authentication, allowing only authenticated users to access their own data.
  - **Error Handling:** Implement robust error handling for API requests (e.g., invalid requests, database errors, unauthorized access).
  - **Documentation:** Create API documentation for front-end developers (Kobe, Alesna) to consume the new endpoints.

4. Task: S2.4 - Implement Core Dashboard (UI/UX)

- **Lead:** Kobe Amaro, Alesna Christine Anne
- **Timeline:** July 29 – August 09, 2025
- **Description:** Design and develop the initial user interface for the EnerGreen web dashboard. This includes creating the layout, navigation, and components to display real-time and basic historical energy consumption data.
- **Preparation Steps:**
  - **UI/UX Framework Selection:** Confirm the web framework (Confirmed! Vue Framework is used) and UI library (Tailwind CSS) will be used.
  - **Dashboard Layout:** Based on S2.1, create the visual layout for the main dashboard, including sections for real-time readings, summary statistics, and initial charts.
  - **API Consumption:** Integrate with the backend APIs (S2.3) to fetch and display user profile data and energy readings.
  - **Charting Library:** Select and integrate a charting library (e.g., Chart.js, Recharts) to display energy consumption trends.
  - **Responsiveness:** Ensure the dashboard is responsive and user-friendly across different screen sizes (desktop, tablet, mobile).

5. Task: S2.5 - AI Model Integration (Initial Data Prep)

- **Lead:** Cyril John Ypil
- **Timeline:** July 29 – August 09, 2025
- **Description:** Begin the foundational work for AI integration. This involves preparing the energy consumption data in a format suitable for machine learning models and potentially setting up initial data pipelines for future model training.
- **Preparation Steps:**
    - **Data Cleaning & Preprocessing:** Define and implement steps to clean and preprocess the raw energy data (e.g., handling missing values, outliers, normalization) from Firestore/Cloud SQL.
    - **Feature Engineering (Initial):** Identify potential features from the energy data that could be useful for predictive models (e.g., time of day, day of week, seasonal trends).
    - **Data Export/ETL Pipeline (Conceptual):** Plan how processed data will be moved to a suitable environment for model training (e.g., BigQuery, Cloud Storage). This might involve new Cloud Functions or Dataflow jobs.
    - **Initial Model Selection:** Cyril can research and select potential AI models for energy consumption prediction or anomaly detection, considering data requirements.
    - **Collaboration:** Coordinate with Kobe/Alesna for data needs, and with Bryl for data accuracy from the device.

6. Task: S2.6 - Internal Testing - Iteration 2

- **Lead:** Kobe Amaro, Alesna Christine Anne (As per Gantt Chart, but often a collaborative effort from the whole team)
- **Timeline:** August 07 – August 11, 2025
- **Description:** Conduct thorough internal testing of all newly implemented features in Iteration 2, including firmware accuracy, backend API functionality, dashboard display, and initial AI data preparation. Identify, report, and prioritize bugs.
- **Preparation Steps:**
    - **Comprehensive Test Cases:** Develop detailed test cases for each task (S2.2, S2.3, S2.4, S2.5). This includes functional tests, integration tests (e.g., dashboard correctly fetching from API, API correctly fetching from DB), and basic performance tests.

- **Bug Tracking System:** Utilize a shared system (e.g., Trello, Jira, or a simple spreadsheet) to log bugs with clear descriptions, steps to reproduce, and severity.
- **Regression Testing:** Perform quick checks on critical functionalities from Sprint 1 (IoT data ingestion to Firestore/Cloud SQL) to ensure no new bugs were introduced.
- **Feedback Loop:** Establish a clear process for developers to receive bug reports and for testers to verify fixes.
- **Test Data:** Prepare realistic test data scenarios for the dashboard and AI data preparation.