Energy Monitoring System Software Development – Project Management Outline

I. Introduction and Background

The Electric Power System Integrator (EPSI) profession aims to manage the power generation and consumption processes through automation systems and digitalization of essential metrics data, where engineers break the system into smaller parts and monitor how they are interacting with one another. Energy Monitoring System (EMS) is a part of this project, aiming to gives investors or stakeholders a simple and easy-to-use interface to dynamically monitor and manage the parameters involved in the process of energy consumption. This project focuses on developing a software solution for the EMS, with applications that allows remote users to easily interact with the power system, while communicating effectively with the existing hardware infrastructure though the Modbus RTU protocol and visually representing these data in a useful way. The software also needs to perform basic data analytics, compiling the received parameters into weekly, monthly, and annually reports to provide users with an overview of the power state.

II. Tasks and Timeline

a. User Interface Design and Data Acquisition

- Objective
  - Create a simple and functional user interface that displayed raw dummy data.
  - Connect and read raw data from hardware via Modbus RTU communication protocol.
- Deliverable
  - Working prototype of the UI.
  - Successful data logging using the required protocol.
- Deadline
  - 30 days (07/10/2024 – 06/11/2024).

b. Data Visualization

- Objective
  - Display the visual representation of the data collected from the hardware on the user interface.
  - The required parameters needed to be collected and displayed are:
    - Current (A).
    - Line voltage (V).
    - Phase voltage (V).
    - Power Factor ($\cos \phi$).
    - Active Power (W).
    - Reactive Power (W).
    - Apparent Power (W).

    - These data are to be represented as different types of graphs or charts, which are automatically updates as new readings received.
- Deliverable
    - An UI with fully functional display of the required parameters.
- Deadline
    - 30 days (07/11/2024 – 07/12/2024).

c. Data Logging and Reporting

- Objective
    - Continuously save the data collected from the hardware in a database.
    - Generate weekly, monthly, and yearly reports of these data as per user requests.
    - Implement option to generate these reports on the user interface.
- Deliverable
    - Power consumption parameters database.
    - An UI with reports generation functionality integrated.
- Deadline
    - 60 days (07/12/2024 – 05/02/2025).

d. System Optimization

- Objective
    - Analyze the data collected from the hardware based on a pre-set standards and offer suggestion for energy consumption.
    - Implement this functionality to the existing user interface.
- Deliverable
    - A complete UI with all required functionalities.
- Deadline
    - 60 days (06/02/2025 – 07/04/2025)

III. Initial Implementation Plan

a. User Interface Design and Data Acquisition

- C# with .Net framework will be used as the programming language for the project.
- WPF framework will be used as the GUI library with the .Net framework to design the user interface.
- WSMBT tool will be used as a library to control the Modbus RTU over TCP/IP protocol command in the .Net framework.
- The code base design will be structured as object-oriented.
- The initial UI will contain multiple pages corresponding to a specific parameter that required monitoring.
- Raw data will be shown as characters in each page, with the main page showing a combination of data from all parameters.

b. Data Visualization

- The graphical representations of the Current parameters will be line chart, bar chart, and gauge chart.
- The graphical representations of the Line and Phase Voltage parameters will be line chart, scatter plot, radar chart, and gauge chart.

- The graphical representations of the Power Factor parameters will be line chart, gauge chart, and donut chart.
- The graphical representations of the Apparent, Active, and Reactive Power parameters will be stacked bar chart (active and reactive stacked to apparent), area chart, and donut chart.
- The gauge chart and donut chart (for the Power parameters) will be shown in the main page (dashboard).
- All types of charts of a specific parameter will be shown in their corresponding page.

c. Data Analytics and Logging

- The read data will be stores using SQL server as the database structure.
- A function/class will be added to the code base to calculate the weekly, monthly, and yearly average, maximum, minimum, and median values for each parameter.
- These functions/classes will fetch data from the SQL server and pass them as parameters to perform the calculation.
- The results of these analysis are then stores in another SQL server.
- The UI will have a button in the main page to generate a report, where pressing it will show a drop-down list to choose between weekly, monthly, or annually. Choosing an option will call the analysis function to generate the report.

d. System Optimization

- A function/class will be added to the code base to compare the analysis values of the parameters with the standardized values provided by the investors.
- The function/class will return a suggestion message based on the result of the comparison.
- The message will appear in the generated reports.

IV. Communication Plan

- The project will be proceeded based on the sprint model of design, where a weekly report will be sent to the supervisor via email to summarize the project’s progression as well as specify the next objective and deadline.
- The supervisor will respond with feedback and recommendations on the project's progress.
- Student will reflect on the feedback and adjust the solution correspondingly, before trying to finish the next objective of the project before the specified deadline.
- This will be repeated each week until the project finished.