

Project Requirements Document

Team Pina Colada | AgMonitor

[Kaiwen Li](#) (Team Lead), [Alex Mei](#) (Scribe), [Jasun Chen](#), [Jayden Yu](#), [Yuyuan Wang](#)

Introduction

Background

Solar energy can only generate power during the day, which introduces fluctuation into the grid. Energy usage during the day can be cheaper because of excess solar generation, whereas usage during the night can be more expensive because of higher demand. Microgrid owners face the problem of having to decide when to activate their variable load, or perhaps don't want to decide at all. This can lead to inefficiencies in energy usage, waste money, and contribute to unnecessary greenhouse gas emissions depending on when power is used. Optimizing usage requires operators to have the relevant knowledge and foresight to smartly manage their resources.

Existing Solutions

In the current market, there are a few solutions that attempt to address this issue. Multiple companies like AgMonitor encourage shifting energy usage away from peak hours by sending price signals to the customer, indicating when they can potentially save on utility costs by either turning things off or changing when they turn on. This directly influences human behavior, which requires the customer to be involved in shifting energy usage. There is also currently research into smart charging and several pilot programs that attempt to implement smart charging programs, where grid information from the utility company is used to remotely manage when electric vehicles charge. These programs are not available to the wide public yet, and furthermore centralize control once again. Not to mention, a bottom-up approach from the microgrid perspective is better able to capture specific energy generation and needs. Optimizing on the macrogrid from the utility perspective doesn't necessarily guarantee optimization on the microgrid.

Project Specification

Our objective is to make energy management easy for microgrid owners. We expect our users to range from business owners to homeowners. The project will specifically focus on creating a

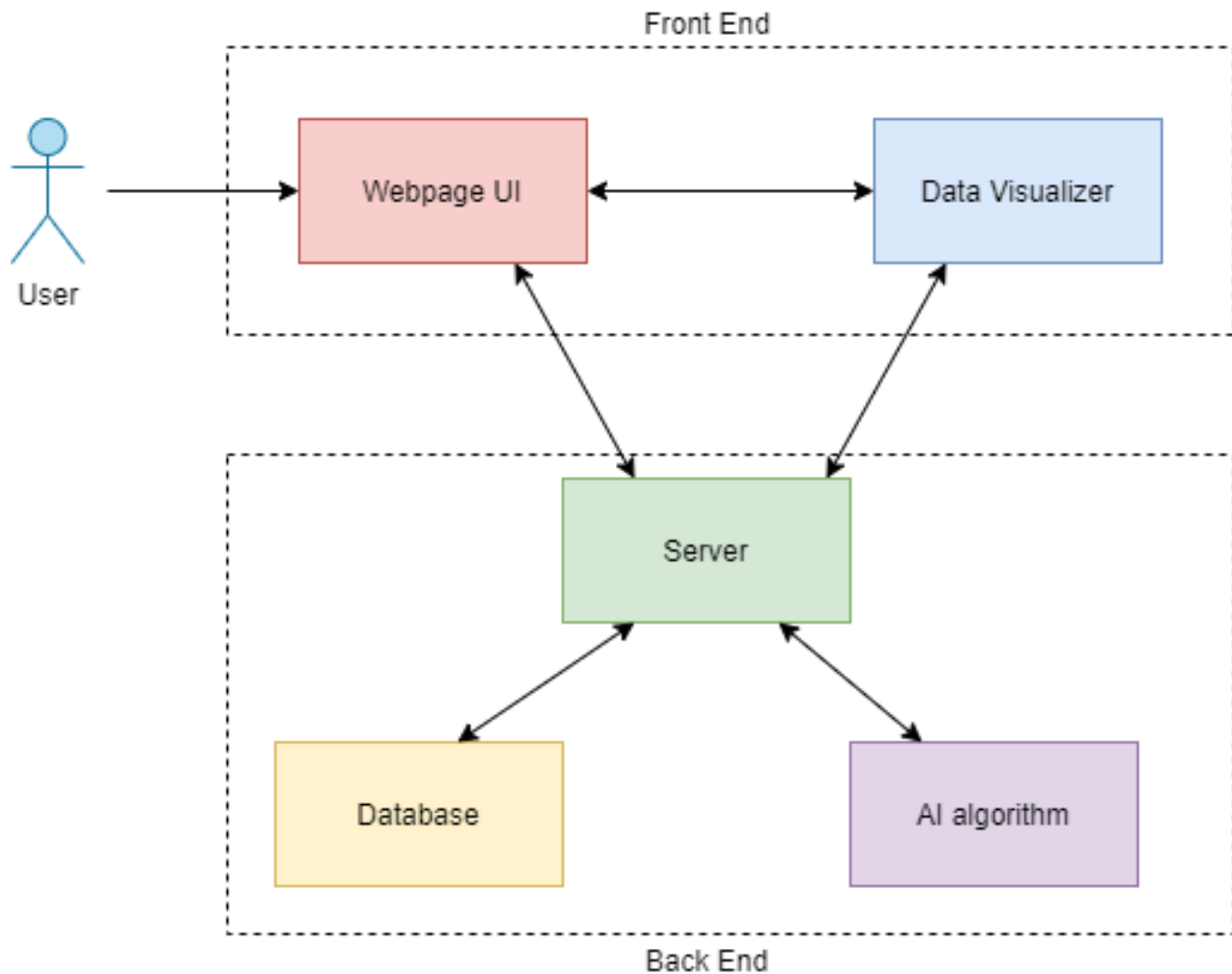
user-friendly tool that can effectively manage energy assets and variable load with minimal effort and knowledge required from the individual. Furthermore, this tool must constantly balance potentially competing interests such as reducing cost, integrating more renewable energy, and maintaining resiliency in the grid. While we focus our project mainly around the usage of electric vehicles, we hope to be able to generalize our solution to multiple forms of variable load in the larger context of a microgrid. Our product will have to be able to balance between maximizing renewable energy usage, saving cost for the user, and ensuring that bad weather or shut offs do not cause critical failures.

Our team goals are to:

- (i) Develop a web application to manage the impact energy storage (ES) and electric vehicle (EV) assets in the context of a micro-grid
- (ii) Implement an algorithm that maximizes the usage of local renewable energy and reduces the reliance on the California grid at times of peak load (4-9 pm)
- (iii) Validate the implementation of the web application on real-world use cases such as farms

Our project makes a couple of assumptions. While historical data and statistics can be used to make predictions for utility grid generation and usage because of the scale, sufficient data might not exist at the microgrid level. Furthermore, there is an extremely wide range of data sources as different utility providers and manufacturers might provide data in different ways. Integrating these different data streams into our application is outside our scope, but we can assume that given enough resources and time we would be able to.

System Architecture Overview



The diagram above shows a high-level architecture of our web application. The front end mainly contains the webpage user interface and the data visualizer Tableau (or equivalent) to display data visualization.

The user will be able to log in to the webpage to interact with the webpage UI. The user can upload the data and there will be a transformation in the backend that allows automated data processing. Then, the data will be stored in the database in the backend. The user will be able to visualize the categorical and quantitative data with the help of the data visualizer. By visualizing the data, the user can have a more vivid understanding of their energy consumption and usage. Furthermore, there will be an AI model in the backend connecting to the database that applies the

algorithm to the user's data. The webpage UI will show the result of the AI algorithm, therefore, the user can learn about what he/she could've done better to improve efficiency on energy usage.

Functional Requirements

Define a "user" as a "farm manager" or "owner" who manages their microgrid.

As a user, I can log in and log out on the website to ensure my personal data is secure.

- Tasks:
 - <https://trello.com/c/9nTDucC7/10-login-feature>
 - <https://trello.com/c/1dC0RH5d/11-logout-feature>
- Scenario 1: User puts in correct credentials and successfully logs in to the website, with the ability to access account-specific details.
- Scenario 2: User puts in incorrect credentials and it errors on attempted login.
- Scenario 3: User clicks logs out and removes the ability to access account-specific details.

As a user, I can upload a CSV file of data so I can use the data for optimization.

- Tasks:
 - <https://trello.com/c/MKilzcxr/15-csv-upload>
- Scenario 1: User uploads correctly formatted CSV file and data is stored to the internal database.
- Scenario 2: User uploads incorrectly formatted CSV and site throws error.

As an internal engineer, I can help transform customer data to provide automated data processing for customer convenience.

- Tasks:
 - <https://trello.com/c/USXlzsBk/19-data-cleaning>
- Scenario 1: Internal process does data cleaning to remove erroneous or missing data from the system.
- Scenario 2: Internal process applies appropriate data transformations based on user selection.

As a customer user, I can visualize my energy data for a better understanding of my energy usage.

- Tasks:
 - <https://trello.com/c/fli9Ollx/21-visualization-integration>
- Scenario 1: User visualizes categorical data using a pie chart / bar graph.
- Scenario 2: User visualizes quantitative data using histogram.

As a data scientist, I can visualize my data so I can help perform ad hoc data analysis for energy optimization.

- Tasks:
 - <https://trello.com/c/fli9Ollx/21-visualization-integration>
- Scenario 1: Data scientist visualizes data using scatter plot for correlation analysis.
- Scenario 2: Data scientist visualizes time series data to see seasonal shifts.

As a user, I can add and select metrics to compare my energy data to national averages.

- Tasks:
 - <https://trello.com/c/WKVK8n5J/26-visualization-metrics>

- Scenario 1: User selects moving average for time series data and generates new line on graph to compare.
- Scenario 2: User selects trend line to add trend line to visualize moving average.

As a user, I can select a subset of data to perform visualization on a contextualized subset of data.

- Tasks:
 - <https://trello.com/c/KwH0O3NY/27-visualization-data-filtering>
- Scenario 1: Users can select parts of data that they want to use because the whole data might be a large set to visualize.
- Scenario 2: Users only want to visualize a subset of data they select, and they want to choose their own way to visualize those data.

As a user, I can receive some criteria of actions to tell me how to optimize my microgrid energy usage.

- Tasks:
 - <https://trello.com/c/1IEh6fRP/25-ai-algorithm-implementation>
- Scenario 1: Users receive daily notifications of how to optimize their energy usage based on historical data.
- Scenario 2: Users can interact with the algorithm on the spot for immediate optimization strategy.

As an internal engineer, I can determine which optimization algorithm is best to help users optimize their microgrid most effectively.

- Tasks:
 - <https://trello.com/c/EzOVcL8J/23-ai-literature-review>
- Scenario 1: Some algorithm that is not really applicable to our context is omitted from the shortlist.
- Scenario 2: Algorithms that show potential are added to a shortlist for further consideration.

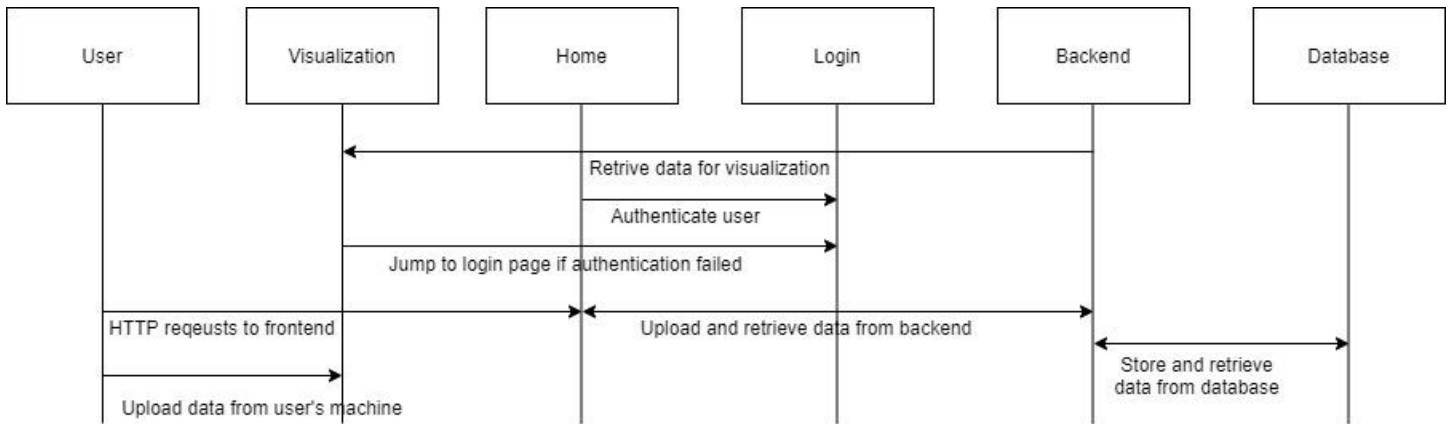
As a user, I can link my variable load assets so that its charging/discharging can be automatically controlled.

- Tasks:
 - <https://trello.com/c/BZUDkAn2/28-automatic-asset-configuration>
- Scenario 1: Users configure variable load assets and configure those parts to be automatically controlled by our optimization algorithm.
- Scenario 2: Should our algorithm fail, the variable load assets should default to a manually configured setting so they do not take down the entire microgrid.

Nonfunctional Requirements

- The system should be secure so credentials cannot be easily stolen.
- The system should maintain data integrity and privacy for users to trust the platform.
- The system should be organized in the backend so data storage is efficient.
- The system should be fast to allow for seamless data visualization analysis.
- The system should clearly disclose disclaimers for algorithmic predictions as accuracy may vary.
- The system should be able to support X users.
- The system should be able to support Y data per user.

System Models



Appendices

Technologies Applied

- a. Frontend:
 - ReactJS
 - Tableau
 - b. Backend
 - Django
 - c. Database
 - PostgreSQL
 - d. REST API
 - Tesla API
 - e. User authentication:
 - GoogleAuth
 - f. Deployment platform
 - Heroku
-