**CPSC 8740 – Final Project**

**Testing and Refinement Report**

**Name: - Sumit Reddy Munnuru**

**Smart Home Energy Consumption Prediction System**

---

## Overview

This week focused on testing the functionality, usability, and stability of the Smart Home Energy Consumption Prediction System. The goal of the testing phase was to identify any issues that may affect user experience, prediction accuracy, layout consistency, or system performance. The refinements made during this period were based on hands-on evaluation, user feedback, and iteration on both the back-end processing and the Streamlit-based interface.

Testing was conducted using manual walkthroughs, scenario-based input variations, and informal peer evaluations. These efforts helped ensure that the UI components behave as expected, that model predictions are generated reliably, and that the transitions between different sections of the dashboard are smooth.

---

## Testing Process

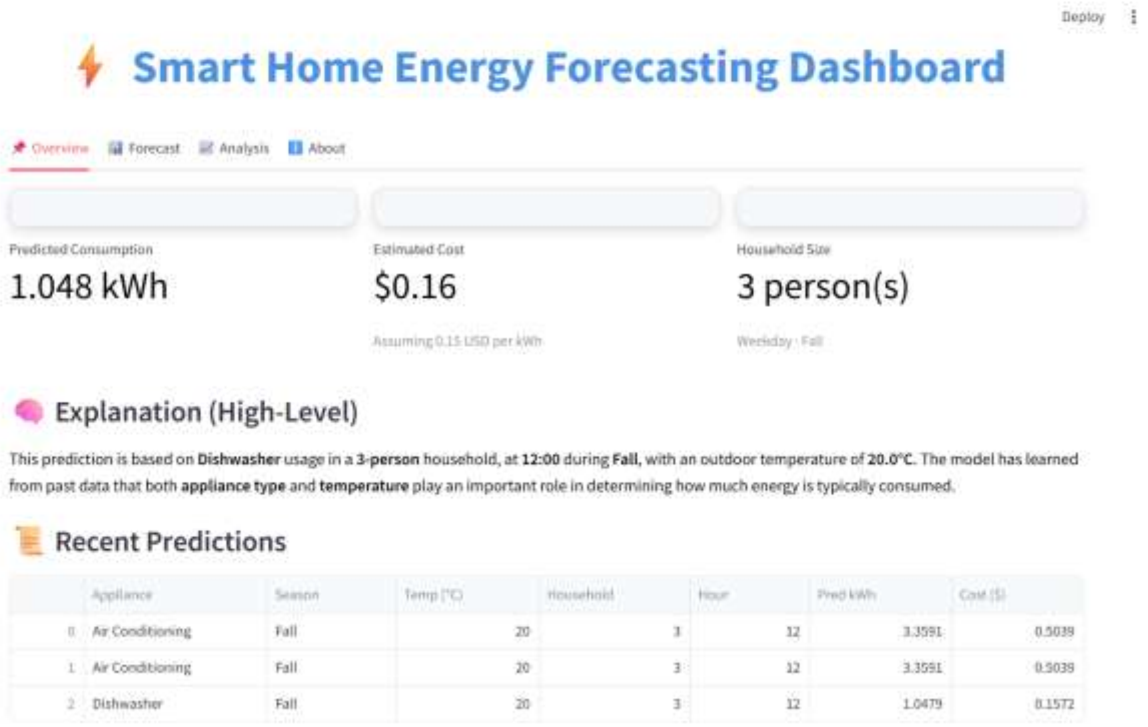
The testing process consisted of three major parts: functional testing, usability testing, and interface responsiveness testing.

### 1. Functional Testing

Functional testing focused on verifying that each feature in the system behaved correctly.

Areas tested include:

- Prediction output generation
- Sidebar input interactions
- Label encoder transformations
- Hourly forecast generation
- Appliance comparison computations
- CSV download functionality
- Session-based prediction history tracking



Smart Home Energy Forecasting Dashboard

Predicted Consumption
1.048 kWh

Estimated Cost
$0.16
Assuming 0.15 USD per kWh

Household Size
3 person(s)
Weekday · Fall

Explanation (High-Level)

This prediction is based on **Dishwasher** usage in a **3-person** household, at **12:00** during **Fall**, with an outdoor temperature of **20.0°C**. The model has learned from past data that both **appliance type** and **temperature** play an important role in determining how much energy is typically consumed.

Recent Predictions

| | Appliance | Season | Temp (°C) | Household | Hour | Pred kWh | Cost ($) |
|---|---|---|---|---|---|---|---|
| 0 | Air Conditioning | Fall | 20 | 3 | 12 | 3.3591 | 0.5039 |
| 1 | Air Conditioning | Fall | 20 | 3 | 12 | 3.3591 | 0.5039 |
| 2 | Dishwasher | Fall | 20 | 3 | 12 | 1.0479 | 0.1572 |

Figure 1: Successful Prediction Output During Functional Testing

Multiple input combinations were used to confirm that the model consistently produced a prediction and that the UI updated without errors.

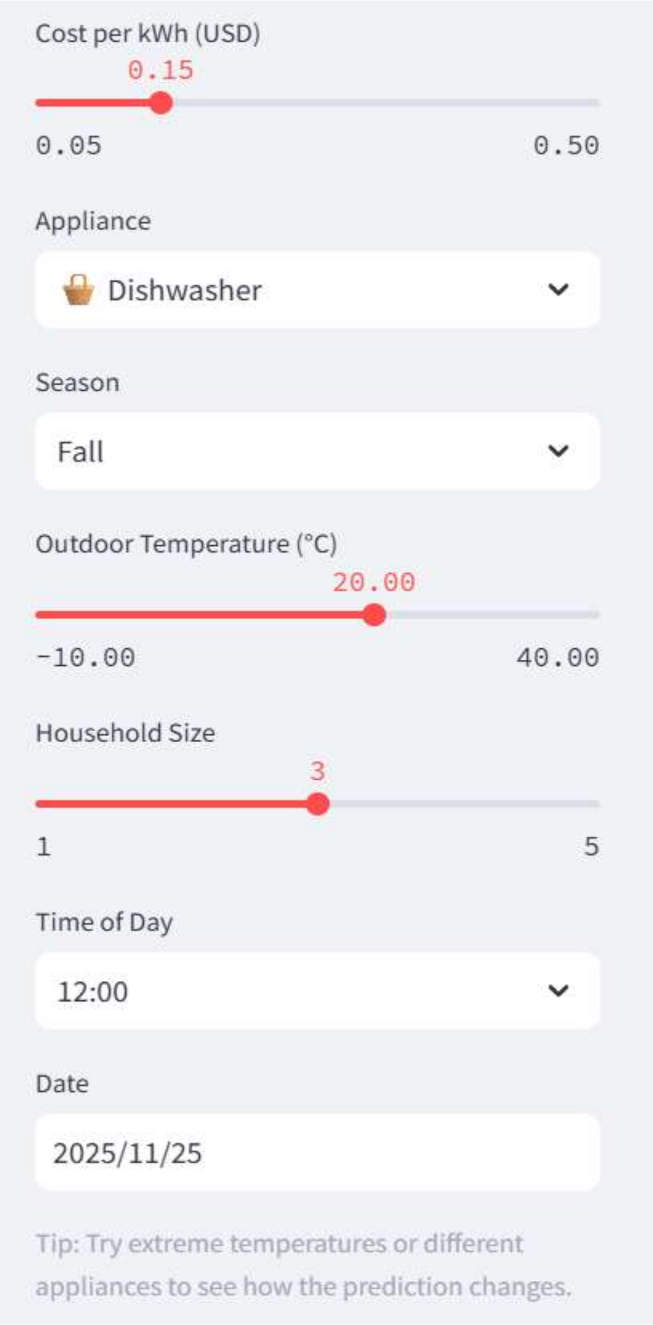
**2. Usability Testing**

Usability testing was completed by asking peers to interact with the dashboard and comment on the ease of use, clarity, and layout.

Participants were asked to complete tasks such as:

- Entering appliance and environmental settings
- Generating a prediction
- Interpreting the Forecast and Analysis pages
- Locating the cost estimation output
- Navigating through the tabbed layout

This feedback helped identify elements that required adjustment, including spacing around metric cards, text clarity in the explanation section, and the need for a brief introduction message before predictions are made.



Cost per kWh (USD)

0.15

0.05 0.50

Appliance

Dishwasher

Season

Fall

Outdoor Temperature (°C)

20.00

-10.00 40.00

Household Size

3

1 5

Time of Day

12:00

Date

2025/11/25

Tip: Try extreme temperatures or different appliances to see how the prediction changes.

Figure 2: User Input Panel Evaluated During Usability Testing

### 3. Interface Responsiveness Testing

The UI was tested on different screen sizes and browser environments to confirm that the layout remains readable and aligned.

Testing included:

- Desktop (primary target)
- Laptop screens
- Browser refresh and rerun behavior

- Theme switching responsiveness

Particular attention was given to the dark/light mode toggle due to previous issues with layout inconsistencies.

---

**Findings and Issues Identified**

**1. Dark Mode Rendering Issues**

Initially, dark mode caused inconsistencies in chart colors and card backgrounds. Since Streamlit re-renders components after each interaction, styling changes were not applying consistently.

**2. Prediction Button Re-Run Behavior**

Some users did not immediately realize that Streamlit re-executes the entire script when inputs change. This caused confusion when the forecast and analysis sections disappeared before a new prediction was generated.

**3. Chart Visibility**

In earlier versions, the hourly forecast chart was difficult to read in dark mode. Line thickness, color contrast, and grid visibility required improvements.

**4. Spacing Between UI Sections**

Metric cards and section headers were initially too close together, creating a congested appearance. This affected readability, especially in the Overview section.

**5. Label Encoder Errors**

During initial tests, entering an appliance name not recognized by the encoder produced an error. This was resolved once the UI was aligned with the exact appliance classes used during training.

**6. Session History Accuracy**

The session history table sometimes duplicated entries when the “Predict” button was clicked multiple times quickly. This was addressed by ensuring each entry was appended once per prediction.

---

**Refinements Implemented**

**1. Dark Mode Refactoring**

The dark mode feature was rewritten using CSS variables rather than HTML class manipulation to eliminate rendering inconsistencies. This resulted in stable theme changes without layout glitches.

**2. Improved Forecast Visuals**

Matplotlib's "dark_background" style was applied automatically in dark mode, improving readability. The line chart was adjusted to provide better contrast and smoother transitions.

**3. Enhanced Spacing and Layout**

Padding and spacing were increased around metric cards and section headers. This aligned the layout with the style seen in high-fidelity dashboards and made the interface easier to read.

**4. Clearer User Messaging**

Informational messages were added when no prediction had been generated yet. This helped prevent confusion when navigating between tabs.

**5. Sidebar Input Validation**

All input widgets were updated to directly correspond to valid encoder classes, preventing mismatched values and back-end errors.

**6. Streamlined Prediction History**

The prediction history system was refined to store only valid, completed predictions. This improved the accuracy of the Analysis page summary.

---

**Remaining Issues and Future Improvements**

While the current version of the interface is stable and functional, there are still areas that may be improved:

**1. Mobile Responsiveness**

Streamlit applications are primarily optimized for desktop usage. Additional adjustments would be required to fully support mobile layouts.

**2. Long-Term Data Logging**

Currently, prediction history is stored only for the active session. A future iteration may incorporate persistent storage (e.g., CSV logging or lightweight database integration).

**3. Extended Forecasting**

The system currently generates a single-day hourly forecast. Adding multi-day forecasting would enhance the analytical capabilities of the project.

### 4. Appliance-Specific Insights

Further enhancements may include explanations tailored to each appliance, offering more detailed reasoning behind prediction values.

---

### Conclusion

The testing and refinement phase significantly improved the functionality and usability of the Smart Home Energy Consumption Prediction System. The UI is now more stable, structured, and accessible, with predictable behavior across various user interactions. The refinements made during this week addressed the primary challenges identified during testing, resulting in a dashboard that is more intuitive and visually consistent. These improvements position the system well for final integration into the complete project submission.