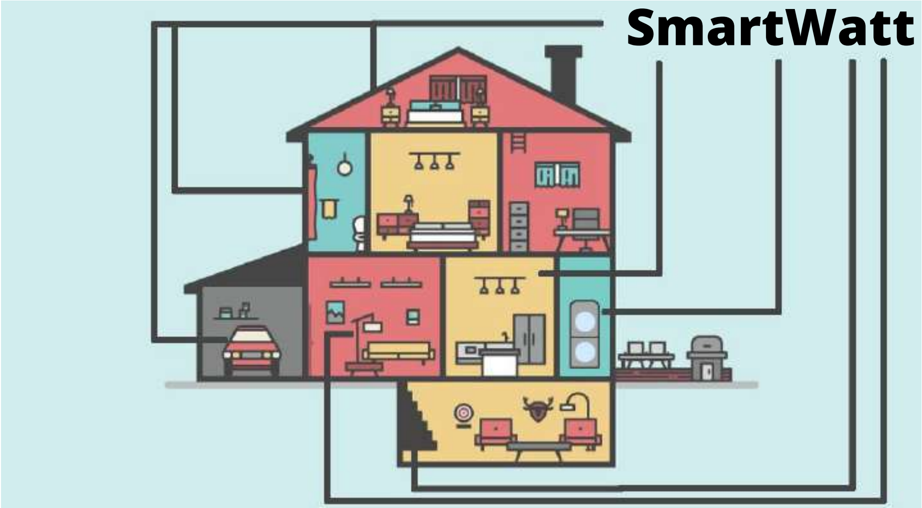
Use Case / Application

Studies show that 35% of home energy consumption in the US is wasted energy [1]

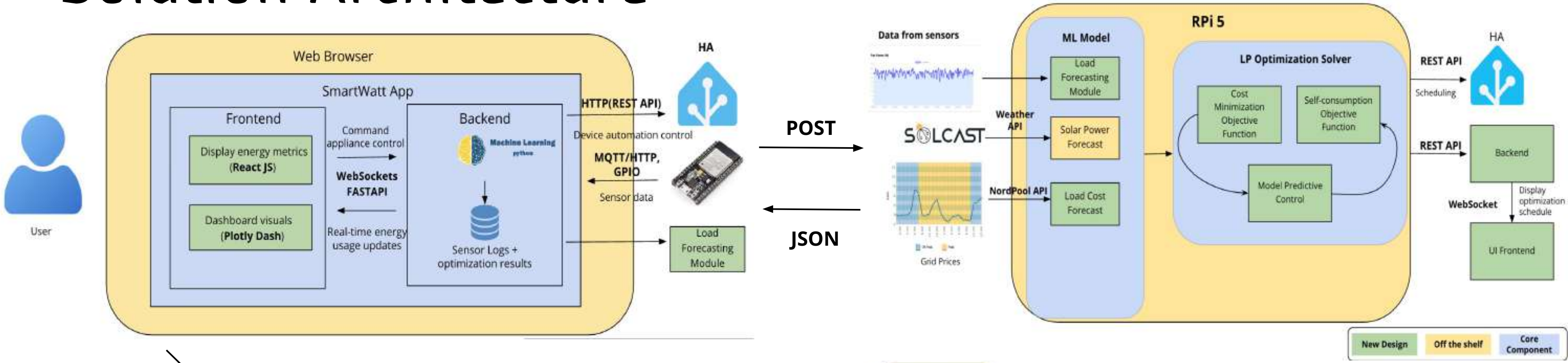
Energy demand is **increasing**, and in energy supply is now struggling to keep up

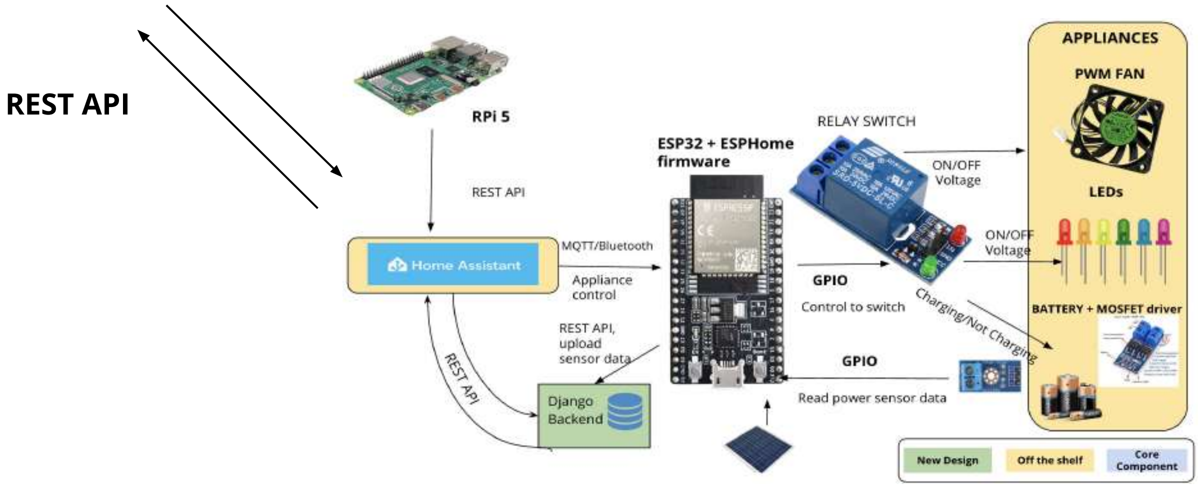


Quantitative Design Requirements

Use Case Requirement	Design Requirement
≥ 75% of solar panel energy consumed onsite	**≥ 30%** of deferrable loads are scheduled during peak solar generation hours.
≥ 10% reduction in electricity bills	**RMSE <25%** for grid price forecasting **≥20%** load shifting to lower-cost Time-of-use (TOU) periods
Dashboard updated with power consumption data every **5 minutes**	ESP32 sensors sampling at **1 Hz,** averaged over **5 minutes**
Power sensor should measure current and power within **98%** of actual reading.	**INA226**, error of **±0.1% - ±0.5%** (max) for current and power measurements
Actuate device (user can schedule devices to switch on/off) with **2s** delay	Fast API POST request latency **< 500ms** Time taken for ESP32 to receive & trigger on MQTT = **1s**

Solution Architecture

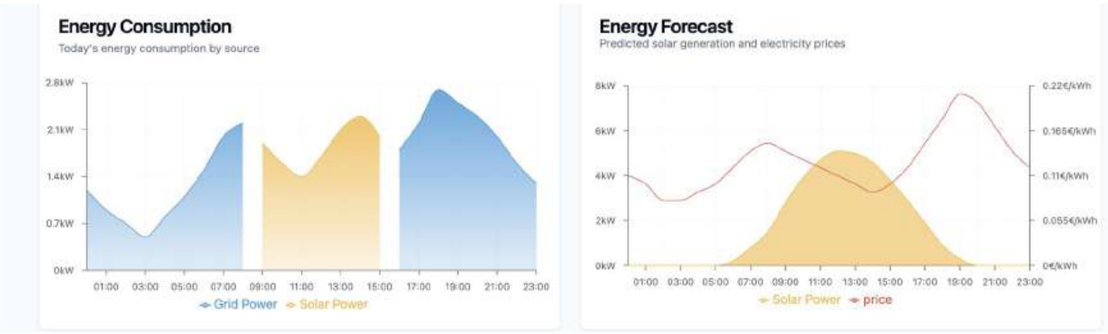




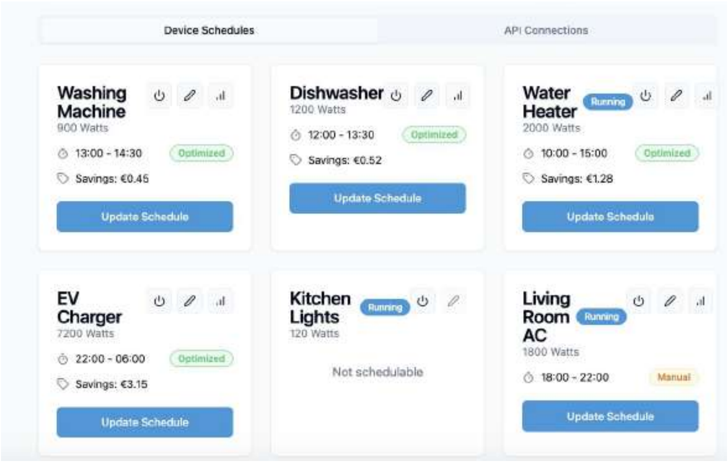
Ethical Considerations:

- Promotes low-cost, low-emission energy use
- Safe automation adaptable to diverse energy practices
- Easy to use, even for those with limited tech literacy or access

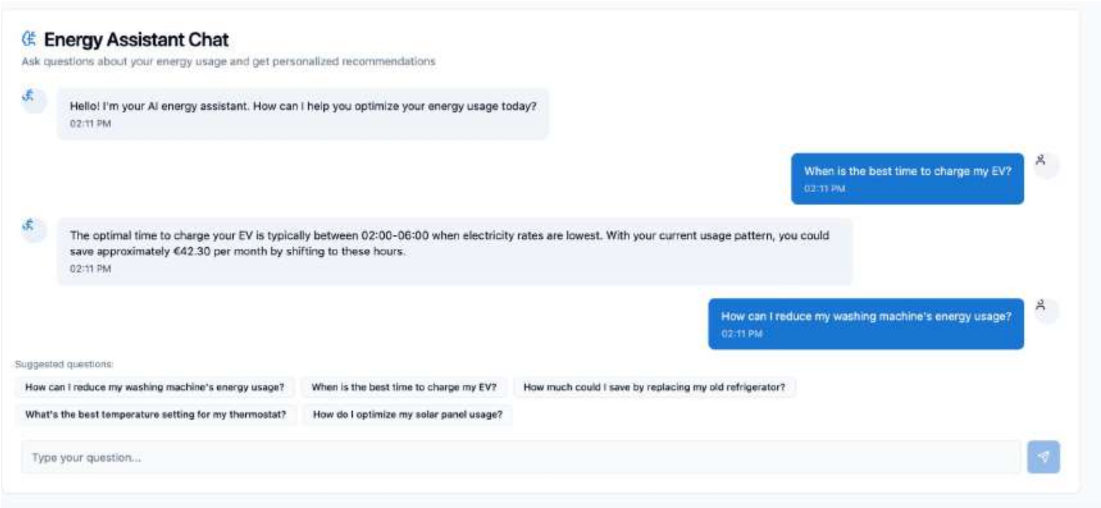
Dashboard



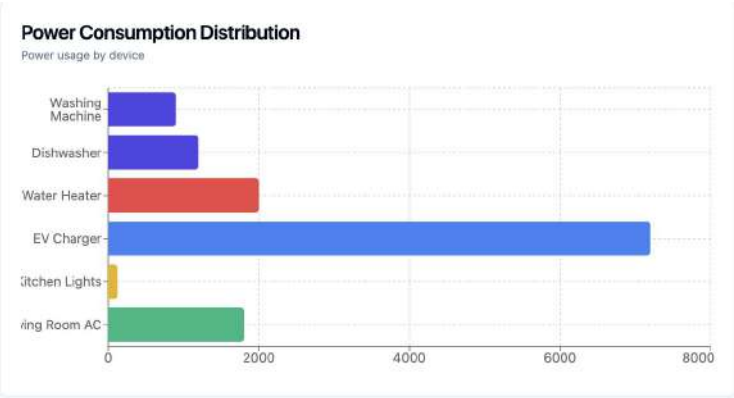
Energy Consumption
Today's energy consumption by source
2.8kW
2.1kW
1.4kW
0.7kW
0kW
01:00 03:00 05:00 07:00 09:00 11:00 13:00 15:00 17:00 19:00 21:00 23:00
Grid Power
Solar Power
Energy Forecast
Predicted solar generation and electricity prices
8kW
6kW
4kW
2kW
0kW
0.22€/kWh
0.165€/kWh
0.11€/kWh
0.055€/kWh
0€/kWh
01:00 03:00 05:00 07:00 09:00 11:00 13:00 15:00 17:00 19:00 21:00 23:00
Solar Power
price



Device Schedules
API Connections
Washing Machine
900 Watts
13:00 - 14:30
Optimized
Savings: €0.45
Update Schedule
Dishwasher
1200 Watts
12:00 - 13:30
Optimized
Savings: €0.52
Update Schedule
Water Heater
Running
2000 Watts
10:00 - 15:00
Optimized
Savings: €1.28
Update Schedule
EV Charger
7200 Watts
22:00 - 06:00
Optimized
Savings: €3.15
Update Schedule
Kitchen Lights
Running
120 Watts
Not schedulable
Living Room AC
Running
1800 Watts
18:00 - 22:00
Manual
Update Schedule

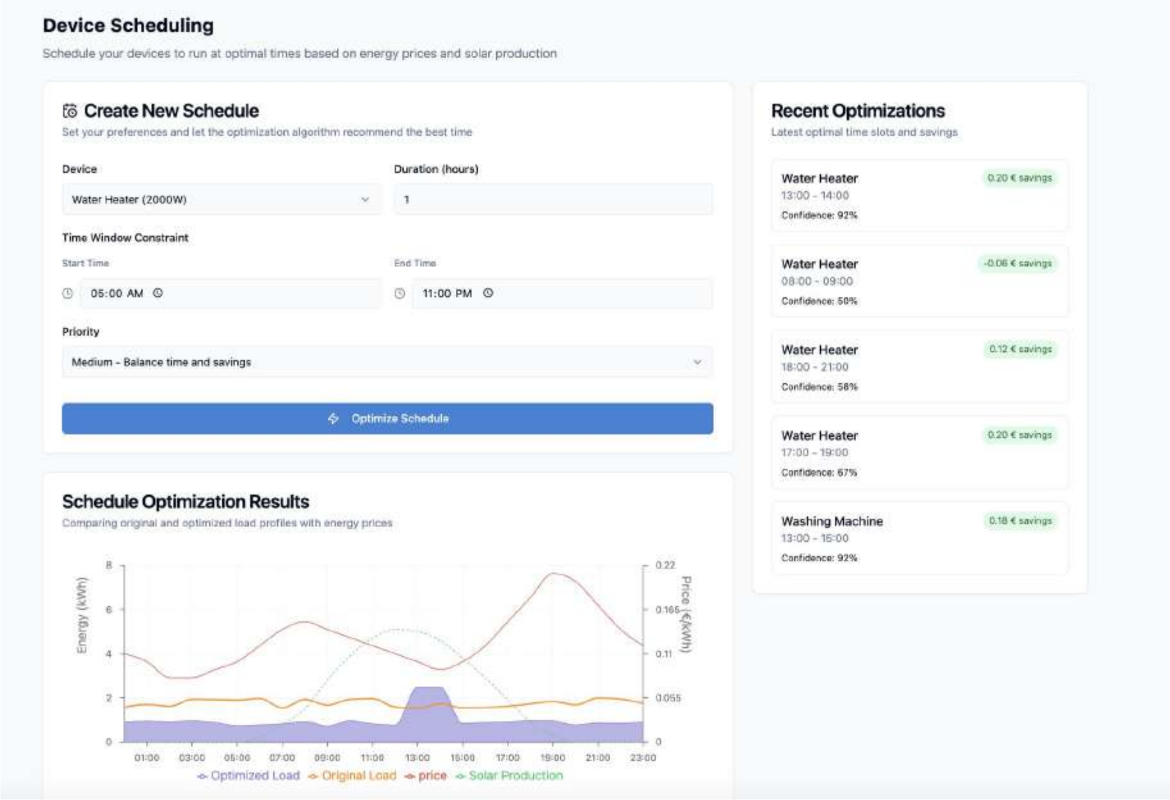


Energy Assistant Chat
Ask questions about your energy usage and get personalized recommendations
Hello! I'm your AI energy assistant. How can I help you optimize your energy usage today?
02:11 PM
When is the best time to charge my EV?
02:11 PM
The optimal time to charge your EV is typically between 02:00-06:00 when electricity rates are lowest. With your current usage pattern, you could save approximately €42.30 per month by shifting to these hours.
02:11 PM
How can I reduce my washing machine's energy usage?
02:11 PM
Suggested questions:
How can I reduce my washing machine's energy usage?
When is the best time to charge my EV?
How much could I save by replacing my old refrigerator?
What's the best temperature setting for my thermostat?
How do I optimize my solar panel usage?
Type your question...



Power Consumption Distribution
Power usage by device
Washing Machine
Dishwasher
Water Heater
EV Charger
Kitchen Lights
Living Room AC
0
2000
4000
6000
8000

Complete Solution - Optimization



Linear Programming Solvers - PULP_CBC_CMD and GLPK

Objective Function
Minimize total operational cost across all eligible time slots

$$\sum_t x_t \cdot (\alpha \cdot \text{price}_t - (1-\alpha) \cdot \text{solar}_t)$$

User defined priorities

Priority	Objective Focus	α
High	Minimize Grid Cost	0.8
Medium	Balanced Cost and Solar	0.5
Low	Maximize Solar Self-Use	0.4

Constraints

$$\sum_{t=s}^{e} x_t = D \quad \text{(run for exactly } D \text{ slots duration)}$$

$$x_t = 0 \quad \forall t < s \text{ or } t > e$$

$$x_t \in \{0, 1\}$$

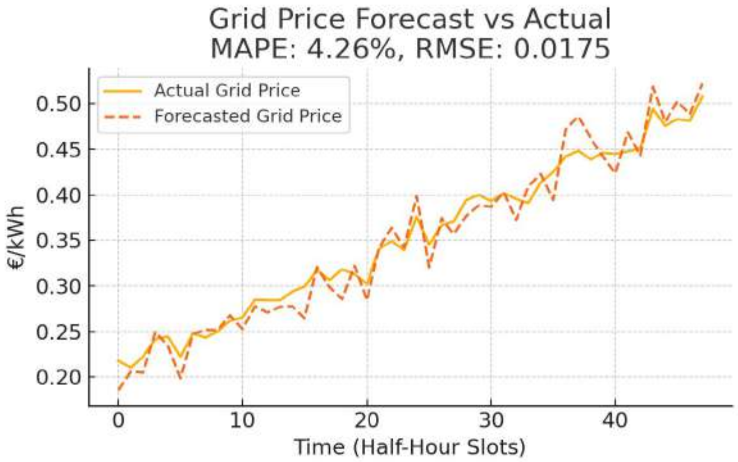
Complete Solution - Device Control


```
INFO:     Started reloader process [79820] using WatchFiles
INFO:     Started server process [79822]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO:     127.0.0.1:59503 - "GET / HTTP/1.1" 200 OK
INFO:smartwatt: Schedule POST received → entity=switch.maya_big_le
d, start=19:17, end=19:18
INFO:smartwatt:Schedule saved for switch.maya_big_led: {'start': '
19:17', 'end': '19:18'}
INFO:     127.0.0.1:59519 - "POST /schedule HTTP/1.1" 302 Found
INFO:     127.0.0.1:59519 - "GET / HTTP/1.1" 200 OK
```

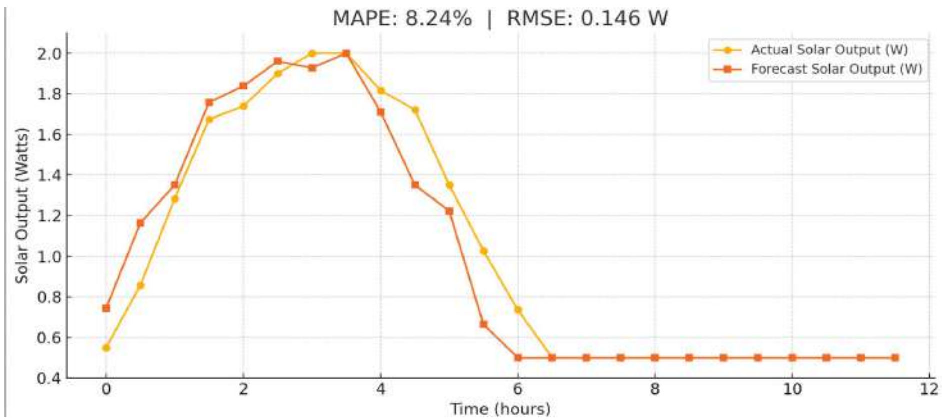
Design Tradeoffs

Proposed Solution	Final Solution
Meta's LLaMA 2 / Google's Gemma - Inconsistent answers, long latency - interface added to improve interpretability and allow natural language queries	**OpenAI GPT-4 via API** - More reliable, concise, and relevant - ~20-40% better accuracy, ~750 ms latency
Direct GPIO actuation via ESP32 polling - Required custom firmware and exposed bugs	**REST-based FastAPI control** - Easier debugging, centralized management - ~180ms actuation latency
On-demand ML inference during POST requests - Crashed server under load	**Preloaded background ML model** - Reduced inference latency from 3.2s → 620ms
Manual scheduling inputs (start/end only) - Not intuitive, poor user adoption	**Priority-based LP optimization** - Adaptive to constraints and user goals - Higher user satisfaction (↑ adherence)

Testing & Verification - ML Forecasts

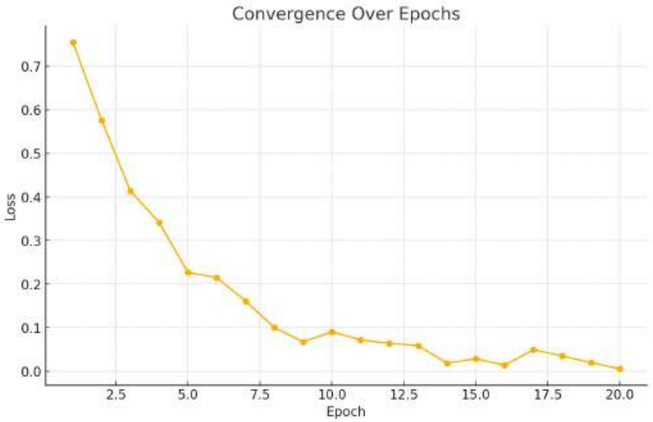


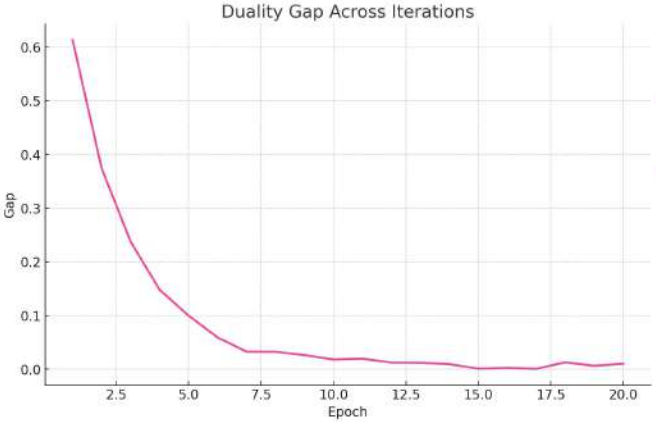
Solar Power Forecast vs Actual



- **Model complexity vs latency:**
  - Chose XGBoost for interpretability and fast inference (over LSTM/CNN + Transformer based regression models)
- **Data availability vs accuracy:**
  - Trained on 4 days of history, rest of data used in testing & validation

Testing & Verification - Optimization Performance





- Algorithm **converged across all test scenarios**, even for large 48-slot scheduling windows (24 hours).
- Final solutions showed a **near zero duality gap**, confirming that optimal solutions were reached with no discrepancy between the primal and dual problems.
- Feasibility Testing performed to ensure constraints (device duration, grid/solar capacity limits) were never violated

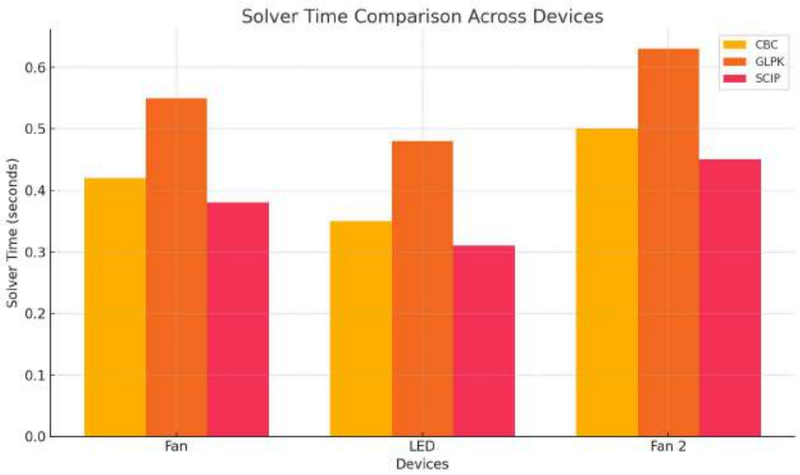
  and

  Stress tested with empty solar forecasts or flat grid prices, injected artificial price spikes.

Functionality	Measurement	Test Input	Test Output
Load Shifting Efficiency	% of load shifted to low-cost hours, high solar output hours	Power profile under default vs optimized schedule	**24%** of consumption moved to hours with price < daily median
Cost Reduction via Optimization	Monthly Energy Cost Change (€ and %)	Run devices with and without scheduling (assign random times) over 7 days	**17%** lower cost using SmartWatt optimizer

Testing & Verification - Optimization Latency

- **Convergence Time over Epochs**:
  - LP optimization for Fan, LED, and PWM Fan 2 devices consistently converges *within 15–20 epochs.*
- **Wall Clock Time:**
  - Algorithm converged in under 1s for most test scenarios, meeting use case requirements for device scheduling.
- Lightweight LP formulation avoids large overhead even when adding more devices or priority constraints.



Functionality	Measurement	Test Input	Test Output
Optimization Solver Performance	Solve Time (LP/MILP) (Wall Clock Time)	24-48 hr slot optimization with user defined priorities, price and power data	Solver returned within **1s** with valid schedule that satisfies constraints

Testing & Verification

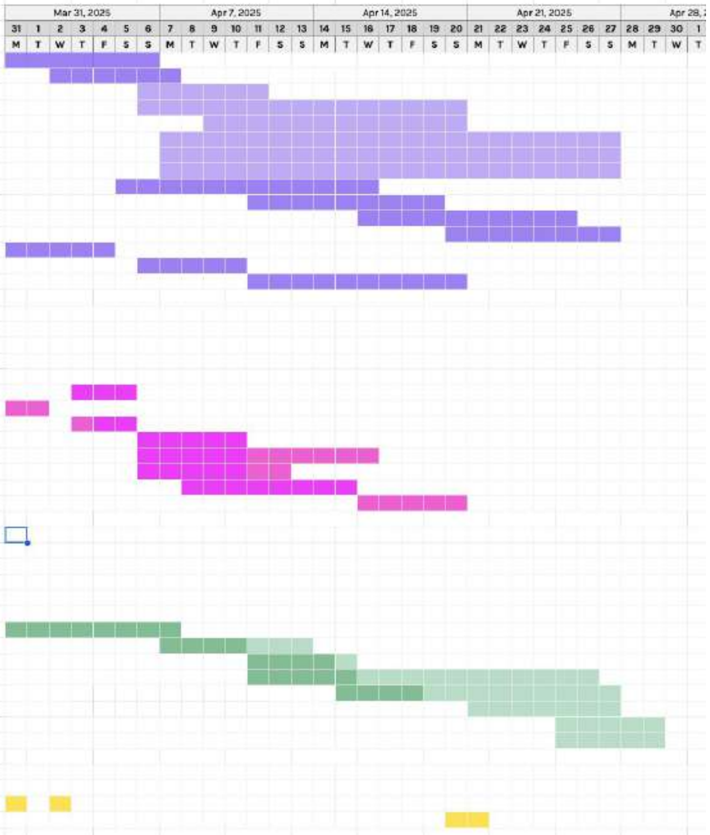
Functionality	Requirement	Test Input	Test Output
Device Control + Actuation Latency	**2s** delay	POST request to endpoint with device action and scheduling	All control signals result in correct ON/OFF state. The latency is **1.4s**
Backend Data Fetching Accuracy - Power readings from INA226 sensor	Measure current and power within **98%** of actual reading.	Fetch voltage, current, and power from INA226 sensor	**0.5%** power calibration error (from measured V and I values)

Lessons Learned

- **Stress Testing:** Running simulations with flat data and random spikes helped catch edge cases in the scheduling logic.
- **API Logging Is a Must-Have:** Debugging device scheduling is challenging! Request/response logging helped identify malformed payloads and latency bottlenecks.
- **Asynchronous Tasks Improve Responsiveness**
  Background task queuing (via FastAPI) was needed to avoid blocking main threads during optimization and inference. We learned that synchronous execution caused UI freezes and slower response to user actions.

Project Management

TASK	ASSIGNED TO	PROGRESS	START	END
		Week of:		
Software (ML Forecasting)				
Solar Power Forecast Output (API)	Anya Bindra	100%	2/3/25	2/10/25
Build load power forecast model	Anya Bindra	100%	2/12/25	4/4/25
Build energy cost forecast model	Anya Bindra	100%	2/18/25	4/4/25
Build solar panel energy forecast model	Anya Bindra	100%	2/25/25	4/4/25
Build Backend Typescript and Python App	Anya Bindra	100%	2/18/25	4/20/25
Build frontend dashboards	Anya Bindra	100%	2/25/25	3/25/25
Linear Programming and Optimization	Anya Bindra	80%	3/10/25	4/10/25
REST API connection to Home Assistant	Anya Bindra	100%	3/31/25	4/10/25
Device Actuation and Control	Anya Bindra	100%	3/31/25	4/20/25
LLM API connection to Optimization	Anya Bindra	30%	3/31/25	4/20/25
Test Optimization Framework	Anya Bindra	100%	3/21/25	4/20/25
Test API Connection	Anya Bindra	100%	3/25/25	4/20/25
Full System Integration Testing	Anya Bindra	50%	4/6/25	4/10/25
Slack	Anya Bindra	0%	4/11/25	4/20/25
Hardware (Model House)				
Define list of power consuming appliances	Maya Doshi	100%	2/7/25	2/10/25
Set up Raspberry Pi	Maya Doshi	100%	2/10/25	2/15/25
Source materials	Maya Doshi	100%	2/16/25	2/27/25
Build prototype for power systems with HA	Maya Doshi	50%	3/8/25	3/13/25
Start logging data and figure out scaling factors	Maya Doshi	100%	3/13/25	3/19/25
Write docs about how to access the data	Maya Doshi	10%	3/20/25	3/23/25
Prototype circuit to simulate power consumption	Maya Doshi	90%	3/27/25	4/1/25
Implement circuit to simulate power consumption	Maya Doshi	0%	4/3/25	4/9/25
Home Assistant Integration and Addon Packaging	Maya Doshi	20%	4/3/25	4/12/25
Help Anya with backend and developemnt	Maya Doshi	20%	4/3/25	4/12/25
Integrate the power components with the house	Maya Doshi	0%	4/10/25	4/16/25
Work on the Python Backend	Maya Doshi	5%	4/10/25	4/16/25
Slack	Maya Doshi	0%	4/16/25	4/20/25
System Integration (Dashboard)				
Source materials	Erika Ramirez	100%	2/12/25	2/19/25
Plan UI layout & wireframes	Erika Ramirez	100%	2/19/25	2/26/25
Design demo home layout	Erika Ramirez	100%	2/26/25	3/7/25
Start UI development (basic structure)	Erika Ramirez	100%	2/26/25	3/7/25
CAD Double Layer Wall	Erika Ramirez	100%	3/10/25	3/13/25
Laser Cut Wood Sheets	Erika Ramirez	100%	3/24/25	3/26/25
Laser Cut Acrylic Sheets	Erika Ramirez	100%	3/27/25	4/7/25
Chatbot - AI Recommendations + Analysis	Erika Ramirez	50%	4/7/25	4/13/25
Electronic Day/Night Visual	Erika Ramirez	90%	4/11/25	4/15/25
Construct Model House	Erika Ramirez	70%	4/11/25	4/30/25
Integrate Front-End / Back-End with Fast API	Erika Ramirez	20%	4/15/25	4/25/25
Finalize UI & demo home refinements	Erika Ramirez	0%	4/21/25	4/30/25
Final integration, testing & documentation	Erika Ramirez	0%	4/25/25	4/30/25
Slack	Erika Ramirez	0%	4/25/25	4/30/25
Deadlines				
Proposal Presentation	Anya Bindra	100%	2/2/25	2/3/25
Design Presentation	Maya Doshi	100%	2/16/25	2/17/25
Interim Demo	Everyone	0%	4/2/25	4/2/25
Final Presentation	Erika Ramirez	0%	4/20/25	4/21/25



Anya:

- Optimization+ML
- Backend + Frontend
- Dashboard
- Device Control

Maya:

- Home Circuitry
- Setup RPi Network

Erika:

- Demo Home Construction
- Chatbot
- Web App Front-End / Back-End Integration