Smart Watt Proposal

Anya Bindra,
Maya Doshi,
Erika Ramirez

Feb 07, 2025

Use Case

Households with solar panels underutilize self-generated power, relying on the grid excessively. At best, a household consumes between 20% and 50% of its self-generated solar power.[1]

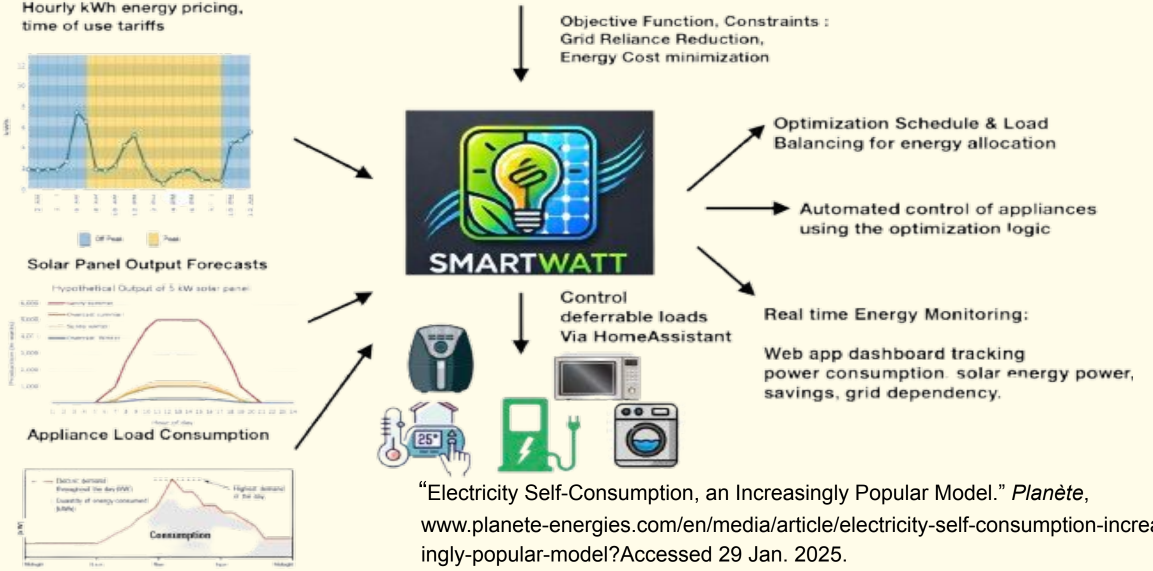
Rising energy prices strain household budgets, especially when energy usage is not optimized.

Existing energy management solutions lack dynamic optimization based on real-time weather, time-of-use tariffs, load demand, fluctuations electricity prices, grid feed-ins for surplus energy

Expected Behavior of SmartWatt:

Shift high-power appliance usage to times when solar generation is highest or grid electricity is cheapest

ECE Areas : Software, Signals (Machine Learning + Optimization)



[1] "Electricity Self-Consumption, an Increasingly Popular Model." *Planète*, www.planete-energies.com/en/media/article/electricity-self-consumption-increasingly-popular-model?Accessed 29 Jan. 2025.

Use Case Requirements : Optimization/ML Performance

Feature	Metric	Target
Solar self-consumption % produced by optimization schedule	Percentage of solar energy produced that is consumed onsite.	≥ 75%
Forecast solar power generation for 24h	Predict PV output off solar irradiance data, weather and panel characteristics.	≥80% (accuracy based off MAPE/RMSE on test data), updated every 30 min
Forecast time-of-use grid prices for 24h	Prices fed into the objective function for the scheduling	≥75% (accuracy based off MAPE/RMSE on test data), updated every 30 min
Electricity cost + usage reduction	Percentage reduction in electricity cost through optimal appliance scheduling.	≥ 10% (on test data)
Time Of Use (TOU)-Based Load Shifting	Percentage of deferrable load consumption shifted to low-cost TOU periods.	≥ 20%
Real-Time Load Balancing Accuracy	Algorithm's accuracy in matching load demand with available power.	≥ 90%

Use Case Requirements : Model House Simulation

Feature	Metric	Target
Simulated Household Power Demand	Ability to replicate realistic residential energy consumption patterns.	Scaling factor of 1/1000 to simulate 1–5 kW dynamic load range
Appliance Power Draw Simulation	Ability to adjust appliance loads dynamically based on schedule & demand response.	0.1W – 1W per appliance
Load Switching Granularity	Ability to switch appliances on/off at defined time intervals.	1–5 min resolution
Real-Time Load Variability	Ability to simulate energy demand spikes, peak hours, and seasonal variations in PV O/P	Fluctuations of ±20-30% in total demand, power consumption
Solar Energy Simulation	Ability to integrate solar PV generation & battery storage simulation.	1–10 kW PV output (scaled down to 10 - 100 W),5–20 Wh battery (scaled down)
Grid Import & Export Simulation	Ability to mimic real-world TOU pricing and energy export conditions.	Dynamic TOU tariff adjustment based off API calls

Use Case Requirements : System Responsiveness + Control

Feature	Metric	Target
Real-Time Energy Optimization Latency	Time taken by optimization algorithm to compute optimal load scheduling.	≤1s
Power Monitoring Update Rate	Frequency at which the system logs household power consumption	1 Hz (1 sample/sec)
Grid Switch Latency	Time taken to switch between solar, battery, and grid power sources dynamically.	≤1s
Dashboard UI Response Time	Time taken to load real-time energy monitoring data onto UI	≤3s
Appliance Switching Latency	Time to execute appliance on/off commands and control appliances , control signals to actuators (relays, switches)	≤250 ms
Sensor & Actuator Update Rate	Sampling rate for getting sensor data	0.1 Hz

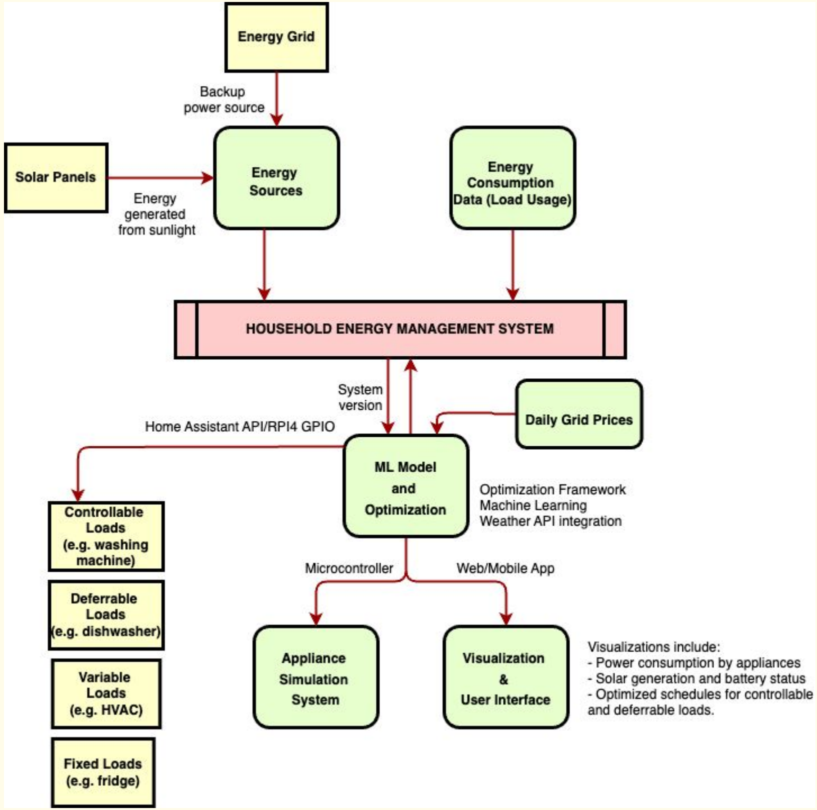
Technical Challenges

Challenge	Proposed Solution
Ensuring Load Balancing Accuracy (≥ 90%)	Use a hybrid ML + rule-based approach (autoregressive forecasting, real-time rules for immediate response). Implement rolling time windows (5-15 min) for adaptive balancing. Test the algorithm under various scenarios (e.g., cloudy days, peak hours) to ensure robustness
Scaling Power Simulation Accurately	Use resistive loads instead of live AC loads Apply power profiling to mimic real appliance curves, and include relay-based safety cutoffs. Use simulation software (Simulink) to validate the scaling factor and ensure accuracy.
Real-Time Control of Appliances (≤1 Second Response Time)	Use microcontrollers (e.g., ESP32, Raspberry Pi 5) to handle real-time control. Cache scheduling data locally on ESP32 for instant execution Implement prioritization logic to handle critical tasks (e.g., switching energy sources) before non-critical tasks (e.g., logging data).

Technical Challenges

Challenge	Proposed Solution
Achieving ≥ 75% Accuracy in Forecasting Time-of-Use grid prices for 24h. TOU prices fluctuate due to real-time demand, market conditions, and unpredictable factors (e.g., weather, grid congestion).	ARIMA for Short-Term Predictions (0–6 Hours Ahead) → Captures immediate price variations. LSTM/Transformer Model for 6–24 Hour Forecasting → Accounts for longer-term TOU price trends. XGBoost for Feature-Based Forecasting → Uses real-time energy market data (demand-supply, temperature, grid demand spikes).
Circuitry & Hardware Integration with Home Assistant	Use ESPHome firmware on ESP32 relays for direct HA integration. Use Modbus RTU (RS485) for high-power load monitoring
Real-Time Communication Between ML Model & Home Assistant (low latency <250ms)	Use MQTT JSON-based payloads for sending appliance schedules from ML model to HA. Implement REST API integration as a backup method if MQTT fails.

Solution Approach



Software/Algorithm	Purpose
XG Boost/LSTM/autoregressive forecasting (Scikit learn + Tensorflow)	Forecast solar generation & TOU pricing for scheduling.
Linear Programming (Model Predictive Control)	Optimize load shifting & appliance scheduling.
Grafana + InfluxDB + Django	Logs & visualizes energy data trends.
Home Assistant API (REST / WebSockets)	Remote automation & UI interaction.

Component	Purpose	Justification
ESP32/ESP826	Microcontroller for load control & sensors	Low-cost, Wi-Fi-enabled, integrates with ESPHome.
Solid State Relays	High-speed switching of appliances	Faster, safer, and longer lifespan than mechanical relays.
Power Monitoring Sensors (INA219, PZEM-004T)	Measure real-time power usage	1 Hz sampling rate ensures accurate tracking.
Programmable DC Electronic Load	Emulates appliance power consumption dynamically	0-150V, 0-30A, 0-200W
DS3231 RTC Module	Provides accurate real-time scheduling when offline.	I2C interface, ±2ppm accuracy

Solution Approach : ML Datasets (as of now)

ML Task	Dataset	Source	Purpose
TOU Price Forecasting (Next 24h)	Real-Time Grid Status Data	https://www.gridstatus.io/	Fetch real-time TOU pricing & demand forecasts for different regions
Solar Power Generation Forecasting	Historical & Forecasted Weather Data	https://solcast.com/ Solcast API	Provides solar irradiance, cloud cover, and weather conditions for PV output prediction
Household Load Demand Forecasting	Individual Household Electric Power Consumption Dataset	https://archive.ics.uci.edu/dataset/235/individual+household+electric+power+consumption	Real-world household power usage patterns for load prediction
Appliance Energy Usage Profiles	REFIT Smart Home Dataset energy consumption from 20 UK homes, Tracebase Appliance → Detailed power traces of over 150 household appliances, UK-DALE Dataset → High-resolution (1 Hz) individual appliance power consumption.	https://pureportal.strath.ac.uk/en/datasets/refit-electrical-load-measurements-cleaned	Load Flexibility Classification → Train a model to categorize appliances as deferrable or non-deferrable, Improve appliance-level scheduling & load shifting

Testing, Verification and Metrics

- Meeting Use-Case Requirements
  - **Energy Cost Reduction:**
    - Compare power consumption and grid cost before and after optimization, ensuring at least a 15% reduction in energy expenses.
    - Run simulations using real TOU pricing from GridStatus.io API. Compare results using baseline (random load scheduling) vs. ML-based scheduling.
  - **User-Controlled Scheduling:**
    - Conduct usability testing to confirm users can successfully defer or prioritize appliance usage via the UI in under 10 seconds. Home Assistant Event Listener → Captures real-time user inputs and actions
  - **Real-Time Monitoring:**
    - Validate that the system provides live energy consumption updates within 2s latency.
    - Python Time Profiling (`time.time()`) to measure execution time per iteration.
    - MQTT Broker Logs → Measure timestamp differences between data sent & displayed
- Simulate various conditions using real world data in both software and the model house

Tasks & Division of Labor

Anya

- **ML & Optimization Lead:**

Focusing on forecasting models and scheduling algorithms

Maya

- **Hardware:**

Setting up sensors, power measurement, and system communication

Erika

- **Integration:**

Making the dashboard, user experience, and performing system validation.

Schedule

TASK	ASSIGNED TO	PROGRESS	START	END
Software (ML Forecasting)				
Build load power forecast model	Anya Bindra	0%	2/12/25	2/19/25
Build energy cost forecast model	Anya Bindra	0%	2/19/25	2/26/25
Generate household power consumption data	Anya Bindra	0%	3/5/25	3/12/25
Build solar panel energy forecast model	Anya Bindra	0%	3/12/25	3/19/25
Optimize models & integrate	Anya Bindra	0%	3/19/25	3/26/25
Integrate - POST outputs to HomeAssistant API	Anya Bindra	0%	3/30/25	4/6/25
Slack	Anya Bindra	0%	4/11/25	4/20/25
Hardware (Model House)				
Define list of power consuming appliances	Maya Doshi	0%	2/12/25	2/19/25
Source materials	Maya Doshi	0%	2/12/25	2/19/25
Connect simulated sources to HomeAssistant	Maya Doshi	0%	2/19/25	2/26/25
Build circuit to simulate power consumption	Maya Doshi	0%	3/10/25	3/19/25
Build model house	Maya Doshi	0%	3/19/25	3/26/25
Integrate model solar panel for demo	Maya Doshi	0%	3/26/25	4/2/25
Slack	Maya Doshi	0%	4/13/25	4/20/25
System Integration (Dashboard)				
Source materials	Erika Ramirez	0%	2/12/25	2/19/25
Set up Raspberry Pi	Erika Ramirez	0%	2/19/25	2/26/25
Establish communication protocol	Erika Ramirez	0%	3/12/25	3/19/25
Implement HomeAssistant API	Erika Ramirez	0%	3/19/25	3/26/25
Create user interface / dashboard	Erika Ramirez	0%	3/26/25	4/2/25
Interactive scheduling UI	Erika Ramirez	0%	4/2/25	4/9/25
Test whole system integration	Erika Ramirez	0%	4/9/25	4/16/25
Slack	Erika Ramirez	0%	4/16/25	4/20/25
Deadlines				
Proposal Presentation	Anya Bindra	0%	2/2/25	2/3/25
Design Presentation	Maya Doshi	0%	2/16/25	2/17/25
Final Presentation	Erika Ramirez	0%	4/20/25	4/21/25