TÉCNICO LISBOA

Smart Home Energy Management

João Gabriel Sofio Batista Peres Sabino

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisor(s): Prof. Alberto Manuel Ramos da Cunha

Examination Committee

Chairperson: Prof. Pedro Tiago Gonçalves Monteiro
Supervisor: Prof. Alberto Manuel Ramos da Cunha
Member of the Committee: Prof. Miguel Filipe Leitão Pardal

July 2022

In memory of two mentoring figures that passed during my academic path. To Matias Batista, my grandfather, and to Prof. Renato Nunes, my original supervisor. Both were incredibly helpful, supportive and understanding during the ever so short time we shared.

Acknowledgments

I would like to express my gratitude to Prof. Alberto Cunha, my dissertation supervisor, who accepted to replace my late supervisor Prof. Renato Nunes and gave me the opportunity to continue pursuing the topic of energy management. To the Instituto Superior Técnico institution and community that taught me so much, and challenged me to go beyond.

I also wish to thank my close family and friends. This dissertation work wouldn't be possible without the immense emotional support given to me by all. My parents, José Gabriel and Maria Delfina, and my grandmother, Catarina Batista, did everything they could to help me in this journey that is now coming to a close. My girlfriend, Ana Rita, has accompanied me for the better half of my university years. I carry the weight of countless moments of happiness, laughter and solidarity shared with everyone.

Resumo

No último século, o consumo de energia decuplicou a nível mundial. Apelos pela sustentabilidade ambiental, acompanhados de aumentos no custo energético, levam empresas e indivíduos a gerir os seus consumos de energia de forma mais eficiente. Programas de Resposta Dinâmica de Consumos (RDC) foram criados de modo a incentivar contributos individuais e coletivos para a eficiência da rede energética. A presente dissertação explora diversos programas de RDC implementados por empresas de distribuição de eletricidade, hábitos de consumo por parte de clientes residenciais e a dinâmica de incentivos em edifícios e complexos residenciais multifamiliares. Nesta dissertação são também analisados Sistemas de Gestão de Consumo Residencial (SGCR) relativamente à abordagem, poupanças energéticas e flexibilidade para o utilizador. É proposto um SGCR focado na distribuição da energia de pico. O sistema apresentado agenda a execução de eletrodomésticos à medida que o utilizador efetua os pedidos, decidindo com base na informação presente. As decisões são baseadas no consumo de energia e nas preferências do utilizador quanto à prioridade e agendamento dos dispositivos. Além disso, são integrados sistemas de baterias e de auto-produção. Múltiplas instâncias do sistema podem cooperar no sentido de equilibrar o consumo ao longo do tempo. Foram alcançadas reduções no consumo de pico até 45%, quando comparado com casas sem sistema de gestão de energia, em ambiente de simulação. Os testes efetuados revelam maior equilíbrio de consumo particularmente durante a madrugada.

Palavras-chave: Redes Inteligentes, Gestão de Energia, Agendamento de Cargas, Resposta Dinâmica de Consumos

Abstract

In the last century, energy consumption has increased tenfold. Global calls for environmental sustainability along with rising energy costs push companies and individuals to manage energy more efficiently. Demand Response (DR) programs were developed to promote individual and collective contributions to the efficiency of the grid. This dissertation reviews DR programs implemented by energy distribution companies, consumption habits from residential consumers and the incentive dynamic in shared residential complexes. Existing Home Energy Management Systems (HEMS) are analysed regarding approach, energy savings and flexibility. A HEMS focused on peak distribution is proposed. The system is able to schedule the execution of appliances as they are requested, deciding with present information. Scheduling decisions are based on energy consumption and user preferences in device priority and schedulability. Moreover, battery and PV systems are integrated, and different instances of the server application are able to cooperate towards load distribution. Peak reductions of up to 45% versus unmanaged households were achieved in testing simulations, with additional benefits to demand balance in overnight periods.

Keywords: Smart Grid, Energy Management, Demand Response, Appliance Scheduling

Contents

List of Tables

List of Figures

Nomenclature

ADAWR	Average Delay-to-Acceptable-Wait Ratio
BEV	Battery Electric Vehicle
BSS	Battery Storage System
CLI	Command Line Interface
CP	Common Property
CPP	Critical Peak Pricing
DR	Demand Response
DSM	Demand-side Management
HEMS	Home Energy Management System
HVAC	Heating, Ventilation and Air Conditioning
KPI	Key Performance Indicators
MQTT	Message Queuing Telemetry Transport
ORM	Object-Relational Mapping
PAR	Peak-to-Average Ratio
PHEV	Plug-in Hybrid Electric Vehicle
PLC	Power Line Communication
RTP	Real-Time Pricing
SDG	Sustainable Development Goal
TOU	Time-of-Use

Chapter 1

Introduction

Energy management solutions are increasingly necessary, amid disruptions in European energy chains, soaring energy costs and growing environmental awareness. This dissertation provides an overview on problems in energy management at household and distribution levels, along with existing solutions. Literature on end-user consumption habits, appliance categorization and demand-side management techniques is reviewed. Moreover, an interactive, best-effort scheduling solution focused on preventing consumption peaks is proposed. This chapter elaborates on the relevance of energy management, outlines the contents of each chapter and the goals to be achieved with the thesis.

1.1 Motivation

Energy harnessing has been a key factor for the development of humankind since the control of fire. Nations and territories have been invaded and exploited for their resources. Industrialization accelerated the correlation between energy consumption and economic development, still found to this day [1]. Energy consumption grew exponentially during the last century, leading to concerns of resource exhaustion and environmental damage [2]. From the late 1980s, climate awareness spread and paved the way to a renewed notion of sustainable development.

In 2018, during the High-Level Political Forum on Sustainable Development, it was stated that "energy is central to nearly every major challenge and opportunity the world faces today, including poverty eradication, gender equality, adaptation to climate change, food security, health, education, sustainable cities, jobs and transport" [3]. The assertion was made regarding the 7th Global Goal of the 17 Sustainable Development Goals, a series of objectives set by the United Nations General Assembly in 2015 to be achieved globally until 2030 [4]. In particular, SDG 7 aims to "ensure access to affordable, reliable, sustainable and modern energy for all", and is tracked by 5 targets and 6 indicators. Among the targets, nations are called to double the global rate of improvement in energy efficiency, to promote access, technology and investments in clean energy, and to increase the percentage of renewable energy in the global energy mix.

Electricity is a secondary energy source. Despite not being extracted directly from nature, it quickly

	Total Residential /Households	Space heating	Space cooling	Water heating	Cooking	Lighting and appliances
Electricity	24.7	3.4	0.4	3.0	3.0	14.1
Derived Heat	8.3	6.4	0.0	1.9	0.0	0.0
Gas	32.1	24.2	0.0	6.1	1.9	0.0
Solid Fuels	2.9	2.6	0.0	0.2	0.0	0.0
Oil & Petroleum Products	11.8	9.2	0.0	1.7	0.8	0.0
Renewables and Wastes	20.1	17.7	0.0	1.9	0.3	0.0
Total	**100.0**	**63.6**	**0.4**	**14.8**	**6.1**	**14.1**

Figure 1.1: Share of fuels in the final energy consumption in the residential sector by type of end-use, EU, 2019. Source: Eurostat.

became an essential aspect of modern life. Developed countries have complex networks, able to provide electricity to a large majority of residents. These electrical grids must maintain enough firm capacity to match demand at all times. However, demand is variable: high-demand periods require higher capacity, often met with expensive peaking plants or a higher base capacity. The fluctuation of demand over time - i.e., the demand curve - forces energy companies to maintain a surplus of energy supply, create consumption forecast models, and use additional infrastructure to sustain the grid [5]. The issue deepens as mobility solutions powered by electricity become the norm. Battery Electric Vehicles (BEV) and Plug-in Hybrid Electric Vehicles (PHEV) draw a significant amount of power and are often charged at similar periods (e.g., at night or early morning). In a future where residential areas and building garages include numerous charging stations, accommodating simultaneous charging of BEV will not only result in additional imbalance of demand, but may even become unsustainable for the existing infrastructure.

Moreover, many regions have aging grid infrastructure that does not support bidirectional electrical flow [6]. New homes are being built with solar panels and Battery Storage Systems (BSS) to improve energy efficiency and utilize more renewable energy, yet, in a traditional grid, the energy produced locally must be used for self-consumption. Overproduction cannot be returned to the network, leading to penalties for end-users and waste of energy. Traditional grids are being modernized with bidirectional flow of energy and communication signals, incorporating sensors and smart meters, thus becoming smarter [7]. Smart grids enable consumers to participate in managing the grid, through Demand Response programs and utilization of excess production.

From data on household consumption published by the Eurostat in 2019 [8] and shown in figure 1.1, space heating represents 63.9% of energy end-uses, followed by 14.8% consumed in water heating, 14.1% in lighting and appliances and 6.1% in cooking. Slicing the household consumption by source fuel, electricity represents 24.7% of the energy mix. Odysee, the European energy efficiency project, shows the share of electricity is increasing rapidly, coming from 21% in 2000 [9]. When accounting exclusively for electricity consumption, lighting and appliances represent most end-uses, as most of the heating in the European Union uses gas and renewables.

The aforementioned factors - drive for energy efficiency, management of self-production, and relevance of appliances in household consumption - expose the need for tools that optimize energy consumption and coordinate demand from a bottom-up approach, compatible with traditional and modern grids.

1.2 Objectives and Deliverables

The goal is to develop an Energy Management System able to improve demand inefficiencies, both within residential households and local grids, with minimal impact to the consumer. The system is a server application that can control the execution time of electrical appliances to perform at a more adequate period, preventing switchboard trips and local shortages, meant to be installed in a computer within the household. The system is interactive, receiving execution requests as they come and distributing them using present-time information. In shared residential spaces, such as an apartment building or condominium, users are incentivized to distribute their energy loads not only to benefit from a lower contracted power, but also to reduce shared electricity bills, as the electrical installation requires less power.

The solution should account for most types of appliances, BEV, and include support for alternative energy sources such as Photovoltaic Systems and Battery Storage Systems. User preferences must be considered in order to reduce disruption. Energy loads require accurate parameterization for the system to be as effective as possible. The system should be able to simulate the residential environment, including characterization of devices, programs and executions.

To complement the program, a simple user interface is to be developed along with test consumption scenarios. Results are produced from these scenarios and evaluated using consumption indicators such as peak, peak-to-average ratio and total delay time, when compared against an unmanaged system.

1.3 Thesis Outline

The following chapter - Background - provides an overview on statistics, strategies and products that can be used to shape demand and improve energy efficiency. Chapter 3 (Architecture) is focused on the concepts behind the proposed solution, exploring the structure of the system and benefits to stakeholders. Chapter 4 (Implementation) examines the technical aspects of the system, also elaborating on development decisions and limitations. Testing data, simulations and results are exhibited and commented on in chapter 5, Results. Finally, chapter 6 (Conclusions) closes the document with additional remarks on the work accomplished, as well as improvements to be developed in future iterations.

Chapter 2

Background

Chapter 2 covers literature regarding previous work on energy management, including consumption behaviors, energy management techniques, smart grid features and previous approaches to improve energy efficiency. Processes for shaping the energy consumption are explained in section 2.2 and revisited in later chapters, as they are integrated in energy management solutions both from demand and supply sides.

2.1 Household consumption

Domestic consumption data is crucial for identifying patterns in demand, as well as paths to optimize distribution. As mentioned in chapter 1, electrical consumption in households is dominated by lighting and appliances (57%) which do not fit in other categories such as space heating (14%), water heating (12%) and cooking (12%) [8]. This section covers more energy consumption data, influential factors and profiling of appliances.

2.1.1 Factors for demand

Space heating is used to maintain indoor temperatures at acceptable levels for human thermal comfort. Naturally, more energy is required to do so when outside temperature is lower. Geographic location affects solar irradiance and plays a major role, both in heating necessities and broader lifestyle choices that end up shaping consumption. Figure 2.1 compares household consumption by end-use between EU countries. Countries from southern Europe and the Mediterranean sea, where the climate is more moderate, such as Malta, Portugal and Cyprus, not only display a lower percentage of energy used in space heating, but also rank among the countries with lower overall household energy consumption.

Other factors include the number of residents in the household, seasonality, intraday temperature fluctuation and periods of absence. In the northern hemisphere, average temperatures are higher in spring and summer, and lower in autumn and winter. In cold weather, less heat generation is required during direct sunlight hours, contrasting with periods at night and early morning, where demand increases. Outside temperature affects other forms of heating as well, as more energy is required to

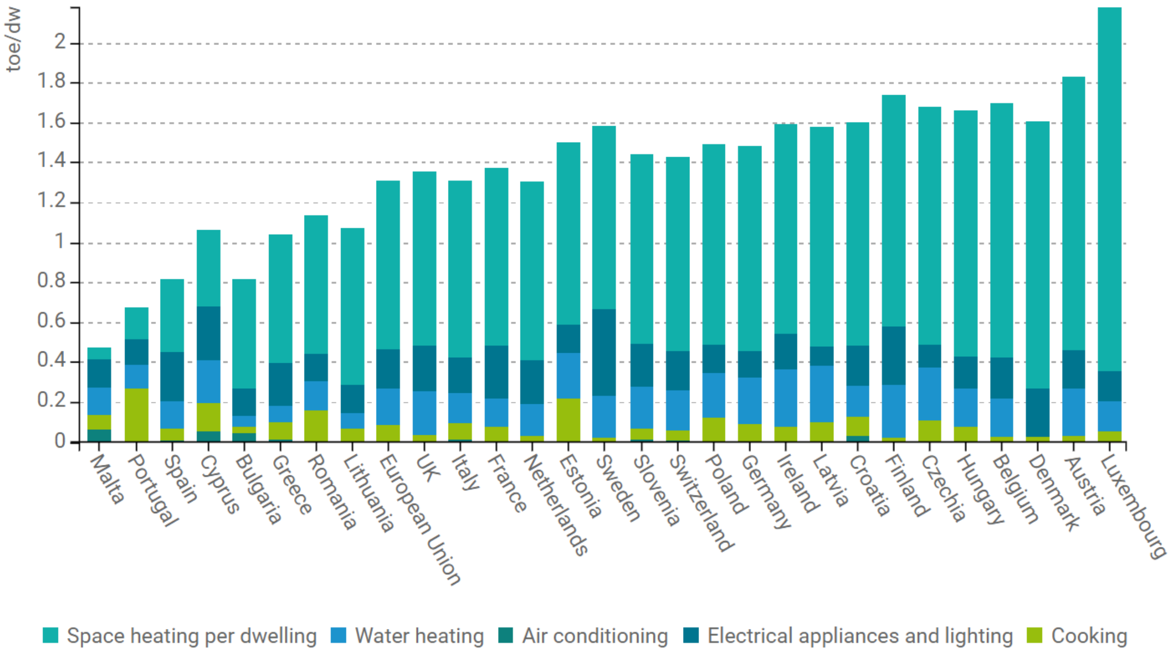

Figure 2.1: Specific consumption of households by end-use. Source: Odysee.

achieve certain temperatures, meaning that over 38% of energy consumption has a seasonal variation.

Concerning the intraday variation on energy consumption, temperature is only one of the many factors that influence expenditure [10]. A significant portion of home owners leave for work in the morning and come back at late afternoon. Cooking is prevailing before launch and dinner time. Lighting is more necessary at night. Additionally, workers commonly use their free time at early night for leisure activities or house chores. These behavior patterns are associated with the use of specific appliances: meal preparation may require the oven, toaster, stove or microwave, house chores include the iron, dishwasher and washing machine. Most of these are large appliances that consume a significant amount of energy. Intraday variation, when plotted in a graph, is called the demand curve.

In Portugal, electricity is the main energy source for households [11]. Heating is predominantly electric and activated during winter season. Air conditioners are utilized in the summer to combat the warmer temperatures. Figure 2.2 shows four different demand curves, representative of each of the seasons. It can be observed that consumption is heavier in winter and peaks at early night, when temperatures drop. A lower high can be observed in late morning, especially in spring and autumn, matching with launch preparation time. In the summer, air conditioners are used during the afternoon, smoothing the transition to night-time activities. Thus, the night peak is less evident than in other seasons.

2.1.2 Consumption in shared residences

In 2014, multi-family dwellings represented 49.5% of all housing in the European Union [12]. In a shared residential building, consumption is an aggregation of the values in its housing units plus common property (CP): garages, parking lots, service rooms, pools, etc. This infrastructure may in-

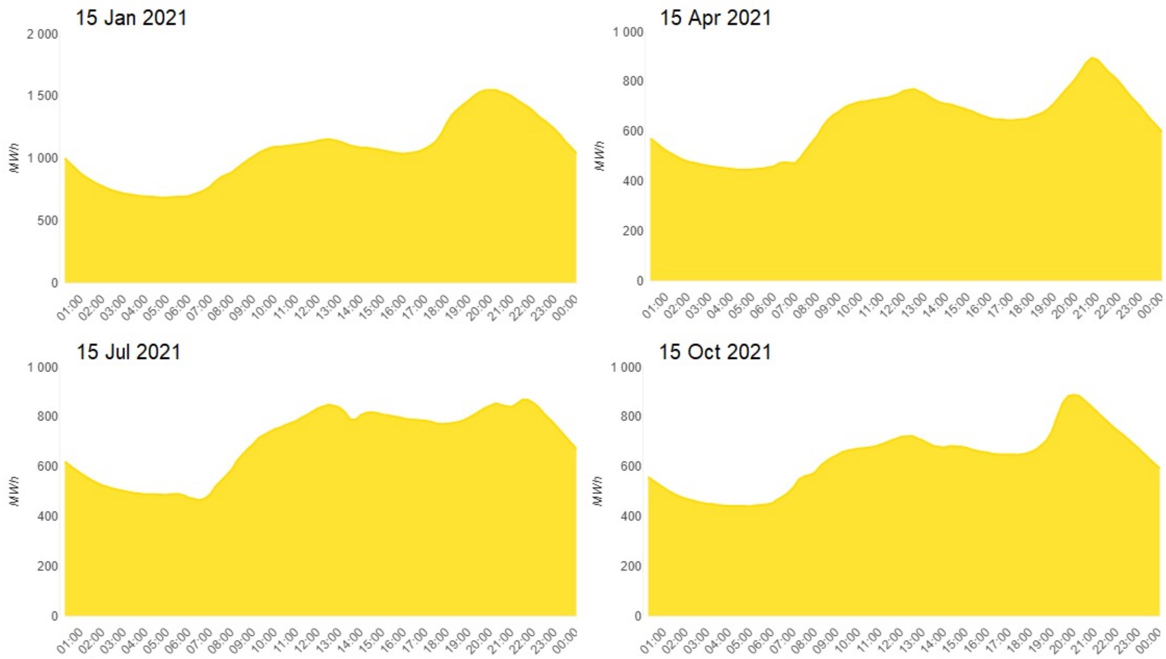

Figure 2.2: Portuguese low-tension energy consumption in four different days. Source: E-Redes.

clude energy-intensive services, such as BEV/PHEV charging in garages, laundry appliances in shared laundry rooms, and vertical transportation to different floors. Energy production and storage may also happen at building level. Single housing units in shared residences often do not have the space or capacity to install these systems. Installing PV systems in shared rooftops and parks is a more efficient alternative, able to accumulate much more energy and serve multiple properties.

The portion of total energy spent in common property also depends on the shape, thermal isolation, surface and rooftop areas of the building, as much is utilized in pumps and HVAC. Japanese high-rise buildings reportedly consume 10% of the total load in CP, while in Australia results ranged from 36% to 57% [13, 14]. Studies have shown that demand aggregation in common property can benefit consumers tremendously, with the potential to reduce the contracted power of the dwelling and satisfy the common demand with clean energy [15].

2.1.3 Appliance information

Appliances come with information regarding power ratings, energy efficiency and capacity (if any). Rated power is the power needed for execution, even if the consumption is not constant; it represents the maximum power draw of the device. As such, it can be used as an upper bound to represent the consumption of a machine, e.g., in estimating the ideal capacity for a PV system installation. Some appliances have different consumption profiles, depending on the task. The potency of a microwave is adjustable, between maximum power for fast heating or the lowest mode for defrosting. Even when turned off, many appliances enter a ”standby” mode, reducing consumption to a minimum to power digital clocks, smart hub connection, etc. Different modes often have a different rated power. The energy efficiency label, enforced by EU regulation since 1992, offers a more accurate comparison between machines of the same type. Appliances are labeled from A to G, where A is the highest efficiency class and G the lowest. Consumers should pursue machines with higher efficiency in order to save on energy consumption.

Appliances are generally categorized regarding their mobility and size. Non-portable or semi-portable appliances are called large appliances, such as the fridge and the washing machine, while lightweight appliances that generally rest on table or counter-tops (e.g., toaster, coffee maker) are designated small appliances. Large appliances usually consume more energy and, until the last decade, represented most of consumption in households [9].

A more interesting categorization from an energy management standpoint is according to the load profile. Resistance-based appliances use resistors to turn electricity into heat. Toasters, heaters, ovens, and other devices focused on producing heat work this way. Incandescent light bulbs also use resistors, heating a filament until it produces light. These appliances use constant power: the consumption curve of these devices is steady until interrupted by a timer or the user. As such, the rated power is an adequate estimate of real consumption. Motor-based appliances such as fans also fit this category.

Some appliances have simple ON/OFF self-regulation, like the fridge and some heaters. Self-regulating heaters, despite also being resistance-based, have a thermostat to stop producing heat until temperature falls again below a threshold. The fridge alternates between ON and OFF on a time basis, with additional power used when the door is opened or during defrost cycles. Air conditioners and water heaters also activate periodically, in order to retain temperature. The fan of an air conditioner operates in separate from the cooling mechanism. Consumption of these devices can be estimated adequately using equation models that parameterize internal and external factors [16].

Other appliances, such as the washing machine, dishwasher and clothes washing, have more complex self-regulating mechanisms, with the power consumption varying according to the operating cycle [17]. A full cycle of the dishwasher includes washing, rinsing and drying. A heating element is used at the beginning of the washing and rinsing phases to heat the water to a set temperature, then the consumption lowers until the end of the stage. The washing phase has many interruptions, and may be preceded by a pre-washing routine. Stages of the washing cycle are hard to identify without a power meter, and vary according to brand and model. Thus, accurate modelling of the consumption curve of these appliances requires specific data from the machine, not accessible to the average consumer.

2.2 Peak demand reduction terminology

The following processes shape the demand curve with focus on reduction of peaks. They are used and often combined in energy management solutions, or encouraged by energy suppliers through variable pricing.

Load Distribution is a strategy focused on evenly distributing energy loads across time, without affecting the overall energy consumed. The amplitude of the demand curve is reduced, as valleys are filled with loads from peaks. This strategy can be applied by shifting energy loads from peak to off-peak periods. In some literature, it is decomposed into peak shifting and valley filling.

Peak Shaving is the elimination of consumption peaks whenever they would be formed. It encompasses load denying - rejecting new loads after a certain consumption threshold - and the use of secondary energy sources during peak demand, such as solar panels or batteries. The consumption peaks become plateaus with lower maximum value.

Load Reduction consists in reducing the overall consumption. The shape of the demand curve remains similar to the original, but with lower values. Utilizing more energy-efficient appliances falls into this category.

2.3 Supply-side techniques

Energy suppliers already implement a range of techniques to shape the demand curve of consumers, both in traditional and smart grids. The most basic control of demand comes from the maximum available capacity chosen by the consumer - the contracted power. Consumers cannot exceed this value; if the instant consumption exceeds the contracted power, the switchboard will trip. The capacity is measured in kVA, and a higher contracted power results in higher energy cost per kWh. Consumers are encouraged to contract power above their average consumption, in order to remain below the threshold during periods of higher activity.

Through financial incentives, contracts may also motivate consumers to adapt their consumption behavior to a more favorable pattern for energy distribution. This is called demand-side management (DSM). Customers can enroll in static or dynamic rates. Static pricing is the most common practice. Consumers with static pricing contracts can opt between flat and Time-of-Use (TOU) models. The flat rate can be considered the baseline for comparison with all other pricing models: electricity costs the same regardless of time of day, meaning that there are no incentives to consume more or less energy in certain periods. In TOU, the cost depends on the month and period of the day. These models break the day in time periods, according to expected demand. The cost is static and respective for each of the periods, established in the contract.

In Portugal, TOU offerings consist of bi-hourly and tri-hourly tariffs. A bi-hourly contract partitions the day in blocks of empty and non-empty hours. Empty hours are periods where the average consumption is low for the time of the year, with non-empty hours composing the rest of the timetable. Thus, energy is made available at cheaper prices during empty hours. The tri-hourly tariff adds a third category, peak hours, for short periods where the grid needs to provide full capacity. Empty hours in the latter two models are evidently cheaper than the flat rate, but non-empty hours are often more expensive than the baseline, encouraging consumers to perform selective load distribution if they actually want to save on costs, rather than simply getting a discount in low demand hours. Peak hours are a deterrent to consumption during those times, effectively acting as a peak shrinking mechanism.

Through new grid monitoring practices available in smart grids, distributors can measure consumption more accurately and within smaller timeframes. Consumers report their consumption to the distribution company automatically and in short intervals, with the aid of smart meters installed in their households. Active monitoring creates an opportunity to predict consumption with tighter ranges and create price models more adjusted to real demand - paving the way for dynamic pricing. Through dynamic pricing, the cost passed down to the consumer reflects the production and distribution costs more closely, with some added volatility when compared to static models.

Consumers can enroll in Demand Response programs and actively contribute to the stability to the grid. Real-Time Pricing is a model where the electricity cost fluctuates over short intervals, and the consumer is charged with varying prices depending on the time period. The cost may vary according to current demand, or from day-ahead market prices; in the latter case, households may know the costs in advance and plan their consumption accordingly. Critical Peak Pricing compliments flat and TOU models by significantly increasing the cost of electricity during select critical peak events, announced to consumers a day ahead. In exchange, the rate for other time periods is slightly discounted. Other models based on rebates or coupons also reward consumers for reducing consumption during peak hours.

2.4 Monitoring services

Energy distribution companies are promoting the installation of smart meters in consumer households, creating a communication interface between home and energy provider that opens the door for more efficient grid management. However, most smart meters installed by companies do not disclose real-time data to the user, serving exclusively as a secure platform to communicate readings. In Portugal, clients with smart meters only get access to the delayed consumption history for the last 30 days, with no possibility to deconstruct the data. To obtain plug or whole-house energy monitoring, other commercial or do-it-yourself solutions must be installed.

Advanced monitoring solutions frequently provide real-time consumption data for the entire household, including separation by divisions or specific appliances, accessible to the user through web and mobile applications. As an example, Sense [18] uses clamp-on current sensors to collect consumption and production data. Devices are identified by their energy usage signature, so that the periods of use and power consumption can be monitored individually. A mobile application displays the total consumption of the household, consumption of identified devices in a timeline, which devices are on, and the predicted energy cost for the month using a flat rate. Users can also set consumption goals and receive notifications for abnormal consumption of specific devices. The hardware costs 349$ and installation by an electrician is recommended. It does not support variable-rate energy plans, and device identification is dependent on correct energy signature recognition.

Studies in Korea and Australia show benefits of up to 10% in energy savings using in-house displays [19, 20]. Householders were able to improve their consumption habits by being more conscious about their energy waste and regulating the devices themselves, using information provided by an energy monitoring system.

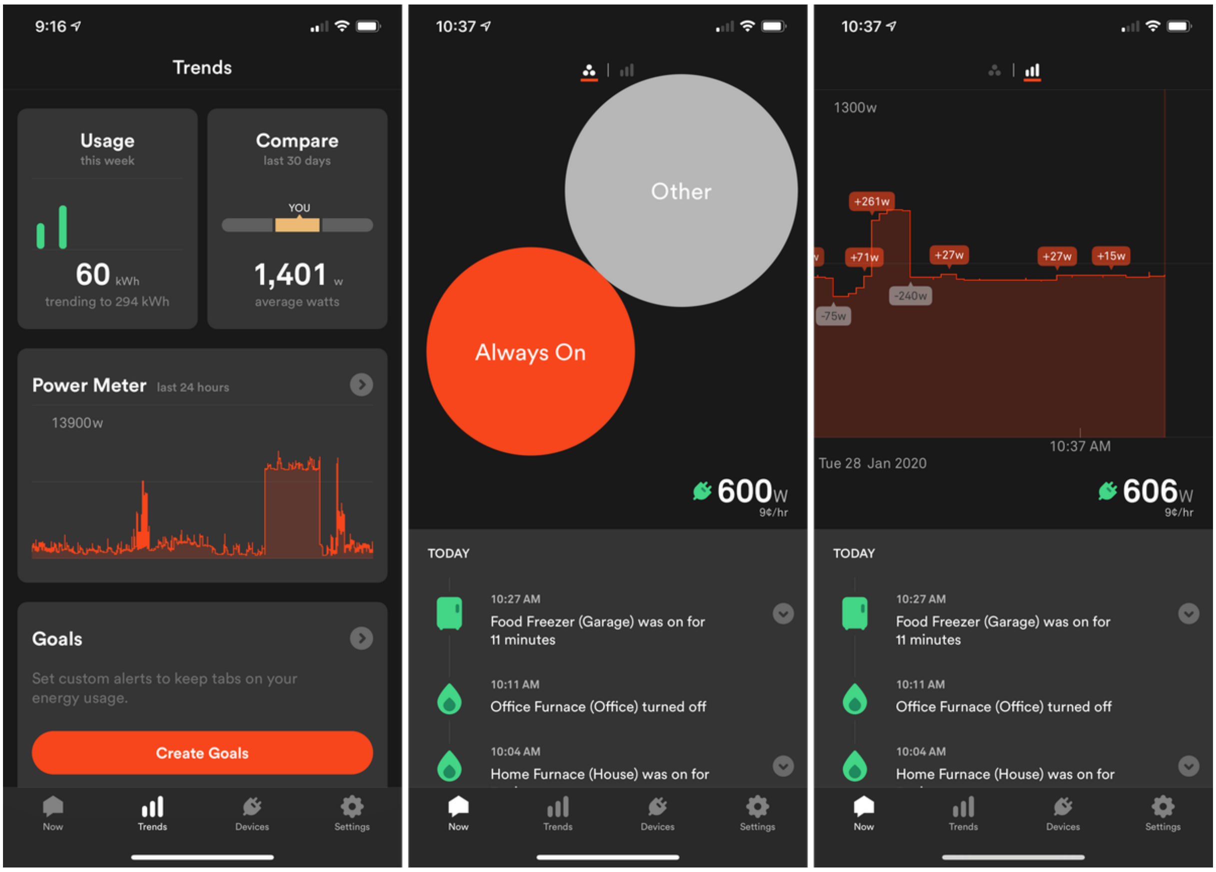

Figure 2.3: Screenshots of the Sense application. Source: ZDNet.

2.5 Active energy management

Home Energy Management Systems (HEMS) provide additional control to consumers, offering remote control over home appliances and automation to varying degrees. Smart plugs are used to control the state of non-smart appliances, where as smart machines can communicate directly with a network. Communication is standardized through protocols such as ZigBee and MQTT [21, 22]. Devices communicate with a coordinator to inform their status or receive new instructions. Depending on the appliance and the level of control over its functionality, the status can range from ON/OFF to more fine-grained information such as temperature, intensity and water usage. A web or mobile application can communicate with the network to display the statuses and provide control to the user.

The Portuguese company EDP Comercial launched EDP Re:dy in 2014 [23], a hardware kit that includes an array of smart plugs (for general purpose, controlling A/C or monitoring solar production), switches for appliances that do not connect to plugs, a smart meter and a box that coordinates the devices. The solution uses ZigBee and PLC as communication protocols. The hardware is complemented with both a mobile application and a website. Re:dy promises savings on energy despite having a monthly subscription price; besides monitoring and controlling devices, it also offers on/off scheduling, keeps consumption statistics to inform of unexpected device activity, predicts monthly consumption and recommends lowering the contracted power, according to peak consumption in the last months.

The hardware kit cost ranges between 102€ and 157€, depending on the plugs, and has a monthly subscription price of 5€ for households with less than four solar panels.

In some HEMS, light, temperature and movement sensors can be incorporated to trigger actions, such as turning on a heating system or the lights in a corridor. Other conditions may also be set as triggers. Time-based automation enables the scheduling of appliances to match user routines or occur in absence of the user. It is possible to schedule tasks to off-peak hours, improving grid efficiency. In particular, the American company DTE Energy launched the SmartCurrents program [24], attracting consumers to a dynamic tariff and offering a smart thermostat. The thermostat is WiFi-enabled and aware of demand periods. Thus, it is able to regulate the household temperature by operating at periods where the energy costs are lower.

2.6 Scheduling solutions

At their core, the systems mentioned above are tools to reduce energy waste. In order to effectively optimize the demand profile of a house, solutions require more control over consumption of appliances. A system that can decide or recommend the time of execution for each appliance is called a scheduling system. These solutions attempt to formulate consumption as a mathematical problem and develop algorithms.

Mohsenian-Rad et al. (2010) [25] approach the issue by formulating two optimization problems: peak-to-average ratio minimization and energy cost minimization. Appliances are categorized between shiftable and non-shiftable. Shiftable loads have a time period limiting when they can be scheduled, equal or greater than the desired time of consumption to fulfill its purpose, which is predetermined. Households are aggregated and the scheduling follows a distributed algorithm, to maintain an equilibrium in demand between consumers powered by the same local energy network. While energy cost to the consumer can be minimized at a household level in variable pricing programs, grid efficiency depends directly on total consumption over time. Each household has a Energy Consumption Scheduler that solves its own part of the algorithm, obfuscating the actual device scheduling to other households for privacy reasons. This strategy, while theoretically effective and privacy-concerned, lacks flexibility due to the assumption that appliances are used during a fixed amount of time every day, with no variation in consumption. Device categorization disregards nuances in interruptible loads.

Chen et al. (2013) [26] consider uncertainty in household consumption, as a variable in the three-stage proposed algorithm. A day-ahead pricing program is assumed for calculations on single households. Consumption of each device is defined probabilistically, following a Gaussian distribution. Self-production is also accounted for in this solution: solar energy, although representing a small portion of the energy consumed, should be prioritized over grid consumption. As a single-house solution, it may fail to benefit the interests of energy distribution companies and the equilibrium between demand and supply, as there is no communication between instances to coordinate the loads moved to empty hours. In a grid where all nodes use this system, peaks would certainly form in lower priced hours. Besides, the uncertainty considered in the algorithm is more geared towards the variation in consumption of a device, due to the duration of use or other usage factors, but does not account for utilization (or the lack of it) outside the schedule prepared in advance. Thus, it fails to consider spontaneous changes in the timing of household activities.

Ozturk et al. (2013) [27] propose a DR scheduling system for customers enrolled in TOU pricing programs. Using a learning algorithm, the solution can learn from usage patterns to offer load forecasting and adapt the appliance scheduling to customer preferences. It has a peak shrinking component in addition, by managing heating, ventilation and air conditioning (HVAC) more efficiently and shifting heavier loads to periods where solar power is available. The system can aggregate household consumption in residential areas and provide data to energy companies, which in turn can better adjust the available resources and update TOU prices. Appliance utilization is perceived as ”a list of tasks to be scheduled at preferred time intervals”, once again focusing on user comfort. Consumers interact with the system by manifesting their intention to use an appliance within a certain time period. The scheduler either turns on the device or informs the user that delaying its activation will result in cost savings. The user is then prompted to accept the recommendation or turn on the appliance immediately. Beyond the cost savings and efficiency gains, consumers also benefit from more information and consumption awareness.

Zhou et al. (2016) [28] elaborate on device categorization. Shiftable appliances can be further categorized between interruptible and non-interruptible. Interruptible loads offer the advantage of having their activity split into different time intervals, which can improve scheduling strategies. HVAC and PHEV are notable shiftable interruptible machines, able to run in periods of excessive supply and suspend when other loads need to execute.

Chen et al. (2014) [29] introduce a two-layer control architecture. In the lower layer, an energy management controller communicates with appliances at household level and locally schedules the loads. The upper layer consists of communication between households to apply a distributed consensus algorithm, in order to match the desired demand profile of energy distributors. The consensus decisions from the upper layer define the load limit to the lower layer scheduler. The desired demand profile is usually known in advance, allowing the upper layer to decide in advance how the consumption of households can be shaped, while the local scheduler can act in real time with the limits already set.

Group optimization has been further explored in the context of shared apartment complexes or building districts. Pinto et al. (2022) [30] delivers a HEMS able to manage multiple buildings with heterogeneous consumption, equipped with thermal energy storage and PV systems. Two architecture models - cooperative and coordinated - are compared in performance against uncoordinated rule-based controllers, using Key Performance Indicators (KPI) such as the peak, peak-to-average ratio, self-sufficiency and flexibility factor[1]. The results showed improvements across all KPI with both architectures, with the coordinated approach - a centralized model similar to the proposed solution in this document - achieving reductions of the consumption peak by 9.7% and the peak-to-average ratio by 7.7%.

[1]The flexibility factor is defined as the ratio between off-peak imported electricity consumption and total imported electricity consumption. Not a metric of flexibility from the user perspective.

2.7 Conclusion

Domotic systems offer a wide range of monitoring and actuation tools to consumers. Most of the commercialized systems are focused on providing information and control to the user, with intricate interfaces that detail consumption, generate statistics and present remote control. Solutions focused on autonomous or semi-autonomous management of devices are mostly academic. Researchers have developed algorithms that control the demand curve of households by scheduling the use of appliances in advance, with varying degrees of device categorization. These systems typically offer benefits to consumers enrolled in DR programs with TOU or dynamic rates. Due to the scheduling happening in advance, these solutions face difficulties adjusting to impulsive or seasonal changes in consumer behavior. Additionally, contracted power is a considerable factor in electricity prices regardless of the rate model, and lacks representation in the aforementioned research.

Chapter 3

Architecture

The proposed solution is an autonomous HEMS that schedules household appliances with a focus on peak shifting. Peak shifting is accomplished by limiting consumption to a threshold defined by the user. If the threshold is surpassed, energy loads are delayed or shifted to an available period. The system is able to start, shift, delay and interrupt appliances interactively, using a priority system based on user preference to decide as new scheduling requests are made. Scheduling in real time using dynamic priorities increases flexibility to the user, enabling sudden consumption changes from unexpected behavior and quickly adapting to them. BSS and PV systems are integrated as peak shaving and valley filling mechanisms. The solution incorporates a minimal interface and is able to coordinate loads between residential units in shared dwelling complexes.

Chapter 3 discusses the stakeholders for the system, presenting the parties and discussing the benefits and drawbacks for each. Moreover, defines the scope of appliances covered by the system, all the system components and its interactions, both within the system and with the outside world.

3.1 Stakeholders

There are three stakeholders that theoretically stand to gain with such system: **residential consumers**, **property managers** and **energy providers**. This section analyses the mission, behavior and goals of each stakeholder, concluding with potential benefits from adopting a new solution.

The core business of energy providers is to buy energy in wholesale markets and resell it to retail customers. They are responsible for maintaining the grid infrastructure operational and able to deliver the energy to the end-user. In general, distribution companies profit from more demand, as there is a profit margin taken from redistribution. However, as energy is purchased in advance, demand needs to be predicted. Energy companies aim to buy enough energy to accommodate the demand of consumers, with a surplus for unexpected peaks. A traditional grid does not store energy. Thus, energy that is not consumed goes to waste, causing a loss for the company.

Peaking power plants, used to respond to demand when the supply from base stations is insufficient, are more expensive to operate. The additional cost is transferred to the distribution companies. Therefore, companies would benefit from a more balanced demand, not only to prevent underutilization of the grid during low demand periods, but also to avoid buying energy from more expensive sources.

Consumers are billed for their consumption, usually monthly. Pricing models per contracted power and per kWh were discussed in section 2.3. Consumers benefit from reducing their contracted power and overall consumption. Blackouts are a significant inconvenience for residents, especially in the night.

Property managers are responsible for maintaining the shared private infrastructure in condominiums or apartment buildings. The entity may just exist implicitly, if there is infrastructure with expenses split between residents. The property manager ensures the quality of shared space and charges a periodic fee to cover costs. Depending on the services that are offered, energy costs may represent a significant part of management expenses: the Association of Condominium Owners of Ontario claims that typical condominium corporations spend 35–50% of their annual operating budget on utilities [31]. Managing the consumption within shared spaces to flatten the demand curve may result in a cheaper electricity plan. Additionally, the whole multi-family complex shares grid resources that deliver energy to each residence. The energy provider is responsible for ensuring the grid structure is sufficient to power the building or residential complex, but peak prevention at this level can create a new relationship of incentives between company and residential entities, similar to buildings-to-grid relations present in the commercial building sector.

In countries such as Japan, it is common for a closed multi-family building to have a single contract with the energy provider. The power received at high voltage is then distributed locally to the apartment units. These shared residential complexes stand to gain the most with a solution that distributes energy loads in coordination with other individual dwellings, maximizing the potential reduction in contracted power.

The scheduling system that is proposed can operate in **single-house** and **multi-house** modes. In single-house mode, loads are distributed locally with focus on reducing delay inconveniences, preventing energy shortages and respecting the consumption limit imposed by the user. This mode directly benefits consumers. Energy companies benefit indirectly from a flatter curve in multiple households, although there is no coordination between instances. The property manager can run its own unit to manage machines in shared spaces, behaving as an individual consumer and reaping the same benefits.

In multi-house mode, consumers provide their total scheduled consumption to a central unit - represented by the property manager - that aggregates demand and provides scheduling recommendations to individual units, based on the loads already scheduled. Large appliances and EV charging are scheduled with awareness of peaks forming in the complex grid, getting distributed more evenly across time. For privacy reasons, aggregate data is stored pseudo-anonymously and temporarily. Moreover, no individual device data is shared outside the local unit. Even so, the central unit could be accessed maliciously and expose consumption of individuals, by gathering enough consumption data to match it with the house. To the consumer, the multi-house mode adds a privacy risk. Executions may also be further delayed to support the grid, although the delay can never surpass the limit set by the user.

However, the benefits to the grid should be much more significant. Evidently, the energy provider is directly benefited by the system, as the aggregate demand curve is softened, and thus less reliant on energy from peaking power plants. In countries where shared energy contracts exist, property manager and consumers gain from the system together, with an overall lower energy expense due to a reduced contracted power. In shared European residences, the additional immediate reward in multi-house mode is the prevention of energy outages due to excess consumption at building/condominium level. Nonetheless, the increased stability provided to the grid has a positive environmental impact to all stakeholders, which could be rewarded by the energy provider or state programs, in a similar manner to existing financial incentives discussed in section 2.3.

3.2 Application components

The system has two application components, the **Coordinator** and the **Aggregator**. A high-level representation of interactions between components, interfaces and external devices is represented in figure 3.1.

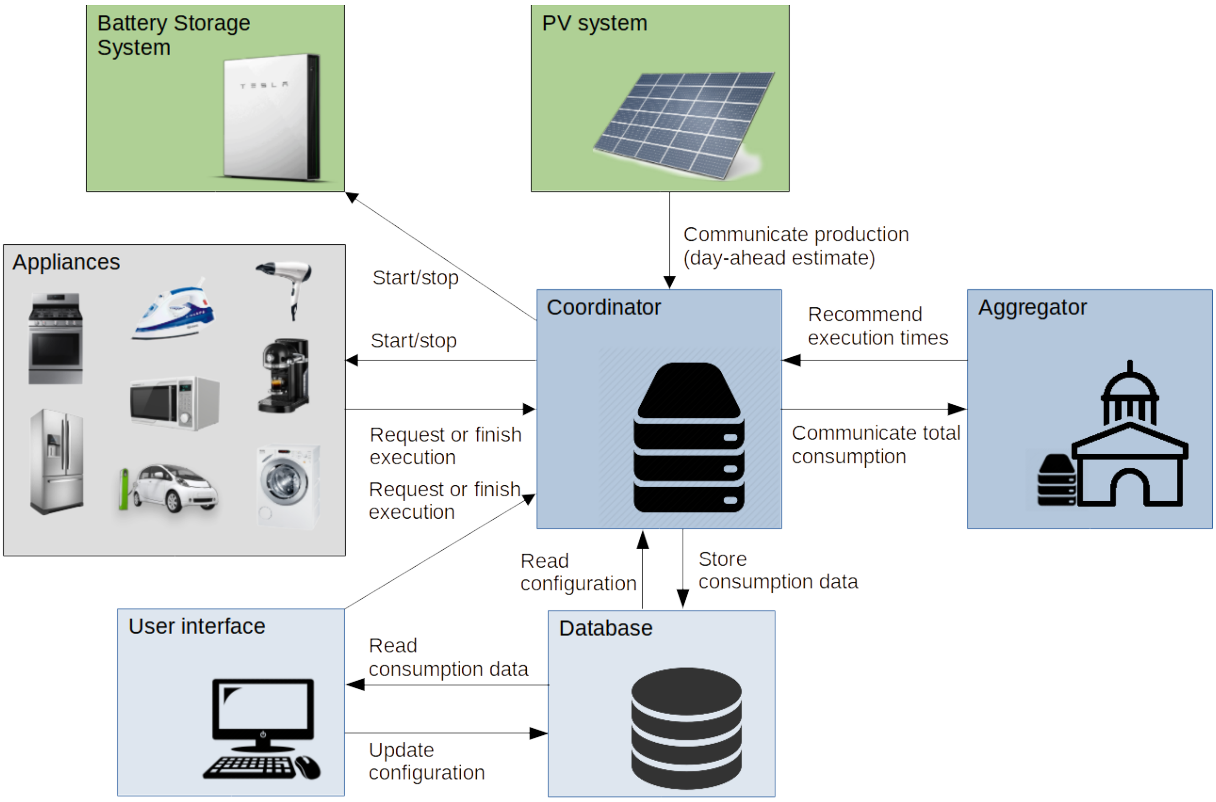

Figure 3.1: System architecture diagram.

The Coordinator fully manages the production and consumption of an individual household when the system is disconnected from an Aggregator - operating in single-house mode. All appliances, alternative energy systems, executions and user parameters are stored in this component. It is expected to be operating inside the household, communicating with a Smart Home hub to interact with the devices. The Coordinator receives and handles requests from appliances to start or stop activity. Moreover, it is able to connect to an Aggregator, provide the necessary data for the latter to store an aggregated demand curve and eventually request recommendations for execution times. If BSS exists in the household, the Coordinator always manages the charge and discharge cycles locally, based on individual household consumption.

The Aggregator receives connections from Coordinators that decide to operate in multi-house mode. It stores pseudo-anonymous consumption data in bulk and responds to scheduling recommendation requests. The Aggregator is to be installed outside of any housing unit, but represents the shared interests of the residential complex; as such, it can be maintained by the property manager. Data regarding individual appliances is not disclosed to the Aggregator. Instead, the Aggregator is only informed of the time and power needed for a new schedule.

3.3 Appliances

An appliance, by definition, is an instrument or device designed for a particular use or function. When looking at appliances in a household, it may be intuitive that large devices such as the fridge, washing machine or the oven are included, as residential appliances typically assist in cooking, cleaning and food preservation. However, portable appliances and consumer electronics that may or may not be used at home also affect residential consumption. Consumer electronics provide entertainment, communications and recreational activities, influence the indoor and outdoor lives of consumers, and mostly require electricity (directly or stored as chemical energy in batteries) for normal operations. BEVs are one of the top energy consuming devices connected to the household, despite being used outside. As such, the scope of appliances to be managed by the system includes all devices that are powered or charged by a home electrical system, intended for personal use. Devices customarily charged at home are included, regardless of usage location.

The system is able to stop and postpone the execution of devices, if a new execution request arrives with a higher priority. However, not all appliances can be interrupted and resume their activity, either due to loss of progress (e.g. in heating systems, as the heat dissipates) or inability to resume at the previous state. As such, appliances are classified between **interruptible** and **non-interruptible**. An interruptible machine can resume its progress after being interrupted by the system. Therefore, appliances of this category add more flexibility to the scheduler, enabling more urgent requests to commence immediately, favoring user satisfaction. A non-interruptible device will not be considered when looking at shiftable appliances to accommodate new requests, as stopping it could cause unpredictable behavior.

3.4 Priority function

Under heavy loads, executions are started or delayed according to a monotonic priority function. The function is used to attribute priority values to executions dynamically. As the scheduler receives a new instruction to schedule or stop an appliance, priorities of devices are recalculated to ensure that the most time-sensitive executions have the shortest delay. A higher priority will lead to a quicker start, if no energy is available to start immediately without stopping other appliances, and supplemental energy sources are absent or unusable.

Part of the priority formula for an execution is based on the priority class defined for the energy profile (or program) of the appliance. Users can set a class between **urgent**, **normal** and **low-priority** for each different mode of operation of a machine:

- Urgent requests correspond to time-sensitive appliances, typically small appliances that require human intervention to be used, such as the hair dryer, the coffee machine and the cake mixer. These devices often operate in short spans. As such, this priority class can interrupt appliances with lower priority, to be resumed shortly after. Urgent requests are scheduled locally, ignoring load distribution-based preferences from the demand aggregating system component.

- Normal priority is given to appliances that provide benefits to the user during operation or require some form of active monitoring, but can start safely without supervision (e.g., the oven). Entertainment and recreation machines commonly fit the criteria. These devices are still time-sensitive, but the acceptable delay is often more flexible than in urgent profiles. Normal requests are, too, scheduled locally, with preference for the closest available time.

- Low priority is given to machines with little to no time restrictions, typically large appliances which just need to finish operation before a longer deadline. These devices do not require supervision and generally can operate overnight, such as the washing machine, the dishwasher, and the BEV charger. Low-priority devices are prime candidates for a load distribution scheduling strategy. In single-house mode, they will be allocated to periods with the lowest local consumption, while in multi-house mode, the Aggregator decides on the period with lowest aggregate consumption.

The other factor for the priority formula is the time since request, compared against the maximum acceptable delay, defined by the user. As the remaining acceptable delay approaches zero, the priority of the execution increases at a faster rate, potentially gaining higher priority than other executions in upper priority classes. This prevents new requests from further delaying an execution that is already delayed to the limit of acceptableness. The formula is as follows:

$$delay_{\text{request}} = \begin{cases} time_{\text{current}} - time_{\text{request}} & \text{if execution is unscheduled,} \\ time_{\text{start}} - time_{\text{request}} & \text{otherwise.} \end{cases} \tag{3.1a}$$

$$delay_{\text{remaining}} = delay_{\text{maximum}} - delay_{\text{request}} \tag{3.1b}$$

$$f = priority_{\text{base}} + floor(\frac{60 \times \alpha}{delay_{\text{remaining}} + 60}) \tag{3.1c}$$

$$priority = \begin{cases} f & \text{for } f \leq 10, \\ 10 & \text{for } f > 10. \end{cases} \tag{3.1d}$$

Where $delay_{\text{remaining}}$ is the value, in minutes, of the remaining acceptable delay, $priority_{\text{base}}$ is one of the values in table 3.1 and α is a constant used to modulate the steepness of the priority curve. By

Priority class	Base value
Urgent	7
Normal	4
Low-priority	1

Table 3.1: Base priority values.

Priority value	Urgent	Normal	Low-priority
10	[0; 100]	[0; 20]	
9	]100; 180]	]20; 36]	
8	]180; 420]	]36; 60]	[0; 8]
7	]420; ∞[	]60; 100]	]8; 20]
6		]100; 180]	]20; 36]
5		]180; 420]	]36; 60]
4		]420; ∞[	]60; 100]
3			]100; 180]
2			]180; 420]
1			]420; ∞[

Table 3.2: Priority values by category and remaining acceptable delay interval, in minutes.

flooring the fraction in equation 3.1c, priorities are discretized between 1 and 10. Discretization allows for a more predictable and coherent behavior. The value of $delay_{\mathrm{request}}$ is dynamic if the execution is not yet scheduled, as the current time keeps changing. However, it becomes static once a start time is attributed. Therefore, appliances do not change priority constantly and do not interrupt each other consecutively when trying to regain execution.

When a new request arrives, if there is not enough power available below the threshold using other procedures, the scheduler will compare the priority value of the new request with the static values of the executions already scheduled. If an execution is stopped manually, the priority of all delayed executions is recalculated in order to activate the appliances with highest priority at the moment of evaluation. The scheduling logic is elaborated in section 3.6.

The values 6, 8 and 10 were experimentally tested for α. With a value of 8, the curves adjusted better to round numbers that the user might input more often as maximum delay, such as twenty minutes and one hour, while keeping an adequate differentiation of priority values for the three categories as the acceptable delay approaches zero. For urgent energy loads, any delay of less than one hundred minutes gets maximum priority (10). Urgent executions generally have short acceptable delay periods, so a new urgent request will typically get maximum priority and start immediately, even without enough energy available, as long as interruptible executions are found. Normal executions have a wider range of priority values, starting from 4. This is the default priority. Appliances with normal priority and longer acceptable delay may not start immediately under heavy load, but reach the higher echelons at one hour remaining, and hit maximum priority at twenty minutes. Finally, low-priority devices are expected to never hit maximum priority. The priority class is intended for devices that will run whenever energy consumption is lower, filling valleys and possibly activating overnight. Table 3.2 shows the time intervals at which the three priority categories reach each value.

3.5 Interaction with devices

The system is able to simulate interactions with all the appliances within scope. It is assumed that, in a real-life integration scenario, appliances have a way to communicate with the system so that pressing the power button or changing the program during operation would send an execution request to the Coordinator, rather than starting execution immediately. Smart devices with electrical controls could be modified to operate within this logic, while mechanical appliances and plug-in chargers could be controlled using smart plugs. As examples, the following usage scenarios were considered:

- A washing machine is loaded with clothing. The user turns on the machine, and the machine will communicate with the Coordinator. As a low-priority device, it will be scheduled for a low-demand period using a load balancing strategy, local or aggregated. It will operate during the night without requiring further action from the user, and be finished by the morning.

- The household is warm on a sunny day. The resident wishes to use the air conditioner, a normal priority device, whenever there is enough power available. A new execution request is created with high priority, but not maximum. If enough energy is available, the air conditioner will start operating. It can be interrupted if new, urgent requests are made and no energy is available, then resume after the urgent execution finishes.

- The user wants to use the coffee machine. After turning on the device, it will ask the Coordinator for immediate activation. If the ongoing consumption is below the limit, the request will be conceded. If consumption is too close to the limit, an interruptible device will be stopped before activating the coffee machine. In a situation where no device can be stopped, the machine cannot start immediately and the user should consider increasing the limit.

Simulating a PV system in a scheduling system requires, at least, a day-ahead prediction of the energy that is captured and transformed into current over time. This data is simulated using System Advisor Model (SAM) [32], a free software developed by the U.S. Department of Energy to model solar panel installations, in tandem with typical meteorological year data from the European Commission's Photovoltaic Geographical Information System (PVGIS) [33]. SAM generates averaged hourly energy production estimates for each month of the year, imported to the scheduler database using a simple script.

Battery Storage Systems are also supported and fully controlled by the scheduler. Discharges are activated whenever consumption surpasses 70% of the threshold value, to soften the demand peak or, eventually, to allow the user to activate more appliances within the time frame of heavy demand. Charging is scheduled every night, broken in separate low-priority executions of one hour each. If the household has both a BSS and a PV system, the BSS will preferentially charge during underutilized solar hours. The system will attempt to prevent waste of solar energy, as long as the battery is not fully charged. If solar production is insufficient or nonexistent, the battery will charge whenever consumption is below 30%.

3.6 Scheduling process

When a new scheduling request arrives, the scheduler will preliminarily select three possible start times, according to the operations needed to find enough energy available. There are three possible scheduling procedures:

1. Simple scheduling: there is enough available power to schedule at the suggested time (including PV generation).
2. Battery-enabled scheduling: the BSS can provide complementary energy to enable the execution.
3. Priority scheduling: shiftable loads with lower priority can be interrupted to provide enough power for execution.

The start time obtained in each of these scenarios, as well as the procedure used to satisfy the request, are chosen based on the operation mode of the system and priority class of the appliance. In households without a BSS, the scheduling decisions are simplified to deciding between procedure 1 and 3. Conversely, if the household has a storage system, the start time obtained from procedure 3 will include available energy discharges, acting as a composite of procedures 2 and 3.

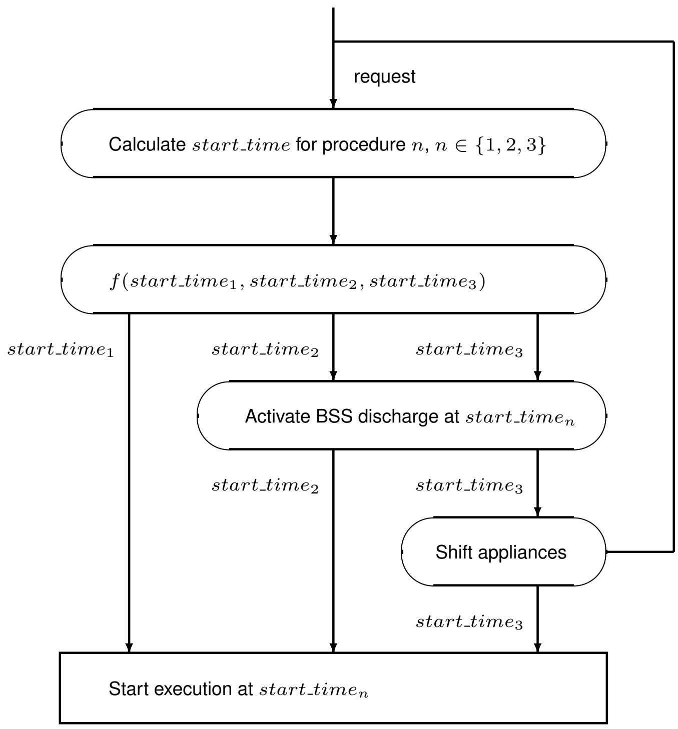

Figure 3.2: Algorithm for scheduling appliances.

For urgent and normal executions, the strategy is focused on **peak shifting**. The goal is to schedule as soon as possible while respecting the threshold. As such, the three scenarios are evaluated to find the simplest procedure that provides the soonest execution time. The suggested time for each scheduling procedure is the soonest where the conditions apply. In low consumption periods, all three will propose the current time: the execution starts immediately, without requiring a battery discharge or shifting appliances. During high consumption, the procedures may find different start times or even fail to find a viable execution period. If requesting a battery discharge anticipates the execution time over the simple scheduling option, or procedure 2 provides a viable execution time where as procedure 1 did not return a start time, the battery will be activated and provide complementary power. If shifting appliances provides an additional advantage in reducing the execution delay over procedure 2, or procedure 2 also failed to find an execution time, the battery will provide as much power as possible and the remaining power needed will be met by shifting appliances.

For low-priority executions, the scheduler aims to schedule to the period of lowest consumption, using local or aggregate values - local if operating in single-house mode, aggregate if in multi-house mode. These strategies are called, respectively, **local load distribution** and **aggregated load distribution**. With both strategies, each procedure returns the start time for the period with the most energy available, if any is found. However, where as in local load distribution this period is found by merely sorting the available periods by available power, in aggregate load distribution all locally available periods are sent to the Aggregator, which returns the start time with less aggregated load. For these requests, delay time is not the decisive factor: instead, the scheduler will choose the first procedure that returns an available execution time. As such, procedures 2 and 3 are only used if there is not enough power available along the entire period of acceptable delay.

Considering the interactive nature of the scheduler, the algorithm adopts a best-effort approach. It will always attempt to find a start time earlier than the maximum delay, regardless of schedule. Under heavy consumption, the dynamic priorities explained in section 3.4 promote the shifting of executions with less time-sensitivity to integrate more urgent appliances in the schedule. While the remaining acceptable delay is not the only parameter in the formula, executions with a lower priority value are still more likely to be reschedulable at a later period. Thus, the developed algorithm can be grouped within the dynamic priority scheduling class.

PV energy generation is always treated as complementary energy to the grid, effectively raising the threshold during solar hours. Figure 3.2 represents the scheduling process. For normal and urgent requests, function f returns the minimum value, akin to a $min()$ function. For low-priority requests, f returns the first existing value.

3.6.1 Battery Storage System handling

Battery charges and discharges are treated as executions, although with special rules. BSS executions are created by the system when necessary, scheduled using a different logic. Discharges cannot be interrupted and, as they do not require energy from the grid, are never delayed. Charges, on the other hand, are low priority executions, meant for low demand or solar production periods. Charges can only be interrupted if there are no scheduled discharges depending on the energy that would be charged. Interrupted charges are not rescheduled. Instead, new charges are scheduled when the battery is depleted, or periodically every day.

After a regular appliance is scheduled, the system checks if a consumption peak was formed. If consumption is above 70% of the user-defined threshold and the BSS has enough power, a battery discharge with the power difference between consumption and half of the threshold is scheduled for the corresponding period. Discharges can be stacked, still enabling scheduling procedure 2, as long as the battery can deliver enough continuous power. However, the system cannot charge and discharge simultaneously, nor discharge below a maximum depth-of-discharge (depending on the model).

When a discharge finishes, the system checks if the battery is close to its maximum depth-of-discharge. If it is, a battery charge is scheduled up to full power. Additionally, every midnight, the system checks the battery status and schedules a recharge if it is not at full power. Battery charges are broken in executions with duration up to an hour, to improve flexibility and match the hourly solar production data. Scheduling a battery charge also follows one of two procedures:

1. Solar charge: charge during underutilized solar hours to prevent production waste.

2. Low-demand charge: charge during low consumption periods.

If the household has a PV system, the Coordinator will attempt to schedule based on the first procedure. If there is not enough solar production to fully charge the battery, the system will test if the battery could be fully charged using a combination of solar production and grid energy, still during solar hours. This procedure will only fail if solar production if already fully utilized, or there are not enough solar hours to satisfy the request. The second procedure will attempt to recharge the battery whenever consumption is below 30% of the user-defined threshold, creating executions with the power difference between 30% of the threshold and scheduled consumption. The decision of charging without solar production is justified by the usefulness of the BSS as a load balancing and peak shaving mechanism.

3.7 Execution life cycle

An execution is created with information regarding the appliance and energy profile it serves, as well as a timestamp of the request time. By default, an execution is created as part of the scheduling request handling process, immediately followed by the attribution of a start and finish time. In the default scenario, the request time is the moment the execution is created. However, executions can be created with request times in the future, if the user does not want to schedule immediately but rather at a later time. The execution is in the Pending state until it starts.

Executions have four states: **Pending**, **Started**, **Interrupted** and **Finished**. Once the execution starts, the scheduler updates the data object and, in a real household, would prompt the appliance to start running. Energy profiles have a maximum execution time, which limits the duration of the execution after it is started. The execution may end in one of three circumstances: the maximum execution time is reached, the user manually terminates the execution, or the execution is stopped by the scheduler to enable higher priority executions. The first two scenarios change the execution state to Finished, while the latter signals the execution as Interrupted. Both state changes stop the appliance.

If the execution is terminated manually, there may be opportunities to anticipate Pending executions: the scheduler will automatically attempt to reschedule Pending executions if a nearer start time is found, starting with the highest priority ones and down to the lowest. If the execution is Interrupted, the scheduler will create a new execution with the same parameters to follow it, and retake the Started state at a later time.

3.8 Conclusion

After clarifying the scope of appliances to be managed by the system, scheduling rules and handling of alternative energy systems, all the concepts necessary to understand the operation of the software are presented. The next section - Implementation - focuses on a particular implementation of the system according to the aforementioned features, while also including demonstrative user and testing interfaces.

Chapter 4

Implementation

At its core, a scheduling solution is responsible for starting and stopping executions at the right time. There must be a server to receive the scheduling requests and schedule accordingly, with an asynchronous module to perform the necessary actions at the time of start and finish. The Aggregator, being another system entity, must be a separate module, able to be deployed outside the household and communicate through a common network with Coordinators. This chapter contains a more detailed inspection of the inner workings of system components, data structures and technologies used.

4.1 Data models

It is essential for a scheduler to parameterize appliances, energy profiles, executions and production data from alternative energy sources. The proposed solution attempts to be compatible with all common appliances and include discretizable user preference attributes, to enable the priority algorithm. The implementation of the scheduling logic discussed in section 3.6 is heavily dependent on the data representation of executions and the timetable. The database model with application-relevant data types is represented in figure A.1. This section will elaborate on the aforementioned types and discuss some implementation decisions.

Home

The `Home` structure represents an individual household. Users must define the consumption threshold, in watts (W), before starting the system, through the variable `consumption_threshold`. It is recommended to start with a value based on the contracted power. Energy providers disclose the contracted power in kilovolt-amperes (kVA), a unit used to measure apparent power, instead of watts or kilowatts (kW), used for active power. Knowing the exact consumption limit from a given electrical installation requires calculating its power factor, so it is often more practical to set an estimated value and tweak the value experimentally. Most household equipment is already power factor corrected, resulting in around 80% to 90% of apparent power being usable as active power.

Another crucial variable is `accept_recommendations`, a boolean that represents the operation mode of the system. If `accept_recommendations` is True, the system communicates total consumption values to the Aggregator and requests decisions on the running time of low-priority executions. `outside_id` is the unique identifier of the Coordinator when communicating with the Aggregator.

A home is limited to one PV system and one BSS, both defined through a one-to-one relationship. The scheduler is compatible with houses with or without these systems.

Appliance

`Appliance` represents a single machine or charger in the household. Its main purpose is to store a name and maximum acceptable delay for energy loads belonging to this machine. The variable `name` should unequivocally reference the machine, separating it from appliances of the same model in the same or other rooms. `maximum_delay` defines how much time the user is willing to wait for the appliance to activate after the request. An appliance may have multiple profiles.

Profile

A profile is the description of an energy load. It includes a name, schedulability and priority preferences, maximum duration of usage and rated power. `schedulability` stores whether the energy load is interruptible or non-interruptible. `priority` accepts one of three values: low-priority, normal and immediate. `maximum_duration_of_usage` is a duration field indicating the longest possible execution time. It accepts null values, indicating that the appliance will run indefinitely (e.g., fridge). `rated_power` is the maximum expected consumption, in watts, during a complete execution. If power is being injected to the household, namely during a BSS discharge, the value of `rated_power` is negative. Finally, `hidden` is a flag used to show or hide the profile from the Django-admin interface.

While it is intuitive that an `Appliance` may have multiple profiles, a `Profile` object may also correspond to multiple appliances, in a many-to-many relationship. More than one appliance can have the same energy profile and importance to the user. Schedulability and priority preferences are detailed in sections 3.3 and 3.4, respectively. Maximum duration of usage is used to limit the duration of executions as specified in section 3.7, so that the scheduler can ensure enough energy is available across the specified time.

Execution

`Execution` is a data structure that represents each past, present or future running instance of an appliance. It tracks all changes along the execution life cycle in a persistent manner. An Execution object is created once the user inputs the request to run an appliance, registering the `request_time`, `home`, `appliance` and `profile`. The scheduler decides when the device should execute, writing to the `start_time` and `end_time` fields. The latter is calculated using the previously mentioned `start_time` and `maximum_duration_of_usage` from `Profile`. The boolean variables `is_started`, `is_interrupted` and `is_finished` are updated asynchronously by the scheduler, reflecting the status of the execution. If the execution was created after a previously interrupted execution, fields `previous_progress_time` and `previous_waiting_time` are used to represent the time already spent awaiting execution or running.

BSS charges and discharges, being treated as special executions, follow the same data structure. Profiles are generated by the system to enable charging and discharging with variable wattage, according to the energy needed or available.

BatteryStorageSystem

As the name indicates, `BatteryStorageSystem` is a specialized data type that represents the BSS. It stores model-specific information about the system, such as `continuous_power`, `depth_of_discharge` and `maximum_power_available`. The former two variables refer to the maximum power that the battery can discharge continuously and the maximum depth-of-discharge supported, respectively. A configuration parameter `last_full_charge_time` is used to calibrate the charge/discharge cycles within the scheduler, and can be adjusted in the web interface in case of desynchronization. When the BSS is registered in the system, an `Appliance` object associated to it is created and registered in the `appliance` field.

PhotovoltaicSystem

`PhotovoltaicSystem` stores information pertaining to the PV installation of the household. The fields `latitude` and `longitude` describe the geographic location of the solar panels, used to gather solar radiation data from PVGIS. `tilt` describes the vertical inclination of the installation, and `azimuth` indicates the horizontal orientation of the panels in relation to the equator. `capacity` is the system nameplate capacity provided by the manufacturer, measured in W_{DC}. `tilt`, `azimuth` and `capacity` are parameters required by SAM to generate hourly energy production estimates.

ProductionData

Hourly energy production estimates from SAM are exported as .csv files, one for each month, containing average production values for each hour of the day. `ProductionData` is the data type that stores this data in the database, making it usable without external files. The structure stores the month, hour and corresponding generated power in the `month`, `hour` and `average_power_generated` fields. To end, `system` is a foreign key that identifies the PV system the data pertains to.

ConsumptionData

`ConsumptionData` is the only data structure belonging to the Aggregator. It merely registers the power consumption of a connected household at a certain period. The period is defined using the fields `start_time` and `end_time`, consumption is stored in `power` as a value in watts, and `home_id` corresponds to the `outside_id` from `Home`.

4.2 Program structure

The program is organized in three main components: the Coordinator and Aggregator Django applications mentioned in section 3.2, and a Processor module containing the scheduling logic. From the project root folder, the three are accessible under the `/coordinator/`, `/aggregator/` and `/processor/` folders, respectively. From a layered architecture perspective, Django applications have multiple modules that represent the persistence, functionality and presentation layers, while the Processor represents the core application layer. Figure 4.1 represents the interactions between system modules.

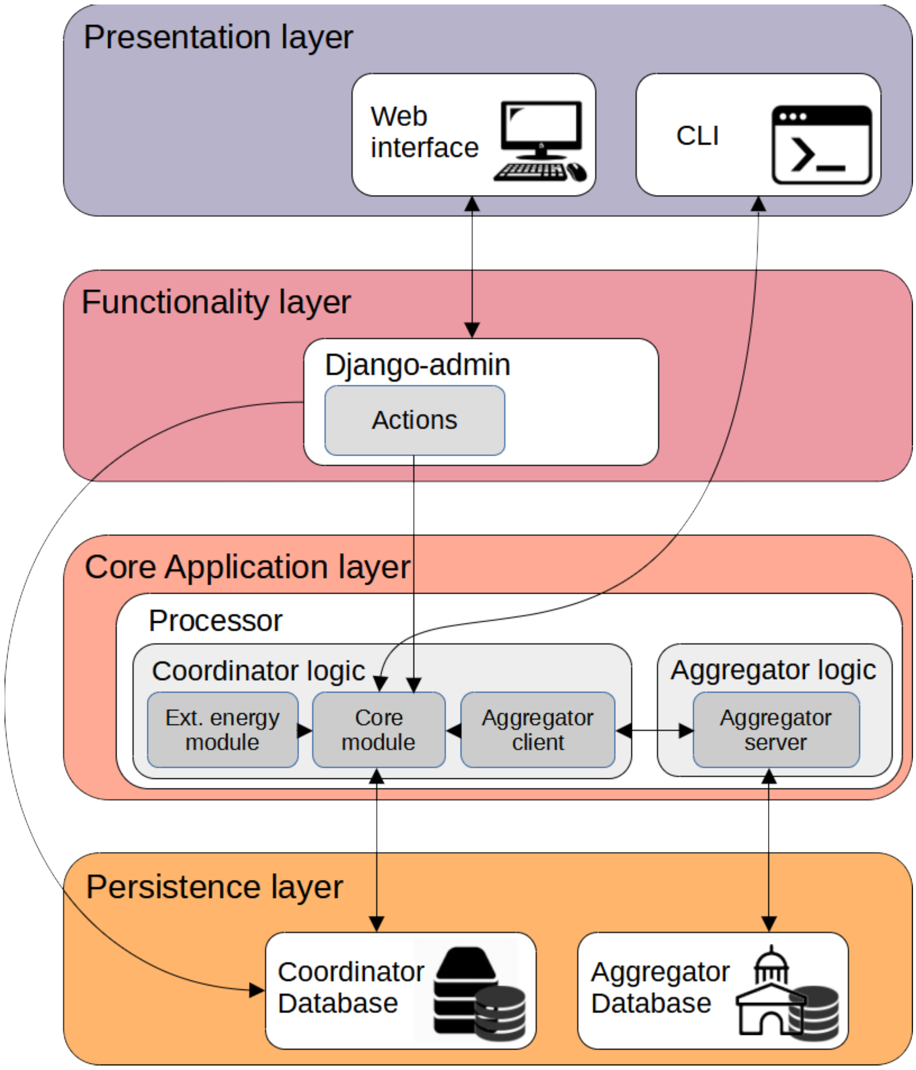

Figure 4.1: Layered software architecture diagram.

In a Django project, database models and web interface views and controllers are declared within its respective applications. All data structures mentioned in the section above, except `ConsumptionData`, are structured using Django models in the `models.py` file of the Coordinator folder, while the single data model used by the Aggregator is listed in `/aggregator/models.py`. When installing the application, Django migrates the instructions in `models.py` to the database. Files with object data can be preloaded to the system as `.json` files, placed under `/coordinator/fixtures/`: the folder contains sample data for the user to adapt to the context of the household, although appliances and profiles can be modified later in the web interface.

The web interface of the system, later expanded in subsection 4.4.1, also pertains to the Coordinator application. Django-admin provides a default interface offering CRUD operations. The interface was modified to reflect the context of the application and clickable actions (e.g., scheduling an execution or switching to multi-house mode) were added under `admin.py`. Filters for executions, appliances and profiles are included in `filters.py`. Furthermore, an extensive testing suite is present in `tests.py`. This testing suite tests all the core functionality of the system present in the Processor component, including communication with the Aggregator.

The Processor includes all the core application logic regarding scheduling, management of executions and alternative energy sources, and client-server code for the interactions between Coordinator and Aggregator. The scheduler itself is a function-based stateless module defined in `core.py`. Being stateless, its functions can be called by any interface and the database is updated directly. For code separation, scheduling functions related to solar and battery energy sources are placed in `external_energy.py`. Further iterations of the system that implement communication with devices could include a new module designed for connecting with a Smart Home hub. It would be a part of the Coordinator logic, interacting with the core module at the start or end of scheduling operations.

Communication with the Aggregator follows a client-server architecture. The Aggregator is expected to be operating in a different computer outside of the local network. As such, it needs a server constantly listening and handling requests. The server code is present in `/processor/aggregator/server.py` and can be activated in the Aggregator machine by running the shell command `run_aggregator.py`, which will listen to connections and requests in an endless loop. A message queue ensures that concurrent requests are handled sequentially and not lost. The client module is present in the same `/processor/aggregator/` folder. It should not be instantiated directly, instead being used by the core module when the system is operating in multi-house mode.

Outside of the three components, the folder `/solardata/` is the destination of solar production data generated by SAM or produced manually, to be read by the system upon configuration. A script located in `/scripts/`, `load_solar_data.py`, imports the `.csv` files. Each file should contain two columns, one for the hour of the day and another for the production value, in kW. Lastly, a PowerShell script found in `/scripts/clean_setup.ps1` automates the entire installation process.

4.2.1 Core application details

Elaborating on the contents of the Processor modules, this subsection explores the scheduling logic implementation of `core.py` and `external_energy.py` in detail. A scheduling request is originated by creating an `Execution` and passing it to the `schedule_execution` function from the core module. As explained in section 3.6, the algorithm calculates the more favorable start times for the characteristics of the appliance and profile. This happens in `propose_schedule_execution`, first by finding all available periods and then by choosing one to represent the procedure. The choice between available times is performed in `choose_execution_time`, according to priority and operation mode.

The scheduler does not implement an explicit timetable with fixed timeslots. Doing so would require maintaining a state of the application, or creating infinite timeslot objects in the database with a rolling window. The former option would affect the modularity of the Processor, while the latter would increase the complexity of the system. More importantly, a timetable would require high granularity to prevent the user from noticing additional execution delays. Instead, the scheduling core takes advantage of the efficiency of the Django ORM to find reference times - moments where executions start and end - and calculate the power consumption or production at each time. Available times for execution are found using this logic. By finding the reference times of unfinished executions, calculating the available power at each time and subtracting consumption, the scheduler can find periods where there is enough power available to accommodate a new execution from its proposed start time along its maximum execution time.

```python
def choose_execution_time(execution, available_periods):
    chosen_time = None
    maximum_delay = execution.appliance.maximum_delay
    home = execution.home
    if maximum_delay is not None:
        available_periods = ...
    if available_periods:
        priority = execution.profile.priority
        if priority is LOW_PRIORITY and home.accept_recommendations:
            chosen_index = int(cli.send_choice_request(...))
            start_time = list(available_periods.keys())[chosen_index][0]
        elif priority is LOW_PRIORITY:
            sorted_periods = {k: v for k, v in \
            sorted(available_periods.items(), ...)}
            start_time = next(iter(sorted_periods))[0]
        else:
            start_time = next(iter(available_periods))[0]
        if start_time != INF_DATE:
            chosen_time = start_time
    return chosen_time
```

Listing 4.1: Code sample of function choose_execution_time.

Executions are linked to a single `Appliance` and `Profile` objects. The rated power of the execution is statically defined in the profile. Therefore, consumption is constant across time within the duration of an execution. This is a limitation of the system implementation: loads of machines that self-regulate their consumption in different phases during the same execution are represented by an upper bound that may be loosely adjusted to real consumption. This may lead to inefficiencies in scheduling. Regardless, the upper bound provided by the rated power serves its purpose of preventing local energy shortages.

Battery charges, being equally treated as executions, are scheduled in `schedule_battery_charge` using the same principles. In this case, the goal of the scheduler is to find enough low demand periods to fit the charge demand so it can be distributed as evenly as possible. Scheduling the battery on solar periods uses the `schedule_battery_charge_on_solar` function. This method will first attempt to schedule exclusively on solar production.

If production exceeds the capacity of the battery, the scheduler attempts to distribute the charging executions evenly. This policy evens the energy excess as much as possible, providing more opportunities for the user to schedule appliances at a later time and prevent waste. While not enough energy is found, the scheduler will gradually raise the threshold of grid energy that is to be subtracted or consumed along with solar to satisfy the request. This parameter, `threshold_multiplier`, starts at -0.25 (production exceeds capacity, charges are capped below solar production), then -0.1, then 0 (only power production can be used). Moving into positive values, 0.1 is tried (10% of the consumption threshold of the residence can be used for charge), then 0.25. These values are subject to change, as there is a trade-off between granularity and request handling runtime. If not enough energy is gathered, the scheduler will simply schedule BSS charges to all periods found with consumption below 30%, regardless of solar production.

Life cycle of executions is managed with asynchronous code running in the background. The background life cycle manager is declared in `\processor\background.py`. When an execution is scheduled, but not started immediately, the scheduler creates a job (similar to Unix cron jobs) for starting and finishing execution at the selected times. The background manager contains a job store that is persisted in the database. As such, it can enact jobs even if the process is terminated, as long as it restarts. To maintain the interactive scheduling functionality, the background manager should be started in the command line and kept running indefinitely, irregardless of the usage of the CLI as an interface for other functionality.

4.3 Technologies used

The scheduling system was developed using **Python** as the primary language for implementing the system logic. All data is stored in a **SQLite** database, accessed and manipulated through **Django**'s Object-Relational Mapping (ORM) as an abstraction layer. Django doubles as a web framework, enabling interactions with the system using a browser interface.

Python [34] is a high-level, imperative, object-oriented programming language with great community support and a large range of applications. The syntax simplicity, ease of integration with front-end frameworks and databases make Python an ideal candidate for small, upscalable projects. Java and C++ were also considered as alternative languages, offering similar feature sets with increased program performance. However, the performance increase is negligible for the requirements of the proposed solution, and Python has a wide support for lightweight development stacks.

SQLite [35] is a stable, open-source, object-relational database system, considered one of the most robust and well-performing database options oriented for small data volumes. It is highly compatible with the aforementioned technologies and able to represent all the interactions and constraints of the proposed solution. Persisted data is expected to range between kilobyte and megabyte orders of magnitude, as a consequence of the executions being periodically cleaned - a small volume of data for a modern database.

Django [36] is an open-source Python web framework based on the Model-View-Controller development pattern. It provides a CRUD interface with automatically generated administrative models, which can be used to easily manipulate program settings. Most security issues are handled out-of-the-box, such as SQL injections and clickjacking.

Finally, ZeroMQ [37] and APScheduler [38] are relevant Python libraries that greatly enabled the implementation. APScheduler enables code to be executed at specified times. It was chosen for its ability to persist scheduling jobs in the database of the system, offering high integration with Django and SQLite and the ability to recover system state. ZeroMQ is a networking module that facilitates both inter-process and TCP communications. It is used as a messaging library for communication between Coordinator and Aggregator. Due to the protocol implementing a level of abstraction over the transport layer, testing can be performed in a single machine with an Aggregator and multiple Coordinator processes running, executing the same exchange of messages that would happen between different machines.

4.4 Interfaces

4.4.1 Django-admin

The Django-admin interface provides limited, authenticated access to data stored in the Coordinator. It is accessible through the browser and can be made available to other machines in public or private networks, potentially offering remote access to the system.

The interface provides basic functionality through CRUD operations. Notably, appliances and profiles can be created or modified and the consumption threshold can be adjusted. Besides CRUD operations, critical scheduling operations are also made available. Users are able to create executions and prompt the Coordinator to schedule them, or manually finish an execution before the maximum execution time is reached. Communication with the Aggregator can also be activated and deactivated, to switch between operation modes. Screenshots of the functionalities mentioned above can be found in appendix C.

Information regarding the PV system installation is visible, but not changeable. In fact, most information within the `Home`, `BatteryStorageSystem` and `PhotovoltaicSystem` database objects is not modifiable, as it corresponds to fundamental configuration to be set up during installation. The web server can be made accessible to external machines, potentially offering remote access to the Coordinator.

4.4.2 Command line (CLI)

The command line is the main interface for launching the web server, configuring the system and preloading data. A clean installation uses Django command line instructions for creating the database, loading sample data from fixtures and creating the user credentials for web access. Whether it is a new installation or the PV system suffers changes (a new model, different azimuth or inclination, etc.), new solar production data must be generated, exported from SAM and imported to the system database. A script `load_solar_data.py` is made available for this purpose.

Besides system configurations, the command line also provides full control over data in the system, including the creation and scheduling of new executions, through the Python shell. Updates on the life cycle of existing executions are printed to the console, and so are the scheduling decisions taken. Request times can be manipulated in order to test the system with custom schedules.

4.5 Conclusion

The suggested implementation meets the system specifications discussed in chapter 3. Data structures include the necessary fields for representation of the household, external energy systems, appliances, energy loads and respective executions. The service logic is separated from the persistence and presentation layers, creating a reliable software architecture that follows modern engineering practices. Besides the core functionality, two interfaces were created to enable user interactions with the Coordinator, along with scripts for configuring the system and importing solar data.

Chapter 5

Results

Chapter 5 evaluates the performance of the system described in the previous chapters, using metrics to quantify the impact on user routines, peak reduction and potential savings against unmanaged households and residential complexes. To conclude, the overall benefits and drawbacks of the system are discussed.

5.1 Methodology

Measuring the performance of the proposed solution requires selecting representative consumption data for different household profiles. However, research data on household consumption with daily granularity and segmentation by appliance is incredibly scarce. As such, it is necessary to create demand scenarios based on aggregate consumption data and patterns identified in section 2.1.

Karananos et al. [10] collected data from Greek residential houses and compared the consumption between users. The most significant differences in consumption came from the number of rooms in the household and the existence of electrical HVAC systems. Based on the demand curves from individual users presented in this study, three types of consumption profiles were created to represent households with different energy dependencies, habits and sets of appliances:

- Household 1 represents a small residence with low overall dependency on electricity. Gas is used for water and space heating in addition to the cooker, leaving the dishwasher, oven, microwave, toaster and coffee machine as the only high consumption electrical kitchen appliances. A single resident produces consumption. Additionally, the household has a vacuum cleaner, a small washing machine, a fridge and a hair dryer. The user is absent from the home during the morning and afternoon.

- Household 2 is home to two residents. In addition to the household appliances present in household 1, the second residence includes an air conditioner, a water heater and an induction cooker. Users have separate morning routines and meal preparations using different appliances.

- Household 3 is a larger house, with a diverse set of electrical appliances, belonging to a family of four elements. At least one family member is always in the house. There is a gaming computer, a cake mixer, three televisions and a soundbar system. Moreover, the family has a PHEV that charges in the garage with energy from the household installation.

The developed system is also used as the testing environment, as the implementation allows accepts the insertion of custom request times through internal testing functions. These scenarios are to be simulated using fabricated daily routines to generate the demand curve over a day. To represent an unmanaged system that will constitute the baseline for comparison, the consumption threshold is set to infinite and all appliances have the same priority class and value. Then, the same demand simulation is repeated using standard contracted power values (3.45, 4.6, 5.75, 6.9 or 10.35 kVA) as the consumption threshold and the default appliance categorization. As the power factor of modern installations is around 90%, the contracted power values are translated to an active power threshold mirroring that difference. To test the system in multi-house mode, the scenarios will be combined to generate aggregate demand.

The chosen Key Performance Indicators for evaluating the system are Peak, Peak-to-Average ratio (PAR) and Average Delay-to-Acceptable-Wait ratio (ADAWR). Peak is the maximum instantaneous net consumption reached, in watts. Net consumption corresponds to the power drawn from the grid, obtained by subtracting production from solar panels and BSS discharges from the total consumption. A lower Peak can promote savings to the consumer by reduction of the contracted power. Peak-to-Average ratio is a measurement of load balance, indicating how far the peak was from the average net power consumption. It is obtained by dividing the Peak by the average value. A value of 1 indicates perfect load distribution, although values between 2 and 5 are considered acceptable. Lastly, ADAWR computes the average delay verified between request and start times of all executions divided by the `maximum_delay` parameter set for each appliance. It is a value between 0 and 1. Values closer to zero are desired.

While Peak and PAR are natural choices for measuring benefits to the user and grid (assuming no preference for certain time periods), ADAWR is a custom measurement chosen to evaluate user forfeiture of control. It is a simple metric to produce and compare, although weighted averages of delays could also have been used. A flat average of delay times would reflect the impact on the user poorly, due to the different priorities for each execution. E.g., a low-priority device is expected to execute with delays, without adverse effects on the user routine. Considering the maximum delay of an appliance is the time after which the user expects to have the machine running in the worst case scenario, regardless of priority, it was chosen instead to provide a more objective parameter for analysis.

5.2 Single-house mode

The household profiles in this chapter are expanded to create the ensuing timetables with execution request times, to be loaded to the scheduler in order to perform the necessary simulations. The details of each simulation set are succeeded by the respective KPI table and discussion of results.

5.2.1 Household 1

Household 1 follows the schedule in table 5.1. The user wakes up at 07:45 AM, takes a shower, prepares breakfast and leaves to work. Comes back at noon for a quick lunch, then leaves again until 05:30 PM. The evening is spent watching television, cleaning the house and preparing dinner. The dishwasher and washing machine are activated at night, before going to bed.

Appliance	Power (W)	Start time (unmanaged)	Start time (managed)	Duration
Fridge	150	00:00	00:00	24:00
Hair dryer	1875	08:05	08:05	00:05
Coffee machine	800	08:22	08:22	00:10
Toaster	1500	08:24	08:24	00:10
Microwave	800	13:10	13:10	00:06
Television	150	17:30	17:30	04:00
Vacuum cleaner	1500	17:50	17:50	00:30
Oven	2500	18:00	**18:20**	01:00
Dishwasher	1200	22:20	22:20	03:00
Washing machine	1800	22:23	**01:20**	01:00

Table 5.1: Sample appliance schedule for household 1. Time format: hh:mm.

Figure B.1 illustrates the consumption curves produced from the schedule, with and without interference from the system. As expected, peaks are formed in the evening and night, since the morning routine mostly uses urgent appliances sequentially and in short periods, and the house is vacant during the morning and afternoon. The water heater is not electric and thus does not integrate the list. To perform this routine without a peak prevention system, the user needs an energy contract with a contracted power value of 4.6 kVA.

In a managed scenario, the scheduler delays two executions. The oven starts at 06:20 PM, after the vacuum cleaner finishes. The oven is defined in the system as a normal-priority appliance with an acceptable delay of 40 minutes: as such, the delay proposed by the system comes within the acceptable range. The second delay is applied to the washing machine. The dishwasher and washing machine are low-priority executions that can execute autonomously, as long as they are loaded with the necessary items (in the case of the washing machine, it should be loaded with laundry and detergent). Household 1 would now be able to lower its contracted power to 3.45 kVA.

KPI	Value (unmanaged)	Value (managed)
Peak	4300 W	2800 W
PAR	7.66	4.99
ADAWR	0.00	0.07

Table 5.2: Key Performance Indicators for household 1.

Analyzing the KPI, Peak and PAR are reduced by approximately 35%. It is a very significant reduction. The PAR values are relatively high, in consequence of the data representing a single user: consumption is residual during most of the day, when the house is vacant or the user is asleep. An ADAWR of 0.07 is caused by the delay in starting the oven. It is a mild inconvenience to the user, who turned on the oven to preheat while finishing household chores; however, the oven is one of the less time-sensitive appliances in the kitchen, as it can preheat or start baking unattended.

5.2.2 Household 2

In household 2, residents start their routine at 07:00 AM and 07:30 AM. Both take showers in the morning, separately, using the water heater at two times. While the second is still in the shower, the first prepares breakfast, creating an overlap. The washing machine is used during the morning so that clothes can naturally dry outside, during the afternoon. One of the users prepares lunch at 12:15 PM. Outside temperature peaks during the afternoon; to maintain the household at a pleasant temperature, the air conditioner is activated. Vacuum cleaning is done during this period. Finally, dinner is prepared using both the induction burner and the oven, and the dishwasher is activated afterwards. The schedule can be found in table 5.3.

Appliance	Power (W)	Start time (unmanaged)	Start time (managed)	Duration
Fridge	150	00:00	00:00	24:00
Water heater	1500	07:10	07:10	00:20
Coffee machine	800	07:34	07:34	00:10
Microwave	800	07:37	07:37	00:06
Water heater	1500	07:40	07:40	00:20
Hair dryer	975	08:00	08:00	00:10
Television	150	08:15	08:15	02:00
Coffee machine	800	08:22	08:22	00:10
Washing machine	2000	09:00	**10:15**	01:00
Induction cooker	2000	12:15	12:15	00:20
Air conditioner	2000	14:00	14:00	04:00
Vacuum cleaner	1500	15:50	15:50	00:30
Oven	2500	19:10	19:10	01:00
Induction cooker	1500	19:55	19:55	00:30
Dishwasher	1200	22:00	22:00	03:00

Table 5.3: Sample appliance schedule for household 2. Time format: hh:mm.

The immediate observation from table 5.3 and figure B.2 is that the scheduling system is unable to improve most of the unmanaged appliance utilization. This is correct: as the oven is a non-interruptible execution, it cannot be interrupted to accommodate the induction cooker, an immediate execution with maximum delay of five minutes. Unfortunately, the two executions together form the peak of consumption in this scenario. It could be reasoned that the user could manually start the induction cooker after the oven is finished, or schedule it before the oven. However, when preparing meals simultaneously, the goal is often to finish cooking all components or dishes at the same time and eat together. As such, household 2 cannot reduce the contracted power below 4.6 kVA (with a threshold of 4150 W), unless it installs a Battery Storage System.

Notably, without the oven and the induction cooker limiting the possible threshold reduction, there is always enough available power to activate the vacuum cleaner while the air conditioner is active. With a threshold below 3650 W, the air conditioner would have been interrupted at the time of start of the vacuum cleaner, and resume after the latter finishes. The only delay is verified in the washing machine, which starts after the time limit of the television is reached. If the television were to be turned off earlier, the washing machine would be rescheduled to start at that moment.

KPI	Value (unmanaged)	Value (managed)
Peak	4150 W	4150 W
PAR	4.21	4.21
ADAWR	0.00	0.01

Table 5.4: Key Performance Indicators for household 2.

In terms of KPI (table 5.4), there is no difference in Peak and PAR. However, the PAR has an acceptable value, indicating a busy schedule with relatively good distribution over the day. ADAWR indicates a very low impact on the household routine, in part due to the limiting factors mentioned above.

5.2.3 Household 3

The third household has a complex schedule, with baths in the morning and evening, multiple cooking appliances used for breakfast, lunch and dinner, a PHEV, a PV system and a BSS. The PV system supplements the household with energy during solar hours. Its module has a nominal efficiency of 20.55% and a module area of $1.6m^2$, supplementing the household with energy during solar hours. The Battery has an energy capacity of 18 kWh, while the PHEV requires 28.6 kWh to fully charge.

The laundry is done in the afternoon and dried (using a clothes dryer) at night. Entertainment devices, such as the soundbar and the gaming computer, are used after dinner. The EV is slow-charged after the owner gets home, at 06:00 PM, and must finish charging before 08:00 AM. It is assumed that the BSS battery is empty at the start of the simulation and must charge during the day. The full schedule is available in table 5.5.

In the baseline scenario, BSS is set to charge and discharge using static logic: the battery is charged between 10:00 AM and 02:00 PM at 4500 W, during peak solar production. Discharges happen between 06:00 PM and 02:00 AM of the next day at 2250 W. This is an effective way to shift solar production from the morning and afternoon to evening and night periods. The EV starts charging immediately when its owner arrives at home, around 06:00 PM, coinciding with the BSS discharge and taking full use of it. The baseline scenario is very optimistic regarding the charge and discharge cycle times, as there is no battery energy wasted. For comparison, the same routine without the Battery Storage System charge/discharge cycle was included in table 5.6 below, as well as in annexed figure B.3.

Using the proposed scheduling system, the Battery Storage System is set to recharge during solar hours, adjusting to the PV energy production. As shown in figure B.3, the household can now operate comfortably under a contracted power of 5.75 kVA.

The scheduling of the battery charge is done at midnight, before the routine of the users, thus being unaware of future demand. As the production exceeds the capacity of the battery, the scheduler distributes BSS charges uniformly below production. At 7:45 AM, the system detects a consumption peak after the start of the smaller induction cooker burner and interrupts charging until 8:00 AM. The induction cooker is an urgent device with a maximum delay of just five minutes: however, when activating the second burner, there are no interruptible devices running, not enough energy stored, and the consumption

Appliance	Power (W)	Start time (unmanaged)	Start time (managed)	Duration
Fridge	150	00:00	00:00	24:00
Water heater	1500	07:00	07:00	00:40
Toaster	1500	07:36	07:36	00:10
Coffee machine	800	07:39	07:39	00:10
Hair dryer	975	07:43	07:43	00:10
Induction cooker	1500	07:45	07:45	00:08
Induction cooker	2000	07:45	**07:49**	00:08
Television	150	08:15	08:15	00:30
BSS (Charge)	*Variable*	10:00	**07:00**	*Variable*
Vacuum cleaner	1500	10:00	10:00	00:30
Television	150	13:00	13:00	02:00
Microwave	800	13:10	13:10	00:20
Coffee machine	800	13:50	13:50	00:10
Dishwasher	1800	14:10	**15:00**	01:50
Washing machine	2400	14:20	**16:50**	01:20
Water heater	1500	17:15	17:15	00:20
Soundbar	600	17:40	17:40	01:30
Electric vehicle	3700	18:00	**19:10**	08:00
Cake mixer	800	18:07	18:07	00:30
Oven	2500	18:10	18:10	01:00
Induction cooker	2000	18:30	18:30	00:20
Water heater	1500	18:40	18:40	00:20
Microwave	800	18:55	18:55	00:06
Hair dryer	975	19:03	19:03	00:10
Gaming computer	500	21:28	21:28	04:00
Clothes dryer	2000	21:30	**03:10**	00:30

Table 5.5: Sample appliance schedule for household 3. Time format: hh:mm.

would exceed the consumption threshold of 4887 W set by the user (85% of the apparent power) plus production. The execution is delayed by four minutes, starting after the toaster and the coffee machine finish, reducing the peak.

Battery discharges are used to mitigate the peak forming between 06:07 PM and 07:10 PM, mostly caused by cooking and showers. The oven requires battery-enabled scheduling to activate. The PHEV, arriving and the garage at 06:00 PM and needing to be fully charged before 08:00 AM, has a maximum delay of six hours. When the car arrives, it is scheduled as a low priority request to execute after the soundbar. The clothes dryer is expected to operate overnight and also has a maximum delay of six hours. This delay is fully utilized to schedule after the car is fully charged.

Battery discharges are activated throughout the evening and night as a load balancing mechanism. As a drawback of this dynamic model, the battery charge is not fully used: at the end of the time period, the battery is at 22% charge. However, one of the goals of the battery energy cycle management logic is to enable executions when there is not enough energy available. The remaining energy is available to use if additional scheduling is requested.

KPI table 5.6 demonstrates the significant benefits offered by the system regarding peak reduction and load balancing. Using the scheduler with a BSS offers improvements of 57% in these regards when compared to an unmanaged household with no storage solution, and a 45% benefit over unmanaged households with timetabled BSS systems. ADAWR remains at very low values, with the most delayed

KPI	Value (unmanaged, no BSS)	Value (unmanaged, static BSS)	Value (managed)
Peak	10303 W	8053 W	4435 W
PAR	10.85	8.48	3.97
ADAWR	0.00	0.00	0.07

Table 5.6: Key Performance Indicators for household 3.

appliances being the clothes dryer, a low-priority execution, and the induction cooker. Waiting four minutes for an urgent execution may be unacceptable for some users. If the maximum delay were set to one minute, the Peak would increase to 4935 W, as the appliance would still start after the coffee machine. Thanks to the solar production of 1.05 kWh during that period, the demand would still fit within the contracted power comfortably.

5.3 Multi-house mode

The operating mode is to be tested concerning the impact on peak consumption and added delays to individual households, as well as its load balancing effect on the aggregate data. Data will be gathered using two sets of simulations. First, an heterogeneous set of households and requests will reuse the scenarios from the section above. The second will focus on the distribution of low-priority executions in five households with similar routines to household 1.

Similarly to the testing performed in the previous section, an unmanaged scenario is used as a baseline for comparison with the system results. The simulation is repeated with the three houses in single-house mode, sending consumption data to the Aggregator but not accepting suggestions, and lastly in multi-house mode.

5.3.1 Heterogeneous households

To test the multi-house mode in an heterogeneous environment, the three houses represented by the household profiles created in this chapter were connected to the Aggregator and simulated together. The requests from each residence remain the same as represented in tables 5.1, 5.3 and 5.5, although the start times are expected to vary in multi-house mode. However, due to the complexity of the schedules, discussion will be based on the KPI values, obtained for each household and conjoined. The net consumption curves corresponding to the results for each simulation can be found in figure B.4.

KPI	Value (unmanaged)	Value (single-house)	Value (multi-house)
Peak	11003 W	11651 W	8002 W
PAR	3.60	3.78	3.25
ADAWR	0.00	0.06	0.12

Table 5.7: Key Performance Indicators for the heterogeneous simulations.

Baseline results compound the unrestricted consumption of each house, leading to a high Peak. Surprisingly, the KPI data for the cluster in table 5.7 shows that the Peak, when operating in single-house mode, is even larger: each house, operating in its best interests without concerns about grid stability, will create a larger consumption peak at 10:20 PM. Using the communication with the Aggregator to improve the distribution of low-priority appliances, Peak is reduced to 8002 W (formed in the late evening). In this scenario, multi-house mode offers a 27% peak reduction in the network shared between the three houses against the baseline, with slightly more disruption to the user than in single-house mode.

Despite not shown in the metrics, the system also accomplishes to prevent energy waste. In the baseline simulation, energy consumption is negative at 09:00 AM and 02:00 PM for the entire complex. This corresponds to excess production from household 3 that is not matched by the consumption of all three residences. Depending on the ability to sell energy to the rest of the grid, this energy excess may not be monetized or may even be charged by the energy company.

KPI	Household 1	Household 2	Household 3
Peak	2800 W	4150 W	4926 W
ADAWR	0.24	0.06	0.07

Table 5.8: Key Performance Indicators for the individual households, operating in multi-house mode.

Table 5.8 shows the influence of the operation mode in the user-centered metrics. ADAWR is low across the three houses, although the value in household 1 is larger than in single-house mode. The Peak increases for household 3, forcing it to raise the consumption threshold to accommodate the execution of the oven at 06:00 PM. The issue arises from a decision made by the Aggregator hours before, to schedule the washing machine for that same period. The washing machine, despite being an interruptible appliance, could not be delayed further due to the maximum delay of four hours set by the user. As the components are unaware of future demand, when the oven request arrived, the two machines had to execute simultaneously. The peak increase is still below the recommended threshold for households with a contracted power of 5.75 kVA. Thus, in this situation, the benefits provided to the user were not affected.

5.3.2 Night distribution

The second multi-house scenario is exclusively dedicated to distributing low-priority appliances at night. Household 1 represents a fraction of users that prefer to have large machines operating overnight. As such, this simulation set uses five houses with the same schedule as household 1. The scheduling request times for the dishwasher and washing machine are spread using pseudo-random values, taken from a normal distribution with 0 mean and a standard deviation of 60, added to the original time of request present in table 5.1.

KPI	Value (unmanaged)	Value (single-house)	Value (multi-house)
Peak	15000 W	9000 W	3000 W
PAR	10.00	6.00	2.00
ADAWR	0.00	0.12	0.47

Table 5.9: Key Performance Indicators for the pattern-repeated simulations between 09:00 PM and 12:00 AM.

The benefits are overwhelmingly clear in this situation. Table 5.9 reveals that, in an unmanaged scenario, all machine requests would overlap and create a Peak of 15000 W. In single-house mode, the dishwasher and washing machine are ensured to run at separate times within each household, but the aggregate PAR is still excessive due to the lack of cooperation. Only in multi-house mode the loads are distributed with consideration for the state of the network, reducing the peak to 3000 W and achieving an acceptable PAR. On average, appliances started execution within less than half of their maximum delay. Considering the executions under scrutiny have low priority, the increased delay time is still within acceptable values, despite being the highest registered.

The concept is very effective in distributing energy loads with large maximum delays in low-demand periods. In fact, this was the original idea behind multi-house mode. Loads that would otherwise start when residents go to sleep are now distributed across the full night. Annexed figure B.5 exhibits the consumption curves for the three simulations.

5.4 Conclusion

The results show the high effectiveness of the presented solution in most consumption scenarios. In single-house mode, the system is able to reduce consumption peaks by up to 35% purely by scheduling consumer appliances more efficiently. When used along with a PV system and a BSS, the scheduler is able to shift the energy produced in excess to later periods of the day, and use the battery discharges both as a peak shaving and a load balancing mechanism. Residents in households 1 and 3 were able to reduce their contracted power, receiving an economic incentive.

In multi-house mode, the cooperative distribution of low-priority appliances reduced the aggregate Peak by 27% in an heterogeneous environment, at the expense of limiting the ability to shift appliances locally for higher priority requests. When used in households with more affinity to operating low-priority appliances at night, the benefits can be much greater. Nonetheless, the results show there is margin for algorithm improvements regarding the handling of low-priority executions in high demand periods.

Evidently, the energy distribution company stands to gain by having the same demand distributed more evenly, requiring less resources to accommodate consumption peaks, and the benefits to individual consumers are similar to single-house mode. In residential complexes with shared energy contracts, property managers also benefit directly from a lower contracted power. The benefit to property managers is more indirect outside of shared contracts. Nonetheless, either through new incentives promoted by distribution companies or by running its own Coordinator to lower expenses in common property, there is also potential for profits to this stakeholder.

Chapter 6

Conclusions

As the world population grows and most energy generation is still dependent of finite resources, optimizing energy consumption is paramount to a sustainable future, where dependency on peaking plants can be reduced and shortages can be prevented. The system developed is an effective tool to promote sustainability. Chapter 6 reiterates the scope and capabilities of the system, along with the areas where it could be improved in future iterations.

6.1 Achievements

In this document, an interactive scheduler focused on peak control is proposed. It is a server application able to start, delay, interrupt and resume appliances based on present demand, cooperate with other instances, include power generation from alternative sources and utilize battery energy storage systems as peak shaving and load balancing mechanisms.

Many scheduling solutions discussed share a common goal in maintaining demand equilibrium and lowering energy costs for consumers; however, most are unpractical due to the necessity of scheduling in advance. The proposed solution distinguishes itself by taking scheduling requests as they happen in the consumer routine. Acceptable delays are defined by the user for each appliance, furthering the adaptability of the system to the reality of consumers. Another differentiating factor is the multi-house mode, which promotes cooperation between households sharing grid infrastructure. This can be used to reduce shared expenses, prevent energy shortages and reduce demand imbalance during low-activity periods, rewarding multiple stakeholders. The results, although not comprehensive, corroborate the benefits to all parties, achieving reductions in contracted power for two out of the three individual scenarios and better energy distribution in shared buildings.

6.2 Future Work

The proposed solution, in its current state, is a software system that lacks integration with physical hardware of devices. The first step towards real-world application would be to integrate the system with a smart home hub. The hub would act as a gateway, able to send and receive communications with the Coordinator through WiFi, and translate ZigBee (or other protocol) instructions to and from the machines.

The system could be further iterated with improvements on the algorithm and energy estimation. Regarding the former, the system showed limitations in fully utilizing the BSS charge/discharge cycle due to the need to have energy available for future execution requests, even if they do not come to exist. As such, the system could be improved by learning when the residents are absent or asleep, in order to consume the remaining energy on the executions to be completed. Another limitation is in the scheduling of interruptible low-priority appliances using aggregate consumption data: appliances may be scheduled in periods close to the maximum delay from the beginning, leaving no room for interruptions and removing flexibility to local, higher-priority schedules. A time limit for the first schedule relative to the maximum delay could be implemented. Potentially, users could also parameterize the times at which they want to let the Aggregator make decisions, or select which appliances are to be scheduled cooperatively.

All appliance profiles are represented with a fixed power within the server application. This depiction is accurate for resistance- and motor-based appliances. Appliances with different programs of operation set by the user are also covered adequately, as the user can select the profile when creating the execution request. However, the rated power is often an upper bound that represents a small fraction of the execution time in self-regulating appliances. The system lacks automated support for periodical switching between profiles, which could be used to represent different phases of the operating cycle in machines such as the dishwasher and air conditioner. Alternatively, dynamic consumption curves for each model and operation mode could be implemented to improve the characterization of the energy loads, at the expense of scalability.

In a more advanced solution, energy resources could be shared within the microgrid of the shared residential space. Excess PV and BSS energy could be distributed and requested by other households creating a local energy market, or the resources could be pooled together to optimize aggregate consumption.

Bibliography

[1] M. T. Toman and B. Jemelkova. Energy and economic development: an assessment of the state of knowledge. *The Energy Journal*, 24(4), 2003.

[2] W. D. Nordhaus, H. Houthakker, and R. Solow. The allocation of energy resources. *Brookings Papers on Economic Activity*, 1973(3):529–576, 1973. ISSN 00072303, 15334465. URL http://www.jstor.org/stable/2534202.

[3] United Nations. Achieving targets on energy helps meet other Global Goals, UN forum told - United Nations Sustainable Development, 2018. URL https://www.un.org/sustainabledevelopment/blog/2018/07/achieving-targets-on-energy-helps-meet-other-global-goals-un-forum-told-2/. Last accessed on 27 May, 2022.

[4] United Nations. Transforming our world: The 2030 agenda for sustainable development, 2015. URL https://sdgs.un.org/sites/default/files/publications/21252030%20Agenda%20for%20Sustainable%20Development%20web.pdf. Last accessed on 27 May, 2022.

[5] G. M. Masters. *Renewable and efficient electric power systems*. John Wiley & Sons, 2013.

[6] T. Stetz, M. Rekinger, and I. Theologitis. Transition from uni-directional to bi-directional distribution grids. *Management Summary of IEA Task*, 14, 2014.

[7] X. Fang, S. Misra, G. Xue, and D. Yang. Smart grid — the new and improved power grid: A survey. *IEEE Communications Surveys Tutorials*, 14(4):944–980, 2012. doi: 10.1109/SURV.2011.101911.00087.

[8] Eurostat. Energy consumption in households, 2021. URL https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Energy_consumption_in_households. Last accessed on 27 May, 2022.

[9] Odysee-Mure. Energy efficiency trends for households in the EU, 2021. URL https://www.odyssee-mure.eu/publications/efficiency-by-sector/households. Last accessed on 27 May, 2022.

[10] A. Karananos, A. Dimara, K. Arvanitis, C. Timplalexis, S. Krinidis, and D. Tzovaras. Energy consumption patterns of residential users: A study in Greece. In *International Conference on Computer Vision Systems*, pages 639–650. Springer, 2019.

[11] I. N. de Estatística. Preliminary results of the survey on energy consumption in households - 2020, 2021. URL `https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_destaques&DESTAQUESdest_boui=512530049&DESTAQUESmodo=2&xlang=en`. Last accessed on 27 May, 2022.

[12] S. Nasarre-Aznar, M. Economidou, V. Sagaert, E. Laes, M. Wüstenberg, J. Kauppinnen, P. Puhakka, P. Pylsy, C. Schmid, T. Pinkel, M. Habdas, R. Wnuk, S. Passinhas, S. Bright, T. Fawcett, D. Weatherall, C. Peyton, and F. McCarthy. Energy efficiency upgrades in multi-owner residential buildings. *JRC Publications*, 02 2019. doi: 10.2760/966263.

[13] T. Ao, H. Nagai, T. Iwata, N. Hitsuda, M. Yamaha, and K. Shuku. A study on electric power consumption of common space in multiple dwelling houses. *Journal of Environmental Engineering (Transactions of AIJ)*, 76(669):1017–1023, 2011. doi: 10.3130/AIJE.76.1017.

[14] M. Syed, G. Morrison, and J. Darbyshire. Energy allocation strategies for common property load connected to shared solar and battery storage systems in strata apartments. *Energies*, 13:6137, 11 2020. doi: 10.3390/EN13226137.

[15] S. Malaviya, S. Krishnan, S. Cibi, S. Nath, D. S. Krishnan, et al. Assessing clean energy opportunities through demand aggregation in bengaluru's apartment buildings. Technical report, World Resources Institute India, 2020.

[16] E. H. Et-Tolba, M. Ouassaid, and M. Maaroufi. Smart home appliances modeling and simulation for energy consumption profile development: Application to moroccan real environment case study. In *2016 International Renewable and Sustainable Energy Conference (IRSEC)*, pages 1050–1055, 2016. doi: 10.1109/IRSEC.2016.7983908.

[17] M. Pipattanasomporn, M. Kuzlu, S. Rahman, and Y. Teklu. Load profiles of selected major household appliances and their demand response opportunities. *Smart Grid, IEEE Transactions on*, 5: 742–750, 03 2014. doi: 10.1109/TSG.2013.2268664.

[18] Sense. Sense.com, 2021. URL `https://sense.com/`. Last accessed on 27 May, 2022.

[19] T. Choi, K. Ko, S. Park, Y. Jang, Y. Yoon, and S. Im. Analysis of energy savings using smart metering system and ihd (in-home display). In *2009 Transmission & Distribution Conference Exposition: Asia and Pacific*, pages 1–4, 2009. doi: 10.1109/TD-ASIA.2009.5356956.

[20] D. Crossley. *Advanced metering for energy supply in Australia*. Energy Futures Australia, 2007.

[21] D. Han and J. Lim. Smart home energy management system using ieee 802.15.4 and zigbee. *IEEE Transactions on Consumer Electronics*, 56(3):1403–1410, 2010. doi: 10.1109/TCE.2010.5606276.

[22] M. G M, C. Vyjayanthi, and C. Modi. *An Open-Hardware Approach for IoT Enabled Smart Meter Monitoring and Controlling System Using MQTT Protocol*, pages 303–317. Springer, 03 2020. ISBN 978-981-15-3665-6. doi: 10.1007/978-981-15-3666-3_25.

[23] EDP Energias de Portugal. Re:dy, 2018. URL https://www.edp.com/en/innovation/redy. Last accessed on 27 May, 2022.

[24] DTE Energy. Smartcurrents: A smarter way to save energy, 2019. URL https://empoweringmichigan.com/smartcurrents-a-smarter-way-to-save-energy/. Last accessed on 27 May, 2022.

[25] A. Mohsenian-Rad, V. W. S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia. Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid. *IEEE Transactions on Smart Grid*, 1(3):320–331, 2010. doi: 10.1109/TSG.2010.2089069.

[26] X. Chen, T. Wei, and S. Hu. Uncertainty-aware household appliance scheduling considering dynamic electricity pricing in smart home. *IEEE Transactions on Smart Grid*, 4(2):932–941, 2013. doi: 10.1109/TSG.2012.2226065.

[27] Y. Ozturk, D. Senthilkumar, S. Kumar, and G. Lee. An intelligent home energy management system to improve demand response. *IEEE Transactions on Smart Grid*, 4(2):694–701, 2013. doi: 10.1109/TSG.2012.2235088.

[28] B. Zhou, W. Li, K. W. Chan, Y. Cao, Y. Kuang, X. Liu, and X. Wang. Smart home energy management systems: Concept, configurations, and scheduling strategies. *Renewable and Sustainable Energy Reviews*, 61:30 – 40, 2016. ISSN 1364-0321. doi: 10.1016/J.RSER.2016.03.047.

[29] C. Chen, J. Wang, and S. Kishore. A distributed direct load control approach for large-scale residential demand response. *IEEE Transactions on Power Systems*, 29(5):2219–2228, 2014. doi: 10.1109/TPWRS.2014.2307474.

[30] G. Pinto, A. Kathirgamanathan, E. Mangina, D. P. Finn, and A. Capozzoli. Enhancing energy management in grid-interactive buildings: A comparison among cooperative and coordinated architectures. *Applied Energy*, 310:118497, 2022. ISSN 0306-2619. doi: 10.1016/J.APENERGY.2021.118497.

[31] B. MacLeod. Benefits of modern energy management platforms. *Condominium Manager*, page 49–50, Dec 2019.

[32] N. R. E. Laboratory. System Advisor Model, 2018. URL https://sam.nrel.gov/. Last accessed on 27 May, 2022.

[33] European Commission. JRC Photovoltaic Geographical Information System (PVGIS), 2022. URL https://re.jrc.ec.europa.eu/pvg_tools/en/#TMY. Last accessed on 27 May, 2022.

[34] Python Software Foundation. Welcome to python.org, 2022. URL https://www.python.org/. Last accessed on 27 May, 2022.

[35] SQLite Consortium. SQLite Home Page, 2022. URL https://www.sqlite.org/index.html. Last accessed on 27 May, 2022.

[36] Django Software Foundation. The web framework for perfectionists with deadlines, 2022. URL https://www.djangoproject.com/. Last accessed on 27 May, 2022.

[37] ZeroMQ community. ZeroMQ, 2022. URL https://zeromq.org/. Last accessed on 27 May, 2022.

[38] A. Grönholm. Advanced Python Scheduler, 2022. URL https://apscheduler.readthedocs.io/en/3.x/. Last accessed on 27 May, 2022.

Appendix A

Database structure

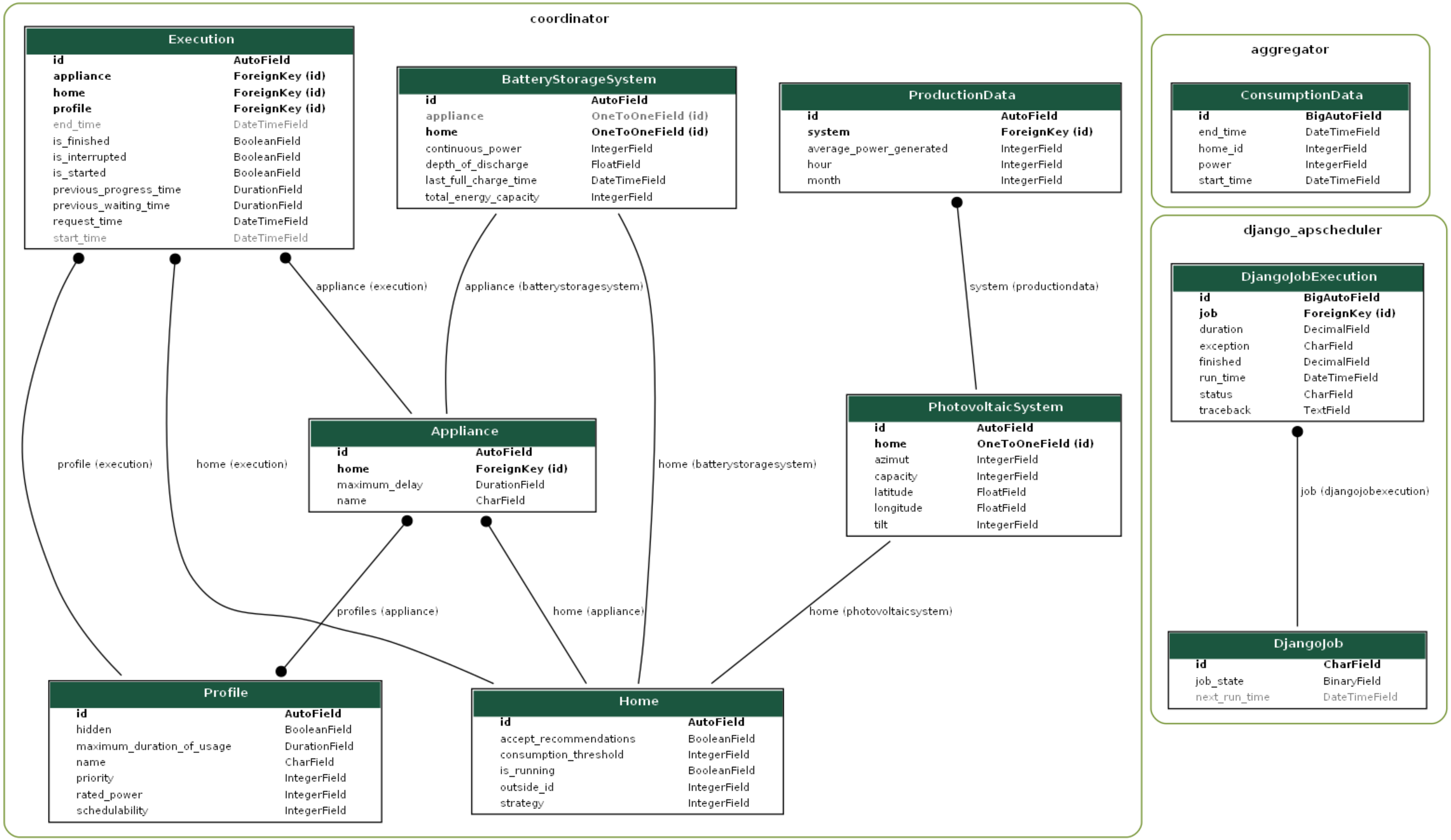

Figure A.1: Database representation of data structures.

Appendix B

Sample consumption data plots

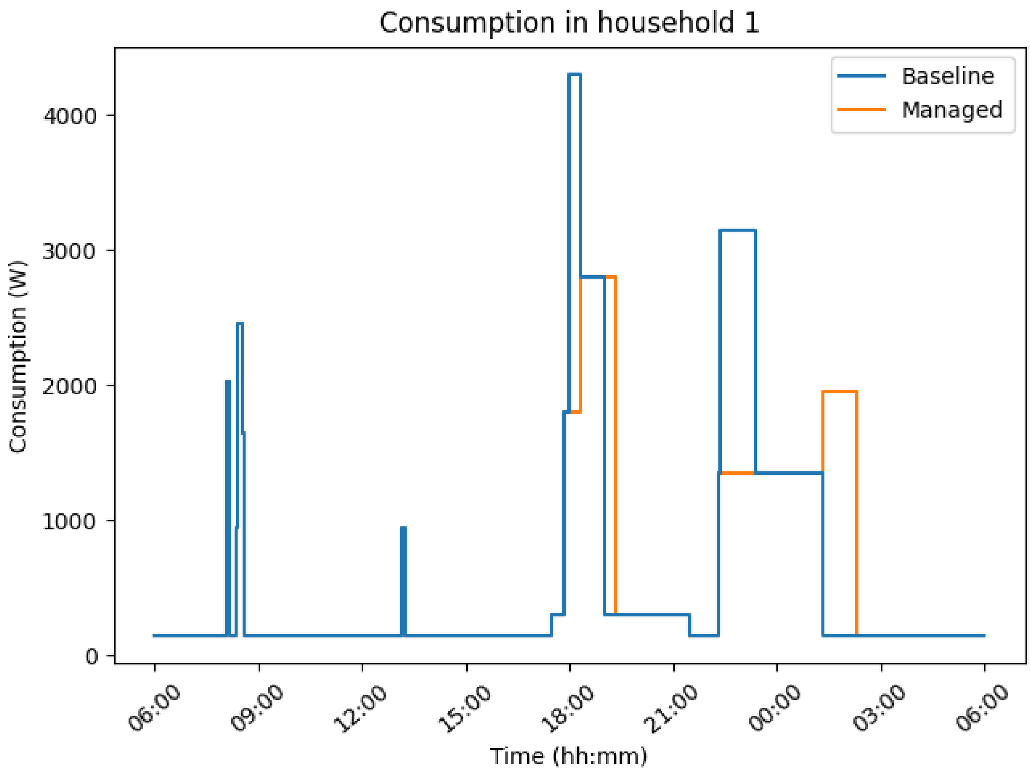

Figure B.1: Comparison of consumption curves for household 1.

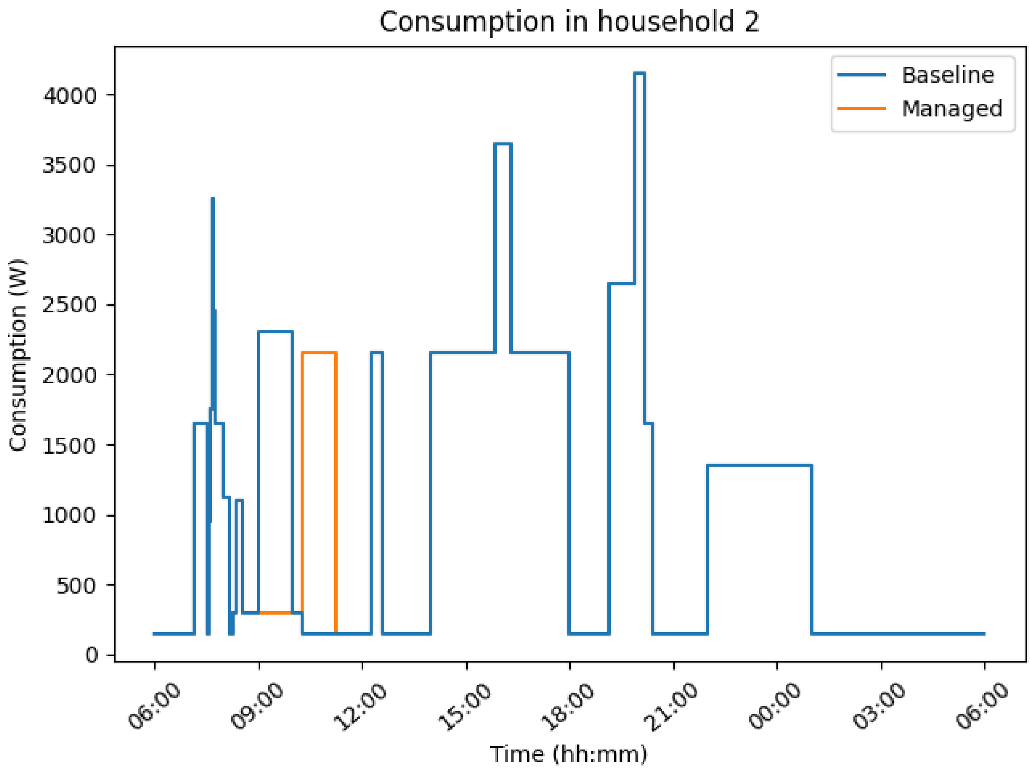

Figure B.2: Comparison of consumption curves for household 2.

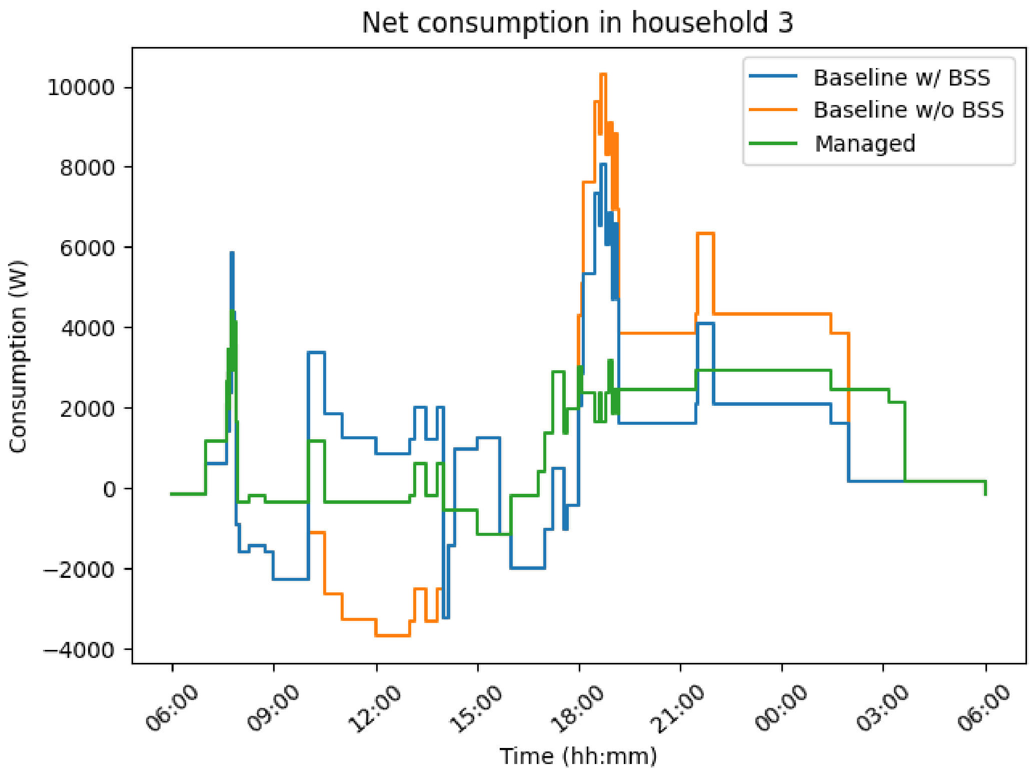

Figure B.3: Comparison of consumption curves for household 3.

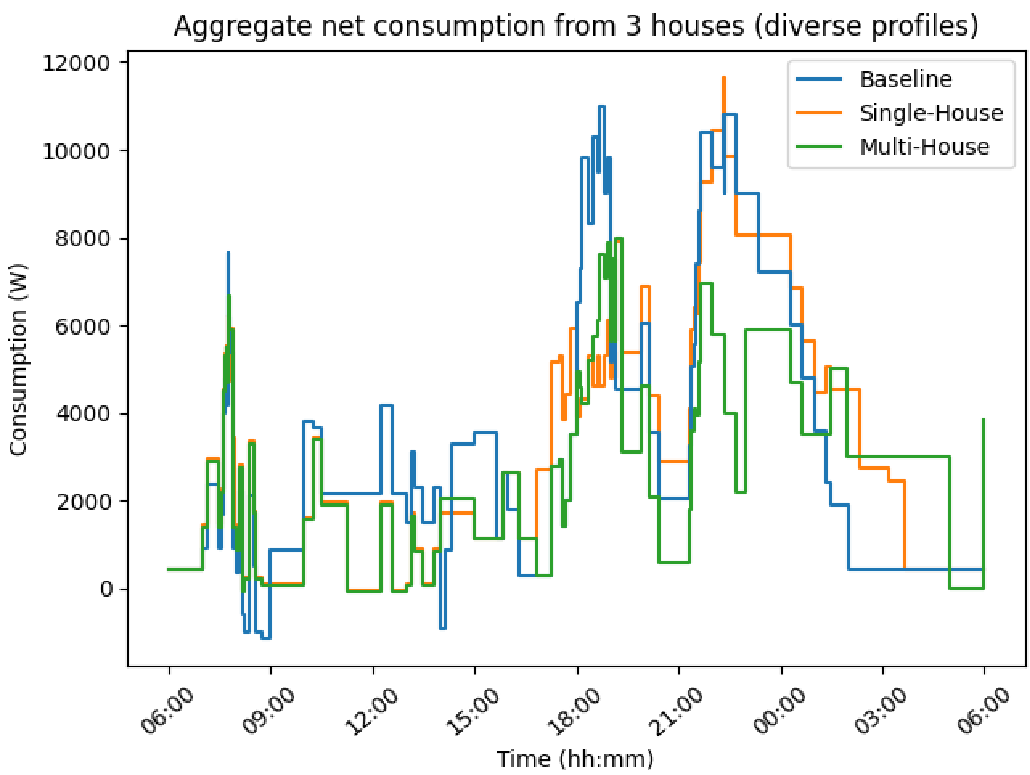

Figure B.4: Comparison of consumption curves for the heterogeneous simulation.

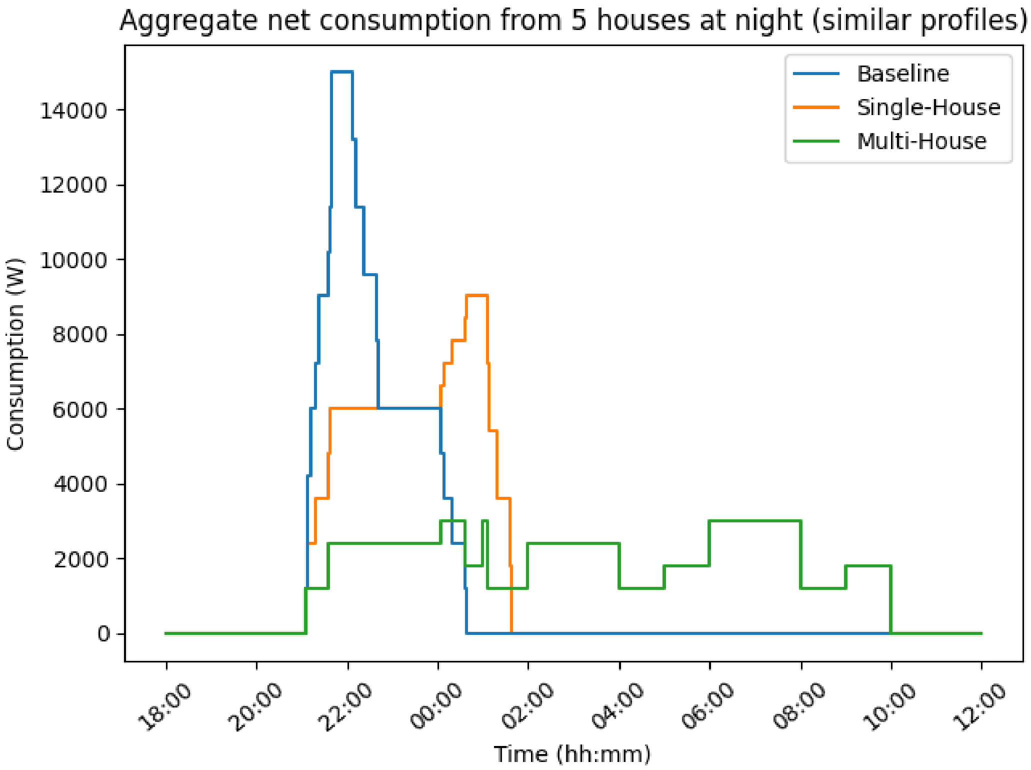

Figure B.5: Comparison of consumption curves for the pattern-repeated simulation at night.

Appendix C

Django-admin interface screenshots

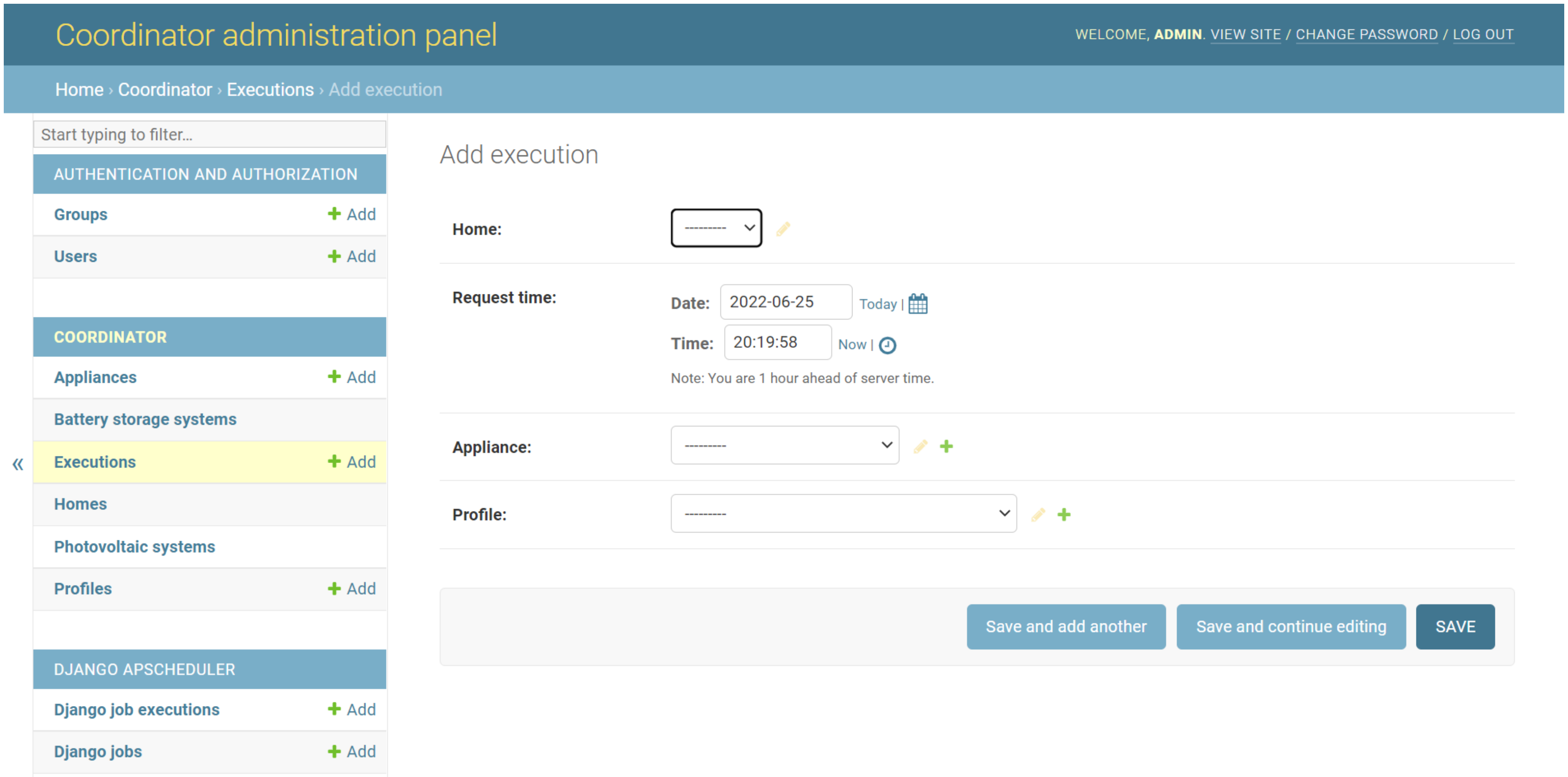

Figure C.1: Django-admin interface: creating a new execution.

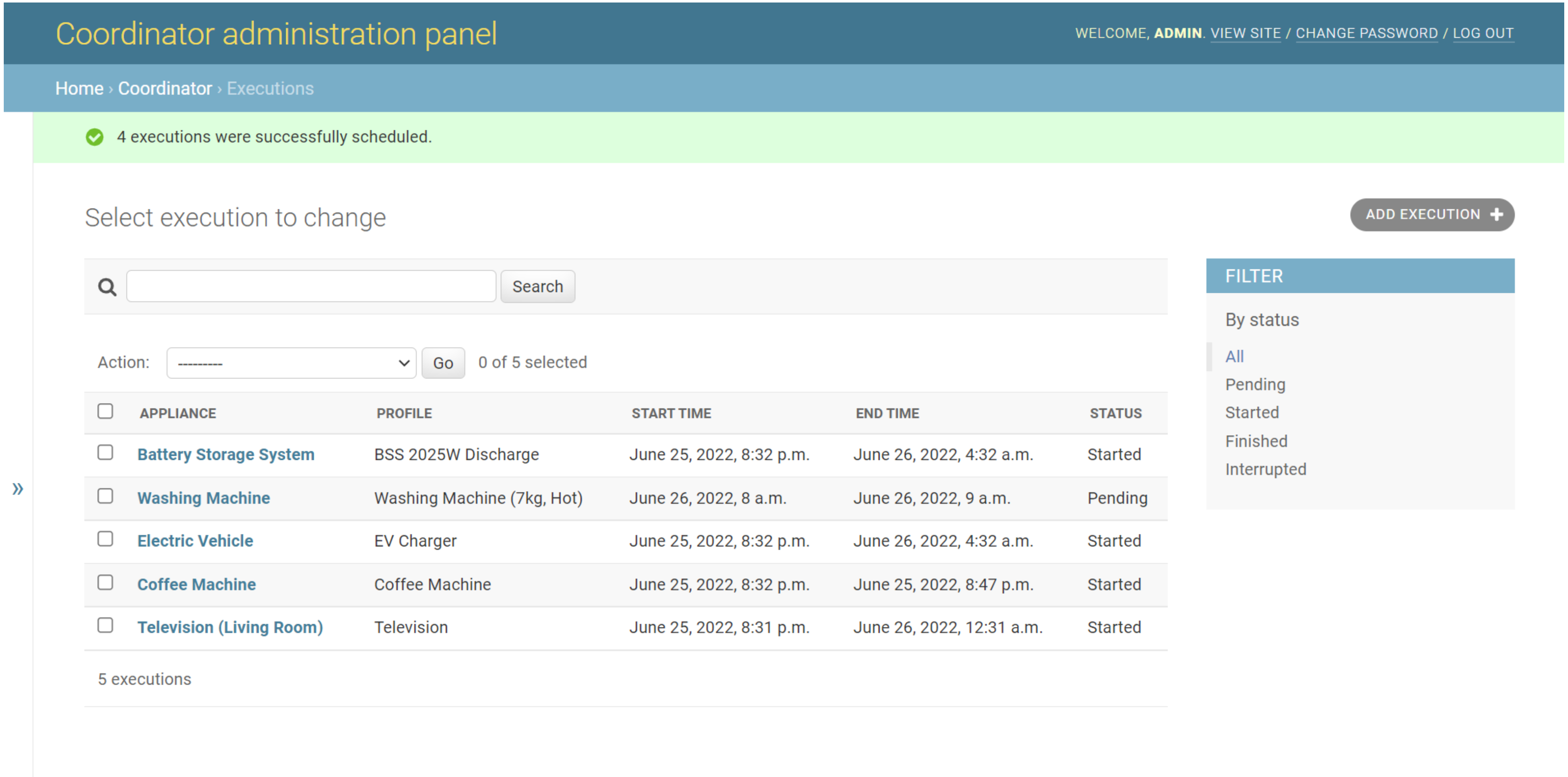

Figure C.2: Django-admin interface: displaying executions after scheduling.

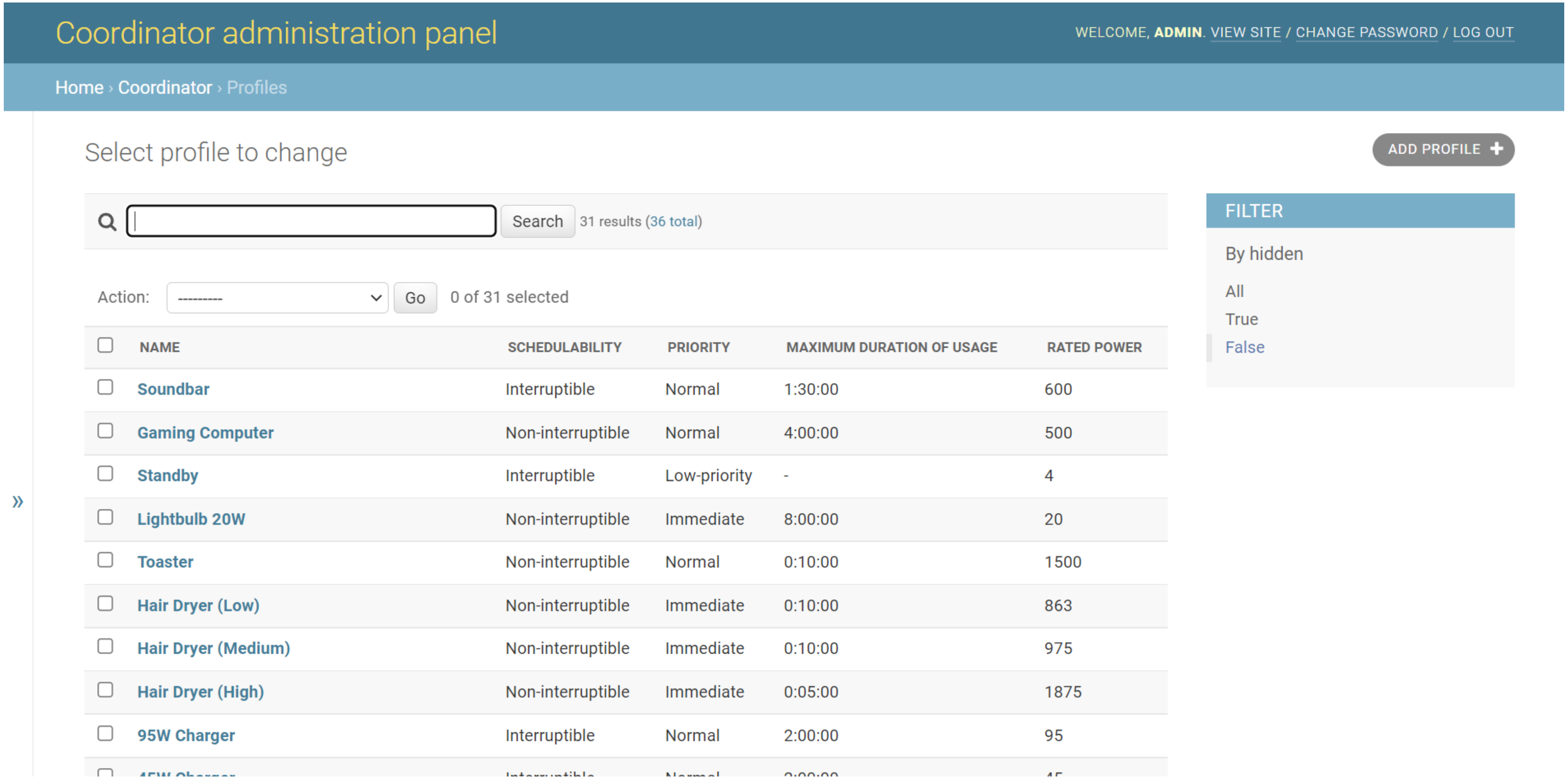

Figure C.3: Django-admin interface: displaying energy profiles.

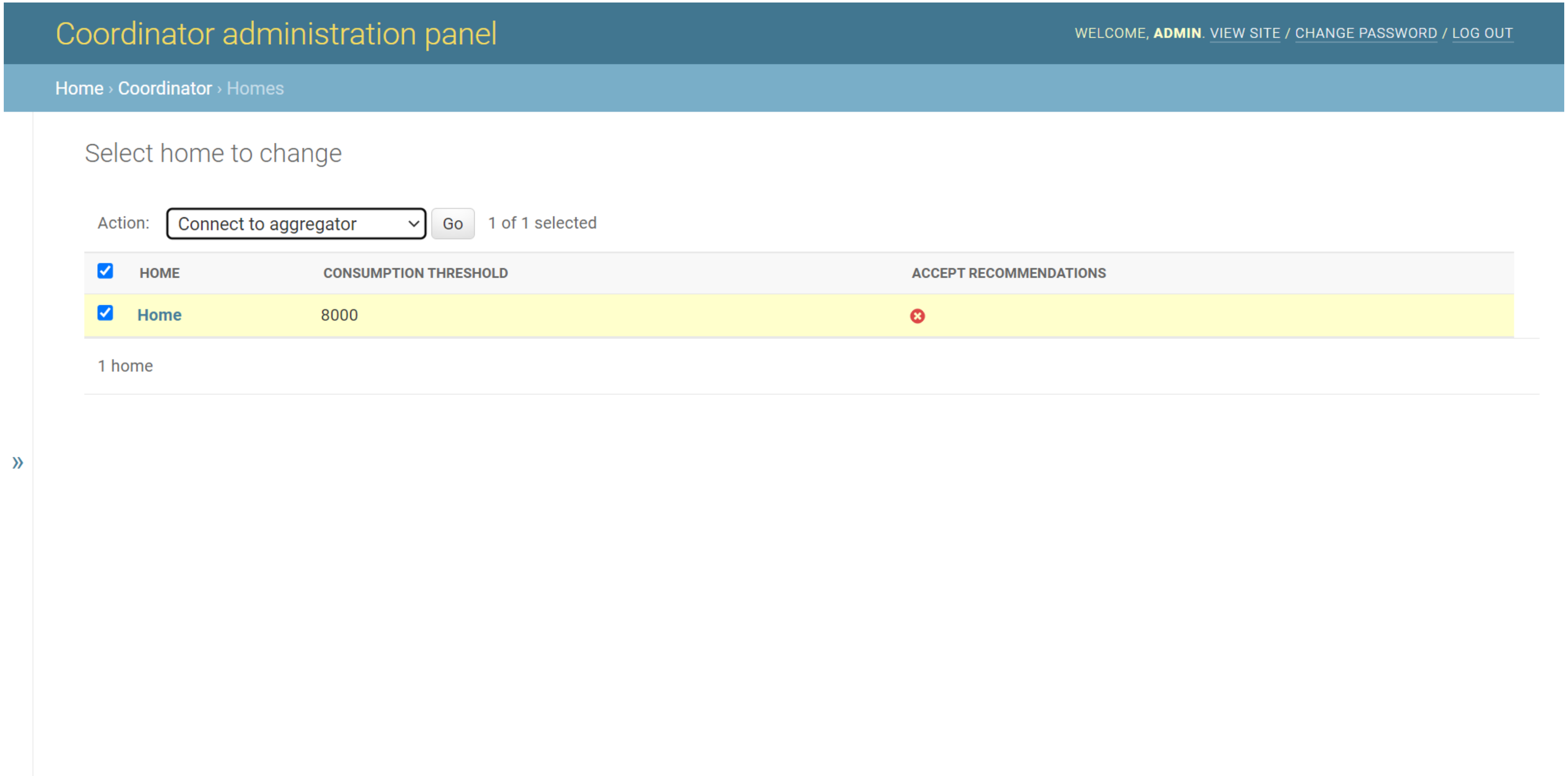

Figure C.4: Django-admin interface: switching to multi-house mode.