



**Politécnico
de Viseu**

Escola Superior
de Tecnologia
e Gestão de Viseu

Contribution to electric mobility - A Mesh approach for Intelligent Electric Vehicle Charging

Pedro Miguel Pereira Baptista

Trabalho de Projeto

Mestrado em Engenharia Informática - Sistemas de Informação

Trabalho efetuado sob a orientação de
Professor Doutor Filipe Caldeira
Professor Doutor Filipe Cardoso

Novembro de 2022



**Politécnico
de Viseu**

Escola Superior
de Tecnologia
e Gestão de Viseu

Contribution to electric mobility - A Mesh approach for Intelligent Electric Vehicle Charging

Pedro Miguel Pereira Baptista

Trabalho de Projeto

Mestrado em Engenharia Informática - Sistemas de Informação

Trabalho efetuado sob a orientação de

Professor Doutor Filipe Caldeira
Professor Doutor Filipe Cardoso

Novembro de 2022

Declaration

I, Pedro Baptista, of the Department of Computer Science, School of Technology and Management - Polytechnic of Viseu, confirm that this is my own work and figures, tables, equations, code snippets, artworks, and illustrations in this report are original and have not been taken from any other person's work, except where the works of others have been explicitly acknowledged, quoted, and referenced. I understand that if failing to do so will be considered a case of plagiarism. Plagiarism is a form of academic misconduct and will be penalised accordingly.

Pedro Miguel Pereira Baptista

Abstract

The path to sustainable electric mobility depends on multiple factors such as reducing emissions, energy efficiency and conscious usage. This project presents itself as a contribution to the improvement of Electric Vehicle (EV) charging, with a positive impact from the moment of connection to the Energy Distributor (ED) until the EV charging finishes.

The electric charging solutions for the residential market imply, in many situations, an increase in the contracted power to respond to an inefficient charging cycle that starts when the charger is connected and ends when the VE battery maximum charge fulfils. The increase in contracted power is not always the best solution for faster and more efficient charging.

When single points of connection to the power grid are available, common to a group of tenants, it becomes harder to adopt electric mobility solutions. Managing the contracted power, optimising charging time, controlling costs according to the connected user, or even balancing the charging of EVs according to the source of energy at a given time, are challenges that the presented proposal contributes to solve.

With focus on the residential market, this solution is suitable for both single-use connection points and shared connections, which are very frequent in apartment buildings without a closed garage or which, for other reasons, share the available electrical connections to the grid. The proposed architecture allows the use of one or several common charging points, applying a mesh network of intelligent chargers orchestrated by a residential gateway.

Keywords: Electric Vehicles, EV Charging, Mobility, Mesh, Fog Architecture, Load Management, Mobility, Data Management.

Abstract (Português)

O caminho para a mobilidade elétrica sustentável depende de múltiplos fatores, como redução de emissões, eficiência energética e uso consciente. Este projeto apresenta-se como um contributo para a melhoria do carregamento de Veículos Elétricos (VE) com impacto positivo desde o momento da ligação ao Distribuidor de Energia, até ao final do carregamento do VE. As soluções de carregamento elétrico para o mercado residencial implicam, em muitas situações, um aumento da potência contratada para responder a um ciclo de carregamento ineficiente que se inicia quando o carregador é ligado e termina quando se atinge a carga máxima da bateria do VE. O aumento da potência contratada nem sempre é a melhor solução para um carregamento mais rápido e eficiente.

Quando estão disponíveis pontos únicos de ligação à rede elétrica, comuns a um grupo de condóminos, torna-se mais difícil adotar soluções de mobilidade elétrica. Gerir a potência contratada, otimizar o tempo de carregamento, controlar os custos de acordo com o utilizador ligado, ou mesmo equilibrar o carregamento dos VE de acordo com a fonte de energia num determinado momento, são desafios que a solução apresentada pretende resolver.

Com foco no mercado residencial, esta solução é adequada tanto para pontos de ligação de uso único como para ligações partilhadas, muito frequentes em edifícios de apartamentos sem garagem fechada ou que, por outras razões, partilham as ligações elétricas disponíveis à rede. A arquitetura apresentada permite a utilização de um ou vários pontos de carregamento comuns, através da aplicação de uma rede mesh de carregadores inteligentes orquestrados por um gateway residencial.

Acknowledgements

I would like to express my deepest gratitude to my advisor, Professor Filipe Caldeira, whose sincerity and encouragement I will never forget. Prof. Caldeira has been an inspiration as I hurdled through the path of this Masters degree. He is the true definition of a leader and the ultimate role model. This thesis would not have been possible without Prof. Filipe Cardoso, whose guidance from the initial steps in Electric Vehicles research enabled me to develop an understanding of the subject. I am thankful for the extraordinary experiences he arranged for me and for providing opportunities for me to grow professionally. It is an honor to learn from Prof. Filipe Caldeira and Prof. Filipe Cardoso.

I am grateful for my family, my wife Daniela, and my daughters Helena e Carolina whose constant love and support keep me motivated and confident. My accomplishments and success are because they believed in me.

Foreword

The work described in this thesis was developed in the scope of the IEVCC project involving the following entities, Polytechnic of Viseu, CISEd and INESC Coimbra. During the project's development, the institutional partners provided the development boards like ESP-32 and other hardware tools needed for development and testing. This support made the project easier to set up and gave the testing environment the required resources.

The work done during this thesis resulted in the following publications co-authored by the candidate (appendices B, C and D):

- F. Cardoso, J. Rosado, M. Silva, C. J. C. Teixeira, C. I. F. Agreira, F. Caldeira, P. Baptista, F. Barreto, and P. G. Pereirinha. Intelligent electric vehicle charging controller. In *2021 IEEE Vehicle Power and Propulsion Conference (VPPC)*, pages 1–5, 2021. doi: 10.1109/VPPC53923.2021.9699236
- P. Baptista, J. Rosado, M. Silva, F. Caldeira, and F. Cardoso. Ievcc - a mesh managed network for electric vehicle charging. In *2022 IEEE Vehicle Power and Propulsion Conference (VPPC)*, 2022a. doi: Awaiting-publication
- P. Baptista, J. Rosado, M. Silva, F. Caldeira, and F. Cardoso. Design of data management service platform for intelligent electric vehicle charging controller - multi-charger model. In *Energy Informatics.Academy Conference 2022*, 2022b. doi: Awaiting-publication

The candidate also co-authored the following publications that resulted from the work done during the curricular units of the master's program.

- P. Baptista, M. Rodrigues, M. Costa, P. Martins, and M. Abbasi. Shared dock-less vehicle location distribution data-set visualizer. In *2021 16th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–4, 2021. doi: 10.23919/CISTI52073.2021.9476220
- M. Costa, M. Rodrigues, P. Baptista, J. Henriques, I. M. Pires, C. Wanzeller, and F. Caldeira. Covid-19 next day trend forecast. In I. M. Pires, S. Spinsante, E. Zdravevski, and P. Lameski, editors, *Smart Objects and Technologies for Social Good*, pages 44–50, Cham, 2021. Springer International Publishing. ISBN 978-3-030-91421-9
- M. Costa, M. Rodrigues, P. Baptista, C. Wanzeller, P. Martins, and M. Abbasi. Database encryption performance impact on postgresql and mongodb. In J. L. Reis, E. P. López, L. Moutinho, and J. P. M. d. Santos, editors, *Marketing and Smart Technologies*, pages 121–127, Singapore, 2022. Springer Nature Singapore. ISBN 978-981-16-9268-0

Contents

1	Introduction	1
1.1	Background	1
1.2	Problem statement	2
1.3	Aims and objectives	2
1.4	Selected approach	3
1.5	Organisation of the report	3
2	Electric mobility and charging	4
2.1	Electric vehicles	4
2.1.1	Electric vehicle disorder	4
2.2	Electric vehicle charging	5
2.2.1	Charging optimisation	5
2.2.2	Charging monitoring	6
2.2.3	Electric Vehicle Supply Equipment	7
2.3	Electrical Grid	7
2.3.1	Domestic Grid	8
2.4	Mesh Networks	8
2.4.1	Constrained Application Protocol (CoAP)	10
2.4.2	Message Queuing Telemetry Transport (MQTT)	10
2.4.3	Advanced Message Queuing Protocol (AMQP)	11
2.5	Existing "smart" EV chargers in market	12
2.5.1	Rolec EV Charging - VendElectric	12
2.5.2	EVBox	13
2.5.3	EvoCharge	14
3	Contribution to IEVCC - Charging improvement	16
3.1	Waterfall Model	16
3.2	Scrum	17
3.3	Planning	18
3.4	Intelligent EVSE Prototype	19
3.4.1	Architecture	20
3.4.2	Tech Stack	24
3.4.3	Message Protocol	25
3.4.4	Intelligent Electric Vehicle Charging Controller	26
3.4.5	IEVCC Broker/Manager	28
4	Discussion	32

CONTENTS

vii

5 Conclusions and Future Work	36
5.1 Conclusions	36
5.2 Future Work	37
Appendices	42
A Gantt diagram	42
B Vehicular Power and Propulsion 2021 (VPPC 2021) Article	44
C Vehicular Power and Propulsion 2022 (VPPC 2022) Article	50
D Energy Informatics.Academy Conference 2022 (EI.A 2022) Article	56

List of Figures

1	Rolec EV Charging - VendElectric (Rolec, 2020)	13
2	EVBox - BusinessLine (EVBox, 2022)	14
3	EvoCharge Spec Sheet (EvoCharge, 2022)	15
4	Waterfall Model (Marjerison, 2013).	16
5	Scrum process (von Wangenheim et al., 2013).	17
6	IEVCC system diagrams (Cardoso et al., 2021).	20
7	Fog Architecture (Baptista et al., 2022b).	21
8	Multi-Charger Model (Baptista et al., 2022b).	22
9	Proposed Architecture (Baptista et al., 2022b).	23
10	Tech Stack (Baptista et al., 2022b).	25
11	IEVCC Prototype (Cardoso et al., 2022)	27
12	Consumption monitor hardware/Energy Meter	27
13	Protocol time-lapse (Baptista et al., 2022a).	28
14	Protocol message examples.	29
15	Protocol message codes.	29
16	Client Factory UML.	30
17	Tests sample.	31
18	Software simulators.	32
19	Standalone simulation.	33
20	Multi-charger simulation.	34
21	Gantt diagram	43

List of Tables

2.1	Protocol objectives and related features (Wytrebowicz et al., 2021)	9
2.2	Messaging features (Wytrebowicz et al., 2021)	9
3.1	Transport features (Wytrebowicz et al., 2021)	26

List of Abbreviations

AC	Alternating Current
AMQP	Advanced Message Queuing Protocol
BEV	Battery Electric Vehicle
CNCF	Cloud Native Computing Foundation
CoAP	Constrained Application Protocol
DC	Direct Current
DER	Distributed Energy Resources
DSO	Distribution System Operators
ED	Energy Distributor
EV	Electric Vehicle
EVCS	Electric Vehicle Charging Station
EVSE	Electric Vehicle Supply Equipment
EVSP	Electric Vehicle Service Provider
FOSS	Free and Open Source Software
HEV	Hybrid Electric Vehicles
IEVCC	Intelligent Electric Vehicle Charging Controller
IoT	Internet of Things
LV	Low-voltage
M2M	Machine to Machine
MQTT	Message Queuing Telemetry Transport
PHEV	Plug-in Hybrid Vehicles
PV	Photovoltaic
QoS	Quality of Service
RCBO	Residual Current Breaker with Over-Current
SoC	State of Charge
V2G	Vehicle-to-grid

Chapter 1

Introduction

Electric vehicles (EV) are environmentally friendly since they do not emit any gas directly into the atmosphere, require less maintenance and operating expenses, and offer a quieter driving experience (Dubey and Santoso, 2015) (Aghabali et al., 2021). These are the primary advantages of EV, which are gaining more and more traction as the technology evolves. Even though they presently represent only 2.7% of global sales, according to the Bloomberg report by Henbest et al. (2020), the tendency is for them to grow. Actual projections refer that by 2025, EV will account for 10% of worldwide passenger vehicle sales, rising to 28% in 2030 and 58% in 2040, respectively.

According to an analysis conducted by the Association of Electric Vehicles Users (UVE) for Portugal, the sale of EV increased by 80% in November 2020 compared to the same month in 2019 (Nascimento, 2020).

As automakers ramp up production to meet demand, EV charging infrastructure will need to scale in response to the increasing number of EVs produced for consumer and fleet adoption over the next few years.

Scalability is one of today's most significant challenges facing the electric vehicle charging business. To handle exponential market expansion and a diverse range of charging use-cases, the EV charging infrastructure must be scalable. Electric Vehicle Service Providers (EVSP) must be able to expand their networks while simultaneously expanding their offers and delivering a seamless charging experience to drivers. In this context, developing a back-end system capable of increasing charger availability by remotely assessing and correcting faults will be required.

An EV ready charging infrastructure should also feature flexible billing options to accommodate expansion and the introduction of newer business models. Depending on the type of charging location, such as private residences, multi-family housing complexes, workplaces, commercial and industrial sites, petrol stations, retail establishments, municipalities, and public charging stations, there are different standards to meet. EVSP may serve a wide range of locations and business models by using a modular back-end system that also supports open standards for the grid and intra-network communication, as well as APIs for legacy applications, among other features.

1.1 Background

In the next five to ten years, the use of EV in Portugal is predicted to expand consistently. More and more people are moving toward EV as a way to reduce their carbon footprints and noise pollution while increasing their transportation energy efficiency.

The path to sustainable electric transportation depends on many factors, including reducing emissions, increasing energy efficiency, and adopting conscientious use.

Electric Vehicles face several obstacles as they seek to acquire acceptance globally. While they are advantageous regarding sustainability and environmental stewardship, they include inevitable trade-offs. When most people consider EV, they anticipate short ranges, high costs, and even the need to install a charging station at home. Certain preconceptions likely hinder some people from considering an electric car at all. While some of these perceptions are incorrect, others are not. Nonetheless, manufacturers of EV experience many of the same technological and economic hurdles as manufacturers of conventional cars. Several significant obstacles are to overcome, such as reliability, certification, and functional safety. Additionally, there is also considerable pressure to reduce costs.

Although battery and micro-controller optimization is still a priority, the current study focuses on the next big challenge: charging infrastructure and charging process for residential customers.

1.2 Problem statement

In many cases, the EV charging solutions for the home market implies an increase in the contracted power to respond to an inefficient charging cycle that begins when the charger is connected and stops when the EV battery's maximum charge is reached. Increased contracted power is not necessarily the most effective approach for charging faster and more efficiently. When the number of available power grid connections is limited and shared by a large number of tenants, it becomes difficult to implement electric charging solutions and solve challenges such as controlling expenses by connected user, optimising charging time, and even balancing the charging of EV based on the source of energy available at a given time.

Occasionally, charging an EV at an underground car park outlet can cause issues. The landlord may object, as charging an electric car for several hours continuously places a significant burden on the building's electric system. And if neighbors have the same idea, there's the risk of blowing a fuse and wreaking havoc throughout the complex.

Another factor to consider is that energy consumption in subterranean parking garages is not metered by user. As a result, all tenants in the building will be responsible for the additional electrical costs incurred as a result of this practice.

Accordingly Shepelev et al. (2013), by communicating, dynamically monitoring, and controlling the charging process, the future smart grid promises to ease most of the referred issues. This will be accomplished primarily by arranging charging and discharging activities so that the grid load is kept within safe limits while still serving the needs of EV customers.

1.3 Aims and objectives

With a specific focus on the residential market, current work is designed for both single-use grid connections and shared connections, which are extremely common in apartment buildings that do not have a closed garage or that, for other reasons, must share the available electrical connections to the power grid for EV charging.

Aims:

- Create a mesh network of intelligent chargers, orchestrated by a home gateway (Manager/Broker), and allow the use of one or more common charging stations at the same

time.

- Research about scalability and tolerance fault applied to proposed architecture.
- The charging process can be optimised to prioritise the use of "green" energy and adapt to the number of EV connected simultaneously, preventing damage to the electrical infrastructure.

Objectives:

- Propose a software architecture capable of implementing the solutions for the challenges stated in 1.2.
- Create a Manager/Broker solution that acts as a gateway for the compatible Intelligent Electric Vehicle Charging Controller (IEVCC).
- Secure Message Queuing Telemetry Transport (MQTT) broker implementation to enable secure data transfer.
- Implement the full tech stack to manage charging and store user data.

1.4 Selected approach

The architectural proposal shall be conceived and reference all needed entities as part of the solution.

Considering the defined objectives (1.3) and the lack of solutions currently available, it's essential to identify all the necessary applications that will take place on the solution "tech stack".

Focusing on Cloud Native Computing Foundation (CNCf) projects and adopting Free and Open Source Software (FOSS) principles enables faster component development and community validation.

Due to time restrictions and the number of CNCf projects available, this work will not focus on benchmarking or comparing possible solutions for the "tech stack". Decisions will be made considering the flexibility, ease of use, and speed of implementation.

1.5 Organisation of the report

This document comprises five chapters. The first chapter, this introduction, begins with a thematic context for the project's development, followed by the motivation for its creation. Chapter 2 focuses on the literature review, which is crucial in defining the target subject's conceptual boundaries. Chapter 3 discusses the methodology for setting up the solution. The experimental results are reported and discussed in chapter 4. In chapter 5, final observations and conclusions are presented.

Chapter 2

Electric mobility and charging

The modern era of electric mobility still takes small steps on the way to the mass implementation of the EV (Electric Vehicle). This chapter focuses on the current state of electric mobility and the solutions for optimising EV charging.

2.1 Electric vehicles

There is a crisis in the traditional automobile industry, which consists of vehicles powered by internal combustion engines, whether diesel or gasoline. Internal combustion engine vehicles face a 41.2% decrease in sales by 2020 (Nascimento, 2020).

The same report by Nascimento (2020), about sales of 100% EV, states that there has been a considerable gain in nine of the eleven months of 2020, and sales have increased steadily since the month of June, with increases in the last five months in the tens of percentage points. The market share of electric vehicles (BEV+PHEV) reached an incredible 12.6% for the entire year 2020, which is unprecedented in the history of the automobile industry.

In comparison to hybrid electric vehicles (HEV) or plug-in hybrid vehicles (PHEV), automobiles that rely entirely on battery energy for transmission are referred to as battery electric vehicles (BEV). BEV may operate only on the energy stored in their batteries, and their range varies according to battery capacity. Additionally, the battery is the single largest component of the BEV production cost, and its capacity becomes a significant market differentiation. In general, regular BEV can go between 100 and 250 kilometres on a single charge, but the top versions can travel much further, between 300 and 500 kilometres (Stamatescu et al., 2020). In recent years, the automobile industry's research and development operations have focused on the creation of highly efficient and emission-free modes of transportation. With this in mind, electric and hybrid vehicles look to be the greatest alternatives for replacing vehicles powered by internal combustion engines. However, there are a few key considerations that must be taken into account in order for EV to become a commercial reality. The most significant obstacle preventing mainstream adoption of electric vehicles is range anxiety. Charging time reduction is also seen as a crucial goal in making EV more accessible to a wider audience (Khan et al., 2019).

2.1.1 Electric vehicle disorder

Accordingly Moghaddam et al. (2018), it's essential to cut down on the time spent waiting at public charging stations. A smart charging approach can make a significant contribution to the efficient use of energy grid resources. Recognising the importance of smart charging in

the context of public charging networks, researchers have recently explored and presented a variety of smart charging solutions aimed at reducing range anxiety, charging time and charging efficiency. However, there is still a gap in research because none of the previous studies take into account numerous charging and dynamic pricing options at charging stations. Presently, a charging station (like a regular petrol station) can offer multiple charging alternatives with real-time pricing. Every charging station's current state (e.g., the available number of sockets, queue status, price, etc.) can be collected using a smart grid. The grid can then deliver this information to the particular charging station user in real-time. This data can be used to generate a route to the destination that will save time and money in the billing process. Smart charging, coordinated charging, and regulated charging are all terms used to describe the opportunity to improve the EV charging process by adjusting EV charging over time. The use of intelligent charging algorithms has been explored for a wide variety of purposes. Using vehicle-to-grid (V2G) technology, the vast majority of the proposed charging algorithms perform economic enhancement of EV charging. These algorithms aim to reduce the charging costs of EV owners and aggregators in the case of unidirectional charging, or to maximise their economic benefits in the case of bidirectional charging. Alternative electric vehicle smart charging algorithms are being developed with the goal of reducing CO₂ emissions, increasing self-consumption of photovoltaic (PV) solar energy, or alleviating congestion and power quality issues in low-voltage (LV) grids. Smart charging algorithms have been implemented and tested in real-world scenarios as a result of the research and development activities described above (Brinkel et al., 2020).

2.2 Electric vehicle charging

Batteries for electric vehicles are classified as either batteries or fuel cells. Among them, fuel cells are used sporadically to power electric vehicles. The majority of electric vehicles are powered by batteries. Its advantages include a high power density, a high energy output, and a long life. From a battery charging safety perspective, we may consider dividing electric car charging safety into four categories: power grid side safety, charging equipment side safety, vehicle side safety, and platform side system safety. Through the power grid and charging side, two angles are used to conduct extensive analysis, which results in a safe and faster charge (Linru et al., 2020). The same authors also refer that the EV is linked to the distribution side of the power grid, which has random and extensive access. Each vehicle has a different performance and charging mode, and the dispatching centre management will have different characteristics. Considering the future, the cost of owning an EV will be significant. A hierarchical architecture is used to build the integrated structure of intelligent charging, charging, and power monitoring in order to access any EV at random on a large scale.

2.2.1 Charging optimisation

The optimisation of the EV charging process is becoming more important as the number of EV charging transactions on low-voltage grids grows. Most EV now charge in an uncontrolled way, so the vehicle begins charging immediately after connecting to a charging station and continues charging until its charging requirement is reached. Because a substantial percentage of EV begin charging in the late hours, this charging approach is widely seen as unfavorable. This generates grid congestion and power quality issues since the EV demand peak coincides with the nighttime energy demand peak of residences. Furthermore, uncontrolled charging is not always cost-effective or efficient in terms of CO₂ emissions (Brinkel et al., 2020). Accordingly Nour et al. (2019), the benefits of EV owners are maximised by regulating EV

charging based on the price of power. By controlling the EV charging using a fuzzy controller based on electricity price and State of Charge (SoC) inputs and a charging power output, the authors concluded that smart charging reduced the maximum power requirement by 100 kVA, or 20% less than uncontrolled charging. The apparent power demand measured during the day appears to have been the same as in the base instance. When compared to uncontrolled charging, smart charging reduces the maximum transformer load by 20% during the day. The maximum transformer loading observed for smart charging was 80%. When compared to uncontrolled charging, smart charging reduces the cable's maximum load by more than 10%. When compared to uncontrolled charging, smart charging improved the voltage profile, and the lowest voltage recorded was greater than 0.96 p.u..

Almost all EV optimisation models optimise the charging behaviour of individual EV transactions, as each charging transaction has its own plug-in and plug-out times, charging requirements, and maximum charging rate. With the widespread adoption of EV, optimising individual EV transactions may impose a considerable computing cost, as each individual EV transaction introduces a new set of variables into the optimisation problem (Brinkel et al., 2020).

On a broad scale, the random charging of EV does seem to increase the conventional load on the power grid. If EV charging is implemented and an effective charging control technique is developed to charge the EV at a relatively low load on the power grid, this will contribute to peak reduction and valley filling (Linru et al., 2020).

2.2.2 Charging monitoring

While the majority of studies has concentrated on optimising car charging behaviour, communication networks and the underlying communication infrastructures have received less attention and discussion. Sensor nodes, metering devices, and dependable communication infrastructures are required as key parts in EV systems in order to monitor, optimise, and coordinate the integration of EV into the power grid (Ahmed et al., 2020).

There are two types of remote monitoring systems for Electric Vehicle Charging Station (EVCS), centralised and decentralised. To monitor the charging operation and to limit voltage variations, the centralised architecture requires a direct connection between the unit control centre and individual charging stations. The unit control centre processes and stores data obtained from charging stations. In the decentralised architecture, charging operations and decisions are made at the parking lot level (Ahmed et al., 2020).

The same authors (Ahmed et al., 2020) categorised charging stations as either blind EVCS or networked EVCS. The blind EVCS system is equipped with the essential charging feature and enables low-voltage EV charging. However, there is no exchange of external monitoring or control information with the grid side. In comparison to blind EVCS, the networked EVCS system provides more functionality. It communicates monitoring data (status information and analogue measurements) on a periodic basis to a local controller located in a smart dwelling, a smart building, a parking lot, or a charging station. EVCS exchanges information with EV during the charging service, such as charging mode, metering, and payment. Additionally, the maintenance system operator continues to collect, store, and analyse monitoring data from every EVCS.

Cloud monitoring and administration enables customers to access necessary data at any time and from any location. Cloud computing allows users to access resources remotely, resulting in significant cost savings. The enhanced flexibility enables EV users to scale up or down quickly and easily, while connectivity offers efficient and effective field assistance and maintenance of each charger (Saqib et al., 2017).

According to Ahmed et al. (2020), implementing scheduling and control strategies for EV charging coordination requires proper communication infrastructures and bidirectional communication between EV subsystems. However, the study majority disregards the underlying communication infrastructure as a component of the EV system, while some assume ideal communication networks. These assumptions are incorrect, as data loss and communication delays are an unavoidable part of system operation and may affect system performance.

2.2.3 Electric Vehicle Supply Equipment

Typically, an Electric Vehicle Supply Equipment (EVSE) designed for commercial use is made of three distinct levels of charging power and type. They are classified as follows by SAE EV AC Charging Power Levels. Alternating Current (AC) Level 1 equipment, frequently abbreviated as Level 1, provides charging via a 120 volt AC connector. Level 1 charging is typically used when just a 120V outlet (such as a socket) is available. The majority of electric vehicles come equipped with a Level 1 cord as a standard. The amount of energy required to completely charge an electric vehicle using a Level 1 socket is comparable to the amount needed by a toaster (Stamatescu et al., 2020). Accordingly the same author, AC Level 2 equipment, or Level 2, charges from a 208V to 240V AC plug. Level 2 charging is compatible with both commercial and residential applications. The amount of energy required to completely charge an electric vehicle at a Level 2 charging station is comparable to the amount consumed by a clothes dryer. Direct Current (DC) Fast Charging charges the automobile battery directly, as opposed to AC charging, which runs via the EV internal power electronics systems for conversion. This allows rapid charging along high-traffic areas that are often reserved for commercial charging. The amount of energy required to fully charge an electric vehicle at a DC charging point is comparable to the amount consumed by five to ten central air conditioners.

Currently, the majority of charging cycles occur at owners' dwellings, where vehicles are parked and charged overnight, but this is predicted to change as more public charging alternatives become available. The share of home charging in the European Union is expected to fall from around 75% in 2020 to 40% by 2030 as more middle- and lower-income families without access to home charging purchase EV. Renters, who make up an increasing proportion of the US population, are unable to home charge since landlords have little motivation to invest in EVSE. The inability to charge is a significant barrier to acquire an EV, making nonresidential EVSE development crucial for not only long-distance travel, but also for people who are unable to charge effectively at home. A significant majority of nonresidential EVSE is located on existing retail properties. Due to the longer refuelling intervals associated with EV compared to gas-powered vehicles, the ability to shop while refuelling is especially appealing. Charging stations have already been deployed in a number of Walmart and Target shops, as well as other retail locations. Retailers have an incentive to develop EVSE, as 89% of electric vehicle drivers often make a purchase while charging at a retail site (Gillera et al., 2021).

2.3 Electrical Grid

With the progress of human civilisation and economic expansion, the need for electricity is increasing rapidly, resulting in two significant worldwide concerns: the energy crisis and environmental degradation. As a result, more distributed energy resources (DER) are being integrated into power systems, including renewable energy sources such as solar and wind, energy storage devices, and electric vehicles (Zhong et al., 2019). The residential sector accounts for a significant portion of electricity demand. Consumers tend to use an increasing

number of electrical equipment as their lifestyles improve and their income level increases. The efficiency of electrical appliances has been increased. However, as the number of electrical devices expanded alongside the rise in living standards, electricity usage increased. Load shifting programmes are used to reduce peak consumption by transferring flexible loads from peak to off-peak hours. Direct load control and indirect load control are two methods for encouraging consumers to shift their electrical load. While indirect load programmes rely on consumers changing their consumption habits on their own, direct load programmes require users to purchase smart technology and communication equipment (Ayan and Turkay, 2018).

2.3.1 Domestic Grid

Gradually, communication and information networks are being integrated into power systems. Naturally, the additional communication channels are expected to provide the required infrastructure for all power system participants to cooperate, even at the lowest levels of control. However, this usual scenario may raise severe questions regarding reliability and security (Zhong et al., 2019).

Peak load is a significant issue in the electricity grid. In the past, this issue was largely resolved through supply-side control. However, demand-side management sources have gained prominence in recent years as a result of economic and environmental restrictions. Usually, countries can maintain reliable electricity supply to customers by either increasing network capacity (supply side management) or by reducing peak electricity demand on the network (demand side management). Reducing electricity consumption through demand side management, rather than increasing network capacity, provides operational and financial benefits to countries. Demand response can play a significant role in reducing peak loads in the residential market (Ayan and Turkay, 2018). Accordingly Calearo et al. (2019), the potential load created by home charging of EV is not as large as is frequently assumed. Because most EV do not require daily charging, simultaneous charging of more than 45% of EV with single-phase chargers is never possible. Additionally, when three-phase chargers are used, the charging power is tripled, but the total peak increases by only 50% due to the simultaneous charging being lowered to 25%.

It may be inferred that, in the short term, with few EV, the Distribution System Operators (DSO) should prioritise grid balancing. Thus, even if EV are evenly distributed across three phases, the discrepancy in charging behaviour can increase grid instabilities. On the other hand, in the long run, high EV penetration combined with rising electricity usage could result in congestion being the primary concern in distribution power systems (Calearo et al., 2019).

2.4 Mesh Networks

Modern WiFi networks consist of an access point, or base station, which is typically located in a central area and communicates with each user's device. Mesh networks enable peer-to-peer communication and distribute devices, or nodes, throughout the network. Each of these devices connects with the others, forming a "mesh" that enables a reliable signal to be transmitted over a greater distance. Devices then connect to the nearest node rather than the central base station, establishing a more reliable network connection. Mesh networking is an efficient approach to coordinate data transmission between Internet of Things (IoT) devices. It is great for transporting little data packages and enables you to connect additional nodes while maintaining a high level of network speed (Anh, 2021). The Internet of Things (IoT) is a network that connects computers, mechanical and digital machines, objects, animals,

and even people. Each of these things has a unique identifier and can send and receive data over a network without having to interact directly with people or computers. Some common Internet protocols, such as HTTP, were initially used to connect the first Internet of Things devices, however, newer Internet protocols that are lighter and better able to fit the constrained requirements imposed by IoT environments have emerged. These newer Internet protocols include Constrained Application Protocol (CoAP), Message Queuing Telemetry Transport (MQTT), Advanced Message Queuing Protocol (AMQP) amongst others (Iglesias-Urquia et al., 2017) (Table 2.1 and Table 2.2).

Table 2.1: Protocol objectives and related features (Wytrowski et al., 2021)

Protocol	Main Purpose	Defined Roles	Architectures	Discovery of	Real Time	Specific Features
MQTT	M2M & IoT small code footprint limited bandwidth and high latency networks	publisher, subscriber, broker	client-broker-client	topics	partial	
MQTT-SN	constrained devices	publisher, subscriber, broker	client-broker-client	topics, gateway	partial	
CoAP	constrained devices, lossy networks	server, client, support pub/sub mechanism	p2p, client-server, master-slave	resources	yes	RESTful
AMQP	corporate environments	publisher, subscriber, broker	client-m_router-client, client-broker-client		partial	multiple links in one connection

Table 2.2: Messaging features (Wytrowski et al., 2021)

Protocol	Payload Size Limits	Payload Data Representation	Labelling	Metadata	Transaction Support
MQTT	256 MiB	UTF-8 text, unspecified bytes	yes	yes	
MQTT-SN	64 KiB	unspecified bytes	yes		
CoAP	40 B–1 KiB (without IP fragmentation), 1 MiB–1 GiB with block-wise transfer	internet media type		yes	
AMQP	defined by end-points	internet media type	yes	yes	yes

2.4.1 Constrained Application Protocol (CoAP)

The Constrained Application Protocol (CoAP) is a web transfer protocol that was developed specifically for devices and networks with limited resources. It functions quite similarly to HTTP due to the fact that it is built on top of UDP and it follows the REST concept. It uses a subset of HTTP's request verbs, such as GET, POST, PUT, and DELETE, as well as response codes and content-formats, even though there is an extra content-format in this case. The port number that is used by default for CoAP is 5683, while the port number for secure CoAP is 5684. The following is an example of the format of a CoAP URI: `coap://host[:port]/[path][?query]` (Iglesias-Urkiá et al., 2017). CoAP can be thought of as a version of the HTTP protocol that is made to work with devices that have limited resources. The URIs point to resources that the server provides. The client talks to a server with a known, predefined address, but in local networks, the client can also use a mechanism for dynamic server discovery to find the server. Both a server and a client can be done by the same device. RFC 7641 defines an extension as something that lets the client subscribe to updates to server resources over time. This is the best-effort publish-subscribe communication pattern. CoAP can communicate in a few different ways, such as peer-to-peer, client-server, and master-slave. It can also be used in architectural designs that are more complicated (Wytrebowicz et al., 2021). In order for the user to access the accessible resources, they will first need to be familiar with the route, and then they will be able to submit queries to the server using the URI. CoAP does not ensure the delivery of the packets, and as a result, it provides two primary categories of messages that differ in whether or not they require acknowledgement: confirmable and non-confirmable messages. The other two forms, known as Reset and ACK, also have the ability to piggyback data. Due to the fact that the underlying UDP protocol requires ordered and assured message delivery, CoAP is unable to employ TLS to secure the connections since it was not designed to do so. UDP's alternative to TLS is called DTLS, using it comes at the expense of losing some of the benefits of using UDP, which stem from the fact that it does not require an open connection (Iglesias-Urkiá et al., 2017). CoAP makes it easy to connect physical devices to Web content while meeting requirements like support for multicast, low signaling overhead, and simplicity for environments with limited network resources. In a constrained network environment, most of the devices are embedded and need much less processing power, memory, and power supply than Internet devices. More specifically, constrained nodes often have 8-bit microcontrollers with small amounts of ROM and RAM, while constrained networks like IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs) often have high packet error rates and a typical throughput of 10's of kbit/s. In a WoT environment, it is very important to meet the requirements of multicast support, low signaling overhead, and simplicity. Also, constrained networks in a WoT environment usually have a limit on the size of packets, may lose a lot of packets, and may have a lot of devices running in sleep mode. In a duty-cycled network, sensor nodes have four different ways to work: they can send, receive, do nothing, or go to sleep. During sleep states, the sensor is turned off and a low-power timer is turned on to wake it up later. So, it can only use a very small amount of the energy that the active mode uses (Chun et al., 2015).

2.4.2 Message Queuing Telemetry Transport (MQTT)

MQTT is an open standard maintained by OASIS. Recently, MQTT has been sold as a way to communicate between machines - Machine to Machine (M2M). This is because the Internet of Things is based on communication between machines. Specifically, the IoT needs to be able to work with low-powered devices in order to be sustainable, and MQTT meets all of the IoT needs in that way. This publish-subscribe protocol is great for connecting low-

bandwidth devices with limited CPUs and little overhead. It's a good choice for connecting and communicating between small groups of low-energy devices and a central hub. MQTT can be used for different types of performance and message transmission, which is important because many devices used for IoT are simple and don't have a lot of performance capacity. MQTT is an important part of making IoT projects easier to set up in terms of technical specifications while still allowing devices, servers, and applications to connect with each other. When considering low-power wireless devices with unstable connections, this might be a good solution (Anh, 2021). TCP/IP is the underlying protocol that MQTT uses. The Internet Assigned Numbers Authority has designated port 1883 as the one to use for this service (IANA). Port 8883 is utilized whenever MQTT is utilized over SSL. Mosquitto, Mosca, and HiveMQ are just a few of the local brokers that support MQTT. There are a variety of public brokers, some of which are iot.eclipse.org, test.mosquitto.org, broker.hivemq.com, www.cloudmqtt.com, and mqtt.dioty.co (Masdani and Darlis, 2018). There are three ways to send data based on QoS. It is a best-effort delivery with a QoS0 of "at most once." QoS1 - at least once - duplicates can occur. QoS2: a reliable delivery exactly once. More PDUs are exchanged as the QoS type goes up. A subscription can be durable or non-durable. The durable subscription tells the broker to keep messages for a certain client even when it's not connected, like when it's sleeping. A publisher can also tell the broker to keep messages even if no one has signed up to get them for that topic. In this case, the next message will overwrite the previous one, and a new subscriber will get the most recent value of the topic. MQTT can be used by any IoT platform for devices that are not limited. The upper layer is in charge of how to deal with network availability and how much it might cost to send data. Due to the rare transfer of short data, some limited devices can support TCP and messages that are not compressed. Even if an IoT device is just a publisher, it can still be used to get data and look at notifications and alarms. An IoT device that is just a subscriber can be used to carry out commands that have been sent. An IoT device that can act as both a publisher and a subscriber can be used to find and set up other devices, ask for data, and remotely control other devices. MQTT is not good for peer-to-peer communication between devices that are close together, though. MQTT is used in a wide range of Internet of Things (IoT) apps. Industrial MQTT deployments have been seen with hundreds or thousands of topics and subtopics and the same number of active terminal nodes. Facebook Messenger is a good example of how MQTT is used (Wytrebowicz et al., 2021).

2.4.3 Advanced Message Queuing Protocol (AMQP)

AMQP (Advanced Message Queuing Protocol) is managed by OASIS and standardized by ISO/IEC. An AMQP communication node can act as a producer, consumer, or message queue. A process can have more than one of these nodes, which makes it possible to build very complex network parts. The process can act as a client, a server, a broker, or a router. Depending on what the application needs, the router can send a message right away or store it. A peer-to-peer network is made up of connected terminal processes. The protocol's main way of communicating is client-message-router-client. But in place of the message router, an implementation can only offer simple broker functionality. The AMQP standard describes how to encode simple and complex data types, how to send messages, and how to handle complex transport (i.e., the structure and semantics of messages carried by the transport mechanism). Also, the AMQP specification says how TLS and SASL are used to authenticate and encrypt communications. The encoding lets binary data be represented and also lets data types be marked to meet the needs of applications. The transport mechanism makes it possible for two processes to connect using AMQP and open several communication sessions at the same

time. In one session, you can set up as many one-way links as you want. The link's job is to move frames. AMQP can break up a large message into pieces and send them in separate frames. The limits of the message and frame sizes are shown on the communication end points. Flow parameters are set for a link. For example, message delivery guarantees (QoS) like "at most, at least, and exactly once" are flow parameters. Also, filtering functions can be linked to the sending node to stop messages from being sent that don't meet the criteria. In the case of a broken connection, the links are remade, and the state of the messages that have been exchanged is kept. AMQP may send a message with a complicated structure. The structure must have a field for the application data. Header, transport annotations, message annotations, standard properties, application-defined properties, and footer are the fields that you can choose not to use. From the sender to the receiver, application data and its properties stay the same. Message routers can use and change the rest of the fields. The standard sets out a large number of elements that can be used in the above fields. These things can change how messages are handled in intermediate nodes, set the priority and lifetime of messages, change how reliable their transmission is and make sure the necessary security is in place, among other things (Wytrowski et al., 2021).

2.5 Existing "smart" EV chargers in market

When we write about a "smart EV charger," we're referring to a charging station that can do tasks like communicate with other wireless electronic devices, accept payments from established systems, connect to a smart power grid, and dynamically change how much power is sent to the EV.

There are a variety of charging equipment options available for electric vehicles. The following "smart" EVSE solutions are some of the available options which offer multiple features that are identical to our target solution.

2.5.1 Rolec EV Charging - VendElectric

Rolec Services designs, manufactures, installs, and supplies outdoor electrical equipment worldwide.

The firm employs 120 people in its purpose-built offices, showrooms, warehouse, and industrial complex and has agents in 40 countries.

Rolec Services focuses on EV-charging, Marina Services, and Caravan Hook-Ups.

For this work, we focus on EV-Charging features, like managing single or more charge points with the ability to support private network charging (staff members, fleet drivers, etc.), ensure that charge points are protected from unauthorised use. Possibility to add more charging stations for electric vehicles to your infrastructure as demand develops. Full visibility of real-time charge point activity, monitoring, and energy consumption at the push of a mouse, allowing you to comprehend charging behaviours and determine where more infrastructure is necessary. Accessing and exporting all historical data and reports is simple. The author Rolec (2020) also states it's ideal for sites requiring several EV charging stations where the total demand may exceed the available electrical supply and customers may opt to include load control mechanism which guarantees that all charging stations are accessible. Although load control is announced, there's no available information on how it's implemented. The Figure 1 provides an overview of all the features available.

	Essential	Premium
OLEV GRANT ELIGIBLE:		
WCS (Workplace Charging Scheme)	✓*	✓
USER ACCESS:		
Plug & Play Charging	✓	✓
Mobile Phone App		✓
RFID Card/Fob		✓
HOST MANAGEMENT:		
Unlimited Back-Office Platform Access		✓
Chargepoint Management Dashboard		✓
Visibility Of Your Entire Chargepoint Network		✓
Real-Time, Historical & Analytical Feedback		✓
Customisable Tariffs		✓
Optional Revenue Stream		✓
Private, Public & Fleet Charging Options		✓
Exportable Usage, Revenue & BIK (Benefit In Kind) Data		✓
Automatic Maintenance Response Notifications		✓
Electrical Load Management (Optional)		✓
	£50 per 3 Years (each socket)	£135 per 3 Years (each socket)

* PLEASE NOTE: The VendElectric Essential management plan provides the host with an automated quarterly chargepoint consumption report as required in order to meet eligibility for the OLEV Workplace Charging Scheme Grant (WCS).

Figure 1: Rolec EV Charging - VendElectric (Rolec, 2020)

2.5.2 EVBox

EVBox (2022) provides flexible and scalable electric car charging solutions to organisations. Current offers target residential, business, and rapid charging stations as well as scalable charging management software. Customers and partners have access to more than 400,000 charging ports, and EVBox Everon does more than 1.4 million charging transactions every month. EVBox charging management software tracks, manages, and optimises electric car charging. Keep the process safe and easy so users can enjoy their EV without effort. EVBox Elvi may connect to any backend through OCPP to offer charging session details. The author also announces load balancing for EVBox Business Line (Figure 2).

EVBox BusinessLine

Business charging solution

EVBOX

Up to 22 kW per connector
 Charges 2 cars simultaneously on a double station
 Durable, weatherproof design and integrated electrical protections
 Efficiently distributes power via (dynamic) load balancing
 Scalable to include up to 20 charging points per Hub-Satellite network

- 📶 Wi-Fi and 4G LTE connection
- ⚡ MID-certified kWh meter
- 🔌 Optional built-in RCBO
- ⚡ 6 mA DC leakage protection
- 💰 Tariff settings for public charging
- 🔧 Remote maintenance
- 🔧 Easy installation, service and maintenance

evbox.com



Figure 2: EVBox - BusinessLine (EVBox, 2022)

2.5.3 EvoCharge

In North America, EvoCharge is a leading provider of EV home charging solutions and cable management systems. Since the company's founding in 2009, there has been a commitment to offer Electric Vehicle Supply Equipment (EVSE) hardware with real Open Charge Point Protocol (OCPP) for simple integration into a secure and dependable non-proprietary network. The EvoCharge offers more than a standard Level 2 home charging station for electric vehicles. With standard Open Charge Point Protocol (OCPP), optional cable management systems that keep cables out of the way when not in use, and the most up-to-date EV charging technology, EvoCharge will change how electric vehicles are charged. Local load management enables using a building's existing electrical panel power capacity to charge many electric cars via charging stations by allowing the charging stations to communicate with one another and provide a consistent electrical current to each (EvoCharge, 2022). Solution's main features are presented in Figure 3.

EVOCHARGE™

EVOCHARGE® Level 2 EVSE (EVSE, iEVSE, iEVSE Plus) Product Description and Specifications

ELECTRIC VEHICLE CHARGING STATIONS

- **Modern Compact Design:** EVSE form factor smaller than a standard sheet of letter-size paper
- **Robust Construction, Certified for Outdoor Use:** tamper-resistant features; NEMA 4 certified for outdoor and indoor use
- **Simple Operation:** EVSE models simply plug-in to EV and charge
- **Network Options:** iEVSE and iEVSE Plus are OCPP 1.6 networked enabled charging stations to support access control, payment capability, remote management and control, demand response. Network connection via Wi-Fi; iEVSE Plus models also include 4G-LTE cellular, RFID card reader and Local Load Management capability.
- **Cable Management:** Standard Connector and Cable Holster included, optional cable management solutions available



Figure 3: EvoCharge Spec Sheet (EvoCharge, 2022)

Chapter 3

Contribution to IEVCC - Charging improvement

This chapter gives an outline of the work and task organization, the adopted development methodology, and the different stages of the solution design and implementation. When developing a project, there are several elements to consider in order to determine which methodology to apply and which instruments to employ to support this decision. Following industry trends, adopting agile frameworks is the wisest course of action. Although this project was developed using the Scrum framework, it is still important to mention the "Waterfall Model" and the basic agile principles on which Scrum is based as a foundation.

3.1 Waterfall Model

The Waterfall Model is a notion borrowed from industrial and construction development practises. When creating something enormous, such as an aeroplane or a ship, redesigning in the middle of the project is too expensive. As a result, each step or level in the "waterfall" must be finished completely before proceeding to the next (Figure 4).

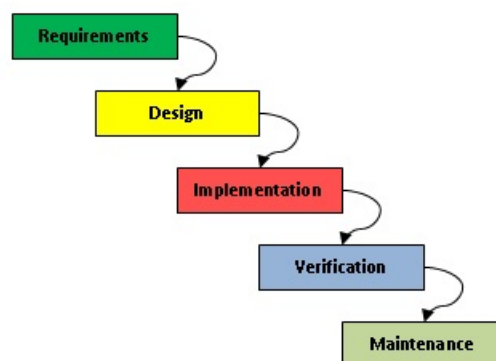


Figure 4: Waterfall Model (Marjerison, 2013).

While a design may appear attractive on paper, it isn't beneficial if the manufacturer can not produce it. As a result, waterfall model projects are typically collaborative, involving everyone from designers to manufacturing personnel. By the end of the design phase, everyone will understand how to complete the process.

The waterfall model has the advantage of encouraging upfront documentation. A downside to

this strategy in terms of software development is that it leaves little room for design changes late in the development cycle (Marjerison, 2013).

According to the agile manifesto's principles, motivated and empowered software developers, focusing on technical quality and simple designs, produce business value by providing functional software to users on a regular, short-term basis. These ideas have resulted in developing several techniques aimed at providing more value to customers. At the heart of these approaches is the concept of self-organizing teams whose members work collaboratively and at a speed that supports their creativity and productivity. The principles promote procedures that allow for requirement changes at any point during the development process. Additionally, customers (or their representatives) participate actively in the development process, facilitating feedback and reflection that can result in more fulfilling solutions. The principles are not intended to be a formal definition of agility but to serve as guidance for rapidly providing high-quality software (Dingsøyr et al., 2012).

3.2 Scrum

Scrum is a methodology for managing agile projects. It breaks a project into sprints. Each iteration (sprint) lasts around 30 calendar days and must result in workable software code that adds new (extra) functionality to the software. Thus, the customer gradually receives components of the solution that are instantly usable. Three critical roles are assigned to a Scrum project:

- Product owner: determines the features or expected outcomes of the product, ensuring profitability and business value;
- Scrum master: accountable for ensuring that the team follows the established procedure, as well as resolving issues and protecting the team from external influence;
- Development team: is in charge of implementing the functionality. The team is functionally diverse and self-organized. Team members assign tasks independently and are collectively accountable for the project's success or failure.

Scrum's process flow (Figure 5) is based on various types of meetings during which project planning and monitoring artifacts are created.

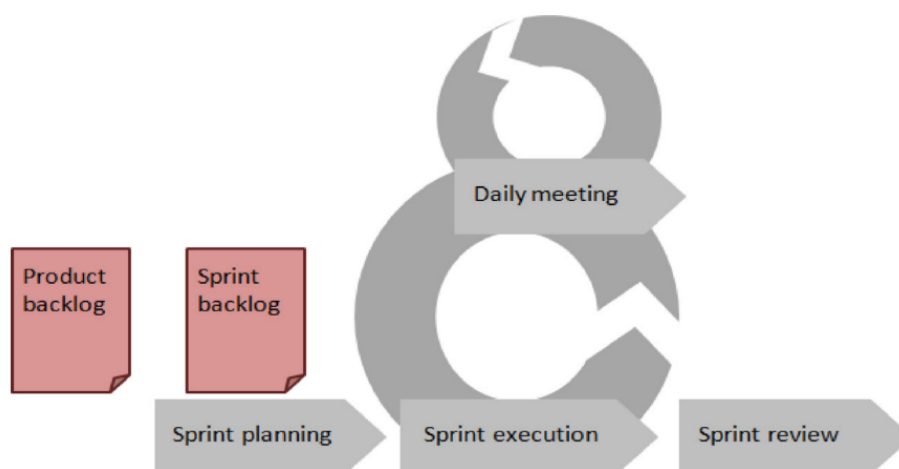


Figure 5: Scrum process (von Wangenheim et al., 2013).

The project plan uses a list of client requirements (usually in the form of user stories) documented in the product backlog. The product owner is responsible for maintaining this artifact. He assesses and prioritizes requirements based on their profitability and business value. At the start of each sprint, the product owner collaborates with the project team to define the subset of requirements that will be developed during this iteration. The team determines how much work they can complete in the upcoming sprint based on their velocity, which is a rough estimate of how much work a team can complete in a current sprint based on the previous sprint's work. The selected needs (user stories) are also refined during the sprint planning meeting into actions necessary to accomplish the specified objectives. The sprint backlog reflects this.

During the sprint's execution, team members carry out the defined activities. Each day, the team gathers to monitor progress and maintain control over the project. Each team member answers three questions during the daily scrum meeting: What did you do yesterday? What do you have planned for today? What constraints prevent you from reaching your sprint and project commitments? After each sprint, the team presents its work to the product owner and other interested users. This sprint review meeting lets users provide feedback on the work completed and establish requirements for the next iteration. After the sprint completion, a retrospective meeting is held to promote ongoing improvement in how the team creates value. During this discussion, the team will identify the strengths and flaws of the Scrum processes that were noticed during the previous sprint and recommend ways to improve them (von Wangenheim et al., 2013), (Srivastava et al., 2017).

3.3 Planning

While project management is critical, many initiatives fail to meet their objectives. This is especially true for IT initiatives, which fail regardless of how they are organised. In 2017, only 34% of projects were finished on schedule and 42% were within budget. In 2018, the situation improved marginally, with 40% of projects finished on schedule and 46% within budget.

Different methods of project management employ a variety of different success criteria for projects. According to Vujović et al. (2020), it is the management team's responsibility to assess what is appropriate and applicable to a particular project. The most frequently cited reasons for IT project failure are: inadequately defined project objectives, inadequate planning and assessment, insufficient project methodology or its disadvantages, a lack of understanding of client requirements, communication problems between the contracting authority and IT experts, and contracting authority expectations that are unrealistic.

The fast evolution of technology has had a significant impact on the complexity of an IT project. The phases of managing an IT project are as follows: design, requirement definition, planning, design, production, and delivery. The success or failure of a project is contingent upon the project manager's readiness and knowledge (Pellerin and Perrier, 2019), (Vujović et al., 2020).

Current project development is based on qualitative and quantitative research methods, used throughout the thesis's analysis and during literature review. In this sense, the procedure followed the following logic:

- Literature review (a process that occurs repeatedly throughout the project cycle and thesis writing);
- Choosing technology and tools for developing the solution;

- Development and structuring of a scalable architecture based on a review of the literature;
- Gateway application component development;
- Installation and configuration of all architecture components;
- Platform review and improvement.

3.4 Intelligent EVSE Prototype

The electric infrastructure in residential buildings wasn't designed for high power usage, and most car parking spots only have a standard 16A Schuko plug. Due to heat and safety concerns, the recommended maximum current is only 10A, limiting the charging capacity. Most manufacturers are making their batteries bigger so EVs can go farther between charges. For 10-16A connections, the time it takes to charge will get too long. So, considering the low current and bigger batteries of modern EVs that need up to 32A per phase, the charging process will take several hours longer. Most residential buildings, where more than one family shares some spaces, aren't ready to use high-current charging devices. If a dedicated cable is connected directly to the building's main electrical board, higher charging currents (up to the main board's design capacity) can be used. Some companies make wall box chargers that can be used to charge at a faster rate. Most of the time, these pieces of equipment have a fixed charging power that is often very close to the house's maximum contracted current/power. If the EV charging current is high and other loads are connected, a tripping order may be set off, which will turn off the whole installation. Priority can be set for charging or non-EV loads, load shedding plans can be put into action, or the rate of charging EVs can be limited to the maximum available capacity, which is the contracted power minus the total power used by non-EVs (Cardoso et al., 2021).

Cardoso et al. (2021) presents the Standalone and Mesh versions of the IEVCC systems (Figure 6). This IEVCC was developed with an ESP-32 board. Both versions can connect to a wireless network to share information about charging status and get information from the main system.

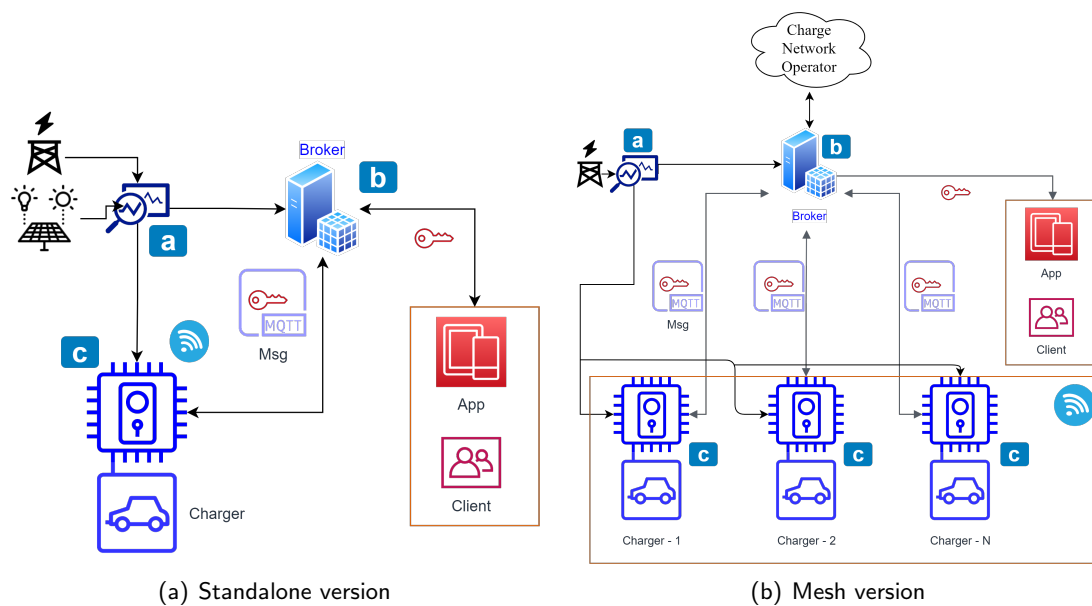


Figure 6: IEVCC system diagrams (Cardoso et al., 2021).

3.4.1 Architecture

With the previously identified challenges (1.2) in mind for designing and implementing multi-client network solutions in the IoT, the proposed technical architecture must be capable of managing high data loads. The solution must be resilient and scalable to address the mesh installation problems and optimize grid usage. These aspects will benefit the end consumers and also assist the electricity distributors. This section proposes an optimized architecture to fit two distinct IEVCC system configurations, the Standalone and Mesh versions intended for single use or in condominiums where tenants do not have access to a portion-powered parking spot as presented by Cardoso et al. (2021).

The proposed solution is considered a streaming analytics system. These systems typically contain three layers: ingestion, processing, and storage. The ingestion layer is the gateway to streaming. Data flow from inputs to processing and storage levels is decoupled, automated, and managed. The processing layer receives the ingestion layer's data streams and transfers the output or intermediate results to storage. The storage layer keeps data in memory for iterative calculations or in databases for long-term storage. The analytics findings are given to a range of display and decision-assistance tools (Isah and Zulkernine, 2018) (Dias de Assunção et al., 2018).

According to Aung et al. (2021), there are five primary data processing architectures on the Internet of Things: fog-based processing, middleware-based processing, cloud-based processing, cloudlet computing, and mobile-edge computing.

The current data load comes from chargers, electrical counters, and message handling devices (broker). While systems operate, it is vital to log and store not only charging data, but also device status and usage metrics. To minimise damage in the event of network or device failures, the local setup (device layer) must be able to store and recover from failures. The final architecture proposal has this into consideration.

The next natural step is to store data in the cloud, but once again, due to the sensitive nature of data and the significant load caused by each local instance, transmitting gathered data straight to cloud servers has proven to be challenging Garcia et al. (2018). The same article

discusses how fog computing may help minimize cloud reliance while improving performance. Nevertheless, the paper concludes that cloud and fog are complementary and can help deliver better and more complete services.

Cloud computing and fog computing infrastructures do not compete with each other, and they're complementary architectural solutions. IoT applications connect across fog nodes, and devices must be linked to at least one of these fog nodes (Fig. 7). Any device part of the IEVCC solution may connect to fog nodes which may be used in specific geographical cloud areas Kanyilmaz and Cetin (2019).

Because each fog node is a single point of failure, its spread and replication across regions should be considered for failure recovery and redundancy.

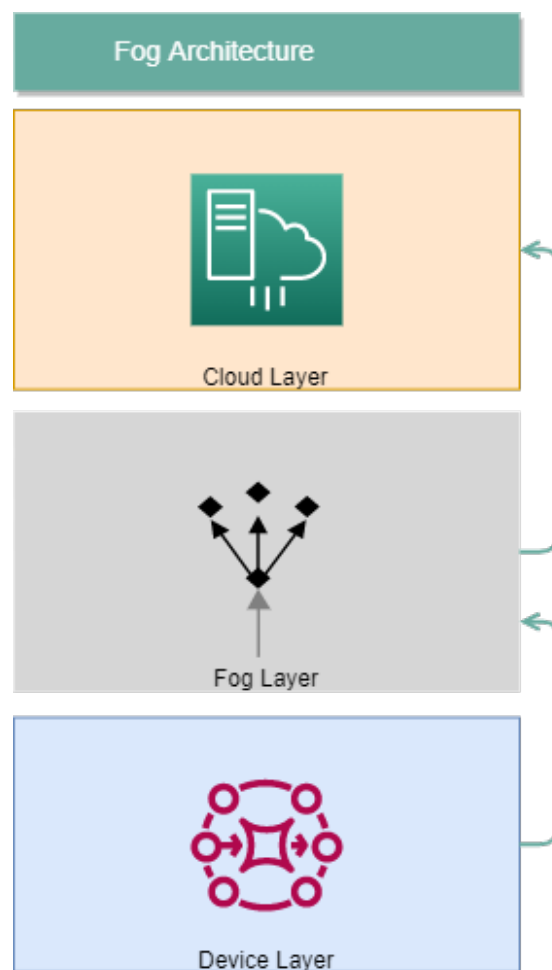


Figure 7: Fog Architecture (Baptista et al., 2022b).

The charging solution installation can include one or more chargers. Multi-charger installations have more than one charger and may also include multiple electrical counters. Fig. 8 illustrates how each device connects to the "heart" of the device layer (Manager/Broker). This device is responsible for message handling and forwarding. At the same time, it manages authorized devices and clients during charges.

Each instance can be configured with custom load balancing rules, charger priority, and energy source selection when more than one source is available or when the provider shares the current source through an API endpoint.

All data is stored locally and forwarded to remote instances for data cleansing and transformation. Electrical power usage, system logs, and client usage are then available for access by clients and providers.

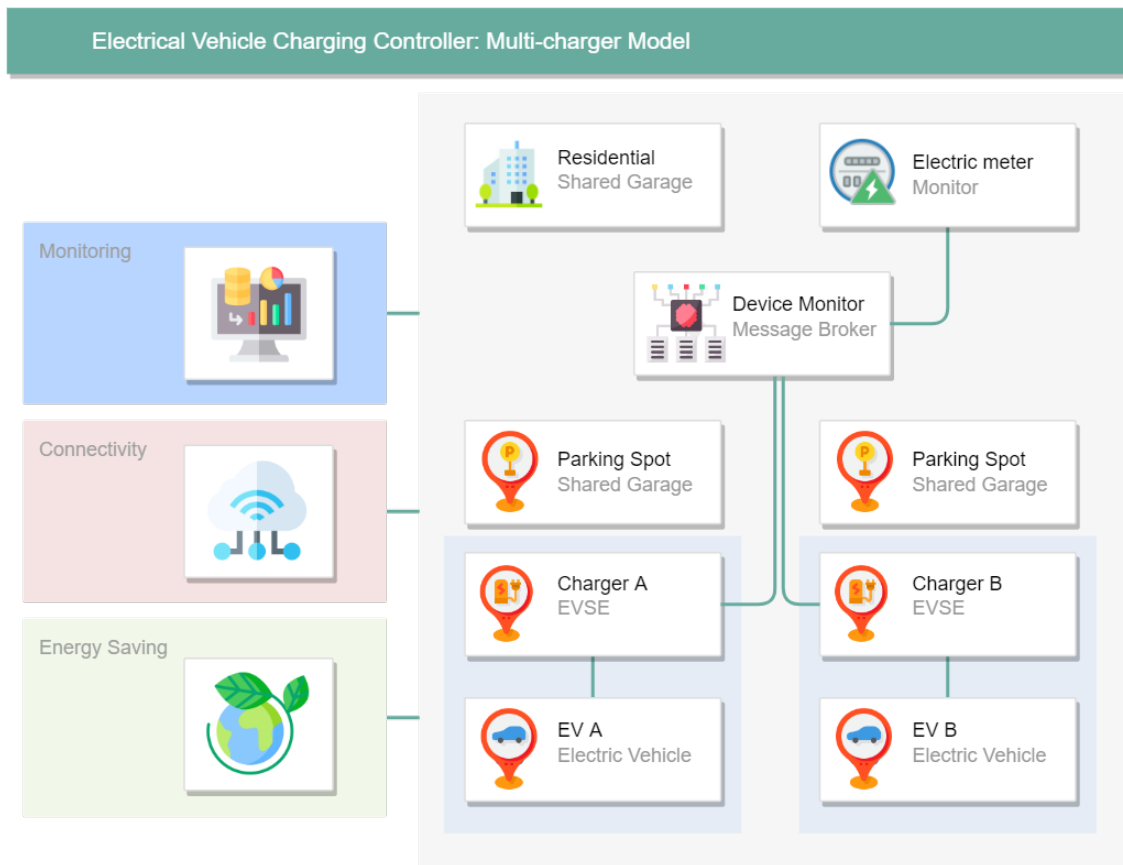


Figure 8: Multi-Charger Model (Baptista et al., 2022b).

Design of Data Management Service Platform

In order to allow data flow between several independent data producers and consumers, a data stream ingestion system must be scalable, resilient, and extendable.

Chargers and electrical counters are the primary focus of the current configuration. However, the Device Manager design ensures that more Internet of Things devices will be able to connect and integrate into the solution in the future.

To demonstrate how to integrate the Fog Architecture into the Multi-Charger Model, the Fig. 9 depicts the position of each entity within the three Fog Architecture levels.

Each colored section maps the three distinct layers (Device, Fog, and Cloud), where generated data is saved and forwarded to the subsequent layer instances. It is crucial to clean and aggregate each record to be stored in the database during this process. It is also important to note that while generated data flow in one direction only, it is mandatory to authenticate users, devices, charging sessions, and others. This responsibility is taken care of by the Device Manager through the API instance in the cloud layer.

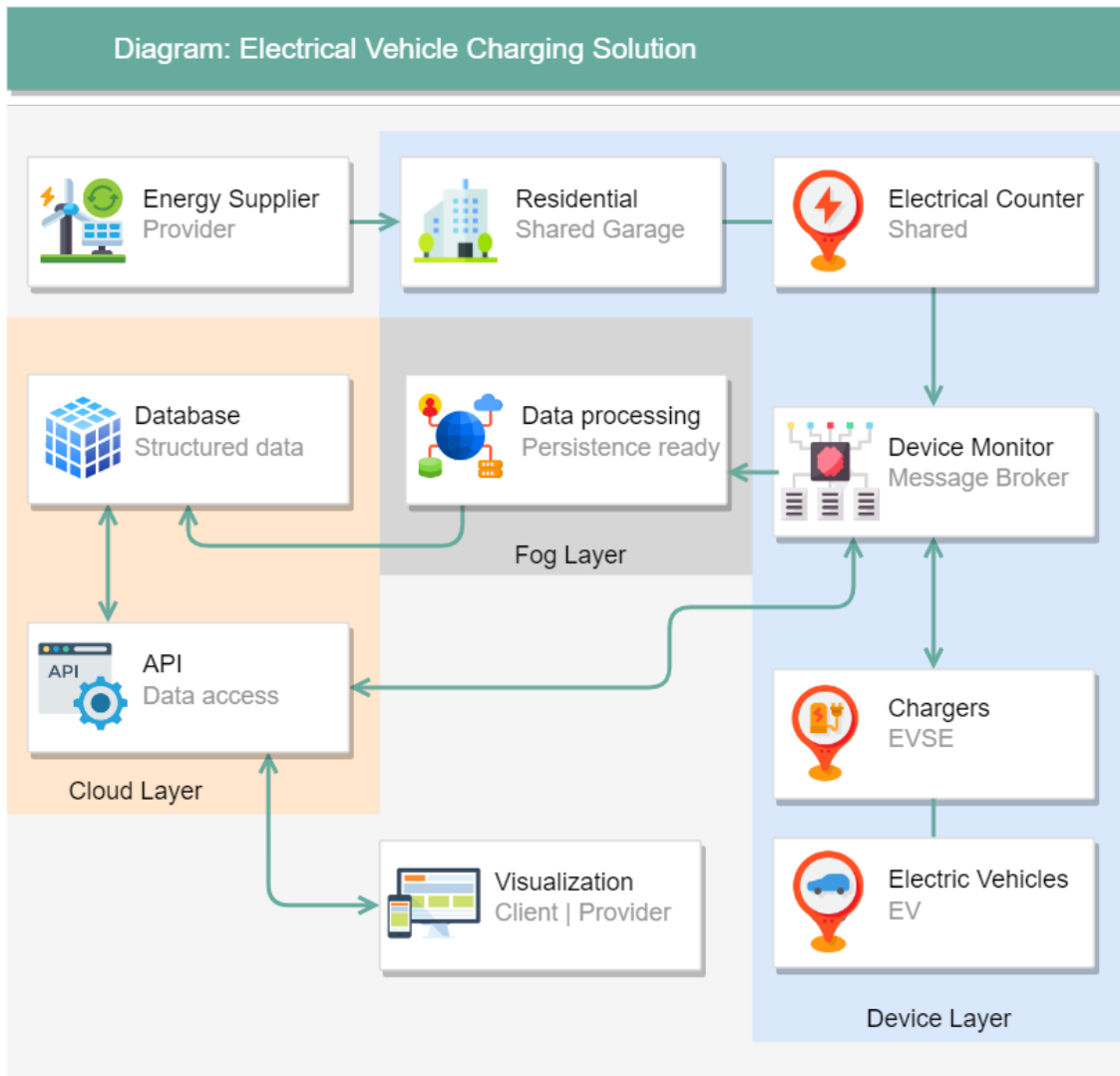


Figure 9: Proposed Architecture (Baptista et al., 2022b).

Device Layer

The device layer includes all the devices that support the local area network, such as routers, switches, wireless access points or extenders, and all the smart devices connected locally. The smart ESP32-based devices are chargers and electricity meters for the current solution. Multiple other IoT devices may be integrated into the solution in the future.

The solution's heart is the Device Manager. Raspberry Pi version 3 boards were tested during development with no performance issues while handling device authorization, messages, local storage, and data forwarding to the fog layer.

For the Manager/Broker device, the hardware choice (board specifications) will consider the principles of IoT family devices like power consumption and price. Since there are no CPU architecture restrictions to execute the developed Manager/Broker applications, the minimum requirements should be a wireless adapter and the ability to run Docker¹.

¹Docker: <https://www.docker.com/>

Fog Layer

The Fog Layer comprises devices in an intermediate layer between the cloud and the Device Layer. In this case, data is transferred to and processed by a computer or data center preferably located in the same region as the device layer. By spreading this processing power across multiple regions, each fog node's total load will reduce, and redundancy will increase.

Fog node hardware must meet the minimum system requirements set for the distributed event streaming platform (e.g. Kafka²) and the applications for cleansing and transforming data.

Cloud Layer

The cloud computing infrastructure builds on top of large-scale clusters that run various applications and pursue the core foundation that enables computing resources to be used to their full potential.

Cloud customers expect the entire system to be reliable, with redundant network and hardware. These cloud solutions allow companies to access data storage, resources, and on-demand services over the internet. Although cloud providers offer a variety of solutions for several operations, based on the presented Mesh-Model (Fig. 9), the core business activities in the Cloud Layer include database services and API web applications.

The Cloud Layer is sub-layered into three layers: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Choosing the best cloud layer depends on the budget, resources, the size of the operations, and multiple other factors.

3.4.2 Tech Stack

With an architecture diagram available, it is essential to decide which applications will implement each architecture entity and how the connectivity and communication will be ensured. A tech stack is a company's choice of technologies to develop and manage an app or project. A tech stack often includes programming languages, frameworks, databases, front-end and back-end tools, and apps linked through APIs (Limón and Schulaka, 2020).

As stated in 1.4, the short time to implement the solution and the limited monetary and hardware resources lead us to choose each component for the Tech Stack based on ease of installation or configuration, availability, previous experience with the tool, and finally, the price.

In a top-to-bottom analysis of Fig. 10, in the Device Layer, the current charger and electricity meter devices are programmed in C++, while the Device Manager is currently being developed in Python 3.8, with a tested compatible version range from Python 3.6 to Python 3.10. MQTT message broker (Mosquitto MQTT) and InfluxDB, an open source Time Series Database, both run on the same hardware.

Each fog node in the Fog Layer will provide one or more Apache Kafka instances, an open-source distributed event streaming platform. Apache Kafka advertises necessary core capabilities like high throughput with low latency (2ms), being prepared to scale, and delivering high availability. It's also important to mention the built-in stream processing that enables the processing of event streams using joins, aggregations, filters, transformations, and exactly-once processing. It is also worth mentioning that the Kafka Connect interface is pre-integrated with hundreds of event sources and sinks, including Postgres, JMS, Elasticsearch, and AWS S3.

In the Cloud Layer, our solution's core depends on a PostgreSQL database instance and the possibility to host web applications like client portals or APIs to access and store information.

²Kafka: <https://kafka.apache.org/documentation/>

The best cloud solution for PostgreSQL service is still open for further analysis, howsoever it is mandatory to have scalability possibilities, backups, and snapshots. Multi-region availability and synchronisation will be decisive when dealing with thousands of clients.

Region and response time are essential factors in web application hosting, but so are high availability with load balancing, security, and scalability (Ramamurthy et al., 2020).

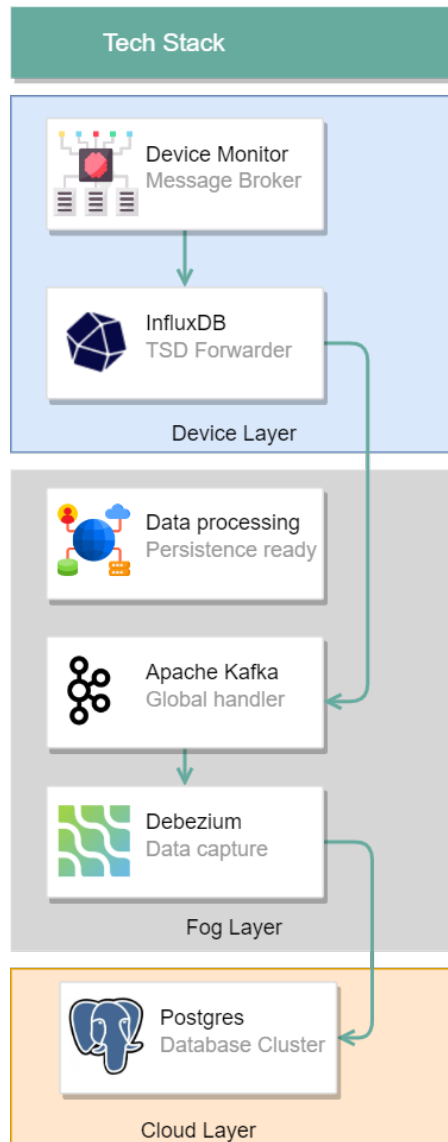


Figure 10: Tech Stack (Baptista et al., 2022b).

3.4.3 Message Protocol

The proposed Tech Stack is based on multiple IoT principles. It aims to incorporate various processes and services that run locally, on small IoT devices, and remotely in the cloud, fog, or edge servers. We'll consider the interactions between local IoT devices and local services and their communication with remote servers for the messaging protocol choice.

Wytrebowicz et al. (2021) analyzed twelve different messaging protocols for IoT systems. Ac-

According to the same authors, choosing one messaging protocol for a new design is not a simple task, and making a wrong choice can slow development and increase design costs. An IoT system may have sub-networks for many device types. Due to pricing or energy constraints, some devices have limited computational and memory resources. These devices should have tiny memory footprints and communication protocols. Moreover, to conserve energy, the protocols should minimize the communication overhead. We selected four from the list of twelve messaging protocols (Table 3.1) for brief analysis and further selection. This selection is based on Quality of Service (QoS) levels, topic addressing, and security capabilities.

It's possible to create a device-level IoT platform or middleware to be deployed on IoT devices, to enable faster system development. The device-level IoT platform includes a message protocol or protocols. Middleware should be optimized for a given IoT device's communication needs. These devices fall into four categories: constrained devices; unconstrained devices connected to the internet via a pay-per-use plan or a flat-rate plan with significant limits on the volume of transferred data; unconstrained devices being not always online; unconstrained, online devices anytime accessible by applications or services.

The messaging protocol transport features also play a big part when considering the final implementation. Due to previous experience, the resources available, and the fact that it matches our solution needs (Table 3.1), the protocol choice stands for MQTT.

Table 3.1: Transport features (Wytrebowicz et al., 2021)

Protocol	Basic [Additional] Transport	Security	QoS	Data Prio- ritisation	Addressing	Filtering Ca- pability
MQTT	TCP,[WebSocket]	TLS profile: authenti- cation + authorisa- tion	3 levels		topic	
MQTT- SN	ZigBee or UDP		4 levels		topic	
CoAP	UDP,[TCP, Web- Socket]	DTLS	2 levels		URI	URI syntax
AMQP	TCP, [UDP, SCTP, Web- Socket]	TLS, SASL	3 levels	yes	queue, topic/rout- ing key	based on message properties

3.4.4 Intelligent Electric Vehicle Charging Controller

IEVCC (Intelligent Electric Vehicle Charging Controller) prototypes are developed on ESP boards (Fig. 11). Both the Standalone and Mesh versions offer connections over a wireless network to deliver charging status information and receive data from the main system. The ESP32-POE is compatible with an Ethernet network, which can be leveraged if the building has this type of infrastructure. The selection of these two boards enables the development of low-cost systems, hence facilitating the swift adoption by users.

The standalone version targets a single-user situation (personal use) where the user has his own parking area and charges from his house electric access point. This version adjusts charging current based on house consumption in Smart mode. It also works on demand (manual

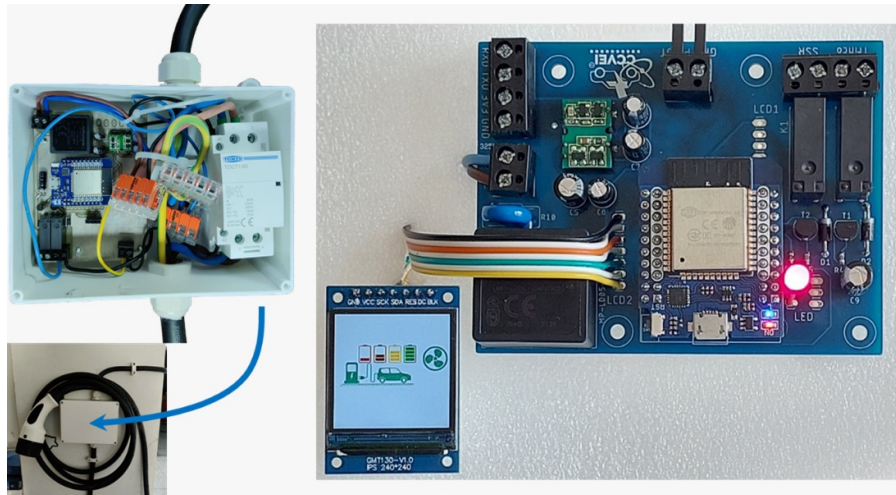


Figure 11: IEVCC Prototype (Cardoso et al., 2022)

mode), allowing the user to define the charging current manually. This mode makes it portable and independent from house-reading systems.

The mesh version must deal with multiple users, possibly with different needs or rights, as well as infrastructure conditioning factors like cable thermal limits or maximum available power from the energy provider. Loading balance is challenging. Other challenges include deciding who receives more power or if everyone can charge simultaneously. The system uses NFC identification to adjust all chargers based on the user profile, the number of users charging at once, and the available power. In the worst case, the user may not receive authorization to charge. Chargers must transmit information to the broker about charging sessions (time, energy used) to apply fees.

The consumption monitor hardware measures Voltage, Current, and Active Power at timed intervals to monitor power consumption. It uses a PZEM-004T V3.0 board with a Vango V98xx ICciteb9 (Fig. 12). The measured values are sent to the consumption monitoring system, which stores them in a database for the user to access. The MQTT protocol sends current information to a broker so IEVCC can adjust charging current based on contracted power. When renewable sources are available (e.g., solar panels), the system sends only the available power to the charger instead of the grid.



Figure 12: Consumption monitor hardware/Energy Meter

3.4.5 IEVCC Broker/Manager

The Manager/Broker carries out the management of all entities involved in the charging process. Dependent on the MQTT broker, it subscribes to all existing MQTT topics and manages clients (chargers and electrical energy meter). The developed application (Manager/Broker) settings allow the user to limit the number of chargers operating simultaneously, set priorities for the chargers, and monitor and manage which devices are authorized to connect to the Manager/Broker. In the performed tests, priority was given to the charger that was connected first.

The Fig. 13 presents an example of the developed communication protocol where the currently supported devices announce themselves to the Manager/Broker.

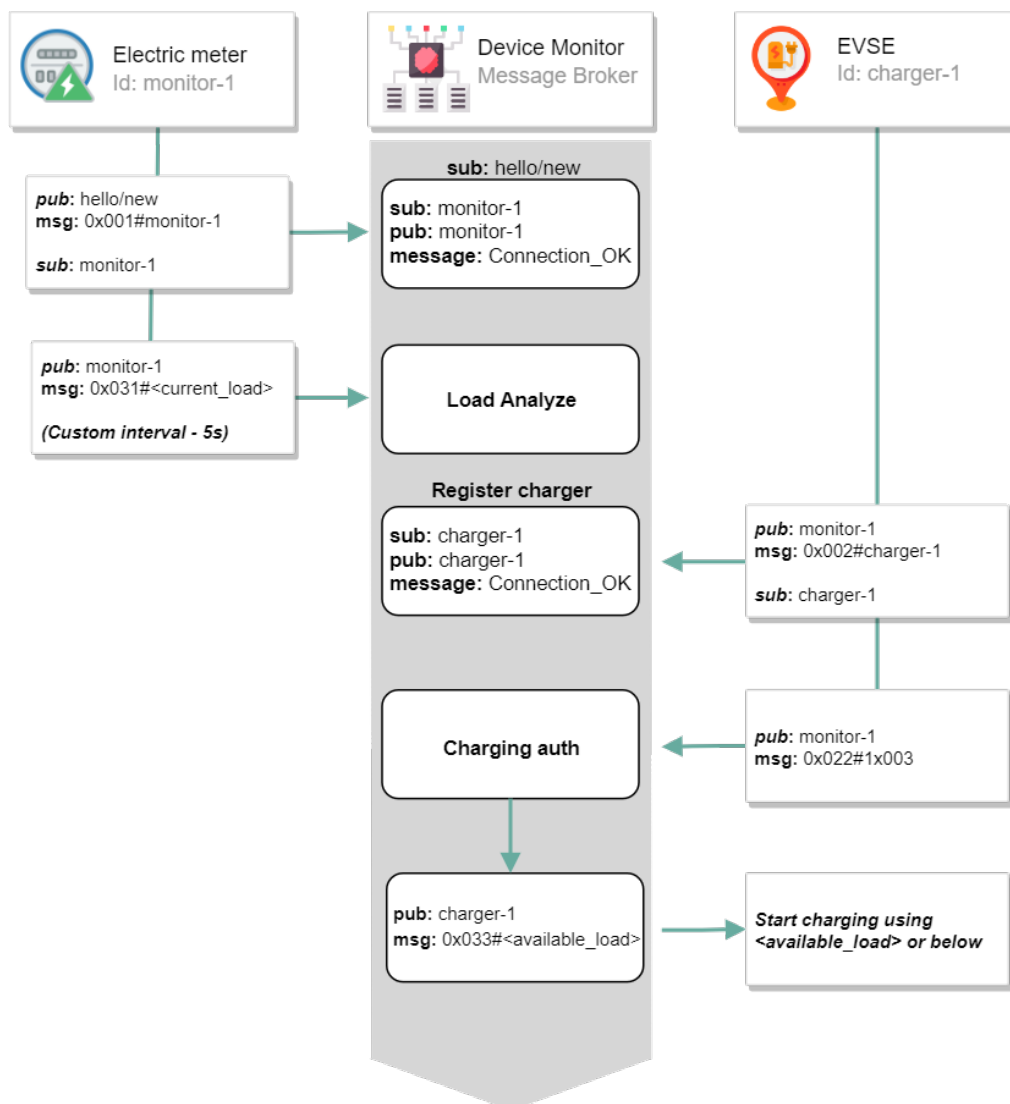


Figure 13: Protocol time-lapse (Baptista et al., 2022a).

According to the established protocol (Fig. 13), the Manager/Broker will be the first entity to connect to the MQTT broker using the protocol message format (Fig. 14).

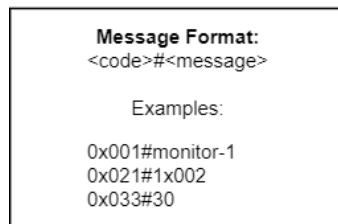


Figure 14: Protocol message examples.

When the chargers and the electrical counters join the system, they must announce themselves by subscribing to the connection topic and then sending a message with the proper content. After they have been properly identified and given permission to access, they will be added to the Manager/Broker registers. The Manager/Broker will then contact each party through a separate MQTT topic.

Through the aforementioned message protocol, it is possible to announce the connection or charging intention through the MQTT topic, in the case of a charger (Fig. 15).

```
1  msg_codes = {
2      "0x001": "new_counter",
3      "0x002": "new_charger",
4      "0x011": "del_counter",
5      "0x012": "del_charger",
6      "0x021": "counter_status",
7      "0x022": "charger_status",
8      "0x031": "counter_load",
9      "0x032": "charger_load",
10     "0x033": "available_load",
11 }
```

Figure 15: Protocol message codes.

Regarding the electric meter, it is crucial that it can report the power that is being consumed. This report will allow the Manager/Broker to calculate the difference between the contracted power and the amount of power being used, which represents the amount of power still available to be shared among all running chargers. The time interval for updating this data can be customized to ensure that the contracted limits are not exceeded and thus causing power outages. This update of the total power available to the chargers also makes it possible to balance the charging within the minimum limits for each charger. Suppose each charger has less than 6A available. In that case, one or more chargers will be put on standby (waiting for a charging signal), increasing the availability for each running charger and ensuring that the charge stays above the minimum required.

The Manager/Broker is also responsible for ensuring the persistence of all generated data

locally in a database, which provides only local operation and also sends all activity to a cloud database storage.

Development

The current Manager/Broker version is based on Python 3.8 and implements best practises for version control using Git with multiple development stages, secret management using the dotenv³ library, code format and lint with pylint⁴ and black⁵.

With code reuse, optimization, and extensibility in mind, multiple design patterns were implemented, such as "Singleton" or "Factory" (Fig. 16). Implementing these design patterns allows for easier and cleaner future development when adding new features, like adding other devices such as solar panel controllers or API endpoints with energy pricing.

Regarding data management, a flexible "Datastore" was implemented to manage both structured and non-structured data.

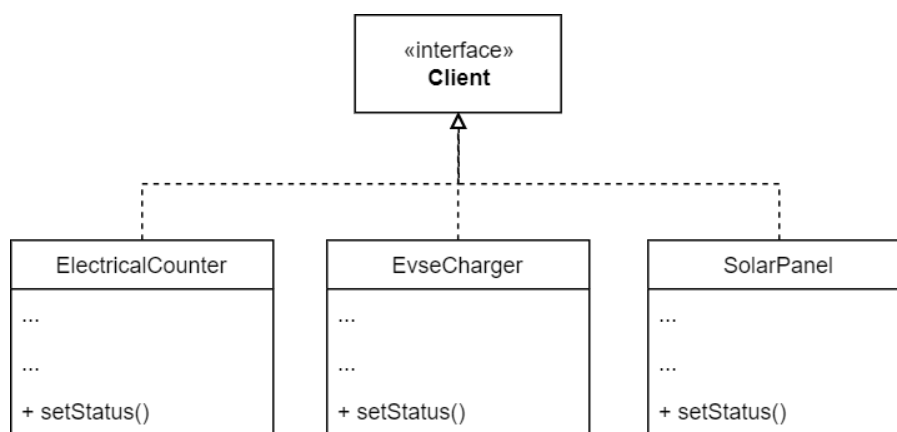


Figure 16: Client Factory UML.

Software Testing

During our solution development, we've created a suite of tests to meet the stakeholders' expectations and the quality standards. Using pytest⁶ for unit testing development enabled a quick and simple method to validate existing code and also guarantee compatibility while adding new features.

Software Testing is described as the process of determining whether or not a particular system meets its established initial requirements. It is primarily a validation and verification procedure determining whether the built system fits the user's needs. Therefore, this activity yields an outcome that differs from what was anticipated. Software Testing identifies faults, errors, and missing requirements in a system or software that has been produced. Therefore, this examination offers stakeholders specific information regarding the product's quality (Jamil et al., 2016).

³dotenv: <https://github.com/theskumar/python-dotenv>

⁴pylint: <https://pylint.pycqa.org/en/latest/>

⁵black: <https://github.com/psf/black>

⁶pytest: <https://docs.pytest.org/en/7.1.x/>

Most of the developed unit test cases involve the message protocol tests, where the different device stages and statuses are tested depending on load or configuration. Another test case example is the ability the Manager/Broker has to allow new connections by checking legitimate devices. This helps to prevent some exploit situations, such as trying to connect unauthorized devices or injecting malicious messages (Fig. 17).

While testing software, continuous risk analysis is required. During testing, it's essential for software testers to know how to turn a vast number of tests into a manageable set of tests and determine which risks need to be tested and which don't. Testing is the most critical aspect of the software development lifecycle since it determines the ultimate delivery of the product. It is a time-consuming and labor-intensive procedure (Jamil et al., 2016).

```
1 def test_01_new_message_invalid(self):
2     datastore.reset_datastore()
3     assert (
4         message_ctrl.new_message("0x000", "0x000", "Counter-1")
5         == "Invalid Message: message code"
6     )
7
8 def test_02_new_message_new_counter(self):
9     datastore.reset_counters()
10    assert (
11        message_ctrl.new_message("hello/new", "0x001", "Counter-1")
12        == "Counter registered: Counter-1"
13    )
14
15 def test_03_new_message_new_charger(self):
16    datastore.reset_datastore()
17    message_ctrl.new_message("hello/new", "0x001", "Counter-1")
18    assert (
19        message_ctrl.new_message("Counter-1", "0x002", "Charger-1")
20        == "Charger registered: Charger-1"
21    )
22
23 def test_03_new_message_new_charger_invalid_topic(self):
24    datastore.reset_datastore()
25    message_ctrl.new_message("hello/new", "0x001", "Counter-1")
26    assert (
27        message_ctrl.new_message("Counter-2", "0x002", "Charger-1")
28        == "Invalid Message: message topic"
29    )
30
31 def test_04_new_message_max_counters_exhausted(self):
32    datastore.reset_datastore()
33    message_ctrl.new_message("hello/new", "0x001", "Counter-1")
34    assert (
35        message_ctrl.new_message("hello/new", "0x001", "Counter-2")
36        == "Max registered counters achieved"
37    )
```

Figure 17: Tests sample.

Chapter 4

Discussion

The findings of an experiment are significant for two main reasons: first, they demonstrate how well a particular design or implementation works, and second, they show how a specific component, set, or system functions as a whole. The primary objective of this study was to produce a charger orchestrator (Manager/Broker) and demonstrate how home customers might benefit from its application. During the development of the charger board using the ESP-32 board, numerous tests and charging sessions took place. In the scenario with multiple chargers, implementing the same tests and charging cycles is a challenging task with high costs, mainly due to access to several electric vehicles and all the infrastructure necessary for charging the vehicles. In this way, we decided that simulating with actual equipment would not be feasible, so software simulators were developed that reproduce the behavior of the chargers and the Manager/Broker (Fig. 18).

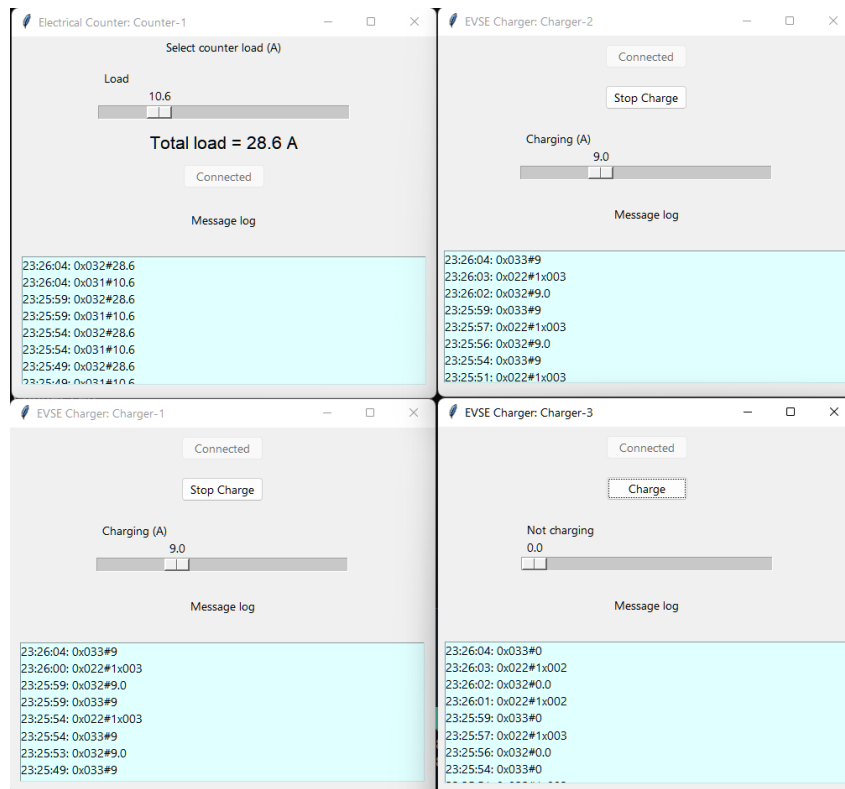


Figure 18: Software simulators.

The simulator software in the Fig. 18 presents a session with a counter at the top left and three chargers so that, in practice, the entire system recreates a real scenario with the possibility to add and remove equipment, change charging values, and domestic use at less restricted intervals without jeopardizing the safety of the equipment. The same simulators can also be adapted to take different roles like solar-panel controllers, which can contribute to EV charging and optimize the charging cycles.

Multiple simulations were conducted using the developed simulator during the development and testing stages. Two scenarios were selected and implemented from the complete list of simulations; the first recreates the standalone charger version, and the second introduces three chargers in total, where the total house consumption changes and chargers connect and disconnect randomly (Figure 6). The simulation settings are identical in both scenarios; the contracted power is 6.9kVA, equivalent to a maximum current of 30A, and the update interval is 5s; the minimum load available for a charger to start charging is 6A. Finally, the Residual Current Breaker with Over-Current (RCBO) is triggered in case the current equivalent to the contracted power is exceeded.

The first simulation (Fig. 19) includes one single charger. When the instantaneous consumption of the household exceeds the threshold limit, the IEVCC automatically adjusts the charging current (which is proportional to the charging power) so that the total current drained by the home appliances and the car does not exceed the maximum value established. The graph reveals that the established value is momentarily surpassed before being swiftly corrected (e.g. 23:23:30 - 23:23:40).

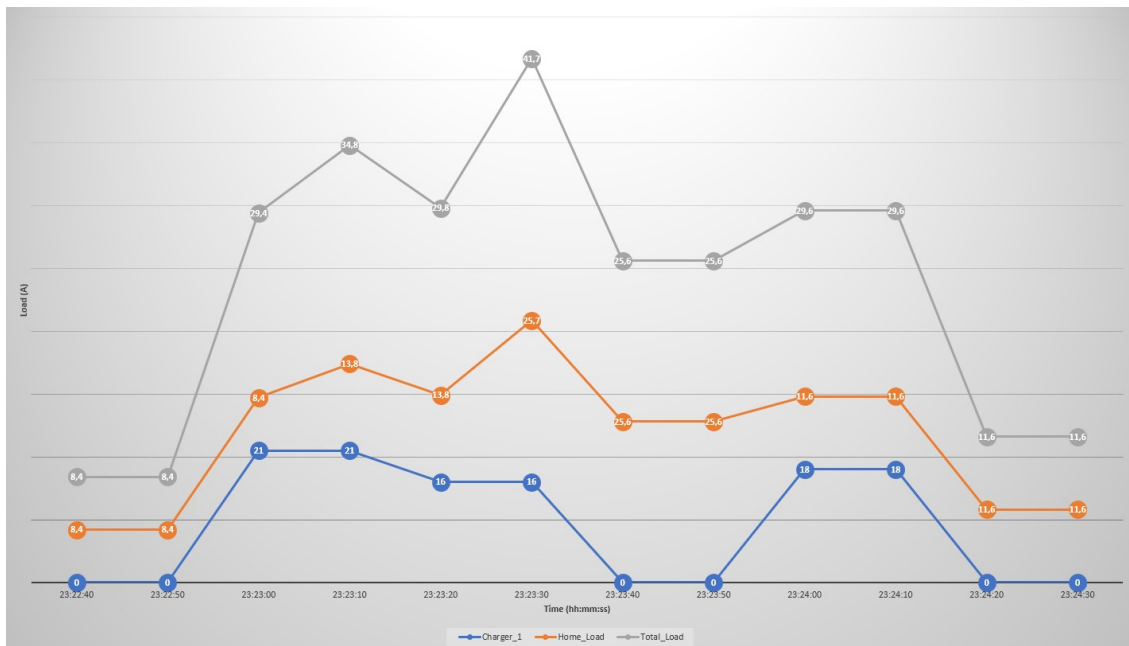


Figure 19: Standalone simulation.

Considering the second scenario, which includes three chargers for the given sample interval, it is possible to examine the load distribution between the home's energy usage and the three linked chargers. Changes in EV charging consumption do not interfere with home usage. The Manager/Broker will orchestrate the available load and guarantee that it won't exceed the load available for charging purposes. The implemented protocol and the MQTT response time prevent power outages faster than a Residual Current Breaker with Overcurrent (RCBO).

Figure 20 presents several different situations, each of which will be discussed in more detail below. Chargers are given priority in the order in which they registered or made a request for charging. In this illustration, charger 1 was the first to register, then charger 2, and finally charger 3 was the one that registered last.

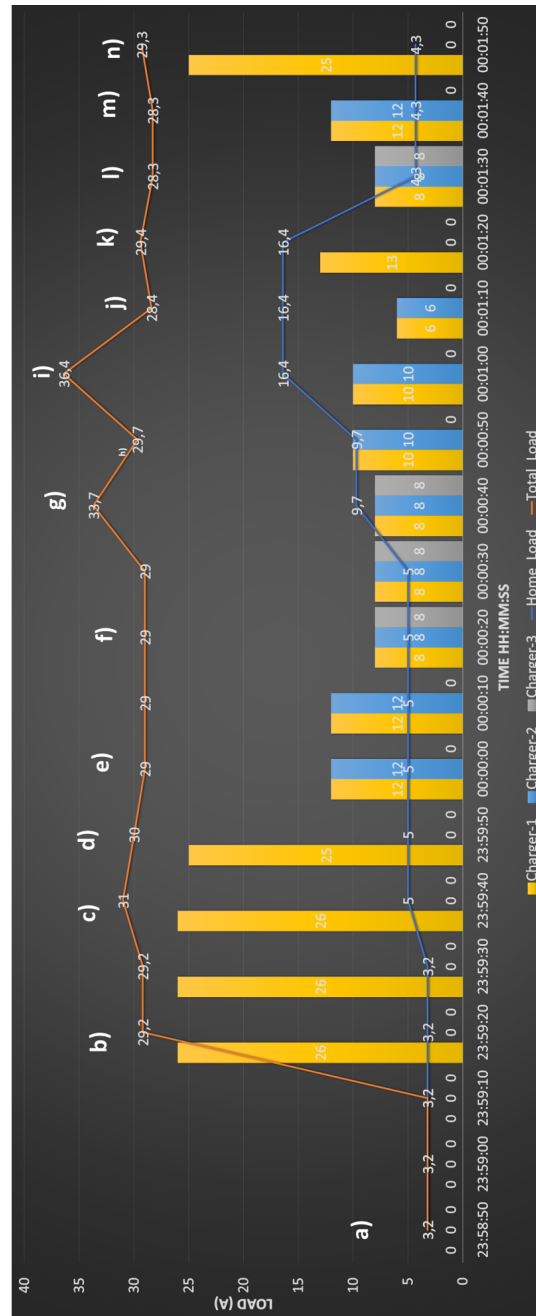


Figure 20: Multi-charger simulation.

The following is what is occurring in each moment:

- (a) In the home, the consumption is 3,2A. Three chargers are connected, but none is currently charging.

- (b) Charger-1 starts its charge. The available load for chargers is fully assigned to Charger-1 (26A).
- (c) Home appliance consumption ("Home Load") increases to 5A and the "Total Load" exceeds the 30A maximum limit.
- (d) Charger-1's load is reduced to 25A so that the 30A maximum is not exceeded.
- (e) Charger-2 initiates the charging process, and the available load is shared equally between both chargers.
- (f) All connected chargers are charging. The charging load per charger is 8A.
- (g) The "Home Load" increases once again. Since the 30A maximum was exceeded, the load per charger must be readjusted.
- (h) During the previous sampling period, the "Home Load" increased again, and the available power is insufficient for all chargers to charge at the minimum power (6A). Therefore, charger 3 (the last charger to register with the manager) must be placed in standby mode.
- (i) A new increase in "Home Load" (16,4A) requires adjustments to the available charging load.
- (j) Due to previous exceeding of the contracted power, the charging power of both chargers has been reduced (6A).
- (k) Charger-2 stops its charging cycle. As a result, Charger-1 load increases (13A), utilising the entire available load.
- (l) All chargers start their charging cycles while the "Home Load" decreases to 4.3A. This allows chargers to use 8A each.
- (m) Charger-3 stops its charging cycle. As a result, Charger-1 and Charger-2 load increases (12A), utilising the entire available load.
- (n) Charger-2 stops its charging cycle. As a result, Charger-1 load increases (25A), making use of the entire available load.

The presented results for both simulations share the same evidence. The Manager/Broken is capable of orchestrating one or more devices. By authorizing only legit devices and monitoring each MQTT topic, it prevented any RCBO triggering by dynamically adjusting each charger load based on the electrical counter instant consumption.

Data persistence enabled chart generation and is helpful for multiple other scenarios. Managing and estimating costs or finding usage patterns may be beneficial for users.

From the above it is possible to infer that this proposal has the potential to set the basis for future charging operators at local, regional and national level. The applications and API's supported by the proposed architecture ensure all the communication needed for both home users and grid operators' use-cases.

Considering the implementation of distributed EV charging clusters, the proposed architecture is capable of handling the generated data through different stages/layers. With a geographic balanced distribution of fog nodes, grid operators may increase the availability of chargers without compromising performance. Furthermore, the granularity of the collected data allows the analysis and accounting of consumer usage.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

Various intelligent charging techniques can facilitate the adoption of electric vehicles as future transport.

This work presents an architecture, a tech stack, and implements an Intelligent Electric Vehicle Charging Control Solution that can adapt charging current based on multiple parameters and settings. Two IEVCC models were implemented, the standalone and the multi-charger versions.

The proposed software architecture and the tech stack guided the development process as a basis for the charging solution. The central Manager/Broker manages the installed devices and controls the available power to each charger based on charger priority and remaining available load. When it is impossible to charge multiple chargers simultaneously (charging current less than 6A), the manager will disable lower priority chargers so that remaining available power can be distributed to the remaining chargers. Once available power allows for the reintroduction of the disabled chargers, the manager will authorize the chargers to resume charging. Regarding the manager/broker development, it's important to point out that modularity, code reusability, design patterns, and clean code principles were followed, resulting in a flexible solution with extensibility capabilities and low-effort onboard needs. As a result, adding multiple entities like solar panels or other devices and integrating them without much time-effort or new code is possible. The developed test suite enables considerable logic error mitigation.

The solution is capable of monitoring energy consumption for each client connected to the EVSE while balancing the resources available to prevent the actuation of protection systems (e.g. RCBO).

Ingestion of data is critical, and ingested data will continue to grow. To keep up with vast amounts of rapidly changing data, the implemented stream processing systems are able to ingest, analyze, and persist data continuously. This work presents an architecture and the corresponding tech stack designed to handle massive time-critical data while performing cleansing and transforming operations, then storing it in a cloud database service.

Considering the planned timeline and by following the agile principles of Scrum and the roles and responsibilities for each stage, we were able to develop and deliver according to the Gantt board (Appendix A).

5.2 Future Work

Artificial Intelligence can be used to improve the safety, resilience, and sustainability of electrical power grids and smart cities. Adding AI so IEVCC can predict when to act based on historical data and act immediately can take resource optimization to the next level. It can also help to learn behavioral patterns and adapt charging cycles accordingly.

Because software testing is not limited to what it is used for but has a broader scope for improving software quality. If the software is made to suit the requirements of our business, it is software testing that ensures that the code follows the development plan to make a perfect product. This leads us to the importance of implementing multiple types of software tests like integration, smoke, or regression tests.

Due to the multiple available solutions, it could also be interesting to benchmark the proposed tech stack and compare it with other possible setups.

Anticipating the final product, the proposed solution still requires the development of custom web and mobile applications for its end users.

Developing a custom Embedded Linux Distro that meets the solution's security and technical installation needs may offer significant future benefits.

References

- I. Aghabali, J. Bauman, P. J. Kollmeyer, Y. Wang, B. Bilgin, and A. Emadi. 800-v electric vehicle powertrains: Review and analysis of benefits, challenges, and future trends. *IEEE Transactions on Transportation Electrification*, 7(3):927–948, 2021. doi: 10.1109/TTE.2020.3044938.
- M. A. Ahmed, M. R. El-Sharkawy, and Y.-C. Kim. Remote monitoring of electric vehicle charging stations in smart campus parking lot. *Journal of Modern Power Systems and Clean Energy*, 8(1):124–132, 2020. doi: 10.35833/MPCE.2018.000502.
- D. Anh. Mesh network based on mqtt broker for smart home and iiot factory. *ASEAN Journal of Science and Engineering*, 2(2):173–180, 2021. ISSN 2776-5938. URL <https://ejournal.upi.edu/index.php/AJSE/article/view/39080>.
- T. Aung, A. Thaw, N. Zhukova, T. Man, and V. Chernokulsky. Data processing model for mobile iot systems. *Procedia Computer Science*, 186:235–241, 2021. ISSN 1877-0509. doi: <https://doi.org/10.1016/j.procs.2021.04.143>. URL <https://www.sciencedirect.com/science/article/pii/S1877050921009583>. 14th International Symposium "Intelligent Systems.
- O. Ayan and B. Turkay. Domestic electrical load management in smart grids and classification of residential loads. In *2018 5th International Conference on Electrical and Electronic Engineering (ICEEE)*, pages 279–283, 2018. doi: 10.1109/ICEEE2.2018.8391346.
- P. Baptista, M. Rodrigues, M. Costa, P. Martins, and M. Abbasi. Shared dock-less vehicle location distribution data-set visualizer. In *2021 16th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–4, 2021. doi: 10.23919/CISTI52073.2021.9476220.
- P. Baptista, J. Rosado, M. Silva, F. Caldeira, and F. Cardoso. levcc - a mesh managed network for electric vehicle charging. In *2022 IEEE Vehicle Power and Propulsion Conference (VPPC)*, 2022a. doi: Awaiting-publication.
- P. Baptista, J. Rosado, M. Silva, F. Caldeira, and F. Cardoso. Design of data management service platform for intelligent electric vehicle charging controller - multi-charger model. In *Energy Informatics.Academy Conference 2022*, 2022b. doi: Awaiting-publication.
- N. Brinkel, W. Schram, T. AlSkaif, I. Lampropoulos, and W. van Sark. Should we reinforce the grid? cost and emission optimization of electric vehicle charging under different transformer limits. *Applied Energy*, 276:115285, 2020. ISSN 0306-2619. doi: <https://doi.org/10.1016/j.apenergy.2020.115285>. URL <https://www.sciencedirect.com/science/article/pii/S0306261920307972>.

- L. Calearo, A. Thingvad, K. Suzuki, and M. Marinelli. Grid loading due to ev charging profiles based on pseudo-real driving pattern and user behavior. *IEEE Transactions on Transportation Electrification*, 5(3):683–694, 2019. doi: 10.1109/TTE.2019.2921854.
- F. Cardoso, J. Rosado, M. Silva, C. J. C. Teixeira, C. I. F. Agreira, F. Caldeira, P. Baptista, F. Barreto, and P. G. Pereirinha. Intelligent electric vehicle charging controller. In *2021 IEEE Vehicle Power and Propulsion Conference (VPPC)*, pages 1–5, 2021. doi: 10.1109/VPPC53923.2021.9699236.
- F. Cardoso, M. Silva, and J. Rosado. Comparison of electric vehicle charging efficiency with ievcc and a typical evse. In *Comparison of Electric Vehicle Charging Efficiency with IEVCC and a typical EVSE*, 06 2022. doi: Awaiting-publication.
- S.-M. Chun, H.-S. Kim, and J.-T. Park. Coap-based mobility management for the internet of things. *Sensors*, 15(7):16060–16082, 2015. ISSN 1424-8220. doi: 10.3390/s150716060. URL <https://www.mdpi.com/1424-8220/15/7/16060>.
- M. Costa, M. Rodrigues, P. Baptista, J. Henriques, I. M. Pires, C. Wanzeller, and F. Caldeira. Covid-19 next day trend forecast. In I. M. Pires, S. Spinsante, E. Zdravevski, and P. Lameski, editors, *Smart Objects and Technologies for Social Good*, pages 44–50, Cham, 2021. Springer International Publishing. ISBN 978-3-030-91421-9.
- M. Costa, M. Rodrigues, P. Baptista, C. Wanzeller, P. Martins, and M. Abbasi. Database encryption performance impact on postgresql and mongodb. In J. L. Reis, E. P. López, L. Moutinho, and J. P. M. d. Santos, editors, *Marketing and Smart Technologies*, pages 121–127, Singapore, 2022. Springer Nature Singapore. ISBN 978-981-16-9268-0.
- M. Dias de Assunção, A. da Silva Veith, and R. Buyya. Distributed data stream processing and edge computing: A survey on resource elasticity and future directions. *Journal of Network and Computer Applications*, 103:1–17, 2018. ISSN 1084-8045. doi: <https://doi.org/10.1016/j.jnca.2017.12.001>. URL <https://www.sciencedirect.com/science/article/pii/S1084804517303971>.
- T. Dingsøyr, S. Nerur, V. Balijepally, and N. B. Moe. A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, 85(6): 1213–1221, 2012. ISSN 0164-1212. doi: <https://doi.org/10.1016/j.jss.2012.02.033>. URL <https://www.sciencedirect.com/science/article/pii/S0164121212000532>. Special Issue: Agile Development.
- A. Dubey and S. Santoso. Electric vehicle charging on residential distribution systems: Impacts and mitigations. *IEEE Access*, 3:1871–1893, 2015. doi: 10.1109/ACCESS.2015.2476996.
- EVBox. Evbox home charging, 2022. URL <https://evbox.com/en/ev-chargers/home>.
- EvoCharge. Evocharge evse solution, 2022. URL <https://evocharge.com/>.
- J. Garcia, E. Simó, X. Masip-Bruin, E. Marín-Tordera, and S. Sánchez-López. Do we really need cloud? estimating the fog computing capacities in the city of barcelona. In *2018 IEEE/ACM International Conference on Utility and Cloud Computing Companion (UCC Companion)*, pages 290–295, 2018. doi: 10.1109/UCC-Companion.2018.00070.
- M. Gilleran, E. Bonnema, J. Woods, P. Mishra, I. Doebber, C. Hunter, M. Mitchell, and M. Mann. Impact of electric vehicle charging on the power demand of retail buildings. *Advances in Applied Energy*, 4:100062, 2021. ISSN 2666-7924. doi: <https://doi.org/10.1016/j.aae.2021.100062>.

- org/10.1016/j.adapen.2021.100062. URL <https://www.sciencedirect.com/science/article/pii/S2666792421000548>.
- S. Henbest, H. Kimmel, J. Callens, A. Vasdev, T. Brandily, I. Berryman, J. Danial, and B. Vickers. New energy outlook 2021: Bloombergnef: Bloomberg finance lp, 2020. URL <https://about.bnef.com/new-energy-outlook/>.
- M. Iglesias-Urkiá, A. Orive, and A. Urbieto. Analysis of coap implementations for industrial internet of things: A survey. *Procedia Computer Science*, 109:188–195, 2017. ISSN 1877-0509. doi: <https://doi.org/10.1016/j.procs.2017.05.323>. URL <https://www.sciencedirect.com/science/article/pii/S1877050917309870>. 8th International Conference on Ambient Systems, Networks and Technologies, ANT-2017 and the 7th International Conference on Sustainable Energy Information Technology, SEIT 2017, 16-19 May 2017, Madeira, Portugal.
- H. Isah and F. Zulkernine. A scalable and robust framework for data stream ingestion, 2018.
- M. A. Jamil, M. Arif, N. S. A. Abubakar, and A. Ahmad. Software testing techniques: A literature review. In *2016 6th International Conference on Information and Communication Technology for The Muslim World (ICT4M)*, pages 177–182, 2016. doi: 10.1109/ICT4M.2016.045.
- A. Kanyilmaz and A. Cetin. Fog based architecture design for iot with private nodes: A smart home application. In *2019 7th International Istanbul Smart Grids and Cities Congress and Fair (ICSG)*, pages 194–198, 2019. doi: 10.1109/SGCF.2019.8782400.
- W. Khan, F. Ahmad, and M. S. Alam. Fast ev charging station integration with grid ensuring optimal and quality power exchange. *Engineering Science and Technology, an International Journal*, 22(1):143–152, 2019. ISSN 2215-0986. doi: <https://doi.org/10.1016/j.jestch.2018.08.005>. URL <https://www.sciencedirect.com/science/article/pii/S2215098617315057>.
- A. T. Limón and C. Schulaka. What’s in your tech stack? *Journal of Financial Planning*, 33(2):22–27, 2020.
- J. linru, Z. yuanxing, L. taoyong, D. xiaohong, and Z. jing. Analysis on charging safety and optimization of electric vehicles. In *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, pages 2382–2385, 2020. doi: 10.1109/ICCC51575.2020.9344906.
- B. Marjerison. *The Software Bug Book: Thoughts on Software Quality*. CreateSpace Independent Publishing Platform, mar 2013.
- M. Masdani and D. Darlis. A comprehensive study on mqtt as a low power protocol for internet of things application. *IOP Conference Series: Materials Science and Engineering*, 434:012274, 12 2018. doi: 10.1088/1757-899X/434/1/012274.
- Z. Moghaddam, I. Ahmad, D. Habibi, and Q. V. Phung. Smart charging strategy for electric vehicle charging stations. *IEEE Transactions on Transportation Electrification*, 4(1):76–88, 2018. doi: 10.1109/TTE.2017.2753403.
- M. Nascimento. Vendas de veículos elétricos em novembro de 2020 aumentam 80% em relação a novembro de 2019, Dec 2020. URL <https://www.uve.pt/page/vendas-ve-11-2020/>.

- M. Nour, S. M. Said, A. Ali, and C. Farkas. Smart charging of electric vehicles according to electricity price. *2019 International Conference on Innovative Trends in Computer Engineering (ITCE)*, page 432–437, 2019. doi: 10.1109/itce.2019.8646425.
- R. Pellerin and N. Perrier. A review of methods, techniques and tools for project planning and control. *International Journal of Production Research*, 57(7):2160–2178, 2019. doi: 10.1080/00207543.2018.1524168.
- A. Ramamurthy, S. Saurabh, M. Gharote, and S. Lodha. Selection of cloud service providers for hosting web applications in a multi-cloud environment. In *2020 IEEE International Conference on Services Computing (SCC)*, pages 202–209, 2020. doi: 10.1109/SCC49832.2020.00034.
- Rolec. Rolec serv - charging platform overview, 2020. URL https://www.rolecserv.com/_files/ugd/1667b4_4797b233f7014129823fae3e2b2626ea.pdf.
- M. Saqib, M. M. Hussain, M. S. Alam, M. M. S. Beg, and A. Sawant. Smart electric vehicle charging through cloud monitoring and management. *Technology and Economics of Smart Grids and Sustainable Energy*, 2(1), 2017. doi: 10.1007/s40866-017-0035-4.
- A. Shepelev, C.-Y. Chung, C.-C. Chu, and R. Gadh. Mesh network design for smart charging infrastructure and electric vehicle remote monitoring. In *2013 International Conference on ICT Convergence (ICTC)*, pages 250–255, 2013. doi: 10.1109/ICTC.2013.6675352.
- A. Srivastava, S. Bhardwaj, and S. Saraswat. Scrum model for agile methodology. In *2017 International Conference on Computing, Communication and Automation (ICCCA)*, pages 864–869, 2017. doi: 10.1109/CCAA.2017.8229928.
- I. Stamatescu, R. Mihalache, N. Arghira, I. Fagarasan, and G. Stamatescu. Model for optimized evse deployment in dense urban areas. *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, page 5170–5175, 2020. doi: 10.1109/iecon43393.2020.9254960.
- C. G. von Wangenheim, R. Savi, and A. F. Borgatto. Scrumia—an educational game for teaching scrum in computing courses. *Journal of Systems and Software*, 86(10):2675–2687, 2013. ISSN 0164-1212. doi: <https://doi.org/10.1016/j.jss.2013.05.030>. URL <https://www.sciencedirect.com/science/article/pii/S0164121213001295>.
- V. Vujović, N. Denić, V. Stevanović, M. Stevanović, J. Stojanović, Y. Cao, Y. Alhammedi, K. Jermsittiparsert, H. Van Le, K. Wakil, and I. Radojkovic. Project planning and risk management as a success factor for it projects in agricultural schools in serbia. *Technology in Society*, 63:101371, 2020. ISSN 0160-791X. doi: <https://doi.org/10.1016/j.techsoc.2020.101371>. URL <https://www.sciencedirect.com/science/article/pii/S0160791X20303596>.
- J. Wytrebowicz, K. Cabaj, and J. Krawiec. Messaging protocols for iot systems—a pragmatic comparison. *Sensors*, 21(20), 2021. ISSN 1424-8220. doi: 10.3390/s21206904. URL <https://www.mdpi.com/1424-8220/21/20/6904>.
- Q.-C. Zhong, Y. Wang, and B. Ren. Connecting the home grid to the public grid: Field demonstration of virtual synchronous machines. *IEEE Power Electronics Magazine*, 6(4): 41–49, 2019. doi: 10.1109/MPPEL.2019.2946700.

Appendix A

Gantt diagram

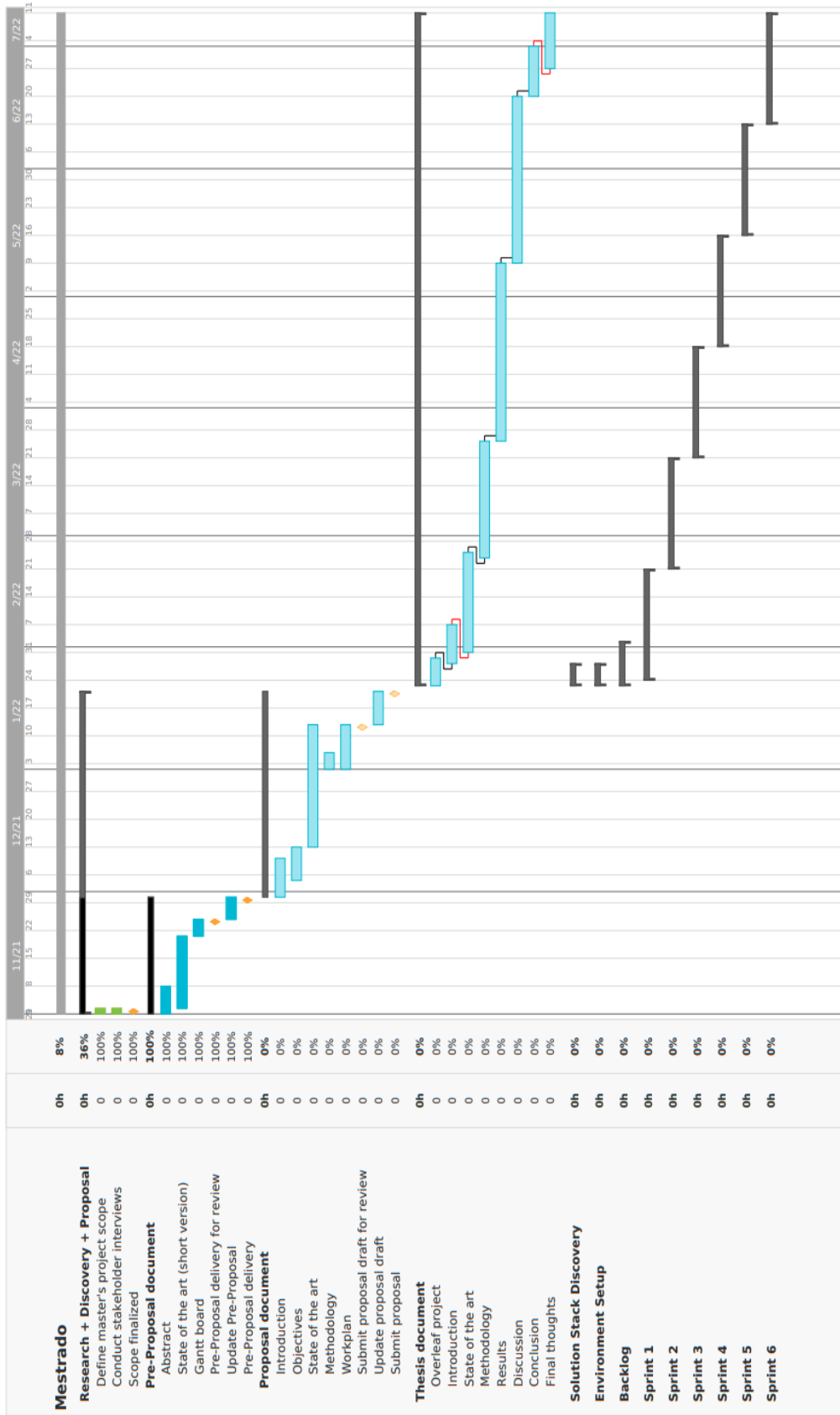


Figure 21: Gantt diagram

Appendix B

Vehicular Power and Propulsion 2021 (VPPC 2021) Article

Intelligent Electric Vehicle Charging Controller

Filipe Cardoso
Viseu Polytechnic - ESTGV
and INESC Coimbra
Viseu, Portugal
ORCID: 0000-0002-3916-5182

J. Rosado
Coimbra Polytechnic - ISEC
and INESC Coimbra
Coimbra, Portugal
ORCID: 0000-0001-5610-7147

Marco Silva
Coimbra Polytechnic - ISEC
and INESC Coimbra
Coimbra, Portugal
ORCID: 0000-0003-3142-1679

C. J. Coelho Teixeira
Coimbra Polytechnic - ISEC
and INESC Coimbra
Coimbra, Portugal
ORCID: 0000-0003-3668-5710

C. I. Faustino Agreira
Coimbra Polytechnic - ISEC
and INESC Coimbra
Coimbra, Portugal
ORCID: 0000-0002-2444-2925

Filipe Caldeira
Viseu Polytechnic - ESTGV
and CISED - Research
Centre in Digital Services
Viseu, Portugal
ORCID: 0000-0001-7558-2330

Pedro Baptista
Viseu Polytechnic - ESTGV
Viseu, Portugal
ORCID: 0000-0003-3666-6552

Francisco Barreto
Coimbra Polytechnic - ISEC
Coimbra, Portugal
ORCID: 0000-0001-7044-5433

Paulo G. Pereirinha
Coimbra Polytechnic - ISEC
and INESC Coimbra
Coimbra, Portugal
ORCID: 0000-0002-3514-4544

Abstract—For domestic consumers, electricity tariffs usually have two components: one is related to the maximum available current/total power (billed in €/day) and the other concerns to the energy consumption (€/kWh). The main switchboard current is usually limited, according to the contracted power level, by way of automatic switches. To avoid main switchboard tripping by current limit violation, Electric Vehicle (EV) owners may decide to increase their contracted power (and the energy bill) or to adopt charging strategies that limit the global consumption (EV plus house needs) to the contracted current/power. In this paper, an Intelligent Electric Vehicle Charging Controller (IEVCC), allowing to use the contracted power to the maximum extent, is presented. A set of user configurable parameters can be used to define the controller behavior, in order to prevent a full switch-off. Two versions are described: a single user version that can be used at private houses and a mesh version that can be used in multi apartment buildings, providing information about consumed energy, time of use, costs and past bills.

Index Terms—EVSE, Electric Vehicles, Intelligent Charging, Home Consumption, Load Management.

I. INTRODUCTION

The number of EV users is increasing all over the world. Be it by the low value of Total Cost Ownership (TCO), low cost per km, environment reasons or government carbon taxes, the fact is that EV sales keep increasing from year to year [1], [2]. As the number of EV users increases, the need for charging points also increases. Residential building's electric infrastructure was not designed for large consumption, and usually only a normal 16A Schuko plug is available in each particular car parking place. By thermal and security reasons, the maximum recommended current is only 10A and, then, the charging capacity becomes very limited. Most

manufacturers are increasing the capacity of their batteries, to improve the autonomy of EV. The charging time, for 10-16A connections, will increase to unacceptable values. Then, modern EV, equipped with larger capacity batteries requiring up to 32A per phase, cannot be charged, at all, or, if the charge is possible, then the low current makes the charging process to extend by several hours.

Residential buildings, where multiple families share some common spaces, are usually unprepared to install high current-level charging devices. If a dedicated cable is connected directly to the building's main electrical board, higher charging currents (up to the main board design capacity) can be used. Some manufacturers offer wall box chargers that allow charging at a higher current rate. Those pieces of equipment usually have a fixed charging power, which is frequently set very close to the house's maximum contracted current/power. If the EV charging current is high and other loads are connected, a tripping order may be triggered, and the complete installation will be switched off. Priority can be defined for the charging process or non-EV loads, executing load shedding plans, or limiting the EV charging rate to the maximum available capacity, which corresponds to the contracted power minus the total non-EV power in use.

A smart EVSE project is presented in [3], however, it is not clear from the document what is the smart part of the system and how it is performed or where it applies. A Stackelberg game model is used in decentralized electric vehicle supply equipment in [4]. Here renewable energies are used to charge the vehicles, and vehicle to grid (V2G) is considered.

Although there are several adjustable EVSE in the market, most of them only allow to manually adjust the charging

power, according to 4 of 5 predefined set points, adjusting the maximum current that the vehicle can draw from the network. This paper presents an intelligent electric vehicle charger controller capable of auto-adjusting the charging power (current) based on a set of parameters, namely, the standing charge, the house consumption, and renewable energy production. This process is defined for the signal present at one pin on the socket on the SAE J1772 standard [5].

This paper is organized as follows: after this Introduction, the IEVCC is described in Section II. In Section III the experimental results are shown, and the Conclusions are presented in Section IV.

II. INTELLIGENT ELECTRIC VEHICLE CHARGING CONTROLLER

This work describes two versions of IEVCC systems, namely a Standalone version and a Mesh version. This IEVCC is built on an ESP board. Both versions allow communications over a wireless network to provide information about the charging status and receive information from the main system. The ESP32-POE supports an Ethernet network, which can be used when the building is equipped with this type of infrastructure. The choice of these two boards allows the development of low-cost systems, enabling the users' fast adoption.

The systems presented in this paper implement the open standard in the industry (OCPP [6]–[8]) that enable charging points to communicate with a central system. Public chargers commonly use the OCPP protocol. Since one of the project's objectives is to foster the development of a network of public or private chargers, it was decided to profit from the capabilities of OCPP and implement it on both versions.

A. Standalone Version

The standalone version targets a single-user environment (personal use) where the user has his private parking spot with charging energy provided from his house electric access point. This version works in Intelligent mode by looking at the house consumption and adjust the charging current according to the house consumption. It also has the advantage of work on demand (manual mode), where the user can connect it to a wall socket and define the charging current manually. This last mode makes it portable and independent from any system reading the house consumption. Fig. 1 shows a diagram of the system, which is composed by the:

- Consumption monitor hardware (a);
- Consumption monitor system - broker (b);
- Intelligent Electric Vehicle Charging Controller (c);
- Communications infrastructure;

The consumption monitor hardware is a system that monitors the power consumption of the house/building by measuring the Voltage, Current and Active Power at a timed interval. It is built using a PZEM-004T V3.0 board, based on the Vango V98xx IC [9]. The measured values are sent to the consumption monitor system that stores them on a database, allowing the user to retrieve information about

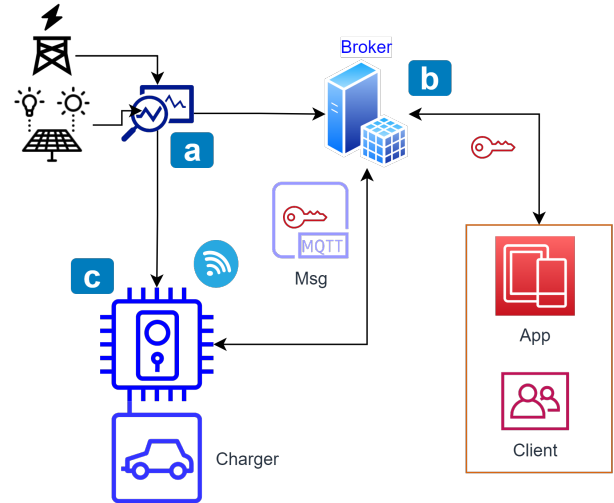


Fig. 1. Diagram of the Standalone Version.

the electricity costs of different home devices. The current information is also sent by MQTTS [10] protocol to a broker so that the IEVCC can adjust the charging current based on pre-configured information about the contracted power. The system also allows taking advantage of renewable sources when they are present (*e.g.*, solar panels), sending only the available power from these sources to the charger instead of being injected on the grid.

B. Mesh Version

The Mesh version aims to be used in condominiums, where the users do not have a garage spot with energy provided from their fraction. This version can be provided to the condominium administration and installed by a certified company or an EV charging network operator. Fig. 2 shows a diagram of the system, which is composed by the:

- Consumption monitor hardware (a);
- Broker (b);
- Intelligent Electric Vehicle Charging Controller + monitor energy meter (c);
- Communications infrastructure;

Although being very similar to the standalone, the operation of the mesh version must deal with multiple users, possibly with different needs or rights, as well as with different infrastructure conditioning factors, like cable thermal limits or maximum available power from the energy provider. Then, the loading balance process is more challenging.

Furthermore, other challenges are to decide which user has the right to more power or if all users can charge simultaneously. In order to charge, all users will have an NFC [11] card that will activate the charger. This card will identify the user on the system, where an account defines the grants access he has (maximum power he can charge, priority over other users, etc.). Based on the user profile, the number of users charging at each moment, and the available power, the system performs adjustments to all chargers, and in the worst scenario,

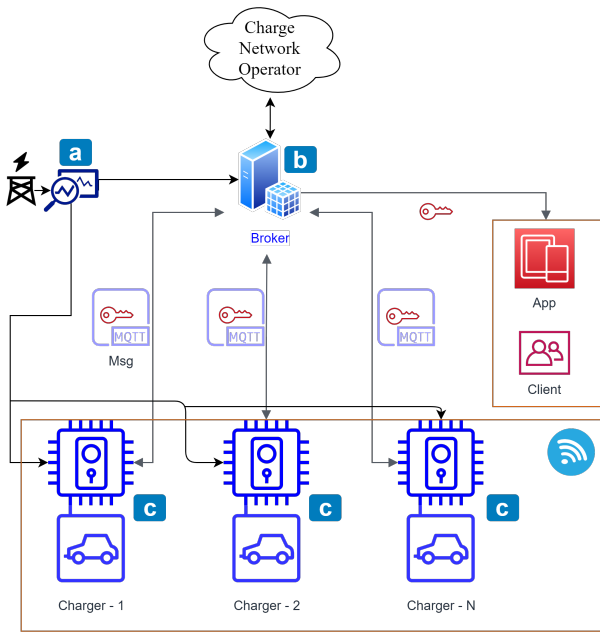


Fig. 2. Diagram of the Mesh Version.

the user may not be authorized to charge. The chargers must communicate to the broker all information about the charging sessions (usage time, used energy) to apply charging fees.

C. Standalone user interface

In this version, the installation is simple. The first time the user connects the IEVCC to the power, the system acts as an Access Point, with a network SSID "CCVEI" (Portuguese acronym for IEVCC). The user can connect to this network using a mobile phone or a computer.

After connecting to it, it can access the configuration pages on a predefined IP address. When it does so, it is presented with the web page shown in Fig. 3, where several fields can be seen, such as:

- Amps: Here the user can set the charging current when using the IEVCC outside his home or if it wants to use the IEVCC in a normal mode.
- EVSE: – Amps. Here the actual charging current value is shown, be it in the normal mode, or the intelligent mode.
- HOME: – AMPS. The actual house instantaneous current consumption is shown. This value is received from the consumption monitor system by MQTTS.
- A set of icons that give information about the wi-fi connection status, the EVSE status, the operating mode, and if a schedule timer is active.

The next tab (Fig. 4) allows the user to configure items related to the home user wifi network and MQTT server parameters.

Only after the user configures these items, the IEVCC can enter the intelligent mode and adjust the charging current automatically.

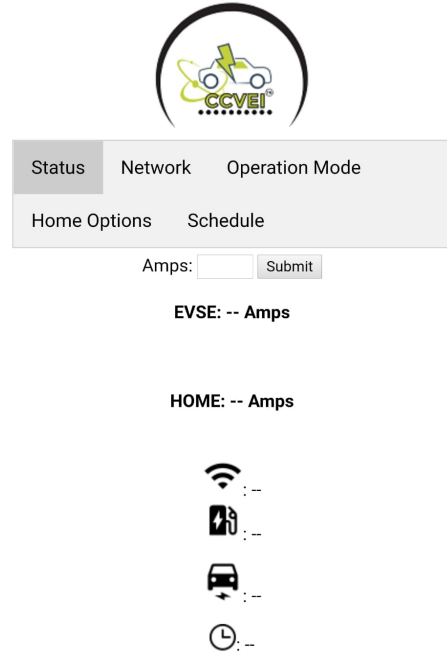


Fig. 3. IECCV Status

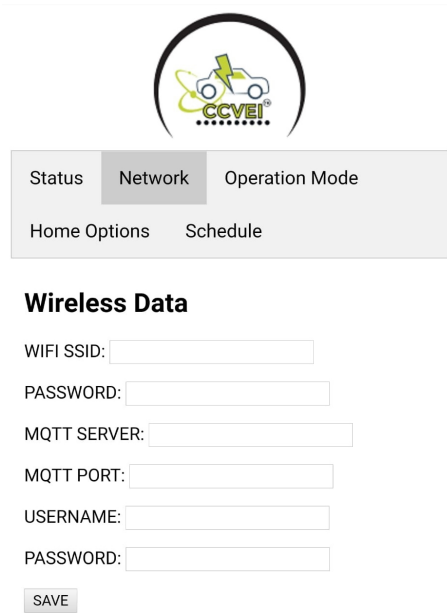


Fig. 4. IEVCC Network Configuration webpage

In "Operation Mode" tab (Fig. 5) the user can select between two options:

- Intelligent mode: in this mode, the IEVCC adjusts the charging current (power) based on the home consumption and the contracted power.
- Manual mode: in this mode, the IEVCC operates at a fixed charging current (power) set by the user.

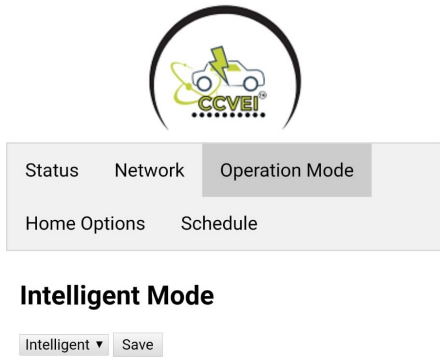


Fig. 5. IEVCC Operation Mode webpage

The Home Options tab (Fig. 6) is where the user can set the contracted power.

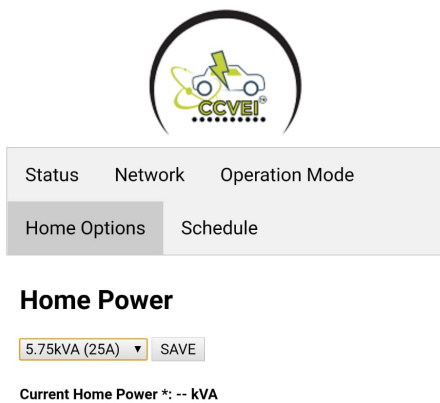


Fig. 6. Home Options tab

III. EXPERIMENTAL RESULTS

A series of tests were performed, including a full charge session. All these tests were carried out in a real environment, in which the parking spot has an available plug with energy provided directly from the apartment and while the residents continued performing their typical tasks (*e.g.*, cooking, turning on washing machines) without any restrictions. The IEVCC has dynamically adapted the available power for charging without the need for human intervention.

Fig. 7 shows the full charging session (from 33% up to 100%) of an electric vehicle (a 2015 24 kWh Nissan Leaf) using a first version of the standalone IEVCC. In this case, the user has a contracted power of 5.75 kVA, equivalent to a maximum current of 25 A. In this house (and today, in many houses in Portugal), the installed meter is an intelligent version that cuts off the power much faster than a Residual Current Breaker with Over-Current (RCBO) in case the current equivalent to the contracted power is overpassed. Thus, a 3 A buffer was defined to avoid total house current to overpass the defined cut current on the intelligent meter.

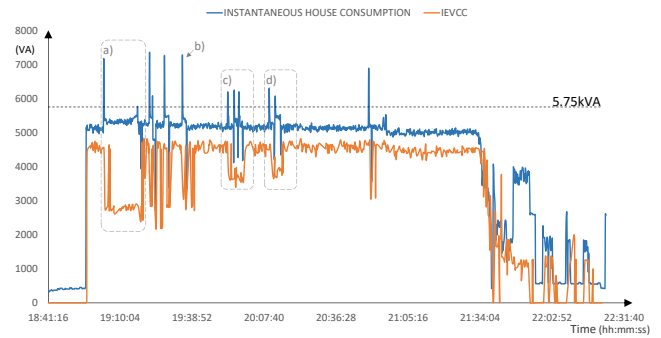


Fig. 7. Charging session data

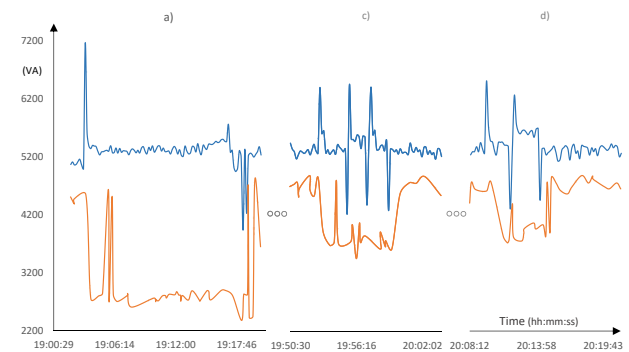


Fig. 8. Charging session data details of Fig. 7

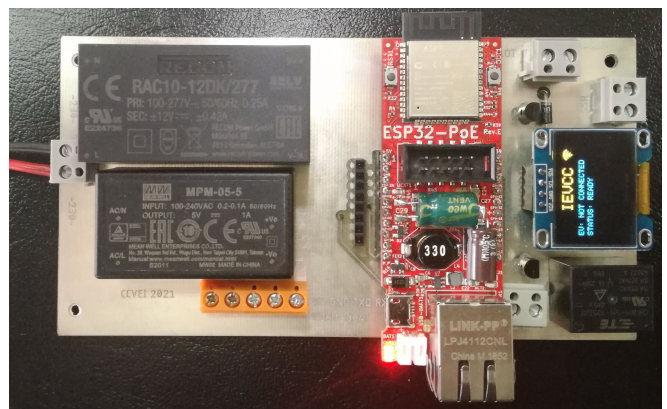


Fig. 9. Prototype of the IEVCC

Whenever the instantaneous consumption of the house exceeds the limit value, the IEVCC automatically adjusts the charging current (which is related to the charging power) so that the sum of the current drained by home appliances working at that time and the vehicle does not exceed the maximum defined value. It is possible to see in the graph that the established value is momentarily exceeded although immediately corrected (moment a, b, c and d in Fig. 7). A zoom of some of these segments are presented in Fig. 8. Here, it is well visible the moments where this peak occurs. These peaks may happen because an appliance in the house

was turned on and requested enough current so that the total current drawn exceeds the maximum allowed. Since the consumption monitor system does not measure the values in real-time, there is always a tiny delay leading to a rapid consumption increase. When the consumption monitor system makes a new measure, it instantly sends this new value to the IEVCC, which immediately intervenes, lowering the charging current available to the electric vehicle. In a more extreme case, the IEVCC can interrupt the EV charging process until the available power is sufficient to continue the charging session (reference to J1772 standard). It should be noticed that although the intelligent meter at the home user is faster than the RCBO, it is not also a real-time process, so the delay that our system takes to act is not significant enough for the intelligent meter to cut the current.

Fig. 9 shows a version of the prototype developed, and used in the tests, which was built in-house at ISEC.

IV. CONCLUSIONS

In this work, an Intelligent Electric Vehicle Charging Control able to dynamically adapt the vehicle charging current based on a set of parameters was presented. This system is not limited to work in this mode and can act as a simply EVSE, when used on travels. Two versions of the IEVCC were described, and the results for the standalone version were presented and discussed. The Mesh version is in its final development phase, and the authors expect to present results from this version in the near future. As future work, the introduction of AI is being considered so that the IEVCC can predict when to act based on historical values and react instantaneously. This version could also overcome potential intermittent connection problems between the home consumption monitor system and the IEVCC.

ACKNOWLEDGEMENTS

The authors would like to thank Prio Energy (www.prio.pt/pt/) for the partnership in this project. This

work is partially funded by National Funds through the FCT - Foundation for Science and Technology, I.P., within the scope of the projects UIDB/00308/2020, UIDB/05583/2020 and MANaGER (POCI-01-0145-FEDER-028040). Furthermore, we would like to thank the Research Centre in Digital Services (CISeD) and the Polytechnics of Viseu and Coimbra for their support.

REFERENCES

- [1] Liikennevirta Oy (Ltd.), "The global electric vehicle market in 2021 - virta." Virta.global, 30-Aug-2019. [Online]. Available: <https://www.virta.global/global-electric-vehicle-market>. [Accessed: 11-Apr-2021].
- [2] International Energy Agency. (Apr. 2021). Global EV Outlook 2021. [Online]. Available: <https://www.iea.org/reports/global-ev-outlook-2021>. [Accessed: 06-May-2021].
- [3] C. Zhu, "System-cost-optimized smart EVSE for residential application: Final technical report including manufacturing plan," Office of Scientific and Technical Information (OSTI), 2015.
- [4] T. G. Alghamdi, D. Said, and H. T. Mouftah, "Decentralized game-theoretic scheme for D-EVSE based on renewable energy in smart cities: A realistic scenario," *IEEE Access*, vol. 8, pp. 48274–48284, 2020.
- [5] SAE International. J1772: SAE Electric Vehicle and Plug in Hybrid Electric Vehicle Conductive Charge Coupler; SAE International: Warrendale, PA, USA, 2016.
- [6] A. Rodriguez-Serrano, A. Torralba, E. Rodriguez-Valencia, and J. Tarifa-Galisteo, "A communication system from EV to EV Service Provider based on OCPP over a wireless network," in *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, 2013.
- [7] J. Schmutzler, C. Andersen, and C. Wietfeld, "Evaluation of OCPP and IEC 61850 for smart charging electric vehicles," *World Electric Veh. J.*, vol. 6, no. 4, pp. 863–874, 2013.
- [8] C. Alcaraz, J. Lopez, and S. Wolthusen, "OCPP protocol: Security threats and challenges," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2452–2459, 2017.
- [9] Vango Technologies, Inc., "V98XX Datasheet", Available: <http://www.vangotech.com/uploadpic/164299406070.pdf>. [Accessed: 12-2021].
- [10] "MQTT - The Standard for IoT Messaging" 7 March 2019. [Online]. Available: <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.pdf>. [Accessed 10-Apr-2021].
- [11] Kevin Curran, Amanda Millar, Conor Mc Garvey, (2012) "Near Field Communication", *International Journal of Electrical and Computer Engineering (IJECE)*, Vol.2, No.3, pp. 371-382.

Appendix C

Vehicular Power and Propulsion 2022 (VPPC 2022) Article

IEVCC - A Mesh Managed Network for Electric Vehicle Charging

Pedro Baptista
Viseu Polytechnic - ESTGV
Viseu, Portugal
ORCID: 0000-0003-3666-6552

J. Rosado
Coimbra Polytechnic - ISEC
and *INESC Coimbra*
Coimbra, Portugal
ORCID: 0000-0001-5610-7147

Marco Silva
Coimbra Polytechnic - ISEC
and *INESC Coimbra*
Coimbra, Portugal
ORCID: 0000-0003-3142-1679

Filipe Caldeira
Viseu Polytechnic - ESTGV
and *CISeD - Research*
Centre in Digital Services
Viseu, Portugal
ORCID: 0000-0001-7558-2330

Filipe Cardoso
Viseu Polytechnic - ESTGV
and *INESC Coimbra*
Viseu, Portugal
ORCID: 0000-0002-3916-5182

Abstract—The implementation and management of Electric Vehicle (EV) charging points in parking spaces (whether in multifamily condominiums or in private company parking lots) presents a challenge, since the available contracted power may be insufficient for the building devices needs and for EV chargers to work simultaneously. The available power is limited to a fixed value generally determined by the expected peak power consumption of the building. An increase in the contracted power leads to unwanted costs and, sometimes, this is not possible without a complete rework of the electrical installation at even higher costs. The available power is not fully used all the time, since not all the buildings devices are always working simultaneously. These spare power can be monopolized by a mesh network of chargers, taking advantage of the full contracted power without increasing costs. In this work we present a manager of a mesh network of chargers that can distribute the available power to an array of chargers based on several conditions, e.g. load balancing, priority of chargers or charging.

Index Terms—EVSE, Electric Vehicles, Intelligent Charging, Mesh Network and Load Management.

I. INTRODUCTION

The sales of EV have increased over the years, specially in last year. Even though that the COVID pandemic has slowdown the sales, last year there was an increase of 108% when compared with 2020 [1]. Studies shown that EV sales will continue to increase, even if at lower rates [2]–[4], it must be taken in account that the values are always related to the previous years. The legislation adoption by the European Commission on 14 July 2021 intends to achieve climate neutrality by 2050, including an intermediate target of an at least 55% net reduction in greenhouse gas emissions by 2030 [5]. The latest unfortunate events on the political context have putted even more pressure on the reduction on the petrol dependence in Europe. With EVs being two to four times more efficient than vehicles with internal combustion engines and the increase in renewable energy sources it is also possible

to generate significant reductions in greenhouse gas emissions and this leads to an increase on the EV adoption. Also, several car makers are stating in their plans they will become fully electric car producers in the horizon time of 2030-2035.

Even with such an increase in sales, most people are reticent in buying an EV, since they don't own a closed parking spot in their residence with an electric connection to their main electric installation. Even with the increase of public chargers, the number and prices demotivate most people, since most times they aren't close to their residence or working places or there is no guarantee that they are not in use when those people need them. One solution goes through the use of the buildings common electric infrastructure (generally used to power lights, elevators and other common devices) to connect vehicle chargers/EVSE. However such infrastructure has a limited power and it would not be difficult to reach the maximum available power just by installing one or two standard chargers with a reasonable power, so a good efficiency charge is attainable [6] and with the risk of tripping the main switchboard.

In this work we present a solution that allows the installation of a high number of the intelligent charger previously developed and presented in [7]. In that work we described the operation of what we called as Intelligent Electric Vehicle Charger Controller (IEVCC) and presented results of its operation when working as the only charger operating on the same electric circuit. At the time, a mesh version was also described, although briefly. Here we review again the key idea of this version and describe its operation, as results obtained. As far as we know and searched, there's only on the market a system that can operate in a similar way (designed as SMART EVSE [8]), but the description is not very clear if it can establish priorities between chargers and it's limited to four chargers. Also the communication between what is called master and the slaves is wired, which can present a problem in old installations where it's highly probable there's no room for

extra wiring passage.

This paper is organized as follows: after this Introduction, the mesh operation of IEVCC is described in Section II. In Section III the experimental results are shown, and the Conclusions are presented in Section IV.

II. IEVCC - MESH VERSION

Fig. 1 shows a diagram of the mesh version system, which is composed by the:

- Consumption monitor hardware (a);
- Broker/Manager (b);
- Intelligent Electric Vehicle Charging Controller + energy meter (c);
- Communications infrastructure;

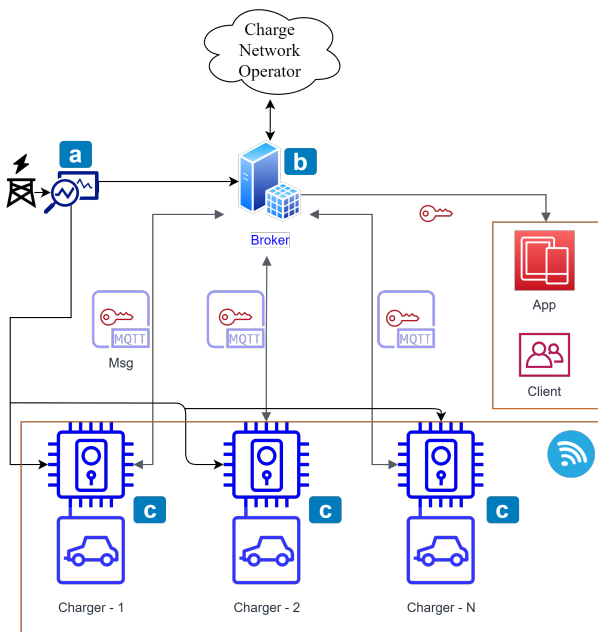


Fig. 1. Diagram of the Mesh Version.

A. Consumption monitor hardware

The consumption monitor (Fig. 2) is responsible to measure the total power consumption of the building and report these values to the Manager using MQTT [9]. It's built using ESP8266 micro-controller [10], a PZEM-0004T V3.0 and a monochrome OLED screen that gives instantaneous information about the measured values and the IP address of the energy monitor.

The first time the system is connected to power, it needs some configuration, which can be done in a web-page provided by the builtin server. To access this, the first step is to connect to the SSID emitted by the consumption monitor hardware (EnergyMeter) using a smartphone and access to a predefined IP address, where a page has the one shown in Fig. 3 is rendered. Here information about the current values being measured can be seen and the IP address of the device on



Fig. 2. Consumption monitor hardware/Energy Meter

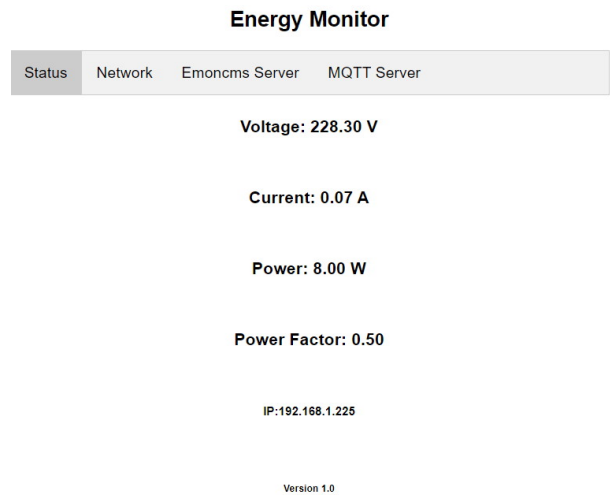


Fig. 3. Energy Monitor Status Web page

the local network is also shown, if the device is connected to the local network.

The Network tab (Fig. 4) is where information about the local network can be entered, so the device can connect to the local network.

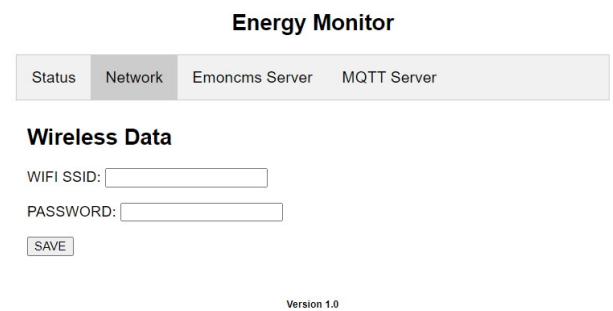


Fig. 4. Energy Monitor Network Settings tab

The Emoncms tab (Fig. 5) allows the configuration of a

emoncms server settings (address and api write key) so the Energy Monitor can send information to that server. Emoncms [11] is an open source software that allows the user to keep track of the consumption of his devices and can calculate the user costs with energy, gains/savings with photovoltaic production, etc.

Fig. 5. Energy Monitor Emoncms Settings tab

The last tab (MQTT server, Fig. 6 is where the MQTT server configurations can be done. This server is usually the device identified as Manager/Broker and is described on the next section.

Fig. 6. Energy Monitor MQTT Settings tab

B. Manager/Broker

The management of all entities involved in the charging process is carried out by the "Manager". Dependent on the MQTT broker, it subscribes to all existing MQTT topics and manages clients (chargers and electrical energy meter). The settings allow us to limit the number of chargers operating at the same time, as well as set priorities for the chargers. In the performed tests, priority was given to the charger that was connected first.

According to the established protocol, the "Manager" will be the first entity to connect to the MQTT broker. By subscribing to a connection topic, the chargers and electrical counters must announce themselves. After they have been properly identified and given permission to access, they will be added to the "Manager" registers. The "Manager" will then contact each of the parties through a separate MQTT topic.

Through the aforementioned message protocol, it is possible to announce through the MQTT topic the connection or charging intention, in the case of a charger. In the case of the electric meter, it is important to report the energy value that is being consumed globally. This will allow the "Manager" to calculate the difference between the contracted power and the amount of energy value being used, which represents the amount of energy value that can be shared among all chargers that are actually charging. These messages exchange is depicted in Fig. 7.

The time for updating this data can also be customized to ensure that the contracted limits are not exceeded and thus cause power outages. This update of the total energy available to the chargers also makes it possible to balance the charging within the minimum limits for each charger. If each charger has less than 6A available, one or more chargers will be put on standby (waiting for a charging signal), increasing the availability for each charger in charge and making sure that the charge stays above the minimum required.

The "Manager" is also responsible for ensuring the persistence of all generated data. Locally in a database, which ensures only local operation and also sends all activity to a cloud database storage.

C. IEVCC + Energy Meter

The third element of the mesh version is composed by at least one IEVCC, which has already been described in [7] and a simple energy meter composed by up to 3 PZEM-004T V3.00 [11]. This version of the IEVCC is a little bit more complex, because includes the energy meter and can include a NFC card reader [12]. The energy meter allows to account costs to the user that used the charger. The NFC card allows different users to use the same charger, when available, setting priorities on the charger to the manager.

III. EXPERIMENTAL RESULTS

At this phase, the experimental results were conducted in a simulation environment, where besides the manager, 3 chargers that replicate some of the cycles already performed and described in [7] were also simulated.

The ability to adjust charging power based on global consumption remains, and aggressive update intervals (<10 second) are not required.

In figure Fig. 8 for a sample interval, it is possible to check the load balance with 3 connected chargers. Changes in consumption that aren't related to charging a vehicle cause a quick change in how much energy each charger can use without going over the maximum load (30 A) that was previously set. This happens faster than a Residual Current Breaker

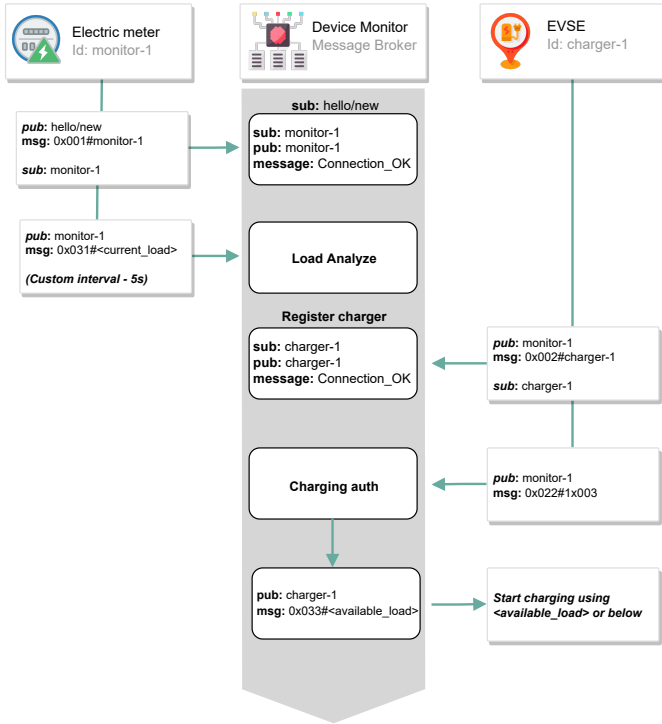


Fig. 7. Protocol time-lapse

with Over-Current (RCBO) in case the current equivalent to the contracted power is over-passed. Several situations are depicted in Fig. 8, which are described below. Priority is given to chargers by order of registration/request for charging. In this example, charger 1 was the first to register, followed by charger 3 and charger 2 was the last one. In each situation, this is what is happening:

- Here charger 1 is already registered and has request to start charging. The available current to charging is 19.9A, but since chargers only work in integer values (between 6A and 32A), the charger starts charging with 19A. At the same time, the consumption of the house appliances raised to 12A, leaving only 18A for the charger, and thus in next cycle the charging current is reduced to 18A.
- In the previous instant to this moment, charger 3 registered in the network and request to start charging. So at this moment, the available power was divided equally by the 2 chargers, and both can charge at 9A.
- Here the building consumption raised and the contracted power was over-passed, so the available power to charge must be reduced.
- As a result of previous over-passing of the contracted power, the charging power of both chargers was reduced.
- Charger 2 (that probably registered on the manager previously to this instant) requested to start charging. Since the available power allows that all

chargers can be used at the same time, charger 2 can start charging.

- In the instant before this, the building consumption reduced, so the available power to each charger increased and the chargers increase the charging current to 9A
- In the previous sampling time, the building consumption increased again and the available power is not enough for all chargers to charge at the minimum power (6A of current), so charger 2 (the last charger to register in the manager) must be put in standby mode.
- Once again the building consumption increases and the remaining power only allows one charger to work. So at next step, only charger 1 is working.
- One more time, the building consumption increased (registered at previous time step) and charger 1 also was ordered to stop charging, since there is no available power to charge at the minimum allowed by the standard.
- In previous time step the building consumption decreased enough to allow charger 1 to start charging again.
- Another reduction in the building consumption made it possible to charger 2 enter in charging mode again. At this moment, the building consumption slightly increased, so in next time step the chargers reduced the charging current from 7A to 6A.
- Here charger 1 has finished the charging session and the building consumption has also decreased, so charger 3 could increase the charging power and charger 2 could resume the charging session.

IV. CONCLUSIONS

In this work, the mesh version of the IEVCC was presented. This system allows the installation of multiple chargers in a shared private parking space taking advantage of the available contracted power without the need of an increase in the maximum peak power. The installed chargers are managed by a central manager/broker that controls the available power to each charger in function of charger priority and the remaining available power. At the extreme case, when is not possible to maintain multiple chargers, charging at the same time (charging current bellow 6A), the manager will disable lower priority chargers in order to distribute the remain available power to the remaining chargers. Once the available power allows the reintroduction of the disabled chargers, the manager will authorize the chargers to recover the charging process.

ACKNOWLEDGEMENTS

This work is partially funded by National Funds through the FCT - Foundation for Science and Technology, I.P., within the scope of the projects UIDB/00308/2020, UIDB/05583/2020. Furthermore, we would like to thank the Research Centre in

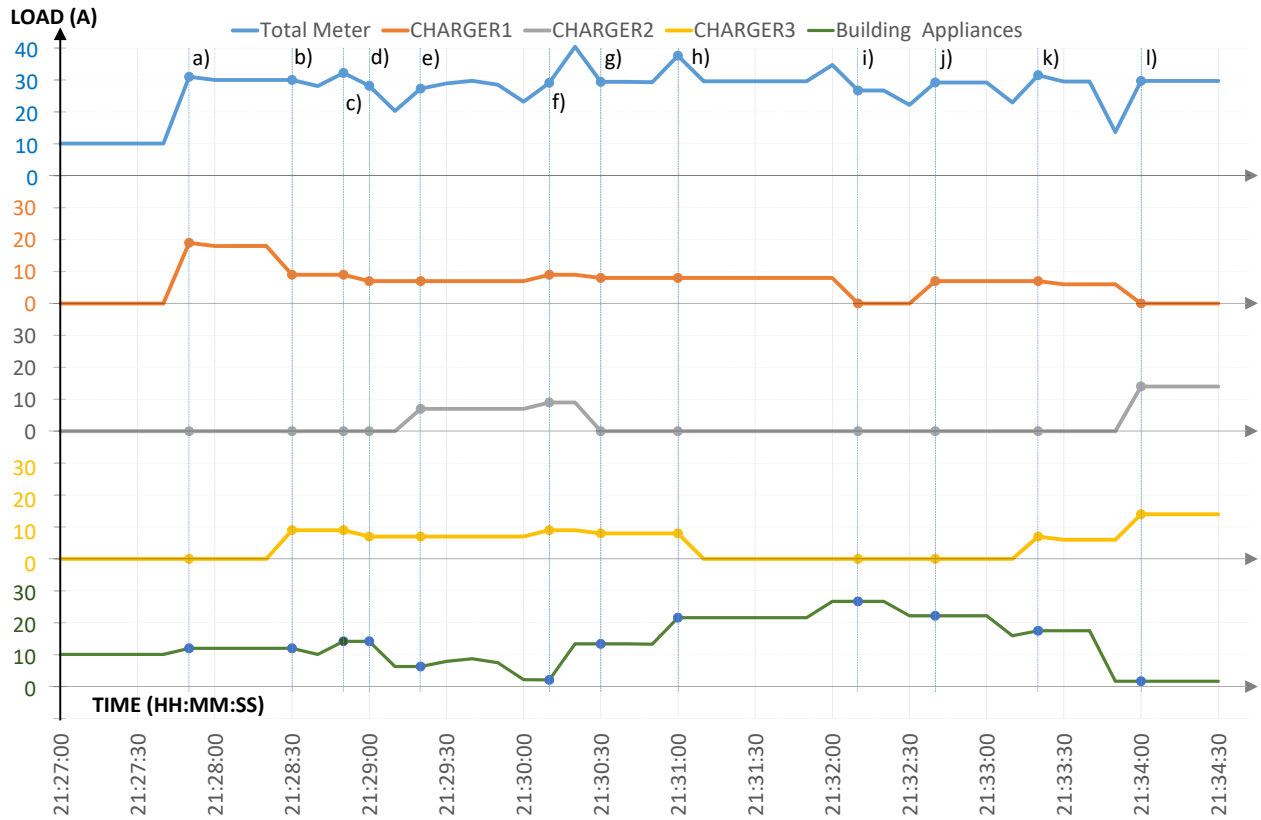


Fig. 8. Mesh Charger Network Simulation

Digital Services (CISeD) and the Polytechnics of Viseu and Coimbra for their support.

REFERENCES

- [1] Liikennevirta Oy (Ltd.), "The global electric vehicle market in 2021 – virta," Virta.global, 30-Aug-2019. [Online]. Available: <https://www.virta.global/global-electric-vehicle-market> [Accessed: 11-03-2021]
- [2] Muratori, M., Alexander, M., Arent, D., Bazilian, M., Cazzola, P., Dede, E., Farrell, J., Gearhart, C., Greene, D., Jenn, A., Keyser, M., Lipman, T., Narumanchi, S., Pesaran, A., Sioshansi, R., Suomalainen, E., Tal, G., Walkowicz, K. & Ward, J. The rise of electric vehicles—2020 status and future expectations. *Progress In Energy*. **3**, 022002 (2021,3), <https://doi.org/10.1088/2516-1083/abe0ad>
- [3] Wu, M. & Chen, W. Forecast of Electric Vehicle Sales in the World and China Based on PCA-GRNN. *Sustainability*. **14** (2022), Available: <https://www.mdpi.com/2071-1050/14/4/2206>
- [4] Graham, J. & Brungard, E. Consumer Adoption of Plug-In Electric Vehicles in Selected Countries. *Future Transportation*. **1**, 303-325 (2021), <https://www.mdpi.com/2673-7590/1/2/18>
- [5] International Energy Agency. (Apr. 2021). Global EV Outlook 2021. [Online]. Available: <https://www.iea.org/reports/global-ev-outlook-2021> [Accessed: 06-May-2021]
- [6] Cardoso, F., Silva, M. & Rosado, J. Comparison of electric vehicle charging efficiency with IEVCC and a typical EVSE. *2022 Eceee Summer Studies On Energy Efficiency Pending publishing*. (2022,6)
- [7] Cardoso, F., Rosado, J., Silva, M., Teixeira, C., Agreira, C., Caldeira, F., Baptista, P., Barreto, F. & Pereirinha, P. Intelligent Electric Vehicle Charging Controller. *2021 IEEE Vehicle Power And Propulsion Conference (VPPC)*. pp. 1-5 (2021), <https://doi.org/10.1109/VPPC53923.2021.9699236>
- [8] Stegen Electronics Smart EVSE. , <https://www.smartevse.nl/> [Accessed: 2022-5-10]
- [9] "MQTT - The Standard for IoT Messaging" 7 March 2019. [Online]. Available: <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.pdf> [Accessed: 2022-03-14]
- [10] Expressif, Inc. Expressif 8266. (2022,5), https://www.espressif.com/sites/default/files/documentation/esp8266-technical_reference_en.pdf, [Accessed: 2022-1-16]
- [11] Emoncms, open-source web-app for processing, logging and visualising energy, temperature and other environmental data, available at: <https://emoncms.org/>, [Accessed: 2022-5-11]
- [12] Kevin Curran, Amanda Millar, Conor Mc Garvey, (2012) "Near Field Communication", *International Journal of Electrical and Computer Engineering (IJECE)*, Vol.2, No.3, pp. 371-382.

Appendix D

Energy Informatics.Academy Conference 2022 (EI.A 2022) Article

RESEARCH

Design of Data Management Service Platform for Intelligent Electric Vehicle Charging Controller - Multi-charger Model

Pedro Baptista¹, José Rosado^{2,4}, Filipe Caldeira^{1,3} and Filipe Cardoso^{1,4*}

*Correspondence:

fcardoso@estgv.ipv.pt

¹Viseu Polytechnic - ESTGV, Viseu, Portugal

⁴INESC Coimbra, Coimbra, Portugal

Full list of author information is available at the end of the article

Abstract

The electric charging solutions for the residential market imply, in many situations, an increase in the contracted power in order to allow to perform an efficient charging cycle that starts when the charger is connected and ends when the VE battery is fully charged. However, the increase in contracted power is not always the best solution for faster and more efficient charging. With a focus on the residential market, the presented architecture is suitable for single-use and shared connection points, which are becoming common in apartment buildings without a closed garage, allowing for sharing the available electrical connections to the grid. The multi-charger architecture allows using one or several common charging points by applying a mesh network of intelligent chargers orchestrated by a residential gateway. Managing the generated data load involves enabling data flow between several independent data producers and consumers. The data stream ingestion system must be scalable, resilient, and extendable.

Keywords: EVSE, Electric Vehicles, Intelligent Charging, Load Management, Mobility, Mesh, Data Management, Fog Computing

Acronyms

API Application Programming Interface. [3](#), [4](#), [6](#), [7](#)

AWS Amazon Web Services. [7](#)

EV Electric Vehicle. [2](#)

IaaS Infrastructure as a Service. [6](#)

IEVCC Intelligent Electric Vehicle Charging Controller. [2](#), [3](#)

IoT Internet of Things. [3](#), [4](#)

JMS Java Message Service. [7](#)

MQTT MQ Telemetry Transport. [2](#), [6](#)

PaaS Platform as a Service. [6](#)

SaaS Software as a Service. [6](#)

UVE Association of Electric Vehicles User. [2](#)

Introduction

Electric Vehicle (EV) are environmentally friendly since they do not emit any gas directly into the atmosphere, have fewer maintenance needs and operating expenses, and offer a quieter driving experience [1] [2]. These are the primary advantages of **EV**, which are becoming more and more attractive as the technology evolves. Even though they presently represent only 2.7% of global sales, according to the Bloomberg report by [3], the tendency is for them to grow. It is predicted that by 2025, **EV** will account for 10% of worldwide passenger vehicle sales, growing to 28% in 2030 and 58% in 2040, respectively. According to an analysis conducted by the **Association of Electric Vehicles User (UVE)** for Portugal, the sale of **EV** increased by 80% in November 2020 when compared to the same month in 2019 [4].

In many situations, the **EV** charging solutions for the home market implies an increase in the contracted power to allow for an efficient charging cycle that begins when the charger is connected and stops when the **EV** battery's maximum charge is reached. Increased contracted power is not necessarily the most effective approach for charging faster and more efficiently. A limited power grid connection shared among a large number of tenants makes it difficult to implement electric charging solutions able to solve challenges such as, controlling expenses by user, optimizing charging time, and even balancing the load based on the energy available at a given time [5] [6].

The authors in [7] present two distinct **Intelligent Electric Vehicle Charging Controller (IEVCC)** system configurations. This work focuses on the Mesh version intended for use in condominiums. In this scenario, tenants do not have access to parking spots with independent electrical connections, and the only solution available is to share the building's common grid. With the difficulties identified when designing and implementing multi-client solutions in mind, this work proposes a technical architecture capable of managing high data loads. The solution must be resilient and scalable to address the mesh installation problems and optimize grid usage. These aspects will benefit the end consumers and also assist the electricity distributors.

Multiple architectures were described using protocols like **MQTT** or **Zigbee** [8] [9]. The common gap is the detailed description of the software stack and how each layer interrelates. Some more detailed articles [10], in what relates to the software stack don't approach high data load scenarios where the scalability and flexibility of the solution is critical.

The planned solution is considered a streaming analytics system, typically consisting of three layers: ingestion, processing, and storage. The ingestion layer is the gateway to streaming. Data flow from inputs to processing and storage levels is decoupled, automated and managed. The processing layer receives the ingestion layer's data streams and transfers the output or intermediate results to storage. The storage layer keeps data in memory for iterative calculations or in databases for long-term storage. The analytics findings are given to a range of display and decision-assistance tools [11] [12].

This paper is organized as follows: after this introduction, the different data stages are presented in Section II. In Section III the full architecture diagram is presented and discussed, finally, the Conclusions are presented in Section IV.

Charging data stages

According to [13], on the [Internet of Things](#), there are five primary data processing architectures, fog-based processing, middleware-based processing, cloud-based processing, cloudlet computing, and mobile-edge computing.

The current data load comes from chargers, electrical counters, and message handling devices (broker). While systems operate, it is vital to log and store not only charging data, but also device status and usage metrics. To minimise damage in the event of network or device failures, the local setup (device layer) must be able to store and recover from failures. The final architecture proposal will take that into consideration.

The next natural step is to store data in the cloud, but once again, due to the sensitive nature of data and the significant load caused by each local instance, transmitting gathered data straight to cloud servers has proven to be challenging [14]. The same article discusses how fog computing may help minimize cloud reliance while improving performance. Nevertheless, the paper concludes that cloud and fog are complementary and can help deliver better and more complete services.

Fog Architecture

Cloud computing and fog computing infrastructures do not compete with each other, and they're complementary architectural solutions. [IoT](#) applications connect across fog nodes, and devices must be linked to at least one of these fog nodes (Fig. 1). Any device part of the [IEVCC](#) solution may connect to fog nodes which may be used in specific geographical cloud areas [15].

Because each fog node is a single point of failure, its spread and replication across regions should be considered for failure recovery and redundancy.

Multi-Charger Model

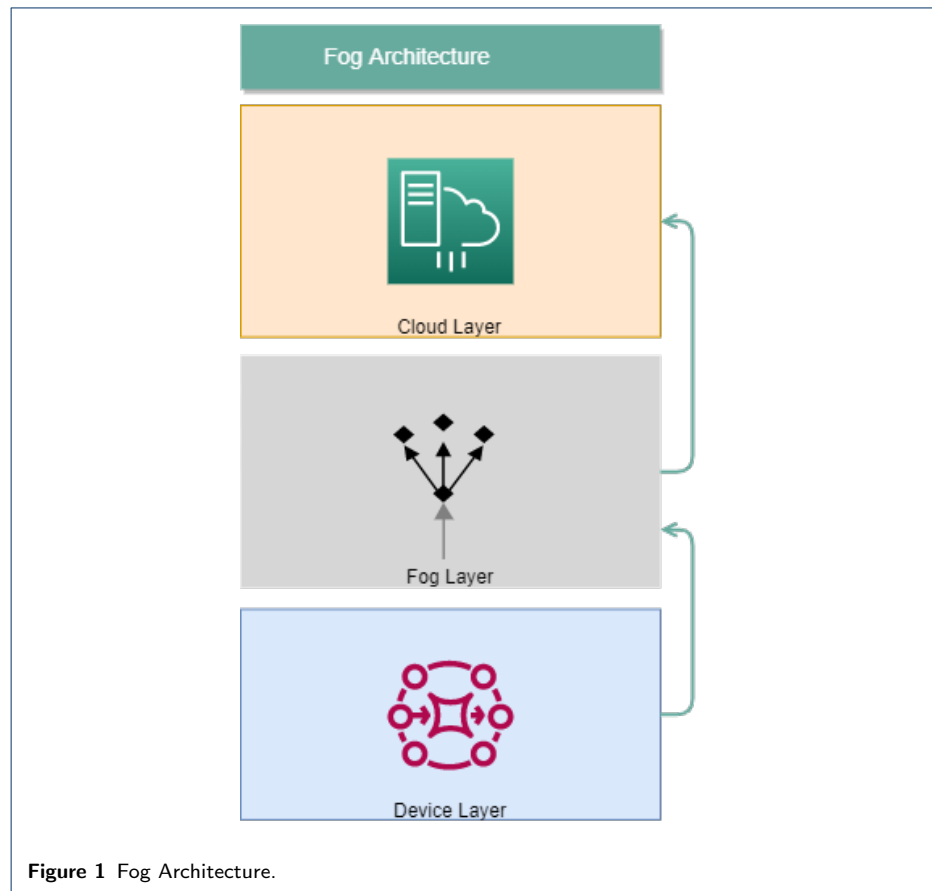
The multi-charger installation includes multiple chargers and may also include multiple electrical counters. Fig. 2 illustrates how each device connects to the "heart" of the device layer (Device Manager). This device is responsible for message handling and forwarding. At the same time, it manages authorized devices and clients during charges.

Each instance can be configured with custom load balancing rules, charger priority, and energy source selection when more than one source is available or when the provider shares the current source through an [Application Programming Interface \(API\)](#) endpoint.

All data is stored locally and forwarded to remote instances for data cleansing and transformation. Electrical communication usage, system logs, and client usage are then available for access by clients and providers.

Design of Data Management Service Platform

In order to allow data flow between several independent data producers and consumers, a data stream ingestion system must be scalable, resilient, and extendable. Chargers and electrical counters are the primary focus of the current configuration. However, the Device Manager design ensures that more [Internet of Things](#) devices will be able to connect and integrate into the solution in the future.



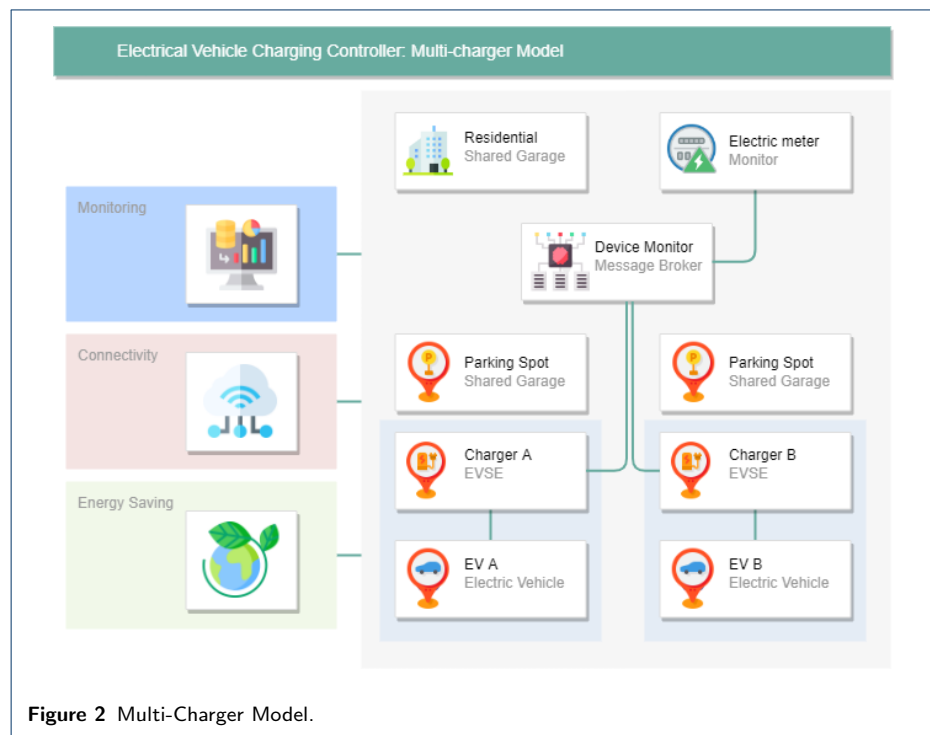
In order to demonstrate how to integrate the Fog Architecture into the Multi-Charger Model, the Fig. 3 depicts the position of each entity within the three Fog Architecture levels.

Each colored section maps the three distinct layers (Device, Fog and Cloud layers), where generated data is saved and then forwarded to the subsequent layer instances. It is crucial to clean and aggregate each record to be stored in the database during this process. It is also important to note that while generated data flow in one direction only, it is mandatory to authenticate users, devices, charging sessions, and others. This responsibility is taken care of by the Device Manager through the [API](#) instance in the cloud layer.

Device Layer

The device layer includes all the devices that support the local area network, like routers, switches, wireless access points or extenders, and all the smart devices connected locally. The smart ESP32-based devices are chargers and electricity meters for the current solution. Multiple other [IoT](#) devices may be integrated into the solution in the future.

The solution's heart is the Device Manager. Raspberry Pi version 3 boards were tested during development with no performance issues while handling device auth,



messages, local storage, and data forwarding to the fog layer.

For the Manager role, it is clear that power consumption and price will affect the board choice. Given that architecture compatibility is not an issue, the minimum requirements must meet the Raspberry Pi 3 specifications, as well as the ability to run Docker.

Fog Layer

The Fog Layer comprises devices in an intermediate layer between the cloud and the Device Layer. In this case, data is transferred to and processed by a computer or data center regionally located. Splitting this processing power across multiple regions decreases the total load each fog node will handle while increasing redundancy, a significant concern when dealing with critical data.

Fog node hardware must meet the minimum system requirements set for the distributed event streaming platform and the applications for cleansing and transforming data. Our prediction suggests each node has 8GB of RAM, 4 CPU cores, 1TB of storage, and 1GbE connection.

Cloud Layer

The cloud computing infrastructure builds on top of large-scale clusters that run various applications and pursue the core foundation that enables computing resources to be used to their full potential.

Cloud customers expect the entire system to be reliable, with redundant network and hardware. These cloud solutions allow companies to access data storage, resources, and on-demand services over the internet. Although cloud providers offer

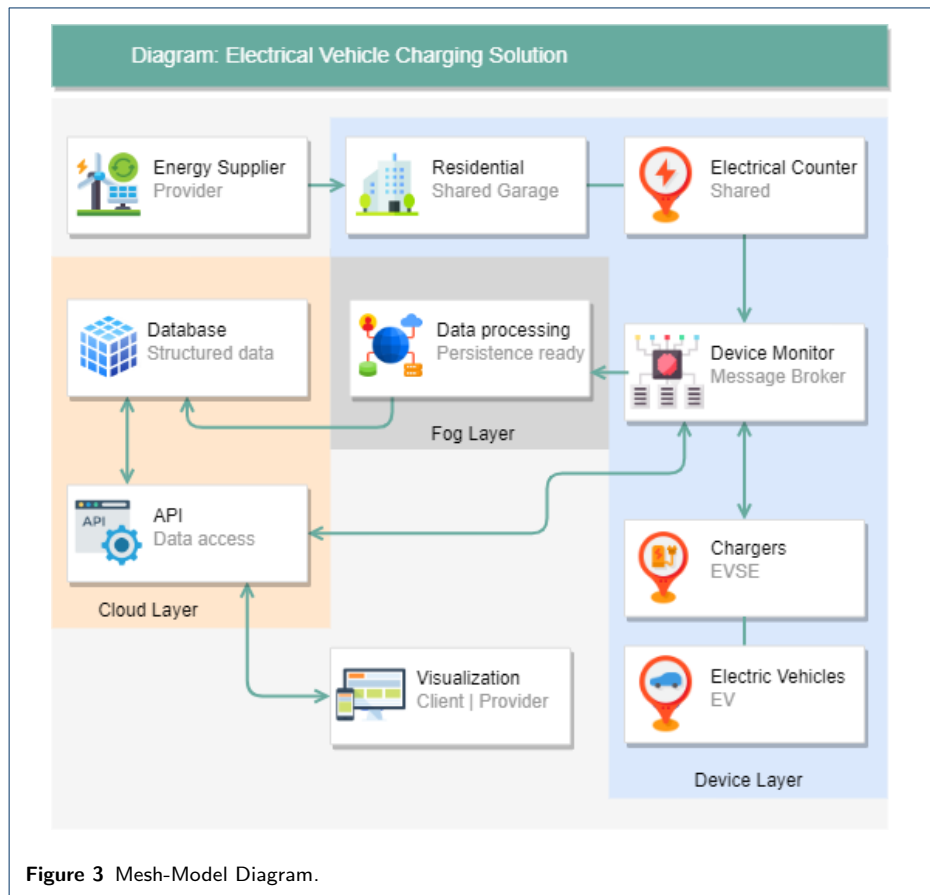


Figure 3 Mesh-Model Diagram.

a variety of solutions for several operations, based on the presented Mesh-Model (Fig. 3), the core business activities in the Cloud Layer include database services and API web applications.

The Cloud Layer is sub-layered into 3 layers: Infrastructure as a [Infrastructure as a Service \(IaaS\)](#), [Platform as a Service \(PaaS\)](#), and Software as a [Software as a Service \(SaaS\)](#). Choosing the best cloud layer depends on the budget, resources, the size of the operations, and multiple other factors.

Tech Stack

A tech stack is a company's choice of technologies to develop and manage an app or project. A tech stack often includes programming languages, frameworks, databases, front-end and back-end tools, and apps linked through API [16].

In a top-to-bottom analysis of Fig. 4, in the Device Layer, the current charger and electricity meter devices are programmed in C++, while the Device Manager is currently being developed in Python 3.8, with a tested compatible version range from Python 3.6 to Python 3.10. MQTT message broker (Mosquitto MQTT) and InfluxDB, an open source Time Series Database, both run on the same hardware. Each fog node in the Fog Layer will provide one or more Apache Kafka instances, an open-source distributed event streaming platform. Apache Kafka advertises necessary core capabilities like high throughput with low latency (2 ms), being prepared

to scale, and delivering high availability. It's also important to mention the built-in stream processing that enables the processing of event streams using joins, aggregations, filters, transformations, and exactly-once processing. It is also worth mentioning that the Kafka Connect interface is pre-integrated with hundreds of event sources and sinks, including Postgres, [Java Message Service \(JMS\)](#), Elasticsearch, and [Amazon Web Services \(AWS\)](#) S3.

As previously stated, in the Cloud Layer, our solution's core depends on a PostgreSQL database instance and the possibility to host web applications like client portals or [API](#) to access and store information.

The best cloud solution for PostgreSQL service is still open for further analysis, howsoever it is mandatory to have scalability possibilities, backups, and snapshots. Multi-region availability and synchronization will be decisive when dealing with thousands of clients.

Region and response time are essential factors in web application hosting, but so are high availability with load balancing, security, and scalability.

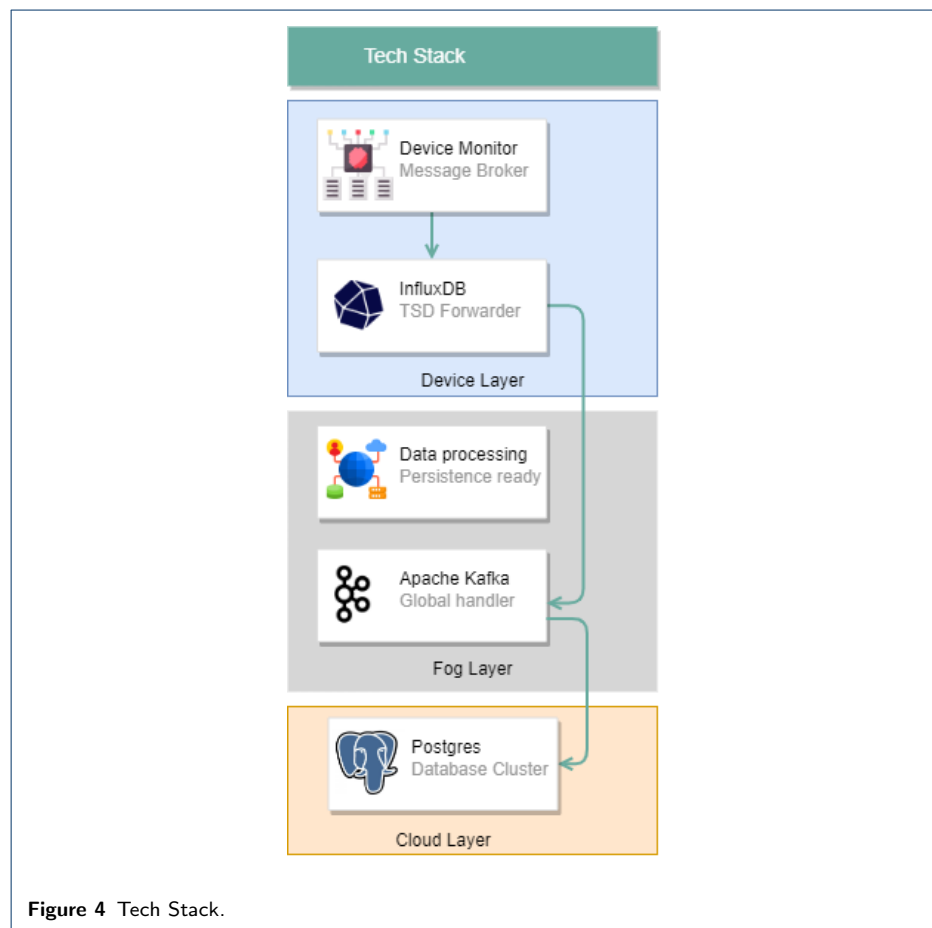


Figure 4 Tech Stack.

Discussion

During the investigation process, we analyzed multiple designs for software solutions with similar requirements. Despite the simplicity of adopting one component for the software stack instead of another, work must be done while trying to describe how each layer interrelates. The specification of infrastructure needs for a large data load is a difficult undertaking that involves numerous factors and criteria, such as data types, scalability and type of processing, communication between tasks or processes, and so on. Software engineering is crucial in ensuring that such workloads make the most use of the underlying hardware resources. As an example, data created by the current setup is handled by a cloud solution setup designed to leverage dynamic and adaptive cluster resource management, dimensioning, and configuration based on economic cost, quality of service, and availability requirements. Agile frameworks are typically used in this application design to lower barriers between development and operations teams, accelerate workflows (i.e. high deployment rates for faster feedback, better code quality leading to fewer errors and lower costs, and so on), and increase the reliability, stability, and resilience of the production environment.

Conclusions

Ingestion of data is critical for businesses and organizations that gather and analyze massive amounts of data. Continuous data streams are often ingested into big data processing and management systems from external sources. They are either processed incrementally or used to create a persistent dataset and related indexes. In order to keep up with vast amounts of rapidly changing data, stream processing systems must be able to ingest, analyze, and persist data continuously.

This work presents an architecture and the corresponding tech stack designed to handle massive time-critical data while performing cleansing and transforming operations, then storing it in a cloud database service. The innumerable options for each entity in the tech stack open new paths to different approaches and benchmarks. This will also help choose the best-tailored cloud provider for the solution's specific needs.

DECLARATIONS

Availability of data and materials

Not applicable

Competing interests

The authors declare that they have no competing interests

Funding

This work is partially funded by National Funds through the FCT - Foundation for Science and Technology, I.P., within the scope of the projects UIDB/00308/2020, UIDB/05583/2020. Furthermore, we would like to thank the Research Centre in Digital Services (CISeD), the Polytechnics of Viseu, Polytechnics of Coimbra and INESC Coimbra for their support.

Authors' contributions

Pedro Baptista has developed the broker/manager software components. Filipe Cardoso and José Rosado had developed the IEVCC software. Filipe Caldeira has tested the software and reviewed the document. All authors read and approved the final manuscript.

Acknowledgements

Not applicable.

Author details

¹Viseu Polytechnic - ESTGV, Viseu, Portugal. ²Coimbra Polytechnic - ISEC, Coimbra, Portugal. ³CISeD IPV, Viseu, Portugal. ⁴INESC Coimbra, Coimbra, Portugal.

References

- Dubey A, Santoso S. Electric Vehicle Charging on Residential Distribution Systems: Impacts and Mitigations. *IEEE Access*. 2015;3:1871–1893.
- Aghabali I, Bauman J, Kollmeyer PJ, Wang Y, Bilgin B, Emadi A. 800-V Electric Vehicle Powertrains: Review and Analysis of Benefits, Challenges, and Future Trends. *IEEE Transactions on Transportation Electrification*. 2021;7(3):927–948.
- Henbest S, Kimmel H, Callens J, Vasdev A, Brandily T, Berryman I, et al.. *New Energy Outlook 2021: BloombergNEF: Bloomberg Finance LP. BloombergNEF; 2020. Available from: <https://about.bnef.com/new-energy-outlook/>.*
- Nascimento M. Vendas de Veículos Elétricos em novembro de 2020 aumentam 80% em relação a novembro de 2019; 2020. Available from: <https://www.uve.pt/page/vendas-ve-11-2020/>.
- Shepelev A, Chung CY, Chu CC, Gadh R. Mesh network design for smart charging infrastructure and electric vehicle remote monitoring. In: 2013 International Conference on ICT Convergence (ICTC); 2013. p. 250–255.
- Ayan O, Turkay B. Domestic electrical load management in smart grids and classification of residential loads. In: 2018 5th International Conference on Electrical and Electronic Engineering (ICEEE); 2018. p. 279–283.
- Cardoso F, Rosado J, Silva M, Teixeira CJC, Agreira CIF, Caldeira F, et al. Intelligent Electric Vehicle Charging Controller. In: 2021 IEEE Vehicle Power and Propulsion Conference (VPPC); 2021. p. 1–5.
- Lee ZJ, Chang D, Jin C, Lee GS, Lee R, Lee T, et al. Large-Scale Adaptive Electric Vehicle Charging. In: 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm); 2018. p. 1–7.
- Qian K, Brehm R, Ebel T, Adam RC. Electric Vehicle Load Management: An Architecture for Heterogeneous Nodes. *IEEE Access*. 2022;10:59748–59758.
- Komasilovs V, Zacepins A, Kviesis A, Marinescu C, Serban I. Development of the Web Platform for Management of Smart Charging Stations for Electric Vehicles. In: VEHITS; 2018. p. 595–599.
- Isah H, Zulkernine F. A Scalable and Robust Framework for Data Stream Ingestion. In: 2018 IEEE International Conference on Big Data (Big Data); 2018. p. 2900–2905.
- Dias de Assunção M, da Silva Veith A, Buyya R. Distributed data stream processing and edge computing: A survey on resource elasticity and future directions. *Journal of Network and Computer Applications*. 2018;103:1–17. Available from: <https://www.sciencedirect.com/science/article/pii/S1084804517303971>.
- Aung TT, Thaw AM, Zhukova NA, Man T, Chernokulsky VV. Data processing model for mobile IoT systems. *Procedia Computer Science*. 2021;186:235–241. 14th International Symposium "Intelligent Systems. Available from: <https://www.sciencedirect.com/science/article/pii/S1877050921009583>.
- Garcia J, Simó E, Masip-Bruin X, Marín-Tordera E, Sánchez-López S. Do We Really Need Cloud? Estimating the Fog Computing Capacities in the City of Barcelona. In: 2018 IEEE/ACM International Conference on Utility and Cloud Computing Companion (UCC Companion); 2018. p. 290–295.
- Kanyilmaz A, Cetin A. Fog Based Architecture Design for IoT with Private Nodes: A Smart Home Application. In: 2019 7th International Istanbul Smart Grids and Cities Congress and Fair (ICSG); 2019. p. 194–198.
- Limón AT, Schulaka C. What's in Your Tech Stack? *Journal of Financial Planning*. 2020;33(2):22–27.