Department of Information Science and Technology (DCTI)

IoT System for EV Charging at Shared Spaces

Jose Pedro Marques da Cruz de Sousa Martins

A Dissertation presented in partial fulfilment of the Requirements for the Degree of Master

Computer Science Engineering

Supervisor:
PhD, Joao C Ferreira, Assistant Professor
ISCTE-IUL

September 2019

This page is intentionally left blank.

Acknowledgements

It is hard to write a fair acknowledgment section, not by the lack of humbly, but by the general sense that all our achievements are due to the contribution of everyone with we have the benefit to interact and learn from, all the behaviours, all successes in where we were involved or observed, and in particular, as we learn by failing, all the failures that we experienced (which marks us) and all the failures that we learn from others.

While being influenced by all my previous live events and interactions, so I would need to thank everyone for allowing me to have this fantastic experience which is "learning", this journey, like every other journey, is strongly influenced by a smaller group of persons, that inspired to it, contributed to it, were affected for it, and sometimes without being allowed to choose, were “forced” to contribute. Those are persons to whom I would like to express my gratitude (taking the risk of missing some), without any particular order of importance.

To my *small family*, Sonia (wife) and kids (and dog), that living behind the same roof, contributed with reviews, critics and with their immense comprehension by the family activities and events missed, from the simple things like arriving to the dinner table (always) late, be absent in important activities (and sometimes forget them) and most of the times, although physically present, completely absent, lost in my thoughts, to them I need to thank for their patience and support. To my large family for their compression and firm support, although less involved, always present ready and available to help.

To my grandparents, in particular to my grandparent Rogério, that have been (and is, at the time of this writing) always present in all my life events, strongly influenced my personality and values, helped me follow my choices and objectives, supporting them in all possible ways, sometimes disagreeing (and advising against it) other times being negatively impacted by them, but always supporting them.

To all my previous employers and colleagues that, while working, allowed me to learn, to expand my knowledge, to tail my behaviour and allowed me to fail and helped to learn with it. A special thanks to everyone that, when joining the companies where I was working for, I (tried to) mentor and taught me, (boldly) challenging my beliefs.

To the ISCTE-IUL community from my colleagues with whom I paired in groups for working in course projects, to share and discuss ideas. To my professors (which I will not disclose the names as I may need to enumerate almost all), for the knowledge and experiences

shared, and mainly, for the inspiring behaviour and, in some cases, for the passion, enthusiasm, and energy transmitted while discussing a specific topic (genuinely inspiring).

Finally, I must express the most definite sense of gratitude, to my thesis coordinator, Professor João Carlos Ferreira, the achievements of this work are a direct consequence of his intense involvement, support and hardly would be reachable without his help. Apart from the coordination and guidance of the work, from the idea (conception) to its realization in the strict sense, Professor João has provided one incredible help, being present and available during the entire process, helping me to keep the focus, motivation, leading me to continue working towards the defined goals even when sometimes the demand on my professional activity started to push to the other direction. His inspiring and mentoring behaviour challenged me to progress further on my academic studies, and I am the most thankful for all the support closing this stage, for the inspiration to embrace a new challenge and help to continue further.

Thanks.

Resumo

No presente trabalho, é aplicado um paradigma de *Internet Of Things* (IOT) para agilizar e controlar o processo de carregamento de Veículos Elétricos (VE) em espaços partilhados de menores dimensões, como por exemplo condomínios residenciais, sem que seja necessária a intervenção (a título de prestação de serviços) de uma entidade externa, sendo todo o processo controlado pela gestão de condomínio.

Uma aplicação móvel permite ao utilizador interagir com o sistema, permitindo a este autenticar-se no mesmo é condição necessária para que seja despoletado o processo de carregamento do VE. O sistema implementado com recurso a um microcontrolador encontra-se ligado a um conjunto de sensores e um atuador permitindo medir a energia que esta ser consumida para carregamento do VE e simultaneamente, ligar e desligar o dispositivo de carregamento do veículo (através do controlo de um interruptor que entrega a energia entregue a este). O processo é controlado por uma unidade de gestão centralizada, que gera a distribuição de energia pelas estações de carregamento de VEs de acordo com as limitações do condomínio através do ligar e desligar destas e em simultâneo regista e processas as medições da energia consumida para consolidar as informações que constituem a transação de carregamento de VE e respetiva contraparte financeira associada à mesma.

Adicionalmente, a unidade de gestão centralizada e a aplicação móvel, disponibilizam interfaces de utilizador mínimas para permitir funções como a consulta de transações, gestão e configuração da plataforma.

Complementarmente, é apresentado um modelo conceptual permitindo escalar a solução proposta para espaços partilhados de maior dimensão, com recurso à utilização de tecnologias *blockchain* para gestão e registo das transações financeiras associadas à operação. Propondo uma abordagem, que poderá ser replicável em cenários mais amplos de utilização como por exemplo, a infraestrutura publica de carregamento de VE de uma cidade.

O protótipo desenvolvido foi testado num espaço partilhado com três VE, usando uma infraestrutura de carregamento durante 3,5 meses.

Palavras-Chave: Veículos Elétricos, Carregamento de Veículos Elétricos, *Blockchain*, *IoT*, Aplicações Moveis.

Abstract

In current work, we apply the Internet of Things (IoT) paradigm to handle the electric vehicle (EV) charging process in small shared spaces, such as condominiums without requiring the intervention of an *external* supervision entity, being that *role* performed by the condominium management.

A Mobile App handles the user interaction with the system, authenticating the request to initiate the EV charging process, a microcontroller connected to set of sensors and an actuator is used for measuring energy consumption and for enabling the charging process and, a Management Unit controls the process end to end, providing the required services to the Mobile App and the microcontroller unit while manages the energy sharing between the EV charging stations accordingly the condominium limitations and processes the energy measures to consolidate the EV charging *energy transaction*. A minimal user interface allows the users to visualise transactions, manage users' preferences, and configure the platform.

Additionally, the conceptual model for a scaled solution is presented, supported on blockchain technologies to handle the financial transitions, allowing current approach to be replicated on broader EV charging scenarios, such as public charging systems in a city.

The developed system was tested in a shared space with three EVs using a charging infrastructure for 3.5 months.

Keywords: Electric Vehicle; EV Charging Process; Blockchain; IoT; Mobile App.

Table of Contents

Tables

Figures

Acronyms and Abbreviations

ADC – Analogic to Digital Converter

BC – BlockChain

CS – Charging Station

DBMS – database management system

DSRM – Design Science Research Methodology

EEPROM – Electrically Erasable Programmable Read-Only Memory

EV – Electric Vehicle

HTTP – HyperText Transfer Protocol

HTTPS – HyperText Transfer Protocol Secure

IoT – Internet of Things

IoT ARM – IoT Architectural Reference Model

IoT DM – IoT Domain Model

IP – Internet Protocol

LAN – Local Area Network

MVC – Model View Controller

PKI – Public Key Infrastructure

REST – Representational State Transfer

SPA – Single Page Application

SQL – Structured Query Language

SOC – (EV's) State of Charge

TCP – Transfer Control Protocol

UML – Unified Modeling Language

URI – Uniform Request Locator

WAN – Wide Area Network

Chapter 1 – Introduction

1.1.Context

One of the significant challenges related with electric vehicle (EV) market penetration is the charging process, where the main problems are associated with the lack of proper infrastructure in the existing residential buildings (condominiums) since they were not prepared to fulfil this new requirement.

Condominiums often have a problem with shared electricity, which fails to meet the EV owner's requirements, another facet of this challenge is the associated with rental houses and the need for supporting EV charging during limited periods only, which due to elevated number for rental houses ('Stat of the Week', 2018) gives to this issue a new dimension.

In condominiums, unfortunately, there is a general reluctance regarding the installation of EV charging stations that will only be used by a few homeowners ('Stat of the Week', 2018). In addition, there is also an issue regarding the safety and limitations of the electrical installations, as they were not built upfront to support EV charging stations and, adapting the condominium electrical infrastructure will require that a consensus between the majority of the owners is reached, often hard to achieve, and eventually authorizations issued by the government building safety entities are also required.

Taking into consideration that most residential buildings have shared spaces with standard electrical installations, not prepared for the installation of new EV charging systems, this is considered a barrier to EV uptake (Axsen et al., 2015). A study by Lopez-Behar et al. ('Putting electric vehicles on the map: A policy agenda for residential charging infrastructure in Canada', 2019) identified four main problem domains in the context of sharing EV charging solutions in buildings:

- Unavailable charging infrastructure;
- Building limitations;
- Regulation issues;
- Parking availability.

1.2. Motivation

Taking into consideration the issues exposed in the previous section, it urges to research, and evaluate the design of solutions that will allow to address the problems identified, aiming to contribute to the increase of adoption of the EV in the existing communities, as the replacement of the combustion vehicles by electric vehicles will increase the use of renewable energy sources reducing the environmental footprint.

The new advances in the Internet of Things (IoT) technologies (Al-Fuqaha, Guizani, Mohammadi, Aledhari, & Ayyash, 2015), associated sensing and acting devices, communication platforms and, information systems have the potential to create new solutions for the identified problems.

The presented problem and the (eventual) availability of technical means to solve it, lead us to the goal of this work, which is, to explore those new technologies and evaluate how can they be used to address and improve the stated problem.

1.3.Research Question and Work Objectives

Based on the exposed, the research performed on this work aims to devise an approach supported on the new IoT developments to solve the identified problems related to the EV charging in shared spaces.

It is the primary goal of this work to design and implement an IoT-based system for handling the EV charging process in a condominium, which can be used in the context of a shared energy infrastructure without requiring a formal supervision entity to control the process, to manage the EV charging processes accordingly the shared space power limitations, controlling the number of simultaneous EV charging.

To achieve this goal, an IoT System was developed, that connected to each power socket or charging device (i.e., between the power grid and the EV charging device), delivers energy only to an authenticated users, guaranteeing their identity and the non-repudiation of the associated energy transaction, while measuring the energy consumption to generate accounting or billing information, allowing each user to pay for their energy consumption.

1.4.Methodological Approach

The Design Science Research Methodology (DSRM), was followed as the methodological approach for developing this work.

Design Science (DS), is defined as the study of artificial, targeting the creation of new and innovative artefacts (Simon, 2008). The Design Science Research Methodology (DSRM) aims to systematise the application of the Design Science Research through the definition of an iterative Process Model composed by the sequence of steps or activities presented below (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2007):

- Problem Identification and Motivation;
- Define Objectives for a Solution;
- Design and Development;
- Demonstration;
- Evaluation;
- Communication.

Aiming to document the current work using a standard graphical notation, the Unified Modeling Language (UML) is used due to its high acceptance under the development community (Fowler, 2004).

1.5.Structure and Organization

This document, structured in seven chapters, is organised as follows:

- Chapter 1 (the current chapter), introduces the current work, focuses context, motivation, and sets the objectives to be achieved.
- Chapter 2 presents the state of the art in related works.
- Chapter 3 describes the approach followed and introduced a conceptual model and establishes its mapping with the IoT Reference Model and Architecture (Bauer, Boussard, et al., 2013; Bauer, Bui, et al., 2013).
- Chapter 4 describes the System implementation flow.
- Chapter 5 presents a case study at a condominium.
- Chapter 6 elaborates a conceptual model for extending the current work to Charging Stations using IoT and Blockchain.
- Chapter 7 closes the document, presenting the conclusions, and discusses future implications of the presented work.

1.6.Scientific Contribution

The work developed supported the writing of the following article for a Special Issue of the Journal Energies (ISSN 1996-1073), published in this Q1 journal in 2 of august 2019:

Martins, J. P., Ferreira, J. C., Monteiro, V., Afonso, J. A., & Afonso, J. L. (2019). IoT and Blockchain Paradigms for EV Charging System. *Energies*, *12*(15), 2987. DOI:10.3390/en12152987

Chapter 2 – State of Art

The proposed approach explores a set of works in several domain areas to create a new approach to handle the EV charging process in shared spaces, including the use of IoT sensing devices to measure electricity consumed on the EV charging process and coordinate the power distribution between the EV charging nodes of the shared space, accordingly the shared space power limitation.

As EV charging payment process is more frequent than fossil fuel refuelling and more complicated due to the immaturity of the service, issues related to the following points are relatively common:

- Transparency and clarity of rates and costs before they are incurred;
- Ability to pick-and-choose best rates and location of available charging points *on the go*;
- Ability to request priority charging and pay for it, when other EVs do not need priority;
- Ability to select a supplier or source of electricity, which would also enable greater competition and increase the trust of customers;
- Preferences for various types of payment, such as post-paid, pre-paid, or one-off payment.

This work complements previous works on an EV charging systems (Joao C. Ferreira, Monteiro, Afonso, & Silva, 2011; Joao Carlos Ferreira, da Silva, & Afonso, 2011) and IoT energy measurements using local sensors on (J. Ferreira, Afonso, Monteiro, & Afonso, 2018) while focusing new challenges for the energy markets discussed in (J. Ferreira & Martins, 2018). Together with mobile device authentication, local sensors, and a management unit, we developed a new approach applicable to shared EV charging spaces.

Another exciting output is to use of mobile devices to provide authentication and payment services in the context of the public EV charging systems, exploring recent advances in mobile device payment systems for public transportation (Baia, Ferreira, Filipe, & Cunha, 2013) and other application areas (Dahlberg, Guo, & Ondrus, 2015).

2.1.Power Limitations

Concerning driver profiles and EV charging with power limitations, several studies have been performed, and we apply a simple approach inspired on a previous work described in (J.

C. Ferreira, Monteiro, & Afonso, 2014). Another issue originated by the increase of the EV charging needs is the impact on the energy demands and the power limitation of the existing infrastructure (C. Liu, Chai, Zhang, Lau, & Chen, 2018), which may not only increase the operational costs to fulfil the required demand but also affects the voltage stability of the network. In (C. Liu et al., 2018), the authors introduced the AdBEV, which is an algorithm to optimise the EV charging schedule, maximising the voltage stability at the power grid side, and minimising the charging costs.

2.2.Authentication

In our implementation, it was also considered an implicit authentication mechanism (Bo, Zhang, Li, Huang, & Wang, 2013), applied on user's mobile devices requests to the Management Unit, which confirms the user authentication based on actions that he had performed previously in a daily basis. This implicit authentication mechanism can be used to prevent fraudulent transactions on a mobile device, verifying that the user is who claims to be during the transaction. After researching systems that meet our criteria, we found some promising works (De Luca, Hang, Brudy, Lindner, & Hussmann, 2012; Feng et al., 2012; Jakobsson, Shi, Golle, & Chow, 2009; Patel, Chellappa, Chandra, & Barbello, 2016; Sae-Bae, Ahmed, Isbister, & Memon, 2012), and we also found a solution targeting the user privacy (no identification is performed) in an approach based on the system proposed called Touchalytics (Frank, Biedert, Ma, Martinovic, & Song, 2013).

2.3.Blockchain

Complementing the work performed, we present a conceptual model to apply a blockchain-based approach to handle distributed transactions without central supervision. The application of blockchain in the domain of smart grids has excellent potential, providing a decentralised approach to implement management systems (Pop et al., 2019) and handle energy transactions. The primary goal of the blockchain is to allow decentralised transactions using a digital currency, such as Bitcoin (Nakamoto, 2008) or Etherum ('A Next-Generation Smart Contract and Decentralized Application Platform', 2014/2019), without the need of a public authority to control the process. From the technical perspective, a blockchain is as a record database composed by a sequence of linked blocks (similar to table rows on the relational model), where the transactional information, each block, is linked to the previous block and cryptographically signed (Panarello, Tapas, Merlino, Longo, & Puliafito, 2018).

From the functional perspective, the User *A* performs a transaction, links the transaction to the previous block and cryptographically signs the block with his private key, and, the new block is sent to all the nodes of the network, being the synchronization and authenticity of each block guaranteed by a consensus protocol between the participating nodes. To verify the integrity of a received block, the User *B* checks the cryptographic transaction signature using the public key of user *A*. This process has the following properties:

- Decentralisation, the new block is sent to the network to validation and certification, without the intervention of a central authority.
- Anonymity, since it allows for the authentication of transactions without giving up any personal information (*B* only needs to know *A* public key);
- Auditability, which is guaranteed based on the fact that each of the transactions is recorded and validated with a timestamp, where users can trace the previous transactions by accessing any node in the distributed network.

The meter sampling information has the potential to use a considerable amount of data, in particular as all the blocks are sent to all the participants on the network. Aiming to minimise the amount of information stored on the chain, (Pop et al., 2019) presents a design to balance the amount of information kept on-chain/off-chain while keeping the properties of blockchain implementation. The authors of (Erdin et al., 2018) suggest that the use of an open public cryptocurrency network, such as Bitcoin or Ethereum, can introduce a high transactional costs, due to the fees associated with cryptocurrency transaction processing (eventually similar to the cost of the energy supplied), and propose the development of a private Bitcoin-based blockchain network for EV charging purposes. Other relevant application cases include micro-generation (J. Ferreira & Martins, 2018; Sanseverino, Silvestre, Gallo, Zizzo, & Ippolito, 2017), as well as the contribution to handle the EV charging payment process without the use of propriety company payment systems.

Some issues identified are also addressed in (Pustisek, Kos, & Sedlar, 2016), which proposes a blockchain-based model with recourse to a bid to identify charging stations (and eventually schedule the charging), complementary to the approach suggested in (Joao C. Ferreira et al., 2011). In (Thakur & Breslin, 2018) the application of a blockchain-based process is suggested to support the EV charging queue management.

As a new topic of research, new publications are appearing in the literature concerning the use of a blockchain approach to handling the EV charging process, such as:

- Testing pilots to use digital currency for the EV charging process (Higgins, 2016; 'RWE and Slock.it – Electric cars using Ethereum wallets can recharge by induction at traffic lights', 2016);
- A proposal of a P2P energy transaction model to handle the EV vehicle-to-grid (V2G) operation in smart grids (Kang et al., 2017);
- Handling the EV authentication issues based on a blockchain approach (Garg, Kaur, Kaddoum, Gagnon, & Rodrigues, 2019);
- Proposal of a cross-domain authentication scheme with blockchain (D. Liu et al., 2018);
- Handling of security and privacy issues for energy transactions based on blockchain.

Moreover, in this context, the EV is identified as part of the energy market (Aitzhan & Svetinovic, 2018), a contribution to the contextualization of the local energy market is presented on (Mengelkamp, Notheisen, Beer, Dauer, & Weinhardt, 2018), where the blockchain plays an essential role in the decentralization process, as well as for optimization purposes (Munsing, Mather, & Moura, 2017).

Chapter 3 – Proposed Approach & Conceptual Model

The current section aims to identify the requirements for the proposed system, derived from the objectives presented in the previous section and presents the conceptual model accordingly the IoT System Architecture Reference Model introduced in (Bauer, Boussard, et al., 2013; Bauer, Bui, et al., 2013). Figure 1 presents the components of the architecture model used to describe the conceptual model of the system addressed in section 3.2.

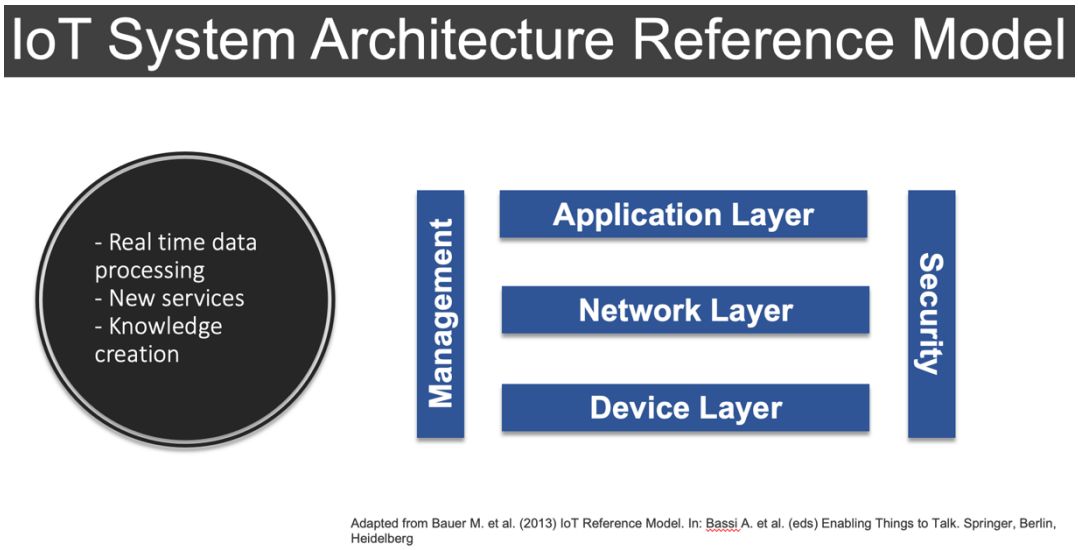


Figure 1.IoT System Architecture Reference Model (adapted).[1]

3.1.Problem Identification – Identification of Requirements

The following requirements were identified based on the initial problem:

3.1.1.Real-Time Energy Metering

A metering device must be easily attachable to the existing devices EV charging devices (between the EV charging device and the power socket) and able to measure and register the energy delivered to the charging EV, the during the charging process. As the energy delivery is a continuous measure, the instantaneous current must be periodically sampled (the discretisation process will introduce a measurement error in the inversely proportional to the sampling frequency).

[1] Adapted from (Bauer, Boussard, et al., 2013)

3.1.2.Manage the energy power limitations

Due to the power limitations of a condominium and aiming to avoid changes in their power infrastructure to increase its capability, allowing to charge several EVs simultaneously, with incurred costs on the adaptation of the existing infrastructure and an eventual increase in the cost of the energy as often the energy pricing is calculated by the energy consumed and also by the available power, the system must be able to manage the number of simultaneous charging EVs.

3.1.3.Secure Identify / Authenticate the charging EV

As the energy transaction will have a financial impact as the EV owner is responsible by supporting that cost, the system must guarantee the identification of the user requesting to charge the EV, to avoid the repudiation of the transaction.

3.1.4.Account and Report Energy Transactions

The system must be able to gather all the measurements, to allow a fair distribution of the energy costs, into a consolidated energy transaction and, associate each energy transaction to a specific EV owner, providing usage reports to allow the correct accounting of the energy spent.

3.1.5.Reduced Implementation Costs

In particular, inside a condominium, the fair division of these costs by the benefiting members can be complex to achieve and, frequently, a consensus must reach to allow the approval of any changes that will incur in costs to the condominium. Devising a solution with reduced or absent of changes in the energy infrastructure is a crucial factor to the acceptance. The following design options can contribute to this objective:

- Self-Contained

  The solution should be complete, self-contained, and able to explore the existing infrastructure, to ease the deployment inside a condominium.

- Absence of energy (or other) infrastructure changes

Changes in the energy distribution infrastructure of a building may require an entire project to be put in place, with the required legal approvals, project plans, inspection, and validation by certifying bodies, which can have a high cost. The devised approach should require minimal or none changes to the existing infrastructures.

Figure 2 and Figure 3 presents the Use Case diagrams (Fowler, 2004) for the actors participating in the system.

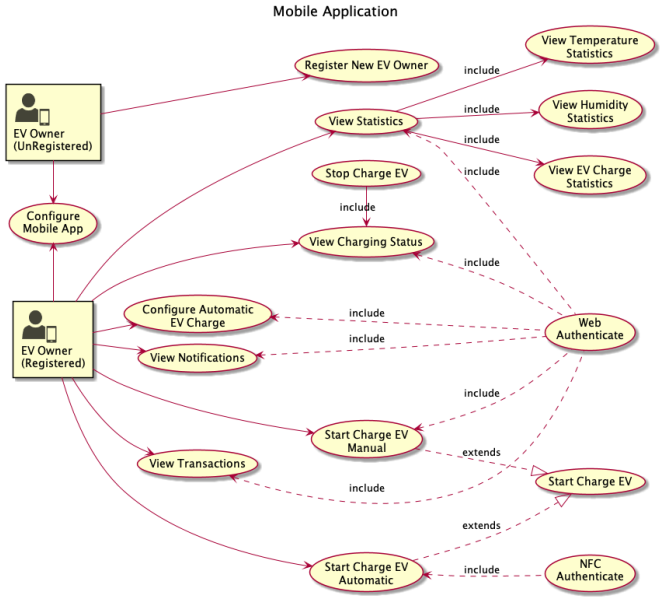


Figure 2.Use case diagram of the Mobile App.

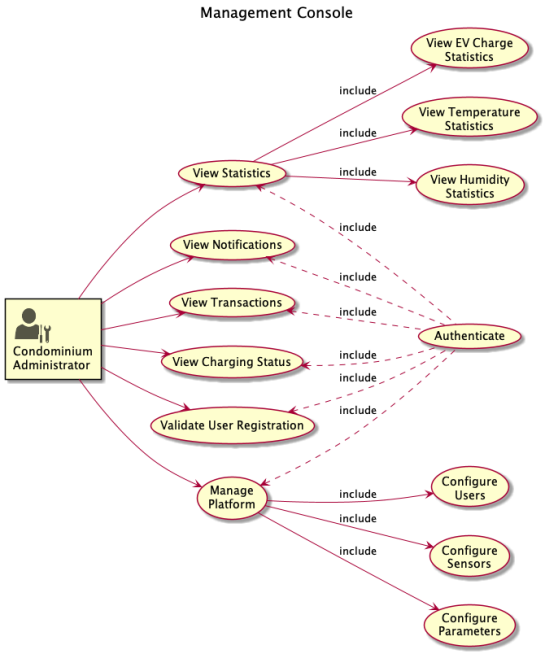


Figure 3.Use case diagram of the Management Application.

3.2.Conceptual Model

The IoT ARM (Bauer, Boussard, et al., 2013) proposes a model to conceptually present the architecture of an IoT independent of the used technology, whereas the IoT Reference Model (Bauer, Bui, et al., 2013) sets the stage for the IoT ARM, defining a set of concepts and relations on the IoT Domain Model. Figure 4 exposes a UML representation of the IoT Domain Model.

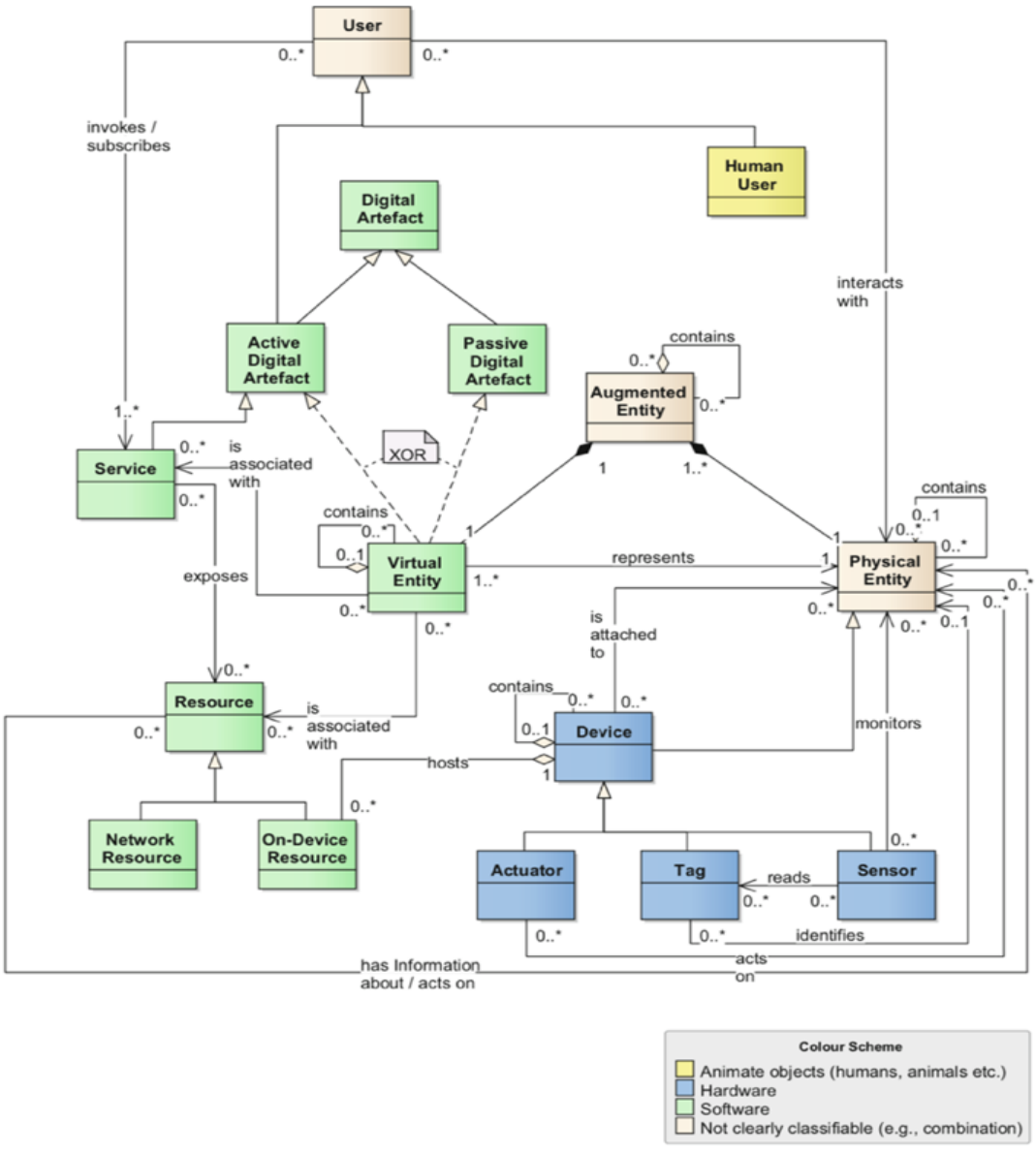


Figure 4.IoT Domain Model[2].

[2] UML representation from IoT Domain Model from (Bauer, Bui, et al., 2013)

Figure 5 presents the instantiation of the EV Charging platform using the IoT DM.

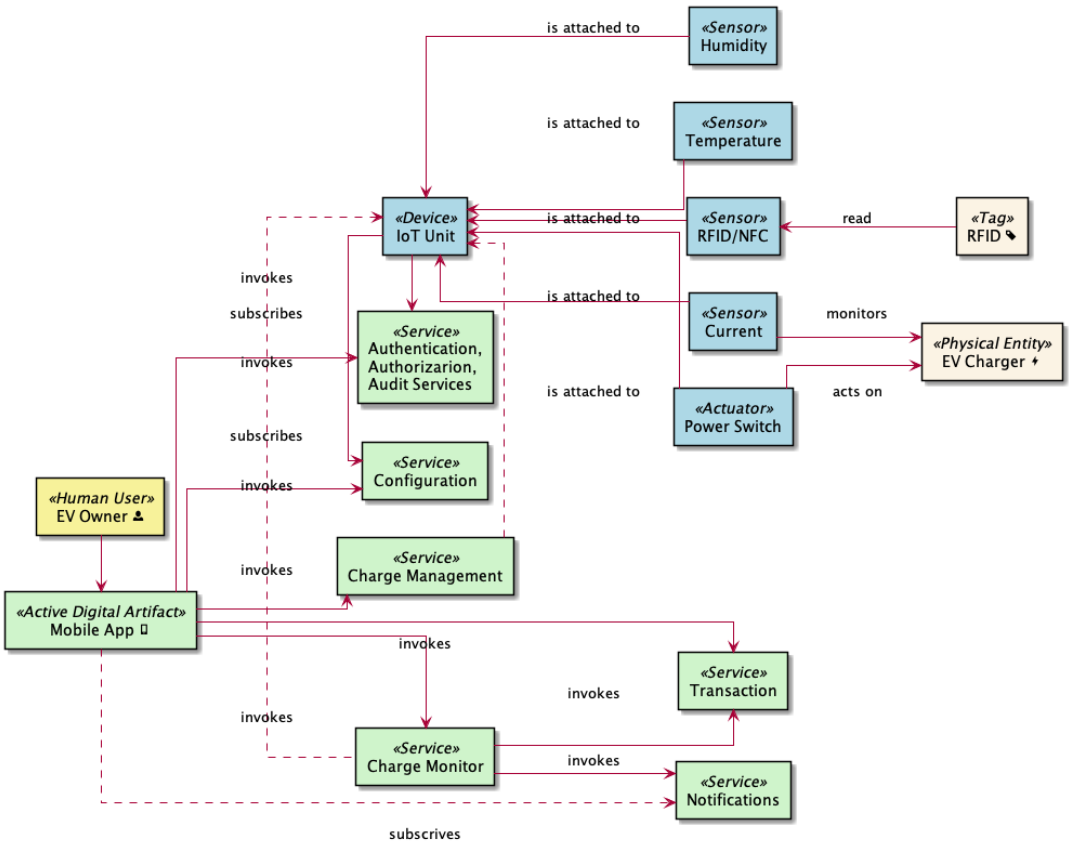


Figure 5.IoT Domain Model instantiation for the proposed solution.

Following the approach suggested by Figure 1, the developed EV Charging platform can be decomposed and organised into several functional perspectives. The IoT ARM suggests the existence of an intermediate layer between the Network/Communication and the Application Layers to include all the supporting activities required to enable the information flows between those layers, however, for the current exposition, and due to the limited scope of the developed system that intermediate layer was not taken into consideration. This layered functional segregation, identifies each existing sub-system abstraction, allowing to consider the entire IoT system as a composition of several building blocks, which can be replaceable independently, providing high flexibility to the developed system.

Follows the mapping the components of the developed IoT system, some identified on the domain model, in Figure 5, into the layers proposed in Figure 1.

- Device Layer

Is considered that all the IoT Unit components are on the Device Layer, respectively the IoT Unit and the sensors and actuators attached to it, the current, temperature, humidity and, RFID sensors as well as the power switch that acts on the EV Charger, switching it on/off;

- Network Layer

  The model suggests that all components required to exchange information between the Device Layer and Application Layer should be organised and identified in this layer. On the developed system, the network components attached to the IoT Unit, to Management Unit and, even the Mobile Device networking capabilities, should be considered, even if physically connected to a specific element:

- Application Layer

  On the Application Layer, we can consider the services or components Charge Management, Charge Monitor, Transaction, Notification Services, as well as the Mobile App, as all these components are accountable for providing the IoT system functionality and core for performing that activity;

- Management Layer

  Are considered elements of this transversal layer, the components that contribute to the management of the platform, from the instantiation of the IoT Charging EV Platform on the IoT Domain model, on Figure 5, we can identify the Configuration service and eventually some components of the Authentication, Authorization and Auditing services. In the IoT ARM model, we can find at this layer, the elements related to Configuration, Fault, Reporting, Membership, and State functional activities;

- Security Layer

  This layer integrates all the security-related activities, from our instantiation, Authentication and, Authorization services, as well as some components (not identified in the domain model) like the HTTPS engine, belong to this transversal layer, from the model it can also be identified, Key Exchange and Management, Trust & Reputation, and Identity Management.

To summarise, the EV charging platform is composed of the elements presented in Figure 6, whose roles are briefly described below, whereas the details for each component are addressed in the next section:

- **IoT Unit.** Sensor and Power management units that support the interaction with the EV charger, used to enable or disable it (on/off switch), to measure the amount of power consumed, gather environment temperature and humidity (complementary measures), and to upload all the information to the Management Unit;
- **Mobile App**. The element that establishes the interaction between the EV owner and the platform authenticates the user, starts/stops the charging process, and provides some everyday operations, such as configuration management, usage dashboards, transactions lists;
- **Management Unit.** Controller heart of the platform, provides all the backend services to support the required operations, orchestrating the IoT units to deliver energy, accordingly the limitations. It also implements the management console for the platform. In the developed prototype, aiming to keep the solution self-contained, it also acts a physical network element, configured as a Wi-Fi access point, providing network access to the IoT units and the Mobile App, but it could also be implemented using a cloud computing platform.

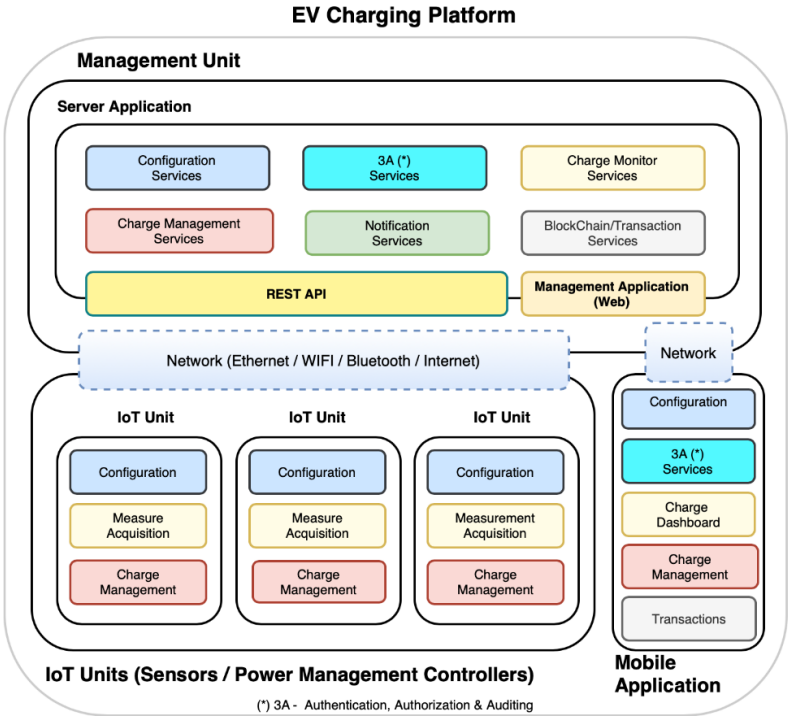


Figure 6. Component architecture of the proposed EV charging platform: Server Application, IoT Units, and Mobile App.

Chapter 4 – System Implementation

The system development followed a *standard* SDLC flow, Figure 7 displays the phases considered, described below, except for the Problem Identification, which previously addressed in section 3.1.

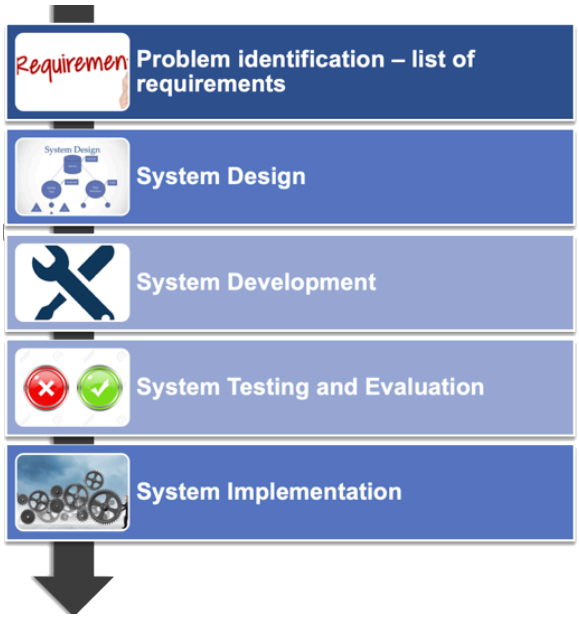


Figure 7.Methodology for creating (IoT) system.

4.1.System Design

Following the model presented in the previous chapter, the proposed EV charging, platform is composed of three significant elements, detailed in the current section:

- IoT Unit;
- Management Unit;
- Mobile App.

Figure 8 shows an overview of a condominium with the proposed EV charging platform, the significant characteristics for the proposed system are:

- Mobile App, to allow the EV owners to interact with the platform, authenticate on the platform, manage user's preferences, monitor charging process and energy transactions;

- IoT Unit, based on Arduino Microprocessor, to control the energy release and collect the delivered energy measures and transmit the required information to the Management Unit;

- Management Unit, based on Raspberry PI configured as a WiFi network access point, and while acting as controller of all the system, store system data, handling energy transactions and financial counterparty and managing the charging according to the power limitations available on the infrastructure.

The following features can be highlighted:

- A pre-registration with a local EV charging providers is not required, avoiding the problem of different cards in different charging infrastructures (every charging infrastructure has its cards, and this is a problem for EV owners because they need several charging cards when different providers are available);

- Reduced costs (almost zero fees), because there is no requirement for a third-party central management entity, apart from the condominium to management, which would create additional costs;

- No infrastructural changes as the system manage the energy distribution accordingly the power limitations of the installation.

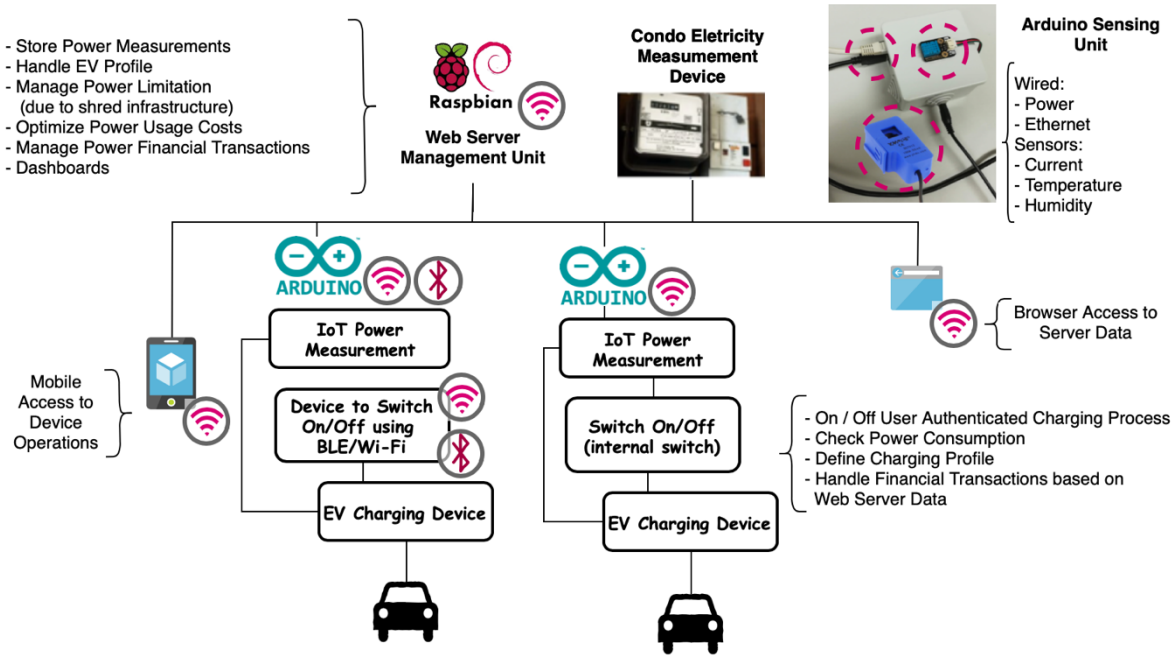


Figure 8.Overview of the proposed EV charging platform in shared spaces.

4.1.1.IoT Unit

Hardware

The IoT unit was developed following the initial approach described in work presented in (J. Ferreira & Martins, 2018). The re-assessment of the surrounding environment, context, overall requirements, conducted to the refinement of the initial architecture into a flexible system architecture able to support different network transmission requirements/devices, current sensor devices, and power switching devices, tailoring their combination to match a specific installation requirement, targeting a commercial level system prototype.

After an initial period of checking and testing hardware, the solution implemented was based on an Arduino Uno (microcontroller) combined with the devices listed in Table 1, where only one component for each type was used to assemble the IoT unit.

Table 1.List of IoT hardware add-ons.

Component Type	Device
Network (Shields)	Sparkfun ESP8266 (Wi-Fi) WIZnet's W5100 (Ethernet)
Current Sensors	SCT-013-000 (non-intrusive) ACS712 20A (intrusive)
Power Switching (*)	SRD-05VDC-SL-C (generic network switch)
Temperature and Humidity	DHT11
NFC RFID (**) Wireless Module	PN532

(*) The Management Unit can control a generic network-controlled switch. Approaches such as BLE-controlled switches can eventually also be used, providing that the IoT unit included a BLE Add-On
(**) Near Field Communication and Radio-Frequency Identification

The most relevant characteristics of the hardware components used for the prototype implementation are:

- Arduino R3 Uno Microcontroller (Figure 9a), based on the microcontroller ATmega328P, it has the following characteristics (from the Arduino R3 Uno dataset);
    - 14 Digital Input/Output pins
        - Two for serial RX/TX (Receive and Transmit)
        - Six with pulse-width modulation (PWM) output capability to mimic an analogue output
        - Six analogue input pins (A0–A6).

  - 16 MHZ clock speed
  - Memory: Flash, 32 K; SRAM, 2 K; EEPROM, 1 K
  - USB type B connection, ICSP Header
  - Power input 9 V (operating voltage 5 V), built-in LED, reset button

- Sparkfun Wi-Fi Arduino Shield, based on ESP8266 (Figure 9b), manufactured by Sparkfun, used to connect the Arduino microcontroller to a Wi-Fi network and using the "standard" internet protocols (TCP or UDP);

- Arduino Ethernet Shield, based on Wiznet W5100 (Figure 9c), used to connect the Arduino microcontroller to an Ethernet network using the "standard" internet protocols. Compliant with the IEEE 802.3 10Base-T and 802.3u 1000Base-TX standards, includes a TCP/IP hardwired stack, supporting up to four simultaneous connections and a transfer rate up to 100Mbps;

- Non-Intrusive Current Sensor SCT-013-000 (Figure 9d), a non-intrusive sensor used to measure the current passing through a conductor without the need to cut or modify the conductor itself. The measurements are collected from the electromagnetic induction, which is proportional to the intensity of the current passing through the conductor. This sensor collects measurements up to 100 A, outputting at 50 mA, with an accuracy of 1% to 2% of the actual value;

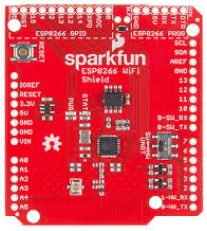
- Intrusive Current Sensor 20 A, based on ACS712 (Figure 9e), based on ACS712 this intrusive Hall effect current sensor can be used to measure currents between –20 A and +20 A, with an output ratio of 100 mV/A;

- Power Switch 10 A, based on SRD-05VDC-SL-C (Figure 9f), a mechanical relay which operates a switch. With a control line (+5 V) that when powered, establishes a connection between the terminals C*ommon* (C) and *Normally Open* (NO). The used part also includes a small LED which is enabled when the circuit between the terminals C and NO is established. An SLA-05VDC-SL-C based switch can replace the selected actuator to support impedances up to 30A;

- Temperature and Humidity Sensor, based on DHT11 (Figure 9g), from DFRobot, can work from 0 to 50 °C and humidity from 20% to 90%, and has low power consumption, with a precision of 2 °C;

- RFID/NFC Reader/Writer, based on PN532 (Figure 9h), has several wireless capabilities, it can be used to read and write RFID and to exchange data with Near Field Communication (NFC) enabled devices.

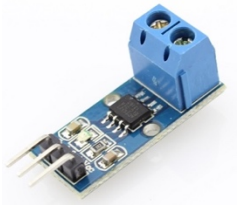
(a) Arduino R3 Uno Microcontroller



(b) Sparkfun Wi-Fi Arduino Shield, based on ESP8266

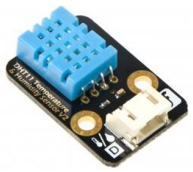
(c) Arduino Ethernet Shield, based on W5100

(d) Non-Intrusive Current Sensors, SCT-013-000 100A

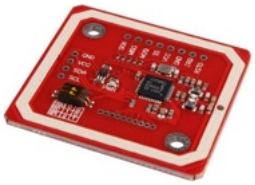


(e) (d) Intrusive Current Sensors, based on ACS712 (20A)

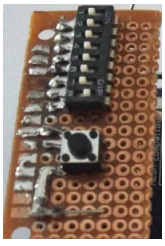
(f) Power Switch, based on SRD-05VDC-SL-C.



(g) Temperature and Humidity Sensor, based on DHT11



(h) PN532 based, Near Field Connection (NFC)



(i) Configuration DIP Switches and Reset Button

Figure 9.Hardware components used to build the IoT Unit for the proposed EV charging platform.

Configuration Options

Follows the brief comparison of the different device types that can be combined tailoring the system to a specific installation requirement.

Ethernet (Wired) / Wi-Fi (Wireless)

Most existing condominiums do not have a wired network infrastructure, based on that assumption, the use of a Wi-Fi network simplifies the deployment of the system, as no other infrastructure components are required, mainly when using the Wi-Fi network provided by the Management Unit. Newer installations or installations with specific topologies/environments with a weak Wi-Fi signal propagation (e.g., multiple floors) a cable-based network approach may be more suitable and less error-prone.

Intrusive / Non-Intrusive

The non-intrusive sensor, SCT-013-000 100A, (Figure 9d) offers the capability to measure the energy that passed through a specific IoT unit, allowing the measures to be gathered without any changes to the existing infrastructure, as the sensor only needs to be "hooked" around one of the wires of the power cable that powers the EV charger device socket. Since the system acts only as a passive observer, measuring the current that flows to the EV as no physical devices are installed between the power plug and the EV charging device, the capability to enable/disable the charging process needs to be implemented by the EV or by the charging station, exposed as a service to the charging platform. This service-oriented approach, to switch on / off the energy delivery (using standard network protocols or other communication technologies, like BLE), introduces a dependency to an external system (eventually hard to manage due to the lack of standards for this purpose). On the other hand, the intrusive approach forces the platform owner to introduce the IoT device between the power grid and the power socket, which requires some intervention in the existing infrastructure, but it is able to provide a sound solution to the platform owner, as it provides a "one-in-a-box" unit that is able to measure and control the energy delivery (enabling/disabling) simultaneously, while providing energy only to authenticated users.

Built-In vs COTS (Commercial-Off-The-Shelf) Power Switching

To enable/disable the EV charging devices, we have considered using the SRD-05VDC-SL-C (Ningo Song relay Co, Zhejiang, China) device (see Figure 9f), which, when connected to the Arduino device, can be used as a switch (the SLA-05VDC-SL-C must be used to support a current up to 30A). A different approach to support this requirement is to use a standard

TCP(Cerf & Kahn, 1974)/IP-based (Transmission Control Protocol - Internet Protocol) switch commonly available as COTS on the market.

IoT Unit Wiring

The SCT-013-000 100A current sensor (Figure 9d) is as a current transformer. The sensor is placed around the power cable (Figure 10), and on the sensor terminals, a *fraction* of the current that is flowing through the power cable can be measured.

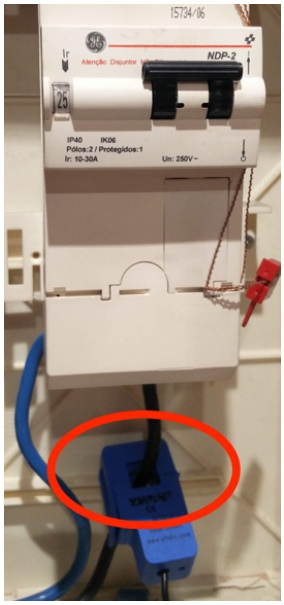
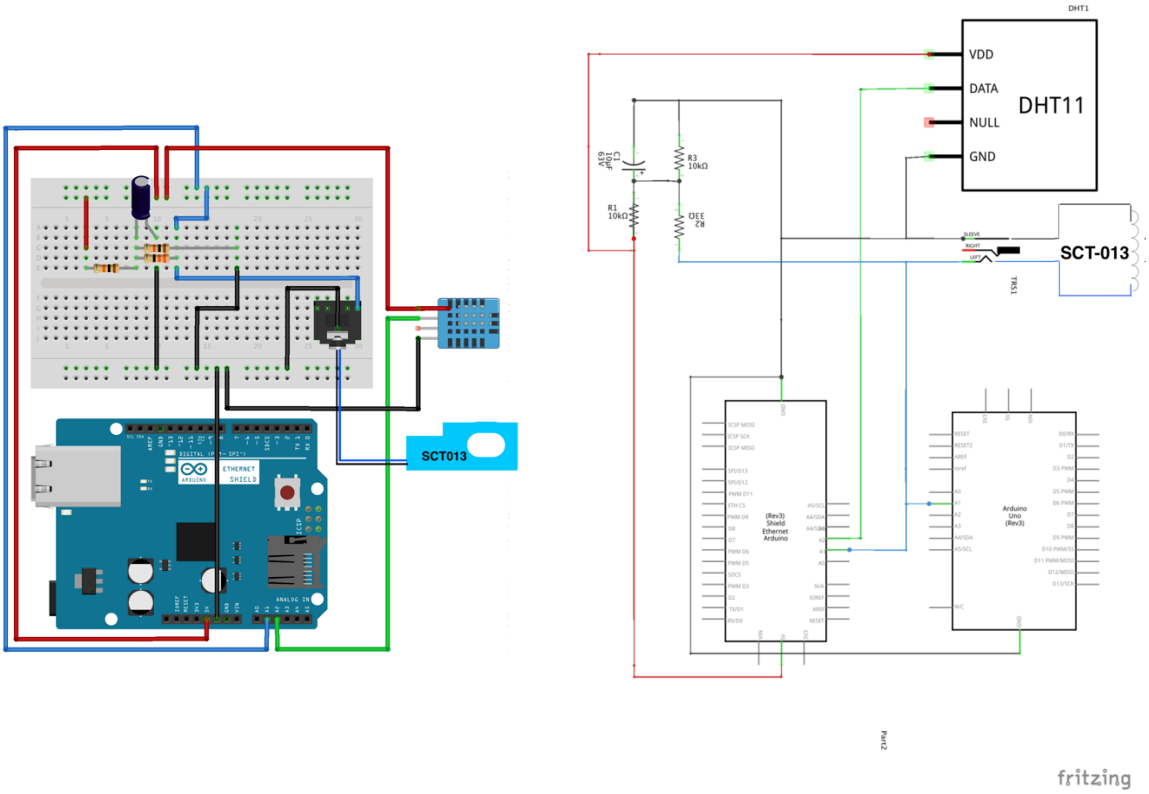


Figure 10. Non-Intrusive Sensor installed for calibration purposes.

Arduino (Figure 9a) has 5 Analogic I/O Ports (A0-A5), accepting a voltage between 0V and 5V, connected to a built-in Analogic to Digital Converter, which translates analogic value to a digital value between 0 and 1024. The current on the sensor's terminals must be transformed into voltage so that information can properly be processed. Several references are found online (Caballero, 2016; 'Learn | OpenEnergyMonitor', 2018; 'SCT-013-030 Energy Meter', 2016, p.). Figure 11 presents the components used to connect the current sensor the Arduino microcontroller as well as the connections between the components.



(a) Breadboard. *(b) Schematics.*

Figure 11.Non-Intrusive Current and Temperature Humidity Sensor Wiring.

Application Software

The software implemented in the Arduino Uno microcontroller was developed in C++ through the Arduino IDE and Microsoft Visual Studio Code. Figure 12 displays the high-level activity diagram for the IoT Unit, which can be organised into the initialisation and work cycle functional blocks. The communication with the server to obtain the configuration and sending of readings is done using the TCP (Cerf & Kahn, 1974) and HTTP (Fielding & Reschke, 2014) protocols, using the GET and POST methods.

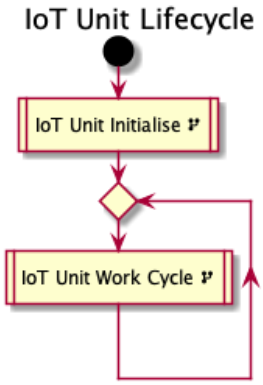


Figure 12.IoT Unit Lifecycle Activity Diagram

Figure 13 presents the sequence of activities executed on the bootstrap of the IoT Unit, that can be described as follows. As soon the IoT Unit is started it checks if the reset configuration button is pressed (Figure 9i) and in that case, the configuration stored on the EEPROM is erased, and the IoT unit is restarted.

After the previous step, the configuration is loaded from the EEPROM, and their checksum and expiration date are validated. If the configuration is absent or invalid, the device contacts the Management Unit in a well-known address to obtain the initialisation configuration data, receiving the unit configuration parameters (e.g., unit identification, network configuration, servers address, sampling rate, etc.). Two approaches can be used to identify the IoT Unit sending the request to the Management Unit. If the configuration is invalid or absent, the sensor id is read from the IoT unit dip switches (Figure 9i), a total of 64 (2^6) IoT units can be configured to obtain configuration data from the Management Unit, otherwise if the configuration is expired, the configured sensor id, stored in the EEPROM is used. After fetching the configuration data from the Management Unit and storing the configuration on the EEPROM, the unit is restarted.

Assuming that the IoT unit has a valid configuration, the network interface is initialised, and the IoT unit fetches the clock time from the Management Unit. A standard network time protocol like NTP (Mills, Delaware, J. Martin, & Burbank, 2010) could be used, but it would require the IoT unit to be able to reach the internet and access to well-known NTP server, adding a dependency to an external system, which could be mitigated by upgrading the Management Unit to provide that services increasing the load on the Management Unit. Additionally, the use of the standard internet time protocol NTP, as it relies on the UDP (Postel, 1980) protocol, a connectionless protocol from the IP stack, would increase the implementation complexity on the IoT Unit (which has limited resources).

At this stage, the unit enters in the standard operation mode, presented in Figure 14 and discussed below.

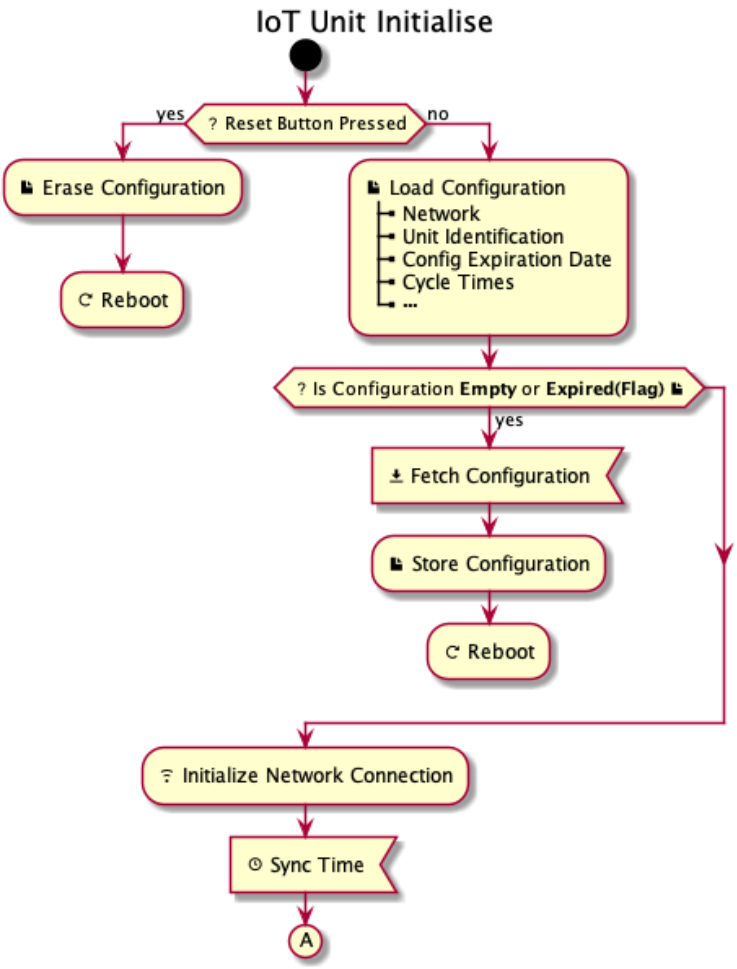


Figure 13.IoT Unit Initialise Activity Diagram

Figure 14 shows the *standard* main loop executed by the IoT unit, following the previous approach, follows a brief description of the process. The primary process of the IoT

Unit consists of an infinite loop (until the unit is restarted or powered off) in, for each cycle iteration, several activities are executed. To provide a reasonable reactivity to the RFID/NFC reader and simultaneously avoid flooding the Management Unit with a high number of sensor readings, the main loop has a time period (every 10 secs, by default) shorter than the time period used by the IoT unit to sync with the Management Unit and send the sensor reading (every 1 minute, by default).

On the beginning of each iteration the IoT unit checks for the availability of RFID/NFC data (if an RFID/NFC was attached to the unit) and, if data is available the unit forwards the information to the Management Unit. The outcome of the operation is not being presented to the user, and nevertheless, as a future extension, it could be used to provide some feedback to the authenticating user related to the operation. If the Management Unit successfully processed an RFID/NFC reading or if the amount of time between the last sync with the Management Unit, the IoT Unit reads the inputs from the sensors (Temperature, Humidity, and Current) and sends that information to the Management Unit, which as an outcome of the operation returns status information and operations to be executed by the IoT Unit. If the unit fails to send the information to the Management Unit, it will retry the request for a specific number of attempts and reboot the unit if the maximum number of attempts is reached (which will switch off the energy delivery).

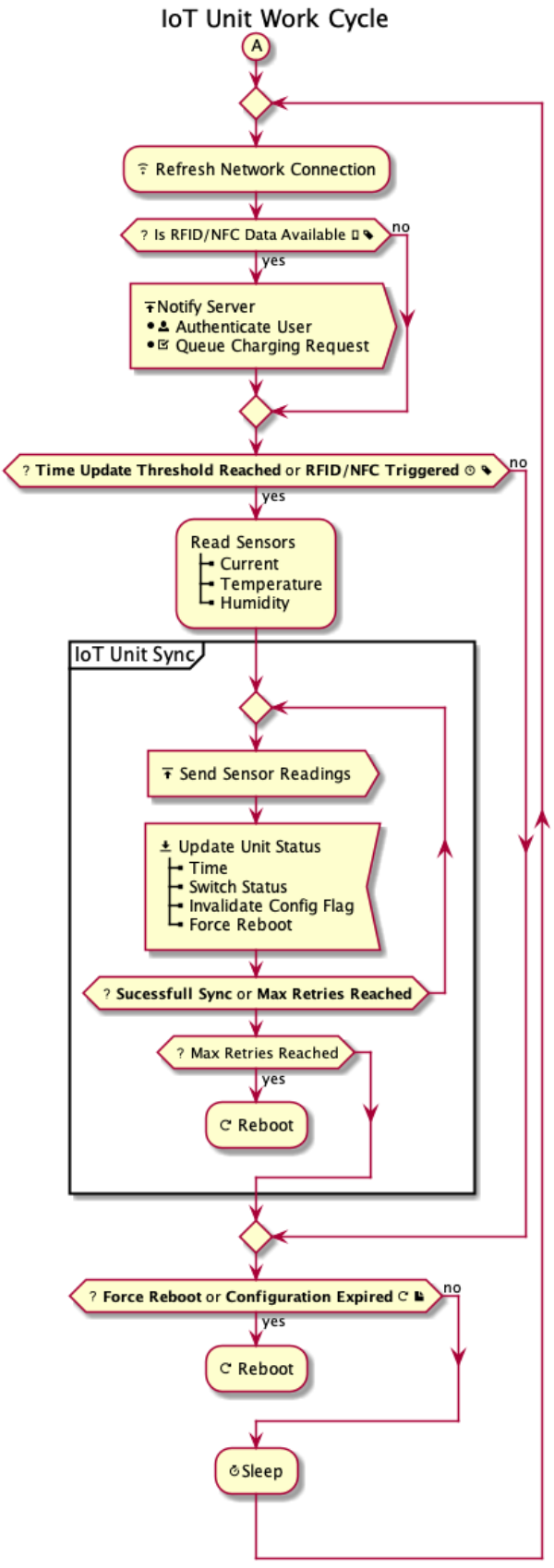


Figure 14.IoT Unit Work Cycle Activity Diagram.

Communication with Management Unit

Figure 15 presents a sequence diagram for the communication flows between the IoT Unit and the Management Unit, as well as the surrounding associated processing. Since the IoT Unit has reduced computing resources, the information is exchanged using a character-oriented protocol, sharing information either on the request URI or on the message payload.

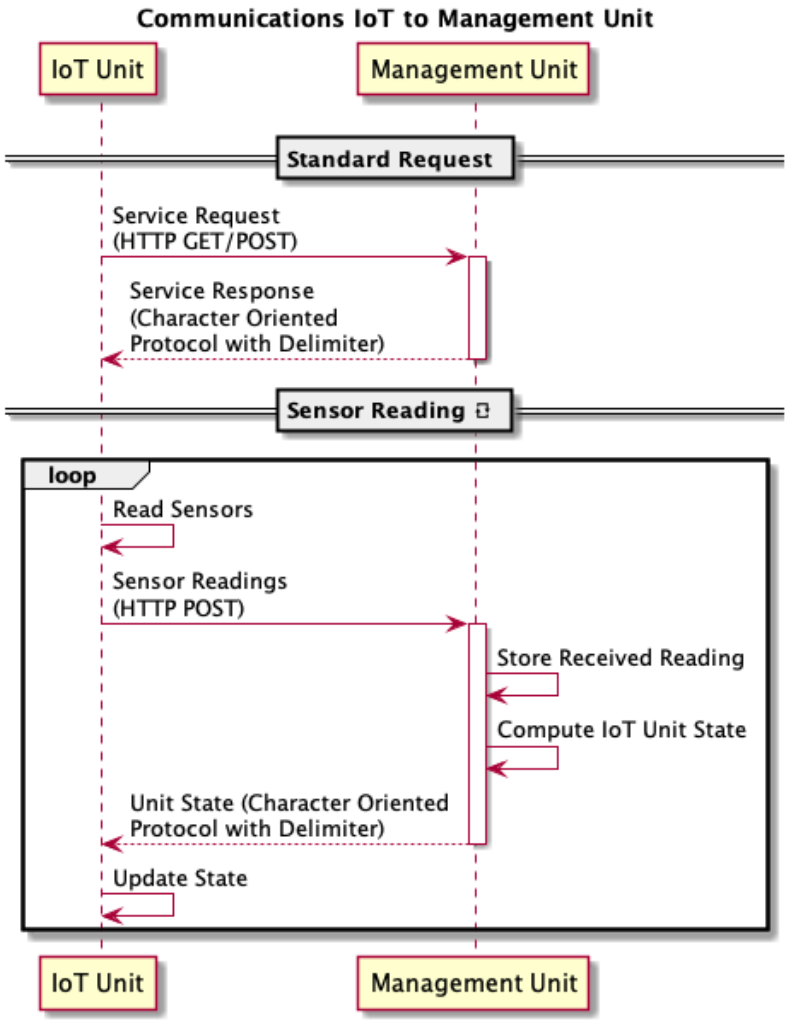


Figure 15.Communication between the IoT Unit and Management Unit

4.1.2.Management Unit

The Management Unit coordinates and orchestrates the IoT Units connected to the platform. This section is divided into the following subsections: Hardware; Software Infrastructure; Application Software and Services Infrastructure; Management Services; Management Web Application.

Hardware

The Management Unit was built using a Raspberry Pi 3 Model B+ hardware (Figure 16), and the Raspbian operating system.

The unit was configured as a Wi-Fi access point, setting up the network to allow Wi-Fi communications between all the platform components (Management Unit, IoT units, and Mobile Devices / App).

This configuration allows the deployment of a completely self-contained, pluggable, low-cost solution, without requiring any other infrastructure components (apart from the energy power grid), while increasing the security of the overall solution by reducing its exposure to external network threats.

Complementarily, if deployed in a location with existing network support, the Management Unit can be connected to the network using the RJ45 Ethernet connector of the Raspberry Pi, allowing the platform to benefit from the existing infrastructure and to eventually be deployed in setups where the use of a Wi-Fi network may not be available or the most suitable option, for instance, a multi-level condominium parking lot, or a parking lot spread over several areas and sharing only one Management Unit.

The Raspberry Pi 3 Model B+ is the final revision in the Raspberry Pi 3 family, the most relevant characteristics, extracted from the component technical specification are:

- Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
- 1GB LPDDR2 SDRAM
- 2.4 GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
- Gigabit Ethernet over USB 2.0 (maximum throughput of 300 Mbps)
- One extended 40-pin GPIO header
- Full-size HDMI
- 4 USB 2.0 ports
- DSI display port for connecting a Raspberry Pi touchscreen display
- 4-pole stereo output and composite video port

- Micro SD port for loading the operating system and storing data
- 5V/2.5A DC power input
- Power-over-Ethernet (PoE) support (requires separate PoE HAT)

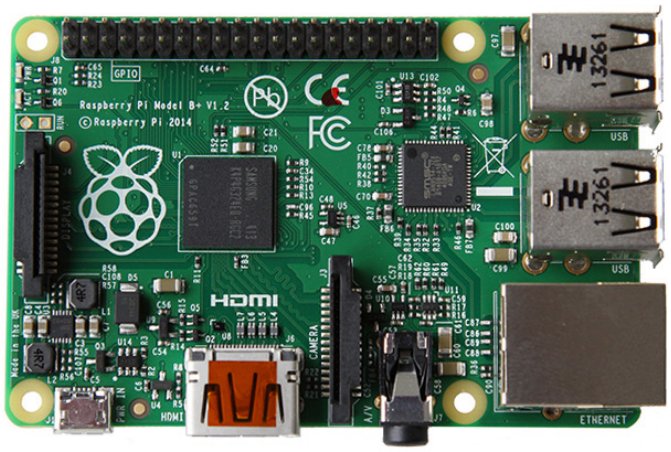


Figure 16.Raspberry Pi Model 3+.

Software Infrastructure

Figure 17 displays the software infrastructure components required to implement the Management Unit platform elements (IoT unit, Mobile App). The primary role of each component will be described in the current section.

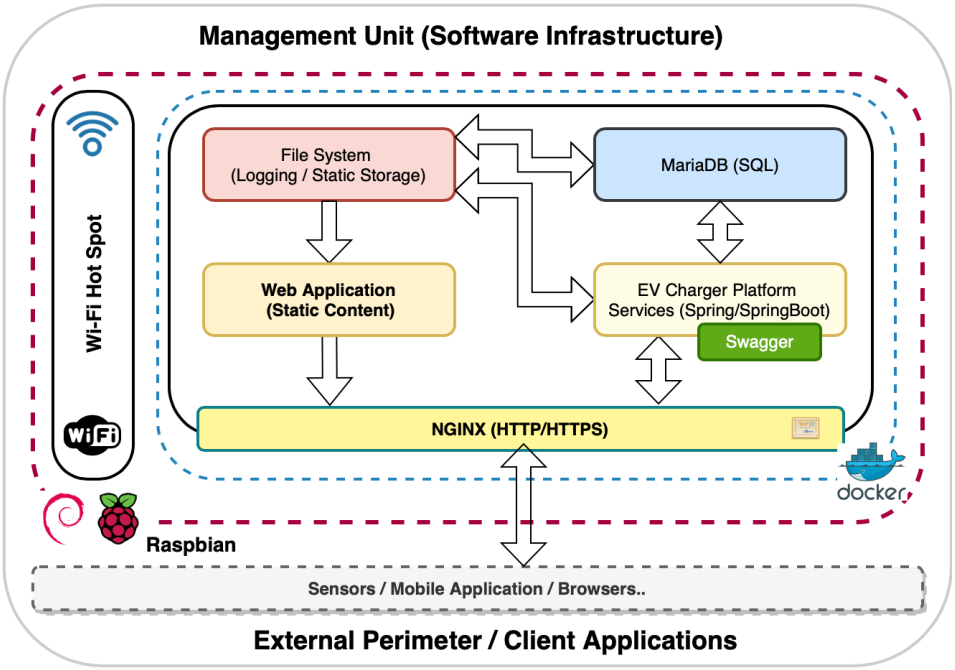


Figure 17.Software infrastructure and flows of information.

From the software perspective, the Management Unit uses as Operating System the Raspbian, which is a Linux, Debian based operating system built and optimized to run on the Raspberry PI, configured as a WiFi Access Point ('Setting up a Raspberry Pi as a Wireless Access Point—Raspberry Pi Documentation', 2019) to provide WiFi network access to the Mobile Application and to the IoT Units. The Management Unit relies on the following infrastructure software/services:

- File System

  As being one of the core services provided by the operating system, apart from the *standard* use, in the context of the current application, the file system is used directly as static storage (storing the static contents to be served by the webserver) and as a storage unit for archiving the platform logs;

- MariaDB

  Relational Database Management System (DBMS), serves as the primary location to store all the information managed by the platform;

- NGINX (HTTP/HTTPS)

  Web server routes the requests received on the standard HTTP(80)/HTTPS(443) ports to the services endpoints exposed internally inside of the Management Unit, acting as a proxy between the "external world" and the services layer. It also serves all the static contents requests (the Management Unit administration is a single web page application that consumes the services exposed by the service layer). Aiming to guarantee the security of the communications between the Management Unit and the Mobile App, all the services exposed to the Mobile App use a secure channel (HTTPS protocol). The use of the HTTPS protocol requires a digital certificate to encrypt the communications. Although a self-signed certificate can be used, aiming to avoid communication errors, a trusted certificate, freely provided by Let's Encrypt, was installed;

- EV Charger Platform (Spring/SpringBoot Framework)

  The platform services, built using the Spring Framework and SpringBoot, are exposed by the framework as a set of Representational State Transfer (REST) endpoints. Exposing an API (Application Programming Interface) in a standard language, that can be easily used by third-party applications, using interoperability tools available on the market, allowing the development of custom-made integrations (for instance, to integrate the platform with a condominium management system);

- Swagger Framework

  Aiming to document the service APIs exposed, the Swagger Framework was integrated into the EV Charger Platform, to generate the services documentation automatically. Figure 18 displays a screenshot of the self-generated documentation.

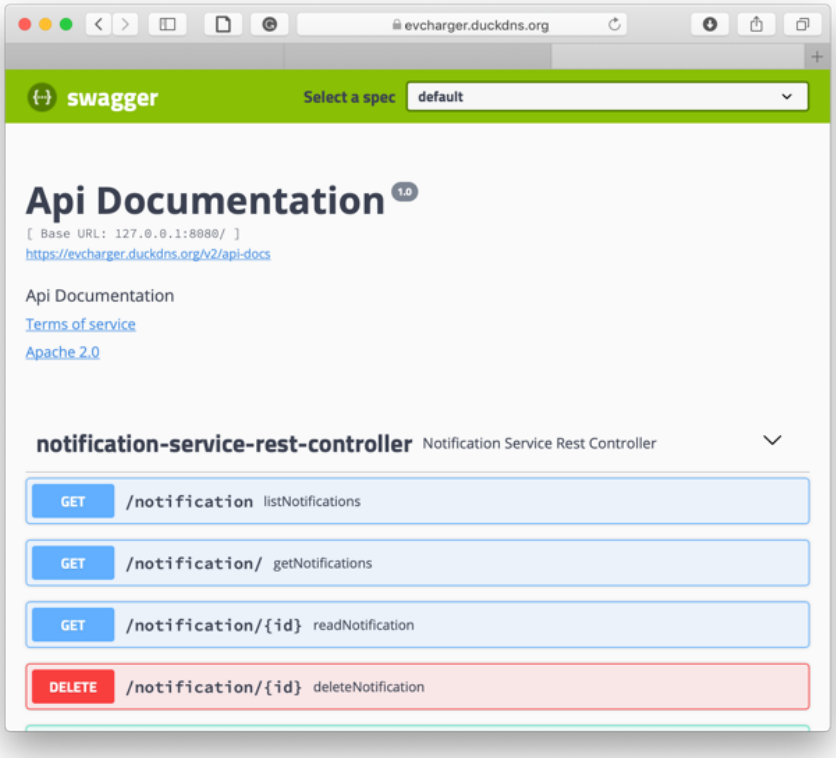


Figure 18.Self-generated API (to support interfacing with third-party applications).

- Web Application

  The Management Unit implements a browser-oriented web interface for management purposes. This web interface, implemented using the Angular Framework, uses a single page application architecture, consisting in a set of static files served by NGINX web server to the client browser, which invokes the services provided by the EV Charger Platform.

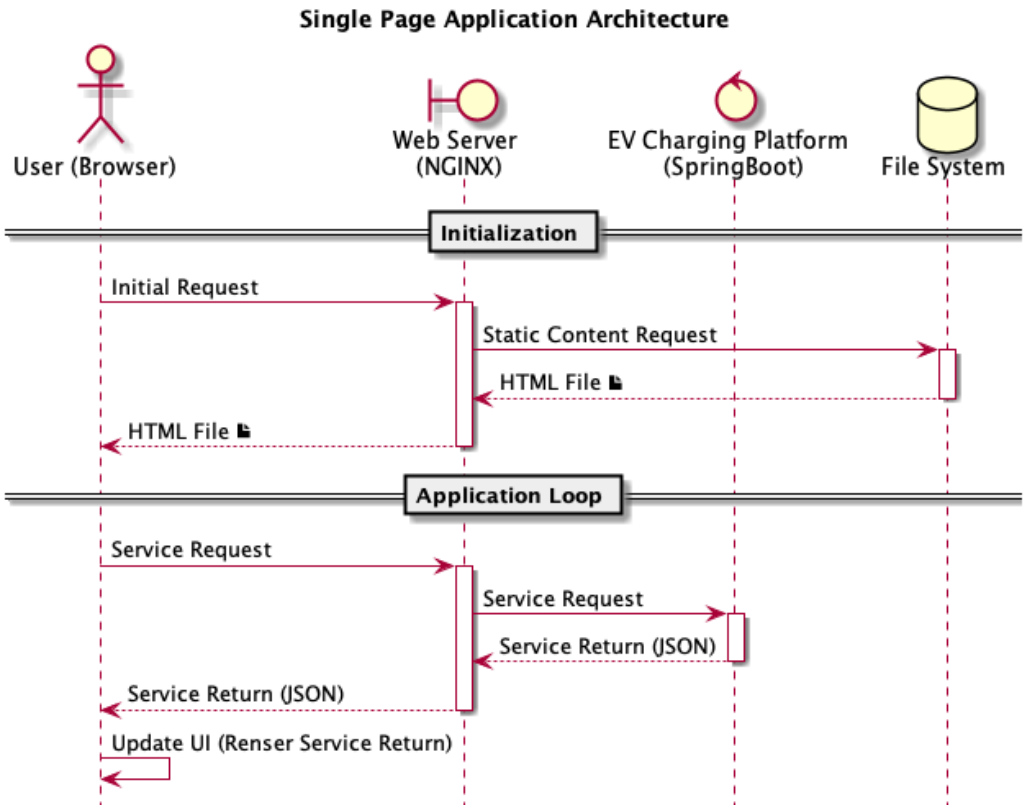


Figure 19.SPA Architecture Diagram.

Although all the infrastructure server software is being deployed directly on the top of the operating system, a container-based approach (for instance, Docker (Ismail et al., 2015)) can also be applied to remove all the dependencies of the between the hardware/operating system and infrastructure/applicational software, encapsulating all the infrastructure and applicational software in a completely self-contained *box* (container), that can be deployed in other operating systems or hardware platforms.

Applicational Software

The applicational software, following a simple responsibility principle (SRP) (Martin & Martin, 2018) is composed by a set of services of modules, exposed with the SpringBoot container, as a set of Representational State Transfer (REST) endpoints. The services are implemented, following a similar layering pattern, presented in Figure 20, where each layer have specific roles:

- Service Layer: Acts as a mapping service, translating the external representation of the information to the internal representation;

- Business Layer: All the application behaviour is defined in this layer, and any interaction between layers made exclusively through the interface provided at this level;

- Persistence Layer: This layer maps the internal representation of the information to the representation used by the database engine.

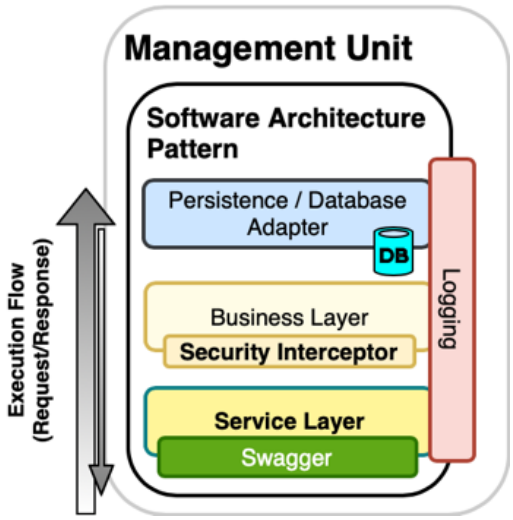


Figure 20.General software architecture pattern.

Figure 21 presents the application level services or modules that constitute the EV charging platform, the description of the implemented services, and their contribution to the overall platform is presented in the current section.

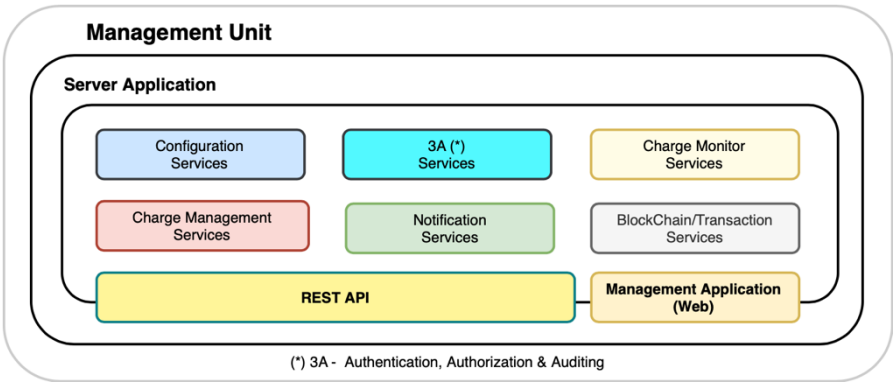


Figure 21.Management Unit services.

- Configuration: Provides a set of services required to configure the EV platform, allowing the user to define several platform parameters, such as the existing sensors and their configuration (e.g., network configuration, maximum current, accounting frequency, measure period), as well as the groups of sensors (e.g., sensors inside the same group, maximum load per group);

- 3A (Authentication, Authorization, Auditing): This module has a central role in the entire platform. It is responsible for centralising all the operations related to user/system authentication ("who is who "), authorisation ("what can do") and auditing ("what was done"). Apart from implementing the set of operations to manage the user access to the platform, it also implements the implicit authentication (J. C. Ferreira et al., 2014) to validate the charging request automatically, based on the current user's usage pattern;

- Charge Monitor: This module collects and processes all information generated from the installed sensors to update the EV charging records and detect the end of the charging events, as well as any anomalies on the charging process (e.g., exceeding the nominal current, temperature, charging time), and triggering eventual notifications when required. This module also collects the user's usage pattern to estimate the power needs for the current charging process, as well as estimate the

leave time of the EV from the charging plug, if that information is not provided explicitly by the user;

- Charge Management: If the installation has the capability to enable or disable the EV charging process, by the use of network-controlled charging devices or by the use of charging switches attached to the sensor unit, the module enables or disables the charging of the EV, aiming to properly distribute the available charging windows between all the EVs connected to the charging group, based on the charging requirements and the amount of time that the vehicle will be connected to the charging device and using the information provided explicitly by the user or inferred by the platform based on the users usage pattern;

- Notification Services: This module provides all the notification related services to the platform, routing the system-generated notifications to users that had subscribed to that notification (i.e., vehicle charged, abnormal charge pattern, etc.);

- Transaction: This module supports all the "financial" related operations, aggregating the information related to the charging operations providing reporting capabilities to allow the financial management and analysis of the platform usage. In a blockchain integrated approach (presented only as conceptual model), this module should also be responsible by recording the changing events in the blockchain ledger, allow the platform managers to transfer "charging tokens" to the user's wallet (if not using an open public crypto-currency network), monitor the reception of user's transferred credit to start the charging process and return the unused credit to the user's wallet.

4.1.3.Mobile Platform

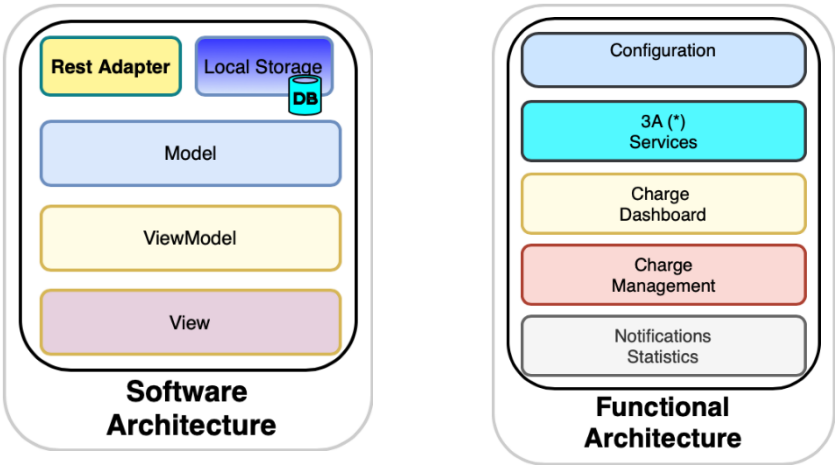
Hardware

No specific hardware requirements were identified for a mobile device running the Mobile App as it acts as a consumer of the services exposed by the Management Unit, apart from being able to connect to a network where the Management Unit is reachable.

Software

Aiming to allow the EV owners to use the platform, a Mobile App was developed in C#, using a development platform and framework Visual Studio and Xamarin.Forms, respectively. The choice to use a multiplatform development platform allows developing software for several platforms simultaneously (Android, iOS, and UWP), minimising the number of changes between platforms.

Figure 22 presents the software architecture pattern used and functional organisation of the Mobile App.



(*) 3A—Authentication, Authorization and Auditing

Figure 22.Mobile App functional and software architecture views.

From a functional perspective, the Mobile App is split into several modules enforcing the separation of concerns between each functional component. On the software architecture perspective, as the Mobile App is mainly a client or a *frontend* for the services provided by the Management Unit, it is implemented following a straightforward application of the Model-View-ViewModel (MVVM) pattern, an extension of the Presentation Model (PM) pattern (‘Presentation Model’, 2004), frequently used in Xamarin.Forms applications and other

software platforms. On software architectural pattern, each logical layer has a clear separation of concerns, briefly described below, and graphically presented in Figure 23:

- View

  Implemented with XAML (eXtensible Application Markup Language), a declarative language used to design and structure the user interface.

- View-Model

  Layer that intermediates the relationship between the View and the Model, binding the information and actions between the model components.

- Model

  Representation of the application data.

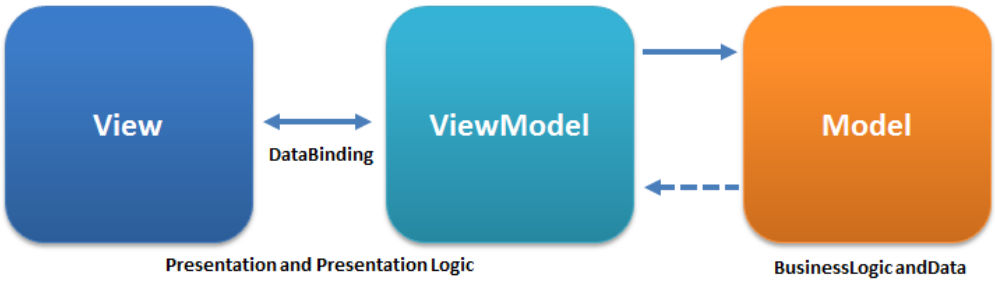


Figure 23.MVVM Pattern Representation[3].

To store data locally (configuration) and to establish communication with the Management Unit, two other components are considered:

- REST Adapter

  As the Mobile App relies on services provided by the Management Unit, this component acts as a proxy between both entities;

- Local Storage

  Small information repository managed by the Mobile App to store configuration data in the mobile device.

[3] Image Source: https://en.wikipedia.org/wiki/Model–view–viewmodel#/media/File:MVVMPattern.png

Communication with Management Unit

Figure 24 presents a sequence diagram for the communication flows between the IoT Unit and the Management Unit. The information exchanged between the Mobile App, and the Management Unit uses the HTTPS protocol (Rescorla, 2000), relying upon digital certificate installed on the server, to securely transfer information by encrypting the exchanged messages. Aiming to the benefit of the authentication and authorisation services provided by the Spring Framework, the standard HTTP (Fielding & Reschke, 2014) authentication headers are communicated inside this secure channel. Stronger authentication schemes could be supported by the use of client certificates at the webserver level (NGINX) to authenticate the Mobile App requests on the server; however, this was not considered for the current implementation to avoid the complexity of introducing a Public Key Infrastructure (PKI) in a platform relying in a Raspbery PI.

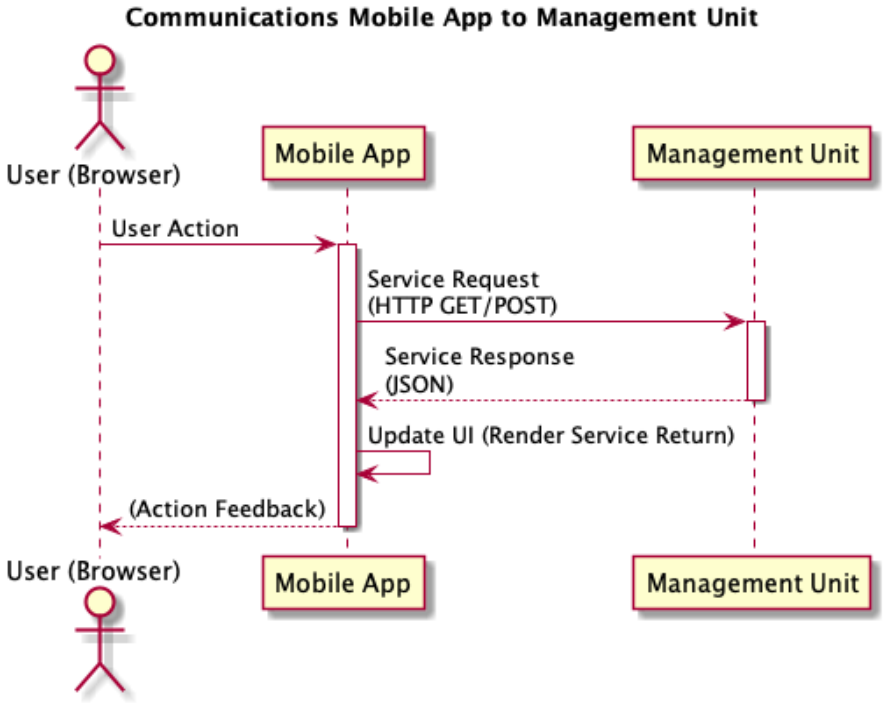


Figure 24.Communication between Mobile App and the Management Unit.

Initiating the Charging Process.

Despite all the communication between the Mobile App and the Management Unit be secured with the use of an HTTPS certificate installed on the Management Unit, that authentication scheme only guarantees the authentication of server platform, in this case, the Management Unit. Aiming to achieve that non-repudiation of the charging request, due to it's financial relevance and, avoiding to rely exclusively in a user/password security scheme, which only guarantees that the Mobile App user, knows the user credentials for a specific user, a signing scheme was implemented to secure the queue (start) charging operation. For this purpose at the user registration instant, a private/public key is generated on the mobile device, being only the public key shared with the Management Unit, Figure 25 presents the registration flow using a UML sequence diagram.

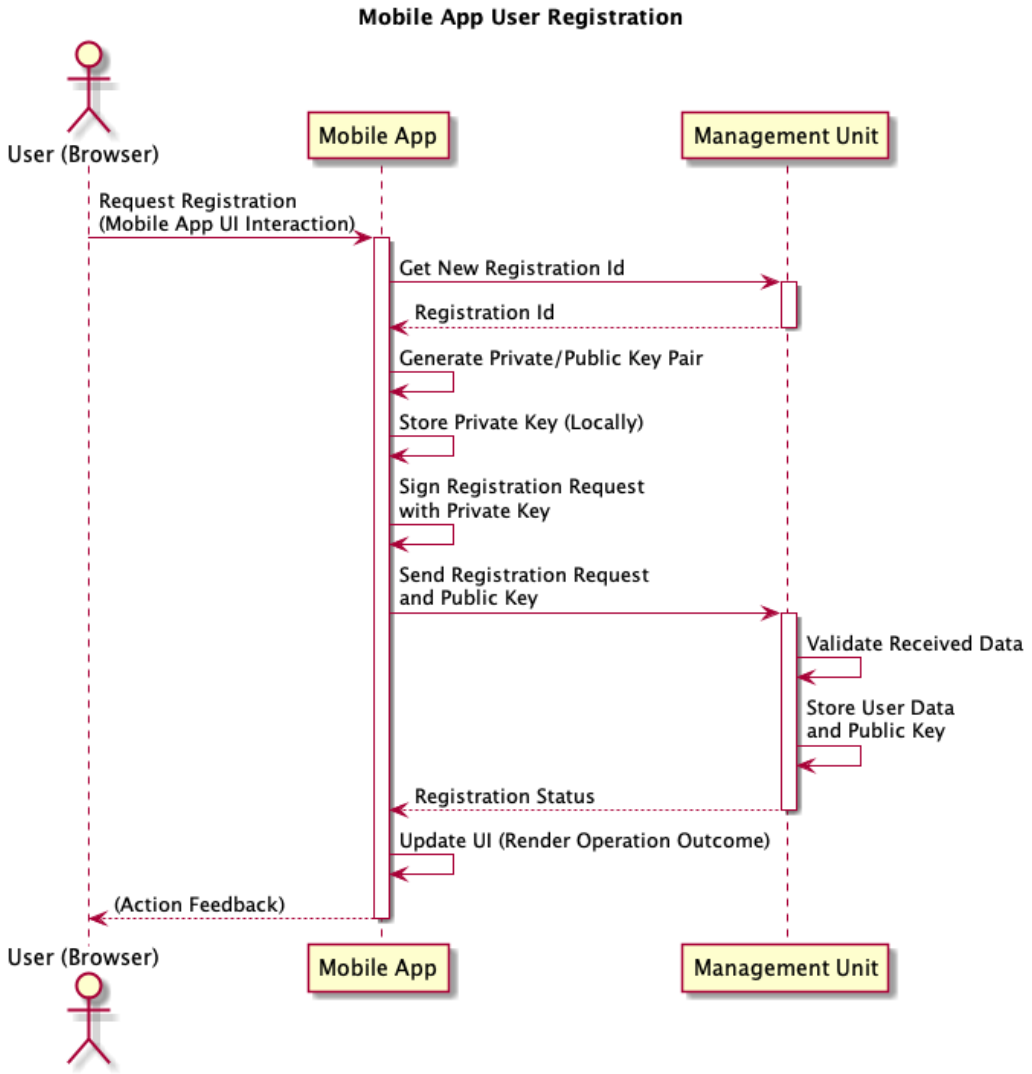


Figure 25.Mobile App User Registration.

Figure 26 presents a sequence diagram from starting (or queuing) charging operation. When requesting the EV Charging, the request is signed by the user using it's private key and sent to the Management Unit, which validates the signature of the message against the user's public key, stores the operation request message as it establishes a binding contract between the user and mobile device and the charging platform and, queues or starts the charing operation if the power being used is below the shared space power limitation.

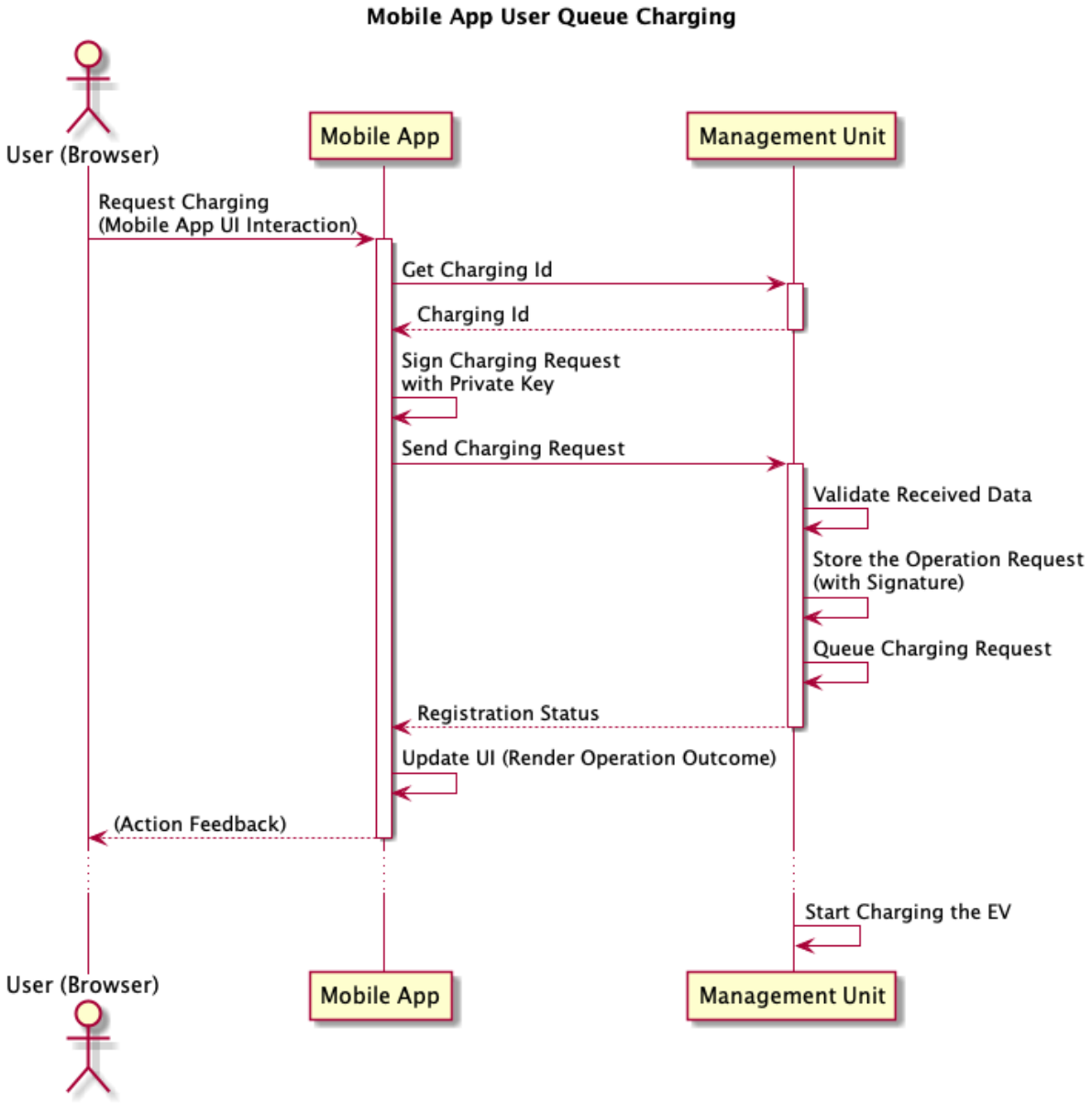


Figure 26.Mobile App User Queue Charging Operation.

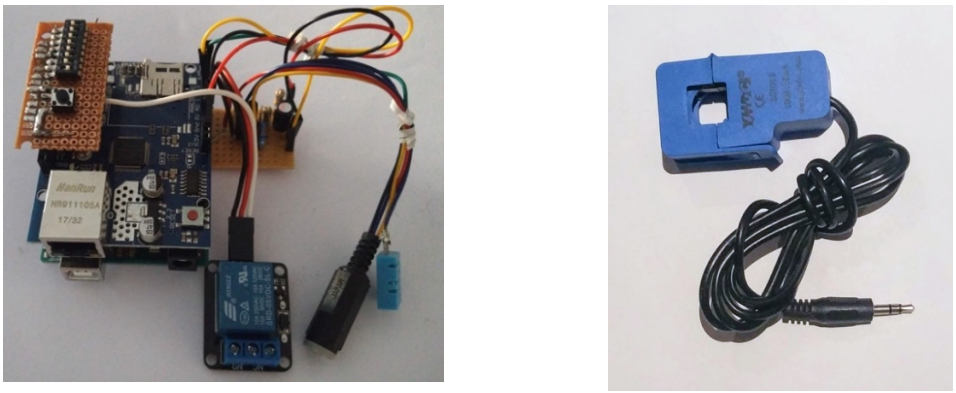
4.2.System Development

The current section presents some of the developments achieved during this project.

4.2.1.IoT Unit

Figure 27 presents one developed prototype, with the following components:

- Configuration dip switches and resets button;
- Arduino board;
- Ethernet Arduino shield;
- Power switch;
- Temperature sensor;
- Plug to the current sensor;
- Current sensor (Figure 27b).



(a) IoT Unit

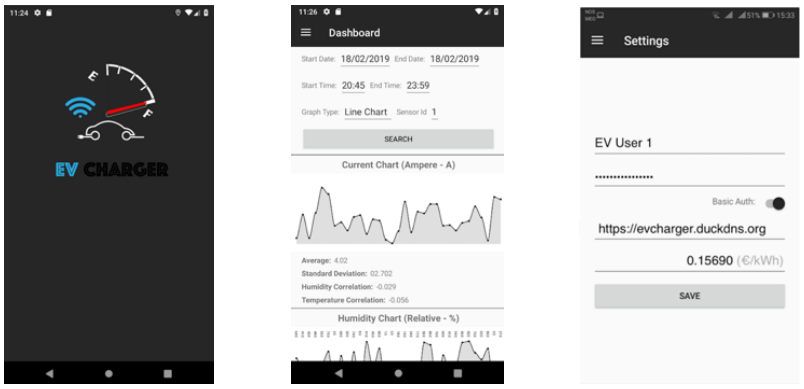
(b) Current Sensor

Figure 27. Implemented Prototype.

4.2.2.Mobile App

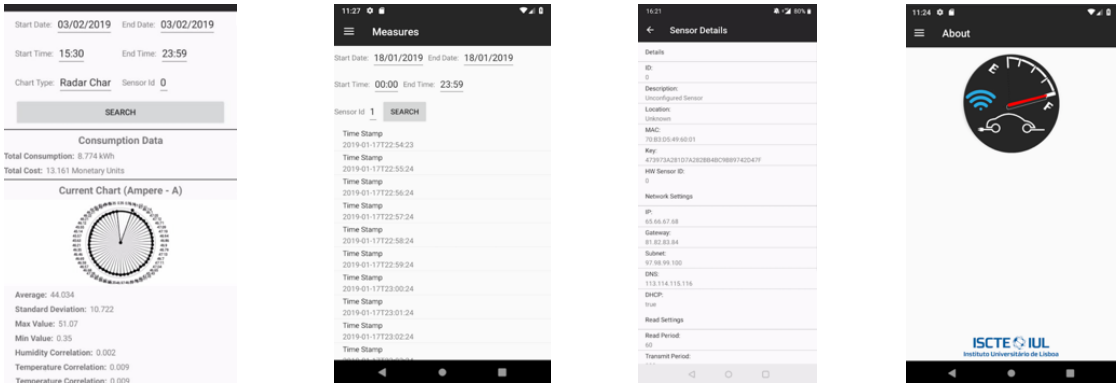
Apart from the EV charging process, which can be considered the crux of the system, the Mobile App also implements several features that, although not as relevant, are required to achieve a production-grade design stage. Figure 28 and Figure 29 shows screenshots for some of the implemented features:

- Application splash screen, Figure 28a and Current usage pattern, Figure 28b;
- Application settings, Figure 28c, and Energy costs calculated based on kWh and statistical sensor measures, Figure 29a;
- List of sensor readings received, Figure 29b;
- Sensor configuration details, Figure 29c and About screen, Figure 29d.



(a) Splash-screen *(b) Current Graphics* *(c) Application Settings*

Figure 28.Mobile App screenshots.



(a) Calculated Power Costs and Current *(b) Sensor Data* *(c) Sensor Details* *(d) About Screen*

Figure 29.Mobile App functionalities.

Initiating the EV charging process is a vital function of the system, and simultaneously the most frequent operation, as charging the EV is the purpose of the entire system, considering that, special arrangements should be observed aiming to minimise the user's effort on that task.

Requiring the EV owner to connect to a network where the Management Unit can be reached, assuming that the system operates in a closed network, to be able to start the charging process adds a non-practical, time-consuming operation. Attaching an RFID/NFC reader to the IoT Unit and using an RFID TAG or eventually, the user's mobile NFC capabilities to authenticate the user on the system, increase the user experience while initiating the charging operation. On this case, the system uses the information from the previous operations to confirm the user authentication, estimate the power needs and the amount of time that the EV will be connected to the charging plug (to forecast the power/time usage).

Complementary to this process, a more controlled approach can be used, when an RFID TAG, an NFC mobile-enabled device is not available or the IoT unit is not attached to an RFID/NFC sensor, or eventually if the vehicle owner needs to configure the charging process (setting parameters such as the amount of the battery energy according to the state of charge (SoC), the amount of time connected to the platform, time-window for charging). On this case, the charging process can be initiated by connecting to the network where the Management Unit is reachable, eventually to the Wi-Fi network provided by the Management Unit, and starting the process, providing the required information, Figure 30 shows the application interfaces to initiate a charging process and to stop the charging process.

(a) (b)

Figure 30.Mobile App interface for starting (a) and stop (b) the EV.

4.2.3.Management Unit / Web Application

The Management Unit exposes a web application to allow the EV platform managers to monitor, configure, and operate the platform. It also provides to the platform users a complementary user interface that, although supporting only a reduced set of the operations available on the native Mobile App, allows the users to interact with the platform using browser-only technologies, available in a broader range of devices Figure 31 displays some screenshots for the Management Unit web application.

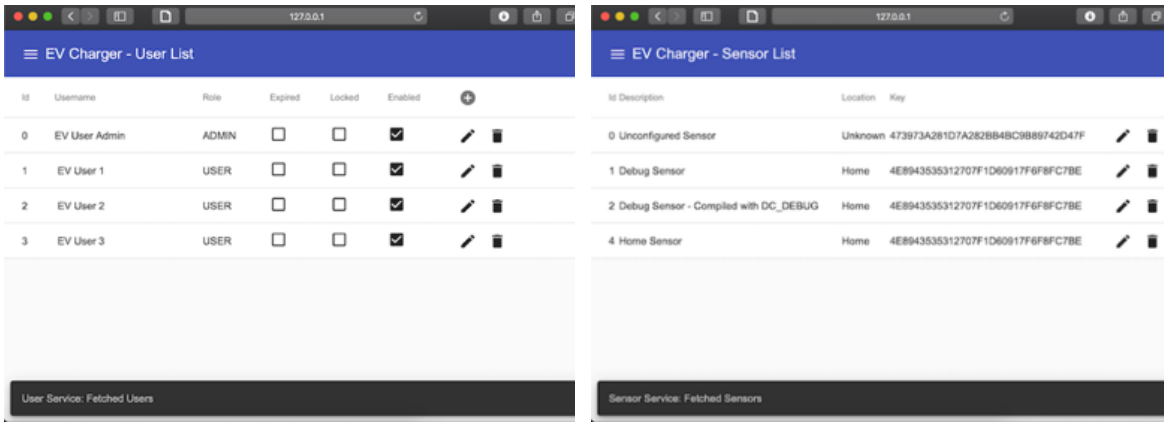


Figure 31.Web application: Users list (left) / Sensors list (right).

4.3.System Testing and Evaluation

The current sections describe some test approaches used during the development of the prototype, as well as the sensor calibration process.

4.3.1.Sensor Calibration

As previously explained the non-intrusive current sensor, acting as a transformer, provides at the sensor terminals a fraction of the electric current that is passing thought the electric conductor being measured. The current analogic reading is then translated to a tension that is connected to one Arduino ADC terminals and transformed in a digital value between 0 and 1024. Aiming to establish a relation between the current reading and the digital value presented on Arduino port, several measures were collected simultaneously on a calibrated amperometric clamp and its digital counterparty. Figure 32 shows the amperometric clamp and the current sensor attached to the primary power wire at the power switch fuse box.

Figure 32. Calibration Process - Amperometric Clamp (14.63 A reading).

Table 2 presents the values read during the calibration process, displayed in Figure 33, on the X-axis is displayed the sensor reading, and on the Y-axis, the value read on with the amperometric clamp.

Table 2.Amperometric Clamp vs Sensor Reading

Sensor Reading	Amperometric Clamp Reading (Amperes)
16,5	2,45
20,6	3,32
6,93	1,29
6,87	0,9
35,34	6,9
54,15	10,8
89,33	16,7
55,92	11,12
90,18	17,2
113,23	22,75
115,35	25,96
66,02	13,1
19,77	3,17
0,14	0

Applying a linear regression to the data obtained from the sensor during the calibration process allows us to devise the equation on (1), where x represents the sensor value and f(y) the current value. The R^2 measure is a statistical measure (between 0 and 1) that represents how close the data fits the model thought the use of the linear regression. A value of 0.99 allows concluding that the relation between the depended variable and the independent variable have a strong linear relation and the function obtained explains the relation between the sensor reading and the current measure. Although no studies were conducted to verify the precision and accuracy of the sensor, the sensor technical specification reports a non-linearity of +/- 3%.

$$f(x) = 0.2103x - 0.6785 \quad (1)$$

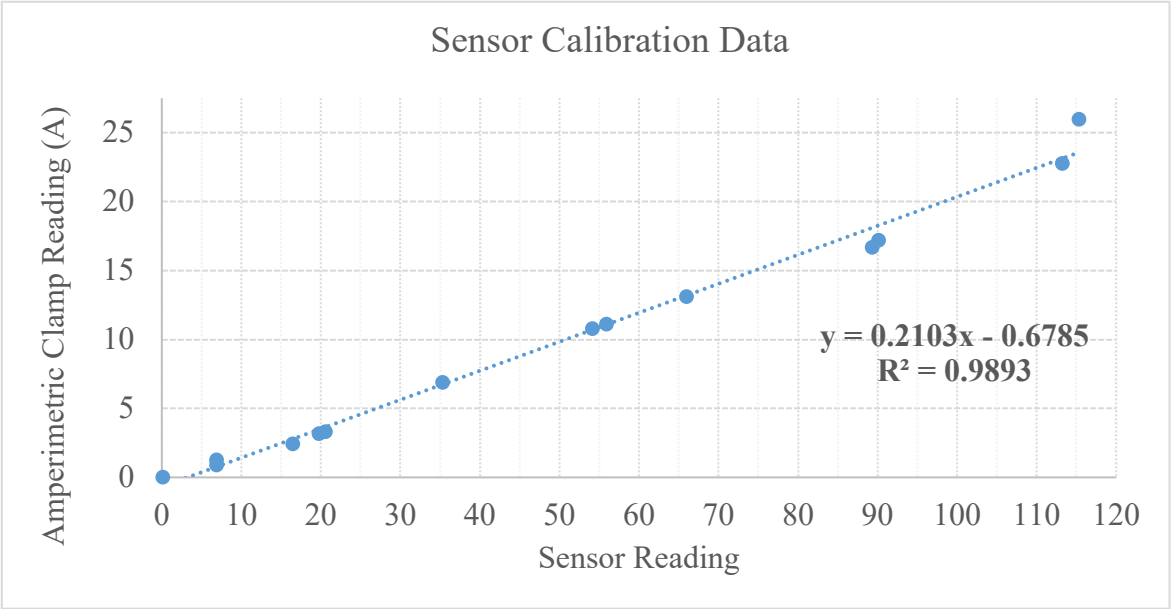


Figure 33.Sensor Calibration Data.

4.3.2.System Testing

Aiming to evaluate the system reliability, the behaviour of the system was evaluated against the common failures scenarios.

Network Failure

If the system experiences a network failure between the IoT Unit and the Management Unit, no current measures will be forwarded to the Management Unit. To reduce the impact of the lack of communications, as soon the failure is detected (by default on every minute), the digital switch is switched off, stopping the delivery of energy to the EV Chargers. As soon the situation is recovered the digital switch returns to its previous state. On the absence of communications between the Mobile App and the Management Unit, the user will not be able to start the charging process, unless the IoT Unit has an RFID/NFC reader.

Power Failure

In the absence of energy power either on the Management Unit or the IoT Unit, the behaviour is similar to the behaviour observed during a Network Failure (i.e., no energy will be delivered to the EV Charging device). However, in the case of a power failure on the Management Unit, any ongoing charging operations be terminated.

4.3.3.Software Testing

Although not used extensively on all developed components, the following strategies were evaluated and used during the software development, aiming to increase the quality of the developed prototype, to reduce the development times.

Unit Testing

Aiming to automate the testing for some of the developed components and also as a proof of concept, a unit testing framework was used to test some for the Management Unit. Figure 34 presents the unit testing approach.

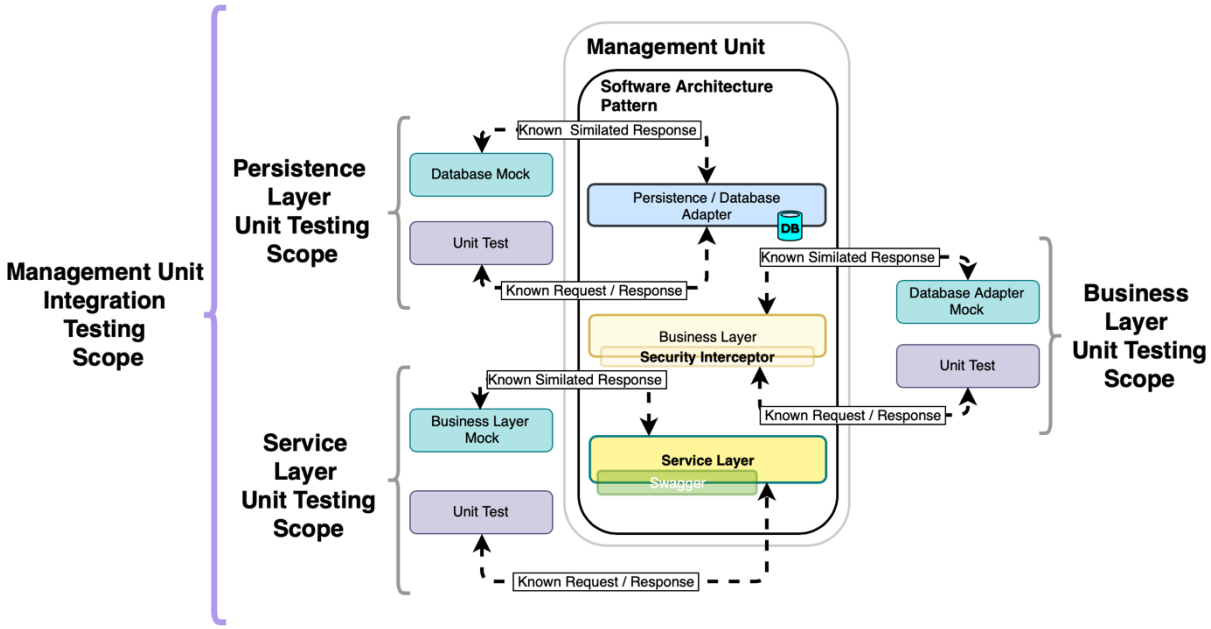


Figure 34.Management Unit Testing Strategy.

The unit testing, as the name defines it, aims test exclusively a *unit* of software, where *commonly*, a *unit* can be considered as a pure function or a *functional unit*, indivisible at that functional level. It (the unit testing) relies on the principle that for a specific input is known the expected output so that input is provided to the functional unit being tested and the answer generated by the functional unit is tested against the expected answer. Since the tested functional unit can rely on other functional units, to avoid that dependency, that could lead to an inconclusive answers, as the test wouldn't be in the control of all the variables, all the requests to other units apart from the unit being tested are mimicked or mocked with forced answers to guarantee that the test controls all variables or context.

Conceptually, a similar approach could be applied to all the components, however, due to the inherent complexly and effort required, was considered out of the scope of this work.

Partial Integration Testing

Due to the number of "moving parts" of the system, the development of an automated test strategy for integration testing was not evaluated.

Nevertheless, and aiming to reduce the dependency between the components IoT Unit, Management Unit and Mobile App, and to test the integration of the software layers inside of each component, and reduce the troubleshooting effort, a simulator strategy was used to mimic the behaviour of the counterparty system, allowing to test and develop a specific component without depending on the correct or complete implementation of the other component. The following examples are the shape of this approach:

- On the IoT Unit, to remove the dependency of having the sensors wired, if a specific digital port was enabled (+5V) at bootstrap, the sensor initialisation/reading code was skipped, and the IoT Unit provided random readings allowing to test the Management Unit without having physical sensors connected;

- On the Management Unit, to be able to perform the development and test without relying on the IoT Unit, a small simulator (active) was used to simulate the requests send by the IoT Unit using HTTP, allowing to test the Management Unit processing and response to that requests;

- On the Mobile App, some fixed Management Unit answers to specific requests were simulated to be able to develop and test the Mobile App independently of the completeness and correctness of the implementation of the Management Unit.

4.4.System Deployment

The developed system was deployed to shared place in a condominium, where three EV owners shared the condominium electric installation available at parking places for 3.5 months.

Each sensor was configured to generate one sample each minute, allowing further study of the current load patterns during a charging event. A set of three EVs (all Leaf vehicles with 24 kWh battery capacity) and three independent sensors (Sensor 0; Sensor 1; Sensor 2) were deployed. Figure 35 presents the diagram of the test environment for the case study. Due to physical constraints of the installation, the charging adapter connected to Sensor 0 was directly connected to the power grid, without one intermediate switch ("always-on" on the scheme).

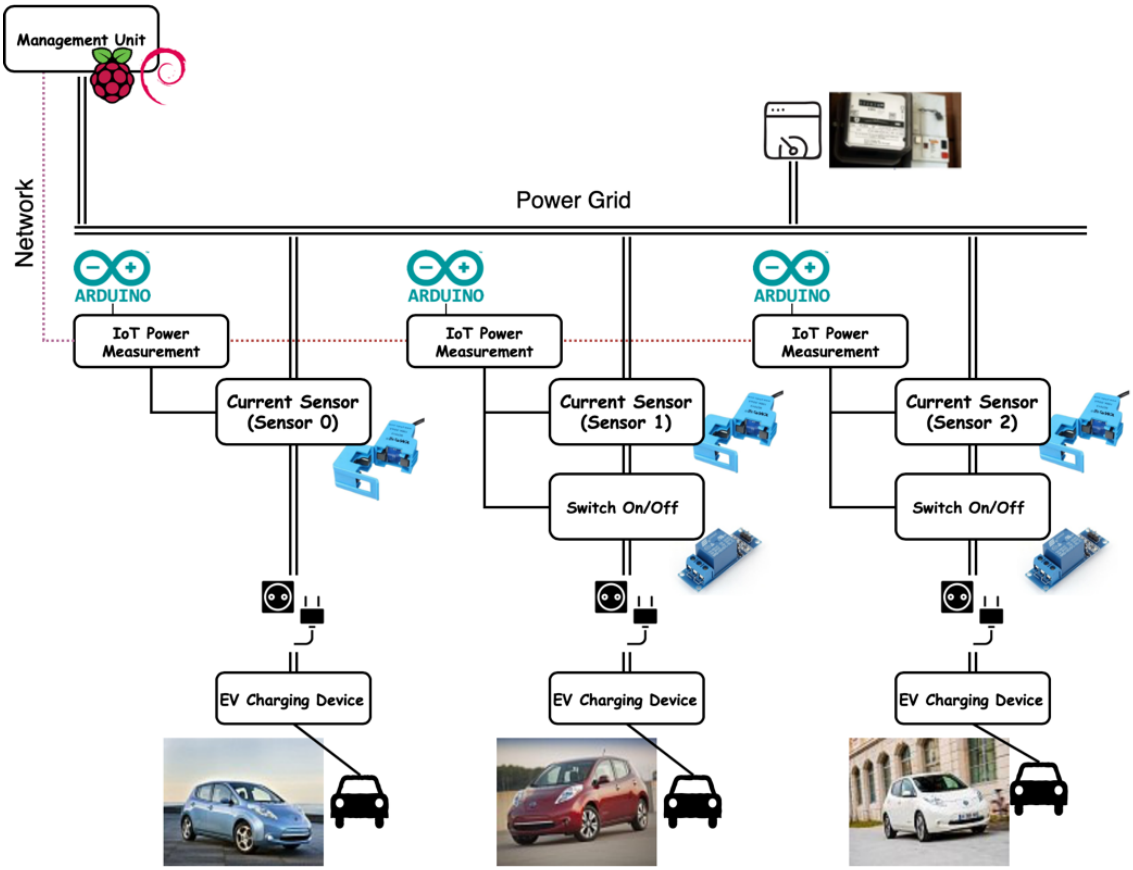
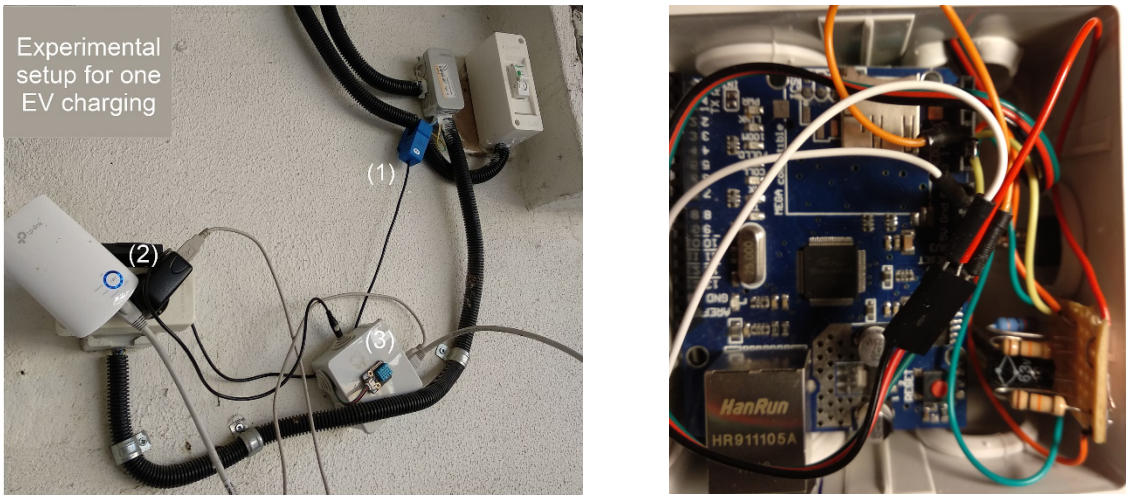


Figure 35.Setup diagram of the case study.

Figure 36 shows photos of one of the developed prototypes. Figure 36a shows a photo of one of the IoT unit prototypes installed (Label (3) in Figure 36a) of the test environment, measuring the current with the non-intrusive SCT-013-000-100A sensor (Label (1) in Figure 36a). In this case, due to the weak Wi-Fi signal at the install location and the absence of other network infrastructure, the sensor unit was connected, using the RJ45 Ethernet interface, to a Wi-Fi Range Extender (Label (2) in Figure 36a) to amplify the signal, allowing the IoT unit to

reach the Management Unit accessible from the network where the Wi-Fi Range Extender was connected to. Figure 36b shows the contents of the IoT unit installed in Figure 36a (Label (3)).



(a) Installed IoT unit *(b) IoT unit contents*

Figure 36.Developed prototypes.

.

Chapter 5 – Results at a Condo

During the Testing and Evaluation Phase, the developed system was deployed to shared place in a condominium, where three EV owners shared the condominium electric installation available at parking places for 3.5 months, Table 3 summarises the data collected during the case study.

Table 3.Data collected during the case study.

Measure	Value
Data Samples	450,000 [1]
Total Time (hours)	2700
Start Date	20 January 2019
End Date	12 May 2019
Charging Data Samples	63,000
Charging Events	300
Total Charging Time (hours)	1060 h (~40%)
Unused Charging Time (hours)	1640 h (~60%)
Total Energy (kWh)	2450 kWh [2]

[1] Estimation, based on the configuration, as "empty" data samples are discarded.
[2] For the current case study, it was assumed a voltage of 230 V.

Figure 37 shows the charging time as well as the average charging power for each charging event (for events with >3 h of charging duration). It is possible to identify an average value of 2.3 kW, approximately (assuming a root mean square, RMS, a voltage value of 230 V).

The absence of a strong correlation between the charging time and the average charging power is also observable (the correlation coefficient between the charging duration and the charging power dataset is −0.30), which allows to conclude that the average charged power by hour load is limited by the charging device and independent of the amount of energy required to charge the EV (e.g., a charging event of 6 hours has a similar average charging power as a charging event with 3 hours).

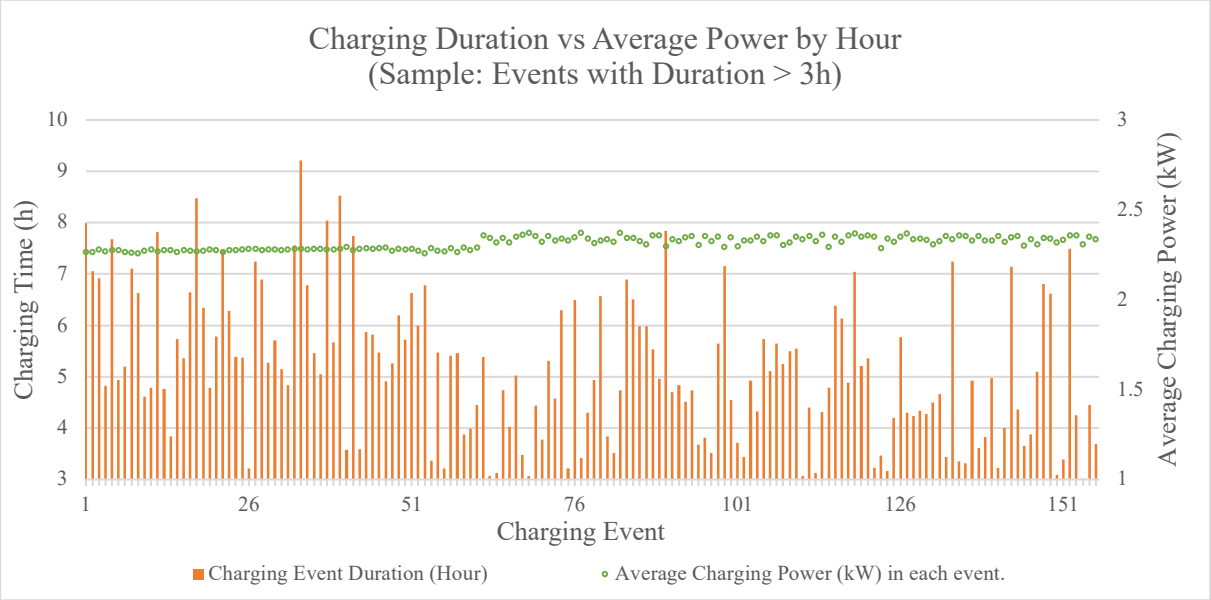


Figure 37.Average charging power and charging duration during each charging event.

Figure 38 displays simultaneous charging events for the entire period analysed (300 charging events on 20 January and 12 May). Due to the power limitations, only two EVs are allowed to be charging at the same time, and the power is delivered on a first-come, first-served (FCFS) basis, where the platform controls the maximum number of stations that are allowed to charge the EVs simultaneously, queueing the remaining charging requests until a charging slot is available.

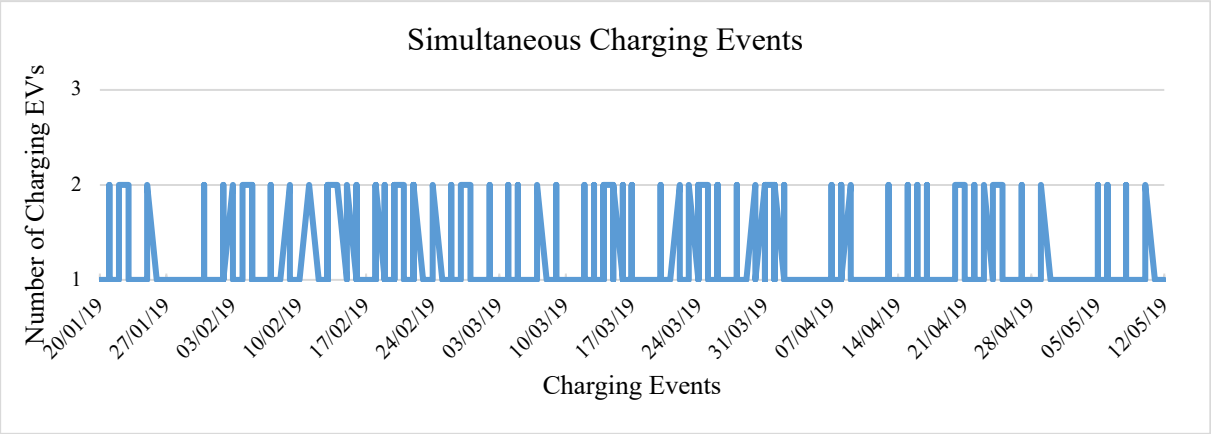


Figure 38.Simultaneous charging (during the test period).

Since the charging platform measures the supplied power continuously, it detects when the EV is fully charged. At that time, it interrupts the EV charging process, registers (closes) the charging transaction, accounts the total amount of energy delivered, and starts supplying energy to the next EV queued.

Figure 39 shows the charging sessions of a Leaf with 24 kWh battery capacity, in a 3.5 month period, where it is possible to verify charging session duration ranging from 1 to 9 h (with an arithmetic average of 5.12 h and standard deviation of 2.03 h), and Figure 40 shows the charged energy, which varies between 2 kWh and 22 kWh (with an arithmetic average of 11.67 kWh and standard deviation of 4.58 kWh).

It is possible to observe on both figures that, on average, this driver only charges 50% of the total charge and uses, on average, 5.5 hours to charge the EV. From the metrics gathered it is possible to identify driver profiles and use this for future charging processes accounting for the power limitation as is suggested in (J. Ferreira & Martins, 2018; Joao Carlos Ferreira et al., 2011).

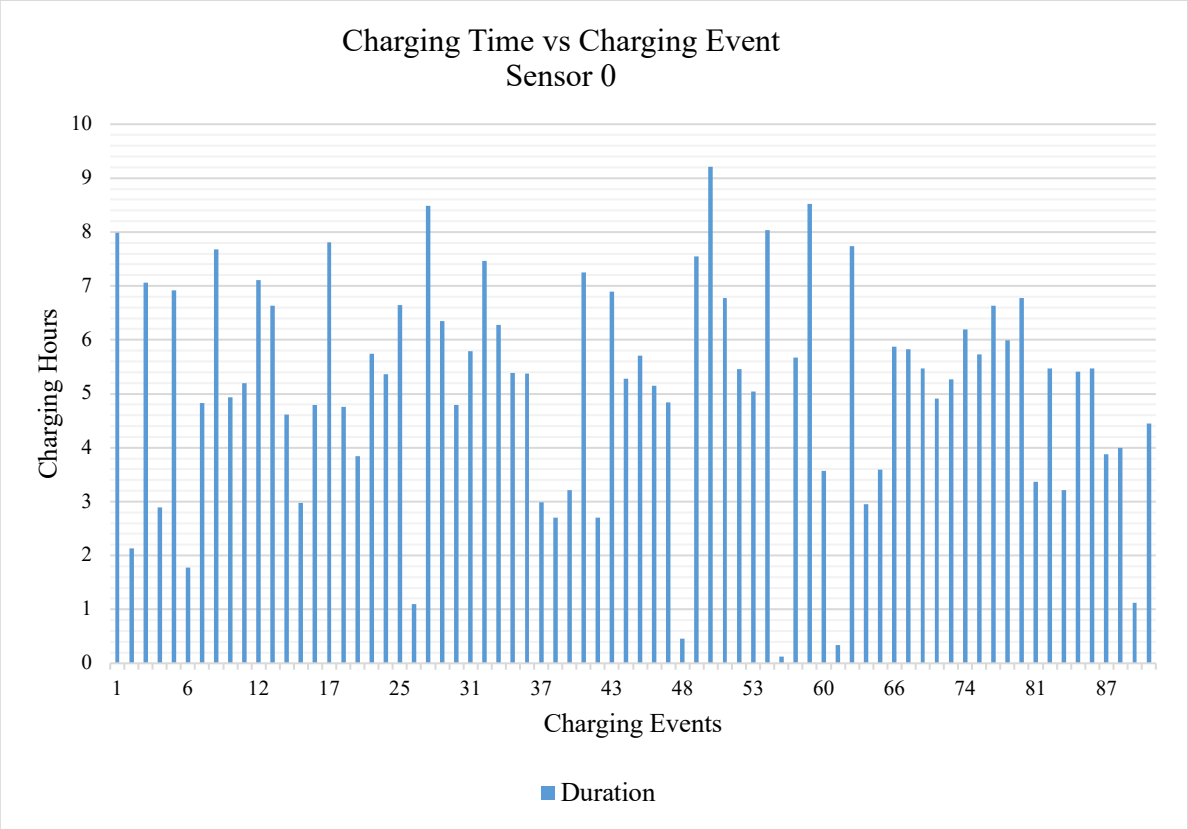


Figure 39. Charging hours (Y-axis) per charging session event in 3.5 months for sensor 0, used to charge a Leaf with 24 kWh battery capacity.

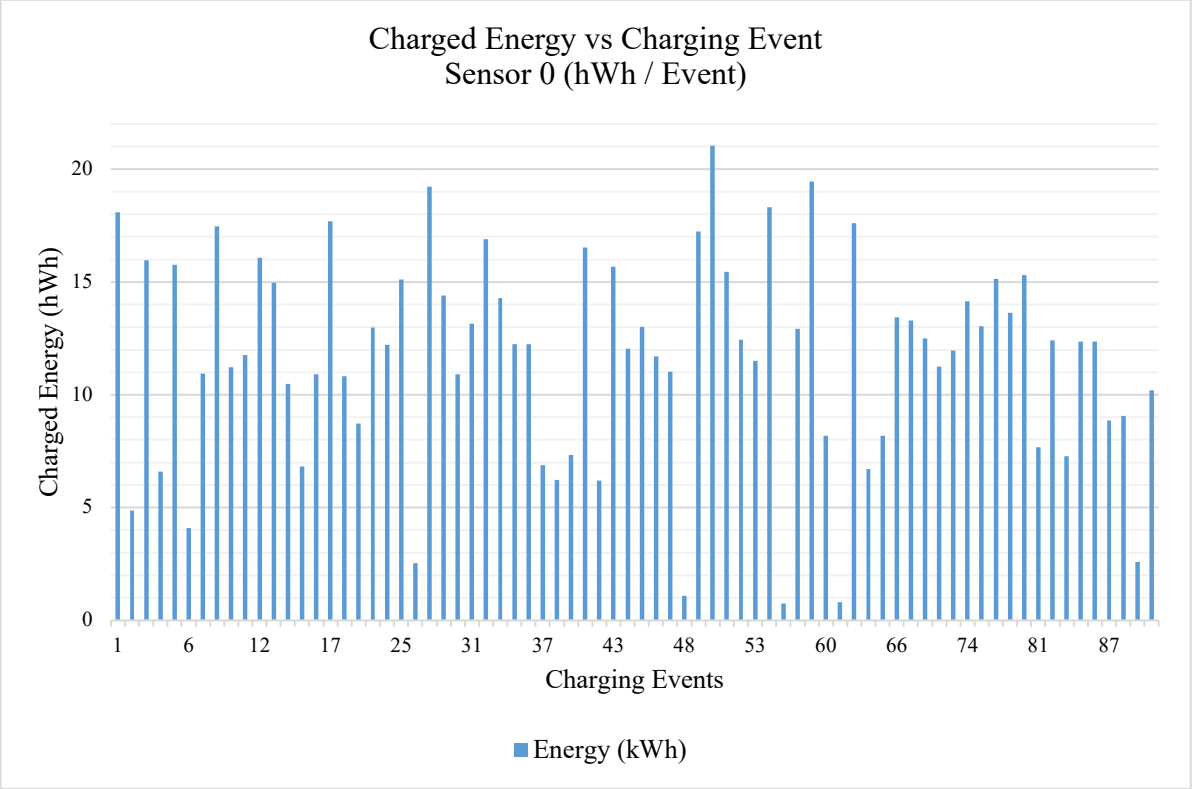


Figure 40.Energy (kWh) per each EV charging session in a Leaf with 24 kWh battery capacity.

Figure 41 shows the charging process with three EVs at the condominium, where it is possible to identify that, due to the power limitation, EV_2 had to wait for an available charging window.

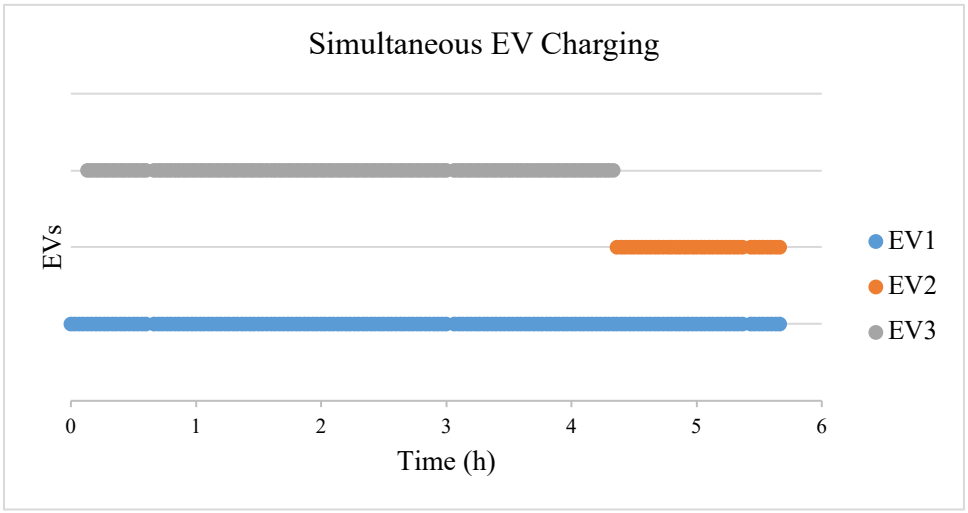


Figure 41. Charging windows (power limitation allows only two EVs to charge simultaneously).

Figure 42 presents the distributions for the charging time (left) and the charged energy (right) for each charging event. It can be observed that for 89% of the charging events ((117 +

82 + 69)/300), the EV will be charging for six hours or less. A similar analysis can be made for the charging energy, where for 92% ((108 + 93 + 76)/300) of the charging events, the EV will charge 15 kWh, which represents roughly 62.5% of the total battery capacity.

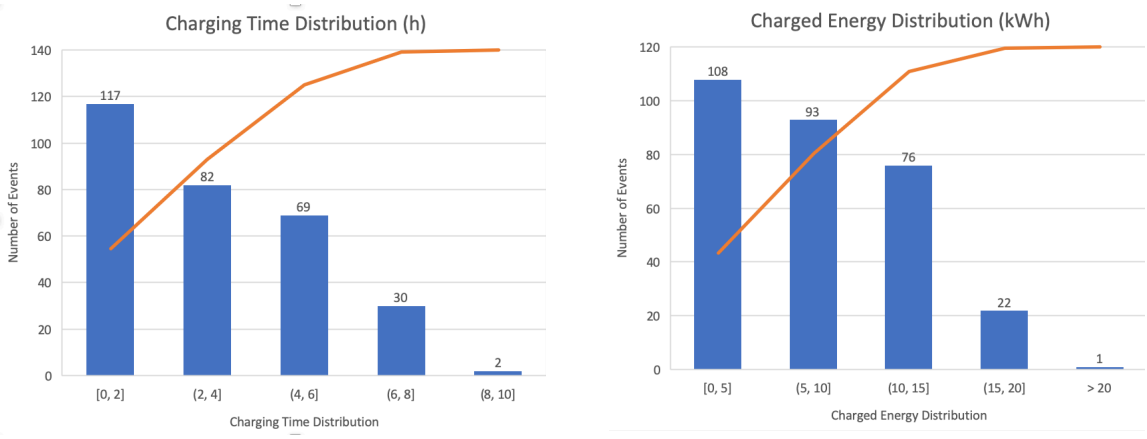


Figure 42. Charging time (left) and energy (right).

Several usages pattern also were observed. Figure 44 presents the distribution of the amount of time between each EV charging event, which shows that for 64% of the times the driver charges the EV with less than 20 h between charging events, which may be correlated with the commute journey.

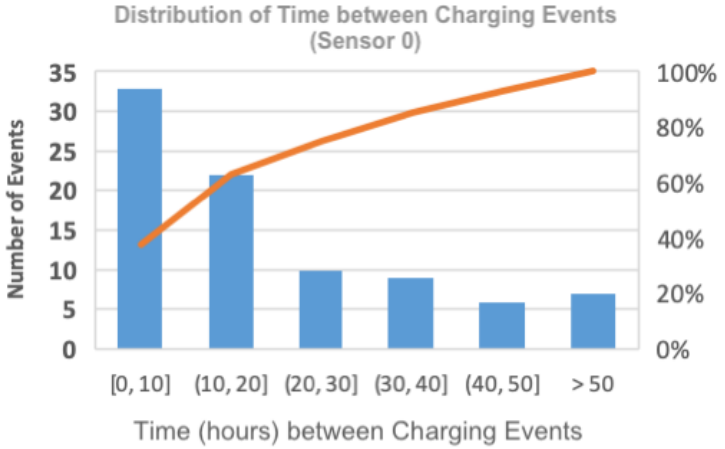


Figure 43.Distribution of time between charging events.

Chapter 6 – Conceptual Model for EV Charging Stations Using IoT, Cloud, and Blockchain

Whereas the current solution was designed to be deployed in a condominium, at a local level, with a limited number of charging stations, in this section the presented work is complemented with a conceptual model to scale up the platform targeting the deployment in more extensive geographical locations. Complementarily, although not strictly required, aiming to support a *user registration free* approach the model is enriched with the use of blockchain cryptocurrencies networks.

6.1.Scaling Up

The scale-up of the model is achieved by increasing the computing power of the Management Unit and by making the unit accessible through a network to all the IoT units deployed. Figure 44 presents a scale-up model which exploits cloud paradigms and, without loss of generality, instantiates the model components in existing cloud platforms. The Mobile App is deployed on the Google Play Store and Apple's App Store, the Management Unit, is packaged in a Docker container (Ismail et al., 2015), deployed on the AWS (Amazon Web Services) cloud computing platform and, the Ethereum open blockchain network used to support the financial transactions originated by the EV charging operation.

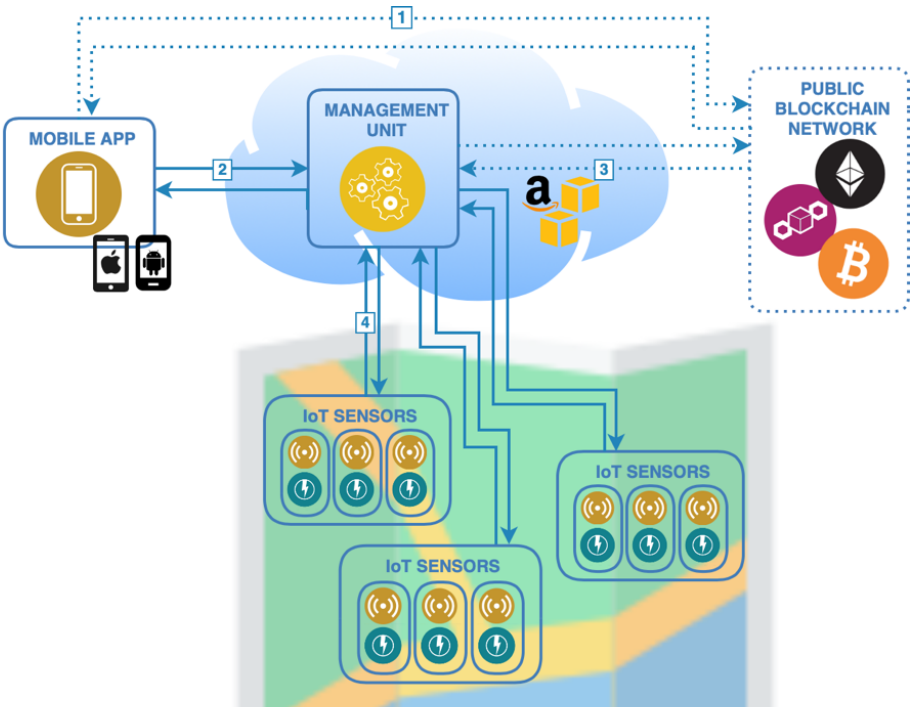


Figure 44.Overview of an IoT/cloud model solution to handle the EV.

While benefiting from the cloud paradigms to simplify the deployment of the platform, the approach also enables the exploitation of the cloud paradigms, such as Platform as a Service (PaaS) or Software as a Service (SaaS).

Figure 44 is also presented the information exchanged between the intervening systems to initiate a charging process:

1. Using the internet connection, the payment is sent from the mobile device to the open blockchain network (Ethereum);

2. Information related to the operation is exchanged between the mobile device and the Management Unit hosted on the AWS;

3. Payment is received from the blockchain network, triggering the charging process on the Management Unit;

4. The EV charging process is enabled on the IoT device (installed on the parking facilities), and the information related to the energy being delivered is sent back to the Management Unit on the AWS.

6.2.Blockchain / Cryptocurrency Integration

As a purposed extension to the conceptual model, the blockchain network is used exclusively to manage the financial transactions between the EV owners and the EV charging station owners. More complex scenarios, like the use of smart contracts (Wang et al., 2019) to establish a *strong* bond between the EV owner (consumer) and the EV charging station owner (provider) or the use of a blockchain network to support the humans-to-machine authentication or machine-to-machine ('Bubbles of Trust', 2018) authentication could be considered.

The transactions between the users and the platform are based on the exchange of EV charging tokens (self-generated), if using a "public" crypto-currency infrastructure like Ethereum (or Bitcoin), the trades are made using that crypto-currencies (which can be exchanged in the market). In the current model, the transaction costs are based on a fixed energy price or based on pre-defined rules but, in the future, the price can be negotiated dynamically in full implementation of a smart grid (J. Ferreira & Martins, 2018).

Figure 45 presents the interactions with a blockchain network using a private/internal blockchain ledger, whereas Figure 46 shows the interactions that would be held when using an "open" cryptocurrency (e.g., Bitcoins).

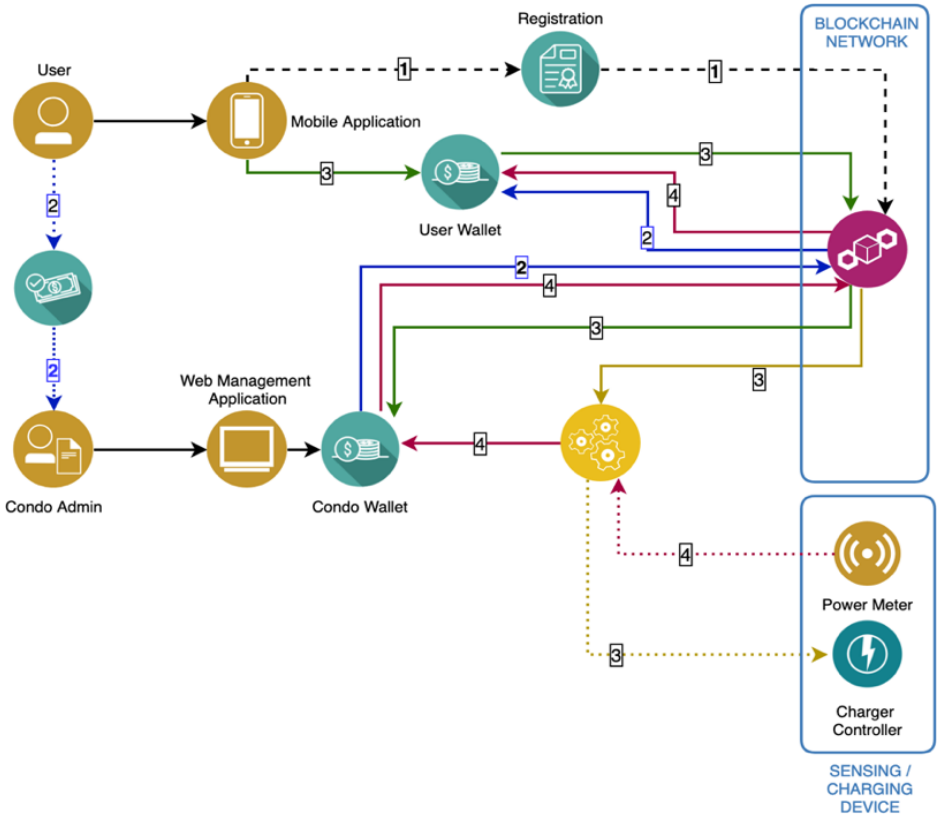


Figure 45.Blockchain interactions with an internal (local) ledger.

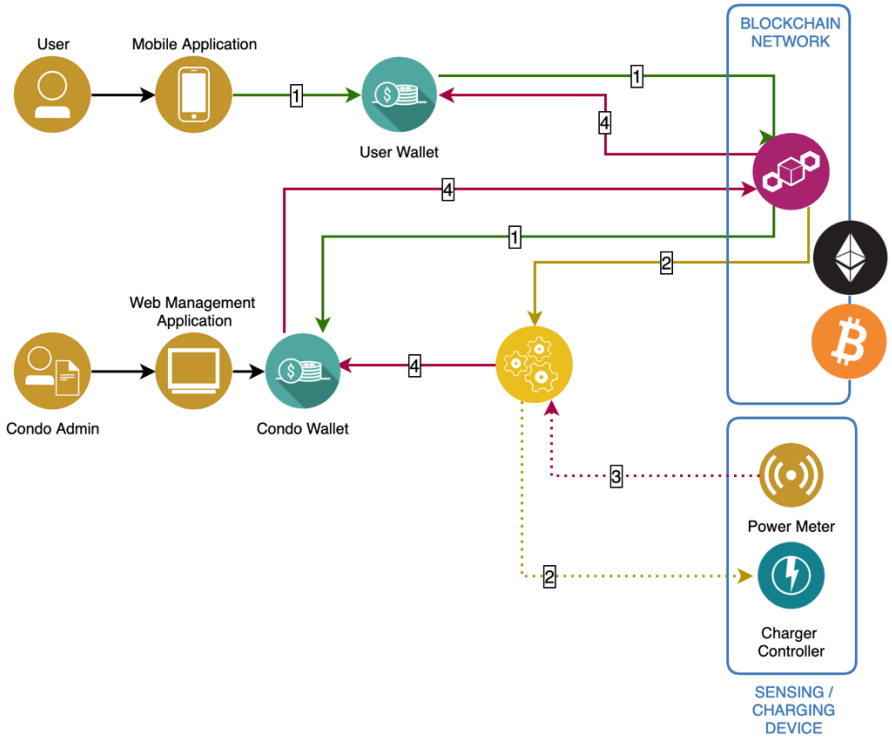


Figure 46.Blockchain interactions using an open cryptocurrency.

Follows the sequence of operations presented in Figure 45 and Figure 46:

1. Using the Mobile App, the user registers/creates his account on the private blockchain network, if using public crypto-currency infrastructure, the user creates his crypto-currency wallet;

2. Using money, the user buys EV charging tokens from the EV platform management, using the web interface of the Management Unit. The charging tokens are transferred from the platform wallet to the user wallet through the blockchain network (if using a public cryptocurrency, the user buys the cryptocurrency on the open market);

3. Using the Mobile App, the user sends charging tokens/cryptocurrency from his wallet to the EV platform wallet, defining the maximum amount to spend (and eventually the maximum time that the vehicle will be connected to the plug, used to optimise the power distribution). The Management Unit receives the transfer from the network and triggers the power Management Unit to start the charging process, which may not be immediate due to the optimisation of the power distribution between the used chargers;

4. The Management Server receives the power measures from the charger, stopping the charging process when the maximum amount is reached, the maximum charging time is reached, or when the vehicle is removed from the charger station (detected by a reduction of the consumed power). If the charging process is interrupted, the remaining amount is returned by the Management Server to the user wallet using the blockchain network.

Chapter 7 – Conclusions and Further Work

7.1.Conclusions

This work explores an approach based on IoT devices and mobile devices to create a solution for the EV charging process in shared spaces providing user identification, transaction management, and reporting services, which allows sharing the installed/existing power infrastructure between the connected EVs while accounting the amount of energy delivered to each EV. The generated accounting information allows the condominium to distribute the energy costs between each EV owner accordingly the energy delivered to its EV.

Taking that into consideration, it can contribute to the proliferation of EVs as aims to solve one of the existing identified barriers on the charging process at condominiums and rented houses. The information gathered related with the energy consumption (which could also be understood as energy requirements), can be used to identify of EV charging profiles, creating charging patterns to handle power limitations and optimise and increase the use of the shared services without the need for new individual services (owner dedicated installations).

A different facet of this problem is related to the issues faced by EVs owners when going abroad (outside of their local area). Due to the existence of multiple operators owning EV charging stations and the absence of *roaming like agreements*, the journey needs to be previously planned, and charging cards for the foreign operators need to be obtained in advance, which due to the lack of standardisation, results in a process complex to be managed.

The cloud / open cryptocurrency conceptual model proposed in Chapter 6 can be applied to mitigate this issue, allowing an EV to be charged (and pay for the charging) without any previous registration with the EV charging station providers.

The solution demonstrated the robustness of the developed prototypes for an EV charging process in shared spaces in the context of the presented case study at a condominium. During the 3.5 month of operation, there was only one failure of an IoT sensor unit due to a general power failure, and the problem was corrected by only delaying the start of the charging process.

7.2.Limitations

Although no network-related limitations were observed in the current study, while using the standard TCP/IP protocol stack, with either wired (Ethernet) or wireless (Wi-Fi) access mediums, to establish communication between the IoT devices and the Management Unit in a local area network (LAN) context, the used protocols from the physical layer (*ISO/IEC 7498-4:1989*, 1996) to the application layer (*ISO/IEC 7498-4:1989*, 1996), may not be suitable or *best tailored* for a large scale (massive number of IoT devices) or more extensive geographical environments in a vast area network (WAN) environment.

The implementation of a similar system in more extensive geographical environments or other topologies may require the use of other (wireless) communication technologies more suitable for each specific context, for instance, low-power wide-area networks (LPWAN) technologies such as LoRa, Sigfox, NB-IoT or LTE-M and the consequent adaptation of the protocols at the application level.

7.3.Future Work

Several topics rose during the development of current work that could lead to further investigation. This section enumerates (not extensively) some topic/areas that can be considered as future work candidates.

The EV charging patterns collected in a condominium can be used to optimise the charging process. Capturing the charging needs for the set of EVs and combining that information with usage profile could be used to devise a charging schedule that optimizes the use of the available resources while maximizes the user's perceived utility (e.g. if an EV, due to its daily usage pattern only needs a SOC of 30%, as soon that value is reached – is that information provided by a systems on the EV or estimated by the amount of energy already delivered – the charging process for that EV can be stopped to allow other EV to be charged). This optimisation-oriented approach can be taken a step forward and applied in local smart grid management scenarios and contexts like the microgeneration.

Following the limitations identified, the extension and evaluation of the current work to support broader deployments can constitute a relevant subject due to expansion on the EV adoption. The conceptual use of blockchain technology presented in Chapter 6 can also be applied to handle energy transactions in other application scenarios, such as micro-generation without a central supervision control mechanism. Although the use of open public cryptocurrency platforms like Bitcoin or Ethereum, due to high transaction costs, can create some barriers to the acceptance of the model, some solutions to use a parallel/specific cryptocurrency network or optimize number of cryptocurrency transitions and on-chain/off-chain data are proposed in (Erdin et al., 2018) and (Pop et al., 2019), respectively. On (Thakur & Breslin, 2018) it is presented a blockchain EV charging queuing process, the extension of model presented, their implementation and evaluation platform with the ideas presented in (Pop et al., 2019) and (Thakur & Breslin, 2018) can lead to relevant results. A blockchain-based approach, supported by a mobile device, can also be applied for mobile ticketing systems, allowing users to buy public transportation tickets using mobile devices (see (Baia et al., 2013) as an example of an Android cloud-based ticket validation implementation), or pay motorway fees and for other services without any previous registration or enrolment process.

Some commercial approaches are currently being tested; for example, charging stations in the UK will be equipped with NFC payment technology ('Charging solutions for your business.', 2019), the study of the use of NFC payment technologies to support or enable cryptocurrency payments can also have some relevance shortly.

References

A Next-Generation Smart Contract and Decentralized Application Platform. (2019, August 28). Retrieved 29 August 2019, from https://github.com/ethereum/wiki/wiki/White-Paper (Original work published 14 February 2014)

Aitzhan, N. Z., & Svetinovic, D. (2018). Security and Privacy in Decentralized Energy Trading Through Multi-Signatures, Blockchain and Anonymous Messaging Streams. *IEEE Transactions on Dependable and Secure Computing*, *15*(5), 840–852. https://doi.org/10.1109/TDSC.2016.2616861

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Communications Surveys Tutorials*, *17*(4), 2347–2376. https://doi.org/10.1109/COMST.2015.2444095

Axsen, J., Goldberg, S., Bailey, J., Kamiya, G., Langman, B., Cairns, J., … Miele, A. (2015). *Electrifying Vehicles: Insights from the Canadian Plug-in Electric Vehicle Study*. Retrieved from Simon Fraser University website: https://sustainabletransport.ca/the-canadian-plug-in-electric-vehicle-study-cpevs/

Baia, A., Ferreira, J., Filipe, P., & Cunha, G. (2013). Android as a Cloud Ticket Validator. *2013 International Conference on Cloud & Ubiquitous Computing & Emerging Technologies*, 1–7. https://doi.org/10.1109/CUBE.2013.11

Bauer, M., Boussard, M., Bui, N., De Loof, J., Magerkurth, C., Meissner, S., … Walewski, J. W. (2013). IoT Reference Architecture. In A. Bassi, M. Bauer, M. Fiedler, T. Kramp, R. van Kranenburg, S. Lange, & S. Meissner (Eds.), *Enabling Things to Talk* (pp. 163–211). https://doi.org/10.1007/978-3-642-40403-0_8

Bauer, M., Bui, N., De Loof, J., Magerkurth, C., Nettsträter, A., Stefa, J., & Walewski, J. W. (2013). IoT Reference Model. In A. Bassi, M. Bauer, M. Fiedler, T. Kramp, R. van Kranenburg, S. Lange, & S. Meissner (Eds.), *Enabling Things to Talk* (pp. 113–162). https://doi.org/10.1007/978-3-642-40403-0_7

Bo, C., Zhang, L., Li, X.-Y., Huang, Q., & Wang, Y. (2013). SilentSense: Silent user identification via touch and movement behavioral biometrics. *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking - MobiCom '13*, 187. https://doi.org/10.1145/2500423.2504572

Bubbles of Trust: A decentralized blockchain-based authentication system for IoT | Elsevier Enhanced Reader. (2018). https://doi.org/10.1016/j.cose.2018.06.004

Caballero, D. C. (2016, March 8). How to use Non-invasive AC Current Sensors with Arduino. Retrieved 18 September 2019, from Scidle website: https://scidle.com/how-to-use-non-invasive-ac-current-sensors-with-arduino/

Cerf, V. G., & Kahn, R. E. (1974). *A Protocol for Packet Network Intercommunication*. (5), 13.

Charging solutions for your business. (2019). Retrieved 7 September 2019, from Allego—Keep driving forward website: https://www.allego.eu/

Dahlberg, T., Guo, J., & Ondrus, J. (2015). A critical review of mobile payment research. *Electronic Commerce Research and Applications*, *14*(5), 265–284. https://doi.org/10.1016/j.elerap.2015.07.006

De Luca, A., Hang, A., Brudy, F., Lindner, C., & Hussmann, H. (2012). Touch me once and i know it's you!: Implicit authentication based on touch screen patterns. *Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems - CHI '12*, 987. https://doi.org/10.1145/2207676.2208544

Erdin, E., Cebe, M., Akkaya, K., Solak, S., Bulut, E., & Uluagac, S. (2018). Building a Private Bitcoin-Based Payment Network Among Electric Vehicles and Charging Stations. *2018 IEEE International Conference on Internet of Things (IThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 1609–1615. https://doi.org/10.1109/Cybermatics_2018.2018.00269

Feng, T., Liu, Z., Kwon, K.-A., Shi, W., Carbunar, B., Jiang, Y., & Nguyen, N. (2012). Continuous mobile authentication using touchscreen gestures. *2012 IEEE Conference on Technologies for Homeland Security (HST)*, 451–456. https://doi.org/10.1109/THS.2012.6459891

Ferreira, J., Afonso, J., Monteiro, V., & Afonso, J. (2018). An Energy Management Platform for Public Buildings. *Electronics*, *7*(11), 294. https://doi.org/10.3390/electronics7110294

Ferreira, J. C., Monteiro, V., & Afonso, J. L. (2014). Vehicle-to-Anything Application (V2Anything App) for Electric Vehicles. *IEEE Transactions on Industrial Informatics*, *10*(3), 1927–1937. https://doi.org/10.1109/TII.2013.2291321

Ferreira, J., & Martins, A. (2018). Building a Community of Users for Open Market Energy. *Energies*, *11*(9), 2330. https://doi.org/10.3390/en11092330

Ferreira, Joao C., Monteiro, V., Afonso, J. L., & Silva, A. (2011). Smart electric vehicle charging system. *2011 IEEE Intelligent Vehicles Symposium (IV)*, 758–763. https://doi.org/10.1109/IVS.2011.5940579

Ferreira, Joao Carlos, da Silva, A. R., & Afonso, J. L. (2011). EV-Cockpit – Mobile Personal Travel Assistance for Electric Vehicles. In G. Meyer & J. Valldorf (Eds.), *Advanced Microsystems for Automotive Applications 2011* (pp. 247–257). https://doi.org/10.1007/978-3-642-21381-6_24

Fielding, R., & Reschke, J. (2014). Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content. Retrieved 30 August 2019, from https://tools.ietf.org/html/rfc7231

Fowler, M. (2004). *UML distilled: A brief guide to the standard object modeling language* (3rd ed). Boston: Addison-Wesley.

Frank, M., Biedert, R., Ma, E., Martinovic, I., & Song, D. (2013). Touchalytics: On the Applicability of Touchscreen Input as a Behavioral Biometric for Continuous Authentication. *IEEE Transactions on Information Forensics and Security*, *8*(1), 136–148. https://doi.org/10.1109/TIFS.2012.2225048

Garg, S., Kaur, K., Kaddoum, G., Gagnon, F., & Rodrigues, J. J. P. C. (2019). An Efficient Blockchain-based Hierarchical Authentication Mechanism for Energy Trading in V2G Environment. *ArXiv:1904.01171 [Cs]*. Retrieved from http://arxiv.org/abs/1904.01171

Higgins, S. (2016, March 7). Why a German Power Company is Using Ethereum to Test Blockchain Car Charging. Retrieved 29 August 2019, from CoinDesk website: https://www.coindesk.com/german-utility-company-turns-to-blockchain-amid-shifting-energy-landscape

Ismail, B. I., Goortani, E. M., Karim, M. B. A., Tat, W. M., Setapa, S., Luke, J. Y., & Hoe, O. H. (2015). Evaluation of Docker as Edge computing platform. *2015 IEEE Conference on Open Systems (ICOS)*, 130–135. https://doi.org/10.1109/ICOS.2015.7377291

ISO/IEC 7498-4:1989—Information technology -- Open Systems Interconnection -- Basic Reference Model: Naming and addressing. (1996, June 15). Retrieved from https://standards.iso.org/ittf/PubliclyAvailableStandards/s014258_ISO_IEC_7498-4_1989(E).zip

Jakobsson, M., Shi, E., Golle, P., & Chow, R. (2009). *Implicit Authentication for Mobile Devices*. 6.

Kang, J., Yu, R., Huang, X., Maharjan, S., Zhang, Y., & Hossain, E. (2017). Enabling Localized Peer-to-Peer Electricity Trading Among Plug-in Hybrid Electric Vehicles Using Consortium Blockchains. *IEEE Transactions on Industrial Informatics*, *13*(6), 3154–3164. https://doi.org/10.1109/TII.2017.2709784

Learn | OpenEnergyMonitor. (2018). Retrieved 18 September 2019, from https://learn.openenergymonitor.org/electricity-monitoring/ct-sensors/how-to-build-an-arduino-energy-monitor-measuring-current-only?redirected=true

Liu, C., Chai, K. K., Zhang, X., Lau, E. T., & Chen, Y. (2018). Adaptive Blockchain-Based Electric Vehicle Participation Scheme in Smart Grid Platform. *IEEE Access*, *6*, 25657–25665. https://doi.org/10.1109/ACCESS.2018.2835309

Liu, D., Li, D., Liu, X., Ma, L., Yu, H., & Zhang, H. (2018). Research on a Cross-Domain Authentication Scheme Based on Consortium Blockchain in V2G Networks of Smart Grid. *2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2)*, 1–5. https://doi.org/10.1109/EI2.2018.8582227

Martin, R. C., & Martin, R. C. (2018). *Clean architecture: A craftsman's guide to software structure and design*. London, England: Prentice Hall.

Mengelkamp, E., Notheisen, B., Beer, C., Dauer, D., & Weinhardt, C. (2018). A blockchain-based smart grid: Towards sustainable local energy markets. *Computer Science - Research and Development*, *33*(1–2), 207–214. https://doi.org/10.1007/s00450-017-0360-9

Mills, D., Delaware, U., J. Martin, Ed., & Burbank, J. (2010). Network Time Protocol Version 4. Retrieved 15 September 2019, from https://www.ietf.org/rfc/rfc5905.txt

Munsing, E., Mather, J., & Moura, S. (2017). Blockchains for decentralized optimization of energy resources in microgrid networks. *2017 IEEE Conference on Control Technology and Applications (CCTA)*, 2164–2171. https://doi.org/10.1109/CCTA.2017.8062773

Nakamoto, S. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System*. 9.

Panarello, A., Tapas, N., Merlino, G., Longo, F., & Puliafito, A. (2018). Blockchain and IoT Integration: A Systematic Survey. *Sensors*, *18*(8), 2575. https://doi.org/10.3390/s18082575

Patel, V. M., Chellappa, R., Chandra, D., & Barbello, B. (2016). Continuous User Authentication on Mobile Devices: Recent progress and remaining challenges. *IEEE Signal Processing Magazine*, *33*(4), 49–61. https://doi.org/10.1109/MSP.2016.2555335

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, *24*(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

Pop, C., Antal, M., Cioara, T., Anghel, I., Sera, D., Salomie, I., … Bertoncini, M. (2019). Blockchain-Based Scalable and Tamper-Evident Solution for Registering Energy Data. *Sensors*, *19*(14), 3033. https://doi.org/10.3390/s19143033

Postel, J. (1980, August 28). User Datagram Protocol. Retrieved 15 September 2019, from https://tools.ietf.org/html/rfc768

Presentation Model. (2004). Retrieved 30 August 2019, from Martinfowler.com website: https://martinfowler.com/eaaDev/PresentationModel.html

Pustisek, M., Kos, A., & Sedlar, U. (2016). Blockchain Based Autonomous Selection of Electric Vehicle Charging Station. *2016 International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI)*, 217–222. https://doi.org/10.1109/IIKI.2016.60

Putting electric vehicles on the map: A policy agenda for residential charging infrastructure in Canada. (2019). https://doi.org/10.1016/j.erss.2018.11.009

Rescorla, E. (2000). HTTP Over TLS. Retrieved 30 August 2019, from https://tools.ietf.org/html/rfc2818

RWE and Slock.it – Electric cars using Ethereum wallets can recharge by induction at traffic lights. (2016, February 22). Retrieved 29 August 2019, from International Business Times UK website: https://www.ibtimes.co.uk/rwe-slock-it-electric-cars-using-ethereum-wallets-can-recharge-by-induction-traffic-lights-1545220

Sae-Bae, N., Ahmed, K., Isbister, K., & Memon, N. (2012). Biometric-rich gestures: A novel approach to authentication on multi-touch devices. *Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems - CHI '12*, 977. https://doi.org/10.1145/2207676.2208543

Sanseverino, E. R., Silvestre, M. L. D., Gallo, P., Zizzo, G., & Ippolito, M. (2017). The Blockchain in Microgrids for Transacting Energy and Attributing Losses. *2017 IEEE International Conference on Internet of Things (IThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 925–930. https://doi.org/10.1109/iThings-GreenCom-CPSCom-SmartData.2017.142

SCT-013-030 Energy Meter. (2016, January 3). Retrieved 18 September 2019, from MySensors Forum website: https://forum.mysensors.org/topic/2700/sct-013-030-energy-meter

Setting up a Raspberry Pi as a Wireless Access Point—Raspberry Pi Documentation. (2019). Retrieved 22 September 2019, from https://www.raspberrypi.org/documentation/configuration/wireless/access-point.md

Simon, H. A. (2008). *The sciences of the artificial* (3. ed., [Nachdr.]). Cambridge, Mass.: MIT Press.

Stat of the Week: Percent of Households That Rent By Country. (2018, March 24). Retrieved 21 May 2019, from https://evadoption.com/stat-of-the-week-percent-of-households-that-rent-by-country/

Thakur, S., & Breslin, J. G. (2018). Electric Vehicle Charging Queue Management with Blockchain. In A. M. J. Skulimowski, Z. Sheng, S. Khemiri-Kallel, C. Cérin, & C.-H. Hsu (Eds.), *Internet of Vehicles. Technologies and Services Towards Smart City* (Vol. 11253, pp. 249–264). https://doi.org/10.1007/978-3-030-05081-8_18

Wang, X., Yang, W., Noor, S., Chen, C., Guo, M., & van Dam, K. H. (2019). Blockchain-based smart contract for energy demand management. *Energy Procedia*, *158*, 2719–2724. https://doi.org/10.1016/j.egypro.2019.02.028