Pan African University
Institute of Water
and Energy Sciences

PAN-AFRICAN UNIVERSITY

INSTITUTE FOR WATER AND ENERGY SCIENCES

(including CLIMATE CHANGE)

Master Dissertation

Submitted in partial fulfillment of the requirements for the Master degree in

[ENERGY POLICY]

Presented by

Mohammed BENTAALLAH

TITLE: Development of mobile sensing applications solutions towards Internet of Things.

Defended on 30/09/2019 Before the Following Committee:

Chair	Zineb BEDRANE	Professor	Affiliation (Institution only)
Supervisor	Roberto MONSORNO	Doctor	Affiliation (Institution only)
External Examiner	Azeddine CHEIKH	Professor	Affiliation (Institution only)
Internal Examiner	Nabila LABRAOUI	Doctor	Affiliation (Institution only)

This research thesis is submitted in fulfillment of the requirements of the Master of Science in Energy Engineering at the Pan African University Institute of Water and Energy Sciences (including Climate Change)-(PAUWES) at the University of Tlemcen in Algeria.

September 2019

DECLARATION

I, **Mohammed BENTAALLAH** do hereby declare that this thesis is my original work and to the best of my knowledge, it has not been submitted for any award in any University or Institution.

Signed______Mohamed__________________________ Date _29-09-2019_

Mohammed BENTAALLAH

CERTIFICATION

I, **Roberto MONSORNO** hereby certify that this thesis has been submitted with my approval as the supervisor

Signed____ Date _**-09-2019_

Dr. Roberto MONSORNO

Abstract

Human beings in communication with the surrounding environment in numerous ways. They perceive the environmental conditions and act, react, or adjust accordingly. If the environment can be made to reciprocate this behavior and respond to human behavior, it will lead to several advantages. Such behavior can automate various tasks that humans have to perform manually and also provide innovative services and facilities using the smart home as a homelike environment that possesses ambient intelligence and automatic control, which allow it to respond to the behavior of residents and provide them with various facilities.

In a future Internet of Things, interconnected systems allow intelligent monitoring and control of smart houses. For example, energy-saving applications control indoor climate and electricity usage by employing context information to switch off appliances (e.g., lights, computers), reduce room temperature, close windows or stop warm water circulation. The primary purpose of this dissertation is to build a basic phone application for home automation using different application program interface for actuation and data acquisition via HTTP communication protocol. To interact with an actuator, after sensing the environmental temperature, humidity…etc. In the first chapter the work presents state of the art for IoT. In the second we have looked into the home automation insight. In the third the design and development work will be presented, considering three projects (Ju-Jami, European Research program FP7 – SINFONIA, and H2020 HEART). Chapter four will look into the outlook of future development of the smart house application.

Keywords: IoT, Android, smart house, Application Programming Interface (API)

RÉSUMÉ

Les humains interagissent avec l'environnement qui les entoure de nombreuses façons. Ils perçoivent les conditions environnementales et agissent, réagissent ou s'adaptent en conséquence. Si l'environnement peut être amené à rendre ce comportement réciproque et à réagir au comportement humain, cela apportera plusieurs avantages. Un tel comportement peut automatiser diverses tâches que les humains doivent exécuter manuellement et fournir de nouveaux services et installations. Utiliser la maison intelligente comme un environnement domestique doté d'une intelligence ambiante et d'un contrôle automatique, ce qui lui permet de réagir au comportement des résidents et de leur fournir diverses installations.

Les appareils interconnectés permettent de surveiller et de contrôler intelligemment les maisons intelligentes dans un futur Internet des objets. Les applications d'économie d'énergie, par exemple, contrôlent la température ambiante et l'électricité en utilisant des informations de contexte pour éteindre les appareils (lampes, ordinateurs, par exemple), réduire la température ambiante, fermer les fenêtres ou arrêter la circulation de l'eau chaude.

L'objectif principal de cette thèse est de construire une application téléphonique de base pour la domotique utilisant une interface de programme d'application différente pour les actionnements et l'acquisition de données via un protocole de communication HTTP. Pour interagir avec un actionneur, après avoir détecté la température ambiante, l'humidité, etc. Dans le premier chapitre, l'ouvrage présente l'état de la technique pour l'IoT. Dans la seconde partie, nous avons examiné la perspicacité de la domotique. Dans la troisième partie, les travaux de conception et de développement seront présentés, en tenant compte de 3 projets (Ju-Jami, programme de recherche européen FP7 - SINFONIA et H2020 HEART). Le chapitre quatre examinera les perspectives de développement futur de l'application maison intelligente.

Mots-clés: IoT, Android, Maison intelligente, Interface de programmation d'applications (API),

Acknowledgments

First and foremost, I would like to thank God Almighty for giving me the strength, knowledge, ability, and opportunity to undertake this research study and to persevere and complete it satisfactorily. Without his blessings, this achievement would not have been possible.

This master dissertation behind, would not have been possible without the exceptional support of my supervisor, Dr. Roberto MONSORNO and co-supervisor, Simone TRITINI.

Providing sincere support and guidance at all times and given me invaluable guidance, inspiration, and suggestions in my quest for knowledge. They have given me all the freedom to pursue my research, while silently and non-obtrusively ensuring that I stay on course and do not deviate from the core of my research. Without their able guidance, this thesis would not have been possible, and I shall eternally be grateful to them for their assistance.

I have great pleasure in acknowledging my gratitude to my colleagues whom I have had the pleasure to work with during my internship in EURAC research, particularly; Dr. Roberto Monsorno, Simone Tritini ,Racheline Bonadio ,Matteo Amatori, Andrea Vianello, Armin Costa, Dr. Abraham Mejia Aguilar, Dr. Stefano Tondini, from the Center of Sensing Solution ensuring that the fire keeps burning and being there at times when I required motivation and propelling me on the course of this thesis. Their support, encouragement, and credible ideas have been great contributors in the completion of the thesis.

I would especially like to thank PAUWES administration and especially Dr. Eric Tambo, my host institute supervisor, which helped much in the process of getting the internship as well as getting to the internship place.

Nothing is as beautiful to offer as the fruit of labour dedicated to heart to those we love and thank you for expressing gratitude throughout our existence.This work is also dedicated to the people most dear to my heart, my father and my mother because it is thanks to their support that I was able to arrive at this stage; I hope that this work is at the height of everything that they could do for me.

Thank you, La ringrazio molto.

Arrivederci,

Content

Table of Figures:

Acronyms

IoT	Internet of Things
API	Application program interface
HTTP	HyperText Transfer Protocol
M2M	Machine to machine
OS	Operating System
IP	Internet Protocol
WPAN	wireless personal area network
HVAC	heating, ventilation, and air conditioning
RFID	Radio-frequency identification
IT	Information Technology
ICT	Information and Communication Technologie
FTP	File Transfer Protocol
DC	direct current
AC	alternating current
REST	Representational state transfer
XML	Extensible Markup Language
JSON	JavaScript Object Notation
BIPV	Building-integrated photovoltaics
IOS	Apple operating system
IDE	Integrated development environment
HTML	Hypertext Markup Language
UI	User Interface
SQL	Structured Query Language

Introduction

With the rapid signs of progress in Internet technologies, a new trend in the era of ubiquity is being realized. The enormous increase in users of Internet and modifications on the internetworking technologies enable networking of everyday objects. (D. Surie, 2008). “Internet of Things (IoT)” is all about physical items talking to each other, machine-to-machine communications and person-to-computer communications will be extended to “things” (Internet 3.0: The Internet of Things, Analysys Mason, 2010). Key technologies that will drive the future IoT will be related to smart sensor technologies.

The projected influence of machine devices to communicate autonomously on the horizon of the Internet of Things is highlighting more and more opportunities for machine-to-machine communication in recent years. Most of the first IoT systems are offering to come to a market which follows a vertical model. That is not surprising, given that converting from the machine-to-machine background.

However, there is little infrastructure available to help horizontal company. However, that is changing. (Lal Das P. , Beisswenger, Mangalam, Yuce, & Lukac, 2017) Following the merge of the two concepts, the horizontal platform can be quickly deploying IoT service. Capabilities with all other aspects of claims such as device activation, device monitoring, or device localization, etc… What has been relatively less explored is how governments and businesses can collaborate to mutually reap the potential benefits of IoT while grappling with the numerous challenges that new technologies inevitably pose. Governments are keen to learn how IoT may make their economies more competitive or make it easier to manage businesses within their jurisdiction.

The main objective for this study is to develop a cloud-based application in the concept of IoT, that is a dashboard for managing and controlling devices and appliances inside a smart house connected to IoT. This dashboard should also obtain sensor data and display it to the users. Starting from a web application, we will develop a smartphone application for smart house automation system, and the application will be connected to the Internet.

The most common so far is native development. When developing a native application, this means that the application will only be compatible with the target platform chosen beforehand, which is Android Studio. The development will be in the Java language. A native development will have the advantage of facilitating access to the own functionality provided by the OS and provide an interface containing the components that users are used to. In the first chapter we will look into the state of art of Machine 2 machine and internet of things than we will see also in the second chapter how home automation was developed during the past decades and more about the different project taken in consideration for the development of the phone application. The third chapter goes through all the steps and programming codes used to develop the application and the final chapter highlight the next steps and future outlook of the phone application.

Chapter 1: M2M to IoT

Contents

1.1.Introduction:

Currently, given the heterogeneous nature of connected objects, communication technologies are emerging in the machine-to-machine (M2M) and Internet of Things (IoT) age (Qusay , Sajjad , & Atta , 2018). However, human understanding and usage of, interaction and experience with "intelligent items" and the systems are creating did not develop at the same speed, and this creates challenges with enormous technical, societal, economic, and political consequences. Leaving a wide range of researchers from academia and industry, as well as businesses, government agencies, and cities to explore the technologies comprising the Internet of Things from three main perspectives: scientific theory, engineering design, and the user experience. Among the techniques underlying the IoT, two main areas stand out:

- Communications to ensure connectivity, through the Internet, with connected objects. Over the past fifteen years, we have seen the development of local networks (Blue-tooth, ZigBee, Wi-Fi) that are now adapting to IoT by reducing their energy consumption. However, the most significant event of the last two years has been the rapid emergence of long-distance and low-energy networks (LPWANs), of which two "seeds," LoRaWAN and Sigfox, are of French origin. Cellular networks, in turn, are entering the market with profiles based on the 4th generation of mobile networks (eMTC and NB-IoT) in anticipation of 5G, which could bring decisive progress.
- Data processing and storage, which is increasingly done in cloud computing but must respect a principle of subsidiarity, data becomes the company's main asset.

1.2.M2M evolution to the IoT:

Machine to machine (M2M) and the Internet of Things (IoT) are new thriving technologies that are positively affecting our lives. The M2M connection enables direct communication that combines the information technology and the autonomous interaction of many devices and machines to perform sensing, processing, and actuation. (Oluwatosin & Mohamed , 2018) IoT paradigm defined as a dynamic global network depends on the identification and the use of many heterogeneous physical and virtual objects, (Dr. Ovidiu Vermesan, 2011) which are connected to the internet.

1.2.1. Machine to Machine:

In general, when someone says M2M communication, they often are referring to cellular communication for embedded devices. Examples of M2M communication, in this case, would be vending machines sending out inventory information or ATM machines getting authorization to dispense cash. Traditionally, M2M focused on “industrial telematics,” which is a fancy way of explaining data transfer for some commercial benefit. However, many original uses of M2M still stand today, like smart meters. Wireless M2M has been dominated by cellular since it came out in the mid-2000s with 2G cell networks. Because of this, the cellular market has tried to brand M2M as an inherently cellular thing by offering M2M data plans. However, cellular M2M is only one subsection of the market, and it should not be thought of as a cellular-only area.

M2M (Machine-to-Machine) has come of age. It has been almost a decade since the idea of expanding the scope of entities connected to “the network” (wireless, wireline; private, public) beyond mere humans and their preferred communication gadgets have emerged around the notions of the “Internet of Things” (IoT), the “Internet of Objects” or M2M. The initial vision was that of a myriad of new devices, mostly unnoticed by humans, working together to expand the footprint of end-user services. (David Boswarthick, 2012)

IoT will create new ways to care for safety or comfort, optimizing a variety of goods delivery mechanisms, enabling individuals or cars to be tracked efficiently while creating new systems, and producing new value.

As with every vision, it has taken time to materialize. Early efforts concentrated on refining the initial vision by testing new business models, developing point solutions to test feasibility, and also forecasting the impact of insufficient interoperability. Over the past few years, the realization that there are new viable sources of demand that can be met and monetized has created the push for a joint effort by the industry. This will turn a patchwork of standalone elements and solutions into a coherent “system of systems”, Gradually shifting the focus from ' what ' to ' how ' and creating suitable techniques and norms.

1.2.2. M2M communication:

M2M implies different solutions which help the communication between things of the same type and for a specific application, all through wired or wireless communication networks. Solutions are allowing for end users to capture data about events from resources, such as temperature or inventory levels. Typically, M2M is deployed to achieve productivity gains, reduce costs, and increase safety or security. It has been applied in many different scenarios, including the remote monitoring and control of enterprise assets, or to provide connectivity of remote machine-type devices. Remote monitoring and control have generally provided the incentive for industrial applications, whereas connectivity has been the focus in other enterprise scenarios such as connected vending machines or point-of-sale terminals for online credit card transactions. However, M2M solutions usually do not allow the full sharing of information or the connection straight to the Internet.

1.2.3. Key application areas:

Existing M2M solutions cover numerous industry sectors and application scenarios. Various predictions have been made by analyst firms that provide market information such as critical applications, value chains, and market actors, as well as market sizes (including forecasts) (ABI 2012, Berg 2013). A selected summary of the main cellular M2M application markets is provided in Figure 2.2, and the figures are estimates of deployed numbers of corresponding M2M devices in the years 2012 and 2016, respectively. The largest segment is currently telematics for cars and vehicles. Typical applications include navigation, remote vehicle diagnostics,

1.2.4. Architecture and application in M2M:

Figure 1-1 shows a simple architecture of M2M systems with its components. The various components and elements of an M2M system are briefly described below.

It contains the middleware layer where data goes through various application services and is used by the specific business-processing engines.

M2M applications will be based on the infrastructural assets (e.g., access enablers) that are provided by the operator. Applications may either target at end-users, such as the user of a specific M2M solution, or at other application providers to offer more refined building blocks by which they can build more sophisticated M2M solutions and services. e.g., customer care functionality, elaborate billing functions. Etc... Those services, or service enablers, may be designed and offered by an application provider, but the operator might offer via the operator platform itself.

There are different technical architectural approaches for M2M systems such as, e.g., end-to-end internet approach, and M2M gateway-based approach. (Latvakoski, et al., 2013; Latvakoski, et al., 2013)

It is possible to establish an Internet connection from an Internet node to the M2M asset device without any additional intermediate node, to make the transformation into the messages in the end-to-end Internet-based approach if there is at least tiny Internet protocol (IP) stack also in small embedded devices.

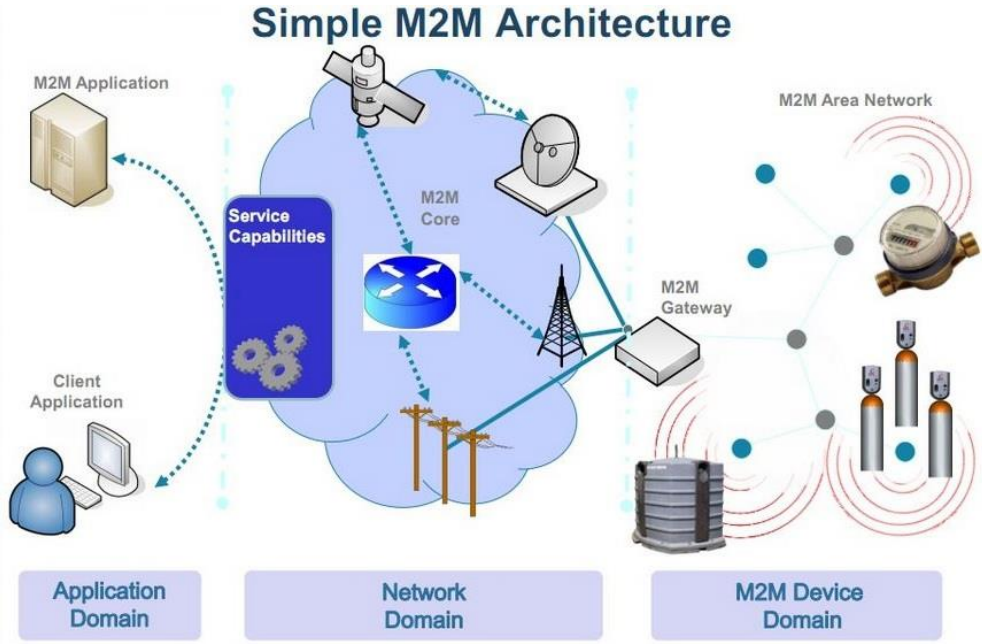


Figure 1-1 Simple M2M architecture *(pantechsolutions, 2013)*

1.2.5. M2M Device:

The device is able to respond to requests for information contained within such systems or capable of autonomous transmission of information.

Sensors and communication devices are the endpoints of M2M applications. Generally, devices can connect directly to an operator's network, or they will probably interconnect using WPAN technologies such as ZigBee or Bluetooth. Backhaul to an operator's network is then achieved via gateways that encapsulate and manage all devices. Consequently, addressing and identifying, e.g., routing, of the devices relies heavily on the gateways. Devices that connect via gateways are generally outside the operator’s responsibility but belong to M2M applications that are provided by service or application providers.

Sensors and devices that connect directly into an operator’s network (via embedded SIM, TPM and radio stack or fixed-line access) are endpoints of the network. Thus, the responsibility in terms of accountability, SLAs, etc., lies within the network operator (or virtual network operator). This holds especially with respect to TPM, where it is necessary to ensure that the module is reliable and well protected.

1.2.6. M2M Area Network (Device Domain):

Provide connectivity, e.g., personal area network, between M2M devices and M2M gateways.

M2M Gateway:

Equipment that uses M2M capabilities to ensure M2M devices inter-working and interconnection to the communication network.

Gateways and routers are the endpoints of the operator’s network in scenarios where sensors and M2M devices do not connect directly to the network. Thus, the task of the gateways and the routers are twofold. Firstly, they have to ensure that the devices of the capillary network may be reached from outside and vice versa. These functions are addressed by the access enablers, such as identification, addressing, accounting, etc., from the operator’s platform and have to be supported at the gateway’s side as well.

Thus, platform and gateway form a distributed system, where generic and abstract capabilities are implemented on the gateway’s side. Consequently, there will be a control flow between gateway and operator’s platform that has to be distinguished from the data channel that is to transfer M2M application data. Secondly, there may be the need to map bulky internet protocols to their lightweight counterpart in low-power sensor networks. However, the latter application might lose its relevance since there are implementations of IPv6 for sensor networks available, that allow an all-IP approach.

1.3.The vertical and horizontal approach

The fast expansion of the M2M and IoT technologies is seen as a promising area for business and researchers to invest in the future. However, there still some challenges that need to be addressed and solved. The solutions developed and implemented to date have been discussing to specific vertical applications requirement in isolation from all others. This created SILO solutions built on very diverse forms of technology, platforms, and data models.

In the vertical business model, the IoT device, the gateway (if used), and the cloud-based service are all provided and controlled by the same company. This approach has the advantage for the end-user that there are no compatibility issues to deal with among the various elements, and a single point of contact to deal with if anything goes wrong. The disadvantages are that the end-user is entirely dependent on the vendor for improvements, enhancements, or upgrades to the offering.

Vertical business models can also result in users needing several different systems to achieve a spectrum of tasks, each with its gateway and cloud operations. In a smart home, for instance, it is easy to end up with separate providers for security, HVAC, and appliance systems. This complicates system management for the end-user.

Interoperability is, in general, limited or non-existent. Overpowering this challenge requires more attention to the determination of more comprehensive standards regarding data models. Lack of joint capabilities and for avoiding these redesigning solutions in every single application, it is essential to add a service platform to reuse for multiple application. The comprehension of skills needed, as well as the rapid technology development, could help in changing the trend from vertically oriented systems, or application-specific silos, towards a more and more horizontal systems approach.

This vertical model follows most of the first IoT products to be released on the market. That is not surprising, given that there is yet little infrastructure in place to support a horizontal business. However, that is changing.

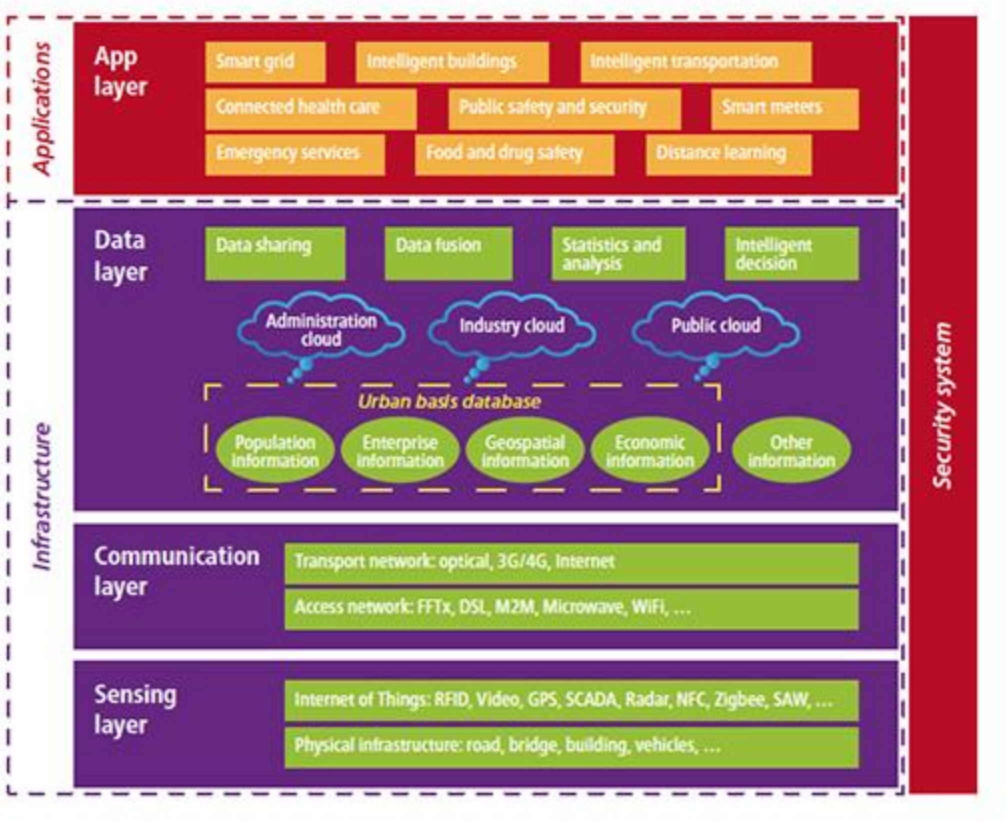


Figure 1-2 Architecture of sustainable city horizontal *(Union, 2015)*

The future IoT framework capabilities are horizontally placed on vertical IoT elements such as device, edge, and cloud. What is meant by horizontal, it is a coherent framework valid across a large variety of business, domain, network and devices that is a set of technologies and architecture and processes that will enable functional separations, in particular, application and system layers.

Such a platform will be based on a set of capabilities in the form of software modules that are offered to the IoT applications to accelerate their development, test, and deployment life cycles. (David Boswarthick, 2012)

The development and deployment of IoT applications could benefit from a set of building blocks that are carefully designed, tested, and optimized, nevertheless of the type the application being deployed.

The further development of that concept allows the IoT application to primarily focus on the business logic, leaving the IoT service capabilities with all other aspects of the claims, such as device activation, device monitoring, device localization, data storage, and mediation to the horizontal platform (running the IoT service capabilities), to name but a few. Once a set of reliable

IoT service capabilities has been specified, the logical next step is to expose them to the IoT applications through the use of application programming interfaces (APIs).

The primary motivation for developing IoT architecture is to promote the reuse of various resources and to allow the rapid introduction and deployment of new IoT services and applications.

1.4.IoT architecture:

The IoT is not a product, nor a network, nor a system: it is a system of systems and, in terms of communications, a network of networks, in which the Internet will play a unifying role, a kind of glue that will allow networks of very diverse natures to concatenate and constitute a mesh allowing objects located in the networks closest to the field, which we call in this study "end networks", to communicate with each other thanks to the universality of the Internet and its protocols.

Through the intensive development of the Internet of Things, it has now become a trend for everyone who works in this area of research. Further, it is now visible as the increasing number of application cases which rely on services of physical facilities exposed, such as sensors, actuators, RFID (Radio Frequency Identification) tags, machines, vehicles, and industrial embedded devices. (Juhani Latvakoski, Jaume, & Granqvist, 2014)

1.4.1. IoT Application:

The word *THING* in IoT means everything and anything around us that embraces machines. Connecting over 27 billion items and objects in 2020, the IoT is considered as the third wave of the Internet and supposed to grow more in the upcoming years.

The important thing is to outline some differences between M2M devices and what is referred to as “Things” or “Objects” in the so-called “Internet of Things” (IoT). M2M and IoT largely overlap, but neither is a subset of the other, and some areas are particularly specific to each:

- IoT is dealing with Things or Objects that may not be in an M2M relationship with an ICT system. An example of this is in the supermarket where radio frequency identification (RFID) “tagged” objects are offered to the customer. These objects are “passive,” and have no direct means with which to communicate “upstream” with the M2M application, but they can be “read” by an M2M scanner which will be able to consolidate the bill, as well as making additional purchase recommendations to the customer. From this perspective, the M2M scanner is the “endpoint” of the M2M relationship.

- There are M2M relationships initiated by devices that are to be direct human-machine interface extensions of a person (e.g., the above-mentioned end-user Kindle™) rather than as Things (e.g., the end-user refrigerator).

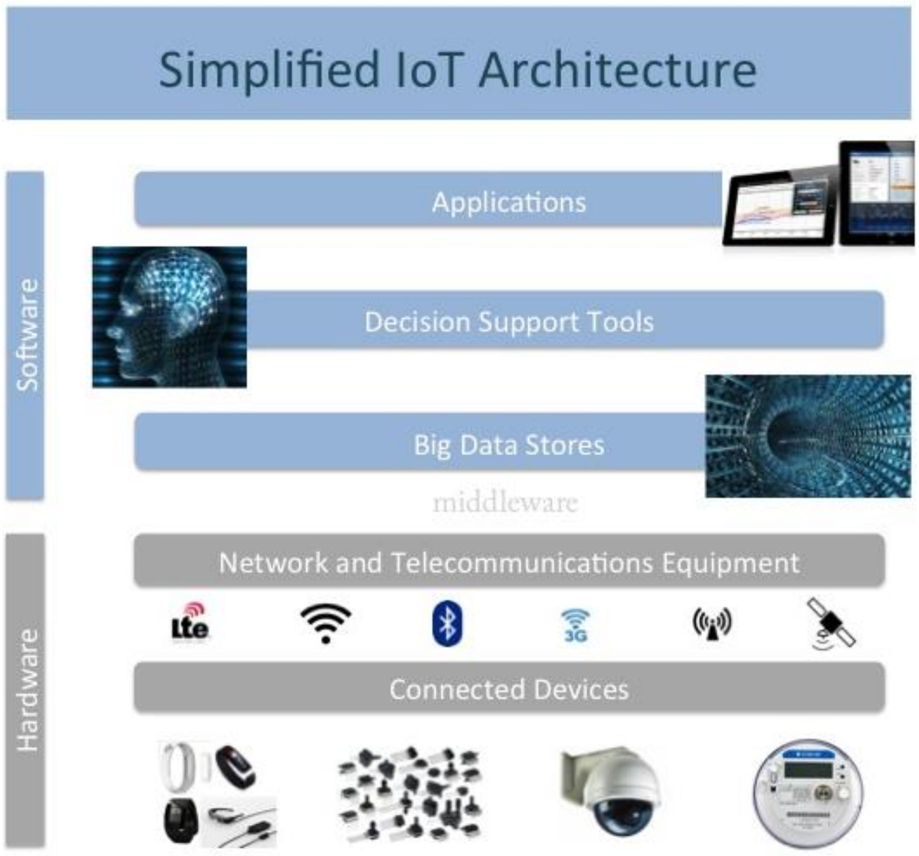


Figure 1-3 Simplified IoT architecture *(Internet of Things (IoT) , 2017)*

1.4.2. Requirements for the architecture:

There are specific particular IoT requirements that are unique to IoT devices and their supporting environments, e.g., various requirements emerge from the limited form factors and power available to IoT devices. (Fremantle, 2015) Other conditions are derived from the manner IoT devices are manufactured and used. The approaches are much more like traditional consumer product designs than existing Internet approaches. Of course, there are a number of best practices available for the server-side and Internet connectivity, and this must be remembered and considered.

The general specifications can be summarized in some main classifications:

- connectivity and communications
- device management
- data collection, analysis, and actuation
- scalability
- security

A. Connectivity and communications:

IoT connectivity technologies provide the network infrastructure and communication capabilities required by IoT devices to collect, transport, and exchange data over the Internet and to be remotely monitored and controlled. The study stream of IoT connectivity includes a variety of communication techniques used in different IoT projects and solutions, including:

- **Low-Power Wide-Area Network (LPWAN) technologies**, e.g., Sigfox, LoRa, NB-IoT

- **Cellular technologies**, including 2G, 3G, 4G, and 5G

- **Short-Range Wireless Technologies**, such as Wi-Fi, Bluetooth, Zigbee, and many others

- **Satellite technologies**, such as VSAT, BGAN, and the newest satellite based LPWAN

B. Device management:

Once an Internet of Things (IoT) device is installed, it is not a “fire and forgets” scenario. In fact, there will be bug fixes and software updates needed. Moreover, some devices will fail and need to be repaired or replaced, and each time this could happen. So, the company is on the hook to minimize downtime not only to keep the customers happy but to ensure that the revenue stream is protected.

How this is ensured by designing thoughtful device management into the product. Any IoT system must cover four critical categories of device management, which are:

- provisioning and authentication
- configuration and control
- monitoring and diagnostics
- software updates and maintenance

C. Data collection, analysis, and actuation:

As the new networks link data from products, company assets, or the operating environment, they will generate better information and analysis, which can enhance decision making significantly. Some organizations are starting to deploy these applications in targeted areas, while more radical and demanding uses are still in the conceptual or experimental stages. (Quarterly, 2010)

When products are embedded with sensors, companies can track the movements of these products and even monitor interactions with them. Business models can be fine-tuned to take advantage of this behavioral data.

In the business-to-business marketplace, one public use of the Internet of Things involves the use of sensors to track RFID (radio-frequency identification) tags placed on products moving through supply chains, thus improving inventory management while at the same time reducing labor capital and logistics costs. The range of potential uses for tracking is expanding. Sensor technologies are driving new business models in the aviation industry.

Data from large numbers of sensors, deployed in infrastructure (such as roads and buildings) or to report on environmental conditions (including soil moisture, ocean currents, or weather), can give decision-makers a heightened awareness of real-time events, particularly when the sensors are used with advanced display or visualization technologies.

The Internet of Things also can support longer-range, more complex human planning, and decision making. The technology is requiring great storage and computing resources connected to advanced software technologies that generate a range of graphic displays for data analysis. rise accordingly.

Making data the basis for automation and control means converting the data and analysis collected through the Internet of Things into instructions that feedback through the network to actuators that in turn, modify processes. Closing the loop from data to automated applications can raise productivity, as systems that adjust automatically to complex situations make many human interventions unnecessary. Early adopters are introducing fairly basic applications which provide a fairly immediate payoff. Advanced automated systems will be adopted by organizations as these technologies develop further.

D. Scalability:

Scale, by definition, refers to “the capability of a system, network, or process to handle a growing amount of work, or its potential to be enlarged in order to accommodate that growth.” Any server-side architecture would ideally be highly scalable and be able to support millions of devices, all continually sending, receiving, and acting on data. However, many “high-scalability architectures” have come with an equally high price – both in hardware, software and in complexity. An essential requirement for this architecture is to support scaling from a small deployment to a very large number of devices. Elastic scalability and the ability to deploy in cloud infrastructure are essential. The ability to scale the server-side Out on small, low-cost servers is an important mechanism to make this architecture affordable for both small and large deployments. Here we tend to define the different types of scalability in context to IoT.

Vertical Scalability:

It is also referred to as scaling up, which is the ability to increase the capacity of existing hardware or software by adding more resources to it. For instance, we add processing power to a server to increase its speed. Moreover, (Anisha Gupta., 2017) we can scale a system vertically by expanding it by adding more processing, main memory, storage, and network interfaces to the node in order to satisfy more requests per system. Hosting services companies surmount by increasing the number of processors. It means to add resources to a single node in a system which involves adding CPUs or memory to a single machine. Such vertical scaling of current systems facilitates them to utilize virtualization technology more productively. The main advantage of vertical scalability is that it consumes less power if we compare to running multiple servers, reduces administrative efforts as we need to handle and manage only one system. Moreover, the implementation is more straightforward, reduces software costs and application compatibility is retained. As there are advantages, there are also disadvantaging of this type of scaling which includes greater includes more significant risk of hardware failure which will cause more significant outages, severe vendor lock-in and the cost of the overall implementation is high.

Horizontal Scalability:

It is also referred to as scaling out, which is the ability to increase the capacity by connecting the multiple hardware or software entities so that they can work together as a single unit. Horizontal scalability can be achieved by adding more machines into the group of resources and adding more nodes to a system, for instance, adding a new computer to a distributed software application. The examples of this can be SOA systems and web servers which scale out by adding more and more servers to the load-balanced network so that the incoming requests can be distributed between all of them. Cluster is a familiar term for describing a scaled-out processing system. An example might include scaling out from one Web server system to the other three systems. System architects may set up a swarm of small computers in a cluster to obtain cumulative computing power that many-a-times exceeds that of computers based on a single traditional processor. Application scalability indicates the improved performance of running applications on a scaled-up version of the system.

E. Security:

Security is one of the most critical aspects of IoT. IoT devices are often collecting highly personal data, and by their nature are bringing the real world onto the Internet (and vice-versa). E.g. Home IoT devices and systems need to manage security with minimal and potentially no consumer intervention, and without the consumer having any specialist knowledge of security or IT principles. This contrasts with other IoT environments which are more formally managed and directly regulated, (Foundation, 2018) such as enterprise and transportation. One key challenge in the consumer environment is that home IoT solution providers cannot assume a reliable or enough level of users' understanding of security governance.

It is also challenging to manage and maintain a complex system of devices available on the market using a variety of proprietary interfaces and protocols in the home environment. Interoperability between IoT devices is a crucial aspect of not only hub architectures like this one but any IoT deployment implementing multiple devices. Good interoperability assists with security management across the IoT ecosystem and reduces the effort required of the home IoT administrator.

1.5.Value of IoT markets:

Beyond the gadget side, a real revolution in business models is underway, based on data and service. This movement will probably be faster than expected. Machine to Machine (M2M) is certainly not new. However, we are no longer talking about M2M, but about multiple connections between machines, and between people, processes. The projections and trends towards IoT become a reality, it may force a shift in thinking about the implications and issues in a world where the most common interaction with the Internet comes from passive engagement with connected objects rather than active engagement with content. The potential realization of this outcome – a "hyperconnected world" is a testament to the overall-purpose nature of the Internet architecture itself, which will not place significant limitations on applications or services that may make use of the technological advances.

As a result, governments, policymakers, and businesses must recognize and embrace the enormous opportunities the Internet can create, even as they work to address the risks to security and privacy the Internet brings. As the Internet's evolution over the past two decades has demonstrated, such work must include helping to nurture the development of a healthy Internet ecosystem, one that boosts infrastructure and access, builds a competitive environment that benefits users and lets innovators and entrepreneurs thrive, and nurtures human capital. Together, such components can optimize the continuous effect of the Internet on financial growth and prosperity.

➢ IoT influence on Competition

In the economic context, the question of whether the IoT can and will influence global competition needs to be addressed. This issue arises in particular when businesses from developing countries can be included in the global exchange of goods through the IoT. On the one hand, competition could be stimulated with the inclusion of more participants in the trade. It is estimated that the current market size of 5.5 billion USD will move to over 20 billion USD in 2019. Established companies from developed countries may need to reconsider their practices and make a more significant effort and possibly increase innovation. Endeavors to stay competitive. Because countries in the East are in the position to produce goods more cheaply, developed countries will shift the emphasis in their production. Different values, such as social interactions, democracy in the producing country. Will emerge. Also, innovation becomes more important than optimization. Businesses have to seize the unknown and be open, i.e., bond with other partners to increase innovation.

Furthermore, an increase of information for businesses also results in competitive advantages in terms of process optimization: the IoT allows for near real-time measurement, enabling businesses to produce what is needed at precisely the time it is needed. On the other hand, an increase of globally accessible information about the production of goods, prices, and supply chains could also harm competition in an extended sense. If businesses can retrieve information from the IoT regarding production methods and innovations, this could not only infringe intellectual property provisions but also lead to a decrease in innovations.

Incentives for businesses to invest in studies and development is diminished if competitors may copy any success. By, e.g., reducing their output, raising prices or degrading the quality of their products, action has to be taken. If businesses use the information provided through the IoT to harm welfare Free and open trade is of central importance in the application of competition law because ultimately, the process of competition is intended to deliver benefits to all market participants.

Therefore, an institution should be assigned with the task to take action against offending (groups of) undertakings, in order to protect the interests of the competitors and the consumers. Such action can, e.g., encompass the order to reduce prices. Nevertheless, the concept of competition should not be used to control the pricing of products. Another issue of international trade that should also be pointed out when discussing the IoT is the principle of fair trade. Whereas this issue is not new, it makes sense to accentuate the fact that companies – in the information provided within the IoT – must not give untrue or misleading particulars, use false titles, induce confusion with other products, make degrading comparisons with products from different manufacturers, etc. An international body to judge incidents of unfair competition does not yet exist; several occasions can only be considered by domestic courts under national law.

- **What governments can Do**

IoT, as a technology and a governance discipline, is still in its infancy, and while there is real excitement about it within both government and the private sector, the evidence of success is still patchy. Governments have a vital role in catalyzing space and contributing as partners/leaders in the long term. Based on the findings and the characteristics of successful pilots, (Lal Das P. , Beisswenger, Mangalam, Yuce, & Lukac, 2017) we present a conceptual toolkit containing ideas and resources for government agencies that want to implement IoT-based initiatives within their jurisdictions. The toolkit has three pillars:

A. Leadership/policy:

Vision toward balancing regulation with a proactively and iteratively engage in policy development. Given the pace of disruption, the wide diversity of stakeholders, the cross-boundary nature of the digital economy, and the scale of new digital services, it is vital for policymakers to look beyond policy models that served the public sector well before the advent of the digital economy.

B. Strategy and implementation:

Maintain sandboxes for pilots and prototype for pilots to test policies and solutions. Sandboxes facilitated directly or indirectly by government, in the form of physical spaces, clusters, and environments for running pilots and proofs of concept was the one constant in all the cases we observed.

C. Capacity and engagement:

Engage and partner with local communities through education and outreach. Community groups and citizens can play an early and proactive role in generating ideas, providing feedback and input for tackling sensitive and delicate issues such as data privacy and data ownership and ensuring the long-term sustainability of such projects. Develop IoT capacity within and outside government (work with academic and educational institutions to develop curriculum for current and future capacity development). IoT-based applications and processes require a very different skillset and competency from the people managing them.

Conclusion:

The internet is not just for communicating with people; it has shown the capability of the connection and the use of many heterogeneous physical and virtual objects which interact with speeds, scales, and capabilities far beyond what people needed initially or used. Involving in more horizontal architecture, IoT opens more extensive opportunities and infinite end-user solutions. Business in the IoT is promising to reach a market size of 20 billion USD by 2019 countries are now considering new policies to enhance the use of IoT, all with keeping privacy and control.

Chapter 2: Home Automation

Content

2.1. Introduction:

Home automation is every system that makes it easier and more efficient to use the lighting, heating, and equipment of your home. It can be as easy as controlling a few lights remotely or automatically, or it can be a full system that controls all significant components of your home, customized to your personal preference. In this chapter, we will see more how smart houses interact in the overall energy loads, and also, we will see the different project that we took into consideration to map our phone application development.

2.2. Smart house monitoring:

Smart home monitoring is a system that is set up in a house which allow the owners to alert them if something is not right at their houses or can allow a person to change something, such as temperature.

With improved usability and new capabilities, the motivations for installing smart home technologies have become broader as well. The vision of green buildings, capable of significantly reducing energy and water consumption, is finally becoming real. (Kyas, 2017) Other use cases are safety management, home automation for the elderly and disabled (assistive demotics), people, and remote building control. Beyond providing standalone solutions for individual buildings, smart home technologies have become a central component of the more substantial, the more considerable Internet of Things concepts such as smart cities, smart industry or smart grids.

2.2.1. Households Energy saving:

In 2017, the European Union office for statistics Eurostat performed a detailed, nationwide survey of home energy consumption. Figure 2-1 presents the energy consumed by the average European household in 2017, according to end-use.

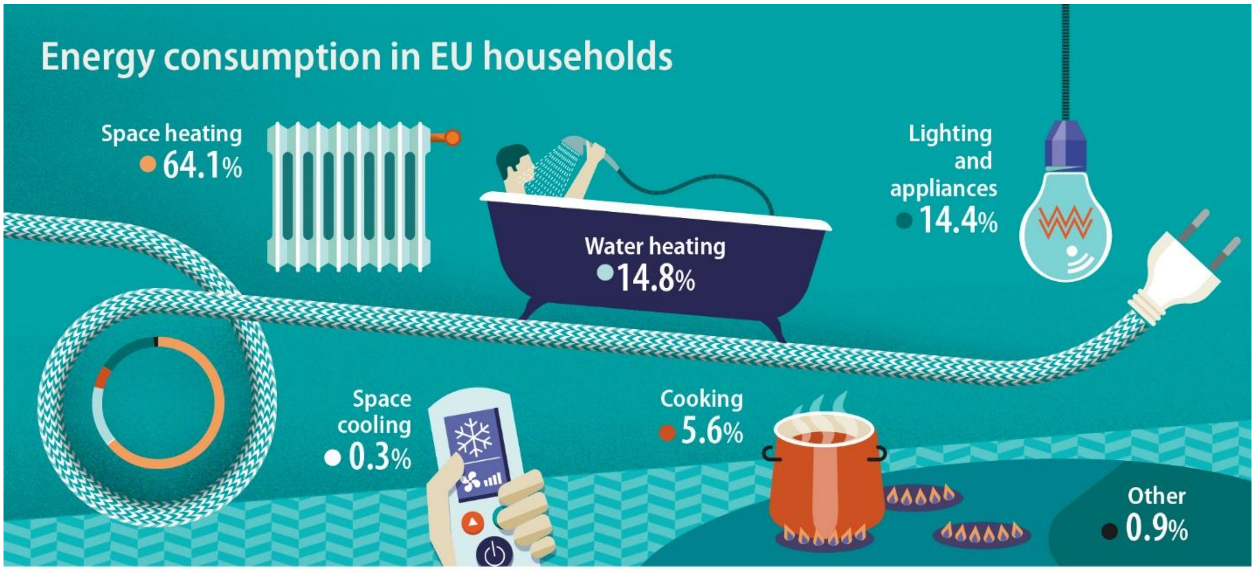


Figure 2-1 Typical European household energy consumption by end-use, 2017 *(Energy consumption and use by households, 2019)*

We ask a great deal from our home because they need to shelter us in comfort in addition to providing heat, light, hot water, and all the modern conveniences and entertainments, and we usually expect all of this to happen economically over a long period. Many people do not think in terms of the "operation" of homes at all; for example, they buy a house looking for the price, square footage, location, and amenities. Optimistically we assume that the builder and the building codes and rules have taken care of everything else and that the house will "work." Builders and building owners rarely envision the house as a living organism with multiple, interdependent systems and functions. Very often, heating systems are installed almost as an afterthought and indoor air quality are largely ignored.

- **Efficiency:**

In today's globe, the need for energy-efficient technological alternatives is increasingly common. Current developments, however, fail to provide end-users with a flexible solution that can be commonly applied in public or industry areas. (Anna , Vikash , & Slobodanka)

Energy efficiency is a measure of energy used for delivering a given service. Improving energy efficiency means getting more from the energy that we use. Capacity to generate the desired effect with a minimum amount of effort or waste. The energy efficiency of an appliance is a comparison, or ratio, of the useful energy output to the total energy input. For example, we want a light bulb to produce light. We all know that after a light bulb has been on for any length of time, it gets hot. Since heat is not what we need from a light bulb, we consider the heat to be a waste product the heat represents inefficiency. This is particularly important at the user interface where customers of energy should be able to participate in readily inefficient energy savings. Using semantically related information, we strive to actively assist end-users in making informed choices to regulate their energy consumption effectively.

- **Cost-Effectiveness:**

The question of how to define the cost-effectiveness of energy efficiency as a critical resource in meeting future energy needs. How cost-effectiveness is defined substantially affects how much of house efficiency potential will be accessed and whether consumers will benefit from the lower energy costs and environmental impacts that would result.

2.2.2. Home automation:

Home automation was mainly focused on controllable power-outlets or light switches using copper telephone lines or infrared (IR) controls. Technologies developed more than thirty years ago, which from today’s perspective are slow, unreliable, and insecure, were at the heart of building control. The rapid developments in mobile communications have introduced a technological leap forward in-home automation since wireless networks (3G, LTE, Wi-Fi, LoRa) and smart devices with wireless communication interfaces (Bluetooth, ZigBee, Wi-Fi) are omnipresent and allow the user to take building automation to the next level. Instead of merely switching power outlets on and off, specific and vital operations of consumer electronics, home appliances or infrastructure materials may be stimulated. As a result, instead of basic functionality, home automation today can deliver capabilities that have a real impact on comfort, security, as well as energy conservation of residential, commercial, and industrial buildings.

Equally important as the mobile communication revolution has been the advances in sensing technologies, while much less in the focus of the public. Comparable with the pace of digital evolution, sensor capabilities have improved. At the same time, size and prices of sensors are at a fraction of what they used to be a few years ago. In addition, new generations of digital sensors have replaced analog sensing technologies, allowing a single sensor to measure a multitude of parameters.

Some of the processes within the Design, execution, and maintenance of smart home devices:

- Deployment of computer control for heating, ventilation, and air-conditioning technologies.
- Internet / remote / network access to all configured elements and tools.
- By installing and maintaining network-enabled security camera and access control systems.
- Central control and management capabilities over electrical fixtures and electronic appliances.

Devices within the home automation system connect and communicate with each over a local wired or wireless network. The entire home automation system usually requires system management software, installation of device/appliance controllers, motion and temperature sensors, and other components.

- **X10 – home automation standard**

The beginning of modern home automation technology can be argued to be found with the introduction of the X10 technology standard. Conceived in 1975 by Pico Electronics, X10 laid out the framework for allowing remote control access of domestic appliances. The X10 standard was designed to allow transmitters and receivers to work over existing electrical wiring systems by broadcasting messages such as "turn off" and "turn on" via radio frequency bursts. X10 products began to make their way into stores geared towards electronics enthusiasts and shortly after. Throughout the years, assistance for Mac and Windows has been included and has given those willing to participate in home automation the capacity to code their lighting equipment, thermostats and garage doors from their home computers.

As protocols such as FTP and HTTP have become the norm for obtaining information around the Internet, hardware programmers have seen the chance to exploit these communication technologies on open-source hardware devices. Whereas X10 appliances had little way to know whether a signal was successfully sent without the acquisition of expensive "two-way" devices, and web technologies provide a framework for the return of error codes and messages. Alongside the explosion in hardware, there was also an equivalent explosion in software. One particular product of interest that we will look at is the open-source Android operating system.

2.3. Design Application for home user experience

“Homes are not smart. People are smart.” (Berry & Elise, 2014) The inhabitant of the house decides if home behaves in an appropriate and meaningful way. Over the last decades, “traditional” homelife has been under the massive attack of ever faster-developing information and communication technologies (ICT). The positive side of the advancement of technology has opened significant opportunities for designers, in particular, to enhance product-service and create new home experiences. Three challenges are mainly addressed to modern designing or studying smart home environments.

- **Worthwhile user experience:**

The focuses of design naturally shift toward the home experience interaction itself. More specifically, the quality of interaction design, then, is defined and evaluated in terms of the usability of the system.

2.4. Jua House Project:

The Jua House’s goal is to bring innovative solutions to address the challenges of housing and cities in Africa, the innovative aspect of the Jua house is in its architecture, solar system, energy management, HVAC, water and waste management systems.

The Jua House will be designed using recycled containers. These containers are widely available, durable, cheap, and flexible to design and adapt to appropriate insulation techniques. The house will implore the North African inspired fountain and patio design for a comfortable indoor microclimatic space and passive cooling using a chimney and maximized windows for cooling in summer. It will come with movable walls, mobile interior design, innovative structure that combines the use of metal and soil compressed bricks, and an Aquaponics’ system. Also, windows will be positioned to maximize daylight access.

With respect to power needs, the house will be powered by rooftop-mounted solar arrays, solar thermal collectors with maximum power-point tracking. This energy will be utilized using the smart integrated renewable energy to meet ends needs where solar energy is not only used for electricity generation but also other needs such as water management, passive lighting, passive and active heating, and cooling, etc. The solar system will come packed with a small battery bank to store electricity for emergency use and optimize the self-consumption of solar energy so that only the surplus electricity is fed back into the grid; moreover, a hybrid inverter to convert DC power from modules to usable AC power that can be used to power loads or fed into the grid. Moreover, this inverter will store solar power to the battery and convert stored AC from the batteries to power critical loads in the case of emergencies.

An integrated energy-monitoring-system will be used to configure, monitor, and control the energy system of the house. The lighting, HVAC, fire, smoke-detection, motion-detector, CCTV, equipment and appliances, etc., will be controlled remotely or using a smartphone. This will be done using sensors, clock controls, programmable thermostats, energy management control systems, and infrared thermal scanning, etc. The application will be a prototype for the monitoring system in the house for future development and enhancement.

2.5. Requirements from specific H2020 projects/European project:

2.5.1. Sinfonia:

European cities have a crucial role to play in the transition towards a low-carbon economy. Faced with the challenge of ensuring the quality of life of their citizens while becoming more energy-efficient, Cities need to look at the system level and develop embedded urban development tactics which will make them respectively sustainable and better housing options.

The SINFONIA project is a five-year initiative to deploy large-scale, integrated, and scalable energy solutions in mid-sized European cities. At the heart of the initiative is a unique cooperation between the cities of Bolzano and Innsbruck, working hand in hand to achieve 40 to 50% primary energy savings and increase the share of renewables by 20% in two pioneer districts. This will be done through an integrated set of measures combining the retrofitting of more than 100,000m² of the existing surface, optimization of the electricity grid, and solutions for district heating and cooling.

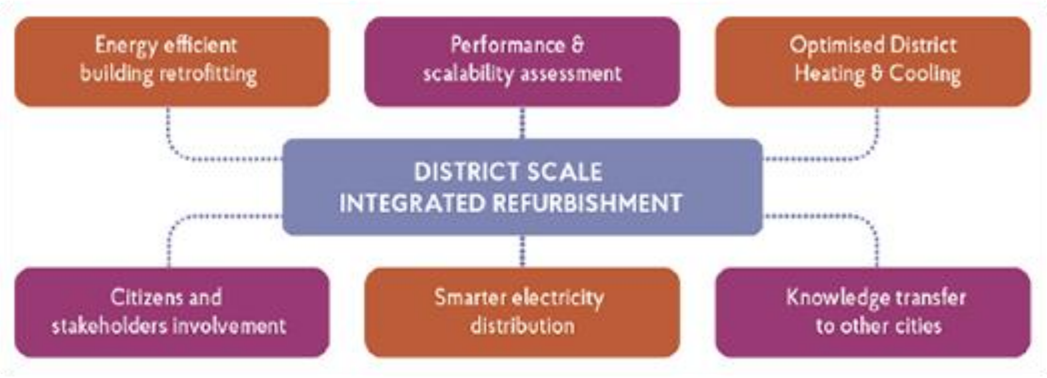

Energy efficient building retrofitting
Performance & scalability assessment
Optimised District Heating & Cooling
DISTRICT SCALE INTEGRATED REFURBISHMENT
Citizens and stakeholders involvement
Smarter electricity distribution
Knowledge transfer to other cities

Figure 2-2 Sinfonia essential topics (Sinfonia, s.d.)

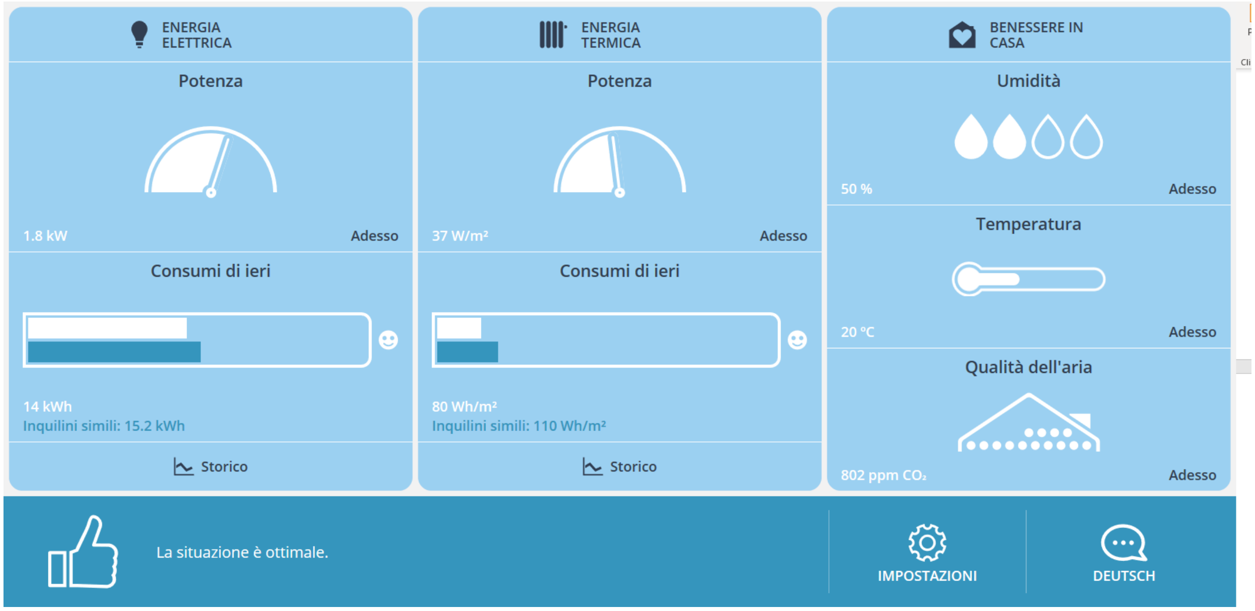
A. Web APIs

A "web service" is a web-based application that provides resources in a format consumable by other computers. Web services include various types of APIs, including both REST and SOAP APIs. Web services are request-and-response interactions between clients and servers (a computer requests a resource, and the web service responds to the request).

The API is typically described as a collection of Hypertext Transfer Protocol (HTTP) request messages when it Included as part of web design. The API is typically characterized as a collection of Hypertext Transfer Protocol (HTTP) request messages when it is used in the context of web development. Besides the definition, we need to set up the structure of response messages, which is usually in an Extensible Mark-up Language (XML) or JavaScript Object Notation (JSON) format. Besides the definition we need to set up the structure of response messages, which is usually in an Extensible Mark-up Language (XML) or JavaScript Object Notation (JSON) format. While "Web API" is virtually a synonym for web service, all APIs that use the HTTP protocol as the transport format for requests and responses are considered "web services." With web services, the client making the request for the resource and the API server responding can use any programming language or platform it does not matter because the message request and response are made through a standard HTTP web protocol.

B. Sinfonia Web API:

The WebApp was developed for individual residentials based on Sinfonia general strategy. Therefore, using a database in storing different parameters (temperature, air quality, electrical consumption) from sensors located in different zones of the house sending live data to the inhabitant of the house. Using a touch screen tablet for parameters visualization inside the house. The WebApp get different information in three significant areas electrical energy, thermal energy, and wellbeing, from different sensors and also from smart meter and display it in the touch screen tablet; one of the general aims is to bring the inhabitant firstly in sensibilization of excessive energy consumption to start good practice in reducing of the energy-intensive zones of the house which are consuming too much energy, the WebApp will also help to identify abnormalities in the system and suggest solutions for instance if the air conditioning is consuming too much energy in a small area the inhabitant will be notified the if there is a window or door open which is leaking the energy.



La situazione è ottimale.

Figure 2-3 Sinfonia WebApp user interface

2.5.2. Heart project:

HEART is a multifunctional toolkit which integrates several components to transform existing buildings into energy-efficient smart buildings. It is also a quick decision-making tool which utilizes advanced data analysis to predict and guarantee energy efficiency. While HEART is developed with a focus on existing buildings, the concept can be extended to new residential and commercial buildings.

The HEART toolkit utilizes components such as ICT, BEMS, HVAC, BIPV, and Envelope Technologies. These cooperate organically to achieve high levels of energy efficiency and allow for an effective interface with the Smart Grid. The system is driven by a cloud-based platform which supports decision-making in the planning/design phase and optimizes energy performance in the operational one.

In designing this toolkit, the project progress and enhances energy savings and the use of sustainable energy in houses across Europe. Especially in Central and Southern Europe, where climatic change leads to an increase in electricity consumption in both summer and winter seasons.

How does HEART contribute?

- Improves European building renovation process by simplifying it.
- Reduces total energy consumption, integrates renewable energy, and rationalizes energy flow between buildings and smart grids.
- Involves stakeholders, supports energy financing, as well as the exploitation of renewable energy.
- The model can be applied to new housing premises buildings in addition to existing buildings.

The phone application associated with the heart project will have the ability to get temperature data from sensors in the house and also to actuate central (heating/AC) to increase or decrease the temperature.

2.6. Conclusion:

This chapter highlighted state of the art for home automation starting from the first standard that was used in this field, it also shows the immense benefit in using a smart controlling devices in the energy efficiency and cost-effectiveness for the house occupants, three main projects have also been described and will be more discussed in the next chapter.

Chapter 3 Phone application development

Content

3.1. Introduction:

In this chapter we are going to go over the different steps we have followed to develop the primary application for home automation using Android Studio interface and Java programming language this application will use a weather forecast API and predefined Sinfonia API for fetching data also we will see the development of HEART actuation API using Python language and using HTTP communication protocol to connect to a database which will simulate house environment.

3.2. Phone application architecture

3.2.1. General architecture:

As it is shown in the figure 3-1 the general architecture is composed in three main parts: the first one which is the phone that will be connected to the APIs server using a specific communication protocol, then the APIs interact directly with the sensors and actuators of the house.

This loop will be closed when the APIs receives the data back from the house and to the mobile phone again, and to visualizing everything in the application, this method will help speeding up the commands from the phone to the house.

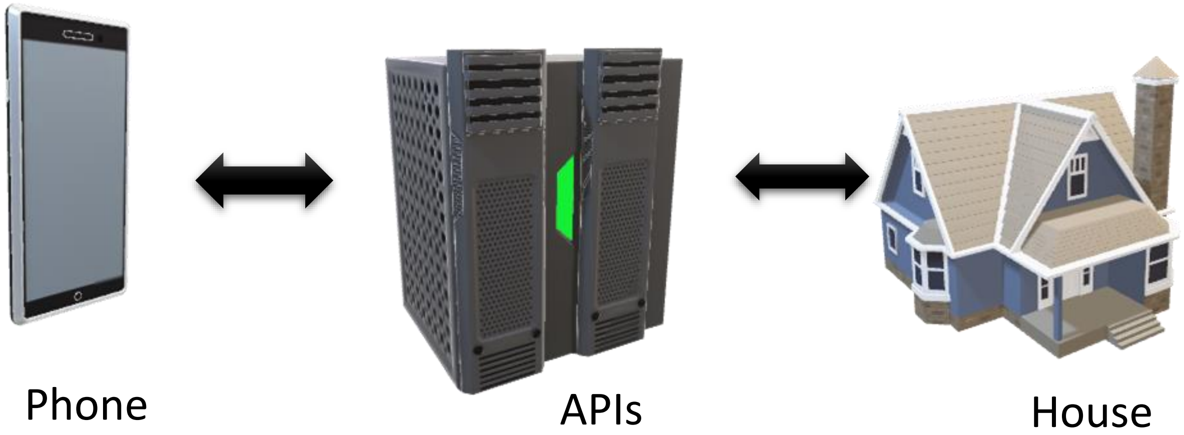


Figure 3-1 General architecture

3.2.2. Detailed app chart:

- **Phone:**

The phone will be the main subject in this thesis we were able to develop an Android application which will interact with the different APIs associated with it. The operating system for development is Android. The primary application has many features and functionalities the most important ones are the comfort, the electrical and the thermal energy.

- **API:**

Application Programming Interfaces (APIs) are the hidden backbone of our phone application, which allows the communication. By implementing the API using Swagger and then Flask as a web server where the API runs the Moreover, web server will send a response of that request via API response. This will make the application work smoothly and without getting bulky. The RESTfull API will interact in two ways in getting information from the phone to the house and sending back to the phone.

- **House:**

The house will be occupied with sensors and appliances that interact with the phone passing by the APIs, also using a communication protocol.

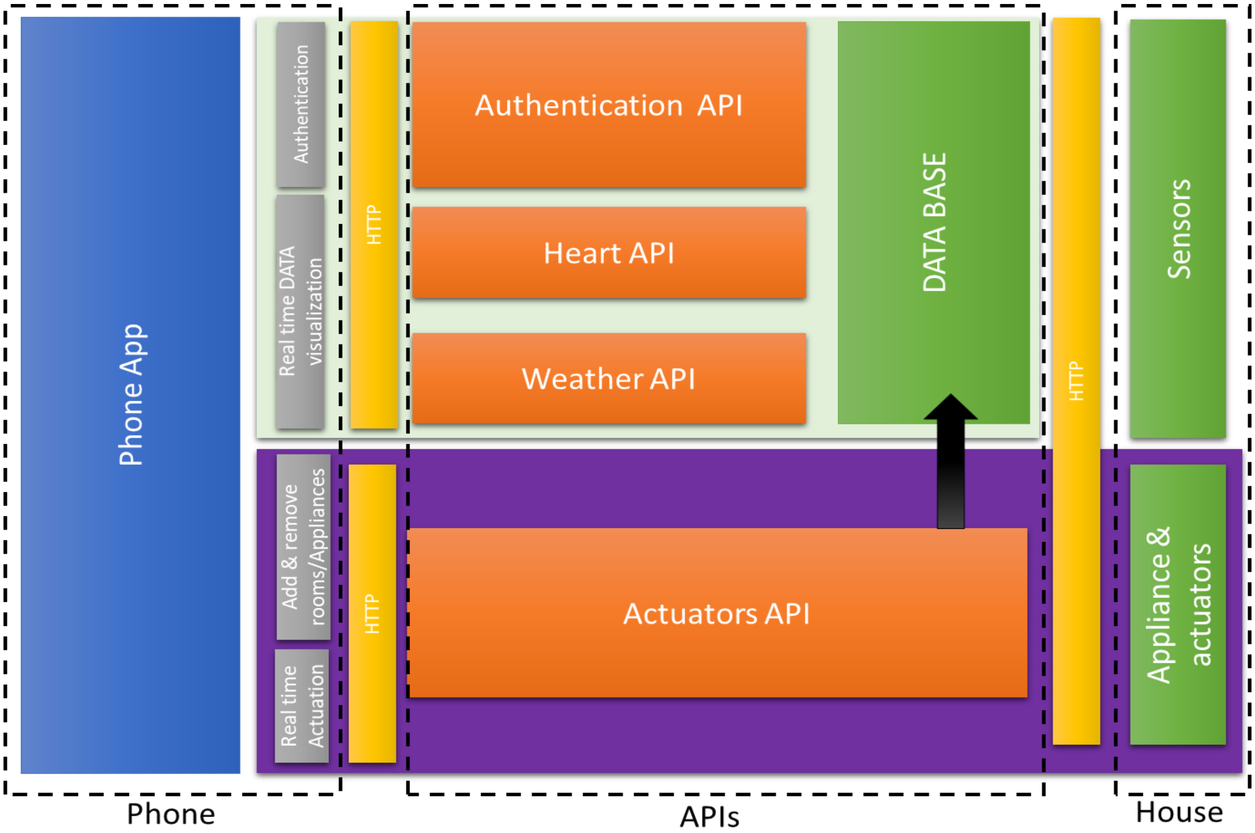


Figure 3-2 Detailed architecture

3.3. Description of the workflow:

3.3.1. Setting up the work environment:

3.3.1.1. Installing Adobe XD (user experience interface):

Adobe XD is a vector-based tool created and released by Adobe Inc for the design and prototyping of user experience for web and mobile applications. The software is available for MacOS, Windows, iOS, and Android applications.

Adobe XD was used to create the user interface prototype for the first and the second application this allows us to speed up the workflow and to create a road map to follow while creating the main application.

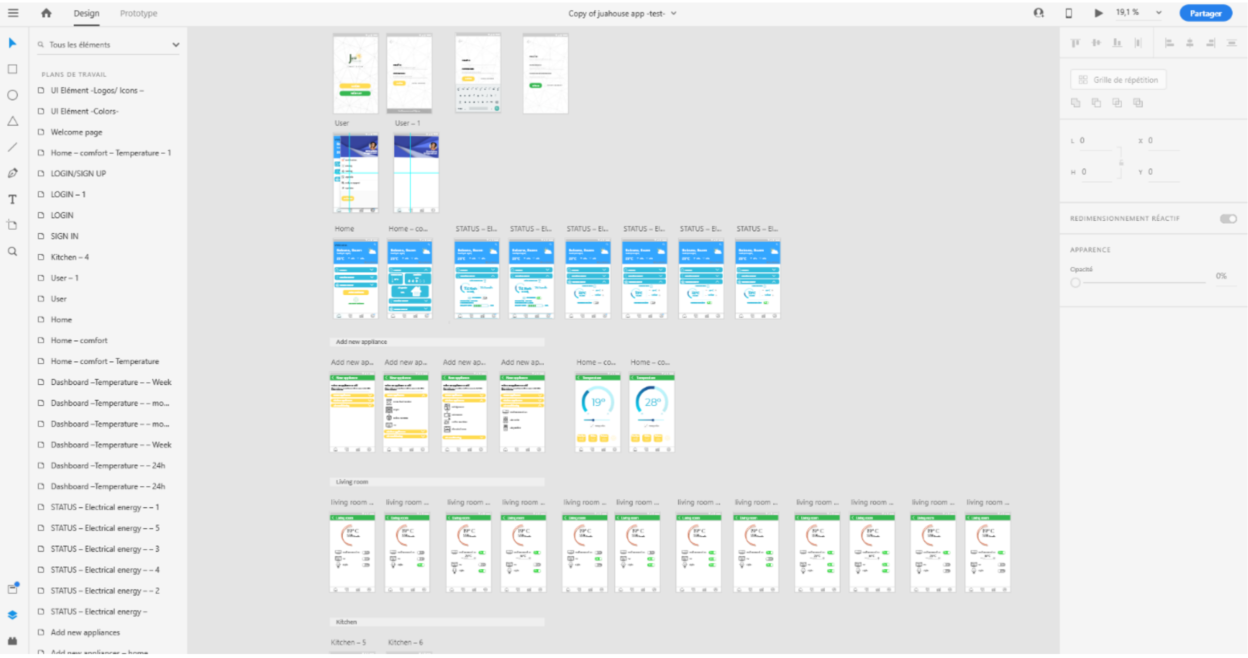

Design
Prototype
Copy of juahouse app -test-
19,1 %
Partager
Tous les éléments
PLANS DE TRAVAIL
UI Elément -Logos/ Icons –
UI Elément -Colors-
Welcome page
Home – comfort – Temperature – 1
LOGIN/SIGN UP
LOGIN – 1
LOGIN
SIGN IN
Kitchen – 4
User – 1
User
Home
Home – comfort
Home – comfort – Temperature
Dashboard –Temperature – – Week
Dashboard –Temperature – – mo...
Dashboard –Temperature – – mo...
Dashboard –Temperature – – Week
Dashboard –Temperature – – 24h
Dashboard –Temperature – – 24h
STATUS – Electrical energy – – 1
STATUS – Electrical energy – – 5
STATUS – Electrical energy – – 3
STATUS – Electrical energy – – 4
STATUS – Electrical energy – – 2
STATUS – Electrical energy –
Add new appliances
Add new appliances – home
Grille de répétition
REDIMENSIONNEMENT RÉACTIF
APPARENCE
Opacité
0%
Add new appliance
Living room
Kitchen
Kitchen – 5
Kitchen – 6

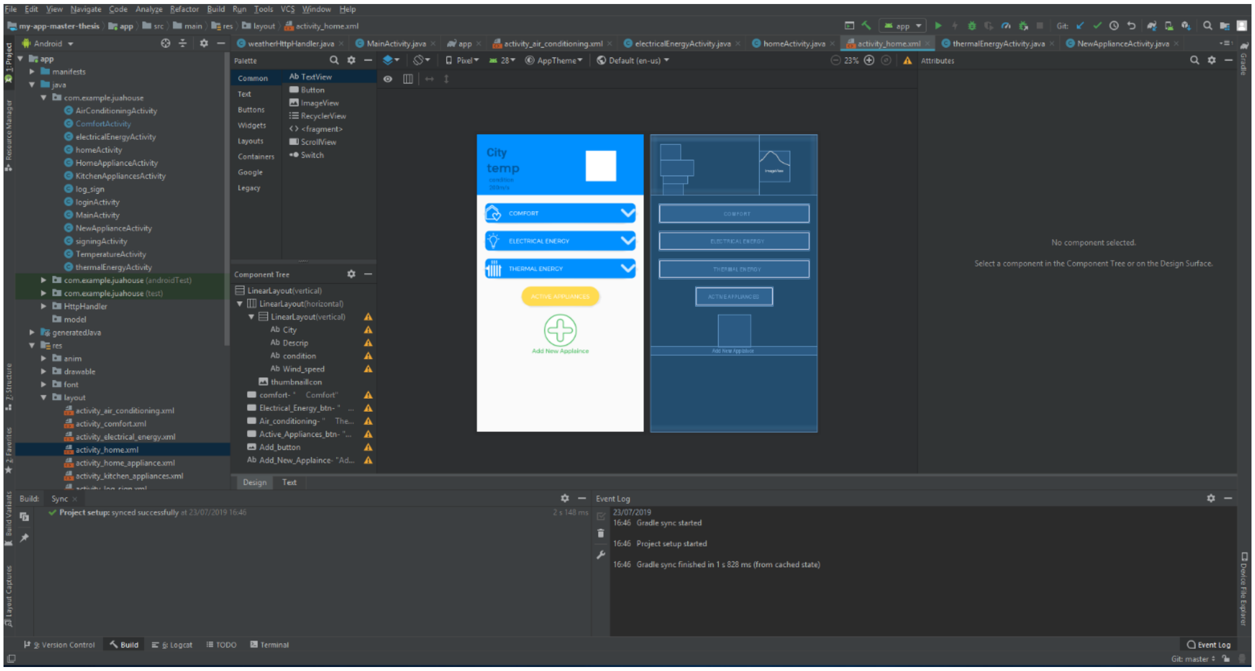
Figure 3-3 Adobe user experience interface.

3.3.1.2. Installing Android Studio:

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance our productivity when building Android apps, such as:

- A flexible Gradle-based build system.
- A fast and feature-rich emulator.
- The unified environment where we can develop all Android devices.
- Instant Run to push changes to the running app without building a new APK.
- Code templates and GitHub integration will help to build standard app features and import sample code.
- Extensive testing tools and frameworks.
- Linking tools to catch performance, usability, version compatibility, and other problems C++ and NDK support.

Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine.



No component selected.

Select a component in the Component Tree or on the Design Surface.

Build: Sync

```
Project setup: synced successfully at 23/07/2019 16:46        2 s 148 ms
```

Event Log

```
23/07/2019
16:46  Gradle sync started

16:46  Project setup started

16:46  Gradle sync finished in 1 s 828 ms (from cached state)
```

Figure 3-4 Android studio interface.

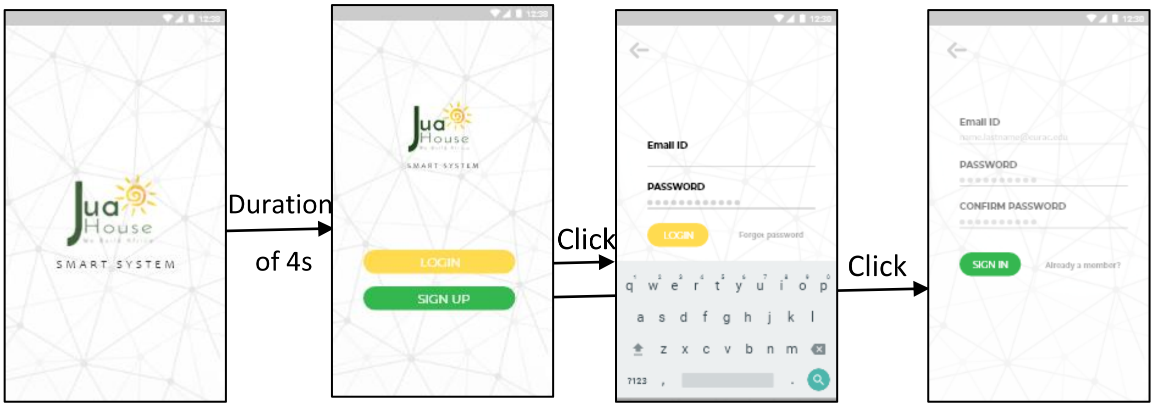
3.3.1.3. Design thinking and application prototyping:

Design thinking is a structured process to find solutions to complex human problems. The uniqueness of the design thinking process is that it helps people to define and solve problems that are unstructured and have no historical references. It helps dissect problems that are complex and frame/reframe areas that require solutions. There are several examples where the application of design thinking has been extraordinarily successful in design-led companies. Design thinking is at the heart of strategic growth and organizational change. It has been applied to the shape of the user interface experience for the basic phone application which was for the Jua house project, and this has helped to build up a great design that can be scalable and customized.

3.3.2. Sinfonia application functionalities:

3.3.2.1. Authentication:

Authentication shall apply to the mechanism of verification of identification. Although sometimes used interchangeably with permission, authentication is a totally different function. In this authentication, the user proves his identity to the application by providing valid credentials for verification. Authentication is often proved through a username and password, sometimes combined with other elements called factors. In this study, we have not focused on the security part, which means the authentication still not been involved in the programming and needs to be developed in the future.



Splash page LogIn_Signup page LogIn page SignUp page

Figure 3-5 Splash page and authentication for Jua house project

3.3.2.2. Home page

The next is the home page of the application, it will show the primary information to the user, in the background the home page activity will get the weather data from weather API and data from the Sinfonia API which was then retrieved from the sensors inside the house.

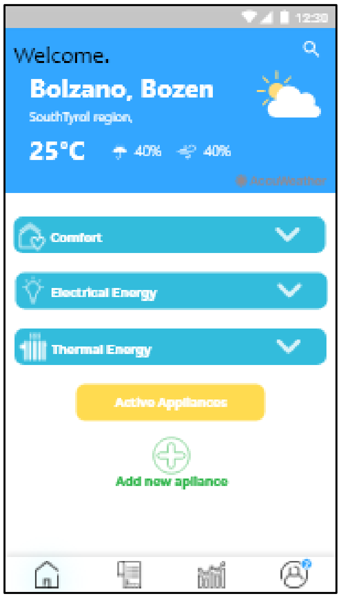
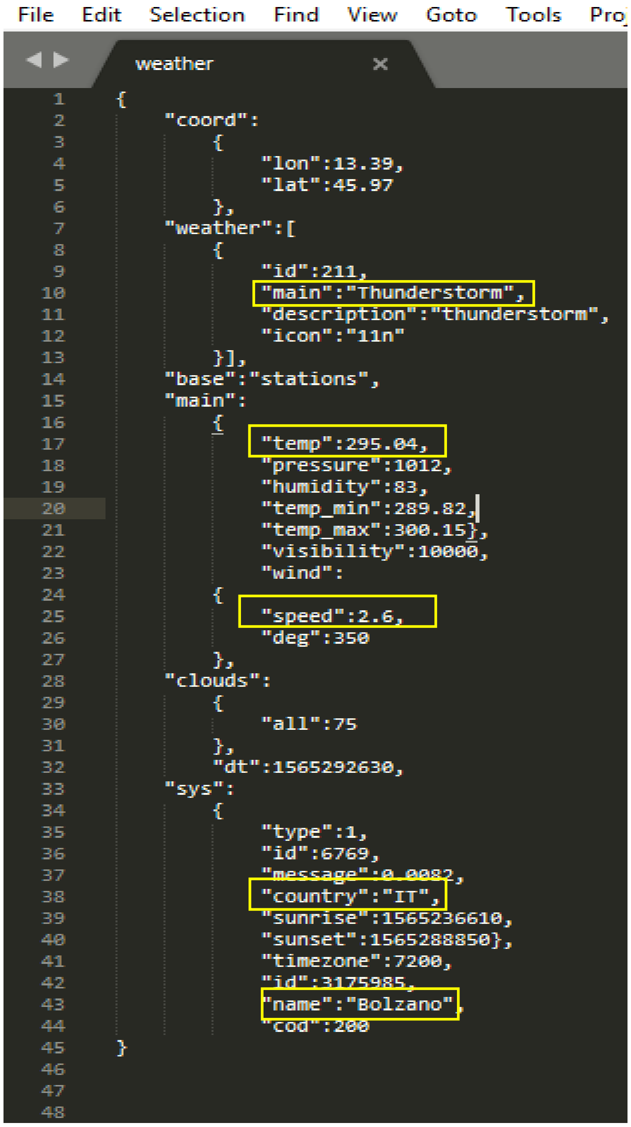


Figure 3-6 Home page Activity

3.3.2.3. Weather API:

To get weather data for the city of Bolzano, for instance, we used OpenWeatherMap API, which provides:

- Access current weather data for any location including over 200,000 cities
- Current weather is frequently updated based on global models and data from more than 40,000 weather stations
- Data is available in JSON, XML, or HTML format
- Available for free and all other paid accounts available for free and all other paid accounts.



```
{
    "coord":
        {
            "lon":13.39,
            "lat":45.97
        },
    "weather":[
        {
            "id":211,
            "main":"Thunderstorm",
            "description":"thunderstorm",
            "icon":"11n"
        }],
    "base":"stations",
    "main":
        {
            "temp":295.04,
            "pressure":1012,
            "humidity":83,
            "temp_min":289.82,
            "temp_max":300.15},
            "visibility":10000,
            "wind":
        {
            "speed":2.6,
            "deg":350
        },
    "clouds":
        {
            "all":75
        },
        "dt":1565292630,
    "sys":
        {
            "type":1,
            "id":6769,
            "message":0.0082,
            "country":"IT",
            "sunrise":1565236610,
            "sunset":1565288850},
            "timezone":7200,
            "id":3175985,
            "name":"Bolzano",
            "cod":200
}
```

Figure 3-7 weather Json file and values to be retrieved

The communication to the weather was via HTTP, which works as a request-response protocol between a client and a server. Two commonly used HTTP methods to request the servers are GET and POST.

When using the GET method, the data will be sent to the server as query parameters. These are appended to the URL as a key-value pair. We can see how data is passed to the server as a key-value pair. These values will be visible at the address bar. URL character length is limited, so you cannot use it if you are sending extensive data. GET is recommended to use for querying information from the server.

To implement the GET method we had to pass by different steps:

Create a new Java class in a new directory called “*WeatherHttpHandler*”

```
private static final String TAG = weatherHttpHandler.class.getSimpleName();
```

An HTTP request can be directly associated with an HTTP Handler or with an HTTP Handler factory object. An HTTP handler is a class that implements the HTTPHandler Interface and is responsible for returning the actual HTTP handler to serve the request. The HTTP handler is used to perform some preliminary tasks on the requested resource before passing it on to the handler. The internal class named Page WeatherHttpHandler represents a typical handler object. It sets out the name of the handler to be used and, if feasible, loads it all from the current assembly.

```
public weatherHttpHandler() {
    }
    public String makeServiceCall(String reqUrl) {
        String response = null;
        try {
            URL url = new URL(reqUrl);
            HttpURLConnection conn = (HttpURLConnection) url.openConnection();
            conn.setRequestMethod("GET");
            // read the response
            InputStream in = new BufferedInputStream(conn.getInputStream());
            response = convertStreamToString(in);
        } catch (MalformedURLException e) {
            Log.e(TAG, "MalformedURLException: " + e.getMessage());
        } catch (ProtocolException e) {
            Log.e(TAG, "ProtocolException: " + e.getMessage());
        } catch (IOException e) {
            Log.e(TAG, "IOException: " + e.getMessage());
        } catch (Exception e) {
            Log.e(TAG, "Exception: " + e.getMessage());
        }
        return response;
    }
    private String convertStreamToString(InputStream is) {
        BufferedReader reader = new BufferedReader(new InputStreamReader(is));
        StringBuilder sb = new StringBuilder();
        String line;
```

In the Java file “homeActivity” the method used to call the Json file and retrieve information is GET method.

```
protected Void doInBackground(Void... params) { weatherHttpHandler sh = new weatherHttpHandler();
```

Making a request to URL and getting response:

```
String jsonStr = sh.makeServiceCall(url);
        Log.e(TAG, "Response from url: " + jsonStr);
        if (jsonStr != null) try {
            JSONObject jsonObj = new JSONObject (jsonStr);
```

Getting JSON Array

```
             String coord = jsonObj.getString ("coord");
            JSONObject coordinat = new JSONObject (coord);
            Lat = coordinat.getString ("lat");
            lon = coordinat.getString ("lon");
```

Get System obj:

```
             String sys = jsonObj.getString ("sys");
            JSONObject system = new JSONObject (sys);
            country = system.getString ("country");
            city = jsonObj.getString ("name");
```

Get weather information:

```
            JSONArray jsonArray = jsonObj.getJSONArray ("weather");
            JSONObject jsonWeather = jsonArray.getJSONObject (0);
```

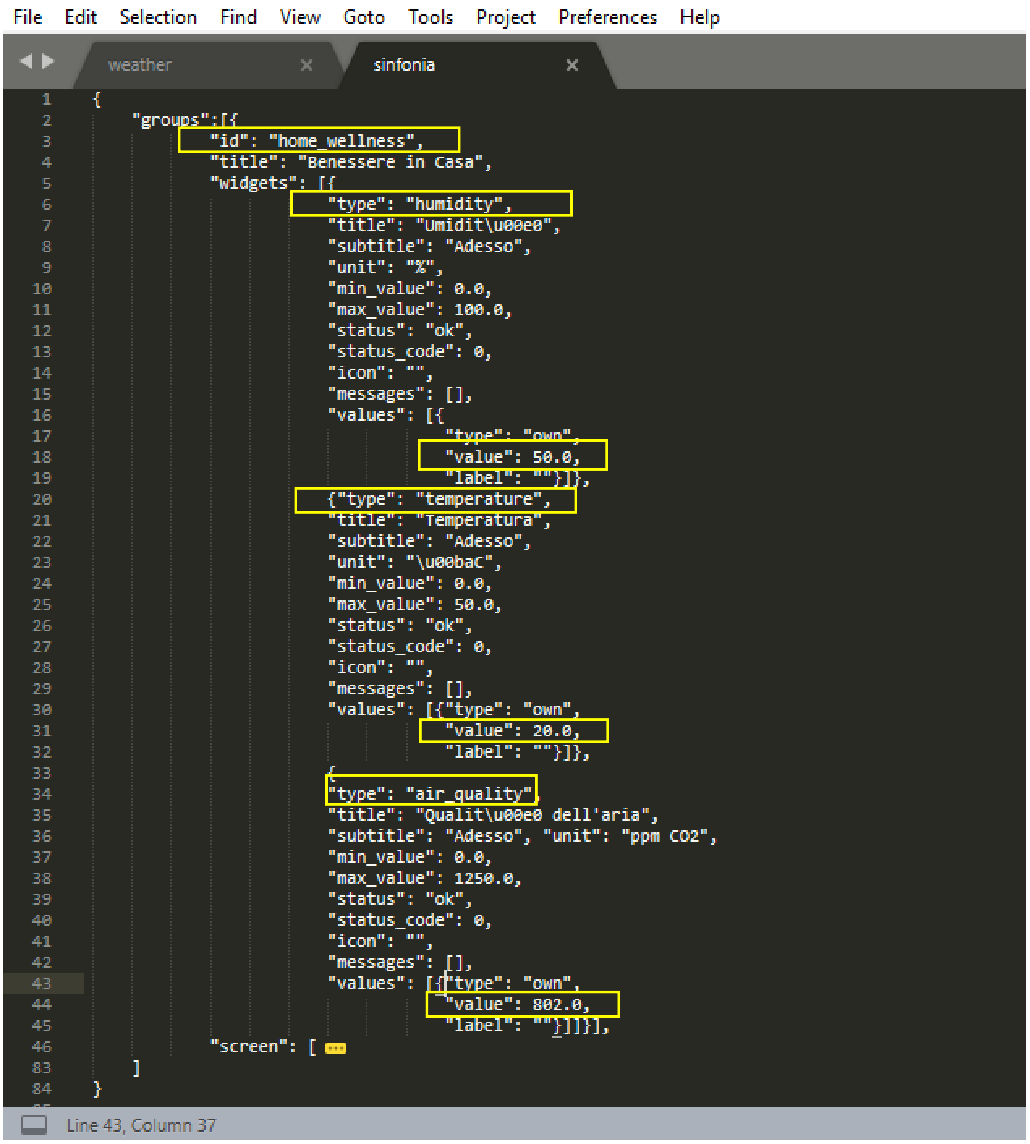
```
        Condition = jsonWeather.getString ("description");
        String Main = jsonObj.getString ("main");
        JSONObject main = new JSONObject (Main);
        temperature = main.getString ("temp");
        String windspeed = jsonObj.getString ("wind");
        JSONObject wind = new JSONObject (windspeed);
        speed = wind.getString("speed");
```

posting information in the phone interface:

```
protected void onPostExecute(Void aVoid) {
   super.onPostExecute (aVoid);
   float tempCelsuis = Float.parseFloat (temperature);
   tempCelsuis = Math.round (tempCelsuis) - 270 ;
   System.out.println (country+city);
   CityName.setText(city + "," + country);
   //CityName.setText(Lat + "," + lon);
   temp.setText(""+ tempCelsuis+ " °C");
  windSpeed.setText("Wind: " + speed + "m/s");
  lastupdt.setText("Condition: " + Condition );
```

3.3.2.4. Sinfonia API:

As the weather API the Sinfonia API sends current house information in JSON file this will make it easy for us to get the predefined values in the API, the GET method is always valid, and we will use it also passing by HttpHandler class.



```
{
    "groups":[{
        "id": "home_wellness",
        "title": "Benessere in Casa",
        "widgets": [{
                "type": "humidity",
                "title": "Umidit\u00e0",
                "subtitle": "Adesso",
                "unit": "%",
                "min_value": 0.0,
                "max_value": 100.0,
                "status": "ok",
                "status_code": 0,
                "icon": "",
                "messages": [],
                "values": [{
                        "type": "own",
                        "value": 50.0,
                        "label": ""}]},
                {"type": "temperature",
                "title": "Temperatura",
                "subtitle": "Adesso",
                "unit": "\u00baC",
                "min_value": 0.0,
                "max_value": 50.0,
                "status": "ok",
                "status_code": 0,
                "icon": "",
                "messages": [],
                "values": [{"type": "own",
                        "value": 20.0,
                        "label": ""}]},
                {
                "type": "air_quality",
                "title": "Qualit\u00e0 dell'aria",
                "subtitle": "Adesso", "unit": "ppm CO2",
                "min_value": 0.0,
                "max_value": 1250.0,
                "status": "ok",
                "status_code": 0,
                "icon": "",
                "messages": [],
                "values": [{"type": "own",
                        "value": 802.0,
                        "label": ""}]}]},
        "screen": [ ...
    ]
}
```

Figure 3-8 Sinfonia Json file and values to be retrieved.

3.3.2.5. Sinfonia HttpHandler:

To make a service call for the Sinfonia API, we are required to add new lines in HttpHandler class to authorize the application and start retrieving information from the Json file:

```
public String makeServiceCallSinfonia(String SinfUrl) {
    String response = null;
    try {
            String payload = "{\"id\":\"electrical_energy\",\"language\": \"it\"}";
        URL url = new URL(SinfUrl);
        Log.e(TAG, url.toString ());
        HttpURLConnection conn = (HttpURLConnection) url.openConnection();
 conn.setRequestMethod("POST");
 conn.setRequestProperty("Content-Type", "application/json");
 conn.setRequestProperty("Accept", "application/json");
 conn.setRequestProperty ("x-api-key", "write here your token");
    conn.setDoOutput(true);
    conn.getOutputStream().write(payload.getBytes());
```

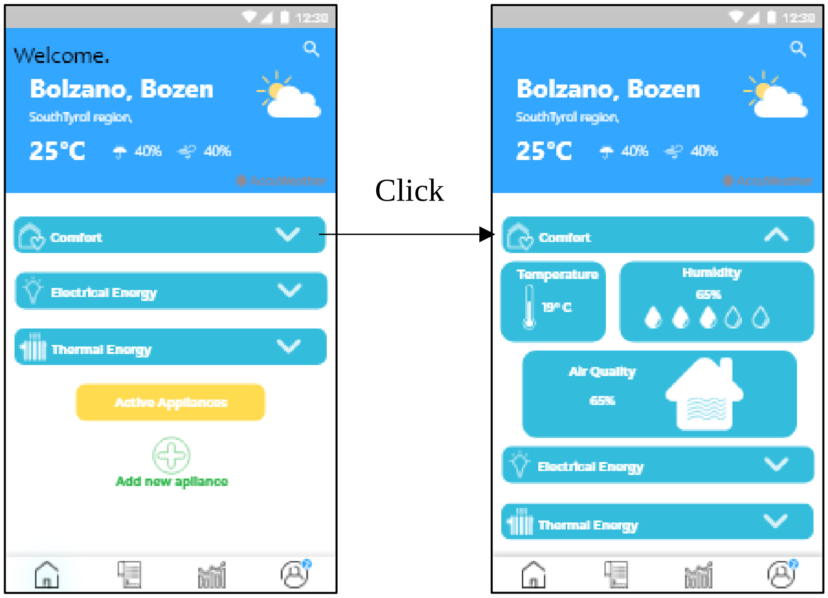
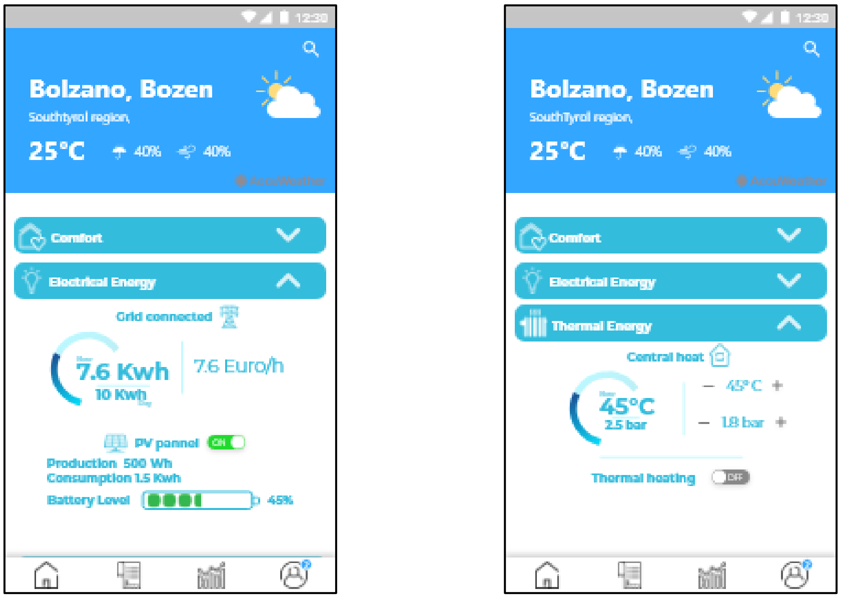


Figure 3-9 Visualization of Sinfonia predefined data in phone application.

3.3.2.6. Standalone system

In the home page, we have two more activities of electrical energy and thermal energy. These activities will have an added value of using renewable energy resources if available like the solar, wind. Using a smart meter to connect to the national grid and to the standalone system the application will be directly connected to this smart meter and the house occupant can also switch on and off the use of renewable energy at any time, also for thermal energy if the house occupant use a renewable energy like geothermal energy connected to a heat pump this will also be possible to control via the application.



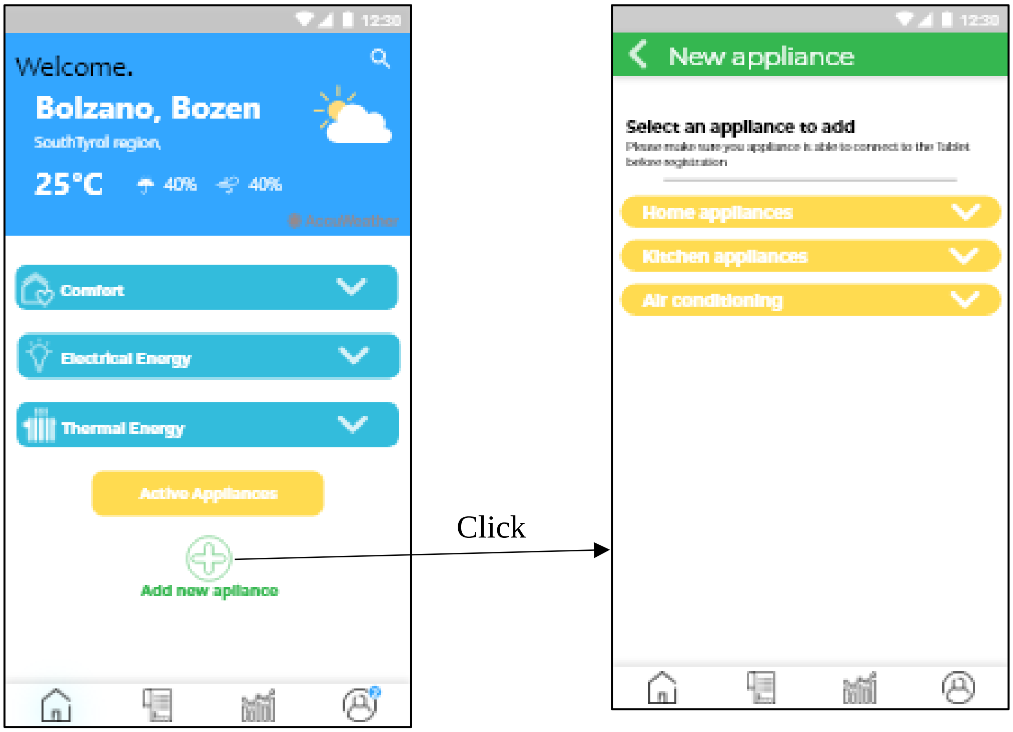
Electric Energy page

Thermal Energy page

Figure 3-10 switch on and off the standalone system

3.3.2.7. Add new appliances

The application feature adds new appliance has the same logic as adding new rooms of adding the new appliances in the different zones of the house the application will have a library of smart appliances that utilize modern computer and communications technology to make functions faster, cheaper and more energy efficient. The appliances can take advantage of an energy "smart grid," being implemented by utility companies nationwide. When the smart grid technology is finally implemented, refrigerators, toasters, dishwashers, and washing machines can tap into the smart grid power source. This feature is also used when setting up the application for the first time and also every time the user buys or bring a new appliance for the moment this feature is not well developed to be fully operational.



Select an appliance to add
Please make sure you appliance is able to connect to the Tablet before registration

Figure 3-11 Add new appliances

3.3.2.8. Get and Change condition in a specific zone

The application philosophy is to give control to the user as much as possible this will give more opportunities to the development of this feature, if the user can change the temperature of only a specific room this will have a positive impact of the overall energy efficiency and cost-effectiveness also will give more comfort to the house occupant.

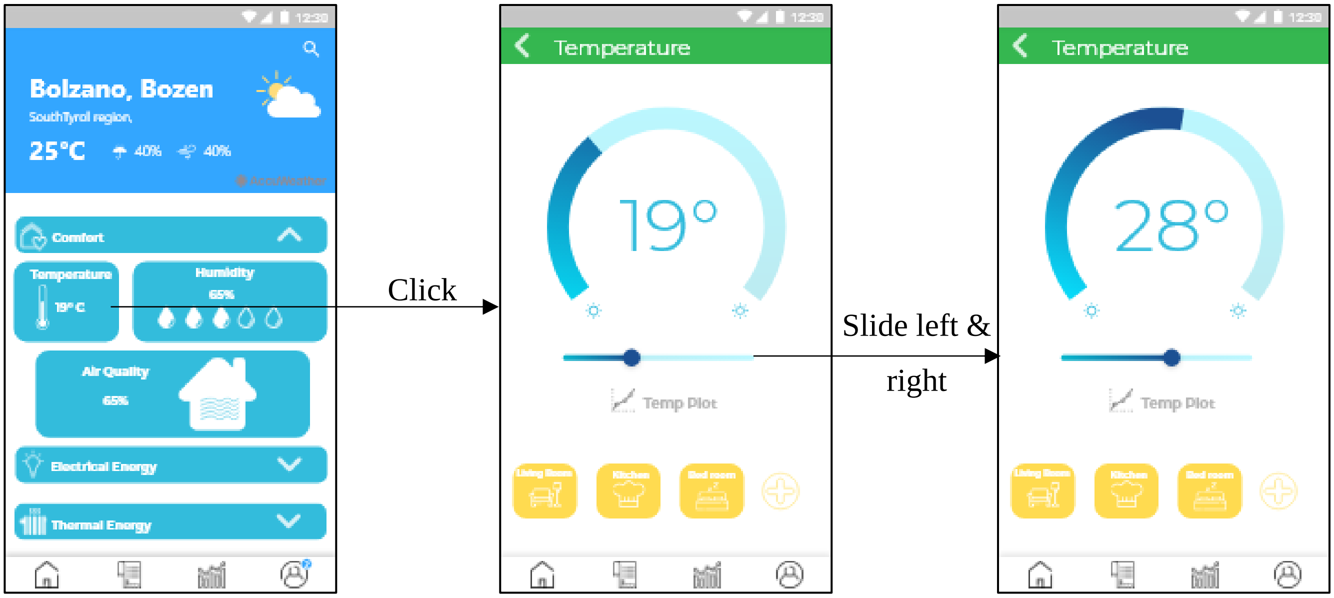


Figure 3-12 change zone conditions.

3.3.2.9. Add new rooms activity:

The application feature **add new room** was defined to be used when the user first starts the application and setting up component and appliances in the house: the rooms are considered as zones inside the house, one zone is active when sensors and appliances are active, and they are sending data to the phone; also this will give the user more power in when they change the house. For example, if the user buys a new house, he will quickly set up the rooms because it will be already defined in previous use.

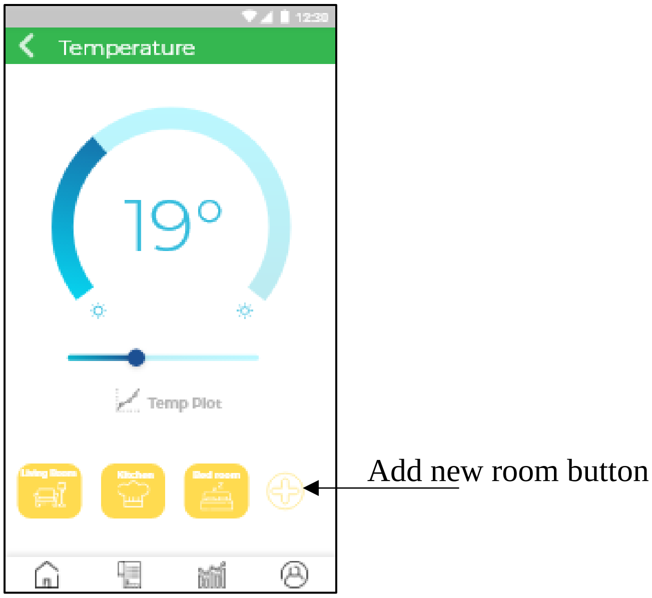


Figure 3-13 Add new room

3.3.2.10. Browse history

The house occupant will also have the possibility to **browse through the history** of energy consumption in a statistic way; this will give the opportunity to keep track of the overall energy flow of the house.

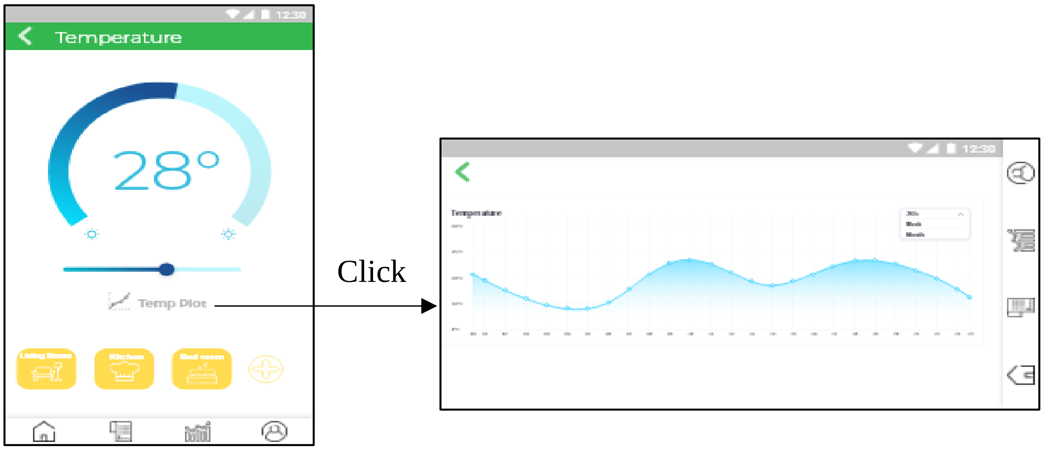


Figure 3-14 temperature statistic visualization.

3.3.2.11. Send notification

Notification will be sent to the user depending on the situation inside the house: for example, the temperature or if there is excessive energy consumption. the resident at any time may have problems related to the appliances or sensors or standalone system, the application will notify the errors in the system and if there is an anomaly somewhere. The phone application will also send a recommendation to the occupant of a feasible solution if there are problems in the system and give the suggestion of closest maintenance service depending on the issue.

3.3.2.12. Savings:

The Jua House application will be able to calculate the amount of money the house is consuming in the future development the calculation will depend on the tariffs the house occupant is using and the hours of the day the energy is consumed.

3.3.3. Creating HEART App:

During the internship, we had the opportunity to work with the Institute for Renewable Energy in their project the HEART project, and the dominant feature of the basic application is the flexibility of customization depending on the customer needs. The HEART application needed two main functionalities; the first is getting data from the sensor and actuating temperature inside the house, and the second is to calculate the cost of energy consumption.

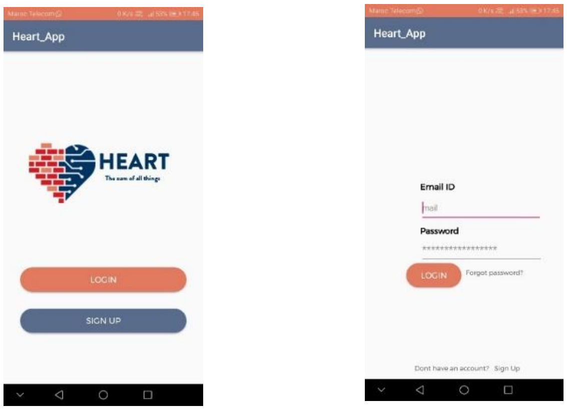


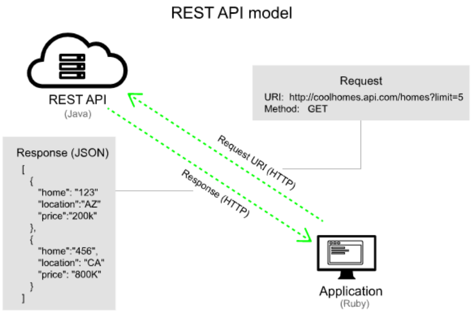
Figure 3-15 Heart application home page and authentication page

3.3.3.1. Creating temperature Actuation REST API:

a) Rest API:

REST (Representational State Transfer) is an architectural style for distributed hypermedia systems that leverages the architecture of the World Wide Web; usually uses JSON format for data interchange.

The web protocol is part of the nature of web services: they are agnostic languages and therefore, scalable across multiple platforms and systems. When documenting a REST API, it does not matter whether the API is built under Java, Ruby, Python, or some other language. Requests and answers are made via the HTTP protocol.



Request

URI: http://coolhomes.api.com/homes?limit=5
Method: GET

Response (JSON)

```
[
  {
    "home": "123"
    "location":"AZ"
    "price":"200k"
  },
  {
    "home":"456",
    "location": "CA"
    "price": "800K"
  }
]
```

Figure 3-16 Rest API work model.

b) Restful API:

RESTful is a way of writing services using the REST architectures. RESTful services expose the resources to identify the targets to interact with clients. Instead, the API reveals a functionality, an application, or a service that exists independently from the API definition. The common practice in API design:

1. Understanding enough of the essential details of the application for which an API is to be created, so that an informed decision can be made what functionality needs to be exposed, now it needs to be exposed, and what functionality can instead be left out.

2. Modeling this functionality an API that addresses all use cases that come up in the real world, following the RESTful principles as close as possible. There are three distinct components involved in RESTful API design: the application, the API code, and the client. The image above illustrates how these three components interact.

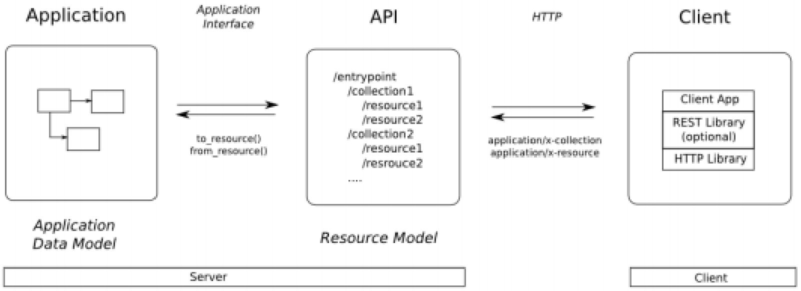


Figure 3-16 RESTful API architecture model

One of the main concepts of REST API is a resource. The resources are the fundamental building blocks of web-based systems. A resource is anything exposed to the Web, e.g., a document, a video clip, a device, etc. The characteristics of a resource are:

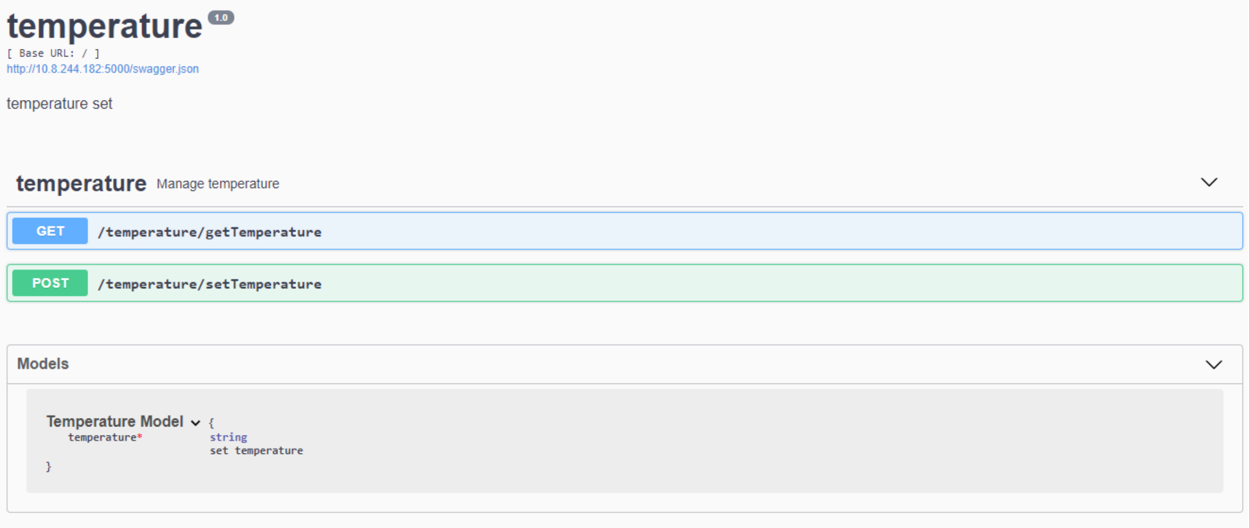
- A resource could be a set of objects or an individual object.
- A resource is identified by URI (Uniform Resource Identifier). The relationship between resources and URIs is one-to-many: a URI identifies only one resource, but a resource may have more than one URI.
- A resource is manipulated through CRUD (Create, Read, Update, Delete) operations, which are usually mapped to HTTP methods/verbs POST, GET, PUT, DELETE correspondingly.
- A REST API endpoint is defined by a combination of a resource URI and an HTTP GET or POST that manipulates it.

c) Swagger environment:

The OpenAPI specification provides a formal way of describing the REST API and includes all the reference sections mentioned in the previous section, Documenting API endpoints. Display frameworks such as Swagger UI can parse the OpenAPI specification and generate interactive documentation that lets users try out endpoints while learning about the API.

With OpenAPI, instead of XML, you have a set of JSON objects, with a specific schema that defines their naming, order, and contents. By describing the API in a standard format, publishing tools can programmatically parse the information about the API and display each component in a stylized, interactive display.

The API was developed using Python and linked to the OpenAPI using Swagger UI for better documentation and understanding of the logic behind the working of the actuation API.



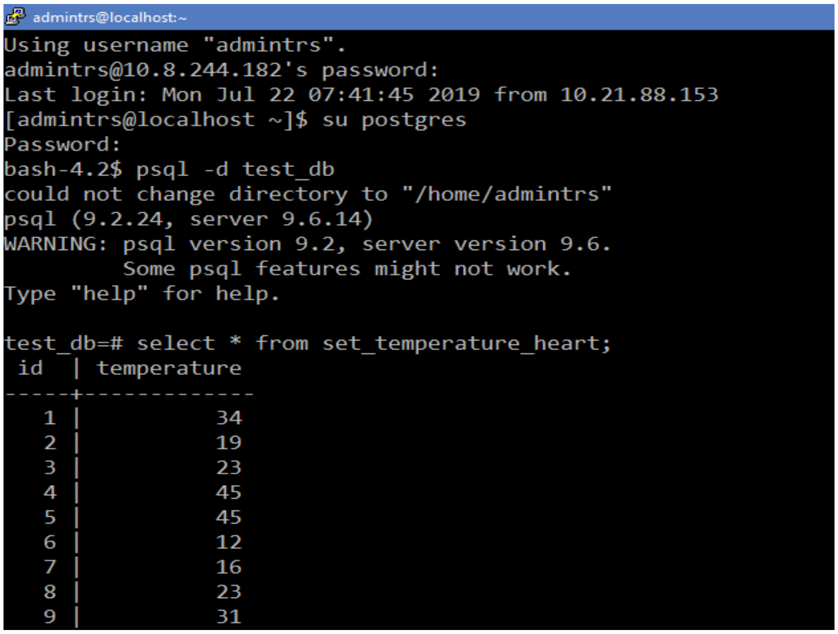
temperature 1.0
[Base URL: /]
http://10.8.244.182:5000/swagger.json
temperature set
temperature Manage temperature
GET
/temperature/getTemperature
POST
/temperature/setTemperature
Models

```
Temperature Model {
    temperature*        string
                        set temperature
}
```

Figure 3-17 Temperature API Displayed in Swagger UI.

d) PostgreSQL database:

It is an open-source database management system (DBMS) and it stores data in tables, the aim of creating a virtual database is to simulate the house, the data will be stored in a table to keep track on the history of the changes and also to retrieve the list changes every time we call to get the value.



```
Using username "admintrs".
admintrs@10.8.244.182's password:
Last login: Mon Jul 22 07:41:45 2019 from 10.21.88.153
[admintrs@localhost ~]$ su postgres
Password:
bash-4.2$ psql -d test_db
could not change directory to "/home/admintrs"
psql (9.2.24, server 9.6.14)
WARNING: psql version 9.2, server version 9.6.
         Some psql features might not work.
Type "help" for help.

test_db=# select * from set_temperature_heart;
 id | temperature
----+-------------
  1 |          34
  2 |          19
  3 |          23
  4 |          45
  5 |          45
  6 |          12
  7 |          16
  8 |          23
  9 |          31
```

Figure 3-18 PostgreSQL database.

e) Installation pipnev:

We firstly created a virtual environment and installing pipenv, which help to manage and create packages and ensure environment support in a single tool.

```
pipenv install flask
pipenv install flask-restplus
```

f) Flask

Flask is a lightweight WSGI (Web Service Get Interface) web application framework. The Flask application exposure of Python functions as APIs. Using Flask-RESTPlus will improve upon the application. It allows us to not only define REST APIs but also brings in Swagger UI for all the APIs.

After importing Flask from flask library, I am also importing the API to begin app definition and resources which are received as a parameter in the various classes.

```
From flask import Flask

from flask_restplus import Api, Resource
```

g) Connecting to the Database:

Psycopg is a PostgreSQL database adapter for the Python programming language. Its main features are the complete implementation of the Python DB API 2.0 specification we have installed it on the virtual machine

```
pip install psycopg2
```

In the database python file, I have imported psycopg2 in an independent python file.

```
import psycopg2
```

The connect command establishes a connection to the PostgreSQL server.

```
def connectdb():
```

specifying host and database name and credentials (username and password)

```
        con = psycopg2.connect (
                host='**. **.***.***',

                database='test_db',

                user='********',

                password='*********')

        return con
```

using inbuilt function in Python that inserts a given element at a given index in a list. Once we have a Connection, we can create a Cursor object and call its execute() method to perform SQL commands:

```
def insertTemp(insValue,con):

        cur = con.cursor()

        query = "insert into set_temperature_heart (temperature) values ("+str(insValue)+")"

        cur.execute(query)

        #commit

        con.commit()
```

h) Building temperature API:

We start by defining the application as a flask app using the method Flask I defined the application as a flask application using the method Flask() which sets the name using __name__.

```
flask_app = Flask(__name__)
```

As is was stated before Flask-RESTPlus automatically documents the API in the Swagger UI, we put a namespace that will appear under the given heading in Swagger.

```
pp = Api(app = flask_app,

                version = "1.0",

                title = "temperature",

                description = "temperature set")

name_space = app.namespace('temperature', description='Manage temperature')
```

We create a temperature get class resource, and this method will get the last value from the database table and send it to the phone application.

```
Class TempGetclass(Resource):

        def get(self):

                try:

                        con = testDB.connectdb()

                        lastValue = testDB.selectTemp(con)[0]

                        print (lastValue)

                        #temperature = list_of_temperature[value]
```

The main job for this API is the actuation of the temperature inside the house, the method that we use to change the value of temperature is POST method which sends a new value to the actuator each time it is clicked on the phone.

```
class TempPostclass(Resource):

        @app.expect(model)

        #@app.route('/post', methods=['POST'])
```

i) Retrofit library:

Retrofit is an HTTP client for Android and Java. It uses OKHttp by default for network operations. What makes it unique is that with Retrofit you do not need to worry about parsing the response – meaning de-serialization is handled. In the background itself. We need to configure any convertor library (GSON, Jackson, etc.) and the Retrofit will parse the information to apply retrofit method we need to pass by many steps:

a) Importing Retrofit dependency:

First of all, we need to add a retrofit dependency in the *build.gradle* file in order to import the library to the project.

```
implementation 'com.squareup.retrofit2:retrofit:2.3.0'
```

Since we are using JSON file in the temperature API, we also need to import retrofit JSON library so the retrofit method will be able to read the JSON file generated from the API.

```
implementation 'com.squareup.retrofit2:converter-gson:2.3.0'
```

We create ta Java interference and create one method it will return a call object which says retrofit and for the type repairs list and then post cause this is what you want to get back from this call a list of posts which is a JSON array of post-JSON objects that we saw earlier. We this method a name get to post because in a Java interface we don't provide a method body we just declared these methods and whatever later implements interface has to provide a body for these methods and this is what retrofitted is taking care of it will auto-generate unnecessary record we need to get our list of posts back into tell retrofit what to do we need to annotate this method, and this is a post request which means that we want to post data to the server.

```
import retrofit2.Call;
import retrofit2.http.Body;
import retrofit2.http.POST;

public interface tempPostApi {

    @POST("/temperature/setTemperature")
    Call<temperaturePost> createTemperaturePost (@Body temperaturePost
temperature);
```

Then we create a new java class ApiManager, this class will have the base URL as a static variable and will also provide the APIService interface by with a getAPIService() static method to the rest of our application.

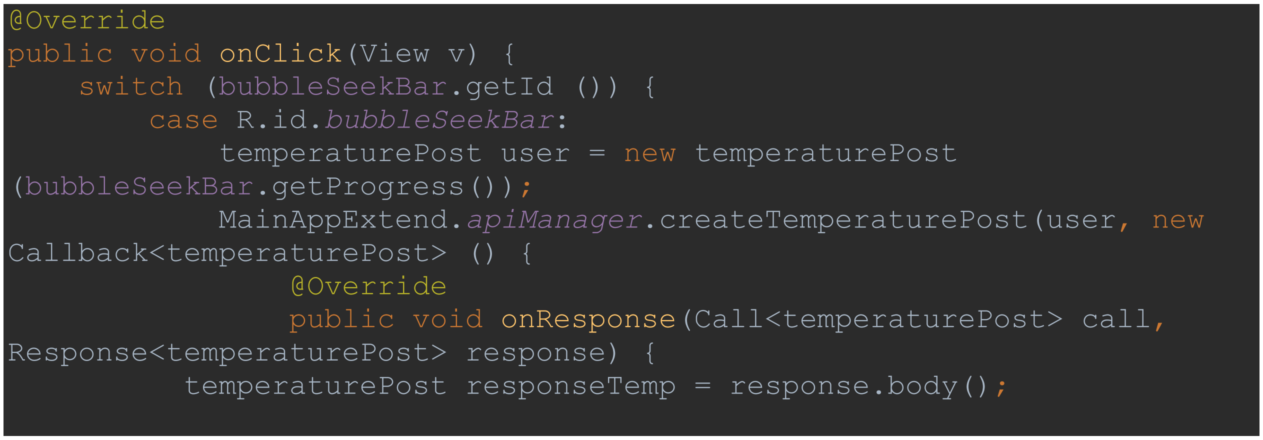
```
    private ApiManager() {

        Retrofit retrofit = new Retrofit.Builder()
                .baseUrl("http://10.8.244.182:5000/")
                .addConverterFactory(GsonConverterFactory.create())
                .build();
```

and then we create the temperature post method inside the ApiManager:

```
public void createTemperaturePost(temperaturePost temperature,
Callback<temperaturePost> callback) {
    Call<temperaturePost> userCall =
```

on the temperature activity, we used a slide progress bar to increase and decrease the value and to send the set value we had to call the retrofit method inside the progress bar class.



```
@Override
public void onClick(View v) {
    switch (bubbleSeekBar.getId ()) {
        case R.id.bubbleSeekBar:
            temperaturePost user = new temperaturePost
(bubbleSeekBar.getProgress());
            MainAppExtend.apiManager.createTemperaturePost(user, new
Callback<temperaturePost> () {
                @Override
                public void onResponse(Call<temperaturePost> call,
Response<temperaturePost> response) {
          temperaturePost responseTemp = response.body();
```

b) Temperature activity:

This is the activity where we will fetch the house temperature value and also actuate the temperature using HTTP connection protocol to the database, which as it was stated before is simulating the house environment.

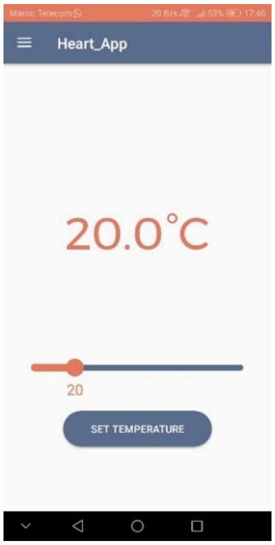


Figure 3-19 Temperature activity: actuation slide progress bar

3.4. Conclusion:

The purpose of this chapter is to show the work that has been done during my internship at the Center for Sensing Solutions, at Eurac Research, the development of the phone application was based on three different projects Sinfonia, Jua house, HEART project. Those three projects have the same determination, which is home automation following the basics of the Internet of things and demanded the knowledge of many programming interfaces and language like Android, Adobe XD, Java, and Python, and the phone application was designed on a flexible way where it can be customized based on the needs.

Chapter 4: Outlook and General conclusion

Content

4.1. Introduction

The application will be deployed in Gitlab for future development, and the aim is to create an underlying platform where developers will have the opportunity to enhance the utility of the app for specific practices. We will also discuss the outlook of the phone application and how we can bring this one to be a reference in the Internet of Things world.

4.2. Security and Privacy

Malicious cyber activity for organizations around the world is a growing challenge. It ranges from simple online fraud such as email scams, websites or chat rooms – to sophisticated cyber espionage and calculated cybercrime, used to steal secrets and other digitally stored information on systems and networks.

The privacy of the home application is essential in this field when connecting to the internet the phone, and the user is uncovered to the different threat from hackers that can lead to a tragic situation. Someone with bad intentions can start controlling all the things connected to the house, even the most sensitive ones, and it can easily access the family's cameras or open their front door. Secret life will be exposed to the open world.

The concept of offering safety in smart home settings is based on maintaining six vital characteristics: confidentiality, integrity, authentication, permission, non-repudiation, and availability. Confidentiality, integrity, authorization, non-repudiation, and availability play significant roles in ensuring of Smart Home internal network security. However, authentication can be considered as the first step in the pyramid of a security mechanism. One of the suggestions is to use two-step authentication, which guarantees that you can only use a trusted device to access your account. If someone attempts to log in to a current device, a code to your phone will automatically be sent to the trusted device. Without this code, even with a valid password, a hacker cannot access your account.

4.3. Integration of Smart Home Devices

As it was discussed before the communication protocol that has been used to connect from the phone application to the API and from the API to the database (which simulate the house) was

HTTP protocol. We cannot oblige a user to use specific smart things in the house, he is free to choose what he likes to buy and to bring into the house, therefore he can encounter a communication problem when he install and start using appliances from different companies and brands using different unique standard and communication protocols; so the way forward in the future development and enhancement of the application in this fields is to build a communication protocol library dedicated to the different brands around the world. In this way, on the add new appliance activity when the user chose the type of the appliance to add, he also will have a list of all the existing brands in the library: in two clicks, he can control the new appliance.

4.4. A Greater Role for Artificial Intelligence

The application is built from scratch this will allow the developers to be more flexible and more creative in the programming and design thinking of the future features that can help the user to be more comfortable inside the house, one of the trending innovation in the IT world is artificial intelligence involving in this world will open as many opportunities as we can imagine, the phone application with time will learn the behavior of the user and start suggesting the setups based on previous commands. The home automation system powered by artificial intelligence will give homeowners peace of mind as it will take care of comfort, home security, and energy efficiency. The system has to overcome barriers such as the original investment price of such a smart system that comes with various appliances in order to obtain widespread adoption. Most importantly, as a single attempt to hack into such a centralized system can result in privacy invasion and loss of sensitive information, these systems should be highly shielded, as stated previously.

4.5. Factors affecting phone development

The future development of the application will need to take in consideration many factors plus to the features and services that have been deployed to have a powerful application that can be a reference in the home automation and Internet of Things world. Those factors are:

A. Wearable devices:

Smart wearables show an upcoming change in computing and the transition from necessary use to a smart wearable’s devices. This opens fresh possibilities for suppliers, developers of apps, and manufacturers of accessories. The smartphone becomes the hub of a network of the personal area.

Wearable smartphone-connected devices will, therefore, affect the next generation of strategies for mobile application development.

B. Innovative Mobile User Experience Design:

The development of the application will always take in consideration user reviews and design thinking technics, it is essential for user experience to show information and content effectively on the mobile user interface. Also, designers are always considered to enhance the application that can address mobile problems, such as partial user attention and disruption. The application needs to exploit new characteristics such as interactive content layers, circular design patterns, cards, and content manipulation. These characteristics generate an "augmented reality" by enabling users to communicate in more detail with the content.

C. Application Performance Management (APM):

There are two factors which lead to performance bottlenecks in-app testing and the non-deterministic nature of mobile networks. However, mobile metrics and monitoring tools are collectively known as Application Performance Management (APM) has improved the testing and quality assurance. APM provides visibility into app behavior, delivers statistics about which devices and OSs are adopted, and monitors user behavior to determine which app features are being successfully exploited.

4.6. Flexibility and adaptability:

This is a powerful feature that was already described, but it will have a crucial role in future development. This will allow the application to take different shapes and to have more or fewer services depending on the customer needs it was highlighted before when the application was scaled up for the Jua house projected and scaled down for the HEART project because they needed only to change temperature.

4.7. Operation systems:

If we would like to tackle as many costumers in the market as we can it will be necessary to have a phone application that works on every operating system, we have designed to the first basic smart house phone application under Java language and Android operation system. However, nowadays we can notice the introduction of new language Kotlin launched by Google lately under Android Studio and also, we can start thinking to develop the application for Apple brand users the future development will be to create an application for IOS operating system.

4.8. General conclusion

This work was dedicated to developing a phone application towards the concept of internet of things. Initializing from Sinfonia project, which is a web application used for data acquisition for smart houses, the project objective was to create a phone application under Android as a starting operation system. The phone application could interact with the different appliances and sensors inside the house using API associated with it. These goals have been successfully achieved on completion of this project. The project was tested on both Android emulator and Android mobile device. The application ran smoothly, and the UI components responded as expected. This thesis projects some of the fundamentals of Android database application. The work was also focused on creating a necessary phone application which will be open for future development the information and code were saved in a flexible format, as it is a relational database, the demonstrations of this was by creating one phone application for three projects Jua House, Sinfonia, HEART. The application proved to be flexible and able to be scaled up or down depending on the different scenarios a house can have, and more possibilities of new applications are enormous and more services, and features can be added.

References

1. Anna , F., Vikash , k., & Slobodanka , D. (n.d.). End-user interfaces for energy-efficient semantically enabled smart homes. *Energy Efficiency*.

2. Qusay , F., Sajjad , A., & Atta , u. (2018). *Internet of Things: Challenges, Advances, and Applications.* CRC press.

3. (2013, 08 02). Retrieved from pantechsolutions: https://www.pantechsolutions.net/blog/machine-to-machine-architecture/

4. Anisha Gupta., R. C. (2017). Scalability in the Internet of things: Features, Techniques, and Research challenges. *International Journal of Computational Intelligence Research* , 1617-1627.

5. Berry, E. b., & Elise, V. d. (2014). *Human-Centered Design and Smart Homes: How to Study and Design for the Home Experience?* Switzerland.

6. D. Surie, O. L. (2008). Wireless sensor networking of everyday objects in a smart home environment. *International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, (pp. 189–194).

7. David Boswarthick, O. E. (2012). *M2M Communications_ A Systems Approach-*. Wiley .

8. Dr. Ovidiu Vermesan, D. P. (2011). IoT cluster strategic research agenda . 2.

9. *Energy consumption and use by households*. (2019, 06 20). Retrieved from ec.europa.eu: https://ec.europa.eu/eurostat/web/products-eurostat-news/-/DDN-20190620-1

10. Foundation, I. s. (2018). *IoT Security Architecture and Policy for the Home - a Hub Based Approach.* IoT Security Foundation.

11. Fremantle, P. (2015, 10). *A Reference Architecture For The Internet of Things*. Retrieved from https://wso2.com: https://wso2.com/whitepapers/a-reference-architecture-for-the-internet-of-things/

12. (2010). *Internet 3.0: The Internet of Things, Analysys Mason*, Singapore.

13. *Internet of Things (IoT)* . (2017, 02 06). Retrieved from Welcome to GS1: https://www.gs1bg.org/bg/2017/02/06/-internet-things-iot/

14. Juhani Latvakoski, *. M., J. M., & Granqvist, N. (2014). *Towards Horizontal Architecture for Autonomic M2M Service Networks.*

15. Kyas, O. (2017). *How To Smart Home: A Step by Step Guide for Smart Homes & Building Automation (5th Edition).* Key Concept Press.

16. Lal Das, P., Beisswenger, S. C., Mangalam, S., Yuce, M. R., & Lukac, M. (2017). *Internet of things : the new government to business platform - a review of opportunities, practices, and challenges.* The World Region: World Bank.

17. Lal Das, P., Beisswenger, S., Mangalam, S., Yuce, M., & Lukac, M. (2017). *Internet of things : the new government to business platform - a review of opportunities, practices, and challenges.* world bank.

18. Latvakoski, J., Iivari, A., Vitic, P., Jubeh, B., laya, M., Monteil, T., . . . Väisänen. (2013). A survey on autonomic M2M service networks. Computers.

19. Oluwatosin , A., & Mohamed , O. (2018). Machine-to-Machine Communication: An Overview of Opportunities. *Elsevier*, 255-276.

Quarterly, M. (2010). The internet of things.

20. *Sinfonia*. (n.d.). Retrieved from Sinfonia eurac: http://www.sinfonia-smartcities.eu/en/project

21. Union, I. T. (2015, 11). *Smart, sustainable cities — a road map*. Retrieved from ITU News: https://itunews.itu.int/en/NotePrint.aspx?Note=5217