Technical Report - **Product specification**

SmartHomes – Final Iteration

Course:	IES - Introdução à Engenharia de Software
Date:	Aveiro, 19/10/2023
Students:	68264: Bruno Lopes 108712: Diogo Falcão 108011: Fábio Matias 107927: Rúben Garrido
Project abstract:	An app for monitoring the client's home resources, such as electricity and water.

Table of contents:

1 Introduction

In the last iteration of the IES final project, we'll firstly present our product, then the architecture used and finally the core user cases and usage scenarios. With this deliverable, we will also explain the organization of the software project, architecture and build the software as a team and use corporate solutions and tools related to the already presented SmartHomes theme.

2 Product concept

Vision statement

The SmartHomes application is designed to address the problem of efficient and sustainable home resources management. Therefore, it provides a solution for homeowners to monitor and control their electricity, water and devices, knowing its costs and environment impact. This solution is made for it to work in all of any devices plugged into the house (unlike HomeKit), as it is not dependent on the devices but on the physical interfaces they are plugged into.

This kind of set up also prevents complex programming skills or automation skills from the user (like it may happen in other cases like Google Home). Adding various functions from other mainstream applications, such as smart device control and control and info about household resources. Instead of merely controlling devices, we gain insights into how to intelligently assist the environment and your wallet.

Personas and Scenarios

Sara Mathews

- **Age**: 39 years old
- **Job**: Doctor
- Sara is concerned about the environmental impact of her pledge. Therefore, she wants to carry out the same values to her home. As she continues to embark on her eco-friendly journey, she seeks innovative ways to reduce her carbon footprint and strive for a greener, more sustainable future for all.
- **Motivation**: Be able to combine both eco-consciousness and easiness, by using a web app to check real-time info (either from the grid or from home-produced sources, like photovoltaic panels and wind turbines), keep clean-energy levels high, and get to know when her carbon footprint is higher from any place with internet connection.

Peter Williams

- **Age**: 28 years old
- **Job**: Software Engineer
- Peter is passionate about tech and consequently likes home automations. With a deep-seated passion for both technology and sustainability, he has equipped his home with a range of smart appliances and clean energy solutions. He can effortlessly monitor and manage the functioning of these devices, optimizing their performance for energy efficiency. This level of control not only brings convenience to his daily life but also reinforces his commitment to a greener, more sustainable future.
- **Motivation**: Be able to manage his home, by using, such as being notified about the grid electricity export, tracking energy consumption, or remotely managing his solar panels, he relishes the power of connectivity in his quest for a smarter and eco-friendly home.

Anna Franklyn

- **Age**: 34 years old
- **Job**: Restaurant Owner
- Anna cares about the environment and takes small steps towards a cleaner and more sustainable future. However, despite Anna's eco-friendly awareness, her main goal is keeping house costs as low as possible. By reducing them, she can get a perfect balance between saving money, energy and water resources.
- **Motivation**: Be able to keep track of energy and water consumption-related costs and reduce them as much as possible by using an app that gives her insights and ways to control her devices at home.

John Lennon

- **Age**: 56 years old
- **Job Description**: Lawyer
- John Lennon is a hardworking individual who values financial stability and needs to have control over his household appliances based on energy consumption. He has various homes in different parts of Portugal due to the typology of his work. His determination is driven by the substantial cost of his daily electricity bills.
- **Motivation** - John believes that having control over his appliances and information about local electricity costs will help him reduce expenses and save money. His ultimate goal is to optimize energy usage and lower his electricity costs from all of his houses. He also wants to be notified about water or electricity being too high on its homes.

Product requirements (User stories)

Epics

- Homes
- House divisions (bedroom, WC, living room, etc.)
- Smart Devices
- Energy management

Core stories

- As Anna, I want to register several houses and their electrical system in one app.
- As John, I need to control every bit of water used in a drought in its region.
- As Peter, I would like to oversee electronic devices connected to the house grid.
- As Ana, I would much like to get notified about the percentage of the grid energy that can supply the house.
- As John, I want to turn off electronic devices with just a few taps through the app.
- As Petter, I need to have a place where I can get a summary of all my electricity and water related costs.

3 Architecture notebook

Key requirements and constrains

As expected, we adopted a multi-layer architecture for our SmartHomes project. The system is not driven by complex deployment concerns, once its main purpose is to display through a client-server model, data from sensors built in from the home(s) renewable energy fonts, water resources and other devices connected to the house(s)'s electrical system. Thus, the system ingests a big amount of data streams (data from APIs, using only websockets). This happens using python functions generating random values, who export data by JSON format.

We used a heterogeneous database, consisting of PostgreSQL, a relational database, for storing other data bedsides the data streams such as the client's and house(s)'s info and we'll be also using a timeseries database - influxDB, due to the storing of complex data structures that can be quickly retrieved and manipulated.

For the message queues, we used RabbitMQ for the delivering of messages from the data sources to the backend. If it happens, the system can occasionally lose some data and the application will not be compromised. Every customer must have the most recent data possible in their application home app. Therefore, the data should be updated periodically. For this to work, access requires always internet access. The system should protect every user's data and be able to deal with high volumes of data. Then, we created one websocket for each house and water, electricity, environment, costs and devices channels for each websocket.

Architectural view

The SmartHomes project adopts a multi-layer architecture to effectively handle the ingestion, processing, and presentation of data from various sources within a smart home environment. The architecture comprises several distinct layers, each with specific responsibilities, ensuring a modular and scalable design.

- **Data Sources:** This layer encompasses the various sensors and devices that generate data streams, including renewable energy sources, water resources, and other smart home appliances. It utilizes Python functions to generate random data values and export them in JSON format.

- **Message Queue:** RabbitMQ serves as the message queue, acting as an intermediary between the data sources and the backend processing layer. Its buffer sand distributes data messages from the sources to the backend components effectively.

- **Backend Processing:** The backend processing layer handles the ingestion, processing, and storage of data streams received from the message queue.

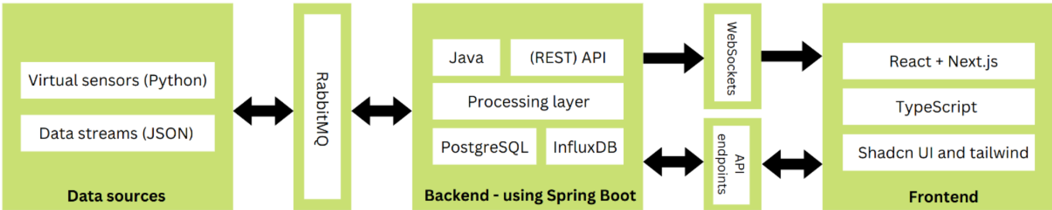
- **Database:** A heterogeneous database consisting of PostgreSQL and InfluxDB, a time-series database, are employed to store different types of data. The Postgre SQL database stores static information such as client and house details, while the InfluxDB time-series database handles the complex data structures and rapid retrieval requirements of data streams.

- **API:** The API layer exposes endpoints for retrieving and manipulating data, adhering to the REST architectural style. It utilizes JSON payloads for data exchange and it is connected to the frontend using WebSocket's, as it was mentioned earlier. The backend is implementing the Spring Framework.

- **Frontend:** The frontend is responsible for presenting data to the user through a user-friendly interface. It is built using React, TypeScript, and PNPM, with Next.js as the build tool and shadcn/ui and Tailwind CSS for styling.

Module interactions

The modules within the SmartHomes architecture interact closely to facilitate the seamless flow of data and achieve the desired functionality.



Data sources generate data streams in JSON format and transmit them through APIs. The message queue (RabbitMQ) receives and buffers data messages from the data sources. The backend processing layer consumes data messages from the message queue and processes them using Java functions. Processed data is stored in the heterogeneous database, with static information going to the PostgreSQL database and the Influx DB time-series data going to the time-series database. The API layer interacts with the backend to retrieve and manipulate data, providing access to the stored information. The frontend communicates with the API through the WebSocket to fetch and display data to the user through a user-friendly interface.

4 Information perspective

5 References and resources

- MDPI. (2023). Geriatric Helper: An mHealth Application to Support Comprehensive Geriatric Assessment. https://www.mdpi.com/1424-8220/18/4/1285