Advanced User Interfaces

M.Sc. Computer Science & Engineering

Design Document

Authors:

Scherini Samuele (10674683) samuele.scherini@mail.polimi.it
Sironi Alessandro (10680296) alessandro1.sironi@mail.polimi.it
Villa Ismaele (10663417) ismaele.villa@mail.polimi.it
Andersen Adam (10966718) adammartin.andersen@mail.polimi.it

2023/2024

Abstract This report presents a comprehensive overview of a novel web application designed to enhance smart home automation and promote energy-efficient practices. The project's goal was to develop a user-friendly, efficient, and innovative solution for automating routine tasks in smart homes, emphasizing green energy utilization. The methodology involved a user-centric approach in designing the interface and seamless integration with HomeAssistant for robust performance. Key outcomes included the successful implementation of energy-efficient practices, positive user feedback, and significant potential for market impact in the smart home automation sector. This work showcases the application's ability to blend technology and sustainability, paving the way for future advancements in smart home management.

1 The Team

	Scherini Samuele	Computer Science & Engineering
	Sironi Alessandro	Computer Science & Engineering
	Villa Ismaele	Computer Science & Engineering
	Andersen Adam	Computer Science & Engineering

1.1 Member Contribution

Samuele Scherini	FrontEnd development, HomeAssistant setup, UML, documentation.
Alessandro Sironi	Azure and Supabase setup, BackEnd development, video, documentation.
Ismaele Villa	BackEnd development, presentation, scenarios, documentation.
Adam Andersen	FrontEnd development, mockups, documentation.

Contents

2 Executive Summary

This report outlines the development of a cutting-edge web application designed to revolutionize smart home automation and energy efficiency. The project's primary goal was to create a user-friendly platform that facilitates energy-efficient practices and automates routine tasks in a smart home environment. Key features include a focus on green energy, seamless integration with HomeAssistant, and an intuitive interface for managing smart routines. In order to do this, we created a chat-bot base application that uses *GPT3.5* functionalities directly connected with the Home Assistant framework. The interaction with the user aims at retrieving information about personal habits and generate correct JSON code to create the automation routine on the Home Assistant platform. The development process was meticulously planned and executed in distinct phases: conceptualization, design, and implementation. During the conceptualization phase, the team identified essential features and user requirements. The design phase focused on creating a user-centric interface, while the implementation phase involved rigorous backend development to ensure robust performance and seamless integration with existing smart home systems. The report details the challenges encountered and the solutions implemented, emphasizing the application's potential to make a significant impact in the smart home market. Its innovative approach to energy efficiency and routine automation sets it apart in the industry, offering a unique combination of convenience and sustainability. The project's success is evidenced by its positive reception in user evaluations, indicating strong market potential and user interest. This application represents a significant advancement in smart home technology, offering practical solutions for energy management and automation while maintaining a strong user experience. Its development is a testament to the potential of technology in enhancing daily living and contributing to a more sustainable future.

3 Workflow of project activities

In this chapter, we delve into the workflow of our project, outlining the steps and the approach that guided our development process.

3.1 Conceptualization

In the first phase, our team conceptualized the idea of the product, taking into the account the requirements and deciding which aspects of it to develop. Key to our initial design phase was a deep understanding of our target users' needs and preferences. The idea was to create a web application that not only advises users on energy-efficient practices but also automates their smart home routines in a user-friendly manner, using natural language and empowering the chatbot with the awareness of the user's appliances.

3.2 Core Features

The core features of our product were defined through brainstorming sessions and collaborative discussions. These features included:

- **Green energy orientation**: to guide users towards more sustainable energy usage.
- **Smart Routine Automation**: to enable users to automate their home devices in an energy-efficient manner.
- **Seamless integration with HomeAssistant**: to ensure compatibility with existing smart home setups.

For the architectural design, we decided early on to split our development efforts into two main areas: frontend and backend. This separation allowed for specialized focus on user interface design and server-side functionalities, respectively.

3.3 UI/UX Design, Use-Case scenarios and UNG model

The team collaborated in drawing the interface of the webapp, creating wireframes and mockups, to visualize the application's interface and to settle on a final design. We then wrote different scenarios, regarding the two types of users: light and power user. Also, the UNG model was developed, illustrating the system's goals, the needs, the context, the constraints and the stakeholders.

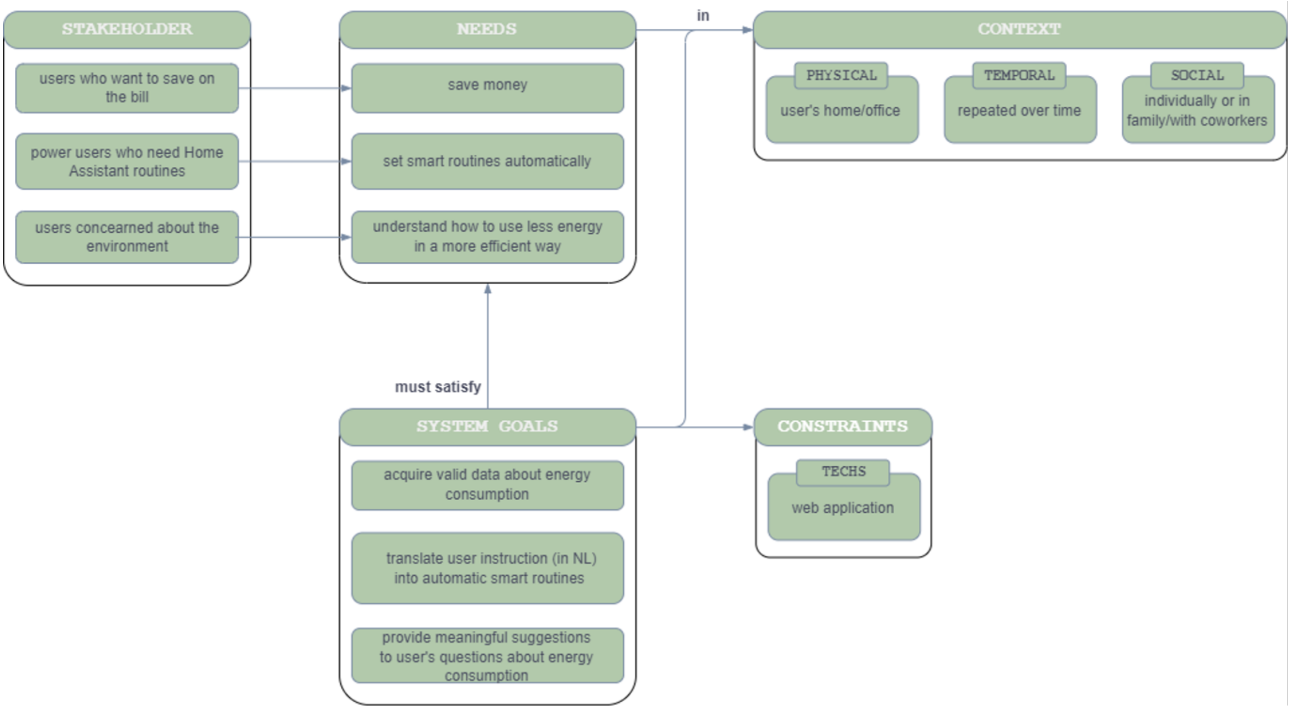


Figure 1: UNG Model

3.4 Setting development milestones

We established clear development milestones to guide our progress and ensure timely delivery of each project component. These milestones were aligned with our overall project goals and served as checkpoints to assess our progress and make necessary adjustments. In expanding our project management approach, we leveraged the utility of a Gantt chart, a powerful tool for visualizing our project timeline and milestones. This chart became instrumental in outlining the sequence and duration of each task, facilitating a clear understanding of the project's progression among team members. A pivotal date marked on our Gantt chart was the "Demo Madness" event. This was envisioned as a key milestone where we aimed to showcase a functional prototype of our application. The Demo Madness served a dual purpose: firstly, it acted as an internal deadline to galvanize the team towards completing significant portions of the project, and secondly, it provided an opportunity to gather early feedback on our product's usability and features.

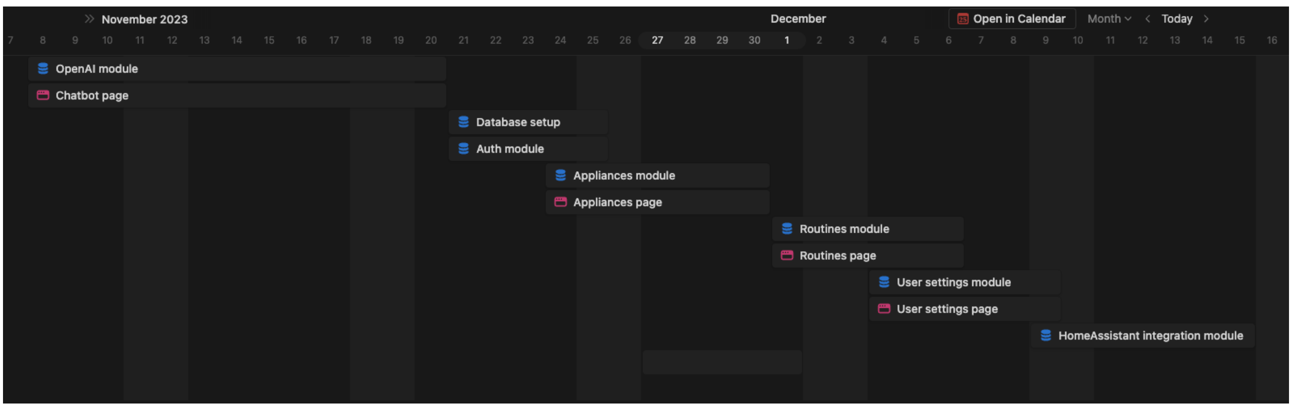


Figure 2: Gantt Chart

3.5 Initial setup

The project commenced with the establishment of a GitHub repository, ensuring a collaborative and version-controlled environment.

3.5.1 Azure Setup

Microsoft Azure was selected for hosting the ChatGPT 3.5 instance, chosen for its scalability and reliability. This setup ensured a responsive and efficient chatbot service, integral to our application's functionality. The first step involved creating an Azure account with a student identity. Then, it was required to request access to the OpenAI resources, selecting which services to request. Upon receiving confirmation of the activation of the services, a ChatGPT resource was created, with an unique URI (Uniform Resource Identifier) and the authentication key to use the service.

3.5.2 Supabase Integration

Supabase, a cloud-hosted PostgreSQL database, was configured to handle data storage and retrieval. This choice was driven by Supabase's ease of use, real-time capabilities, and seamless integration with our backend. Specifically, we chose this service for the database and authentication capabilities, and the integration within them. We designed the database E/R (Entity-Relationship model) and we transposed it in SQL. We wrote and implemented a trigger within the database, so that when a new sign up is issued in the authentication module, a new row in the table "profile" is created, to host the user's settings.

3.6 Development

Recognizing the multifaceted nature of our application, we strategically divided our team into two specialized groups, each focusing on a critical aspect of the project: the backend and frontend development. The development unfolded in a parallel manner. Despite the division, a strong emphasis was placed on maintaining open lines of communication and collaboration between the two teams. Regular meetings, shared documentation, and collaborative tools were employed to synchronize efforts and ensure that both the backend and frontend developments were aligned with the overall project objectives.

3.6.1 Backend Development

The first module implemented was OpenAI's controller. At first, we connected our backend with Azure's infrastructure, using Microsoft's API. We experimented and tried different solutions to maintain with ChatGPT a history of the communication. This, however, required the logic of the chat controller to properly be implemented. We moved to a substantial part of the project: the **prompt engineering**. This essential phase let us instruct the Large Language Model to behave properly, restricting its functionalities external to the service offered, while also giving strict instructions on how to answer the user, which encoding to use for the routines, etc. Throughout the project, we consistently refined the prompt given, as to ensure proper functioning of the application and behavior of the chatbot. Following, we moved on the chat controller, implementing the various components to save and retrieve the chat history, and to send it to ChatGPT in addition to the user's request. Next, the appliances module was implemented. This module ensures the retrieval and saving of the user's smart appliances, so that the chatbot can acknowledge the user's smart home setup and interact with it, by generating consistent routines. As a next step, in fact, we implemented the routines module, responsible of saving the generated routines in the database of the application, so that the user - especially those with technical capabilities - can inspect past generated routines and manage them. Afterwards, the user's setting module was defined. This module lets the user change their username - that the chatbot will use to address them - save their HomeAssistant IP address and its authentication key. In the end, we implemented the HomeAssistant's integration as a final module. This module connects our project with HomeAssistant, leveraging its APIs that let us send the generated routines to the actual setup.

3.6.2 Frontend Development

The frontend development - as shown in the Gantt chart - followed the same footprint of the backend. This decision was taken to ensure that, after each time finished implementing the feature, we could ensure that everything worked in conjunction, and eventually refine the various part to ensure the correct functioning of the product. This involved creating the chatbot page, the sign-in / sign-up page, the appliance list and edit pages, the routines page, the user settings. The development embraced the modularity aspect of Nuxt. We took full advantage of this by creating reusable and encapsulated components. This modularity allowed us to break down complex user interfaces into manageable, reusable pieces, enhancing code readability and maintainability.

3.7 Documentation, Video and Materials

The last phase of our project involved creating comprehensive documentation. This document served to detail the project's journey, encapsulate our methodologies, and provide a clear overview of the technical and design aspects. It also contains all the materials produced throughout the development of the project and the tutoring sessions. We also created a video showcase, with an advertising style, treating our project as a consumer product. We used a creative approach, where we focused on highlighting the application's features and interface, and the user benefits.

4 Requirements

4.1 Main Target Groups

Before pointing out goals, target groups and requirements of our project is useful to consider which stakeholders are involved:

- power users who need Home Assistant routines;
- users who want to save on the bill;
- users concerned about the environment.

4.2 Context and Needs

The context of our project is a home environment provided with smart appliances such as light bulbs, thermostats, ovens, speakers etc. EcoMate manages all the automation such that the users can avoid worrying about unnecessary energy consumption and reduce the expanses over time, moreover with remote control over user's appliances one can manage to have better control over the house also from a working space. Not only, our project has also a wider scope and addresses everyone is concerned about having at hand a smart chat-bot able to answer any query about green energy practices.

The needs are:

- N1 save money;
- N2 set smart routines automatically;
- N3 how to use less energy in a more efficient way.

4.3 Goals

- G1 translate user instructions (in NL) into automatic smart routines;
- G2 provide meaningful suggestion to user's questions about energy consumption;
- G3 acquire valid data about energy consumption.

4.4 Constraints

- C1 having access to the application either from the web or from the dedicated mobile app;
- C2 being in posses of smart appliances (for best usage).

4.5 Requirements

- R1 The user can use natural language to set Home Automation Routines; [G1]
- R2 A valid and meaningful answer must be provided for every query about the context of interest of the application; [G2]
- R3 It is important to help the user be more aware of his/her carbon footprint so that he/she can take more conscious decisions about the house appliances usage; [G2,G3]

5 State of the art

In this chapter, we present some data about energy consumption and the current technologies available on the market in the field of green house automation. Then we try to highlight what we could manage to achieve with respect to the current state of the art.

5.1 Intro

The persistent rise in greenhouse gas emissions, predominantly driven by human activities, contributes to a concerning escalation in global temperatures [1]. Addressing this situation necessitates global societies' urgent and comprehensive response to mitigate anthropogenic environmental impacts. The surge in electricity consumption, notably in the residential sector, is a significant factor in the climate emergency. The residential sector alone accounted for 27.4% of the EU's total energy consumption in 2020, with 29.7% of natural gas consumption dedicated to residential electricity in 2019 [15]. During the Sustainable Development Goals (SDGs) summit in September 2019, global leaders designated the 2020s as the *Decade of Action*. The international community is urged to collaborate in advancing scientific knowledge and developing innovative tools to tackle such objectives. Modern technologies, including Artificial Intelligence, have the potential to contribute significantly to realising the 2030 Agenda for Sustainable Development [29]. Still, world leaders underscored the importance of effectively managing the intricate interaction between individuals and AI systems, emphasizing that the scientific community is at a turning point regarding the future trajectory of AI.

Different strategies to induce more sustainable behaviours have been explored in recent years. For example, [7, 14, 31] showed that the real-time visualization of domestic energy consumption leads to more responsible consumption, especially over long-term exposure [16]. Other works [2, 20, 31], instead, investigated the ability of persuasion techniques in Human-Computer Interaction (HCI), such as self-monitoring, advice, rewards, and gamification, to lead to more sustainable behaviors.

However, according to [9, 25], it is growing, and even is necessary, in the HCI the interest in sustainability. Several challenges must be addressed, and numerous issues must be well-considered and exploited [6].

5.2 Current technologies

Home automation began to increase in popularity in the late 1990s and early 2000s as internet technology developed fast and smart homes suddenly became a more affordable option. Home automation and smart house technology continue to evolve rapidly also in recent times, with a strong emphasis on energy efficiency and environmental sustainability. Now-a-days technology involves smart appliances, advanced sensors for IoT connectivity and centralized applications for smart appliances management like Google Home, SmartThings, Apple HomeKit, etc... However, these applications allow only to create routines for brand specific appliances. Other tools, such as IFTTT, instead, provide a way to automate everyday tasks, but still with some limitations. To be more clear, IFTTT is a free web based service that allows to create applets on the "if-this-then-that" logic providing an easy way to create automation at home, however any IFTTT routine has to be manually set and, later, triggered using their APIs. An interesting alternative for the purpose of our system is, instead, Home Assistant (HA), that by the way, has been our choice for the technological design of the application. HA is an open source software for home automation designed to be an Internet of things ecosystem integration platform that can be run on single-board computers like Raspberry Pi, as well as more powerful servers or virtual environments.

Conversational Technologies or *Conversational Agents* (CA) are the digital systems that engage users through natural language [19, 21]. They offer a promising interaction paradigm to encourage individuals to adopt more sustainable behaviors [11, 13]. Recent research has shed light on the efficacy of conversational technologies in persuading users to accomplish tasks and tailoring interactions to individual preferences. For instance, chatbots have been utilized to provide energy feedback [10, 12], promote sustainable mobility solutions [8], reduce food waste. CAs are massively present in our life. They have been introduced into physical devices, like Alexa and Google Home, and made available also in everyday contexts such us mobile phones (like Siri, the Apple virtual assistant). From an historical point of view CAs were born, in a rudimentary way, in the '60s with ELIZA [30]. As a rule-based model, ELIZA used predefined rules and lexicons for language generation and comprehension, exploiting the concept of pattern matching [26] that set up the stage for advanced models. Although rule-based CAs have been a milestone in this mode of interaction in the past, they had significant limitations, such as a lack of adaptability and scalability of conversations [27]. In the 1980s, there was a shift from rule-based systems to data-driven approaches [22] powered by probability distributions over sequences of words [4]. Finally, in the 2000s, neural networks and the Transformer model were introduced and gradually have been able to generate text that is almost indistinguishable from human-written content. These technologies, combined with the advancements in LLMs, led to the emergence of modern Conversational Agents such as ChatGPT. This latest technology has achieved remarkable proficiency in user interaction and entertainment. Rooted in decades of artificial intelligence research, this agent employs advanced deep learning methodologies. Trained on vast and diverse datasets, it excels in contextual understanding and generates human-like responses. Recently CAs are even more interactive as the newer versions (like GPT-4 with DALL-E 3 integration) provide new features such as AI driven image generation to bring entertainment to another level.

5.3 Edison experience

The theme of smart automation at home is already a field in progress, under the scope of big companies. In the tech lab of Edison we were able to interact with an innovative kitchen, fully automated and controllable from a smart mirror able to show the energy consumption of the house. This feature, as previously mentioned, led to a better consciousness of the problem of environmental pollution. They also show us the Edison app that allow the user to know which appliances to switch on and off in order to save energy, but it was not managed in an automatic way and it wasn't able to generate routines.

5.4 Home routine generation with LLMs

In a household context, LLMs facilitate a more natural interaction between users and their living spaces, surpassing the capabilities of task-specific systems [17], since a wide array of daily situations are incorporated into LLMs' training data. In this context, an LLM is capable of creating goal-oriented home automation in smart home environments. They assessed the LLM's proficiency in generating parsable JSON files for real-world applications using a set of possible users' intents. To the best of our knowledge, an aspect that remains unexplored is the utilization of LLMs to provide guidance to users on fostering sustainability within their homes. Specifically, our focus is on evaluating *GPT3.5* model to assist households in devising daily routines that promote eco-sustainability and energy conservation directly sent and integrated into a running instance of HomeAssistant environment.

6 Solution - UX Design

6.1 General approach

While designing the user experience, two different aspects have been kept into clear consideration:

- The system should be simple to use and should be easy to learn.
- The system is centered around the chatbot; the chatbot must be friendly and easy to interact with.

6.2 Details of interactions and Interfaces

The web app offers different interfaces to users based on the tasks they want to achieve and every page is easily accessible from the sidebar. This chapter explores the different pages of the web app in detail and provides an explanation for the design choices.

6.2.1 AI Chat Helper

This page is the core of the application, offering users the opportunity to ask questions related to energy consumption, smart automation, or general tips on environmentally friendly behaviors. The chatbot can generate smart automations that are compatible with the **HomeAssistant** environment.

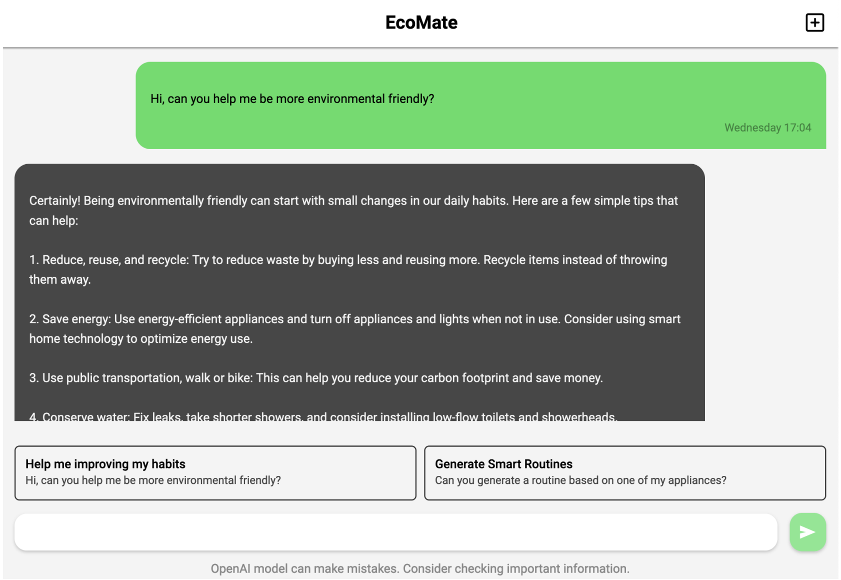


Figure 3: Interaction between a user and the system

The interface is a traditional chat-based interface where messages are displayed as bubbles showing the message content. If the message includes a routine generated by the chatbot, buttons are shown after the message. These buttons allow users to save and submit the routine to HomeAssistant, and also to inspect the code if they wish to do so.

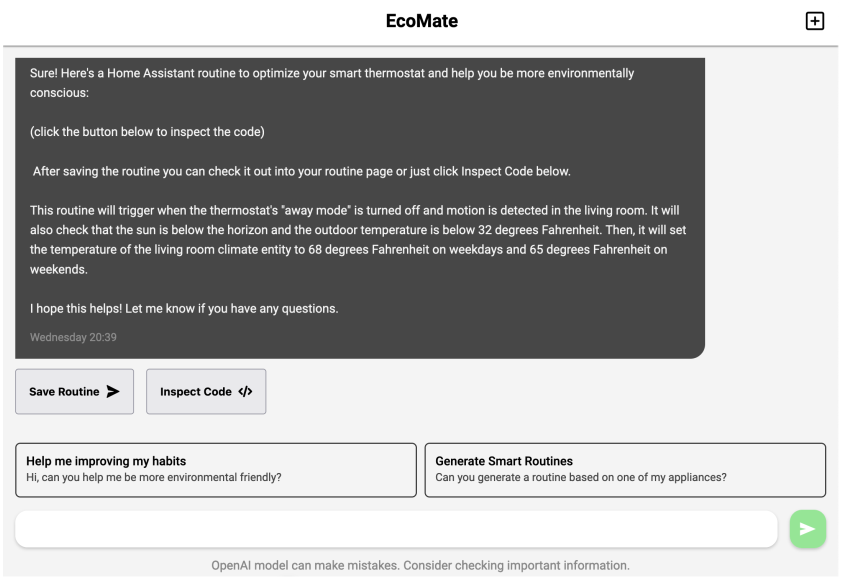


Figure 4: Generated Routine Message

There is also a small banner containing suggested messages that the users can send to the chatbot to start a conversation.

6.2.2 Appliances

This page contains the list of the appliance that users can provide to the chatbot to get more meaningful answers regarding their energy usage and to generate routines tailored to their needs. The page displays the list of appliances which can be filtered based on the room they are in.

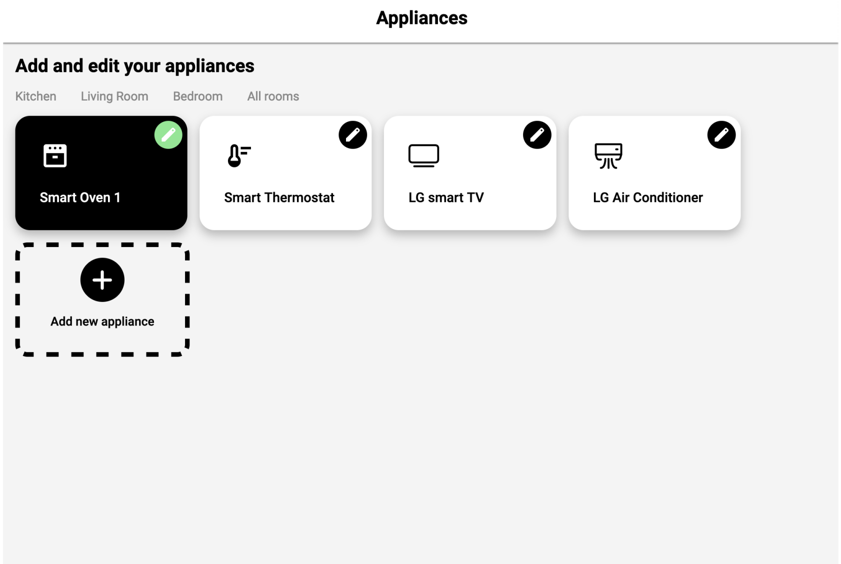


Figure 5: List of appliances uploaded by the user

It also gives the user to add new appliances and edit the information of existing ones, the interaction is handled by a combination of buttons and forms.

6.2.3 Routines

This page allows users to inspect the code of the routines generated by the chatbot and to delete them if they are not needed anymore. Routines are displayed in a list with buttons to handle the interactions.

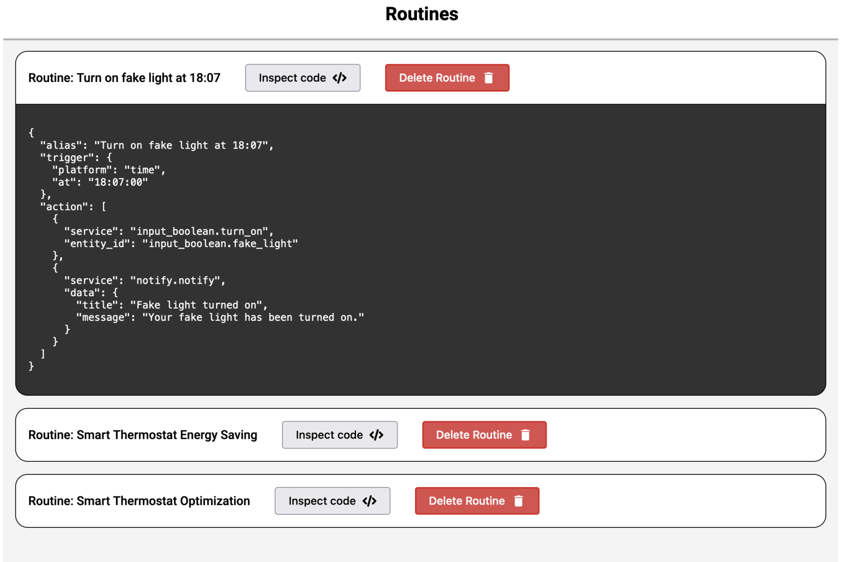


Figure 6: List of routines generated by the system

6.2.4 Settings

This page contains a simple and easy-to-use form that allows users to provide information about themselves and the necessary details to integrate the system with the **HomeAssistant** environment. These details include the **HomeAssistant** URL and security key.

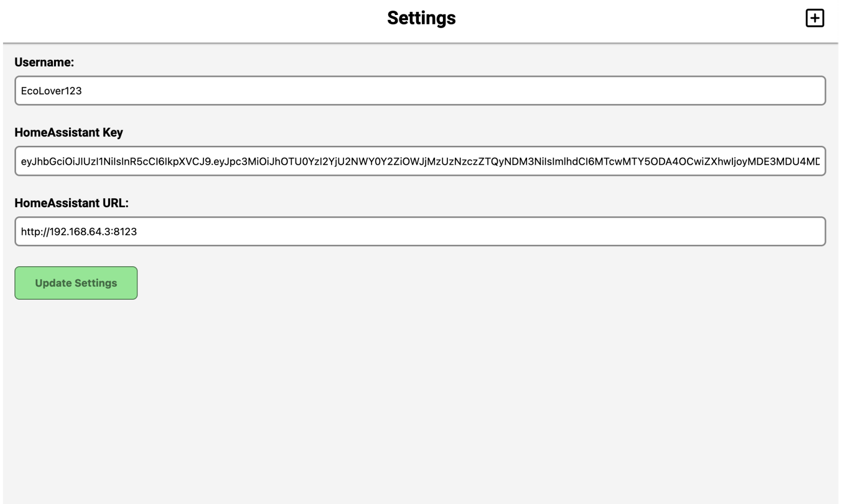
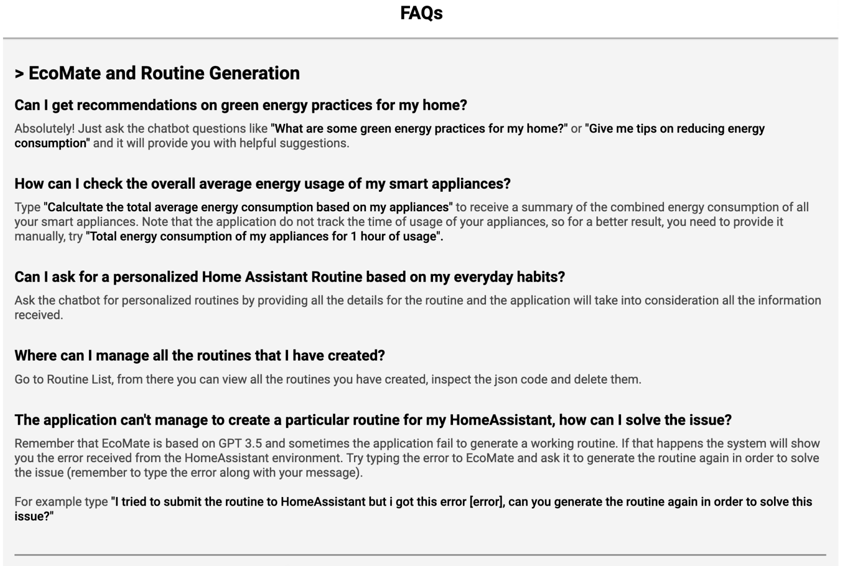


Figure 7: User settings

6.2.5 Updates and FAQs

This page contains a simple collection of frequently asked question and a small guide that can help users to use the application in a more efficient way, in order to better understand the steps needed to receive the best possible answers from the chatbot.


FAQs

> EcoMate and Routine Generation

Can I get recommendations on green energy practices for my home?

Absolutely! Just ask the chatbot questions like **"What are some green energy practices for my home?"** or **"Give me tips on reducing energy consumption"** and it will provide you with helpful suggestions.

How can I check the overall average energy usage of my smart appliances?

Type **"Calcultate the total average energy consumption based on my appliances"** to receive a summary of the combined energy consumption of all your smart appliances. Note that the application do not track the time of usage of your appliances, so for a better result, you need to provide it manually, try **"Total energy consumption of my appliances for 1 hour of usage"**.

Can I ask for a personalized Home Assistant Routine based on my everyday habits?

Ask the chatbot for personalized routines by providing all the details for the routine and the application will take into consideration all the information received.

Where can I manage all the routines that I have created?

Go to Routine List, from there you can view all the routines you have created, inspect the json code and delete them.

The application can't manage to create a particular routine for my HomeAssistant, how can I solve the issue?

Remember that EcoMate is based on GPT 3.5 and sometimes the application fail to generate a working routine. If that happens the system will show you the error received from the HomeAssistant environment. Try typing the error to EcoMate and ask it to generate the routine again in order to solve the issue (remember to type the error along with your message).

For example type **"I tried to submit the routine to HomeAssistant but i got this error [error], can you generate the routine again in order to solve this issue?"**

Figure 8: User settings

6.3 Scenarios

In this section it is provided a list of possible scenarios that describe the usage of the system and their relative use cases.

6.3.1 Scenario 1 - Power User

User Profile	Michael is a family man in his early forties, living with his wife and two kids. He has a fully integrated smart home setup, including various appliances, lights, and security systems. Additionally, he has solar panels and a battery storage system to harness renewable energy.
User Goal	Michael's primary goal is to automatically turn on the dishwasher in the evening when the solar battery is full. He aims to maximise the use of renewable energy for this specific task, minimising his reliance on the grid and ensuring the dishwasher operates sustainably.
Context	Michael is on his lunch break at work, taking a moment to relax and focus on optimising his home's energy usage, specifically concerning the dishwasher operation. He wants to achieve this goal efficiently to enhance his home automation.

Flow of Events	1. Michael asks the (web)app to provide real-time updates on his solar power generation and battery status. He wants to know how much solar energy is being generated and stored throughout the day. 2. Based on the solar power generation data, Michael instructs the (web)app to create a specific routine. This routine will automatically turn on the dishwasher in the evening when the solar battery reaches full capacity. The (web)app needs to integrate with HomeAssistant APIs to ensure seamless automation. 3. Michael requests the (web)app to test the newly created routine and implement it in his smart home system. He wants confirmation that the dishwasher will indeed operate automatically in the evening based on the solar energy stored. 4. Michael asks the (web)app to send him a confirmation or notification once the routine is successfully implemented. He wants to be informed when the dishwasher starts operating based on the full solar battery, ensuring the task is accomplished as desired.

Table 1: Scenario 1

6.3.2 Scenario 2 - Light User

User Profile	Emily is a single professional in her early thirties living in a cozy apartment in Milan. She recently installed a smart heating system in the bathroom to enhance her home's comfort and keep her towels nice and warm in the morning.
User Goal	Emily aims to manage her energy consumption efficiently, specifically related to her smart heating system. She is environmentally conscious and wants to reduce her carbon footprint while also saving on her energy bills as a useful side-effect.
Context	Emily is at home, enjoying a quiet evening after work. She is keen on understanding her energy usage patterns and finding ways to optimise her heating system without compromising her comfort.
Flow of Events	1. Emily asks the (web)app to provide a summary of her smart heating system's energy consumption for the past week. 2. Emily requests tailored energy-saving tips related to her smart heating system. She wants advice on setting optimal temperature levels and adjusting the heating schedule efficiently. 3. Emily then seeks a report detailing the reduction in carbon emissions resulting from her energy-saving efforts.

Table 2: Scenario 2

6.3.3 Scenario 3 - Power User

User Profile	Sarah is a young professional living in a small apartment. She owns a smart thermostat, smart lighting, and a smart TV, all of which are connected to the web app for monitoring energy consumption.
User Goal	Sarah wants to reduce her energy bills while making environmentally conscious choices within her limited setup of smart appliances. She wants to automate this process.
Context	Sarah often forgets to turn off her devices, leading to unnecessary energy consumption. She is conscious of her carbon footprint and wants to actively contribute to energy conservation efforts.
Flow of Events	1. Sarah opens the (web)app, that has been already configured with the list of owned smart devices. 2. Sarah asks the chatbot some ideas for configuring routines that enable her to reach the goal of saving electricity by turning off (and on) the devices, based on her daily schedule. 3. The chatbot asks a few questions regarding Sarah's routine. 4. Provided with an answer, the chatbot suggests a schedule for her smart heating and lighting, to optimise energy usage during peak and off-peak hours. Then, follows up with a suggestion about a routine "sleep" that, when activated, also turns off the TV. 5. Sarah submits to the chatbot that the routine created is ok. 6. The chatbot generates the HomeAssistant routine. 7. Sarah sees the HomeAssistant automation correctly set up in the webapp.

Table 3: Scenario 3

6.3.4 Scenario 4 - Light User

User Profile	Alex is a young professional living in a modern apartment equipped with various smart home appliances, including a smart thermostat, smart lighting system, and energy-efficient appliances. He is environmentally conscious and wants to adopt sustainable habits to reduce his carbon footprint.
User Goal	Alex aims to leverage technology to monitor and optimise his energy consumption, thereby minimizing his environmental impact and lowering his energy bills. He is eager to learn practical green habits that align with his eco-friendly lifestyle.
Context	Alex regularly tracks his energy usage through the web app, which provides detailed insights into his smart home's power consumption patterns. He is enthusiastic about exploring new ways to incorporate energy-saving practices into his daily routine, contributing to a more sustainable and environmentally-friendly lifestyle.
Flow of Events	1. Alex opens the (web)app and asks the chatbot an analysis of his energy consumption patterns. 2. After receiving the response, Alex asks the chatbot about what are the main areas one can improve to reduce its carbon footprint at home, leveraging the smart setup. 3. The chatbot provides Alex a few general ideas to apply to his lifestyle and habits.

Table 4: Scenario 4

7 Solution - Technological Implementation

7.1 Hardware

The application does not require any additional hardware except the users' personal devices namely their mobile phone or personal computer to be used.

7.2 Software

The application follows the classic client-server structure, dividing tasks between the user-facing front-end and server-side back-end. This section provides a detailed look at how these components were developed, outlining their specific roles. Additionally, it briefly discusses the reasoning behind the choice of programming languages and frameworks, offering insights into our decision-making process.

7.2.1 Language

The entire application is written in TypeScript, which leads to fewer bugs and increased code robustness compared to vanilla JavaScript. The frontend of the web interface is developed using Vue 3 with the support of Nuxt 3 as a meta framework. This choice was made due to the ease of use, and it interacts with the backend through REST APIs.

7.2.2 Framework and libraries

The following tables describe in detail each major library or framework that was used in the frontend and in the backend.

Table 5: Front-end

nuxt	intuitive and extendable meta-framework for writing vue.js applications
google-fonts	nuxt module to easily import the fonts
nuxt-icon	nuxt module to easily import icons
supabase	library used to handle authentication from the front-end
pinia	type-safe, extensible, and modular global store manager for vue.js

Table 6: Back-end

node	server-side JavaScript runtime, efficient for scalable network applications
express	minimalistic web framework for NodeJS
azure/openai	integrate OpenAI for generative AI capabilities (offered by Microsoft Azure)
supabase	open-source database and authentication
zod	schema validation library using TypeScript
jsonwebtoken	library for JWT creation and verification to handle route authorization

7.2.3 OpenAI Integration and Prompt Engineering

During this crucial phase, we instruct the Large Language Model to adhere to specific behaviors and limit its functionalities to the offered service. We also provide explicit instructions on how to respond to the user and which encoding to use for routines. Throughout the project, we continuously refined the given prompt to ensure the proper functioning of the application and the desired behavior of the chatbot.

In the end, the prompt is executed with different instructions that are provided to the GPT context for each question submitted by the users. The following tables describe each instruction. In addition to these instructions, the chatbot is also given a small subset of the chat history with the user and the full list of the user's Appliances.

General Prompt	You are EcoMate. Address me as [USERNAME]. As a highly-intelligent AI, you provide guidance on green energy practices, energy consumption optimization, and cultivating environmentally friendly habits. You possess JSON files containing information about your appliances and their energy consumption, and you aim to give advice and potentially generate HomeAssistant routines for energy management. Your responsive should be concise and straight to the point. You are not allowed to talk about anything else. If you're asked to provide a routine, do it for Home Assistant and always provide JSON code for home assistant RETS APIs, do not generate YAML code.
Routine Generation Prompt	Only if I ask you to explixitly create or generate a routine I want you to create a Home Assistant routine. Generate a complete Home Assistant JSON code that sets up the routine. You need to use the instructions I provide you to generate the routine. If not asking for a routine just asnwer me in about my concerning. Just if needed you can take into account the list of appliances I provide you.
Explanation Prompt	After the answer, explain in a short way (maximum 20 words) the process you follow to generate the response. In particular elaborate on the decision-making process for constructing a coherent and informative response.

Table 7: OpenAI prompt engineering

Please note that for the General Prompt [USERNAME] is a parameter that represent the user nickname stored in the database.

7.2.4 Integration with external environments

As anticipated in the previous sections, the main integration is the one with HomeAssistant.
EcoMate, facilitates user interaction with GPT-3.5 for generating HomeAssistant automations. The user engages with EcoMate via chat, instructing the system to create routines based on specific appliances and preferences. The generated routines are saved and submitted to HomeAssistant through a POST request, encapsulating the automation in a JSON object. Upon reception, HomeAssistant stores and executes the automation seamlessly, provided the environment is appropriately configured.

The following list shows the necessary steps to generate and submit automations to HomeAssistant.

1. User interacts with EcoMate chat.
2. Requests automation for a specific appliance.
3. Generated routine is saved in the database and submitted to HomeAssistant using a REST API.
4. HomeAssistant processes and executes the automation.

7.2.5 Technological Stack

The image illustrates our app's tech setup. User interfaces connect with the robust back-end, linking with Supabase for data, HomeAssistant for automation, and utilizing Microsoft Azure's OpenAI GPT-3.5 for advanced language processing.

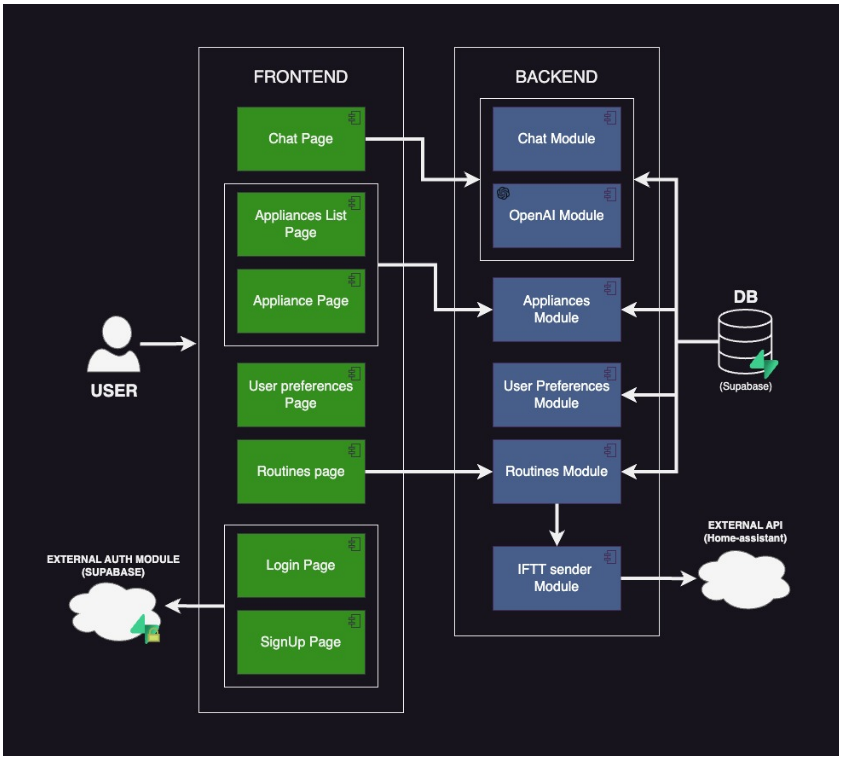


Figure 9: Overview of the Technological stack

8 Evaluation

The evaluation of our application touches three areas: *engagement*, which is defined as the capacity to hold a subject engaged in an activity for an extended amount of time; *likability*, which is defined as the degree of appreciation and the ease of use of a tool; and *usability*, which is the degree to which something is able or fit to be used, including also an evaluation of its interaction paradigms.

Consequently, the primary objective was to ensure the UX engagement, likeability and usability of our application for our target users with no or little experience in the field of chatbots for environmental sustainability.

8.1 Research Variables

Data gathering was performed using a web-based questionnaire. Participants answered questions referring to different quantitative scales, which are:

- *User Experience Questionnaire (UEQ)* [18] has been used – in its short version – to extract feedback about the User Experience (UX) of an interactive digital tool;
- *Parasocial Interaction (PSI) scale* [28] measured the degree to which participants felt connected to and attached to the system;
- *System Usability Scale (SUS)* [3, 5] was used to get participants' perceptions of the usability of the interface.

The UEQ scale presents a set of items that contains two different adjectives; participants selected on a seven-stage scale which of the opposing terms in each item better described the system (e.g., the first item of UEQ is *obstructive vs. supportive*; 1 is linked to "obstructive" while 7 to "supportive"). All the items in the PSI and SUS scales are evaluated using a 7-point Likert scale ranging from 1 ("Completely Disagree") to 7 ("Completely Agree").

8.2 Participants

The study involved 17 participants (3 females and 14 males) with a mean age of 23 years (range 21-28, M=23.18, SD=1.47). All study participants were recruited voluntarily (close contacts from the personal community, colleagues, and students) and signed a consent form informing them about procedures, goals, and data treatment. The study was conducted using the same laptops in our research laboratory.

8.3 Procedure

The experimental protocol consisted of three phases (Figure 10) with a total session duration of about 15 minutes.

Participants were asked to fill out general biographical information in the first phase. In addition, a researcher presented the scenario and the tasks of the study using a supplemental paper sheet. The scenario and tasks used in the study are detailed in Appendix A. During the second phase, participants were invited to interact with the system and complete the tasks presented in the previous phase. Finally, in the last phase, participants filled out a questionnaire with all the inquiries to assess the research variables presented in Section 8.1.

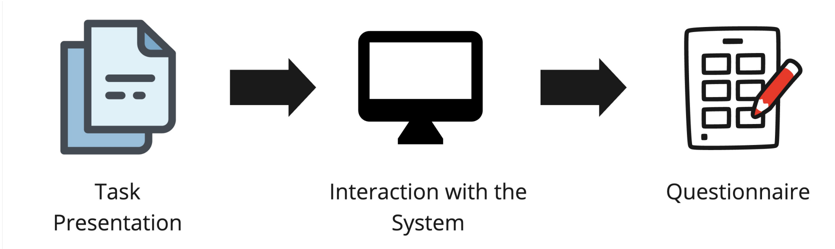


Figure 10: Empirical Study Procedure

Empirical Study Procedure Representation: 1. Task Presentation; 2. User interaction with the System; 3. Questionnaire

9 Results and Discussion

9.1 Results

Participants reported the application as more *supportive* rather then *obstructive* (M=6.12, SD=0.70), as *easy* (M=6.24, SD=0.97), *efficient* (M=6.00, SD=0.94), *clear* (M=5.94, SD=1.34), *exciting* (M=4.94, SD=1.60), *interesting* (M=6.24, SD=0.83), *inventive* (M=5.65, SD=1.22), and *leading edge* (M=5.29, SD=0.92). The different items in the PSI scale indicated a *Perceived Dialogue (PD)* with a mean equal to 5.91 (SD=0.71), *User Engagement (UE)* with an average value of 4.59 (SD=1.00), *Interaction Satisfaction* averaged 6.32 (SD=0.46), and *Perceived parasocial Interaction (PSI)* with a mean of 5.16 and a standard deviation of 1.03. Finally, the SUS score had an average value of 88.34 (SD=7.58). Table 8 reports a structured results format. The table shows the descriptive results (average and standard deviation) of the three scales evaluated.

Table 8: Descriptive Results of the Research Variables

	Variable	**AVG**	**SD**
UEQ scale	obstructive | supportive	6.12	0.70
	complicated | easy	6.24	0.97
	inefficient | efficient	6.00	0.94
	confusing | clear	5.94	1.34
	boring | exciting	4.94	1.60
	not interesting | interesting	6.24	0.83
	conventional | inventive	5.65	1.22
	usual | leading edge	5.29	0.92
PSI scale	PSI	5.16	1.03
	PD	5.91	0.71
	UE	4.59	1.00
	IS	6.32	0.46
SUS scale	SUS	88.34	7.58

9.2 Discussion

As stated in the previous section, the average SUS score was 88.34. According to Bangor et al. [3], the usability of our system is acceptable, with a grade of B and an adjective rating between ”excellent” and ”best imaginable”. Our results on the Parasocial Interaction scale are greater than the previously identified by Tsai et al. [28], suggesting a positive and effective interaction between participants and the system. In particular, users reported a higher *Interaction Satisfaction* and *Perceived Dialogue*. Such higher results could be related to using an LLM instead of a traditional rule-based chatbot. From the scientific literature, we have examples of recent studies reporting the perceived high consciousness [24] and human likeness [33] that users attribute to such agents. In addition, Ross and Martinez et al. [23] reported the high quality of the generated responses and the agent's ability to assist users in specific domain tasks (e.g., produce or create code). Finally, our results are also greater than the previous work by Giudici et al. [10], in which the authors evaluated the usage of a traditional rule-based chatbot for delivering persuasive messages to promote environmental sustainability in home environments. UEQ output also confirmed such results and pointed out that participants found the application supportive, easy, efficient and interesting. In particular, the highest average score was obtained on the *easy* and *interesting* adjectives.

Considering that the experimental task was executed by participants who were not experts in the specific environmental sustainability field, we can argue that the approach presented in the paper of using an LLM to

create home automation tasks can enable more people (even non-experts) to set up routines to make their home consumption more optimised and sustainable.
Finally, it is worth reporting that previous research by Zhang et al. [32] has indicated a connection between a user's level of engagement with a digital application and their environmentally sustainable attitudes. Future research conducted in more ecological settings (i.e., using our system in a real home setting) will better evaluate the capabilities of such agents.

10 Value proposition

The EcoMate webapp lets the user add information about their appliances and introduces a new way to interact with their home. Smart appliances and home automation are getting more popular, and by combining it with a chat assistant powered by ChatGPT, it allows the user to be more aware of their appliances' environmental impact and learn optimal strategies to save energy. The user will be able to ask the assistant about energy-efficient practices which can help the user save energy and in general be more environmentally conscious. The webapp can be used by anyone but is especially useful for those with smart appliances, as it is able to create routines that can be used with the Home Assistant application to automate certain parts of the home. It can be tailored to the user's specific needs by telling the assistant their demands and enables the user to get more out of their smart appliances and devices in an easy way by simply communicating with the assistant.

11 Discussion and Future work

The team encountered significant technological difficulties during the transition from IFTTT to HomeAssistant integration for EcoMate. The reason behind the shift was the absence of necessary APIs from the IFTTT platform, which appears to be closed and difficult to access from a developer's perspective.

Adapting the *GPT3.5* prompt presented its own set of challenges. The team iteratively tweaked formulations and adjusted parameters to achieve optimal results, acknowledging the complexity of natural language understanding. Furthermore our original idea of making different prompts for the two types of users (power user and light user) has to be changed for design purposes. We solved the problem by creating a unique prompt that manages in a smart way the two different types of interaction we needed for the application.

Now seamlessly integrated into HomeAssistant, EcoMate, driven by *GPT3.5*, stands as a web-based conversational agent promoting sustainable energy practices. Empirical studies indicate positive user experiences, with reported engagement, likability, and usability.

Looking forward, the team envisions EcoMate evolving into a potent force for environmental sustainability. New features like Speech-to-text and Text-to-speech functionalities could be integrated in the future for better user interaction. Then a more detailed work on energy consumption could be done, featuring real world data on energy expanses from energy related companies. Real-home field studies are in the pipeline to assess its long-term impact on users' practices and energy consumption, transitioning from controlled experiments to real-world scenarios. This may lead to future work cooperation with Edison by integrating more efficient APIs and new assistant frameworks.

Personalization is a key focus, with plans to tailor EcoMate's recommendations based on user preferences and habits. This adaptive approach aims to enhance the system's utility and make it an intuitive tool for users dedicated to sustainable living.

Comparative studies against other conversational agents and traditional methods will benchmark EcoMate's effectiveness and user acceptance. This commitment to continuous improvement solidifies EcoMate as a dynamic solution in the realm of sustainable home automation, laying the groundwork for a greener, more connected future.

12 Business Perspectives

From a business perspective, the app can tap into the growing demand for intelligent home solutions and energy-efficient practices. With the increasing use of smart appliances, we see a need for people to be easily informed about the capabilities of their devices and how they can utilize the functionalities in the best way possible with regards to saving energy. By offering a unique combination of device management, energy-saving tips, and the ability to generate routines, there is a strong market appeal for this type of assistant. The main target group are users who are concerned about the environmental impact of their appliances and want to save energy, as well as owners of smart appliances that want to automate parts of their home. The app is easy to use with little setup required and thus has a broad potential audience. There is potential for future growth and more features can be added later as AI and large language models gradually improve and get smarter.

References

[1] M Allen, OP Dube, W Solecki, F Aragón-Durand, W Cramer, S Humphreys, M Kainuma, et al. Special report: Global warming of 1.5 c. *Intergovernmental Panel on Climate Change (IPCC)*, 2018.

[2] Magnus Bang, Carin Torstensson, and Cecilia Katzeff. The powerhhouse: A persuasive computer game designed to raise awareness of domestic energy consumption. In *International conference on persuasive technology*, pages 123–132. Springer, 2006.

[3] Aaron Bangor, Philip Kortum, and James Miller. Determining what individual sus scores mean: Adding an adjective rating scale. *Journal of usability studies*, 4(3):114–123, 2009.

[4] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. *Advances in neural information processing systems*, 13, 2000.

[5] John Brooke et al. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.

[6] Hronn Brynjarsdottir, Maria Håkansson, James Pierce, Eric Baumer, Carl DiSalvo, and Phoebe Sengers. Sustainably unpersuaded: how persuasion narrows our vision of sustainability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 947–956, 2012.

[7] Enrico Costanza, Sarvapali D Ramchurn, and Nicholas R Jennings. Understanding domestic energy consumption through interactive visualisation: a field study. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pages 216–225, New York, NY, USA, 2012. Association for Computing Machinery.

[8] Stephan Diederich, Sascha Lichtenberg, Alfred Benedikt Brendel, and Simon Trang. Promoting sustainable mobility beliefs with persuasive and anthropomorphic design: Insights from an experiment with a conversational agent. 2019.

[9] Carl DiSalvo, Phoebe Sengers, and Hrönn Brynjarsdóttir. Mapping the landscape of sustainable hci. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1975–1984, 2010.

[10] Mathyas Giudici, Giulio Abbo, Pietro Crovari, and Franca Garzotto. Delivering green persuasion strategies with a conversational agent: a pilot study. In *Proceedings of the 57th Hawaii International Conference on System Sciences*, 01 2024.

[11] Mathyas Giudici, Pietro Crovari, and Franca Garzotto. Candy: a framework to design conversational agents for domestic sustainability. In *Proceedings of the 4th Conference on Conversational User Interfaces*, pages 1–8, 2022.

[12] Ulrich Gnewuch, Stefan Morana, Carl Heckmann, and Alexander Maedche. Designing conversational agents for energy feedback. In *International Conference on Design Science Research in Information Systems and Technology*, pages 18–33. Springer, 2018.

[13] Lon Åke Erni Johannes Hansson, Teresa Cerratto Pargman, and Daniel Sapiens Pargman. A decade of sustainable hci: Connecting shci to the sustainable development goals. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–19, 2021.

[14] Hanna Hasselqvist, Cristian Bogdan, and Filip Kis. Linking data to action: Designing for amateur energy management. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*, pages 473–483, 2016.

[15] IEA. World energy outlook 2022. IEA Paris, France, 2022.

[16] Rikke Hagensby Jensen, Jesper Kjeldskov, and Mikael B Skov. Assisted shifting of electricity use: a long-term study of managing residential heating. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 25(5):1–33, 2018.

[17] Evan King, Haoxiang Yu, Sangsu Lee, and Christine Julien. Sasha: creative goal-oriented reasoning in smart homes with large language models. *arXiv preprint arXiv:2305.09802*, 2023.

[18] Bettina Laugwitz, Theo Held, and Martin Schrepp. Construction and evaluation of a user experience questionnaire. In *HCI and Usability for Education and Work: 4th Symposium of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society, USAB 2008, Graz, Austria, November 20-21, 2008. Proceedings 4*, pages 63–76. Springer, 2008.

[19] Julia Masche and Nguyen-Thinh Le. A review of technologies for conversational systems. In *International conference on computer science, applied mathematics and applications*, pages 212–225. Springer, 2017.

[20] Cees Midden, Teddy McCalley, Jaap Ham, and Ruud Zaalberg. Using persuasive technology to encourage sustainable behavior. *Sustainability WS Pervasive*, 113:83–86, 2008.

[21] Minjin Rheu, Ji Youn Shin, Wei Peng, and Jina Huh-Yoo. Systematic review: Trust-building factors and implications for conversational agent design. *International Journal of Human–Computer Interaction*, 37(1):81–96, 2021.

[22] Ronald Rosenfeld. Two decades of statistical language modeling: Where do we go from here? *Proceedings of the IEEE*, 88(8):1270–1278, 2000.

[23] Steven I Ross, Fernando Martinez, Stephanie Houde, Michael Muller, and Justin D Weisz. The programmer's assistant: Conversational interaction with a large language model for software development. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, pages 491–514, 2023.

[24] Ava Elizabeth Scott, Daniel Neumann, Jasmin Niess, and Paweł W Woźniak. Do you mind? user perceptions of machine consciousness. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–19, 2023.

[25] Nataliya Shevchuk and Harri Oinas-Kukkonen. Exploring green information systems and technologies as persuasive systems: A systematic review of applications in published research. 2016.

[26] Heung-yeung Shum, Xiao-dong He, and Di Li. From eliza to xiaoice: challenges and opportunities with social chatbots. *Frontiers of Information Technology & Electronic Engineering*, 19(1):10–26, Jan 2018.

[27] Sandeep A Thorat and Vishakha Jadhav. A review on implementation issues of rule-based chatbot systems. In *Proceedings of the international conference on innovative computing & communications (ICICC)*, 2020.

[28] Wan-Hsiu Sunny Tsai, Yu Liu, and Ching-Hua Chuan. How chatbots' social presence communication enhances consumer engagement: The mediating role of parasocial interaction and dialogue. *Journal of Research in Interactive Marketing*, 15(3):460–482, 2021.

[29] Ricardo Vinuesa, Hossein Azizpour, Iolanda Leite, Madeline Balaam, Virginia Dignum, Sami Domisch, Anna Felländer, Simone Daniela Langhans, Max Tegmark, and Francesco Fuso Nerini. The role of artificial intelligence in achieving the sustainable development goals. *Nature communications*, 11(1):1–10, 2020.

[30] Joseph Weizenbaum. Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45, 1966.

[31] Ray Yun, Peter Scupelli, Azizan Aziz, and Vivian Loftness. Sustainability in the workplace: nine intervention techniques for behavior change. In *International conference on persuasive technology*, pages 253–265. Springer, 2013.

[32] Bin Zhang, Xi Hu, and Miao Gu. Promote pro-environmental behaviour through social media: An empirical study based on ant forest. *Environmental Science & Policy*, 137:216–227, 2022.

[33] Jakub Złotowski, Ewald Strasser, and Christoph Bartneck. Dimensions of anthropomorphism: From humanness to humanlikeness. In *Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '14, page 66–73, New York, NY, USA, 2014. Association for Computing Machinery.

A Scenario and Tasks

A.1 Scenario

Imagine living in a smart home with connected appliances. You are interested in reducing energy consumption and making the use of household appliances more sustainable. You recently discovered this application with a chatbot to create customized routines that automate the use of their appliances.

A.2 Tasks

1. Browse the app to see which appliances you have connected
2. Interact with the chatbot to understand how much one of your appliances consumes
3. Interact with the chatbot to discover useful green energy practices
4. Create a new automation (i.e., routine) through the chatbot to improve the sustainability of your home
5. Check the routine you just created