Smart Forest

Advanced User Interfaces - Project Report

Matteo Beltrante, Marco Bendinelli, Luca Incarbone, Francesco Piferi, Lorenzo Poretti

Professor: **Franca Garzotto**
Assistants: **Mathyas Giudici, Pietro Crovari Giulio Antonio Abbo**
Edison Tutor: **Riccardo Silvestri**

A.Y. 2022/2023

POLITECNICO
MILANO 1863

Contents

1 Abstract

Smart Forest is a collaborative project developed by Politecnico di Milano and the energy provider Edison. The project aims to create an interactive mirror that helps users to easily monitor their energy consumption and encourages them to use renewable energy sources. The mirror displays an interactive forest, where users can earn *leaves* for using renewable energy and lose *experience points*, which is the lifespan of each tree, for using non-renewable energy. These *leaves* can then be used to purchase and plant new trees in order to gain more and more *leaves*. The mirror also provides real-time feedback and personalized recommendations for reducing energy consumption. Overall, *Smart Forest* project hopes to engage users in an interactive and fun way to promote sustainable energy practices and raise awareness about the importance of renewable energy.

2 Team members

Matteo Beltrante
Computer Engineer
matteo.beltrante@mail.polimi.it
+39 333 3060712

Master program:
Computer engineering

Marco Bendinelli
Computer Engineer
marco.bendinelli@mail.polimi.it
+39 347 5914618

Master program:
Computer engineering

Luca Incarbone
Computer Engineer
luca.incarbone@mail.polimi.it
+39 333 2726612

Master program:
Computer engineering

Francesco Piferi
Computer Engineer
francesco.piferi@mail.polimi.it
+39 346 6722308

Master program:
Music and Acoustics Engineering

Lorenzo Poretti
Computer Engineer
lorenzo.poretti@mail.polimi.it
+39 338 3909799

Master program:
Computer engineering

Matteo Beltrante	He contributed on the software development of the project, in particular on the development of the State Machine and the train of the conversational agent (Flora)
Marco Bendinelli	He contributed on the software development, in particular on the server side and the communication between back-end and front-end. He also developed the *Smart Forest* API
Luca Incarbone	He mostly developed the front-end of the forest, creating the design of the mirror. He also trained the conversational agent (Flora) and contributed on the software development of the project. He edited the promotional video
Francesco Piferi	He managed the software installation of the tools on the mirror and the training of the conversational agent (Flora). He wrote the report and the presentation slides
Lorenzo Poretti	He developed the *Smart Forest* API, managed the software installation of the tools on the mirror, and contributed on the software development of the project

3 Technology overview

Smart Forest is a home automation system that is integrated into $MagicMirror^2$ which is a mirror that serves as a smart display. The $MagicMirror^2$ is built with a microphone and a speaker by default, allowing for voice commands to be given to *Smart Forest*. The system is powered by a Raspberry Pi. In order to use *Smart Forest*, the home also has some additional requirements. Specifically, the home must have photovoltaic panels and batteries installed. The solar panels are used to generate electricity, which can be stored in the batteries. In principle, the user should use only the electricity that his solar panel has been able to capture, but not always this is possible. *Smart Forest* tries to make the user aware about his consumptions, helping him to reduce his carbon fingerprints and the costs of the bills.

Smart forest is a web application built with Vue.js and Nuxt frameworks. Vue.js is a JavaScript framework which allows to create user interfaces. The reasons why it has been chosen to use Vue.js are its flexibility, and its simplicity. Nuxt is a framework built on top of Vue.js which provides additional features for the server-side of the application. By using Vue and Nuxt, *Smart Forest* has achieved a responsive user interface, even if the Raspberry Pi is not so powerful.

4 Target users, needs and goals

Smart Forest has been designed for a wide range of users, including families, couples, and individuals. The platform caters to both children and adults, with

its interactive game component appealing to kids, and its data analysis and environmental advice features being beneficial to grown-ups.

4.1 Needs

This project aims to address several key **needs** in the community.

One of the primary objectives is to increase public awareness and education on important climate-related issues.

Secondly, this tries to help individuals and families save money on their energy bills through cost-effective solutions.

Additionally, the project seeks to provide entertainment and engagement through the integration of a fun and interactive game.

Lastly, the project aims to enhance the overall living experience by providing access to smart and stylish pieces of furniture.

4.2 Goals

This project aims to achieve several objectives for its users. The primary **goal** is to enhance the user's experience by creating an interactive and enjoyable interface with the *MagicMirror*[2]. Then, *Smart Forest* provide helpful suggestions and encourage conscious consumption through the analysis of household appliances usage. It also aims to provide users with real-time information on energy consumption and to raise awareness about environmental issues.

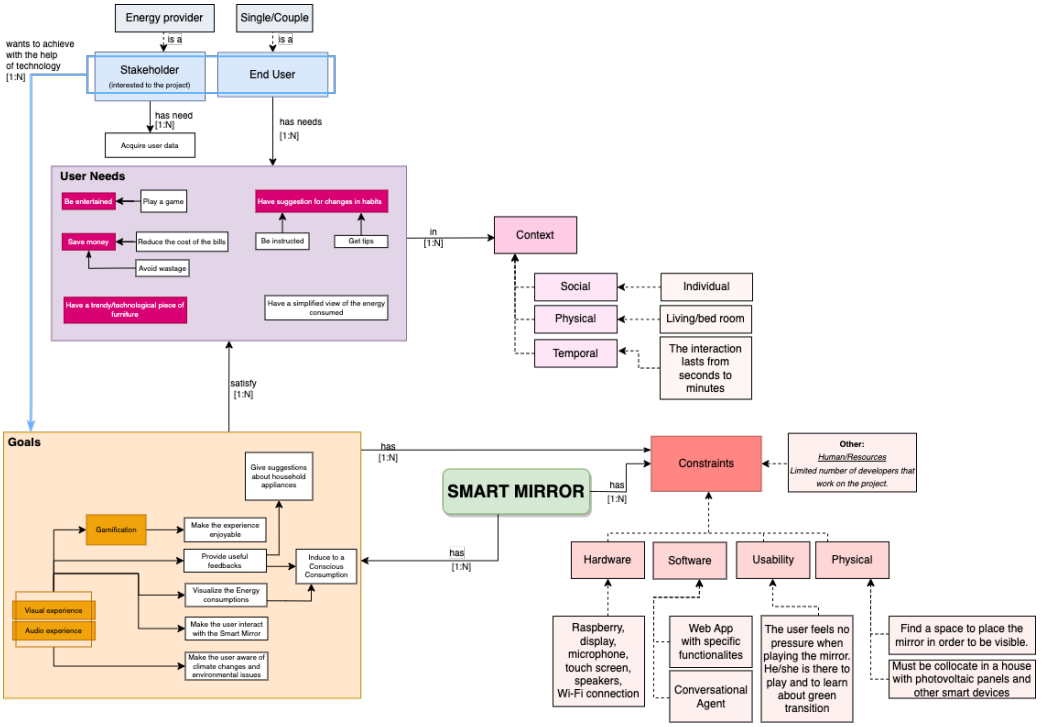


Figure 1: UNG

4.3 Scenario summary

User(s) profile	Families, couple and singles
Needs	Be entertained - Play a game Save Money - Reduce the costs of the bills - Avoid Wastage Have suggestion for changes in habits - Be instructed - Get tips Have a trendy/technological piece of furniture
Goals	Gamification + Visual and Audio experience - Make the experience enjoyable - Make the user interact with the Smart Mirror - Provide useful feedback - Visualize the energy consumptions - Make the user aware of climate changes and environmental issues
Constraints	Physical - Find a space to place the mirror in order to be visible - Must be collocate in a house with photovoltaic panels and other smart devices Hardware - Raspberry, display, microphone, touch screen, speakers, WiFi connection

Constraints	Software - Conversational Agent - Web App with specific functionalities Usability - The user feels no pressure when playing with the mirror. He/she is there to play and to learn about green transition Human/Resources - Limited number of developers that work on the project
Context	Social - Individual Physical - Living room / Bedroom Temporal - The interaction will lasts from seconds to few minutes
Tasks	The user can always observe the status of the forest. The user directly interacts with the mirror through a voice interaction. The user can ask: - about the status of the forest. - about suggestions to improve the forest. - to buy a new plant. - to upgrade a set of plants. - the strategy to follow in order to gain more and more leaves.

5 Solution

The idea behind *Smart Forest* is to take an everyday object, such as a mirror, and give it added functionality. In order to achieve this, npm (node package manager for JavaScript) was installed on the mirror. Additionally, to provide a seamless user experience, the web interface was built using Vue.js. This allows the mirror to not only reflect one's image, but also provide useful information and features. The key point for *Smart Forest* is to make people aware about climatic issue by creating a virtual forest which grown with the responsible behavior of the consumer.

5.1 Software architecture

5.1.1 Vue.js

Vue is a JavaScript framework designed to be easy and flexible. In the following list are reported some of the most important Vue components.

- **Trees.vue**: This component govern every action that trees can make, such as the spawn of a new Level-1 tree.
- **Forest.vue**: This component aims to manage the whole background of the interface, such as the change of the context from "sunny to stormy" and vice versa.
- **VoiceDetectionWidget.vue** and **SpeechRecognition.vue**: These components controls everything about the voice recognition, from the capture of the sound till the actual fulfillment of the task.

5.2 Conversational Agent - DialogFlow

For the conversational agent, the choice has been DialogFlow for its simplicity and scalability.

5.3 Intents

In DialogFlow, an *intent* represents a mapping between what a user says and what action should be taken in order to accommodate the request.

Each intent within **Flora**, the conversational agent built in the project, can be grouped into one of three main categories: **advice**, responsible for providing information and suggestions in order to encourage the user to be more responsible, **forest management**, responsible for handling tasks related to the management of the forest, such as plant new trees or merge trees, and **guide**, responsible for providing information about the usability of the mirror, such as a tutorial.

In the following list some of the most important intents. 9

5.3.1 Advice

- `advice_general`: this intent is triggered when the user asks something about the status of the house or if he wants to receive some tip in order to reduce his carbon footprint.
- `advices_get_more_leaves`: this intent is triggered when the user requests specific information about collecting leaves.
- `advices_start_specific_appliance`: this intent is triggered when the user inquires about the consequences of activating a particular device at the present time.
- `advices_appliances_consumption`: this intent is triggered if the user is interested in identifying an appliance that is using excessive energy.

5.3.2 Forest management

- `forest_management_buy`: this intent is triggered when the user wants to buy a new Level-1 tree.
- `forest_management_buy_position`: This intent is triggered after the previous one, if the back-end controls return a positive result, and it is responsible for determining the location of the new plant.
- `forest_status_overall_levelExperience`: this intent is triggered when the user wants to know the experience point of the whole forest.

5.3.3 Guide

- `guide_general`: this intent is triggered when the user wants to know how to play the game.
- `guide_group_plant`: This intent is triggered if the user wants to know how to merge the trees.
- `guide_strategy`: this intent is triggered when the user is interested in identifying the most environmentally efficient strategy.
- `guide_plant`: this intent is triggered when the user wants to know how to plant a tree.

5.4 *Smart Forest* API

Ideally, the smart mirror should be connected to the Edison servers in order to monitor all the consumption of the home. This would allow for real-time monitoring of energy usage and provide insights into how to reduce consumption and save on energy costs. By connecting to the Edison servers, the smart mirror would have access to data on electricity information on renewable energy sources such as the user's solar panel.

For simplicity in the presentation stage, an artificial API has been used. This API has been created with real values for home consumption, allowing for a demonstration of the smart mirror's capabilities without the need for a physical connection to the Edison servers. This approach would still provide a good representation of how the smart mirror would function in a real-world setting and would allow for testing and refining of the user interface.

5.4.1 Simulated House

This section focuses on the structure of the simulated house. The house is equipped with photovoltaic panels that can convert sunlight into electrical and a battery capable to store the energy. Photovoltaic panels convert energy in a weather-dependent manner according to the following formula:

$$energy = number\_panels \cdot energy\_panel \cdot k \tag{1}$$

The variable k depends on the weather condition ($k = 1$ for sunny day, $k = 0.3$ for cloudy day, $k = 0.1$ for rainy day) while energy panel is the maximum energy a panel can collect in an hour. The simulated house has 18 panels (with a maximum production capacity of $0.35kWh$ each) and a battery with a capacity of $10kW$.

In order to make the API as realistic as possible, the following appliances consumption were chosen:

Appliance	**Consumption**
Air conditioner	3.00kWh
Dehumidifier	0.07kWh
Cooker	1.04kWh
Dishwasher	1.20kWh
Dryer	3.50kWh
Boiler	24.00kWh
Oven	2.30kWh
Washing machine	1.30kWh

Due to the modular code it's easy to modify the home structure or to add new appliances just modifying the home.js file inside the model folder.

5.4.2 Weather

The weather can be:

- Sunny
- Cloudy
- Rainy

The weather is initialized randomly and it changes during simulation using a fuzzy logic function.

5.4.3 Simulate

One of the main functionalities of the API is to simulate one hour of consumption switching on/off the house appliances. During simulation, the API collects the energy from photovoltaic panels and tries first to use those energy for the appliances that are switched on and secondly, if the energy is still available, it starts charging the battery up the maximum battery capacity. It is considered *clean energy* the one coming from batteries or photovoltaic panels, while we define *dirty energy* the energy from any other source.

The API not only collects the amount of *clean* and *dirty energy* during the simulation but also considers the full history of the mirror.

The API outputs a total grade and a current simulation grade which are indicators of the user's clean consumption.

The grades are computed according to the following formulas:

$$CurrentGrade = \frac{clean_{energy}}{clean_{energy} + dirty_{energy}} \tag{2}$$

$$TotalGrade = \frac{1}{N} \sum_{i=0}^{N} CurrentGrade_i \tag{3}$$

5.4.4 Smart Forest interaction

In order to simplify the interaction with the API and to display some relevant data (weather, consumption...) a tablet is added to the mirror to display some relevant data such as weather and consumption.

It is also possible to switch household appliances on and off clicking on the relative button on the tablet.

When the user wants information regarding a specific household appliance, he/she asks flora which automatically makes the right call to the API which provides data.

Total grade and current grade indicators are used to update trees experience and to increment the number of leaves each time the simulation button is clicked.

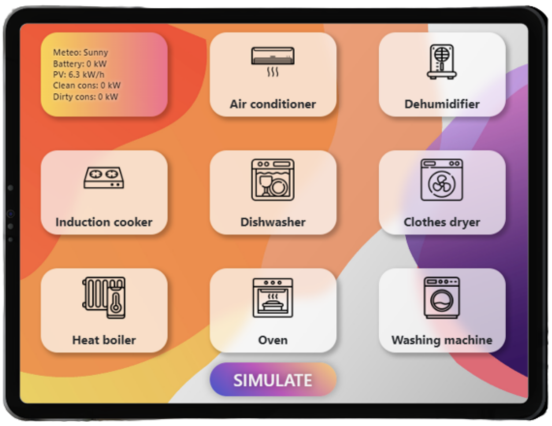


Figure 2: Tablet

5.4.5 API structure

The API has been developed with JavaScript. It is divided into 2 main parts:

- Model
- Router

The model folder contains different classes where the functionalities are implemented and data are stored. At each class is associated a Router class which manage the interaction of the relative endpoints.

5.4.6 API endpoints

HOME ROUTER

The API returns all the information about the house.
`https://smart-home-api-2j4i.onrender.com/home`
The API simulate one hour of consumes
`https://smart-home-api-2j4i.onrender.com/home/simulate`
The API reset all the data
`https://smart-home-api-2j4i.onrender.com/home/reset`

APPLIANCES ROUTER

The API returns all the information about the appliances in the house
`https://smart-home-api-2j4i.onrender.com/appliances`
The API returns all the information about the appliance with name passed as parameter
`https://smart-home-api-2j4i.onrender.com/appliances/name=:name`
The API turns on the appliance with name passed as parameter
`https://smart-home-api-2j4i.onrender.com/appliances/turnOn/name=:name`
The API turns off the appliance with name passed as parameter
`https://smart-home-api-2j4i.onrender.com/appliances/turnOff/name=:name`
The API returns the appliance which is consuming the most at the moment of the call
`https://smart-home-api-2j4i.onrender.com/appliances/mostConsuming`

BATTERY ROUTER

The API returns the state of the batteries
`https://smart-home-api-2j4i.onrender.com/batteries`
The API sets to zero the capacity of the batteries
`https://smart-home-api-2j4i.onrender.com/batteries/reset`

PHOTOVOLTAIC PANELS ROUTER

The API returns the energy collected by the photovoltaic panels (in kWh)
`https://smart-home-api-2j4i.onrender.com/panels/output`

WEATHER ROUTER

The API returns the current weather
`https://smart-home-api-2j4i.onrender.com/meteo`
The API simulate the weather using basic fuzzy logic
`https://smart-home-api-2j4i.onrender.com/meteo/change`

The API change the weather
`https://smart-home-api-2j4i.onrender.com/meteo/sunny`
`https://smart-home-api-2j4i.onrender.com/meteo/cloudy`
`https://smart-home-api-2j4i.onrender.com/meteo/rainy`

6 Game Rules

6.1 Tutorial

When the mirror is turned on for the first time a tutorial will be shown to the user, who will learn how to plant his first tree. At first the mirror will have 1500 leaves, but following the advice given by the mirror, the number of leaves will start growing.

6.2 Earn leaves

The amount of leaves earned each day is determined by both the actions taken that day and in the past. There are two scores: `current gain` and `total gain`. The `current gain` is based on actions taken in a specific time frame, while `total gain` considers all actions throughout the mirror's lifetime.

The amount of leaves is determinate by the following function:

```
totalGain = moneyGainFunction() +
            moneyGainTrees()*(p1 x 1 + p2 x 4 + p3 x 16)
```

where `moneyGainFunction()` is the function which measures the gained leaves due to the current and previous consumption, ignoring the number of trees on the forest, while `moneyGainTrees()` is a coefficient that will be multiplied by the number of trees in the forest, which are multiplied them self by the level coefficient, which is 1 for Level-1 trees, 4 for Level-2 trees and 16 for Level-3 trees.

6.3 Loose Experience

The way the user interacts with the environment can have a significant impact on the plants, both the real ones and the virtual ones in the mirror. If the user does not make an effort to practice eco-friendly habits, he may contribute to the acceleration of climate change. In this case, all the virtual plants will receive a decrease in experience points. If the experience points reach zero, the consequences can be terrible. In particular, if the plant which reach the zero is a Level 1, it will die, while if it is a higher level plant, it will demote to a lower level plant [Fig: 3].

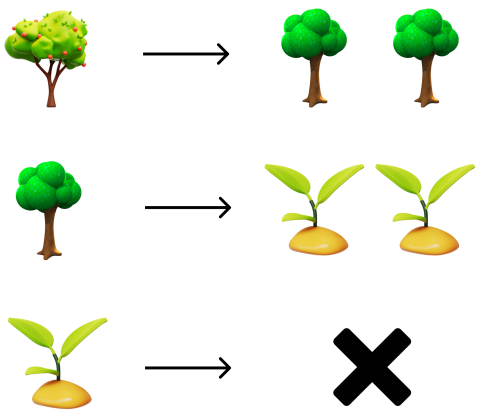


Figure 3: Plants downgrade

The amount of points lost is determinate by the following function:

```
experienceTreeLoss(){
    weighted = (0.08 x totalgrade + 0.02 x currentgrade) / 100
}
```

The function illustrates that the amount of points lost is strictly related to the `total gain`, with the intention of not heavily penalizing situations where weather conditions are unfavorable. For example, if it rains for several days, it wouldn't be reasonable to avoid using the washing machine simply because the solar panels are not producing much energy.

6.4 Flora

Flora is a conversational agent, built in with *Smart Forest*. Through interactions with it, it is possible to accommodate every request, like plant a new tree, merge some of them or ask advice.

6.4.1 Status of the forest

It is possible to see the status of the forest by asking Flora the following phrase: *"Can I see the condition of the forest?"*. A color will be shown in the background of each tree with the following code:

- **Red trees** are those whose experience level is very low, risking the downgrade or even the death.
- **Yellow trees** are those whose experience level is in the middle of the bar (i.e. from 30% to 70%).
- **Green trees** are those whose experience level is high (i.e. from 70% to 99%).
- **Purple trees** are those whose experience is at the maximum, these trees can be merged.

6.4.2 Plant a tree

A simple way to plant a tree is to ask Flora the following request: *"Can I plant a tree?"*. Upon receiving this request, Flora will perform a series of back-end controls to ensure that the request can be fulfilled and accommodate the user accordingly. If the request is approved, Flora will ask for the location of the new tree. The user can decide to place it at the top, bottom, right or left.

The cost of a new Level 1 tree is 1000 leaves, gained using the methods described in 6.2.

6.4.3 Merge some tree

Similarly, it is possible to merge groups of trees. Merge trees is useful in order to gain more and more leaves, in fact three plants, if merged, generate a new plant which makes the user gain like 4 plants of the lower level:

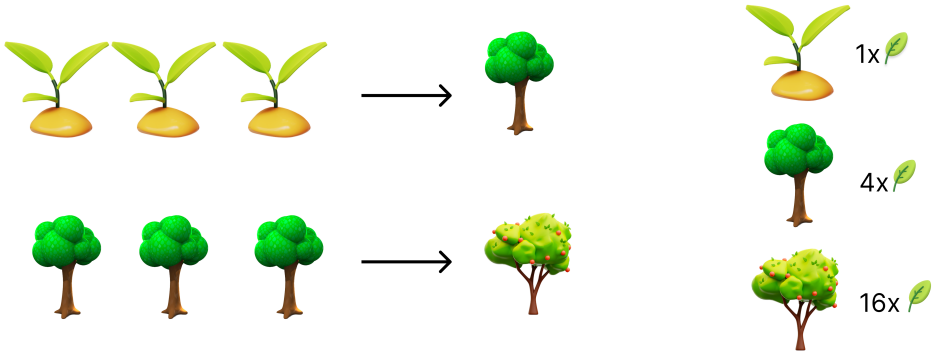


Figure 4: Level Gain

The question will be *"Can I group some tree?"*. Upon receiving this request and after all the back-end controls, the mirror will show the user which trees could be merge by identifying them highlighting them with a purple color as shown in the figure [5].

Figure 5: Highlighted tree

Only plants of the same level can be merged, in fact if there are not 3 purple trees of the same level available, Flora will return that it is not possible to merge any tree.

On the other hand, if the user chose the trees to group, Flora will fulfil the request merging them.

6.4.4 Get some advice

It is possible to ask Flora for advice on reducing one's carbon footprint, and this could be by posing a question like *"Can I get a little tip?"*. Flora will then provide suggestions on how to lower the user's impact on the environment. For instance, Flora can be asked whether it is environmentally friendly to run the washing machine at a certain time, and provide advice accordingly. By utilizing Flora's advice, individuals can make more informed decisions that contribute to a more sustainable future.

7 Story

Once upon a time, there was a man who was very environmentally conscious. He was always looking for ways to reduce his carbon footprint and live a more sustainable lifestyle. One day, he came across a new product called *Smart Forest*. It was a smart mirror that not only allowed him to see his reflection, but also gave him the ability to control and monitor the energy consumption of all of his appliances.

The man was excited to try out *Smart Forest*, so he purchased one and installed it in his living room. As soon as he turned it on, he was greeted by a beautiful forest on the mirror's screen. The forest was composed of three different types of trees: Level-1 trees, Level-2 trees, and Level-3 trees.

The man quickly learned that by using renewable energy sources, such as solar panel, he could earn *leaves* on the *Smart Forest*. These *leaves* could be used to purchase new Level-1 trees, which would be planted in the forest on the mirror. The more *leaves* he earned, the more trees he could plant.

As the man continued to make sustainable choices in his daily life, he soon had three Level-1 trees planted in the Smart Forest. Excitedly, he learned that he could merge these three Level-1 trees to create a Level-2 tree, which was even more beautiful and vibrant than the Level-1 trees.

The man was thrilled with the Smart Forest, not only because it helped him to be more energy efficient but also because it made him feel like he was doing something positive for the environment. He enjoyed watching the forest grow and change as he continued to make sustainable choices.

8 Future improvements

Working on this project has the potential to result in a more advanced and efficient product. Adding AI capabilities to the mirror, such as the ability to modify answers based on context, could make it even smarter. One AI that could be implemented is ChatGPT, but for the moment it is not fast enough for this type of work.

Aside from improving the AI, automating certain operations such as turning on or off appliances could also bring about significant benefits. These automation capabilities could optimize performance and ensure that the appliance is operating at its most efficient.

Additionally, there is always room for further advancements and improvements in technology. As technology continues to evolve, so too can this project. There may be

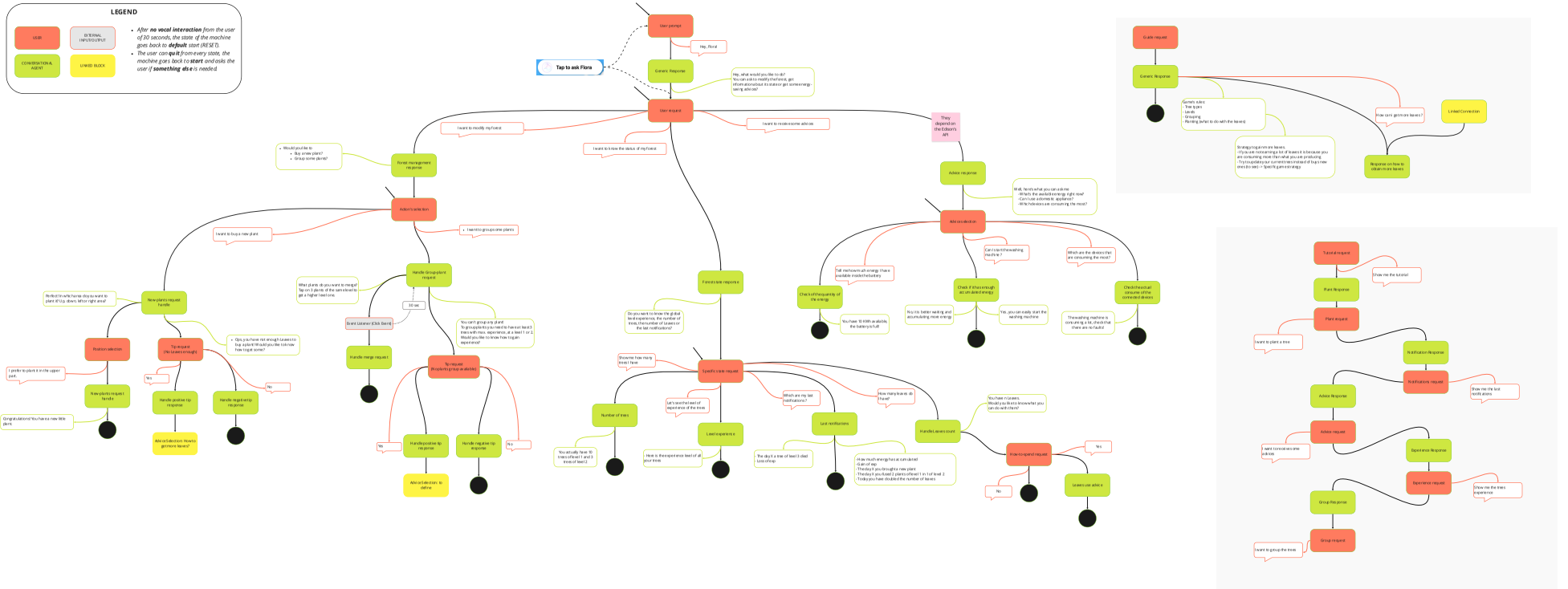
new technologies or approaches that can be incorporated to make the product even better.

Overall, there is a wealth of potential for this project to continue to grow and evolve. By staying current with advancements in technology and always looking for ways to improve, this project has the potential to become a valuable and innovative product.

> To demonstrate the potential of ChatGPT, the *Future improvements* paragraph [8] was automatically generated by the chat.

9 Bibliography

- MagicMirror documentation: `https://magicmirror.builders/`
- chatGPT: `https://chat.openai.com`


LEGEND
USER
EXTERNAL INPUT/OUTPUT
CONVERSATIONAL AGENT
LINKED BLOCK
After no vocal interaction from the user of 30 seconds, the state of the machine goes back to default state (RESET).
The user can quit from every state, the machine goes back to start and asks the user if something else is needed.
Tap to ask Flora
User prompt
Hey, Flora!
Generic Response
Hey, what would you like to do?
You can ask to modify the forest, get informations about its status or get some energy-saving advices
User request
I want to modify my forest
I want to receive some advices
I want to know the status of my forest
They depend on the Edison's API
Would you like to:
Buy anew plants?
Group some plants?
Forest management response
Action's selection
I want to buy a new plant
I want to groups some plants
Handle Group plant request
What plants do you want to merge? Tap on 2 plants of the same level to get a higher level one.
Event Listener (Click Event)
10 sec
You can't group any plant
To group plants you need to have at least 2 trees with max. experience, at a level 1 or 2.
Would you like to know how to gain experience?
Handle merge request
Tp request (Property group available)
No
Handle positive tip response
Handle negative tip response
Ask or Selections to define
Yes
No
New plants request handle
Perfect! In whe home do you want to plant it? Up, down, Which one right area?
Position selection
I prefer to plant it in the upper part.
Congratulations! You have a new little plant.
Tp request (No Leaves enought)
Ops, you have not enough Leaves to buy a plant! Would you like to know how to get some?
Yes
No
New plants request handle
Handle positive tip response
Handle negative tip response
Ask or Selection: How to germore leaves?
Forest's state response
Do you want to know my global level, my global number of trees, the number of Leaves or the last notifications?
Specific's task request
Show me how many trees I have
Number of trees
You actually have 10 trees of level 1 and 3 trees of level 2
Let's see the level of experience of the trees
Here is the experience level of all your trees
Level experience
Which are my last notifications?
Last notifications
The sky X a tree of level 3 died
Loss of exp
How many leaves do I have?
How much energy has accumulated in my circuit
Thanks to you through a new plant
Thanks to you fused 2 plants of level 1 in 1 of level 2
Thnks you have doubled the number of leaves
Handles Leaves count
You have n Leaves. Would you like to know what you can do with them?
How to spend request
Yes
No
Leaves use advice
Advice response
Well, here's what you can ask me:
-What's the available energy right now?
-Can I use a domestic appliance?
-Wich the devices are consuming the most?
Advice selection
Tell me how much energy I have available inside the battery
Check if the quantity of the energy
You have 10 KWh available, the battery is full
Check if I have enough accumulated energy
No, I to better waiting and accumulating more energy
Yes, you can easily start the washing machine
Can I start the washing machine ?
Which are the devices that are consuming the most?
Check the actual consume of the connected devices
The washing machine is consuming a lot, check that there are no faults!
Guide request
Generic Response
Generic asks:
- The types
- Levels
- Grouping
- Anything (what to do with the leaves)
How can I germ ore leaves?
Strategy to gain more leaves.
- If you are noticing a lot of leaves it is because you are consuming more than what you are producing.
- Try to postpone your current needs instead of buy new ones (to see) -> Specific game strategy
Response on how to obtain more leaves
Linked Connection
Tutorial request
Show me the tutorial
Plant Response
Plant request
I want to plant a tree
Notification Response
Notifications request
Show me the last notifications
Advice Response
Advice request
I want to receive some advices
Experience Response
Experience request
Show me the trees experience
Group Response
Group request
I want to group the trees