Chapter 5

Robot Manufacturing

This chapter details the manufacturing process that brought the robot from design to reality. It covers material selection, component sourcing, fabrication techniques like 3D printing and laser cutting, and the mechanical and electrical assembly. The chapter also addresses challenges faced and the modifications made to optimize the final product.

5.1. Materials and Components

The choice of materials and components is crucial in the manufacturing process, as it directly impacts the robot's performance, durability, and overall functionality. This section outlines the materials selected for the robot's construction and additionally covers the sourcing and procurement strategies employed to obtain the necessary components, ensuring that the project stayed within budget and met all design specifications.

5.1.1. Selection of Materials

The materials selected for the robot's construction were chosen based on their specific properties, ensuring that each component meets the necessary durability, flexibility, and functionality requirements. Below is an overview of the materials used and the justification for their selection:

- **PETG:** Most of the robot's components were 3D printed using PETG, a material that combines strength and flexibility, making it ideal for durable and functional parts. PETG's resistance to warping and superior durability and temperature resistance compared to PLA made it the preferred choice for the main structure, where both mechanical robustness and ease of printing were crucial.



Figure 5.1. Robot materials selection from RS Iberia and Amazon: (a) PETG 1.75 mm filament, (b) ABS 1.75 mm filament, (c) 2 mm transparent PVC panel, (d) 3 mm polystyrene panel, (e) 3 mm silicone sheet, (f) Anti-slip foam adhesives, (g) Nylon elastic textile mesh.

- **ABS:** It was chosen for the main neck mechanism due to its durability, impact resistance, and ability to withstand frequent movement and support the head.
- **Polystyrene:** A 3 mm polystyrene panel was selected for diffusing the LED light proceeding from the LED rings. This material ensures uniform light distribution, enhancing the visual impact of the robot.
- **Silicone:** It was chosen for the tactile pads due to its flexibility and soft texture. Silicone's durability and ability to withstand repeated use make it ideal for components that will experience frequent human interaction.
- **Anti-slip foam adhesives:** The lower part of the robot features anti-slip foam adhesives, which provide stability and prevent unwanted movement on smooth surfaces.
- **Nylon elastic textile mesh:** Used to cover the neck mechanism, providing both aesthetic appeal and safety with its flexibility and durability while withstanding repeated movement.
- **PVC panel:** For the robot's head, a 2 mm PVC panel was selected due to its flexibility to achieve a visually appealing finish, particularly important in the head where aesthetics is crucial.

5.1.2. Sourcing and Procurement

The sourcing and procurement was critical to the timely and cost-effective completion of the robot, ensuring that all components were acquired efficiently and within budget.

- **Component Selection:** Components were selected based on availability, cost, and quality, with a focus on sourcing from reliable suppliers such as RS Iberia, Amazon Business and Mouser. Preference was given to components that offered the best balance of performance and price, while also ensuring they met the technical requirements of the design.
- **Supply Chain Management:** Components were ordered well in advance to ensure sufficient time for manufacturing. Most of the components were purchased at the beginning of the project, with additional smaller orders made as needed for extra components as the project progressed. This proactive approach helped to avoid delays and ensure that all necessary materials were available when required.
- **Budget Considerations:** The procurement process was closely tied to the project's budget constraints. Regular cost reviews were conducted to ensure that expenditure remained within the allocated budget, without compromising the quality or functionality of the robot.

5.2. Manufacturing Process

After building the initial SPM neck mechanism prototype to qualitatively test the required torques to assist the motors selection process, the definite version of the SPM neck mechanism was the first thing to build.

5.2.1. SPM Mechanism, Support Structure and Head Panels

The pieces of each of the four arms were printed and then the whole mechanism subassembly was built inserting the roller bearings in the pieces. Figure 5.2 shows the printed SPM mechanism.

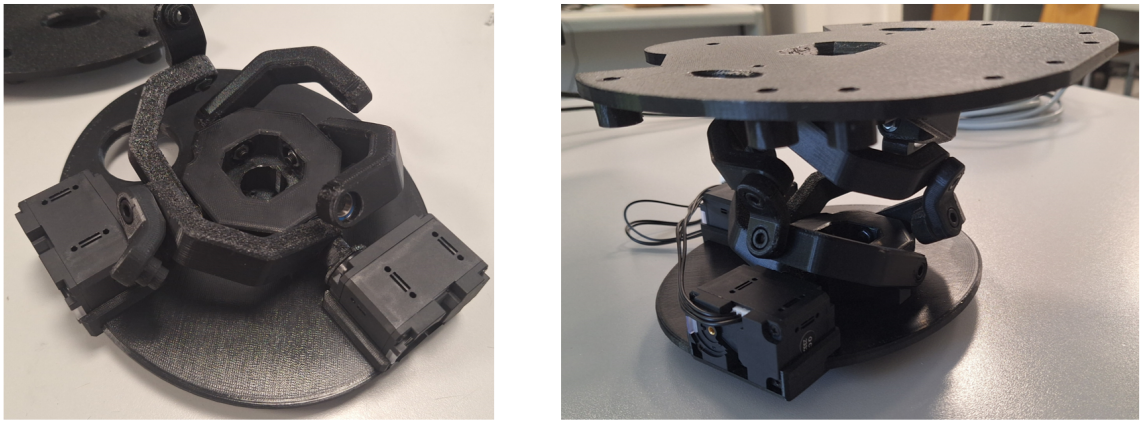
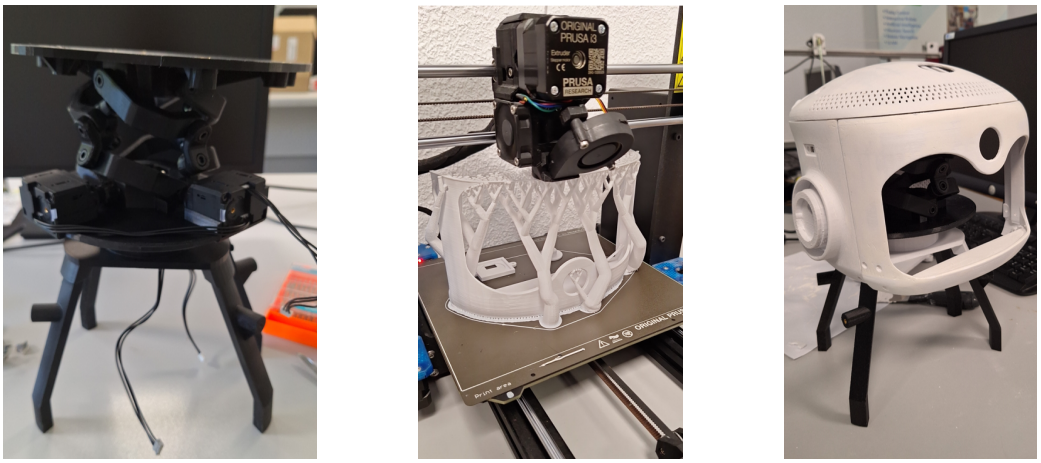


Figure 5.2. 3D printed 2-DOF SPM neck mechanism.

Next, the support structure and the head external panels, antennas and ports panel were printed. The structural elements were printed with black PETG, while the external panels were printed in white PETG. The 3D printing process was monitored through the Prusa Link tool. Once the support structure was printed, it was assembled with the neck mechanism as it can be seen in Figure 5.3a.



(a) **(b)** **(c)**

Figure 5.3. 3D printed parts: (a) Support assembled with the neck mechanism, (b) Front panel during printing process, (c) Head external panels.

The next step was to **post-process** the external the components to improve their appearance. To achieve this, the pieces were first gradually sanded to smooth out the surfaces and minimize the visible layer lines from 3D printing. Putty was then applied to fill in any remaining imperfections and further reduce the appearance of defects, as it can be seen in Figure 5.4a.

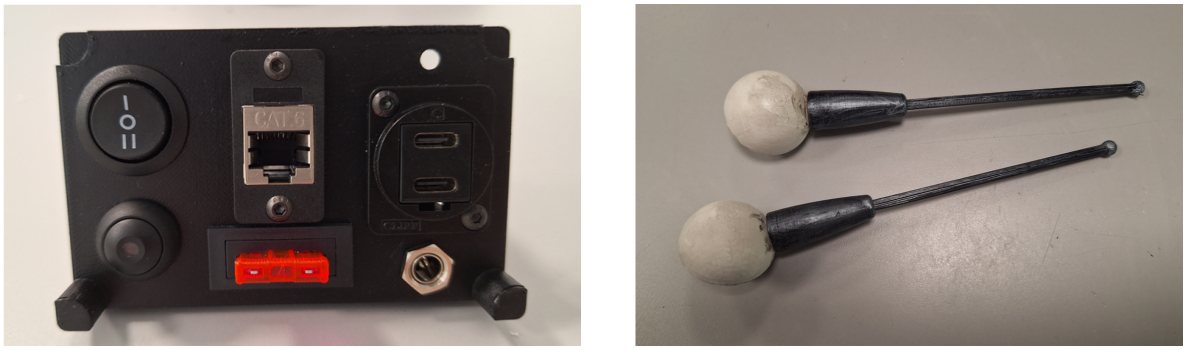
Once the surfaces were adequately prepared, the pieces were primed as it can be seen in Figure 5.4b to create an even better base for the final paint finish. Finally, a couple of layers of paint were applied to achieve the desired aesthetic and ensure a smooth, professional look, as shown in Figure 5.4c. In the end it was decided to paint the robot with a dirt wash style to look it make older and give it a more characteristic style.



(a) **(b)** **(c)**

Figure 5.4. Post-processing steps: (a) Putty application on the front and top head panels, (b) Priming process, (c) Painting process.

After the ports panel piece was printed and painted, the different switches, ports and buttons were assembled onto the panel as it can be seen in Figure 5.5a. The antennas were also assembled by threading the spherical pieces with the antennas' bases.



(a) **(b)**

Figure 5.5. 3D-printed parts: (a) Ports panel, (b) Antennas.

5.2.2. Arduino Proto Shield Connections

To test the designed SPM neck mechanism with the head load, the next step was to **manually wire** the protoshield as planned in the Inventor electronic layout. This required meticulous work to achieve a compact and organized cable arrangement. Figure 5.6 shows the protoshield with the connections that allow to use the Dynamixel shield mounted on the Arduino Mega. The cables hanging from one side are power terminals intended for connection to the ports panel. Additionally male pins were added to the protoshield to facilitate easy plugging and unplugging of the connected sensors and actuators.

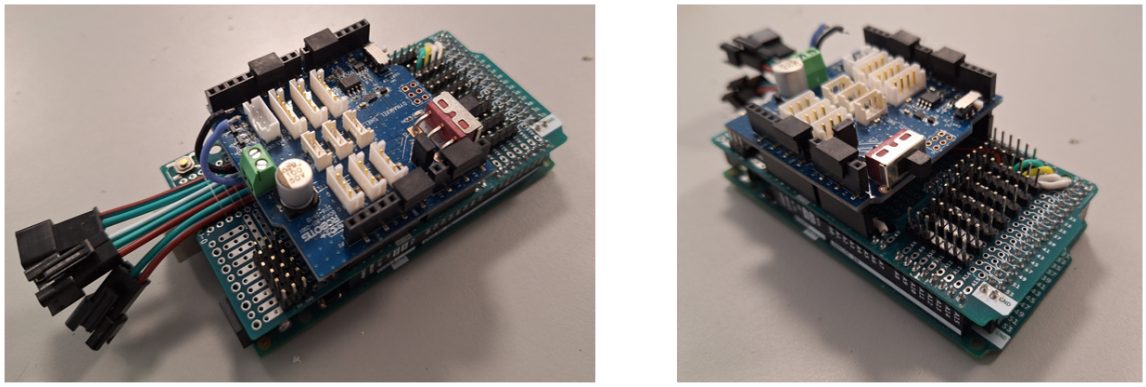
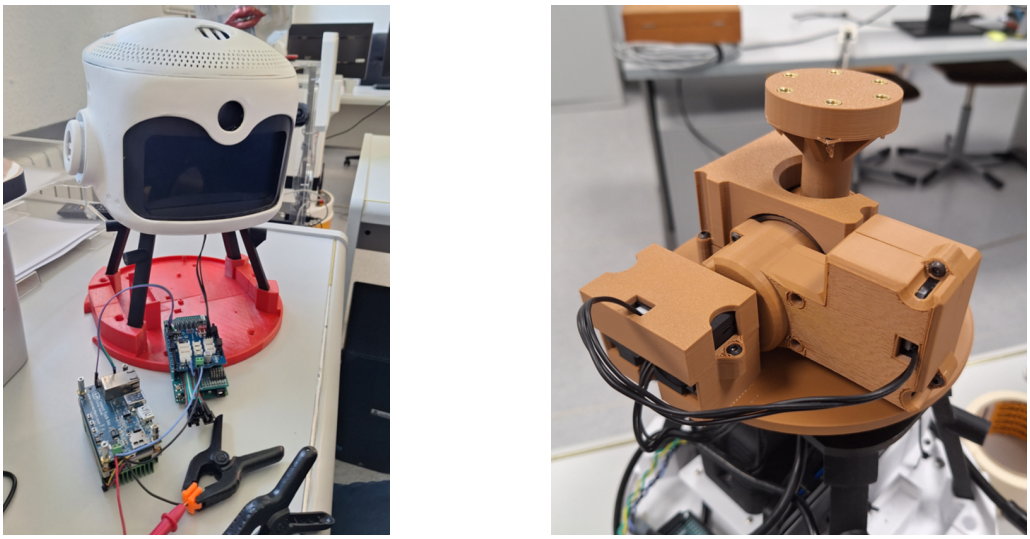


Figure 5.6. Arduino with mounted partially connected protoshield and Dynamixel board.

5.2.3. Spherical Gear Mechanism

Due to a shortage of PETG filament caused by some print errors and iterative design changes, PLA was used to provisionally to print a base for testing the neck mechanism

moving the mounted head, as it can be seen in Figure 5.7a. During testing, it was observed that the mechanism **could not lift the head** (adjust the pitch angle) with the motor at its maximum designed position. This led to the conclusion that the issue was not with the motors, but with the mechanism itself. If manufactured from a more rigid material, such as aluminum, the mechanism would have more effectively transmitted the motor movements to the head's upper platform. However, since it was made of plastic, each piece bent slightly, and minor play resulted in the head being positioned lower than desired, even with motors correctly positioned, due to the head's front panel weight.



(a) **(b)**

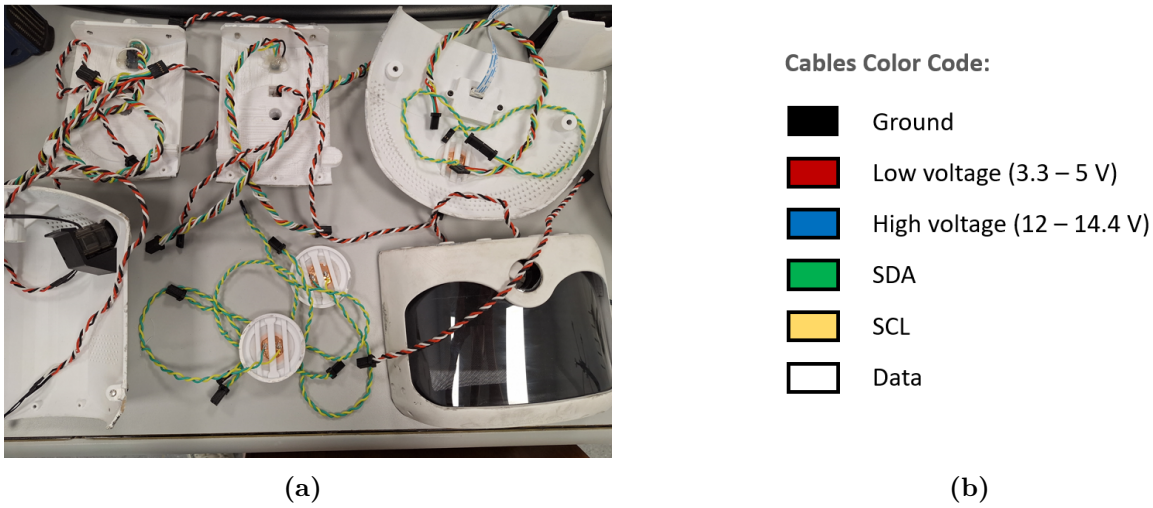
Figure 5.7. Neck mechanism: (a) SPM neck mechanism testing setup, (b) 3D printed spherical gear neck mechanism.

The issue prompted a **redesign**, leading to the development of the spherical gear mechanism presented in the previous chapter, which addressed these challenges and significantly improved the design. This mechanism was 3D printed in ABS and is shown in Figure 5.7b.

5.2.4. LEDs' Covers, Head Cables, Tactile Sensors and Modules

Next, the head front panel was assembled with the LCD display and the PVC sheet was cut and bent to adapt to the face shape, as it can be seen in Figure 5.8a. Between the LCD display frame and the PVC layer, a textile mesh layer was attached to make the color difference between the display background and the frame less noticeable.

Then it was decided to start making the connection cables for the head hardware components. In order to manufacture this home-made custom-length cables, a color code was established for easing cable identification once the whole robot is assembled. These color cables were cut with the needed length, crimped and then attached to DuPont-



(a) **(b)**

Figure 5.8. Head wiring: (a) Head panels cable connections, (b) Cables color code.

type terminals. 3D-printed pieces were used to group several individual DuPonts in one single terminal. It was decided that to maintain the robot's modularity the head cables would only arrive to the neck height, and then extra cables would link these to the microcontroller. The cables were manufactured according to the connections diagram of Figure 4.31a and twisted to minimize electromagnetic interference. Figure 5.8b shows the used cable color code.

The polystyrene sheet was then cut with FabLab's laser cutting machine to obtain the LED rings' covers. These were attached to the head components using a silicon gun as shown in Figure 5.9. Also the forehead and ear tactile sensors were manufactured. In order to do this, two cables were connected to a copper sheet stuck to a silicon circular pad.

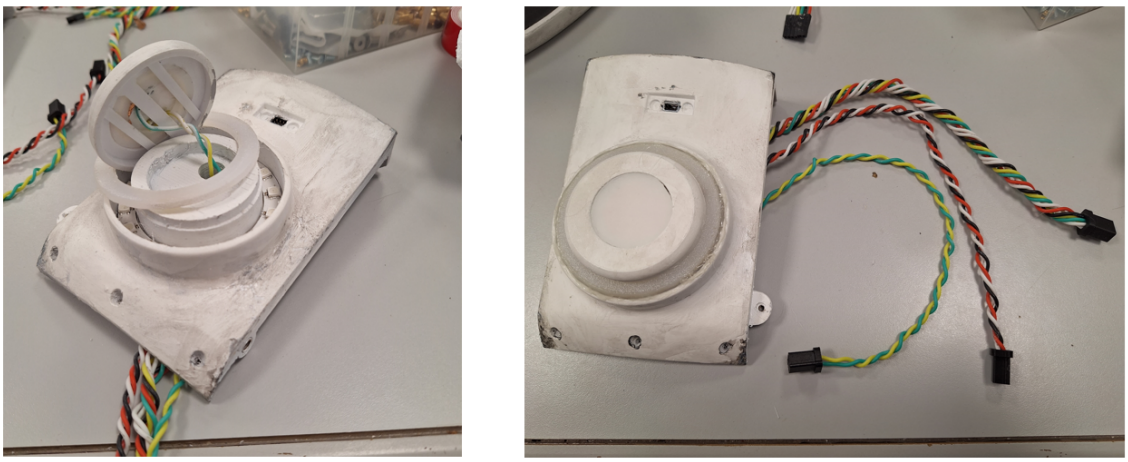


Figure 5.9. Head lateral panel assembly and connections.

The additional modules are designed to be connected through the male connectors embedded in the lateral panels. Each connector has 6 pins: power supply, ground, SCL

and SDA for I2C communication and one analog and one digital GPIO ports. The cables for these modules can also be seen in Figure 5.9. The flashlight and gas detector modules were also 3D printed and assembled. Figure 5.10 shows the manufactured air quality and gas detector module.

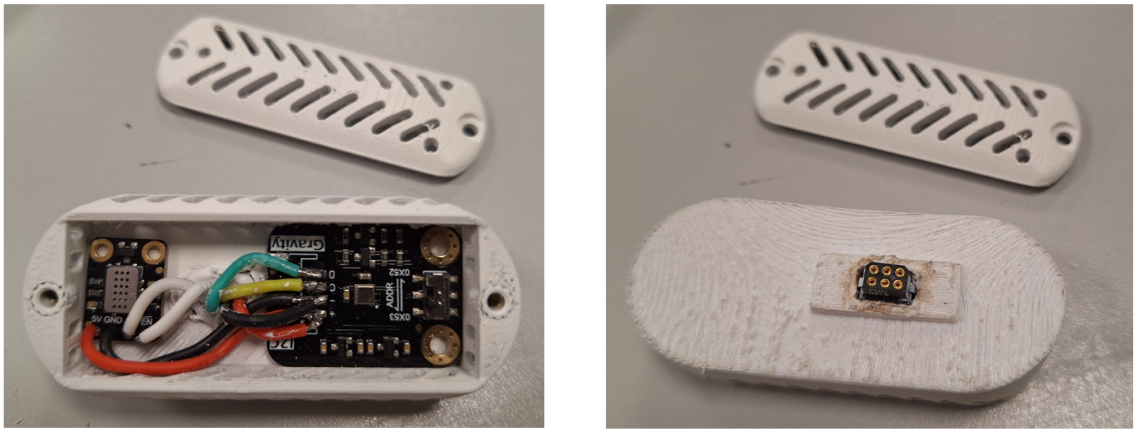


Figure 5.10. 3D printed air quality and gas detector module.

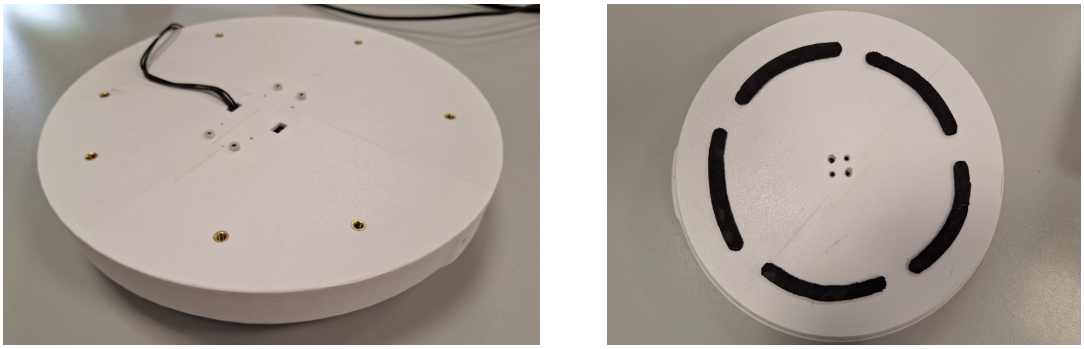
To finish with the head, the connectors were tagged in order to facilitate the connection between the body and head cables. Then, the whole head was assembled passing the cables through the holes designed for this purpose and grouped by mean of a textile tube as shown in Figure 5.11.



Figure 5.11. Cabled 3D printed head sub-assembly.

5.2.5. Base

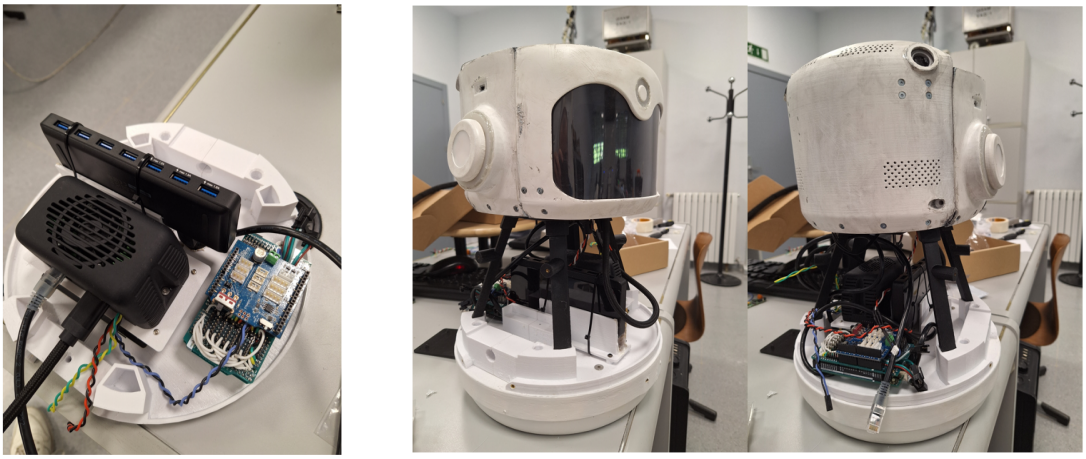
Once the new PETG filament arrived, the base sub-assembly components were 3D printed. Once the printing process was completed, the base was assembled as shown in Figure 5.12a, introducing the brass threaded inserts and the corresponding bearings. Finally, the anti-slip foam adhesives were cut and stuck to the lower part of the robot' base as shown in Figure 5.12b.



(a) (b)

Figure 5.12. 3D printed base: (a) Top view, (b) Bottom view.

Then, the body support base was printed and threaded to the base, enabling the assembly of the complete robot internal structure. Once the Arduino protoshield connections were finished, the SSD together with the Jetson TX2i, Arduino block and USB hub were fixed to this base as it can be seen in Figure 5.13a.



(a) (b)

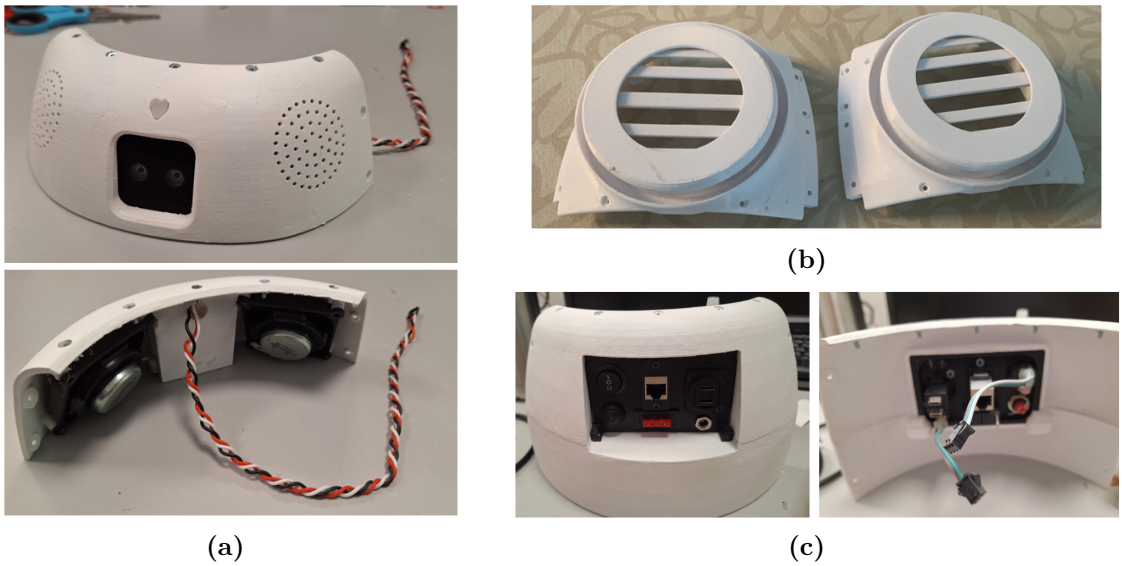
Figure 5.13. Support and internal structure: (a) Body support base, (b) Front and back view.

5.2.6. Body

The final 3D prints included the front, back, and lateral casings of the robot's body. The lateral panels and covers are magnetically attached for easy access. To embed the magnets during printing, the printer was programmed to pause at the layer just before covering the holes designed to hold the magnets. This allowed the magnets to be inserted, after which the printing continued, securing and integrating them into the structure. The same process was followed for the head lateral panels and the back ports panel cover.

After the pieces were post-processed, the heartbeat LED polystyrene cover was glued to the front panel, as shown in Figure 5.14a. The speakers were also mounted onto the

front panel, and the remaining body cables were fabricated and installed. Custom silicone touch pads were created and adhered to the lateral covers depicted in Figure 5.14b, in a similar way to the head ear covers. Lastly, the ports panel was affixed to the back body casing, as illustrated in Figure 5.14c.



(a) **(b)** **(c)**

Figure 5.14. 3D printed body panels: (a) Front body panel, (b) Lateral panels, (c) Back body panel.

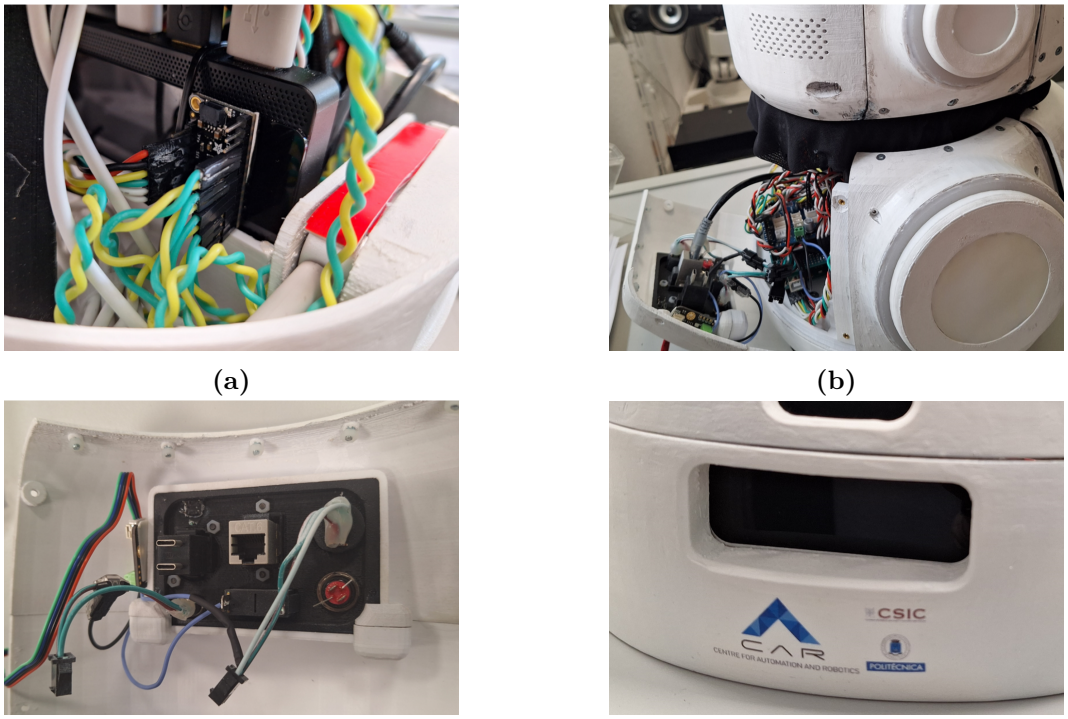
5.2.7. Final Details and Assembly

As the robot neared completion, several final details were carefully addressed to ensure both functionality and a polished appearance. The **installation** of the **missing sensors** and click modules was one of the last steps, with each component securely attached to the robot's body using double-sided tape, as shown in Figure 5.15a for the Adafruit MPR121. This method provided a reliable yet adjustable attachment.

The **elastic textile mesh** was then fixed to the head and body of the robot. It provided a clean and cohesive look by concealing internal components, while also enhancing safety by preventing fingers from accessing moving parts, as depicted in Figure 5.15b.

Final wiring connections were made, particularly in the **ports panel**, to ensure all electrical systems were fully operational. This included setting up mode switching between battery power and plugged-in operation, integrating fuse protection and wattmeter measurement, and securing all wires to prevent any loose connections, as illustrated in Figure 5.15c.

The last aesthetic touch was the application of a **waterslide decal** featuring the **CAR logo** on the front panel of the robot. After carefully positioning the decal, it was varnished to protect it from wear and enhance its durability, as shown in Figure 5.15d.



(a) (b)

(c) (d)

Figure 5.15. Final manufacturing details: (a) Adafruit MPR121 installation, (b) Elastic textile mesh installation, (c) Ports panel wiring connections, (d) Logo decal.

With these final details completed, the robot's final assembly was conducted by threading the head sub-assembly to the spherical gear of the neck mechanism. The completed robot, with the head attached, is depicted in Figure 5.16b. This marked the culmination of the manufacturing phase, successfully integrating all design, fabrication, and assembly efforts into a cohesive and operational social robot.



(a) (b)

Figure 5.16. Final assembly steps: (a) Headless robot assembly, (b) Final robot assembly.

Chapter 6

System Architecture and Firmware Development

This chapter provides an overview of the assistive robot's system architecture and firmware development. After outlining the key components of the system architecture, the chapter provides an overview of the firmware development process focusing on its architecture design, file structure, core functions, and the integration of third-party libraries and hardware components.

6.1. System Architecture Overview

The system architecture of the assistive robot is designed to enable complex emotional and empathetic interactions. This architecture is structured into **four layers** which can be seen in Figure 6.1, each serving a specific function while ensuring seamless integration between the robot's hardware, firmware, software, and input-output interfaces.

6.1.1. Software Layer

At the heart of this architecture lies the software layer, which orchestrates the robot's **high-level functionalities**. This layer is pivotal for managing the robot's interactions, ensuring that its responses are both emotionally aware and contextually relevant. This software layer was **inspired by the architecture used in Potato**.

In Potato, a socket communication system was employed to integrate its six modules, ensuring efficient coordination through a centralized state manager. Similarly, the soft-

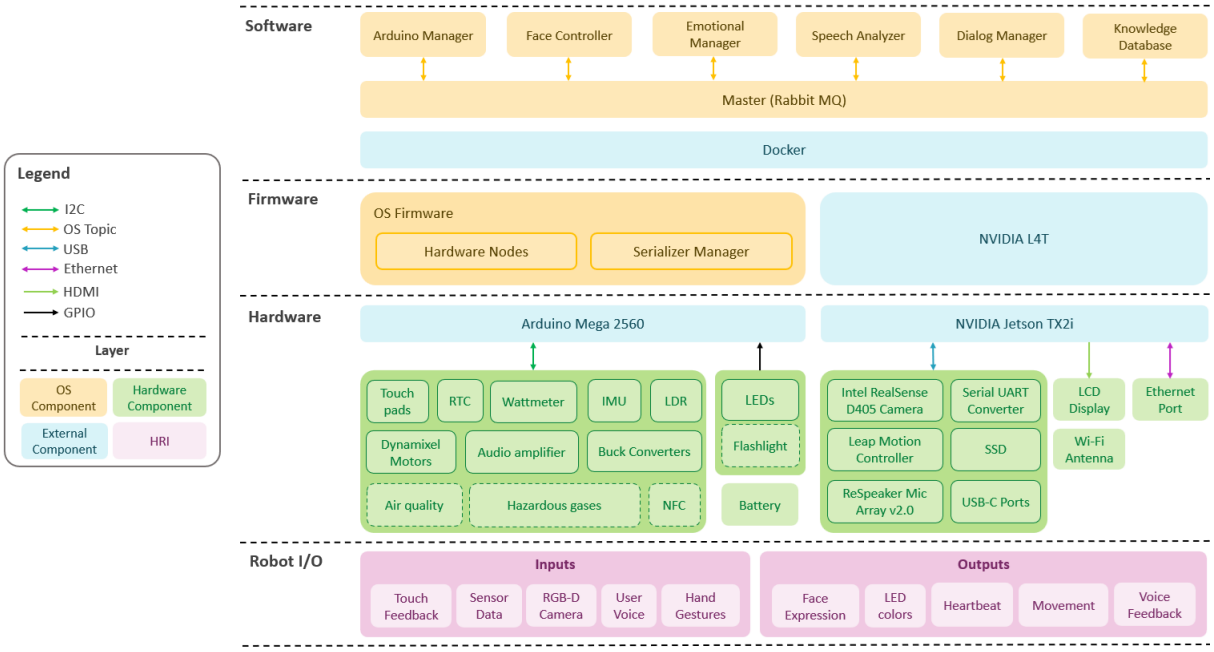


Figure 6.1. Robot general architecture.

ware layer of the robot built in this work adopts a **modular approach**, but with an enhancement in the form of **Docker-based containerization** developed in a different Master's Thesis. Each module, including the Arduino manager, face controller, emotional manager, speech analyzer, dialog manager, and knowledge database, operates within its own Docker container. These containers are orchestrated by a master module using Docker Compose, which coordinates the interactions and data flow between them. The modules are interconnected through the **AMQP protocol**, using RabbitMQ, enabling seamless communication and real-time processing. This setup is analogous to how ROS operates, where **each module functions as a node** that **publishes and subscribes to messages within topic**s, ensuring efficient and organized data exchange.

This advanced setup, building on Potato's successful design, allows for more **scalable, flexible, and maintainable software** architecture. The use of Docker ensures isolation of modules, facilitating independent development, testing, and deployment, while Docker Compose streamlines the orchestration of the entire system. In the case of the **validation demonstration**, an application will be built in a **new container** that interconnects with the rest of modules.

6.1.2. Firmware Layer

Beneath the software lies the firmware layer, acting as the critical **bridge between software commands and hardware actions**. The Jetson TX2i leverages the external firmware provided by NVIDIA, known as NVIDIA L4T (Linux for Tegra). This firmware includes a Linux kernel, bootloader, and various drivers, facilitating the integration of

high-level software with the underlying hardware. For the Arduino, a **custom firmware** was developed specifically for this robot. This firmware is structured around a **node-based architecture**, allowing for modular control of the different robot components.

To ensure effective communication between the Arduino Mega and the Jetson TX2i, a **serializer manager** was implemented within the custom firmware. This manager handles the serialization and deserialization of data, processing communications between the two systems. It ensures that commands from the Jetson TX2i are accurately executed by the Arduino Mega, and that the sensor data from the Arduino is correctly transmitted back for higher-level processing.

6.1.3. Hardware Layer

The hardware layer encompasses the **physical components** that enable the robot to interact with its environment. The **Arduino Mega** 2560 microcontroller manages **low-level operations**, interfacing directly with sensors and actuators, while the **NVIDIA Jetson TX2i** handles more **complex processing tasks** such as emotion detection and speech analysis. This hardware setup provides the necessary computational power and control to execute the robot's functions effectively.

6.1.4. Robot I/O Layer

Finally, the robot I/O layer integrates the input and output interfaces that **connect the robot to its external environment and users** according to the designed HRI. Input interfaces such as touch sensors, microphones, and cameras allow the robot to **gather data** about its surroundings and user input, facilitating real-time responsiveness. Output interfaces, including LED indicators, speakers, and the display screen, enable the robot to **express emotions and communicate effectively** with users. This layer ensures that the robot can perform **meaningful interactions**, translating the processed data from the higher layers into physical actions and sensory outputs that align with the robot's intended functions and user engagement tools.

6.2. Firmware Development

The custom robot C++ firmware aims to provide the necessary control mechanisms to allow the hardware to operate effectively under the software's guidance. Given the complexity of the robot's tasks, the firmware must be robust, flexible, and capable of real-time processing. This section delves into the proposed solution and specific functionalities to seamlessly integrate with both the hardware and software layers of the robot.

6.2.1. Firmware Architecture and File Structure

This section provides a detailed overview of the firmware's architecture and its file structure, offering insight into how the firmware components are integrated.

6.2.1.1. Firmware Architecture

The firmware is structured to follow a **modular and node-based design**, where each node represents a distinct component or function within the robot, either sensors or actuators, as it can be seen in Figure 6.2. This architecture allows for the firmware to be highly flexible and scalable, enabling the addition of new modules or the modification of existing ones without affecting the overall system.

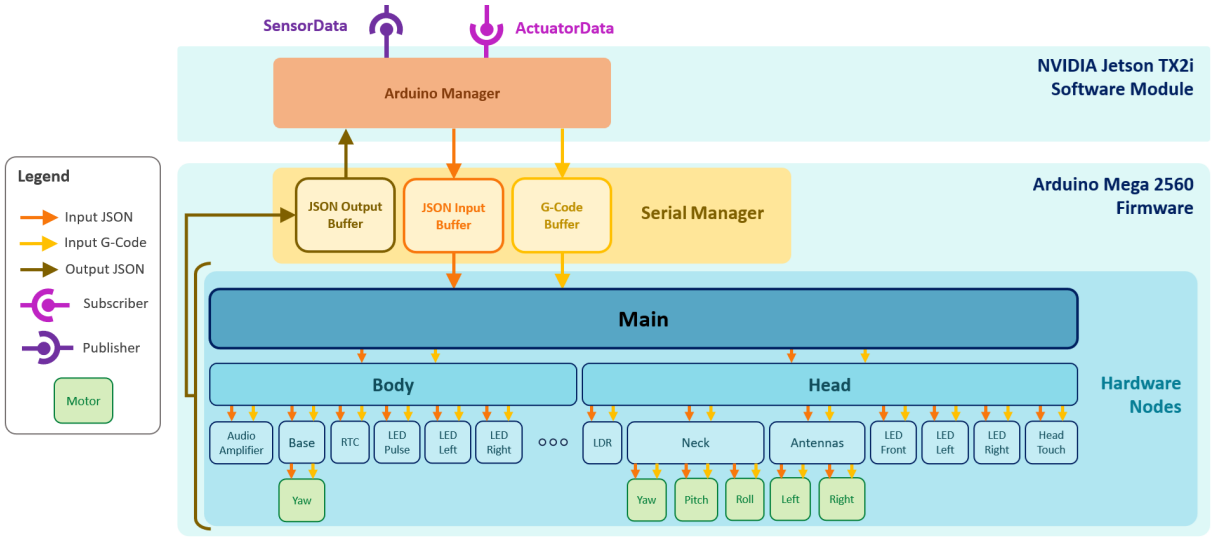


Figure 6.2. Firmware architecture.

As it can be seen in the figure, the firmware is composed of two main parts: the **hardware nodes** and the **serial manager**. The hardware nodes are hierarchic, being the main node composed of two sub-nodes: body and head. This way, each node can have as many sub-nodes as desired. The nodes are programmed by using a node abstract class, serving as base to create the rest of the nodes. Special types of nodes can be created to instantiate them as sub-nodes in an easier way. For example, this is the case of the motor node, instantiated as sub-node for the base, neck and antennas nodes.

With respect to the serial manager, it is responsible for handling all serial communication between the Arduino Mega and the NVIDIA Jetson TX2i. This class manages the transmission and reception of data, including **JSON** and **G-code** messages, through two input and one output buffers. An example of both types of message format can be seen in Figure 6.3a and 6.3b respectively. The JSON format is flexible but might be slow to parse, G-Code on the other hand is efficient and concise but less flexible.

```
//  JSON example:
//    <{"potato.head":{"led_front":{"on":1},
                      "led_left":{"on":1},
                      "led_right":{"on":1}},
       "potato.body":{"led_left":{"on":1},
                      "led_right":{"on":1}}}
```

(a)

```
//  G-Code example:
//    ~M01 0; // Turn all LEDs on
```

(b)

Figure 6.3. LEDs on message example: (a) JSON format, (b) G-code format.

6.2.1.2. File Structure

The firmware's file structure is organized to support its modular architecture, with each module and its associated functionalities within specific files. This organization facilitates ease of development, debugging, and maintenance. Below is an overview of the key files and directories within the firmware:

- **`firmware.ino`:** It is the main entry point of the firmware, responsible for initializing the system, setting up nodes, and managing the main control loop. It orchestrates the overall operation of the robot as described in Algorithm 1. It is important to note that not all nodes are represented, as time constraints prevented the implementation of all the planned hardware devices.

Algorithm 1 Firmware Initialization and Main Loop

Require: Firmware configuration, SerialManager, MainNode
Ensure: Initialization, input processing, and node control
Setup:
Initialize `serialManager` and `mainNode`
Start serial communication and motor system (if present)
Log firmware start and check all nodes are `UNCONFIGURED`
if nodes not `UNCONFIGURED` **then**
    Log error and exit
Configure and activate nodes
if any configuration or activation fails **then**
    Log error and exit
Log successful firmware start
Main Loop:
while loopCount is not zero **do**
    Process serial input
    if JSON or GCode received **then**
        Handle input and clear received flag
    Send JSON response, loop all nodes
    Track loop time and handle any overrun
    Decrement loopCount (if debugging)
Shutdown:
Deactivate, cleanup, and shutdown nodes
if nodes not in final state **then**
    Log error and exit
Log firmware exit and terminate

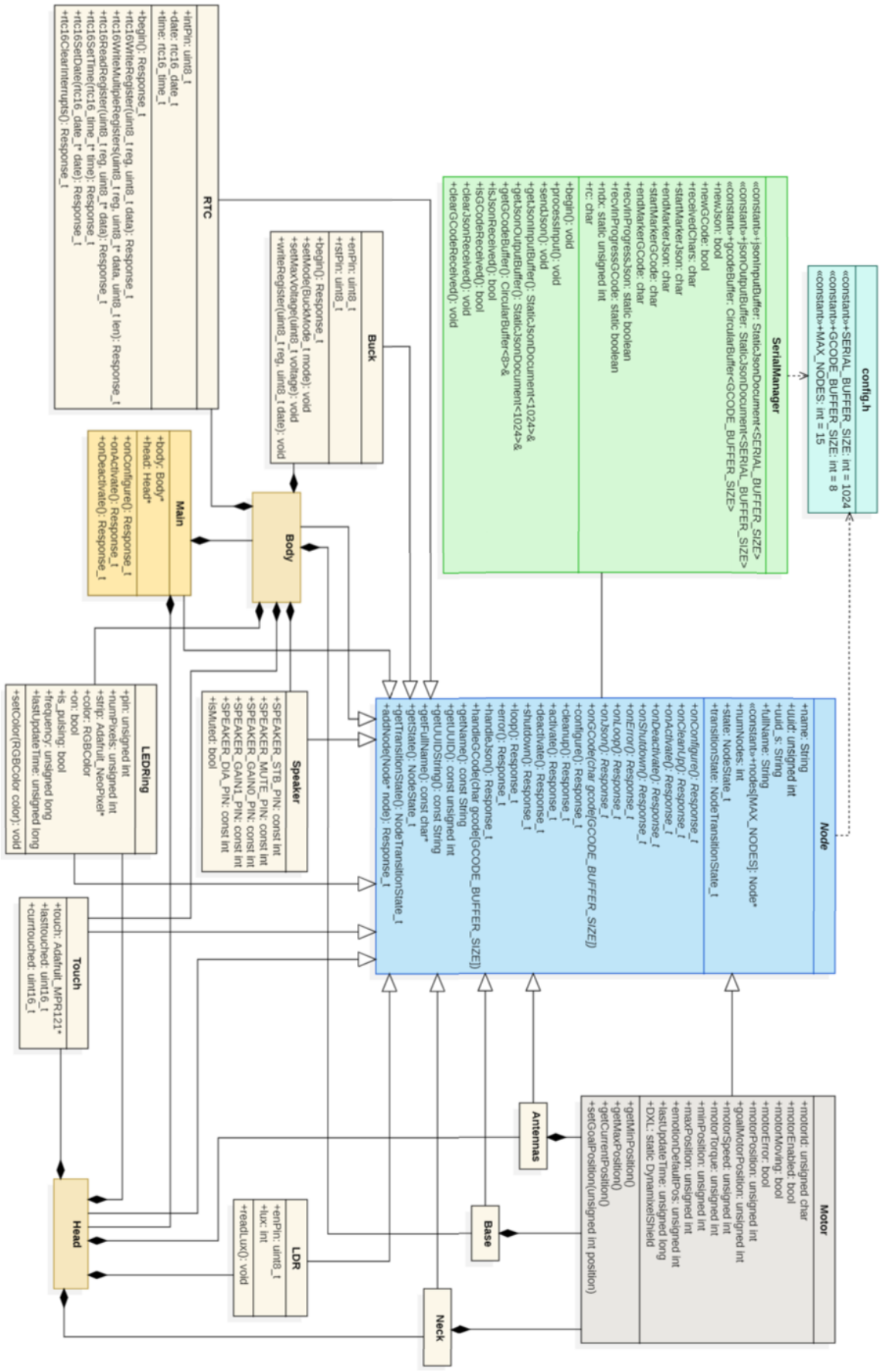


Figure 6.4. Firmware UML diagram.

- **`config.h`:** A configuration header file that defines global constants, such as pin assignments, serial communication settings, and node configurations. This file centralizes all configurable parameters, making it easy to adjust settings without modifying the core codebase.

- **`src/`:** This directory contains the core source files that implement the firmware's functionality.

  - **`node.hpp` and `node.cpp`:** Define the abstract `Node` class, which is the base class for all nodes in the firmware. This class manages state transitions and provides a framework for creating new nodes.
  - **`serial.hpp` and `serial.cpp`:** Implements the `SerialManager` class, handling all serial communication tasks, including parsing incoming messages and formatting outgoing data.
  - **`main.hpp` and `main.cpp`:** Define the `Main` class, which serves as the root node for the firmware. It aggregates all other nodes (e.g., body, head) and manages their interactions.
  - **`nodes/`:** This directory contains header and source files for specific nodes, each corresponding to a particular component of the robot (e.g., `body.hpp`, `body.cpp`, `led.hpp`, `neck.hpp`,...). Note that since this was an initial prototype version, due to time constraints it was not possible to define the full functionality of all the nodes, such as the one for controlling the neck motors.
  - **`3rdparty/`:** A directory for third-party libraries that are integrated into the firmware. These libraries provide additional functionality, such as JSON parsing, circular buffers, and logging.

6.2.2. Firmware Core Functions

The firmware for the robot is designed with key principles that ensure its robustness, modularity, and efficiency, directly translating into its core functionalities. This integrated approach ensures that the robot can perform complex tasks reliably and adapt to future enhancements. The following key aspects illustrate how these principles are implemented in the firmware:

1. **Modularity and node-based architecture:** The firmware is built on a node-based architecture where each node represents a specific hardware component or subsystem, such as sensors or actuators. This modular design enables independent development, testing, and integration of various components. Nodes can be individually configured, activated, deactivated, and cleaned up, which enhances the

system's maintainability and allows for scalability as new features or hardware are added. Furthermore, the hierarchical node structure ensures that each node is responsible for initializing and managing its sub-nodes.

2. **State management and separation of concerns:** Each node operates within a structured framework that includes distinct states: unconfigured, inactive, active, and finalized. This separation of states ensures that each component of the robot performs predictably and can be easily debugged and maintained. The firmware includes routines for state transitions, configuration, activation, error handling, and shutdown, ensuring that the system remains reliable under various conditions.

3. **Efficient communication and data handling:** Communication between the NVIDIA Jetson TX2i and the Arduino Mega microcontroller is managed by a custom serial manager. This module processes and serializes data exchanged between the two systems, ensuring low latency and optimizing real-time performance. The use of JSON for structured data and G-Code for command instructions allows the firmware to handle complex communication efficiently. Additionally, the serial manager includes a recursive mechanism that checks for incoming messages across all nodes, ensuring that each node processes relevant JSON and G-Code messages.

4. **Scalability and flexibility:** The firmware is designed to be scalable and flexible, accommodating the addition of new nodes or functionalities with minimal modifications. The modular architecture and use of configuration files make it easy to expand the system's capabilities, which is crucial as the robot evolves to meet new requirements or integrate additional hardware.

5. **Robust error handling:** Each node incorporates error handling routines to manage unexpected conditions. Whether dealing with configuration errors, communication failures, or runtime issues, the firmware is equipped to detect and respond appropriately. This typically involves transitioning the affected node to a safe state and logging the error for further analysis, ensuring the overall system's robustness.

6. **Real-time processing and loop efficiency:** The firmware's main loop is designed to execute all necessary operations, such as processing inputs, managing communications, and executing node-specific tasks, within as fixed time frame. This real-time processing is critical for the robot's responsiveness. The loop is monitored for any delays, and if it exceeds the expected duration, the firmware logs an error to help identify and address performance issues promptly.

By adhering to this principles, the firmware ensures that the robot operates reliably and efficiently, with a structure that supports complex, real-time interactions with its

environment. This robust design framework underpins the robot's ability to perform its tasks effectively, offering a foundation that can adapt to future enhancements or modifications.

6.2.3. Third-Party Libraries and Hardware Integration

The firmware for the robot leverages several third-party libraries to ensure reliable communication, efficient data handling and processing, and monitoring and debugging capabilities.

Three external libraries were used for developing the firmware architecture: ArduinoJson [119], CircularBuffer [120] and EasyLogger [121]. **ArduinoJson** was employed for efficiently serializing and deserializing JSON data, crucial for structured communication between the robot's microcontroller and other components. **CircularBuffer** managed the real-time storage and retrieval of incoming G-Code commands, ensuring smooth, sequential processing without data loss. Finally, **EasyLogger** provided a flexible logging mechanism, enabling detailed tracking and debugging of system events. This enables to work with two firmware versions: release and debug, depending on the desired amount of logging.

In addition, proprietary libraries from manufacturers were utilized to configure hardware components like the Adafruit MPR121, LED rings, LDR sensor, and Dynamixel motor firmware nodes. For other sensors and actuators, such as the buck converter, audio amplifier, and RTC, which lacked dedicated libraries, macros were defined to represent register addresses, enabling direct interaction with the hardware through register read and write operations.

6.3. Software Integration

The integration of the firmware with the software architecture is a critical aspect of the robot's functionality, ensuring that the hardware components are seamlessly controlled and coordinated by the high-level software modules. This integration is achieved through a well-defined architecture that allows for efficient communication, real-time processing, and modular development.

6.3.1. Software Architecture Overview

Figure 6.5 illustrates the interaction between the various modules of the robot's software architecture through publication and subscription to topics using RabbitMQ.

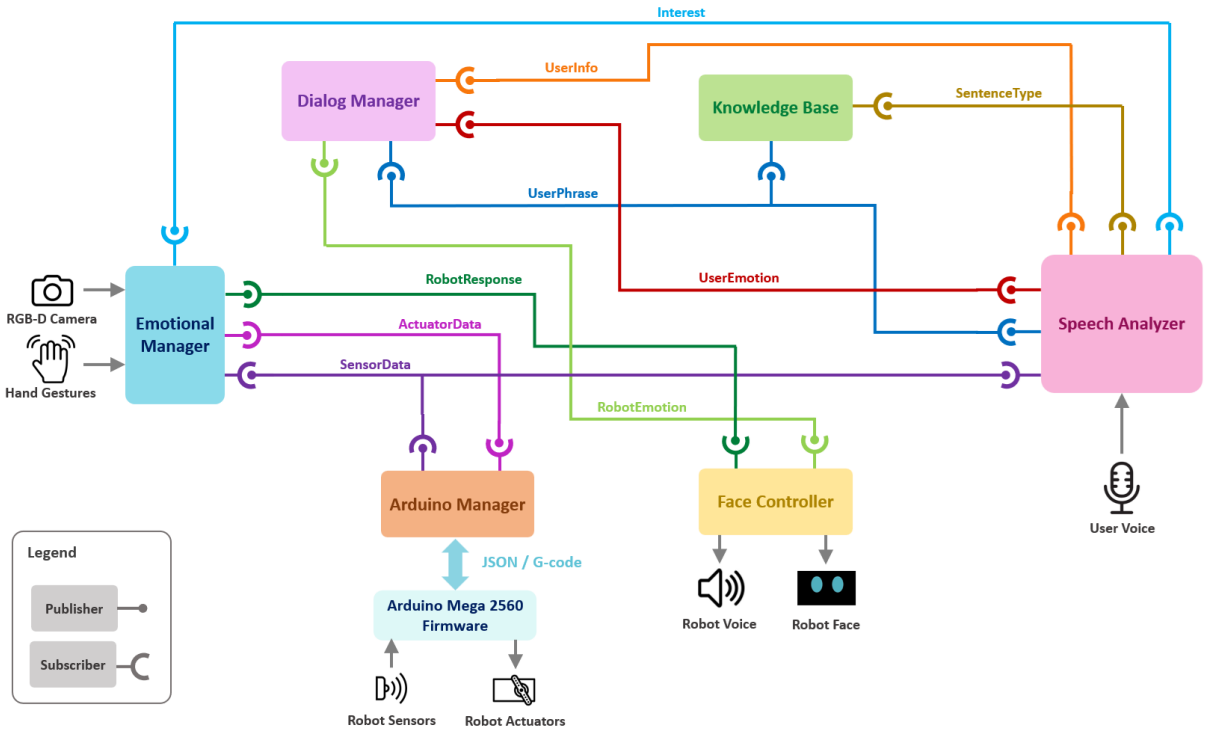


Figure 6.5. Complete software architecture.

The software architecture based on the one of Potato is designed to manage complex interactions and ensure the robot responds appropriately to user input and environmental changes. This architecture is modular, with each component responsible for a specific aspect of the robot's behavior. The key modules are:

1. **Emotional manager:** This module processes inputs from various sensors and interprets the user's emotional state, generating an appropriate emotional response that is sent to other modules for further action.

2. **Dialog manager:** Handles the flow of conversation with the user. It ensures that the robot's verbal responses are coherent, contextually appropriate, and aligned with the detected emotional state.

3. **Arduino manager:** Acts as the bridge between the high-level software modules and the physical components managed by the Arduino microcontroller. It processes sensor data and sends commands to actuators.

4. **Face controller:** Manages the robot's facial expressions, translating emotional data into visual outputs that reflect the robot's current state or response.

5. **Speech analyzer:** Analyzes the user's vocal input to detect emotional cues, speech patterns, and other relevant data, which is then fed back into the system for processing by other modules.

6. **Knowledge base:** Stores user-specific information, historical interactions, and contextual data, enabling the robot to make informed and personalized responses.

The detailed internal functioning description of some of these modules is out of the scope of this work since they were developed for Potato and they have not been implemented yet in the new robotic platform designed in this project.

6.3.2. Simplified Version Implementation

While the complete software architecture comprises six modules, a simplified version, shown in Figure 6.6, has been implemented for the current stage of development. This version includes **three core modules**:

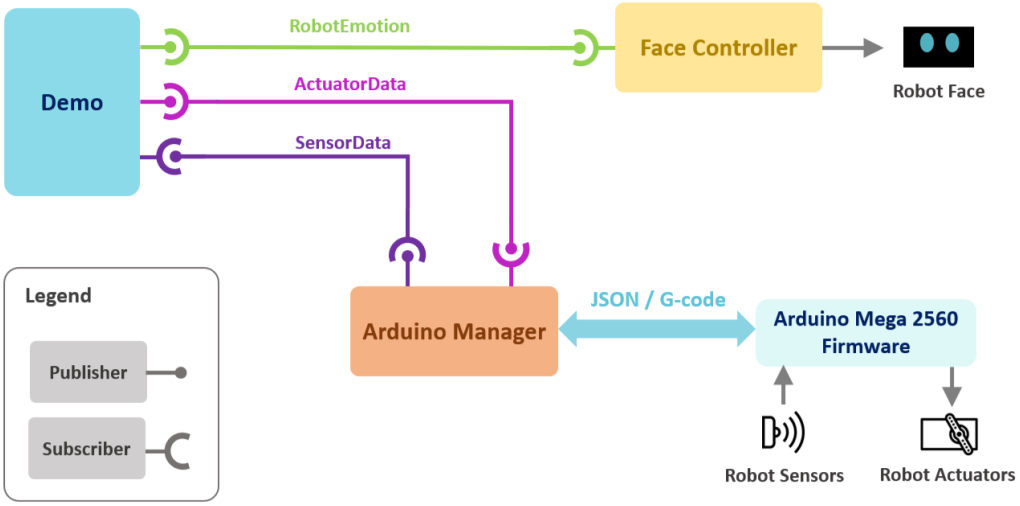


Figure 6.6. Simplified software architecture.

- **Demo module:** This module subscribes to sensor data and publishes emotion data and actuator commands. It is the main application and serves as a demonstration of the robot's capabilities, showing how sensor inputs can influence the robot's emotional state and actions.

- **Face controller module:** This module subscribes to the emotion data published by the Demo Module. It controls the robot's facial expressions, ensuring that they align with the detected emotional state.

- **Arduino manager module:** This module is responsible for interfacing with the Arduino Mega microcontroller. It publishes sensor data and subscribes to actuator commands, acting as a bridge between the robot's hardware and the rest of the software system.

For each topic, specific message types are defined similarly to how it is done in ROS. Sensor data is transmitted using a JSON-structured message, whereas actuator data is sent as G-code string messages. Additionally, JSON can also be used for actuation-related communications. This simplified architecture provides a functional prototype, enabling initial testing and validation of the robot's core capabilities. As development progresses, the additional modules will be integrated, bringing the system closer to its full potential.

Chapter 7

Results and Discussion

This chapter presents a detailed analysis of the outcomes derived from the testing and validation phases of the robot's development. It explores the extent to which the design requirements where met, evaluates the performance of the mechanical and electronic systems, and discusses the results of various tests conducted on sensors, actuators, and firmware. Additionally, the chapter provides an in-depth overview of the demonstration process, detailing its objectives, development, and integration. The insights gained during these stages are discussed, highlighting key findings, challenges encountered, and potential areas for future improvement.

7.1. Assessment of Design Objectives

This section assesses the robot's design against the initial functional and performance requirements, evaluating how well the implemented design meets these goals. Each requirement is reviewed in terms of its fulfillment, limitations encountered, and potential areas of improvement.

- **Compactness and stability:** The robot's design successfully achieved a compact and relatively lightweight structure. The robot is around 40 cm tall when the antennas are disassembled, has a diameter of around 27 cm and weights a bit more than 8 kg, with optimized space for all its hardware components and wiring. Stability proved to be very good thanks to the anti-slip foam pads. Even when the base rotates at high speed the inertia of the robot is not able to unbalance it.
- **Maintenance:** The design allows for easy access to internal components, simplifying maintenance and repairs, particularly for the battery, as it can be seen in Figure

7.1. The modularity of the design ensures that components can be accessed and replaced efficiently, fully meeting this requirement.

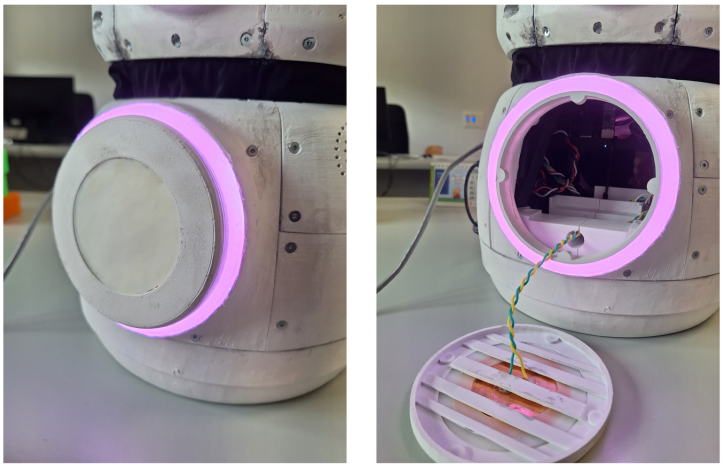


Figure 7.1. Robot's right lateral panel for battery placement.

- **Power management:** The robot can operate effectively both on battery power and when plugged in. However, the battery did not arrive on time for its implementation in the robot so its battery life could not be tested. Rough estimations considering the maximum current measured on the motors and the consumption of the LEDs indicate that it might be able to last 3 hours under normal usage.

- **Air refrigeration:** This requirement could not be tested since the Jetson TX2i was not submitted to heavy computation tasks and several hardware elements were not implemented due to time constraints. It was qualitatively seen that the areas were the LED rings were warmed up but the PETG should be able to resist these temperatures well enough.

- **User-friendly and ergonomic design:** The robot's design was well-received by lab members specialized in social robotics, who provided positive feedback on its approachable appearance, validating the design's effectiveness in a controlled setting. However, further testing is required to evaluate its reception by the target users, particularly in real-world scenarios involving children and elderly individuals, to ensure that the design meets the intended user experience and comfort standards.

- **Cost optimization:** The total material cost for the robot project was approximately 3.500 € as it can be seen at Table C.1 in Appendix C. While efforts were made to optimize costs, the necessity to purchase from specific providers to streamline bureaucracy with the university led to higher expenses for some components. It was observed that significant discounts could have been achieved with alternative suppliers. Moreover, the iterative nature of the design and manufacturing process resulted in a small number of components being discarded. Despite these challenges,

the project remained within its budget, though there is potential for further cost reductions in future iterations by exploring more flexible procurement strategies and avoiding unnecessary purchases.

- **Flexibility and customization:** The design supports future upgrades and customization with the possibility to create and connect additional modules in the head laterals. Since the robot's head is similar in size to a human's, it can also incorporate human accessories such as a cap or hat, further enhancing its customization potential. This flexibility allows the robot to adapt to different environments and user preferences, making it more versatile and relatable in various social robotics applications.

- **Emotion display and interaction:** A key objective was for the robot to display basic emotions effectively. This was accomplished through facial expressions, body movements, and LED indicators. The integration of these features allowed the robot to engage users emotionally, enhancing the overall interaction experience. However, the emotional display system could be expanded in future versions to include a broader range of emotions and more nuanced expressions.

- **Ease of assembly and component integration:** The robot was designed for easy assembly and disassembly, which was crucial for maintenance and potential upgrades. The component integration was carefully planned to ensure that all compulsory and optional components were seamlessly incorporated into the design . This requirement was fully met, with the assembly process being straightforward and well-documented. However, some pieces could have benefited from a more optimized design, as certain steps in the assembly process could have been more comfortable.

- **Minimum functionalities:** All required functionalities, such as tactile interaction and audio communication, were successfully implemented and tested, confirming that the robot meets the essential functional criteria.

Overall, the design requirements were largely met, with only a few areas identified for potential improvement. These findings will inform future development efforts, ensuring that the robot continues to evolve to meet user needs and technical challenges.

7.2. System Testing and Validation

This section presents the testing and validation process for the robot, focusing on mechanical performance, sensor and actuator functionality, and firmware efficiency. Given

that the robot is in the very early stages of development, these tests are primarily **qualitative** in nature. The challenge in finding quantitative metrics for some aspects of the robot's performance also contributed to this qualitative focus. Nonetheless, the tests conducted provide valuable insights into the robot's current capabilities and areas for improvement.

7.2.1. Mechanical Testing

During the mechanical testing phase, the first prototype of the neck mechanism was found to be inadequate, as it lacked the necessary strength to lift the robot's head. This initial failure highlighted the need for a redesign. The new mechanism, which incorporated a more robust structure and improved gearing, was subsequently tested and demonstrated the ability to perform full-range movements without difficulty. However, it was noted that the redesigned mechanism exhibited a slight amount of backlash. This could be attributed to minor play within the spherical gear mechanism. For future iterations, this issue could be mitigated by increasing the top cover of the spherical gear, shown in Figure 7.2, which would help to reduce the backlash and enhance the overall precision of the movements. Note that the top cover had this partially-covering design for enabling easy assembly. In case this modification is done, it would be required to divide the top cover into two parts.



Figure 7.2. Detail of the top cover of the spherical gear mechanism.

Additionally, during these tests, an ammeter was connected to measure the current consumed by the motors while lifting the head and performing full-range movements. The results showed that the current peaks **did not exceed 0.6 A** for two motors operating simultaneously. Given that the robot has a total of six motors—four larger ones and two smaller ones—the maximum estimated current demand would be less than 5 A at peak, which aligns well with the battery's maximum current supply capability. This results seemed to indicate that there is enough margin for the power consumption of the NVIDIA Jetson TX2i, known to have significant current demands, especially under

high computational loads. The current drawn by other components such as sensors and actuators, is relatively negligible compared to the motors and the Jetson TX2i.

7.2.2. Sensor and Actuator Testing

The testing of the robot's sensors and actuators was conducted in a structured manner, beginning with unitary tests of individual components and progressively integrating additional hardware into the firmware. This approach allowed for the isolation and identification of potential issues, ensuring that each element functioned correctly before advancing to more complex integrations. Due to the **limited timeframe, exhaustive testing**, validation, and full integration of all hardware components **were not possible**, but the primary components were evaluated as follows.

7.2.2.1. LDR

The LDR was tested to assess its ability to detect changes in ambient light level. The LDR was integrated in the firmware using the third-party library provided by the manufacturer, and its responsiveness was evaluated by exposing it to varying light conditions. The tests demonstrated that the LDR could reliably detect changes in light intensity, providing accurate data for potential use in environmental awareness features.

However, it was observed that the LDR took approximately **4 seconds to update** light intensity values, even with the shortest firmware loop rate possible of around 200 ms. This delay suggests that the issue likely stems from the LDR hardware or its associated library, rather than the firmware. Future research could explore the possibilities offered by the current library to determine if improvements in responsiveness can be achieved or if the hardware inherently requires such a long acquisition time.

7.2.2.2. Touch Pads

The touch pads were tested to ensure they could detect user interactions with the robot. Each pad was connected individually to the Adafruit MPR121, which was connected to the Arduino Mega through I2C, and a series of touch gestures were performed to evaluate sensitivity and accuracy. The touch pads showed consistent performance, successfully detecting light touches and more deliberate presses. However, it was noted that very light caresses were not consistently detected, indicating that the sensitivity might not be fully optimized. This suggests that further calibration of the Adafruit MPR121 might be necessary to improve the responsiveness of the touch pads to lighter touches.

7.2.2.3. LEDs

The LEDs were tested to validate their ability to display a range of colors and intensities, which are used for emotional expressions and signaling. Each LED module was activated via the firmware, and a sequence of color changes was commanded. The LEDs performed as expected, producing bright and distinct colors, as it can be seen in Figure 7.3. In addition to this, custom G-codes were programmed in the firmware to ease the most common operations with the LEDs, turn them all on, all off, and set them to the color associated to a certain emotion. The heartbeat LED was configured to run individually in a pulsing mode with a frequency determined by its emotional state. The tests confirmed that the LED system is capable of conveying emotional states effectively.



Figure 7.3. LED rings testing.

7.2.2.4. LCD Display

The LCD display was tested to ensure it could accurately present the visual information, including facial expressions and user interface elements. It was also checked the elastic mesh added for aesthetic purposes did not impede proper visualization, as it can be seen in Figure 7.4. The display was connected through HDMI to the Jetson TX2i and functioned correctly, with sharp image quality and responsive display changes. Additionally, the Jetson was configured to not display the cursor and the window top bar from the Python script-generated images, ensuring that only the face itself was shown on the screen, providing a cleaner and more immersive visual experience.

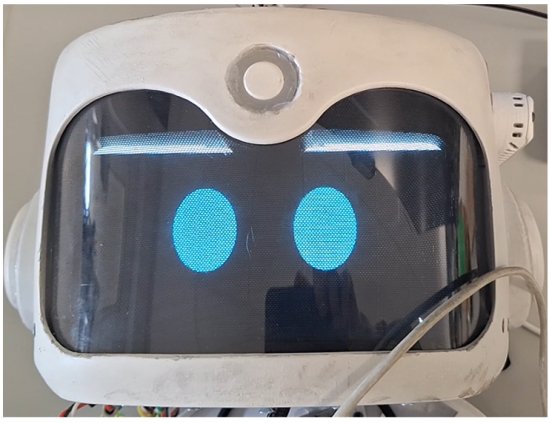


Figure 7.4. LCD display testing.

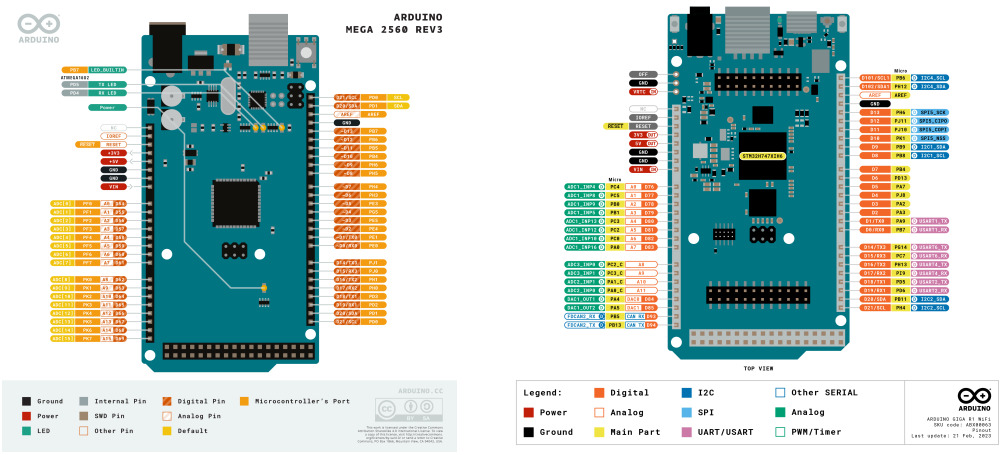
7.2.3. Firmware Performance Testing

The firmware performance testing revealed a significant limitation with the Arduino Mega 2560 microcontroller due to its constrained **dynamic memory** capacity of just **8 kB**. This limited memory means that the Arduino Mega can only run a few nodes simultaneously. Specifically, it was tested to be able to run properly 13 nodes corresponding to the base motor, heartbeat LED, body and head LED rings, LDR sensor and front head tactile pad with their respective parent nodes. This restriction significantly impacts the robot's ability to perform complex operations with all its hardware components engaged simultaneously, leading to a need for careful management of the active nodes to avid exceeding the memory capacity.

To mitigate this issue, the firmware was tested by selectively deactivating certain nodes, allowing to test some nodes in isolation. This testing approach provided useful insights into the performance of the firmware under various configurations, confirming that while the Arduino Mega can handle basic operations, it struggles to manage the full array of nodes without encountering memory-related performance issues.

Despite these limitations, the firmware's communication using JSON and G-code messages wasa tested and shown to work correctly. This successful communication is crucial as it underpins the interaction between the robot's subsystems, ensuring that the commands and data are accurately transmitted and executed.

Looking ahead, the **Arduino Giga R1 WiFi** [122] presents a promising upgrade path. With a substantially larger dynamic memory capacity of 1 MB, **125 times more** than the Arduino Mega, this microcontroller can support a far greater number of simultaneously active nodes. This enhancement would enable the robot to fully utilize all its hardware components without the need for deactivating certain nodes to manage memory limitations. The Arduino Giga R1 WiFi also shares the same form factor as the Arduino



(a) **(b)**

Figure 7.5. Arduino layout comparison: (a) Arduino Mega 2560 [110], (b) Arduino Giga R1 WiFi [122].

Mega, and has very similar connection ports, as shown in Figures 7.5a and 7.5b, meaning that only minimal modifications would be required to the existing protoshield connections to accommodate this upgrade.

Incorporating the Arduino Giga would not only resolve the current memory limitations but also significantly enhance the robot's overall performance, responsiveness, and ability to execute more complex tasks simultaneously. This upgrade would be a key step forward in realizing the full potential of the robot design and capabilities.

7.3. Demo Description and Integration

Given the memory limitations of the Arduino Mega, the demo development had to be meticulously planned to optimize the use of available resources. The initial comprehensive plan was revised to focus on critical components, including the base motor for primary movement, the LCD display for facial expressions, the LED rings for emotional cues, the heartbeat LED, the front head touch pad for user interaction, and the LDR for ambient light detection. Figure 7.6 shows the interactive flow of the proposed demo.

The demo was conceptualized as a state machine due to the absence of higher-level modules such as the emotional manager, dialog manager, knowledge base, and speech analyzer. This approach allowed for deterministic behavior based on specific inputs, ensuring predictable and manageable interactions within the constrained environment. The interaction flow is structured as follows:

1. **Initial state (normal):** The robot starts in a neutral emotional state. The LCD display shows a neutral face, and the LED rings are set to a calm blue color.

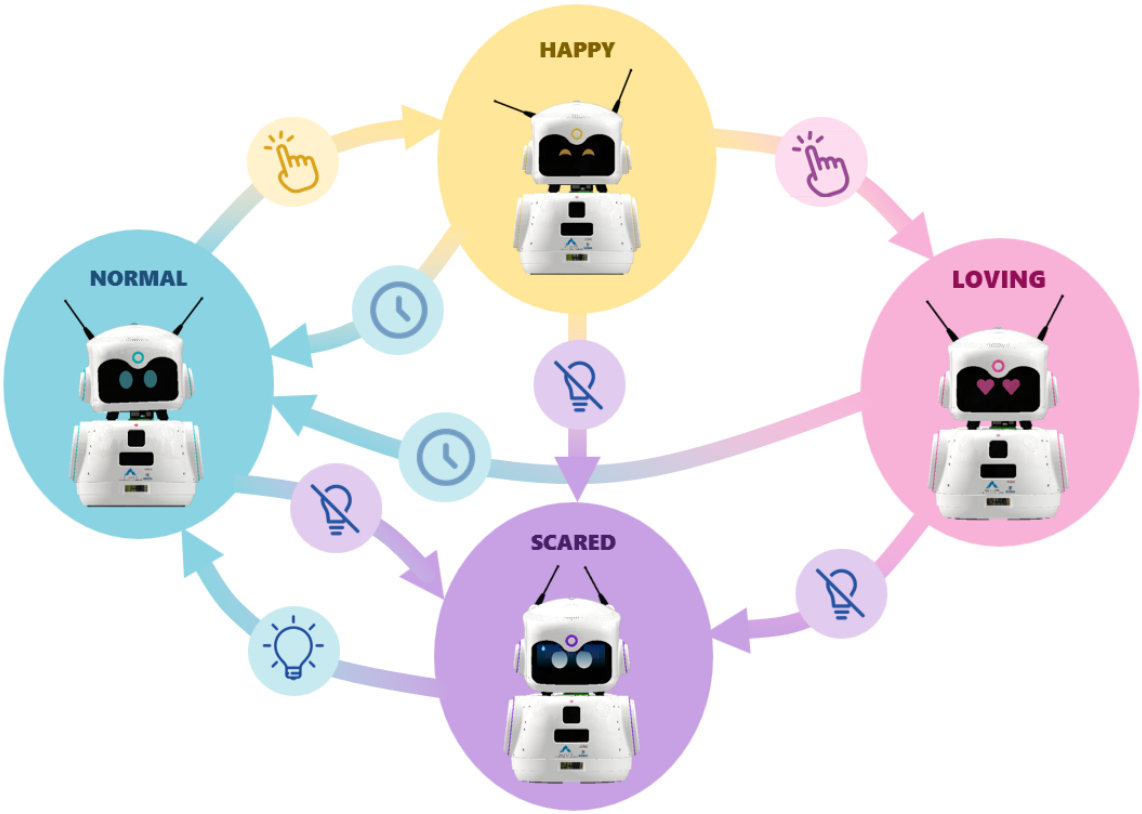


Figure 7.6. Demo state machine diagram.

2. **Touch detected:** When the touch pad is activated, the robot transitions to the happy state. The LCD displays a happy face, the LED rings turn yellow, and the heartbeat LED pulses more rapidly. This happy state will remain during a certain time and then go back to normal again. If the robot is touched again when it is happy, then it will turn into loving state, with pink color LED rings. In the loving state, the base motor is activated to demonstrate a simple movement, symbolizing the robot's excitement or eagerness.

3. **Light level change:** The LDR continuously monitors ambient light levels. If a significant drop in light level is detected, the robot transitions to the scared state. The LCD displays a scared face and the LED rings turn purple, with the heartbeat LED pulsing rapidly. After the light level returns to its initial value, the robot will return to its normal state.

This simplified demo highlights the core interactions possible within the current hardware limitations, effectively showcasing the robot's ability to change states based on touch and light inputs. Additionally, the firmware's communication through JSON and G-code messages was confirmed to be working properly during the demo. Figure 7.7 shows the robot in various states, displaying different emotions as part of the demo, illustrating its ability to express emotions through visual outputs.

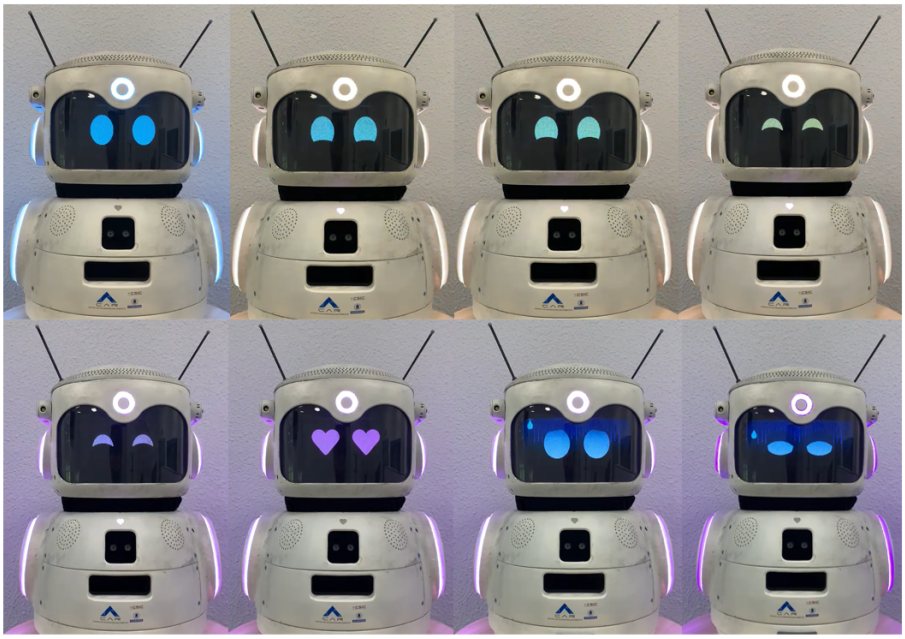


Figure 7.7. Robot's demo animated states.

While the demo was more limited than initially planned, it successfully demonstrated the integration of various sensors and actuators, providing a solid foundation for future enhancements as more powerful hardware components, such as the Arduino Giga, are incorporated.

7.4. Summary of Key Findings

The results from the testing and validation phases revealed several critical insights into the robot's design, performance, and functionality. Key findings from this chapter include:

- **Design requirements fulfillment:** The robot's design largely met the initial requirements, achieving a compact and stable structure, ease of maintenance, and effective power management. However, certain areas, such as air refrigeration and user reception, require further testing and validation.
- **Mechanical and structural performance:** The mechanical testing identified initial weaknesses in the neck mechanism, which were successfully addressed in the redesign. The improved mechanism demonstrated reliable performance with minimal backlash. The current measurements confirmed that the robot's power system is adequately designed to meet the demands of the motors and the Jetson TX2i.
- **Sensor and actuator functionality:** The sensors and actuators performed as expected during unitary and progressive tests, with the LED rings, LCD display,

LDR, and touch pads all functioning effectively. However, some limitations were noted, such as the LDR's slow responsiveness and the need for further calibration of the touch pads to detect light caresses.

- **Firmware performance:** The Arduino Mega's limited dynamic memory was identified as a significant constraint, allowing only a subset of nodes to run simultaneously. Despite this limitation, the firmware successfully managed communication through JSON and G-code messages. The potential upgrade to the Arduino Giga R1 WiFi was highlighted as a solution to overcome these limitations.
- **Demo development:** The demo was adapted to showcase the robot's core functionalities within the constraints of the available hardware and firmware. By focusing on key interactions, such as touch detection, light sensitivity, and emotional display; the demo effectively demonstrated the integrated performance of the robot's primary components.

These findings provide a clear direction for future development, emphasizing the need for hardware upgrades, further testing, and expanded functionality to realize the full potential of the robot's design.

Chapter 8

Conclusions and Future Work

This final chapter summarizes the key findings and accomplishments of the project. It reflects on the project's outcomes and the overall success in meeting the initial objectives. Furthermore, this chapter outlines potential future work and areas for further research and development, aimed at enhancing the robot's capabilities, improving its design, and expanding its application scope.

8.1. Main Conclusions

The development of the social robot presented in this work has been a significant achievement, providing a solid foundation for future research and development in personal assistance robotics. The project **successfully met its primary objectives** by delivering a sophisticated hardware platform that integrates mechanical design, electronic components, and firmware development, all geared towards enhancing human-robot interaction.

A key goal was to create a robot with a modular and robust design capable of **advanced interaction capabilities**. This was achieved through a well-thought-out mechanical design, resulting in a compact, stable structure that not only facilitates easy access to internal components but also ensures the robot is **user-friendly and maintainable**. The integration of essential hardware components equipped the robot with the tools necessary to effectively interact with its environment. The design also prioritized emotional engagement, with the successful development and integration of animated **emotional expressions** into the robot's interaction system, demonstrating its potential to connect with users on a deeper level.

The firmware, although constrained by the **limitation of the current microcontroller**, laid a solid foundation for controlling the robot's basic functions. The hierarchical node firmware architecture introduced in this project is a promising start that, with more powerful hardware, can fully exploit the robot's capabilities. The **modular approach** in both hardware and software ensures that the robot can be continually enhanced, allowing for future upgrades and the addition of new functionalities.

The integration and testing phase proved that the robot could function as a cohesive system, validating the design and implementation decisions made throughout the project. Despite some challenges, particularly in fully deploying the firmware and integrating all planned components, the project has met its key objectives. The robot's design is both **functional and aesthetically appealing**, setting the stage for future developments.

In conclusion, this project has successfully created a **sophisticated hardware platform for a social robot**, laying the groundwork for future advancements in personal assistance robotics. While there are areas for improvement, the results achieved represent a significant contribution, offering numerous opportunities to continued research and development. The project's success in meeting its objectives underscores its potential to advance the capabilities of social robots in **real-world applications**.

8.2. Future Work Lines

The project opens several avenues for future research and development. These include enhancing the robot's interaction capabilities, improving its mechanical design, and expanding its application areas. Future work can be grouped into several key categories: design and hardware enhancements, firmware and software improvements, and simulation. Each area offers potential for significant advancements that can enhance the robot's capabilities, usability, and performance.

8.2.1. Design and Hardware Enhancements

- **Neck Mechanism Improvement:** While the developed spherical gear neck mechanism meets the required torque and position accuracy, there is a **slight play** that could be reduced by better encapsulating the spherical gear with the top cover.

- **Battery Integration:** Integration of the battery is the next step once procurement is complete. This will include making necessary connections and incorporating the battery management system's data acquisition into the firmware for accurate battery charge state estimation.

- **Antennas Integration:** Due to limitations in firmware's node capacity, the antennas' motors have not been tested on the robot. The next step involves configuring a **buck converter** to supply the servomotors with 8 V to ensure proper operation.
- **Face Animations Improvement:** Currently, only thirteen animations corresponding to four emotions have been developed. Future work could involve creating a **spritesheet system for facial expressions** based on Plutchik's wheel, allowing for emotional states to be directly translated into dynamic facial animations, improving animated transitions scalability.
- **Thermal Analysis:** Empirical testing of the robot at various workloads, particularly focusing on the NVIDIA Jetson TX2i, could **thermally validate the robot design**. Monitoring temperatures at critical points using the built-in sensors in the servomotors could identify further design optimization opportunities.
- **Additional Modules Design:** New modules could be designed to expand the robot's capabilities, adding versatility and functionality.
- **Higher Quality Manufacturing:** The design presented in this work could benefit from higher-quality 3D printing and the use of more professional materials. **Technical drawings** may be needed for components that will be externally manufactured.
- **Assembly Manual:** A detailed user-oriented assembly manual could be beneficial, especially if more similar robots are to be manufactured.
- **Transport Solution:** Given the robot's intended use at various events, a custom, **internally-foamed suitcase** would protect the robot during transport.

8.2.2. Firmware and Software Enhancements

- **Arduino Microcontroller Enhancement:** Upgrading from the Arduino Mega 2560 to the **Arduino Giga R1 WiFi** [122] could overcome current dynamic memory limitations, potentially making the robot capable of running all the firmware nodes and more responsive.
- **Firmware Development for Additional Components:** Integrating firmware for components like the IMU, NFC sensor, gas detection sensors, and others is a priority. This will ensure full functionality and improve the system current capabilities.
- **Arduino Firmware Programming via NVIDIA Jetson TX2i:** Future improvements could include programming the Arduino directly from the NVIDIA Jetson TX2i via the robot's USB-C back panel port, eliminating the need for manual

UART connection to the Arduino. This can be implemented using the USB to TTL serial UART converter PCB already acquired, streamlining firmware updates.

- **Neck Mechanism Trajectories Programming:** Firmware could be enhanced to program the motors for optimal neck movement trajectories and **emotional state-dependent body positioning**, facilitating advanced applications development.
- **Application Development:** The hardware platform created in this work could be applied to various applications, such as education or personal assistance. This would require higher-level software development and seamless integration of the different software architecture modules.
- **Automatic Module Connection Detection:** The firmware could be programmed to automatically detect when a module is detected to interact in some way (such as turning on the LED ring of the ear beneath the module).
- **PCB Design and Manufacturing:** Once connections are finalized, designing a custom PCB could provide a more robust and scalable solution than the current manual implementation.
- **LDR Firmware Optimization:** The current LDR sensor firmware lacks responsiveness. Further investigation and potential sensor replacement could improve performance.
- **Firmware Improvements:** Optimization of the firmware to communicate only the changed variables could reduce bandwidth usage and improve efficiency.
- **Software Testing:** Formal unit, integration, and system tests are necessary to ensure that the software functions correctly as its complexity increases.
- **Software Performance Testing:** Monitoring software performance during operation can identify resource usage and optimization opportunities.
- **Firmware and Software Manual:** A developer-oriented manual would help new developers (students and researchers) understand the robot's architecture and follow best practices in code development.

8.2.3. Development Simulation Interface

- **URDF Creation:** Creating a URDF file is the first step in developing a digital twin of the robot, facilitating simulation and further development.

- **Gazebo[1] / Unity[2] Development Simulation Interface:** Developing a Gazebo or Unity simulation would enable the development and testing of applications without the need for physical hardware, which is particularly useful for collaborative projects such as this one where multiple students and researchers might be progressing on different work lines.

[1] It is a powerful robot simulation tool that allows for testing and development of robotics algorithms in realistic environments.
[2] It is a versatile and powerful real-time 3D development platform used for creating and operating interactive, real-time content such as games, simulations, and visualizations.

Bibliography

[1] J. J. Rainer Granados and R. Galán López, "URBANO: A Tour-Guide Robot Learning to Make Better Speeches", in *Proceedings of the Cognitive 2010. The 2nd International Conference on Advanced Cognitive Technologies and Applications*, IARIA, 2010.

[2] B. P. E. A. Vásquez and F. Matía, "A tour-guide robot: Moving towards interaction with humans", *Engineering Applications of Artificial Intelligence*, vol. 88, p. 103 356, 2020, ISSN: 0952-1976. DOI: 10.1016/j.engappai.2019.103356.

[3] G. Fernández-Blanco Martín, F. Matía, L. García Gómez-Escalonilla, D. Galan, M. G. Sánchez-Escribano, P. de la Puente, and M. Rodríguez-Cantelar, "An Emotional Model Based on Fuzzy Logic and Social Psychology for a Personal Assistant Robot", *Applied Sciences*, vol. 13, no. 5, 2023, ISSN: 2076-3417. DOI: 10.3390/app13053284.

[4] C. Lutz and A. Tamó-Larrieux, "The robot privacy paradox: Understanding how privacy concerns shape intentions to use social robots", *Human-Machine Communication*, vol. 1, pp. 87–111, 2020.

[5] C. Breazeal, "Emotion and sociable humanoid robots", *International Journal of Human-Computer Studies*, vol. 59, no. 1, pp. 119–155, 2003, Applications of Affective Computing in Human-Computer Interaction, ISSN: 1071-5819. DOI: 10.1016/S1071-5819(03)00018-1.

[6] M. M. A. de Graaf, S. B. Allouch, and J. A. G. M. van Dijk, "Long-term evaluation of a social robot in real homes", *Interaction Studies*, vol. 17, no. 3, pp. 462–491, 2016, ISSN: 1572-0373. DOI: 10.1075/is.17.3.08deg.

[7] M. S. H. Sunny, M. M. Rahman, M. E. Haque, N. Banik, H. U. Ahmed, and M. H. Rahman, "Chapter 1 - Assistive robotic technologies: An overview of recent advances in medical applications", in *Medical and Healthcare Robotics*, ser. Medical Robots and Devices: New Developments and Advances, O. Boubaker, Ed., Academic Press, 2023, pp. 1–23. DOI: 10.1016/B978-0-443-18460-4.00004-4.

[8] S. Naneva, M. Sarda Gou, T. L. Webb, and T. J. Prescott, "A Systematic Review of Attitudes, Anxiety, Acceptance, and Trust Towards Social Robots", *International Journal of Social Robotics*, vol. 12, no. 6, pp. 1179–1201, Dec. 2020, ISSN: 1875-4805. DOI: 10.1007/s12369-020-00659-4.

[9] D. David, P. Thérouanne, and I. Milhabet, "The acceptability of social robots: A scoping review of the recent literature", *Computers in Human Behavior*, vol. 137, p. 107 419, 2022, ISSN: 0747-5632. DOI: 10.1016/j.chb.2022.107419.

[10] M. Sarrica, S. Brondi, and L. Fortunati, "How many facets does a "social robot" have? A review of scientific and popular definitions online", *Inf. Technol. People*, vol. 33, pp. 1–21, 2019.

[11] T. Fong, I. Nourbakhsh, and K. Dautenhahn, "A survey of socially interactive robots", *Robotics and Autonomous Systems*, vol. 42, no. 3, pp. 143–166, 2003, Socially Interactive Robots, ISSN: 0921-8890. DOI: 10.1016/S0921-8890(02)00372-X.

[12] C. Bartneck and J. Forlizzi, "A design-centred framework for social human-robot interaction", in *RO-MAN 2004. 13th IEEE International Workshop on Robot and Human Interactive Communication (IEEE Catalog No.04TH8759)*, 2004, pp. 591–594. DOI: 10.1109/ROMAN.2004.1374827.

[13] F. Eyssel, D. Kuchenbrandt, S. Bobinger, L. de Ruiter, and F. Hegel, ""If you sound like me, you must be more human": on the interplay of robot and user features on human-robot acceptance and anthropomorphism", in *Proceedings of the Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '12, Boston, Massachusetts, USA: Association for Computing Machinery, 2012, pp. 125–126, ISBN: 9781450310635. DOI: 10.1145/2157689.2157717.

[14] W. G. Walter, "An Imitation of Life", *Scientific American*, vol. 182, pp. 42–45, 1950. DOI: 10.1038/SCIENTIFICAMERICAN0550-42.

[15] J. Deneubourg, "The Dynamics of Collective Sorting Robot-like Ant and Ant-like Robots", *From Animats to Animats*, vol. 2, pp. 353–363, 1992.

[16] R. Brooks, C. Breazeal, M. Marjanovic, B. Scassellati, and M. Williamson, "The Cog Project: Building a Humanoid Robot", *Lecture Notes in Artificial Intelligence*, vol. 1562, Mar. 2002. DOI: 10.1007/3-540-48834-0_5.

[17] M. Fujita, "AIBO: Toward the era of digital creatures", *I. J. Robotic Res.*, vol. 20, pp. 781–794, Oct. 2001. DOI: 10.1177/02783640122068092.

[18] N. L. A. Rashid, Y. Leow, P. Klainin-Yobas, S. Itoh, and V. X. Wu, "The effectiveness of a therapeutic robot, 'Paro', on behavioural and psychological symptoms, medication use, total sleep time and sociability in older adults with dementia: A systematic review and meta-analysis", *International Journal of Nursing Studies*, vol. 145, p. 104 530, 2023, ISSN: 0020-7489. DOI: 10.1016/j.ijnurstu.2023.104530.

[19] A. K. Pandey and R. Gelin, "A Mass-Produced Sociable Humanoid Robot: Pepper: The First Machine of Its Kind", *IEEE Robotics & Automation Magazine*, vol. 25, pp. 40–48, Jul. 2018. DOI: 10.1109/MRA.2018.2833157.

[20] D. López Recio, L. Márquez Segura, E. Márquez Segura, and A. Waern, "The NAO models for the elderly", in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2013, pp. 187–188. DOI: 10.1109/HRI.2013.6483564.

[21] A. A. Ali, J.-J. Cabibihan, N. Meskin, S. Rossi, W. Jiang, H. He, and S. S. Ge, Eds., *Social Robotics: 15th International Conference, ICSR 2023, Doha, Qatar, December 3–7, 2023, Proceedings, Part II* (Lecture Notes in Computer Science), en. Singapore: Springer Nature, 2024, vol. 14454. DOI: 10.1007/978-981-99-8718-4.

[22] O. Korn, *Social Robots: Technological, Societal and Ethical Aspects of Human-Robot Interaction*. Jul. 2019, ISBN: 978-3-030-17106-3.

[23] G. Shaw-Garlock, "Looking Forward to Sociable Robots", *International Journal of Social Robotics*, vol. 1, no. 3, pp. 249–260, Aug. 2009, ISSN: 1875-4805. DOI: 10.1007/s12369-009-0021-7.

[24] C. Bartneck, D. Kulic, E. Croft, and S. Zoghbi, "Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots", *International Journal of Social Robotics*, vol. 1, pp. 71–81, Jan. 2008. DOI: 10.1007/s12369-008-0001-3.

[25] M. Heerink, B. Krose, V. Evers, and B. Wielinga, "Assessing acceptance of assistive social agent technology by older adults: The almere model", *I. J. Social Robotics*, vol. 2, pp. 361–375, Dec. 2010. DOI: 10.1007/s12369-010-0068-5.

[26] C. Breazeal, *Designing Sociable Robots*. The MIT Press, May 2002. DOI: 10.7551/mitpress/2376.001.0001.

[27] R. W. Picard, *Affective Computing*. The MIT Press, Sep. 1997. DOI: 10.7551/mitpress/1140.001.0001.

[28] M. Mori, K. F. MacDorman, and N. Kageki, "The Uncanny Valley", *IEEE Robotics & Automation Magazine*, vol. 19, no. 2, pp. 98–100, 2012. DOI: 10.1109/MRA.2012.2192811.

[29] C. Becker-Asano, K. Ogawa, S. Nishio, and H. Ishiguro, "Exploring the uncanny valley with Geminoid HI-1 in a real-world application", Jan. 2010, pp. 121–128.

[30] D. Shamonsky, "User Experience Best Practices for Human-Robot Interaction", in *Human-Computer Interaction. Interaction Techniques and Novel Applications*, M. Kurosu, Ed., Cham: Springer International Publishing, 2021, pp. 318–328.

[31] S. O. Oruma, Y. Z. Ayele, F. Sechi, and H. Rødsethol, "Security Aspects of Social Robots in Public Spaces: A Systematic Mapping Study", *Sensors*, vol. 23, no. 19, 2023. DOI: 10.3390/s23198056.

[32] J. Broekens, E. Hudlicka, and R. Bidarra, "Emotional Appraisal Engines for Games", in *Emotion in Games: Theory and Praxis*, K. Karpouzis and G. N. Yannakakis, Eds. Cham: Springer International Publishing, 2016, pp. 215–232. DOI: 10.1007/978-3-319-41316-7_13.

[33] F. Cavallo, F. Semeraro, L. Fiorini, and others, "Emotion Modelling for Social Robotics Applications: A Review", *Journal of Bionic Engineering*, vol. 15, pp. 185–203, Mar. 2018. DOI: 10.1007/s42235-018-0015-y.

[34] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion", *Journal of Personality and Social Psychology*, vol. 17, no. 2, pp. 124–129, Feb. 1971. DOI: 10.1037/h0030377.

[35] J. Russell, "A Circumplex Model of Affect", *Journal of Personality and Social Psychology*, vol. 39, pp. 1161–1178, Dec. 1980. DOI: 10.1037/h0077714.

[36] R. Plutchik, "A general psychoevolutionary theory of emotion", in *Theories of Emotion*, Amsterdam, The Netherlands: Elsevier, 1980, pp. 3–33.

[37] R. Plutchik, "The Nature of Emotions", *American Scientist*, vol. 89, pp. 344–350, 2001. DOI: 10.1511/2001.28.344.

[38] P. Ekman and W. V. Friesen, "Facial action coding system", *Environmental Psychology & Nonverbal Behavior*, 1978.

[39] A. Kalegina, G. Schroeder, A. Allchin, K. Berlin, and M. Cakmak, "Characterizing the Design Space of Rendered Robot Faces", in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '18, Chicago, IL, USA: Association for Computing Machinery, 2018, pp. 96–104, ISBN: 9781450349536. DOI: 10.1145/3171221.3171286.

[40] P.-C. Lin, P. C. K. Hung, Y. Jiang, C. P. Velasco, and M. A. Martínez Cano, "An experimental design for facial and color emotion expression of a social robot", *The Journal of Supercomputing*, vol. 79, no. 2, pp. 1980–2009, Feb. 2023, ISSN: 1573-0484. DOI: 10.1007/s11227-022-04734-7. [Online]. Available: https://doi.org/10.1007/s11227-022-04734-7.

[41] R. M. Krauss, P. Morrel-Samuels, and C. Colasante, "Do conversational hand gestures communicate?", *Journal of personality and social psychology*, vol. 61, no. 5, p. 743, 1991.

[42] D. McNeill, *Hand and Mind: What Gestures Reveal about Thought*, 1992.

[43] N. H. Frijda, "Recognition of emotion", in *Advances in experimental social psychology*, vol. 4, Elsevier, 1969, pp. 167–223.

[44] R. Grandia, E. Knoop, M. A. Hopkins, G. Wiedebach, J. Bishop, S. Pickles, D. Müller, and M. Bächer, "Design and Control of a Bipedal Robotic Character", in *Robotics Science and Systems*, Disney Research, Switzerland and USA; Walt Disney Imagineering R&D, USA, Delft University of Technology, Netherlands, Jul. 2024.

[45] A. Niculescu, B. Dijk, A. Nijholt, H. Li, and S. See, "Making Social Robots More Attractive: The Effects of Voice Pitch, Humor and Empathy", *International Journal of Social Robotics*, vol. 5, pp. 171–191, Apr. 2013. DOI: 10.1007/s12369-012-0171-x.

[46] B. Zhang and N. Fitter, "Nonverbal Sound in Human-Robot Interaction: A Systematic Review", *ACM Transactions on Human-Robot Interaction*, vol. 12, pp. 1–46, Feb. 2023. DOI: 10.1145/3583743.

[47] S. Mick, M. Lapeyre, P. Rouanet, C. Halgand, J. Benois-Pineau, F. Paclet, D. Cattaert, P.-Y. Oudeyer, and A. de Rugy, "Reachy, a 3D-Printed Human-Like Robotic Arm as a Testbed for Human-Robot Control Strategies", *Frontiers in Neurorobotics*, vol. 13, 2019, ISSN: 1662-5218. DOI: 10.3389/fnbot.2019.00065.

[48] MatWeb, LLC, *MatWeb Material Property Data*, 2024. [Online]. Available: http://www.matweb.com (visited on 02/08/2024).

[49] Simplify3D, *Ultimate 3D Printing Material Properties Table*, 2024. [Online]. Available: https://www.simplify3d.com/resources/materials-guide/properties-table/ (visited on 02/08/2024).

[50] T. Yonezawa, T. Ogata, and Y. Okabe, "Soft robotics in hri: Improving interaction through softness and compliance", *Autonomous Robots*, vol. 45, pp. 563–576, 2021. DOI: 10.1007/s10514-020-09929-2.

[51] P. D. Rane, V. Mhatre, and L. Kurup, "Study of a Home Robot: JIBO", *International Journal of Engineering Research and Technology (IJERT)*, vol. 3, 2014.

[52] *ASUS Zenbo - Your Smart Little Companion*. [Online]. Available: https://zenbo.asus.com/ (visited on 02/08/2024).

[53] G. Milliez, "Buddy: A Companion Robot for the Whole Family", Mar. 2018, pp. 40–40. DOI: 10.1145/3173386.3177839.

[54] S. So and N. Lee, "Pedagogical exploration and technological development of a humanoid robotic system for teaching to and learning in young children", *Cogent Education*, vol. 10, no. 1, p. 2 179 181, 2023. DOI: 10.1080/2331186X.2023.2179181.

[55] J. K. Westlund, J. J. Lee, L. Plummer, F. Faridi, J. Gray, M. Berlin, H. Quintus-Bosz, R. Hartmann, M. Hess, S. Dyer, K. dos Santos, S. Ö. Aðalgeirsson, G. Gordon, S. Spaulding, M. Martinez, M. Das, M. Archie, S. Jeong, and C. Breazeal, "Tega: A social robot", in *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2016, pp. 561–561. DOI: 10.1109/HRI.2016.7451856.

[56] *Moxie Robot: The World's First AI Robot for Kids Aged 5-10*. [Online]. Available: https://moxierobot.com/ (visited on 02/08/2024).

[57] N. Hurst, C. Clabaugh, R. Baynes, J. F. Cohn, D. Mitroff, and S. Scherer, "Social and Emotional Skills Training with Embodied Moxie", *CoRR*, vol. abs/2004.12962, 2020.

[58] B. Sawik, S. Tobis, E. Baum, A. Suwalska, S. Kropińska, K. Stachnik, E. Pérez-Bernabeu, M. Cildoz, A. Agustin, and K. Wieczorowska-Tobis, "Robots for Elderly Care: Review, Multi-Criteria Optimization Model and Qualitative Case Study", *Healthcare (Basel, Switzerland)*, vol. 11, no. 9, p. 1286, 2023. DOI: 10.3390/healthcare11091286.

[59] T. Tanioka, "Nursing and Rehabilitative Care of the Elderly Using Humanoid Robots", *The Journal of Medical Investigation*, vol. 66, no. 1.2, pp. 19–23, 2019. DOI: 10.2152/jmi.66.19.

[60] B. D. Carolis, V. Carofiglio, I. Grimandli, N. Macchiarulo, G. Palestra, and O. Pino, "Using the Pepper Robot in Cognitive Stimulation Therapy for People with Mild Cognitive Impairment and Mild Dementia", in *ACHI 2020, The Thirteenth International Conference on Advances in Computer-Human Interactions*, Valencia, Spain: ThinkMind, 2020, pp. 452–457.

[61] A. Osorio and S. Caballa, "BeeBot: A Robot that Help Children Manage Their Blood Glucose in a Friendly Way", ser. HRI '23, New York, NY, USA: Association for Computing Machinery, 2023, pp. 841–844. DOI: 10.1145/3568294.3580197.

[62] *Eilik – Energize Lab.* [Online]. Available: https://store.energizelab.com/products/eilik (visited on 02/08/2024).

[63] *Kiki Robot | Zoetic AI Official Website.* [Online]. Available: http://www.zoeticai.com/ (visited on 02/08/2024).

[64] *Autodesk Inventor: software de modelado 3D para diseñadores e ingenieros*, 2024. [Online]. Available: https://www.autodesk.com/es/products/inventor/ (visited on 04/08/2024).

[65] *Autodesk AutoCAD: software de diseño y dibujo en el que confían millones de usuarios*, 2024. [Online]. Available: https://www.autodesk.com/es/products/autocad/ (visited on 04/08/2024).

[66] *Piskel - Free Online Editor for Animated Sprites & Pixel Art*, 2024. [Online]. Available: https://www.piskelapp.com/ (visited on 04/08/2024).

[67] B. Wirtz, *Simple Tools to Make Animation Sprites*, 2023. [Online]. Available: https://www.gamedesigning.org/animation/sprites/ (visited on 04/08/2024).

[68] *Impresora 3D Original Prusa i3 MK3S+*, 2024. [Online]. Available: https://www.prusa3d.com/es/producto/impresora-3d-original-prusa-i3-mk3s-3/ (visited on 04/08/2024).

[69] *Bambu Lab X1 Series*, 2024. [Online]. Available: https://bambulab.com/en-eu/x1 (visited on 04/08/2024).

[70] *Layers and Perimeters*, 2020. [Online]. Available: https://help.prusa3d.com/article/layers-and-perimeters_1748 (visited on 04/08/2024).

[71] M. Hanon, R. Marczis, and L. Zsidai, "Influence of the 3D printing process settings on tensile strength of PLA and HT-PLA", *Periodica Polytechnica, Mechanical Engineering*, vol. 65, pp. 38–46, Nov. 2020. DOI: 10.3311/PPme.13683.

[72] *Infill Patterns*, 2023. [Online]. Available: https://help.prusa3d.com/article/infill-patterns_177130 (visited on 04/08/2024).

[73] *Organic Supports*, 2023. [Online]. Available: https://help.prusa3d.com/article/organic-supports_480131 (visited on 04/08/2024).

[74] *Warping*, 2021. [Online]. Available: https://help.prusa3d.com/article/warping_2011 (visited on 04/08/2024).

[75] *Stringing and oozing*, 2020. [Online]. Available: https://help.prusa3d.com/article/stringing-and-oozing_1805 (visited on 04/08/2024).

[76] *PRUSA Forum*, 2021. [Online]. Available: https://forum.prusa3d.com/ (visited on 04/08/2024).

[77] Arduino, *Arduino*, 2024. [Online]. Available: https://www.arduino.cc/ (visited on 04/08/2024).

[78] Visual Studio Code, *Code Editing. Redefined*, 2024. [Online]. Available: https://code.visualstudio.com/ (visited on 04/08/2024).

[79] git, *Git --everything-is-local*, 2024. [Online]. Available: https://git-scm.com/ (visited on 04/08/2024).

[80] GitHub, *GitHub Dashboard*, 2024. [Online]. Available: https://github.com/ (visited on 04/08/2024).

[81] Standard C++ Foundation, *ISO C++*, 2024. [Online]. Available: https://isocpp.org/ (visited on 04/08/2024).

[82] Python Software Foundation, *Python*, 2024. [Online]. Available: https://www.python.org/ (visited on 04/08/2024).

[83] Docker Inc., *Develop faster. Run anywhere.* 2024. [Online]. Available: https://www.docker.com/ (visited on 04/08/2024).

[84] Docker Inc., *Docker Compose Overview*, 2024. [Online]. Available: https://docs.docker.com/compose/ (visited on 04/08/2024).

[85] Open Robotics, *ROS - Robot Operating System*, 2021. [Online]. Available: https://www.ros.org/ (visited on 04/08/2024).

[86] Broadcom, *RabbitMQ. One broker to queue them all*, 2024. [Online]. Available: https://www.rabbitmq.com/ (visited on 04/08/2024).

[87] Pollen Robotics, *This is Reachy. He is pretty unique*, 2024. [Online]. Available: https://www.pollen-robotics.com/reachy/ (visited on 05/08/2024).

[88] M. E. Rosheim, "Multiple Rotatable Links Robotic Manipulator", US5893296A, US Patent 5, 893, 296, Apr. 1999. [Online]. Available: https://patents.google.com/patent/US5893296A/en.

[89] M. E. Rosheim and G. F. Sauter, "New high-angulation omnidirectional sensor mount", in *Free-Space Laser Communication and Laser Imaging II*, J. C. Ricklin and D. G. Voelz, Eds., International Society for Optics and Photonics, vol. 4821, SPIE, 2002, pp. 163–174. DOI: 10.1117/12.465912.

[90] E. Chang-Siu, A. Snell, B. W. McInroe, X. Balladarez, and R. J. Full, "How to use the OmniWrist III for dexterous motion: An exposition of the forward and inverse kinematic relationships", *Mechanism and Machine Theory*, vol. 168, p. 104 601, 2022, ISSN: 0094-114X. DOI: https://doi.org/10.1016/j.mechmachtheory.2021.104601.

[91] K. Abe, K. Tadakuma, and R. Tadakuma, "ABENICS: Active Ball Joint Mechanism With Three-DoF Based on Spherical Gear Meshings", *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1806–1825, 2021. DOI: 10.1109/TRO.2021.3070124.

[92] ROBOTIS, *XL-320*, 2024. [Online]. Available: https://emanual.robotis.com/docs/en/dxl/x/xl320/ (visited on 05/08/2024).

[93] ROBOTIS, *XL430-W250-T*, 2024. [Online]. Available: https://emanual.robotis.com/docs/en/dxl/x/xl430-w250/ (visited on 05/08/2024).

[94] Tectonic, *TEBM36S05-4*, 2024. [Online]. Available: https://www.tectonicaudiolabs.com/product/tebm36s05-4/ (visited on 05/08/2024).

[95] Mikroe, *2x5W AMP Click*, 2024. [Online]. Available: https://www.mikroe.com/2x5w-amp-click (visited on 05/08/2024).

[96] Farnell, *Midas Displays MCT070HDMI-B-CTP*, 2024. [Online]. Available: https://es.farnell.com/midas/mct070hdmi-b-ctp/pantalla-lcd-tft-rva-horizontal/dp/2888572 (visited on 05/08/2024).

[97] Nitecore, *Tube V2.0*, 2024. [Online]. Available: https://flashlight.nitecore.com/product/tubev2.0 (visited on 05/08/2024).

[98] DFRobot, *Ambient Light Sensor (SKU: SEN0390)*, 2023. [Online]. Available: https://wiki.dfrobot.com/Ambient_Light_Sensor_0_200klx_SKU_SEN0390 (visited on 05/08/2024).

[99] Adafruit, *Adafruit 12-Key Capacitive Touch Sensor Breakout - MPR121*, 2023. [Online]. Available: https://www.adafruit.com/product/1982 (visited on 05/08/2024).

[100] Mikroe, *Smart Sens Click*, 2023. [Online]. Available: https://www.mikroe.com/smart-sens-click (visited on 05/08/2024).

[101] Seeed Studio, *ReSpeaker Mic Array v2.0*, 2024. [Online]. Available: https://wiki.seeedstudio.com/ReSpeaker_Mic_Array_v2.0/ (visited on 05/08/2024).

[102] Intel, *Intel RealSense D405*, 2022. [Online]. Available: https://www.intelrealsense.com/depth-camera-d405/ (visited on 05/08/2024).

[103] Ultraleap, *Leap Motion Controller*, 2016. [Online]. Available: https://www.ultraleap.com/leap-motion-controller-whats-included/ (visited on 05/08/2024).

[104] DFRobot, *Gravitiy ENS160 Air Quality Sensor*, 2024. [Online]. Available: https://www.dfrobot.com/product-2526.html (visited on 05/08/2024).

[105] DFRobot, *Fermion MEMS Gas Sensor MiCS-5524*, 2022. [Online]. Available: https://www.dfrobot.com/product-2419.html (visited on 05/08/2024).

[106] Mikroe, *NFC Click*, 2024. [Online]. Available: https://www.mikroe.com/nfc-click (visited on 05/08/2024).

[107] DFRobot, *Gravity I2C Digital Wattmeter*, 2024. [Online]. Available: https://www.dfrobot.com/product-1827.html (visited on 05/08/2024).

[108] Mikroe, *RTC 16 Click*, 2024. [Online]. Available: https://www.mikroe.com/rtc-16-click (visited on 05/08/2024).

[109] Mikroe, *Buck 6 Click*, 2024. [Online]. Available: https://www.mikroe.com/buck-6-click (visited on 05/08/2024).

[110] Arduino, *Arduino Mega 2560 Rev3*, 2024. [Online]. Available: https://store.arduino.cc/products/arduino-mega-2560-rev3 (visited on 05/08/2024).

[111] Arduino, *Arduino Mega Proto Shield Rev3 (PCB)*, 2024. [Online]. Available: https://store.arduino.cc/products/arduino-mega-proto-shield-rev3-pcb (visited on 05/08/2024).

[112] ROBOTIS, *Dynamixel Shield*, 2024. [Online]. Available: https://emanual.robotis.com/docs/en/parts/interface/dynamixel_shield/ (visited on 05/08/2024).

[113] NVIDIA, *Extend AI Computing with the Jetson TX2i*, 2024. [Online]. Available: https://developer.nvidia.com/embedded/jetson-tx2i (visited on 05/08/2024).

[114] Ezurio, *FlexMIMO 6E MIMO Internal Antenna*, 2024. [Online]. Available: https://www.ezurio.com/internal-antennas/24-5-6-ghz-wi-fi-and-bluetooth/flexmimo-flexmimo-6e-mimo-internal-antenna (visited on 05/08/2024).

[115] Crucial, *Crucial MX500 1 TB 3D NAND SATA Internal SSD*, 2024. [Online]. Available: https://www.crucial.es/ssd/mx500/ct1000mx500ssd1k (visited on 05/08/2024).

[116] HDI Battery, *Arduino Mega 2560 Rev3*, 2024. [Online]. Available: https://www.hdibattery.com/p11807854_bateria-de-litio-4s4p-18650-14-4v-10-4-ah-con-pcb.html (visited on 05/08/2024).

[117] TP-LINK, *Batería de Litio 4S4P 18650 14.4V 10.4 Ah con PCB*, 2024. [Online]. Available: https://www.tp-link.com/en/home-networking/usb-a-hub-and-converter/uh700/ (visited on 05/08/2024).

[118] M. Franca, N. Bolognini, and M. Brysbaert, "Seeing emotions in the eyes: A validated test to study individual differences in the perception of basic emotions", *Cognitive Research: Principles and Implications*, vol. 8, no. 1, p. 67, Nov. 2023, ISSN: 2365-7464. DOI: 10.1186/s41235-023-00521-x.

[119] B. Blanchon, *ArduinoJSON*, 2024. [Online]. Available: https://arduinojson.org/ (visited on 08/08/2024).

[120] AgileWare, *CircularBuffer*, 2024. [Online]. Available: https://www.arduino.cc/reference/en/libraries/circularbuffer/ (visited on 08/08/2024).

[121] A. S. Jensen, *EasyLogger*, 2024. [Online]. Available: https://www.arduino.cc/reference/en/libraries/easylogger/ (visited on 08/08/2024).

[122] Arduino Store, *Arduino GIGA R1 WiFi*, 2024. [Online]. Available: https://store.arduino.cc/products/giga-r1-wifi (visited on 08/08/2024).

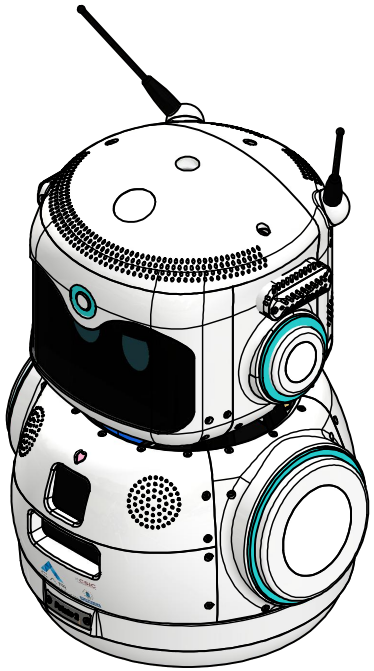
[123] T. Kitamura, *Fusion2URDF*, GitHub repository, 2020. [Online]. Available: https://github.com/syuntoku14/fusion2urdf (visited on 12/08/2024).

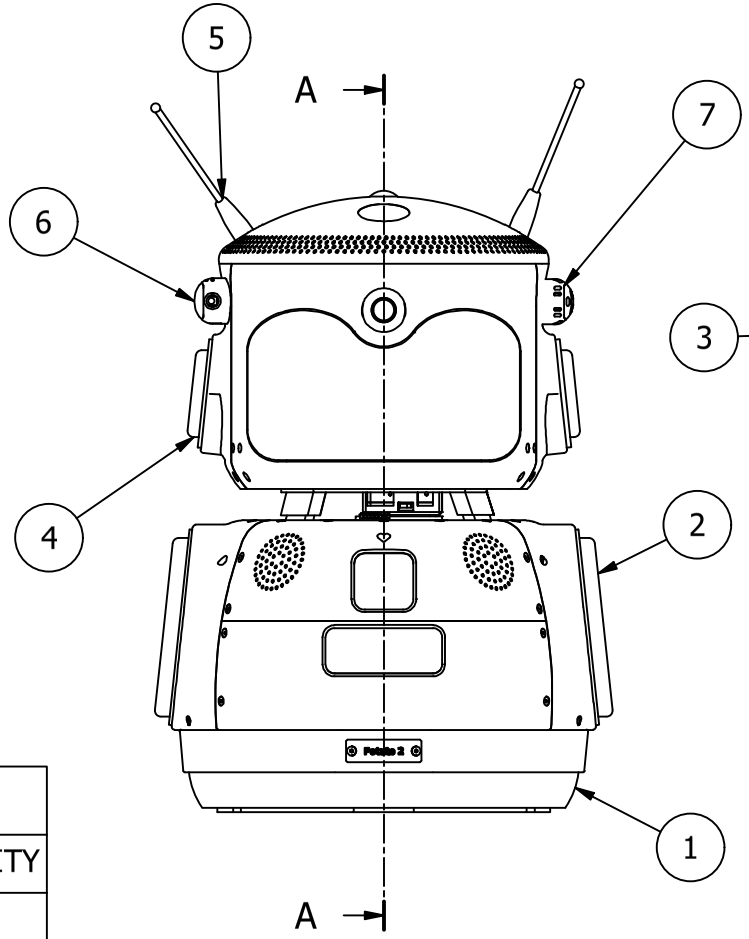
Appendix **A**

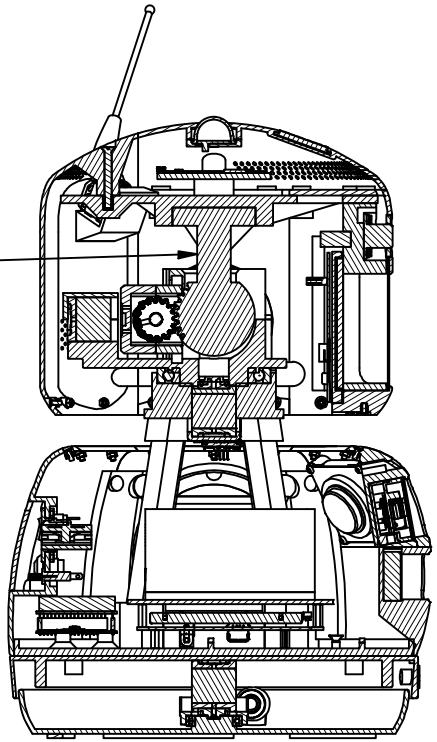
Social Robot Technical Drawings

The appendix presents the detailed technical drawings of the robot's main assembly and its principal sub-assemblies. These drawings provide a comprehensive overview of the structural components and the mechanical design of the robot, serving as a visual reference for the construction and assembly processes. The following technical drawings are included:

- **DRAWING 1.00:** POTATO 2
- **DRAWING 1.01:** BASE
- **DRAWING 1.02:** BODY
- **DRAWING 1.03:** NECK MECHANISM
- **DRAWING 1.04:** HEAD
- **DRAWING 1.05:** ANTENNA
- **DRAWING 1.06:** FLASHLIGHT MODULE
- **DRAWING 1.07:** GAS SENSOR MODULE







SECTION A-A

PARTS LIST			
ELEMENT Nº	CODE	DESCRIPTION	QUANTITY
1	P-101	BASE	1
2	P-102	BODY	1
3	P-103	NECK MECHANISM	1
4	P-104	HEAD	1
5	P-105	ANTENNA	2
6	P-106	FLASHLIGHT MODULE	1
7	P-107	GAS SENSOR MODULE	1

DATE: 12/08/2024

AUTHOR: Daniel Sotelo Aguirre

SIGNATURE:

INDUSTRIALES ETSII | UPM

POLITÉCNICA

PROJECT: DESIGN, CONSTRUCTION AND PROGRAMMING OF A SOCIAL ROBOT FOR PERSONAL ASSISTANCE

DENOMINATION: POTATO 2

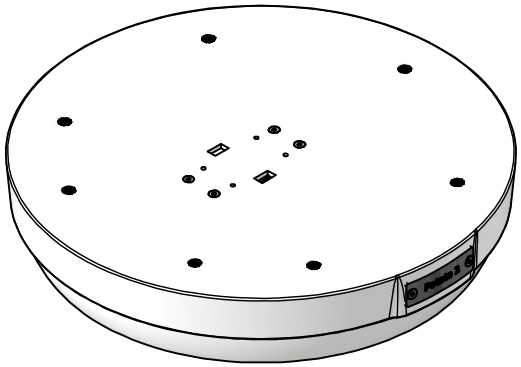
WEIGHT (kg): 8.357

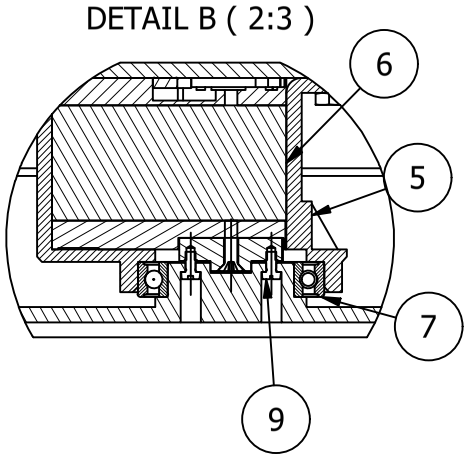
PLANE Nº: 1.00

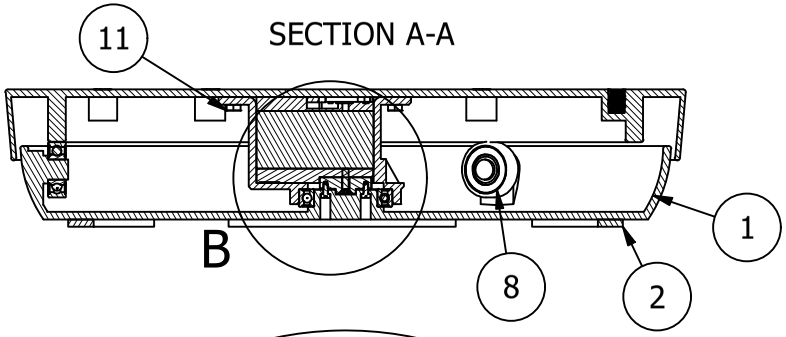
CODE: P-100

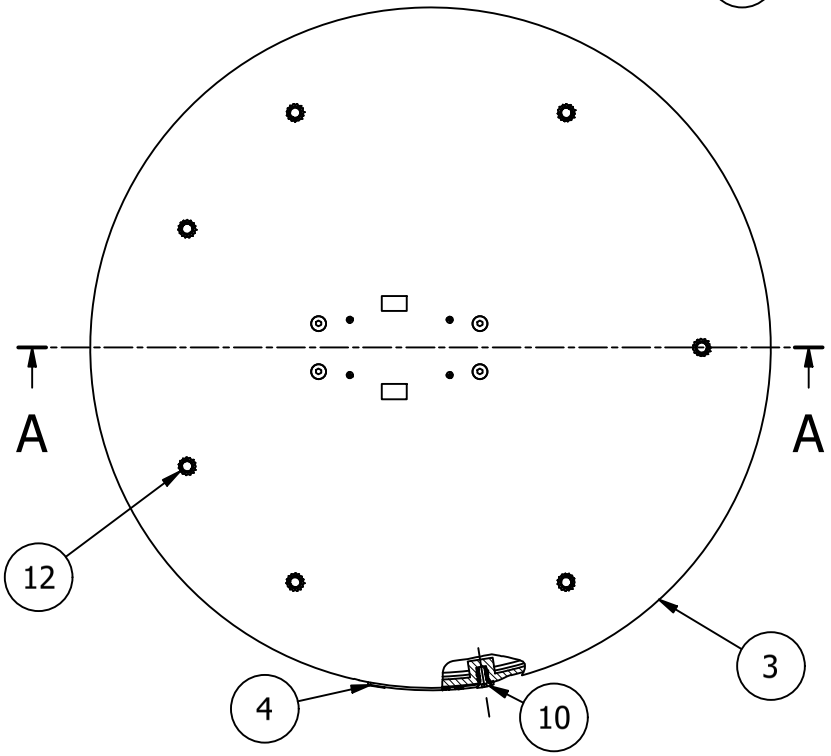
A4

SCALE: 1:5









PARTS LIST			
ELEMENT Nº	CODE	DESCRIPTION	QUANTITY
1	P-1011	BOTTOM BASE	1
2	P-1012	ANTI-SLIP PADS	5
3	P-1013	TOP BASE	1
4	P-1014	NAME PLATE	1
5	P-1015	SOPORTE MOTOR	1
6		DYNAMIXEL XL-430	1
7		25x37x7 mm BEARING	1
8		8x22x7 mm BEARING	3
9		M2 x 4 BOLT	4
10		M3 x 8 BOLT	6
11		M3 x 2.4 NUT	4
12		M5 THREADED INSERT	7

DATE: 12/08/2024

AUTHOR: Daniel Sotelo Aguirre

SIGNATURE:

PROJECT: DESIGN, CONSTRUCTION AND PROGRAMMING OF A SOCIAL ROBOT FOR PERSONAL ASSISTANCE

INDUSTRIALES ETSII | UPM

POLITÉCNICA

DENOMINATION: BASE

WEIGHT (kg): 0.841

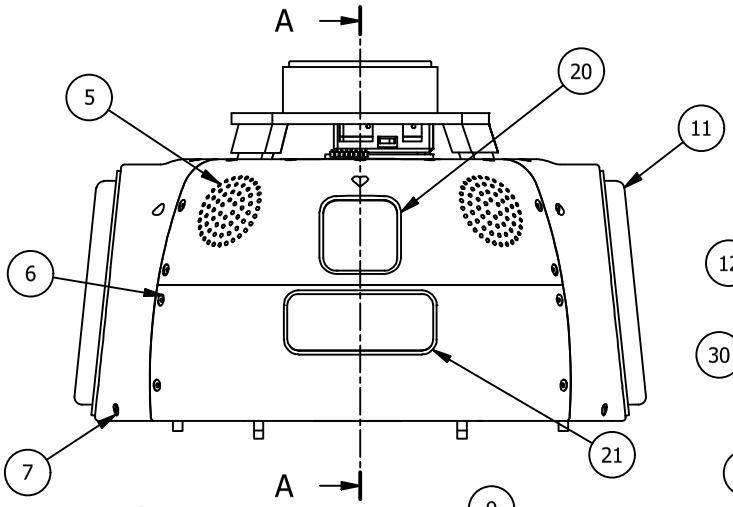
PLANE Nº: 1.01

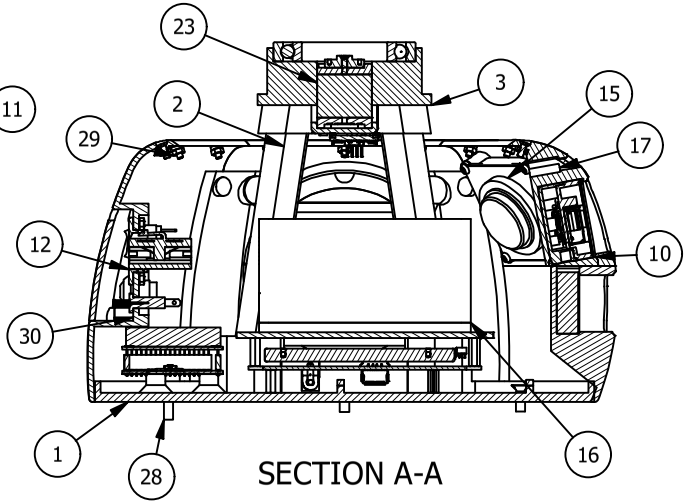
CODE: P-101

A4

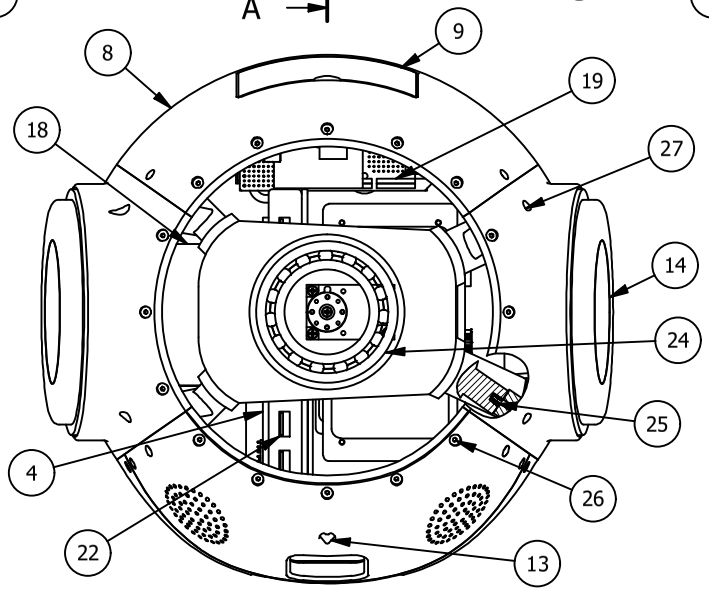
SCALE: 1:3

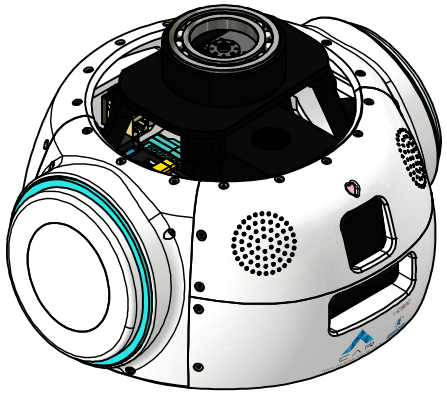
PARTS LIST			
ELEMENT Nº	CODE	DESCRIPTION	QUANTITY
1	P-1021	BODY SUPPORT	1
2	P-1022	LEG	4
3	P-1023	MECHANISM SUPPORT	1
4	P-1024	USB-HUB SUPPORT	1
5	P-1025	FRONT TOP PANEL	1
6	P-1026	FRONT BOTTOM PANEL	1
7	P-1027	LATERAL PANEL	2
8	P-1028	BACK PANEL	1
9	P-1029	PORTS PANEL COVER	1
10	P-10210	CAMERA COVER	1
11	P-10211	LATERAL COVER	2
12	P-10212	PORTS PANEL	1
13	P-10213	LED COVER	3
14	P-10214	TACTILE PAD	1
15		SPEAKER	2
16		NVIDIA JETSON TX2i	1
17		LED ring	3
18		BATTERY	1
19		ARDUINO MEGA 2560	1
20		INTEL REALSENSE D405 CAMERA	1
21		LEAP MOTION CONTROLLER	1
22		USB-HUB	1
23		DYNAMIXEL XL-430	1
24		45x75x10 mm BEARING	1
25		M3 THREADED INSERT	28
26		M3 x 8 BOLT	36
27		M3 x 20 BOLT	8
28		M5 x 20 BOLT	7
29		M3 x 2.4 NUT	16
30		MAGNET	16



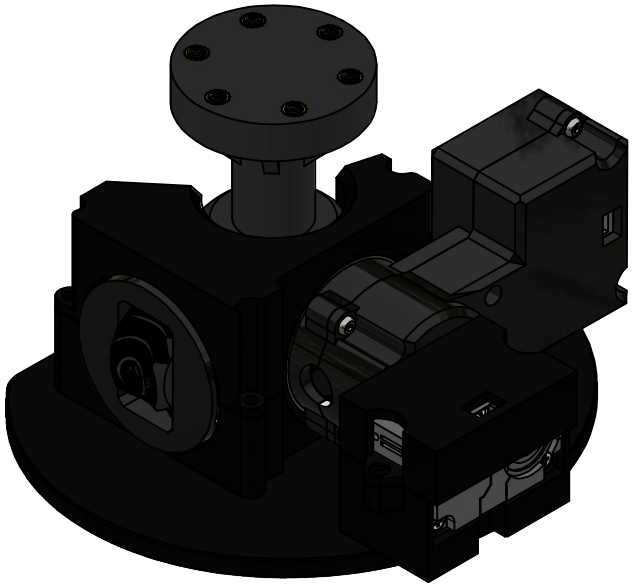


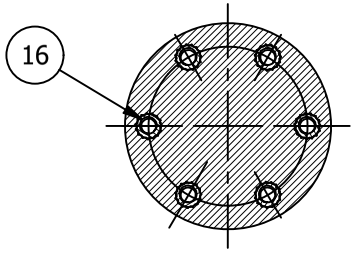
SECTION A-A





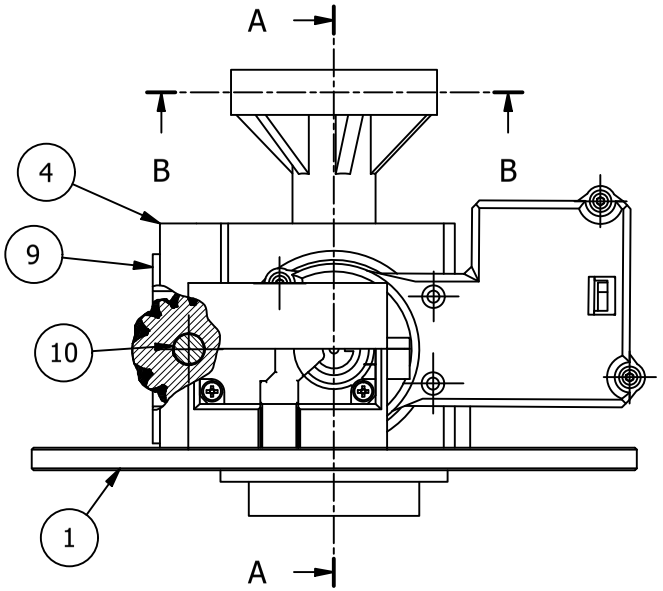
DATE: 12/08/2024	SIGNATURE:	PROJECT: DESIGN, CONSTRUCTION AND PROGRAMMING OF A SOCIAL ROBOT FOR PERSONAL ASSISTANCE	
AUTHOR: Daniel Sotelo Aguirre			
INDUSTRIALES ETSII | UPM — POLITÉCNICA	DENOMINATION: BODY		WEIGHT (kg): 4.51
	PLANE Nº: 1.02	CODE: P-102	A4 — SCALE: 1:4

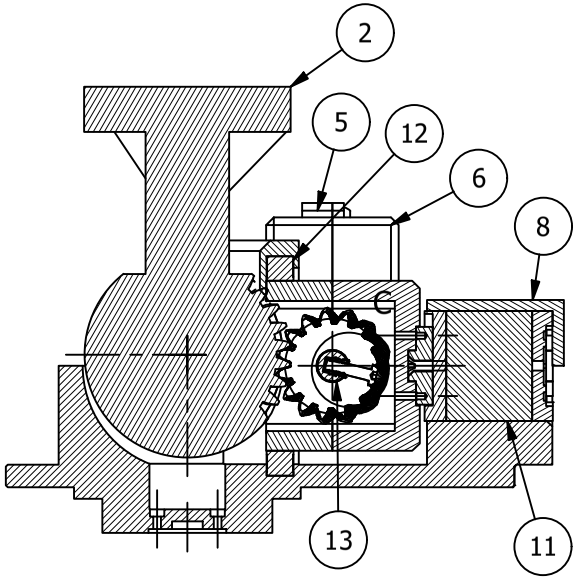




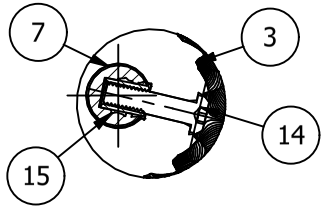
SECTION B-B

SECTION A-A





DETAIL C (1:1)



PARTS LIST			
ELEMENT Nº	CODE	DESCRIPTION	QUANTITY
1	P-1031	BOTTOM BASE	1
2	P-1032	SPHERICAL GEAR	1
3	P-1033	SHERICAL PINION	2
4	P-1034	GEAR COVER	1
5	P-1035	MOTOR PINION CASE	1
6	P-1036	MOTOR PINION COVER	1
7	P-1037	MOTOR PINION AXIS	1
8	P-1038	MOTOR COVER	1
9	P-1039	SUPPORT PINION CASE	1
10	P-10310	SUPPORT PINION AXIS	1
11		DYNAMIXEL XL-430	2
12		45x58x7 mm BEARING	2
13		8x12x3.5 mm BEARING	3
14		M3 x 12 BOLT	5
15		M3 THREADED INSERT	1
16		M5 THREADED INSERT	6

DATE: 12/08/2024

SIGNATURE:

AUTHOR: Daniel Sotelo Aguirre

INDUSTRIALES ETSII | UPM

POLITÉCNICA

PROJECT: DESIGN, CONSTRUCTION AND PROGRAMMING OF A SOCIAL ROBOT FOR PERSONAL ASSISTANCE

DENOMINATION: NECK MECHANISM

WEIGHT (kg): 0.741

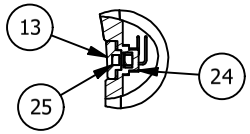
PLANE Nº: 1.03

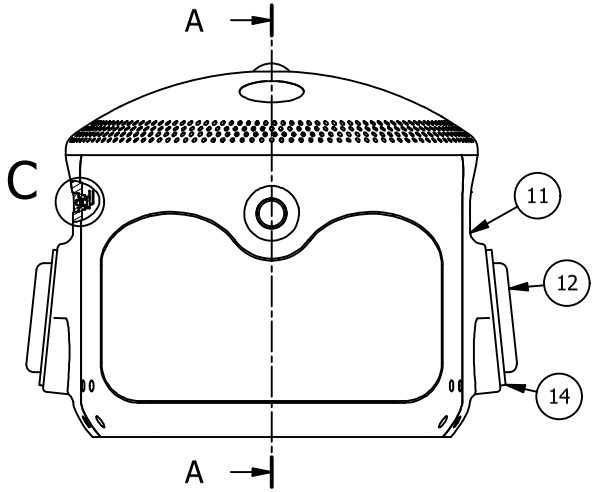
CODE: P-103

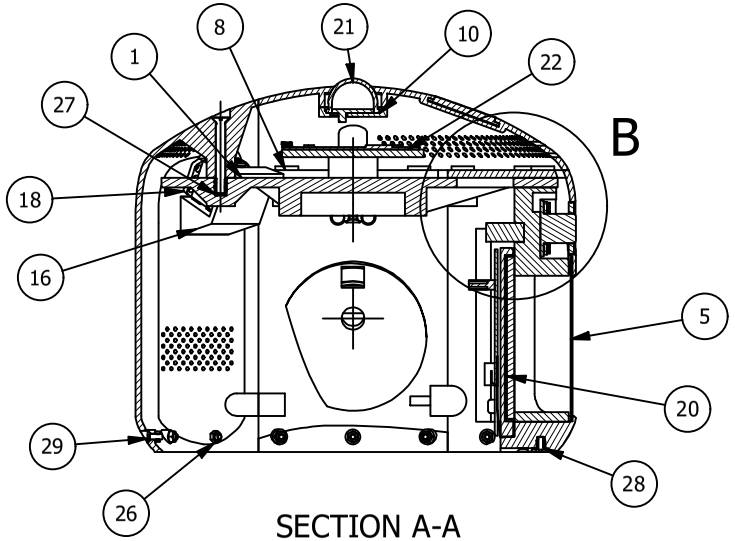
A4

SCALE: 1:2

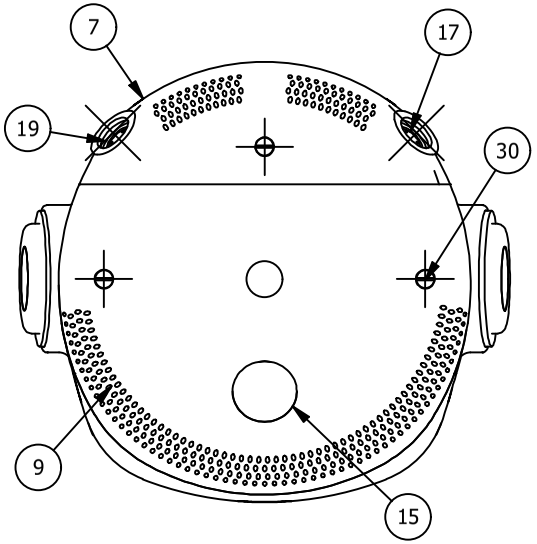
DETAIL C (1:2)

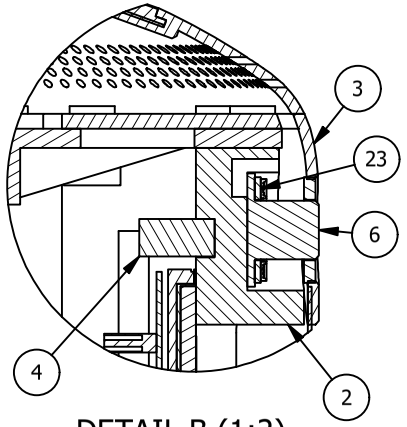




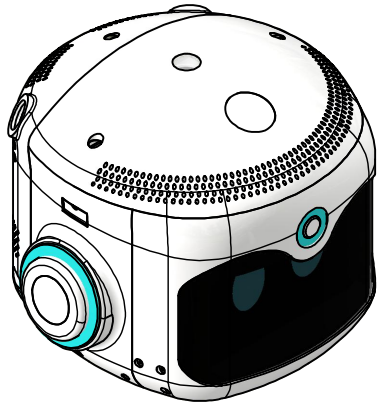


SECTION A-A





DETAIL B (1:2)



PARTS LIST			
ELEMENT Nº	CODE	DESCRIPTION	QUANTITY
1	P-1041	TOP SUPPORT BASE	1
2	P-1042	LCD DISPLAY FRAME	1
3	P-1043	FRONT PANEL	1
4	P-1044	LCD DISPLAY LOCK	1
5	P-1045	PVC PANEL	1
6	P-1046	FRONT LED RING SUPPORT	1
7	P-1047	BACK PANEL	1
8	P-1048	MICROPHONE SUPPORT	1
9	P-1049	TOP PANEL	1
10	P-10410	LDR SUPPORT	1
11	P-10411	LATERAL PANEL	2
12	P-10412	EAR COVER	2
13	P-10413	PIN PORT COVER	2
14	P-10414	LED COVER	3
15	P-10415	TACTILE PAD	3
16	P-10416	MOTOR SUPPORT	2
17	P-10417	ANTENNA MOTOR LINK	2
18		DYNAMIXEL XL-320	2
19		15x24x5 mm BEARING	2
20		LCD DISPLAY	1
21		LDR SENSOR	1
22		MICROPHONE ARRAY	1
23		LED RING	3
24		MALE 6-PIN CONNECTOR	2
25		MAGNET	24
26		M3x2.4 NUT	18
27		M5 THREADED INSERT	13
28		M3 THREADED INSERT	21
29		M3 x 8 BOLT	34
30		M5 x 20 BOLT	16

DATE: 12/08/2024

SIGNATURE:

AUTHOR: Daniel Sotelo Aguirre

INDUSTRIALES ETSII | UPM

POLITÉCNICA

PROJECT: DESIGN, CONSTRUCTION AND PROGRAMMING OF A SOCIAL ROBOT FOR PERSONAL ASSISTANCE

DENOMINATION: HEAD

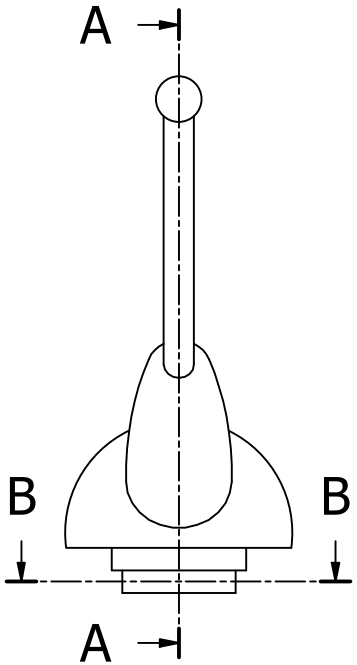
WEIGHT (kg): 2.103

PLANE Nº: 1.04

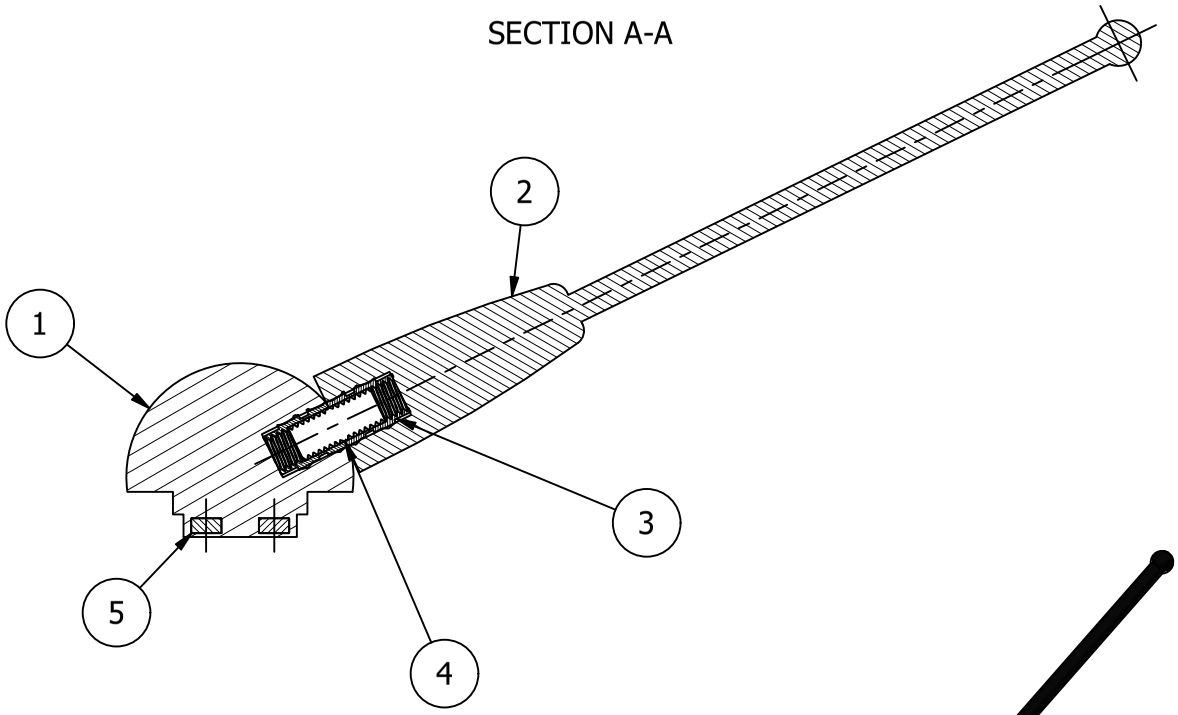
CODE: P-104

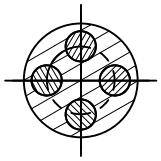
A4

SCALE: 1:4

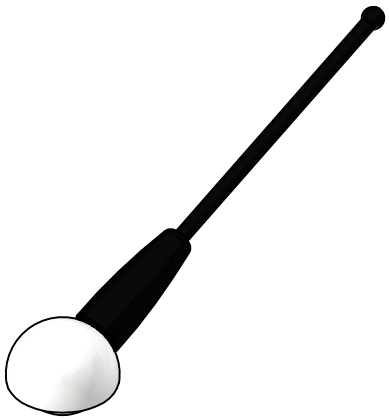


SECTION A-A





SECTION B-B



PARTS LIST			
ELEMENT Nº	CODE	DESCRIPTION	QUANTITY
1	P-1051	SPHERICAL BASE	1
2	P-1052	ANTENNA TIP	1
3		M5 THREADED INSERT	2
4		STUD BOLT	1
5		MAGNET	4

DATE: 12/08/2024

AUTHOR: Daniel Sotelo Aguirre

SIGNATURE:

INDUSTRIALES ETSII | UPM

POLITÉCNICA

PROJECT: DESIGN, CONSTRUCTION AND PROGRAMMING OF A SOCIAL ROBOT FOR PERSONAL ASSISTANCE

DENOMINATION: ANTENNA

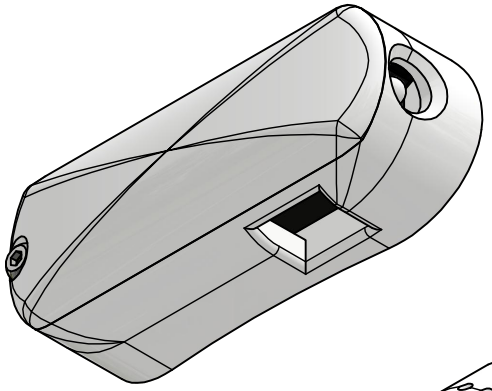
WEIGHT (kg): 0.023

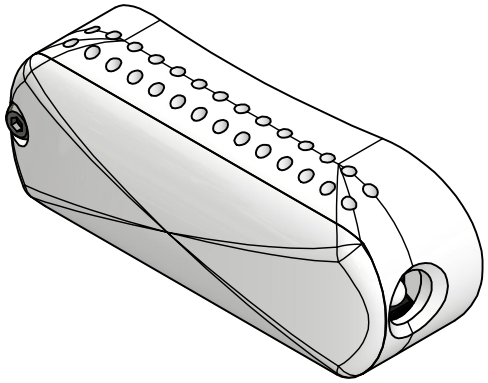
PLANE Nº: 1.05

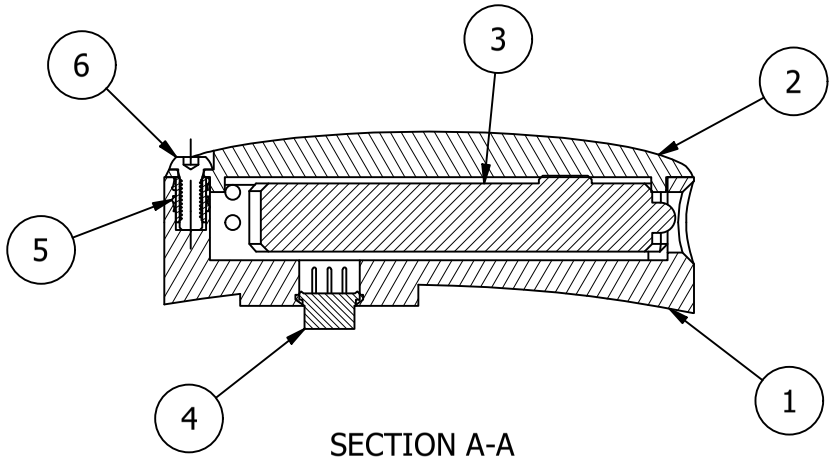
CODE: P-105

A4

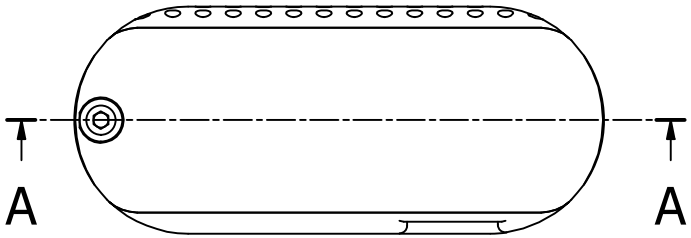
SCALE: 1:1







SECTION A-A



PARTS LIST			
ELEMENT Nº	CODE	DESCRIPTION	QUANTITY
1	P-1061	MODULE BODY	1
2	P-1062	MODULE COVER	1
3		FLASHLIGHT	1
4		FEMALE 6-PIN CONNECTOR	1
5		M3 THREADED INSERT	1
6		M3 x 8 BOLT	1

DATE: 12/08/2024

SIGNATURE:

AUTHOR: Daniel Sotelo Aguirre

INDUSTRIALES ETSII | UPM

POLITÉCNICA

PROJECT: DESIGN, CONSTRUCTION AND PROGRAMMING OF A SOCIAL ROBOT FOR PERSONAL ASSISTANCE

DENOMINATION: FLASHLIGHT MODULE

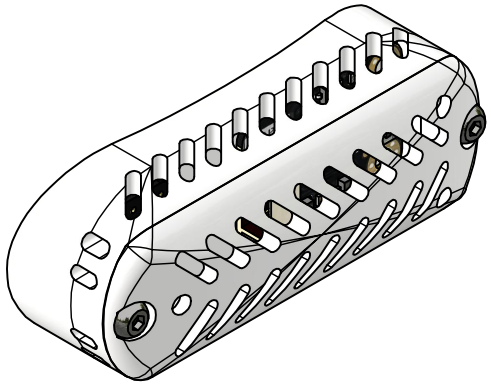
WEIGHT (kg): 0.048

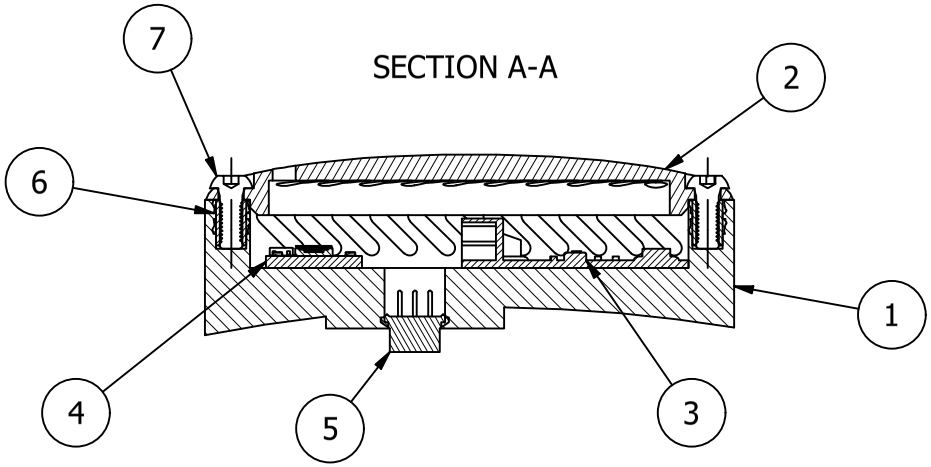
PLANE Nº: 1.06

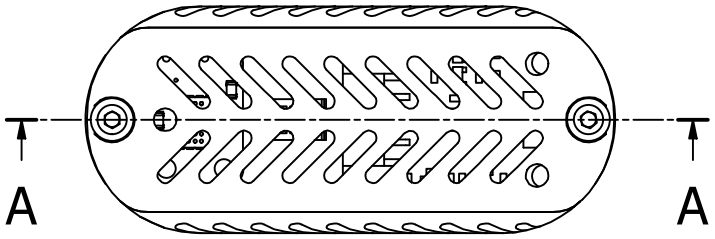
CODE: P-106

A4

SCALE: 1:1







PARTS LIST			
ELEMENT Nº	CODE	DESCRIPTION	QUANTITY
1	P-1071	MODULE BODY	1
2	P-1072	MODULE COVER	1
3		AIR QUALITY SENSOR	1
4		HAZARDOUS GAS SENSOR	1
5		FEMALE 6-PIN CONNECTOR	1
6		M3 THREADED INSERT	2
7		M3 x 8 BOLT	2

DATE: 12/08/2024

AUTHOR: Daniel Sotelo Aguirre

SIGNATURE:

PROJECT: DESIGN, CONSTRUCTION AND PROGRAMMING OF A SOCIAL ROBOT FOR PERSONAL ASSISTANCE

INDUSTRIALES ETSII | UPM

POLITÉCNICA

DENOMINATION: GAS SENSOR MODULE

WEIGHT (kg): 0.025

PLANE Nº: 1.07

CODE: P-107

A4

SCALE: 1:1

Appendix **B**

Applications and Impact Analysis

B.1. Applications

The field of social robotics is expanding rapidly, driven by the increasing need for robots that can interact naturally and effectively with humans in various settings. The social robot developed in this Master's Thesis is designed to be highly versatile, with potential applications across multiple domains. These include **personal assistance** for the **elderly and individuals with disabilities**, where the robot can provide **companionship**, remind users of **medication schedules**, and assist with **daily tasks**. In **educational environments**, the robot can serve as an interactive **learning tool**, engaging through NLP and emotional responses. Additionally, the robot can be deployed in **customer service roles**, where it can enhance user experience through personalized interactions, and in healthcare settings, where it can **support patients** by monitoring their well-being and providing real-time feedback to caregivers.

B.2. Impact Analysis

Developing and deploying this social robot involves **significant economic investment** in terms of research, development, manufacturing, and maintenance. However, the robot's ability to perform tasks traditionally carried out by humans, such as caregiving, customer service, and education, can lead to **long-term cost savings and increased efficiency**. The robot's modular design allows for customization and scalability, which can reduce costs over time by minimizing the need for multiple specialized robots.

The social impact of this robot is profound. By providing companionship and assistance to the elderly, individuals with disabilities, and young people with chronic conditions like diabetes, the robot **can significantly improve quality of life**. It **reduces loneliness**, promotes **independence**, and ensures better **adherence to health management routines**. In educational settings, the robot's interactive features can foster

engagement and motivation among students, leading to better learning outcomes. However, it is important to consider the **ethical implications** of automation, particularly the potential displacement of jobs. While the robot can perform tasks traditionally done by humans, it is essential to balance this with the creation of new job opportunities in robot maintenance, programming, and AI development.

A key legal consideration for the social robot is **data privacy**. To comply with regulations like the General Data Protection Regulation (GDPR), the robot will process and store user data **locally**, avoiding cloud-based systems and reducing security risks. This ensures senstitive information, such as health data, is kept secure. The system in the future could include **encryption** and **user consent** mechanisms, giving users control over their data. By adhering to **data minimization** principles, the robot will collect only necessary information, ensuring compliance with privacy laws while maintaining trust in healthcare and personal settings.

The environmental impact of the robot is a key consideration, particularly in terms of material usage and energy consumption. Constructed primarily from PETG plastic and metals, the robot's **materials contribute to its carbon footprint** through extraction, processing, and disposal. Its electronics also add to its impact. To mitigate these effects, **future iterations** should explore more **sustainable materials** and energy-efficient technologies. A comprehensive life cycle assessment would help identify ways to reduce the robot's environmental footprint across its production, use, and disposal stages.

B.3. Contribution to Sustainable Development Goals

The social robot developed in this project contributes to several Sustainable Development Goals (SDGs), as highlighted below and indicated in Figure B.1.

- **SDG 3 - Good Health and Well-Being:** The robot enhances the quality of healthcare by providing personalized assistance and monitoring, particularly for the elderly, individuals with disabilities, and young people with chronic conditions like diabetes, contributing to better health outcomes.
- **SDG 4 - Quality Education:** By serving as an interactive educational tool, the robot supports inclusive and equitable quality education, promoting lifelong learning opportunities for all.
- **SDG 8 - Decent Work and Economic Growth:** The robot's development and deployment create new job opportunities in the fields of robotics, AI, and healthcare technology, contributing to economic growth and innovation.
- **SDG 9 - Industry, Innovation, and Infrastructure:** The project advances technological innovation by integrating cutting-edge AI and sensor technologies into a practical and socially beneficial application.



Figure B.1. SDGs to which the project contributes.

- **SDG 12 - Responsible Consumption and Production:** Efforts to minimize the robot's environmental impact align with responsible consumption and production practices, particularly through the exploration of sustainable materials and energy-efficient design in future iterations.

Appendix C

Temporal Planning and Budget

C.1. Work Breakdown Structure (WBS)

The Work Breakdown Structure (WBS) shown in Figure C.1 provides a comprehensive overview of the project's phases, outlining the key tasks and milestones required to develop the social robot.

C.2. Planning

This section details the project timeline corresponding to the work packages defined in the WBS using a Gantt chart. The Gantt chart is shown in Figure C.2.

C.3. Budget

This budget section outlines the financial resources required for the project. It includes a breakdown of costs associated with materials, equipment, and personnel.

C.3.1. Material Costs

Table C.1 summarizes the expenses for materials used in the project, including materials, electronics, and other components necessary for the robot's construction.

C.3.2. Equipment Amortization Costs

The amortization costs of equipment used during the project, such as the 3D printers and the soldering station, are calculated distributing their initial costs over the useful life of these tools, as summarized in Table C.2.

C.3.3. Personnel Costs

Table C.3 accounts for the labor costs, including the time and expertise of the team members involved in the project, calculated based on their respective roles and hourly rates.

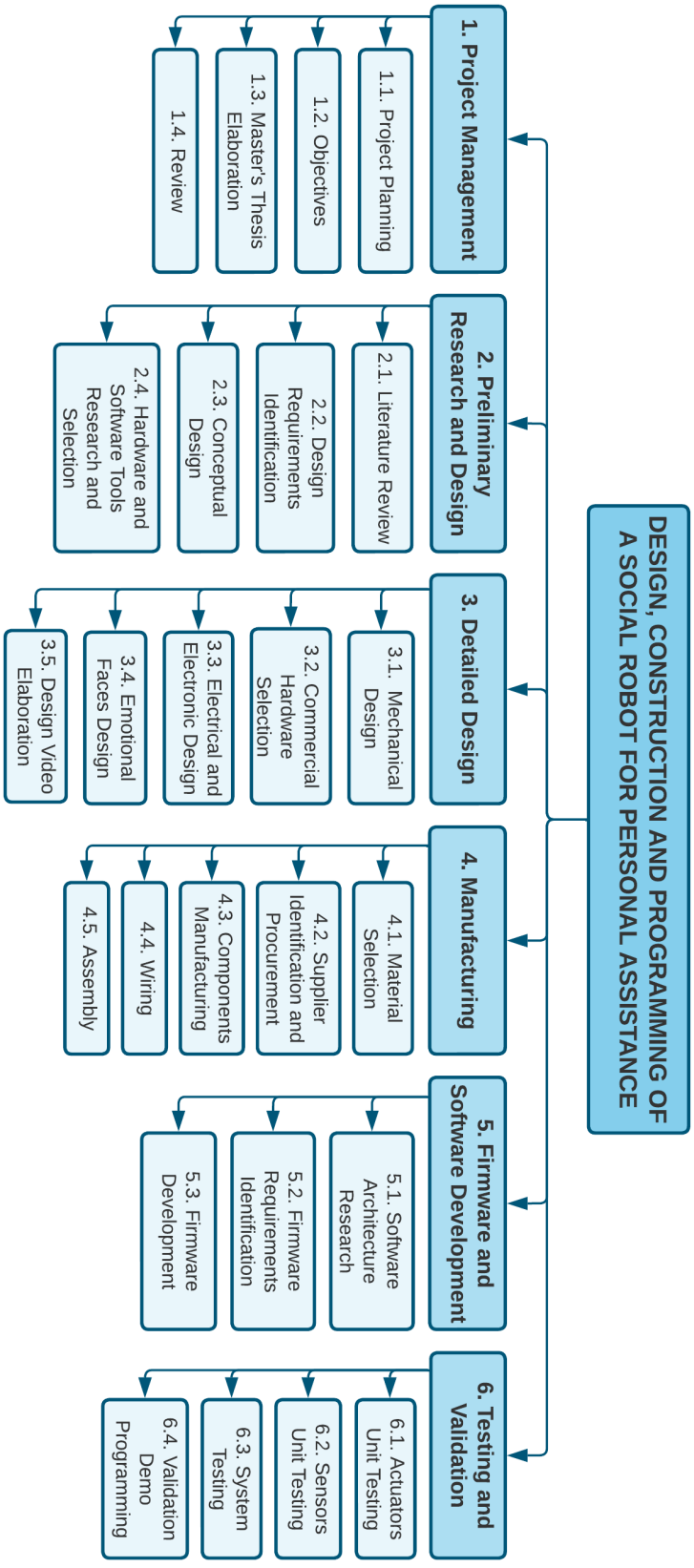


Figure C.1. Project WBS.

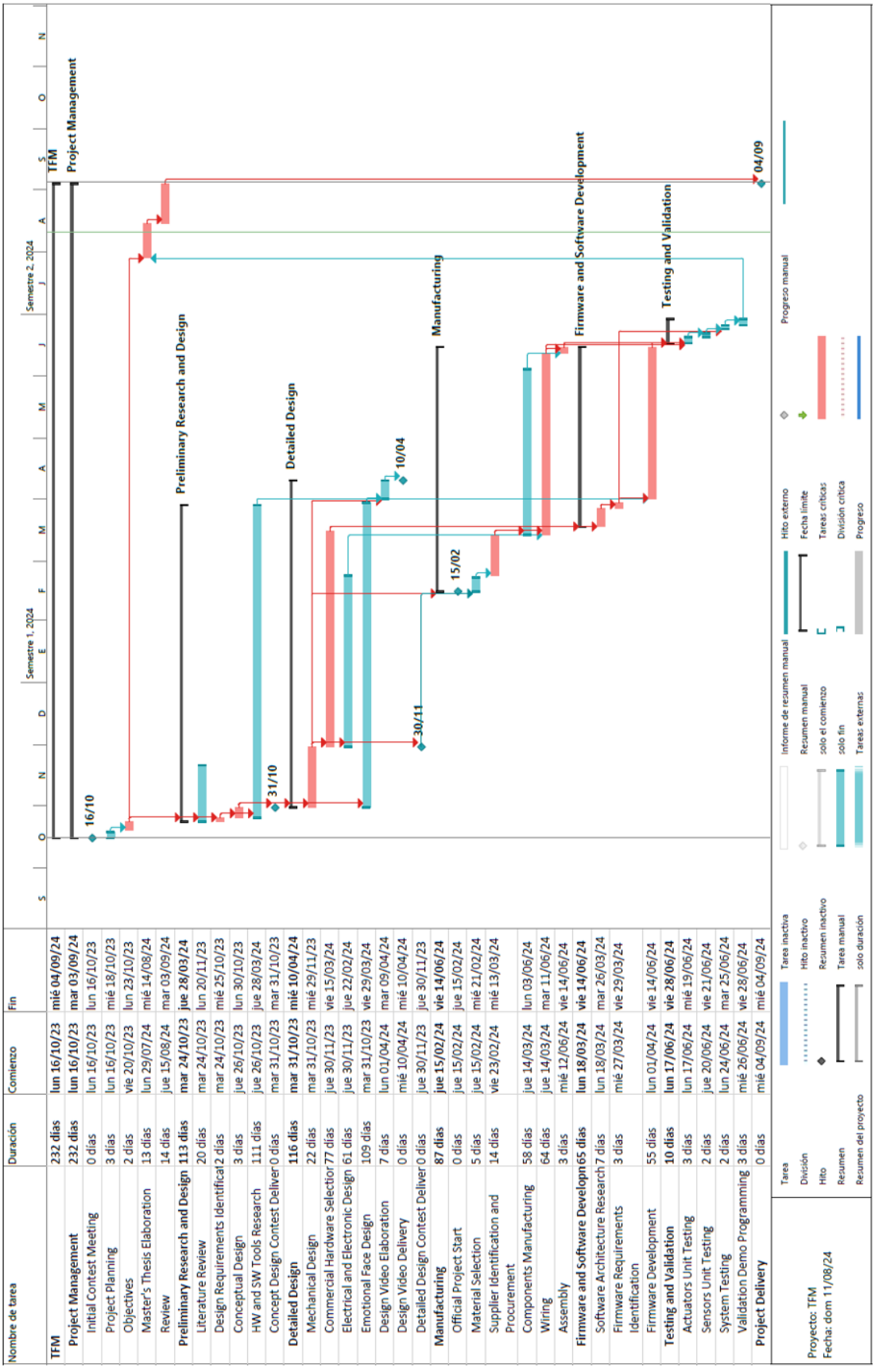


Figure C.2. Project Gantt Chart.

Concept	Units	Unit Price	Total Price
Construction Materials			**506,31 €**
PETG Filament	5	50,19 €	250,95 €
ABS Filament	1	43,10 €	43,10 €
PVC panel	1	3,84 €	3,84 €
Elastic Textile Mesh	1	9,01 €	9,01 €
Neodymium Magnets	6	17,42 €	104,52 €
Silicone Sheet	1	13,99 €	13,99 €
Copper Sheet	1	11,87 €	11,87 €
Roller Bearings	9	7,67 €	69,03 €
Polystyrene Panel	1	1,36 €	1,36 €
Anti-Slip Foam Adhesives	1	12,89 €	12,89 €
Post-Processing Materials			**80,32 €**
Priming Spray	1	14,00 €	14,00 €
Varnish Spray	1	7,50 €	7,50 €
Waterslide Decal Paper	1	18,99 €	18,99 €
Paint Spray	3	6,59 €	19,77 €
Sanding Paper	1	12,86 €	12,86 €
Cyanoacrylate	1	7,20 €	7,20 €
Electrical Assembly Materials			**493,77 €**
Fuse + Fuse Holder	1	0,44 €	0,44 €
Color Cables	1	32,49 €	32,49 €
Battery	1	175,93 €	175,93 €
Battery Charger	1	105,00 €	105,00 €
Other Cables	1	43,20 €	43,20 €
Ports and Connectors	1	89,75 €	89,75 €
Heat-Shrink Tube	1	7,99 €	7,99 €
Crimping Components	1	38,97 €	38,97 €
Total Material Cost			**3.477,56 €**

Concept	Units	Unit Price	Total Price
Electronic Hardware			**2.277,65 €**
SSD Shield	1	31,00 €	31,00 €
SSD 1TB	1	210,41 €	210,41 €
Jetson TX2 + Heatsink + Jetson Shield	1	673,00 €	673,00 €
Leap Motion Controller	1	134,20 €	134,20 €
Intel RealSense D405 Camera	1	279,28 €	279,28 €
LED Rings	2	22,08 €	44,16 €
USB Hub	1	64,80 €	64,80 €
Speakers + Audio Amplifier	1	50,40 €	50,40 €
LDR Sensor	1	10,47 €	10,47 €
IMU	1	30,29 €	30,29 €
Adafruit MPR121	1	22,76 €	22,76 €
XL-320 Motor	2	26,00 €	52,00 €
XL-430 Motor	4	48,27 €	193,08 €
Wi-Fi Antenna	1	10,45 €	10,45 €
Arduino Mega + Proto Shield	1	48,40 €	48,40 €
Dynamixel Shield	1	24,27 €	24,27 €
RTC	1	13,59 €	13,59 €
NFC	1	38,49 €	38,49 €
Microphone Array	1	64,00 €	64,00 €
Wattmeter	1	6,42 €	6,42 €
Serial UART Converter PCB	1	16,55 €	16,55 €
Flashlight + MOSFET	1	17,77 €	17,77 €
Air Quality Sensor	1	18,59 €	18,59 €
Hazardous Gas Detector	1	9,21 €	9,21 €
Buck Converter	2	25,23 €	50,46 €
LCD Display	1	163,60 €	163,60 €
Fasteners			**119,51 €**
M3 Threaded Inserts	1	20,07 €	20,07 €
M5 Threaded Inserts	1	39,37 €	39,37 €
Bolts	1	46,52 €	46,52 €
Washers	1	2,13 €	2,13 €
Nuts	1	11,42 €	11,42 €

Table C.1. Project Material Costs.

Concept	Total Cost	Years	Amortization Cost / Year	Use	Amortization Cost
Prusa MK3 3D Printer	999,00 €	5	199,80 €	6 months	99,90 €
Laser Cutting Machine	5.070,00 €	10	507,00 €	1 day	1,39 €
Welding Station	62,99 €	10	6,30 €	6 months	3,15 €
Painting Personal Protective Equipment	24,99 €	3	8,33 €	2 months	1,39 €
Toolbox	65,30 €	10	6,53 €	6 months	3,27 €
Hot Glue Gun	7,85 €	3	2,62 €	6 months	1,31 €
Heat Gun	29,99 €	5	6,00 €	6 months	3,00 €
Dremel Multi-Tool	35,40 €	5	7,08 €	6 months	3,54 €
Laptop	1.250,00 €	5	250,00 €	232 days	158,90 €
Total Amortization Cost					**275,84 €**

Table C.2. Project Amortization Costs.

Concept	Hour Cost	Hours	Cost
Project Author	7,00 €	460	3.220,00 €
Project Director 1	40,00 €	40	1.600,00 €
Predoctoral Researcher	20,00 €	100	2.000,00 €
Total Personnel Cost			**6.820,00 €**

Table C.3. Project Personnel Costs.

C.3.4. Total Cost

Table C.4 shows the total project costs calculation, combining material, equipment, and personnel expenses to give a complete financial overview of the project's expenditure.

Totals	Cost
Total Material Cost	3.477,56 €
Total Amortization Cost	275,84 €
Total Personnel Cost	6.820,00 €
TOTAL PROJECT COST	**10.573,40 €**

Table C.4. Project Total Cost.

This project was financed as part of the R&D project "Cognitive Personal Assistance for Social Environments (ACOGES)", reference PID2020-113096RB-I00, funded by MCIN/AEI/10.1303 9/501100011033.