**Final Report**

**“Smart Light”**

**1. Brief description of the project**

The project is a smart light system that is controlled remotely via a mobile app. The smart light provides users with the convenience of controlling lighting in their home environment from their smartphone, through Wi-Fi.

**2. Team members and their responsibilities during the project**

1. Scrum Master (Team Lead) – Khanh Do

• Role: Mr. Khanh will was responsible for ensuring the team follows agile methodologies, facilitating communication, and managing the workflow to meet project deadlines. He also helped with any roadblocks that the team encountered and ensured smooth collaboration among team members.

• Responsibilities:

o Organize and lead team meetings (stand-ups, sprints, retrospectives).

o Monitor the progress of the project and ensure tasks are completed on time.

o Facilitate communication between team members.

2. Backend Developer (Firmware/Embedded Systems) – Oumaima Oubihi

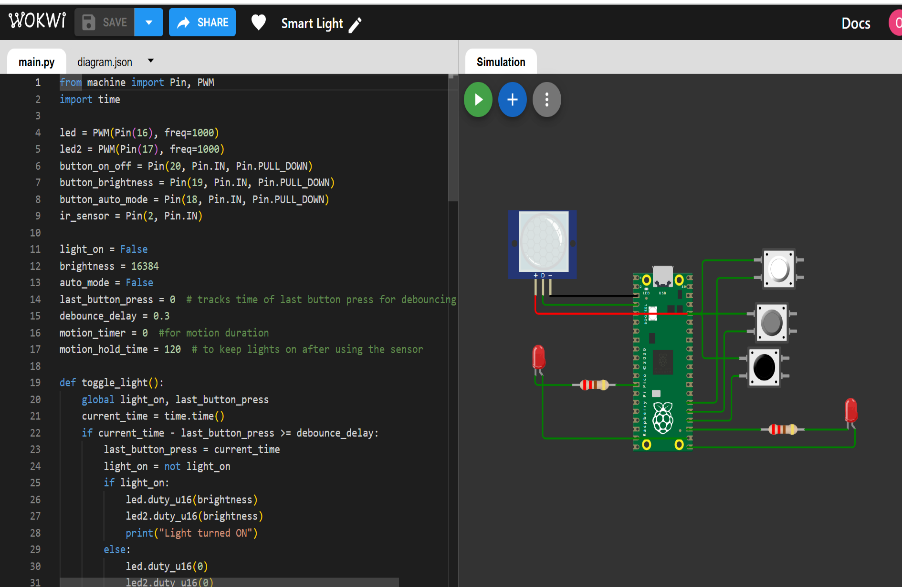
• Role: Ms. Oumaima was responsible for developing and integrating the embedded systems for the project. She focused on programming the Raspberry Pi Pico to control the smart light and created simulations using Wokwi to validate the implemented features.

• Responsibilities:

° Develop and optimize code for the Raspberry Pi Pico to control smart light operations.

° Test and validate features by creating and running simulations on Wokwi.

° Debug and troubleshoot issues in the embedded system during development.



3. Frontend Developer (Mobile App Development) – Jiaxin Li

• Role: Ms. Jiaxin was responsible for developing the mobile app interface that allows users to control the smart light. She focused on user interface design and user experience.

• Responsibilities:

° Develop the mobile app for Android/iOS using appropriate frameworks (e.g., Java, Swift, Flutter).

° Implement features like turning the light on/off.

° Ensure the app’s design is user-friendly and integrates well with the backend.



4. Hardware Engineer (Circuit Design and Power Management) – Oskari Laitinen

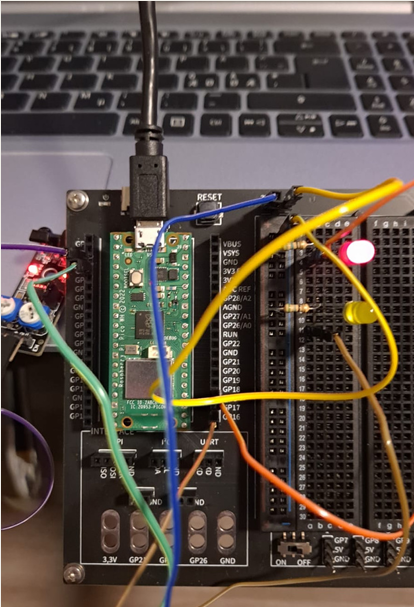
• Role: Mr. Oskari managed the hardware components of the project, ensuring the circuit design is robust and the system operates efficiently with minimal power consumption, and handle the WiFi connection to control the smart light.

• Responsibilities:

° Design the electrical circuit for the smart light, including the connection to the Raspberry Pi Pico.

° Test and troubleshoot hardware components like the light, resistors, and power supply.

° Handle communication protocols (Wi-Fi/Bluetooth) for data transmission.



**3. Project plan**

**Project Roadmap:**

#### 1. Planning and Setup

* Gather and document project requirements.
* Allocate resources and budget for the project.
* Create a detailed project timeline with milestones and deliverables.
* Schedule regular meetings to monitor progress and address roadblocks.

#### 2. Raspberry Pi Pico Setup

* Program the Raspberry Pi Pico using Micro-Python.
* Set up hardware for light control, ensuring accurate GPIO pin configuration.
* Develop and upload firmware to the Pico.
* Validate light control operations using Wokwi simulation.
* Debug and troubleshoot issues in the embedded system during development.

#### 3. Mobile App Development

* Create a design prototype and MVP for the app.
* Develop the mobile app's frontend interface using suitable frameworks.
* Build a backend server to handle app functionality.
* Integrate light control operations into the app’s interface.
* Test and debug the app to ensure seamless functionality.

#### 4. Network Connectivity

* Integrate a Wi-Fi module into the Raspberry Pi Pico.
* Finalize and implement network protocols.
* Establish device-to-server and app-to-device communication.
* Test network performance for reliability and speed.

#### 5. Integration and Testing

* Integrate the mobile app with the Raspberry Pi Pico and the backend system.
* Conduct functional testing for all components and the overall system.
* Test network connectivity.

#### 6. Deployment and Maintenance

* Deploy the final system, including hardware, app, and backend components.
* Monitor the system’s performance post-deployment.
* Address bugs and provide updates if needed.

**Project Approach:**

The project followed a modular approach to integrate hardware, software, and network components effectively. The Raspberry Pi Pico was programmed and tested via Wokwi simulation to control the smart light, while the mobile app was developed with a user-friendly interface for light control and device pairing.

Network connectivity was established for real-time communication between the app, cloud server, and embedded system, ensuring data transmission. After testing to validate functionality and integration, the system was deployed on a VPS for 24/7 availability. Post-deployment monitoring ensures reliability and continuous performance.

**4. Description of the solution**

To estimate the cost of a smart light IoT project, we will break down the total workload into hardware, software, and development time. We’ll also list stakeholders and give an overall project cost estimation in **euros (€)**.

## **1.** **Hardware Costs**

|  |  |  |  |
| --- | --- | --- | --- |
| **Component** | **Quantity** | **Estimated Unit Price (€)** | **Total (€)** |
| Raspberry Pi Pico W (or ESP32) | 1 | €7 - €10 | €10 |
| RGB LED (with controller) | 1 | €5 | €5 |
| Power Supply | 1 | €10 | €10 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Component** | **Quantity** | **Estimated Unit Price (€)** | **Total (€)** |
| PCB Board and Wires | 1 set | €3 | €3 |
| Enclosure (plastic or 3D printed) | 1 | €10 | €10 |

**Total Hardware Cost:** ≈ €38

## **2.** **Software Costs**

If using open-source tools and libraries, the software cost would be minimal. The main costs here relate to development and testing.

* **Development Tools (IDE, libraries)**: €0 (assuming open-source libraries)
* **Mobile App Development (Custom app or Web-based)**: If developing in- house, the cost will include developer time (estimated below).

## **3.** **Development and Labor Costs**

We’ll break this down by role, with the average European rates.

|  |  |  |  |
| --- | --- | --- | --- |
| **Role** | **Hourly Rate (€)** | **Hours** | **Total (€)** |
| Software Developer (IoT/Pico) | €50 | 50 | €2,500 |
| Mobile App Developer | €50 | 60 | €3,000 |
| UI/UX Designer | €40 | 30 | €1,200 |
| Electrical Engineer | €55 | 20 | €1,100 |
| Project Manager | €60 | 20 | €1,200 |
| Testing and QA | €45 | 20 | €900 |

**Total Labor Costs:** ≈ €9,900

## **4.** **Other Costs**

|  |  |
| --- | --- |
| **Other Cost Items** | **Estimated Total (€)** |
| Marketing and User Feedback (if applicable) | €500 |
| Prototyping and Iteration | €500 |

|  |  |
| --- | --- |
| **Other Cost Items** | **Estimated Total (€)** |
| Server hosting (optional) | €100/year |

**Total Other Costs:** ≈ €1,100

|  |  |
| --- | --- |
|  |  |
|  |  |

1. **Total Project Cost Estimated Total:**

* **Hardware**: €38
* **Labor**: €9,900
* **Other Costs**: €1,100

Total Estimated Cost: ≈ **€11,038**

**5. Individual part of each team member**

**Jiaxin Li:**

In our smart light project, I was responsible for developing the mobile application for Android devices. My primary focus was on coding and implementing the user interface and functionality to control the smart light. The application featured two buttons, "On" and "Off," allowing users to toggle the light's status seamlessly.

To achieve this, I ensured the app communicated effectively with the Wi-Fi-connected receiver, integrating with Azure Cloud for polling and updating the light's current status in real-time. My responsibilities included designing the UI for an intuitive user experience, implementing button actions, and establishing the backend communication with the receiver via cloud services. This involved coding the necessary logic to send commands and retrieve status updates, ensuring reliability and responsiveness in the system's performance.

**Oumaima Oubihi:**

For my part, I was responsible for creating a simulation on Wokwi to demonstrate and validate the functionality of the embedded system. My main task was arranging the components in the correct order and writing the Micro-Python code to control the light bulb, ensuring that all features were properly implemented and functioning as intended. To get the desired results, I developed code for the Raspberry Pi Pico to control the light operations as required. I tested and validated the system's functionality using the Wokwi simulation, debugging and troubleshooting the issues I encountered during development to ensure the system operated effectively.

Additionally, I attempted to set up the virtual machine (VM) and Virtual Private Server (VPS) for hosting the backend. While I managed to configure the server and resolve issues like non-responsive ports and firewall settings, the server’s reliability remained inconsistent. After a lot of troubleshooting, and time spent on the task, another team member took over this task and they successfully completed the setup.

Throughout this project, I learned how to work with embedded systems and discovered that even simple tasks can take a significant amount of time due to unexpected bugs.

**Khanh Do:**

**Oskari Laitinen:**

My part of the project was handling the pico board and its wiring to the light(s), and making sure it connected properly to the VM itself. Arguably my part was the simplest in the entire project, since the pico – VM connection has been done a thousand times, so I could just use some premade code and slightly modify it to fit our VM.

Working on the project was useful for learning how the actual pico board works outside of the simulations.

**6. Summary of reflection**

- Looking back on the project and the planning, it was pretty much spot on, the role assignments and proposed deadlines were very well allocated.

- The project was very successful, and we as a team learned a lot about how the pico and VM’s work in practice.

- There were some minor issues towards the end of the project, but nothing that couldn’t be resolved with teamwork.

- As a team we are very happy with the result and the proposed grading.