Project and Seminar

2024/2025 - 2nd Semester

Bachelor in Computer Engineering and Informatics

Remote Lab

Ângelo Azevedo, n.º 50565, e-mail: a50565@alunos.isel.pt
António Alves, n.º 50539, e-mail: a50539@alunos.isel.pt

Advisor: Pedro Matutino, e-mail: pedro.miguens@isel.pt

March 2025

Introduction

The design, development, implementation, and finally, the validation of digital systems require, in addition to simulators, the use of hardware to verify their implementations in real devices. In the current teaching paradigm, in which face-to-face time is reduced and remote and autonomous work is increased, it is necessary to create alternatives to the current model.

The Remote Lab project aims to provide a virtual lab with access to remote hardware. This lab consists of a web application running on an embedded system. The web application, accessed via a website, aims to provide a dashboard where users can join a laboratory. This is where users can control the remote hardware. A hierarchy system will be implemented to provide different roles, each with their own permissions relative to how users can browse the information provided by the web application.

Requirements

Database

A database will be used to store the information of the web application. This includes the user's information, the laboratories' information and the hardware's information.

- User information storage
- Laboratory configurations and schedules
- Hardware specifications and status

Web API

The web API will be a RESTful API that will be used to communicate with the web application. It will be responsible for the communication with the database and the hardware.

- Authentication and authorization endpoints
- Laboratory management endpoints
- Real-time data communication

Authorization

The authorization will be implemented with a RBAC (Role-Based Access Control) system. This will allow the user to have different roles with different permissions:

- Student
  - Enter a laboratory when the professor allows it
  - Configure, view and control the hardware on the laboratory
  - Schedule laboratory sessions through the calendar
- Professor
  - Create, read, update and delete (CRUD) laboratories
  - Invite students to join the platform through a unique code
  - Invite students to join a laboratory

  - See the laboratories' schedules and usage
  - Create, read, update and delete (CRUD) hardware on the laboratory
- Administrator
  - Has all the other roles' permissions
  - Create, read, update and delete (CRUD) users

Authentication API

The authentication API will be a separate RESTful API that will be used to authenticate the user.

Web Application

The web application will have a dashboard where the user can join a laboratory. This is where the user can control the remote hardware.

- User-friendly interface
- Real-time hardware monitoring
- Session management

Hardware Integration

There will be a hardware integration module that will be responsible for the integration of the hardware into the web application.

System Architecture

The system architecture consists of multiple interconnected components that work together to provide the remote laboratory functionality.

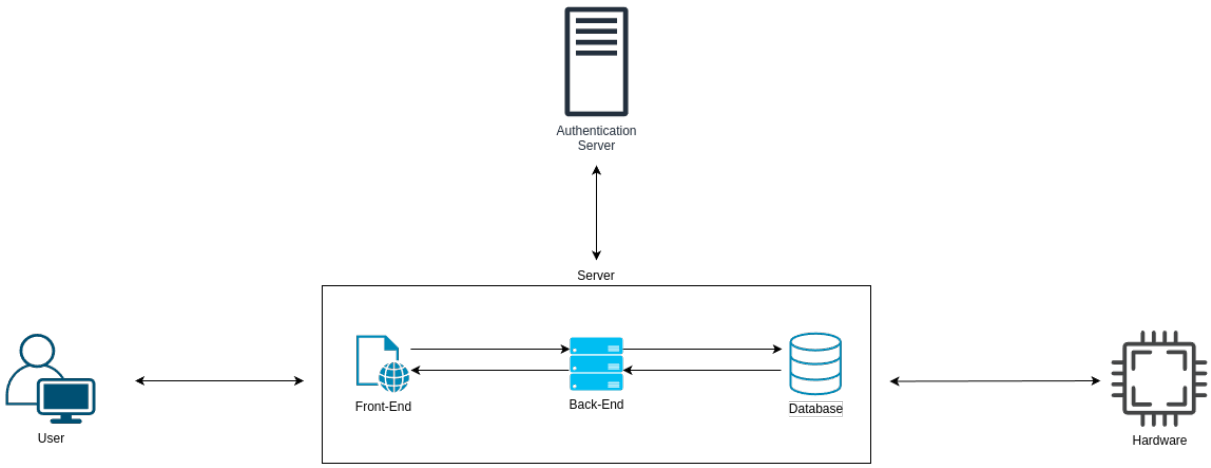

Figure 1: System Architecture Overview

Timeline

The project will be developed following the timeline shown in Figure 2, which outlines the main phases and milestones of the development process.

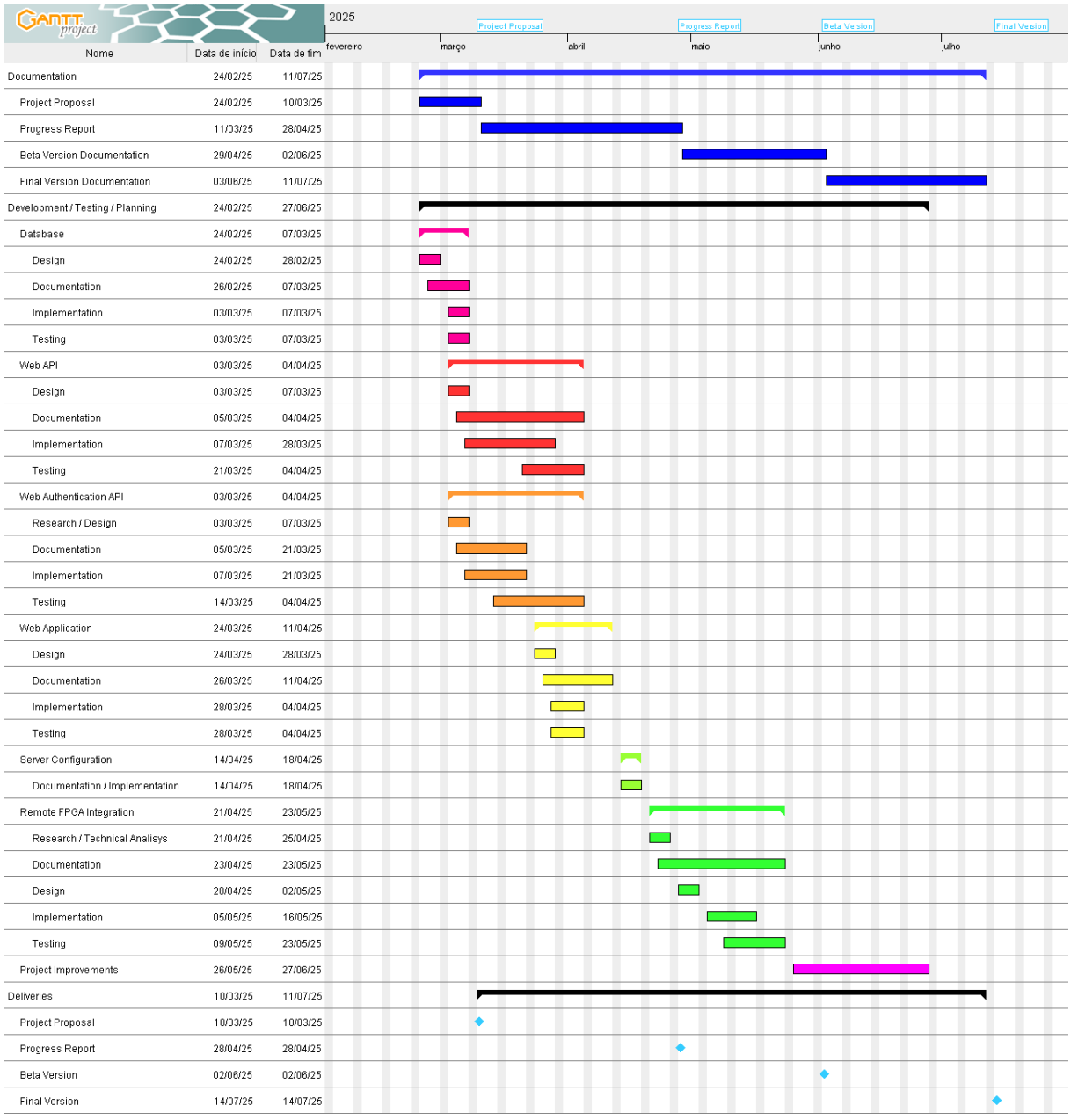

Figure 2: Project Timeline and Milestones

1 User Journey

This section aims to explain the possible actions a user can take in the system.

1.1 General

This section provides general user journeys, that is journeys that share the same characteristics when the user is a Student, a Professor or an Administrator.

1.1.1 General Login

1. The user goes to the login page.
2. The user enters their email and password.
3. The user logs in.

1.1.2 General Account Information

1. The user goes to the account information box.
2. The user can check his information like:
    - Username.
    - Email.
    - Student number if it is Student. If it has the role Professor/Administrator, this field is disabled.
    - Account creation date.
3. The user can change his username (Assigned automatically when the account was created).
4. The user can change his password.

1.2 Student User Journey

This section provides the student journey.

1.2.1 Student Account Creation

1. The user receives an invitation code sent by the Professor.
2. The user goes to the account creation page.
3. The user inserts his email, student number and password.
4. The user creates an account.

1.2.2 Student Dashboard

1. The student goes to the dashboard that is the main page of the application.
2. This page contains:
    - The student's laboratories.
    - The account box on the top of the page.
    - The role that the user is logged in. This only shows if the user has other roles assigned.

3. The user can click on a laboratory to view its content.
4. The user can click on the account box to see his account information or log out.
5. If the user has other roles assigned, he can click on the role to login has other role, like Professor.

1.3 Student Laboratory

1. The student goes to the laboratory.
2. The student sees the laboratory.

1.4 Professor User Journey

This section provides the teacher journey.

1.4.1 Professor Account Creation

1. The user receives an invitation code sent by a Professor.
2. The user goes to the account creation page.
3. The user inserts his email and password.
4. The user creates an account.

1.4.2 Professor Dashboard

1. The professor goes to the dashboard that is the main page of the application.
2. This page contains:
    - The professor's laboratories.
    - A button to create a laboratory.
    - The account box on the top of the page.
    - The role that the user is logged in.
3. The user can click on a laboratory to view its content.
4. The user can click on the account box to see his account information, create an invitation code or log out.
5. If the user has other roles assigned, he can click on the role to login has other role, like Student.

1.5 Professor Laboratory Creation

1. The professor goes to the laboratory creation page.
2. In order to create a laboratory the professor needs to insert:
    - The laboratory name.
    - The laboratory duration. This duration is the time a student has in each session.
3. The teacher creates a laboratory.

1.6 Professor Laboratory

1. The professor goes to the laboratory.
2. The professor sees the laboratory and can delete it.

1.7 Administrator User Journey

This section provides the administrator journey.

1.7.1 Administrator Dashboard

1. The administrator goes to the dashboard.
2. The administrator sees the dashboard.

1.7.2 Administrator User Management

1. The administrator goes to the user management page.
2. The administrator sees the user management page.

Domain Restrictions

The **Invite Code** has the following restrictions:

- The invitation code has 8 bytes of random characters (includes special characters);
- Invites's Time to Live (TTL) is 7 days;
- The number of usages an invitation can be used is 1 time.

The **Token** has the following restrictions:

- The token validation size is 32 bytes of random characters;
- The token Time to Live (TTL) is 24 hours and the Rolling Time to Live (TTL) is 1 hour.
- The number of tokens per user is 3.

The **Group** has the following restrictions:

- The group name needs to be between 1 and 100 characters of length.
- The group description can have 2048 characters of length.

The **Laboratory** has the following restrictions:

- The laboratory name needs to be between 1 and 100 characters of length.

The **Lab Session** has the following restrictions:

- The session state can be active, inactive or scheduled.

The **Hardware** has the following restrictions:

- The hardware name needs to be between 1 and 100 characters of length.
- The hardware status can be available, occupied or in maintenance.

The **User's** student number must be only numbers.