Frontend Setup Guide

LEO-Based Assessment Tool – Frontend

---

1. Purpose

This document describes how to **set up, configure, and run the frontend application** of the *LEO-Based Assessment Tool*.

The frontend is an **Electron-based desktop application** built with **React, Vite, and TypeScript**. It provides the user interface for teachers and students and communicates with the Spring Boot backend via REST APIs.

End users interact **only with the frontend**; all business logic and data persistence are handled by the backend.

---

2. Prerequisites

Required Software

To run the frontend application, the following tools are required:

- Git
- Node.js (LTS version recommended)
- npm (comes with Node.js)
- Docker
- Docker Compose

Optional (for local development)

- Visual Studio Code (recommended IDE)
- Java 17 (only required if backend is started locally)

---

3. Repository Setup

Clone the frontend repository:

```
git clone https://github.com/piy678/SENGPRJ_Group6_FrontendPart
cd SENGPRJ_Group6_FrontendPart
```

---

4. Configuration

4.1 Environment Variables

The frontend communicates with the backend via HTTP. The backend base URL is configured using environment variables.

Example (Vite environment configuration):

```
VITE_API=http://localhost:8080
```

For production deployment, the API URL points to the backend service running on AWS.

4.2 CORS Configuration

CORS is configured on the backend to allow requests from the frontend origin (e.g., Electron or Vite dev server). No additional CORS configuration is required on the frontend side.

---

5. Running the Frontend with Docker (Production / Server Setup)

The frontend can be started together with the backend using **Docker Compose**.

From the frontend project directory:

```
docker compose up -d
```

This command: - Builds the frontend Docker image - Starts the Electron-based frontend service - Connects automatically to the backend service

After startup, the application is accessible at:

```
http://13.53.169.202:5174
```

---

6. Local Development Setup (Recommended for Development)

6.1 Install Dependencies

Install all required Node.js dependencies:

```
npm install
npm install concurrently --save-dev
```

6.2 Start Backend Services

The frontend depends on the backend API. Start the backend (and database) using Docker Compose:

```
docker compose up -d
```

Ensure the backend is reachable before starting the frontend.

6.3 Run the Vite Development Server

Start the frontend development server:

```
npm run dev
```

This starts the Vite server and enables hot module reloading for rapid development.

Default dev URL:

```
http://localhost:5173
```

6.4 Start the Electron Application

To start the Electron desktop application and load the Vite frontend inside it:

```
npm run start
```

This command: - Launches an Electron window - Loads the Vite development server - Provides a desktop-like experience on Windows, macOS, and Linux

---

7. Application Structure Overview

The frontend follows a modular and component-based structure:

```
app/
├── electron/              # Electron main process
├── public/                # Static assets
├── src/
│   ├── api/               # Backend communication (HTTP, error handling)
│   ├── components/        # Reusable UI components
│   ├── pages/             # Main application pages
│   │   ├── teacher/       # Teacher dashboard and tabs
│   │   ├── student/       # Student dashboard
│   │   ├── Login.jsx
│   │   └── Landing.jsx
│   ├── App.jsx            # Root React component
│   ├── main.jsx           # Application entry point
│   └── mockData.js        # Mock data for development
├── docker-compose.yml     # Docker configuration
├── Dockerfile             # Frontend Docker image
├── package.json
├── vite.config.js
└── README.md
```

---

8. Authentication & Backend Integration

- Communication with the backend uses **JSON over HTTP**
- Role-based access control is enforced by the backend
- The frontend adapts views based on the authenticated user role (Teacher / Student)
- Authentication tokens are stored locally and attached to API requests

---

9. Common Issues & Troubleshooting

Frontend Cannot Reach Backend

- Ensure the backend is running
- Check the `VITE_API` configuration
- Verify backend port and URL

CORS Errors

- Ensure the frontend origin is listed in `CORS_ALLOWED_ORIGINS` on the backend
- Restart backend after configuration changes

Electron App Does Not Start

- Verify that the Vite dev server is running
- Ensure Node.js version compatibility
- Check console output for Electron errors

---

10. Related Repositories

- Frontend Repository: https://github.com/piy678/SENGPRJ_Group6_FrontendPart
- Backend Repository: https://github.com/piy678/SENGPRJ_Group6

---

11. Summary

This setup guide provides all necessary steps to run the frontend application in both **production** and **local development** environments.

The Electron-based frontend offers an intuitive user interface for teachers and students and seamlessly integrates with the backend services of the LEO-Based Assessment Tool.

Group 6 — SENGPRJ
Supervisor: *Thomas Mandl*