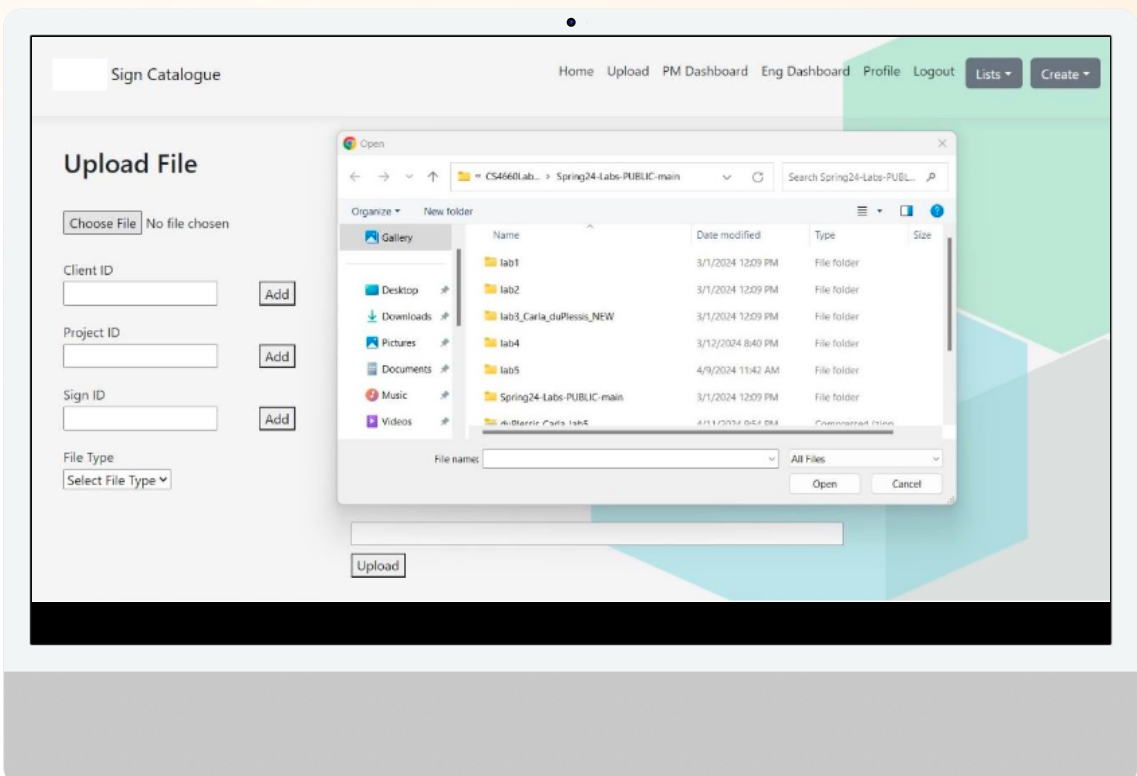


Web Application UI Design Documentation

Client user project to design a sign and file browser



Overview

This case study details a web application project in which I served as the lead UX/UI designer. The application was designed to replace a legacy file upload and storage system used by a team of design engineers and project managers. The redesign centered on enhancing the system's usability, enabling efficient file sharing, and establishing a scalable and intuitive platform. The project aimed to bridge the gap between engineering and project management teams by improving daily workflows around document handling.

This case study will show a detailed project design overview including the user interface design.

Role

Lead UX/UI designer

Tools

ReactJS, HTML, CSS, Visual Studio Code, Azure

Timeline

August 2023 - May 2024

Problem

The existing file management platform was outdated, making file discovery and collaboration between engineering and project management teams inefficient. Users struggled with organizing and retrieving design documentation, which slowed down project communication and delivery. The previous interface lacked clarity, accessibility, and scalability, hindering effective collaboration across departments.

Solution

The team's solution was to develop a secure, in-house web application accessible only to employees. The platform features dedicated dashboards for different user roles, allowing engineers and project managers to perform role-specific tasks such as uploading, downloading, and managing access to design files.

Project goals

- Enhance search functionality for design files.
- Create a scalable, modular backend architecture.
- Ensure intuitive UI for frequent, daily interactions.
- Enable secure file handling and access control.
- Align the application with the company's internal infrastructure and operational needs.

1

System Architecture

2

Data Storage Design

3

Component Design

4

UI Design

System Architecture

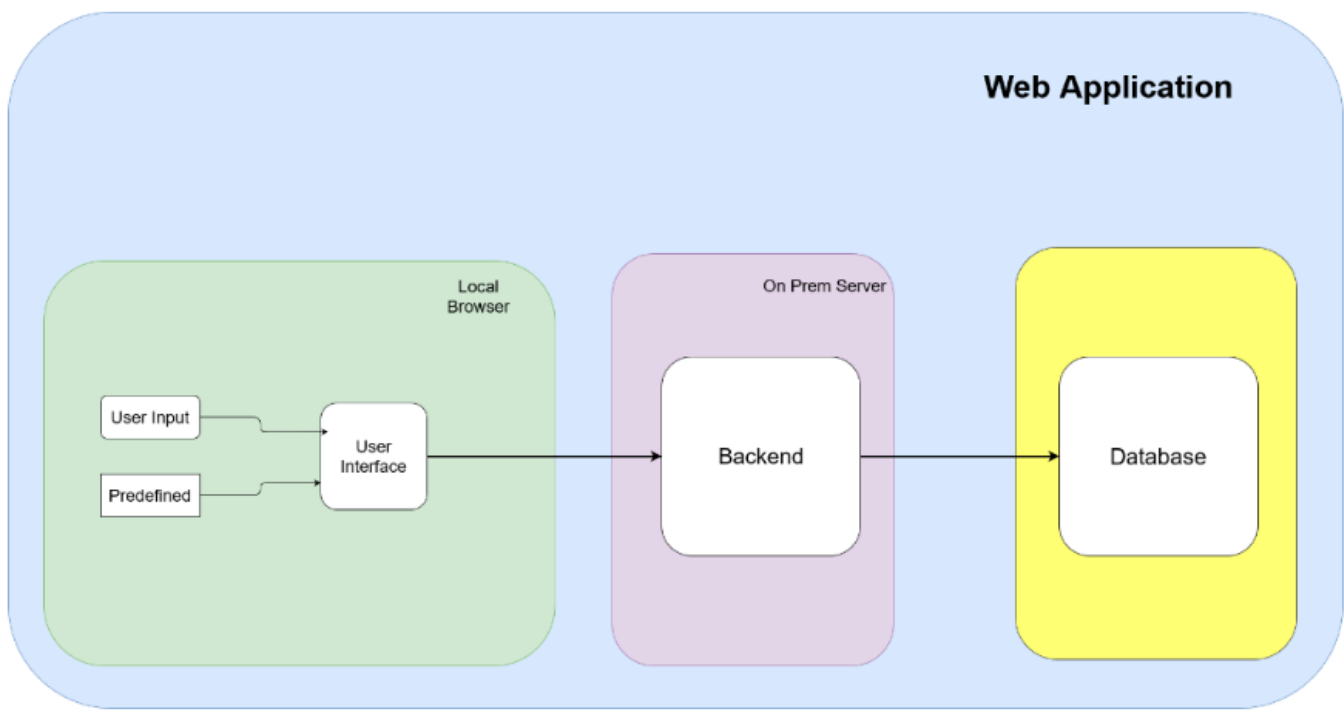
The system was built as a secure, web-based application hosted on internal servers. It separates responsibilities across three main components:

- Front End:** Developed with React.js, responsible for user interactions and visual presentation.
- Back End:** Built using .NET/C#, managing authentication, business logic, and server-side operations.
- Database:** Utilized Microsoft SQL Server for robust and scalable data storage.

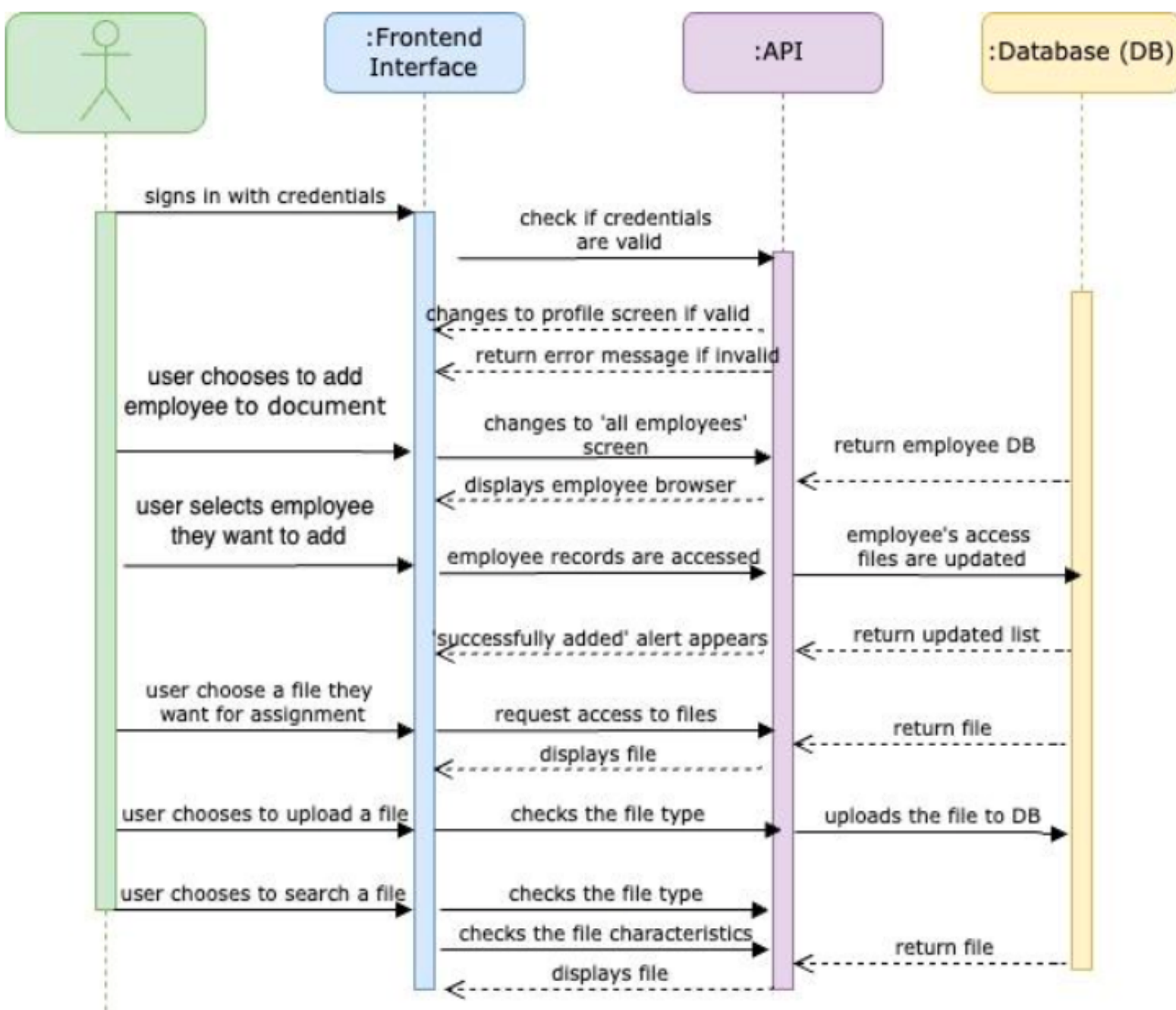
The architecture supports scalability and maintainability while ensuring that role-based access can be granted through company email credentials.

Static vs. Dynamic Views

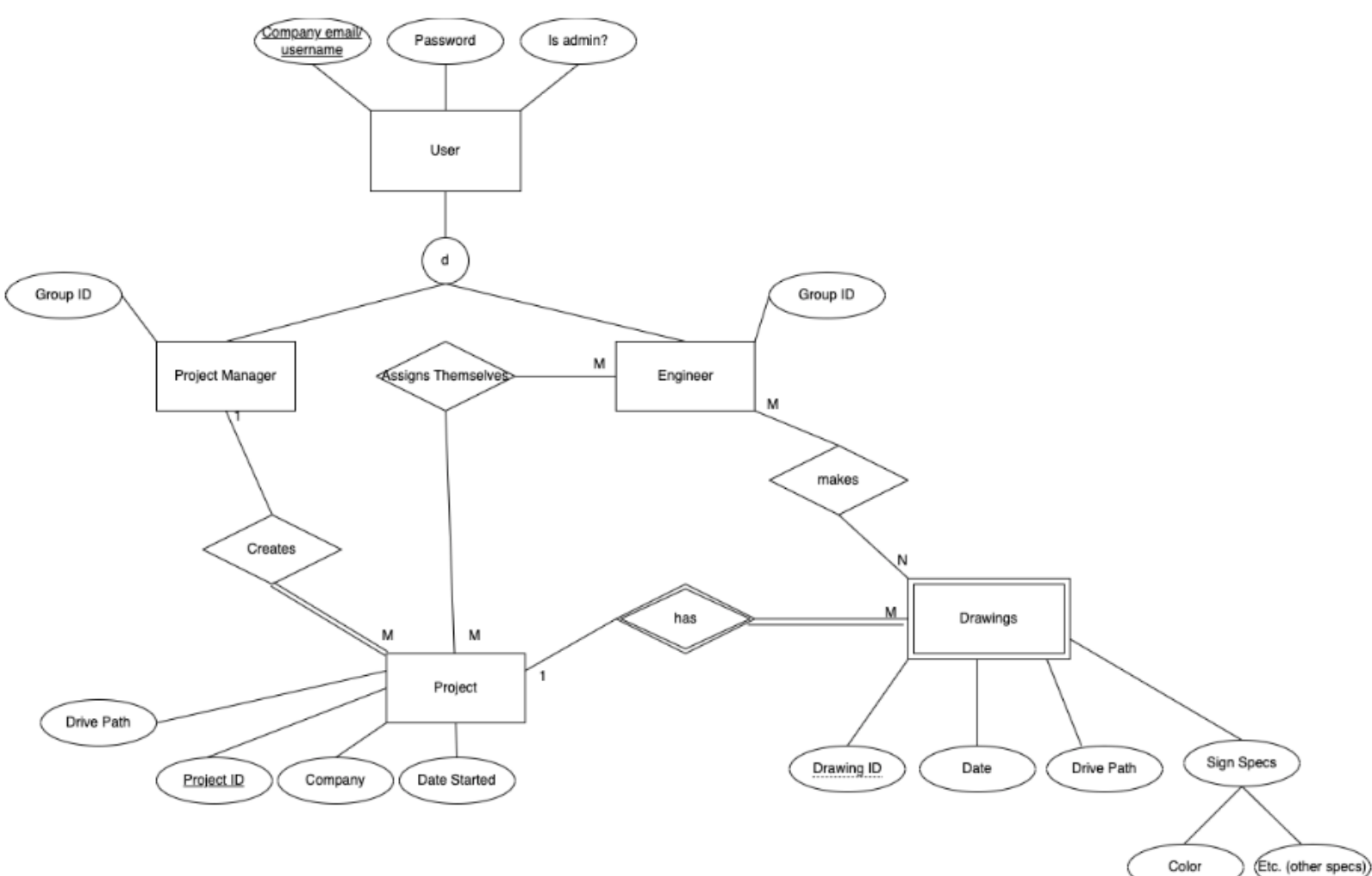
- Static View:** Highlights the interaction between the front end, back end, and database with a clear separation of concerns to simplify development and future updates.
- Dynamic View:** Demonstrates user actions such as login, file upload, and document assignment. These flows follow a System Sequence Diagram to map out user interaction and system responses step-by-step.



Static System Diagram



Dynamic System SSD



Entity Relationship Diagram

UI Design

The user interface was designed to promote ease of use and reduce cognitive load. Core functions include Search, Upload, and Download. The design emphasized efficiency for daily use, enabling team members to locate and share files with minimal friction. Accessibility and clarity were prioritized throughout the design process, ensuring the application aligned with user expectations and operational standards.

Data Storage Design

Data handling strategies focused on ensuring both accessibility and integrity. The database structure supports dynamic queries, making the search feature fast and responsive, while maintaining data security and proper indexing.

Component Detailed Design

All design segments were unified into a comprehensive system, capturing how individual components—architecture, UI, and database—interact to support user goals. This holistic view ensured that each part of the system contributed effectively to the broader project objectives.