Nikita Makurin
Kirill Tolkunov
Yehor Tereshchenko
Miikka Lahtinen
Darijus Seporaitis

Project documentation

Raksa AI challenge

Metropolia University of Applied Sciences, Aalto university

Construction Information technology

November 2024

Contents

1 Teamwork and collaboration

1.1 Team overview

Our project team consisted of members with diverse expertise and educational backgrounds, which contributed significantly to its success.

- **Nikita**: HVAC and civil engineer currently pursuing a first-year degree in Construction Information Technology.
- **Kirill**: An HVAC engineer and fellow first-year Construction Information Technology student.
- **Mikka**: A second-year Civil Engineering student with professional experience as a project manager.
- **Yehor**: A first year IT student with a great experience.
- **Darijus**: A master's degree IT student with advanced technical expertise.

This combination of technical, engineering, and management skills fostered a collaborative environment that balanced innovation with practicality.

1.2 Teamwork Lessons Learned

During the project, the team gained valuable insights into effective teamwork. A key takeaway was the importance of **leveraging diverse expertise.** Coders have brought technical knowledge to software development, while design engineers have brought important knowledge about the practical implementation of the project.

To ensure smooth collaboration:

- **Weekly Meetings:** Regular meetings provided a platform to align goals and address challenges.

- **Always be in touch:** We organized a discord server where you could send a message to one of us or share an idea

- **Task Delegation:** Clearly defining roles prevented overlap and confusion.

Challenges:

- Different levels of knowledge in coding, it was difficult for most of our team to understand how the core of our project works and a lot of time was devoted to writing documentation and instructions for it

1.3 Information Retrieval

The primary source of information for the project was open resources, including modern AI tools like ChatGPT. These tools were used to explore technical solutions and clarify complex questions, enabling quick access to answers and fostering innovative ideas.

For project file management and collaboration, we relied on GitHub. This repository provided all team members with convenient access to code, documentation, and other critical materials. Using GitHub significantly streamlined synchronization and version control processes.

Lessons Learned:

- Utilizing AI expedited information retrieval but required careful validation to ensure data accuracy.

- A shared GitHub repository proved to be an effective tool for collaboration and simplifying communication within the team.

1.4 Team Collaboration Evaluation

The collaboration between team members was overall **highly effective**, resulting in the timely completion of the project.

- **Coordination:** The blend of structured planning and adaptive problem-solving strengthened team dynamics.

- **Conflict Resolution:** Disagreements were resolved quickly through open communication and by prioritizing project objectives.

Outcomes:

- Coders and engineers not only achieved the project's technical goals but also developed a deeper appreciation for each other's expertise.

- As a team, we have improved our ability to work in interdisciplinary settings, which will benefit future collaborations.

2 Continuity and Future Development

The project has been designed with ease of continuation and scalability in mind:

- **Clear and Understandable Code:** The codebase is written with clarity and readability as priorities, making it straightforward for future developers to understand and build upon.

- **Comprehensive Documentation:** Step-by-step instructions detailing the code structure and the core system of the project have been provided.

This ensures that anyone reviewing the work can quickly grasp its purpose and functionality.

- **Flexibility in Goals:** As the project is currently in **Phase 1 of development**, its foundation is flexible enough to allow reorientation or adjustments to the main objectives if needed.

These factors collectively ensure that the work done can be seamlessly leveraged for future development.

2.1 Technical Approach and Implementation

Our project employs modern tools and technologies to ensure functionality and ease of future development.

- **Backend and Databases:**
  Python was chosen for backend development due to its versatility and extensive libraries for AI and data processing. Django or Flask frameworks provide robust foundations. PostgreSQL or MongoDB serve as databases for structured and unstructured data.

- **Frontend Development:**
  The user interface is built using React.js or Vue.js for interactivity. Three.js is employed to enable immersive 360-degree virtual tours.

- **AI Integration:**
  AI models, developed using TensorFlow or PyTorch, support recommendation systems and natural language processing for summarizing research and answering queries.

- **Cloud Infrastructure:**
  The project is hosted on platforms like AWS or GCP, with storage

solutions (e.g., S3) for large datasets and media files. AI models are deployed using Docker or cloud-native tools like SageMaker.

- **Authentication and APIs:**
  Secure authentication (OAuth 2.0/JWT) and RESTful APIs enable smooth user interactions and data access.

2.2 Implementation Highlights

1. Defined user journeys focused on key features like login, browsing, and virtual tours.

2. Structured databases for storing equipment details and user preferences.

3. Developed and tested AI models and integrated them with the backend.

4. Built a responsive frontend and connected it to the backend via APIs.

5. Conducted extensive testing before deployment to ensure performance and usability.

This approach ensures a scalable and flexible foundation for future development.