# Project Description

**Customer:** Karlsruhe University of Applied Sciences, Faculty of Computer Science and Business Information Systems (IWI)

**Industry:** Education

**Link:**
https://www.h-ka.de/die-hochschule-karlsruhe/fakultaeten/informatik-und-wirtschaftsinformatik
https://www.h-ka.de/en/about-hka/faculties/computer-science-and-business-information-systems/overview

**Title of the project topic:** CCP Portal Pro

**Short description of the task**

The Business Informatics Faculty at the Karlsruhe University of Applied Sciences has long employed the CCP Portal as a collaborative platform within the "**C**ustomer **C**ollaboration **P**roject" (aka. “Anwendungsprojekt”) course. This platform has been instrumental in enabling supervisors and project teams to efficiently manage their work and status reports, conduct peer reviews, and share files. Moreover, professors have utilized it to oversee semesters, teams, and projects while documenting notes on group and individual performances.

However, relying on modern technologies and prioritizing functionality alone has led to the development of a legacy system that lacks user satisfaction and poses challenges in terms of maintenance and updates. Consequently, the faculty has entrusted us with the pivotal task of devising a tailored maintainability strategy that precisely aligns with their objectives and general conditions, addressing the aforementioned issues head-on.

The "CCP Portal Pro" project represents a comprehensive initiative aimed at refining and advancing a client-provided application to meet contemporary standards and elevate user experience. Beginning with the establishment of a Minimum Viable Product (MVP), the project emphasizes deploying the application across various environments to facilitate real-world testing and early user feedback. Subsequent phases focus on improving software quality, upgrading security measures, and implementing a modernized DevOps pipeline strategy for continuous enhancement. The overarching goal is to create a robust, user-friendly, and easily deployable application, fostering adaptability and scalability throughout the development lifecycle. Confidentiality and client flexibility play crucial roles in shaping the project's trajectory, ensuring a tailored approach to the unique needs and objectives of the project endeavor.

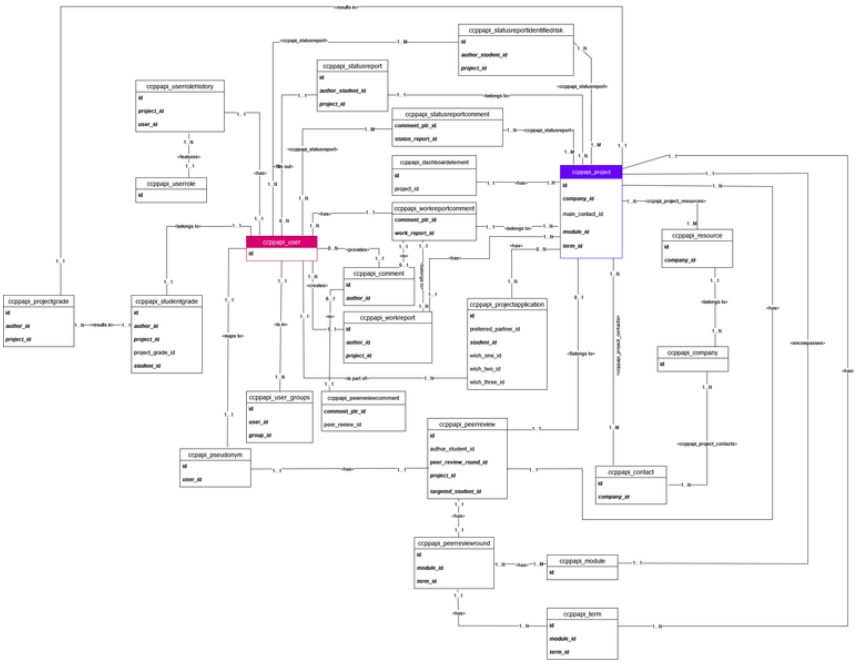
**Short description of the result or highlights of the delivered product**

We delivered a maintainable and high quality software that provides modern tools that simplifies the development and testing process, as well as assures the integrity and confidentiality of sensible data. We can divide our achievements into 3 main topics:

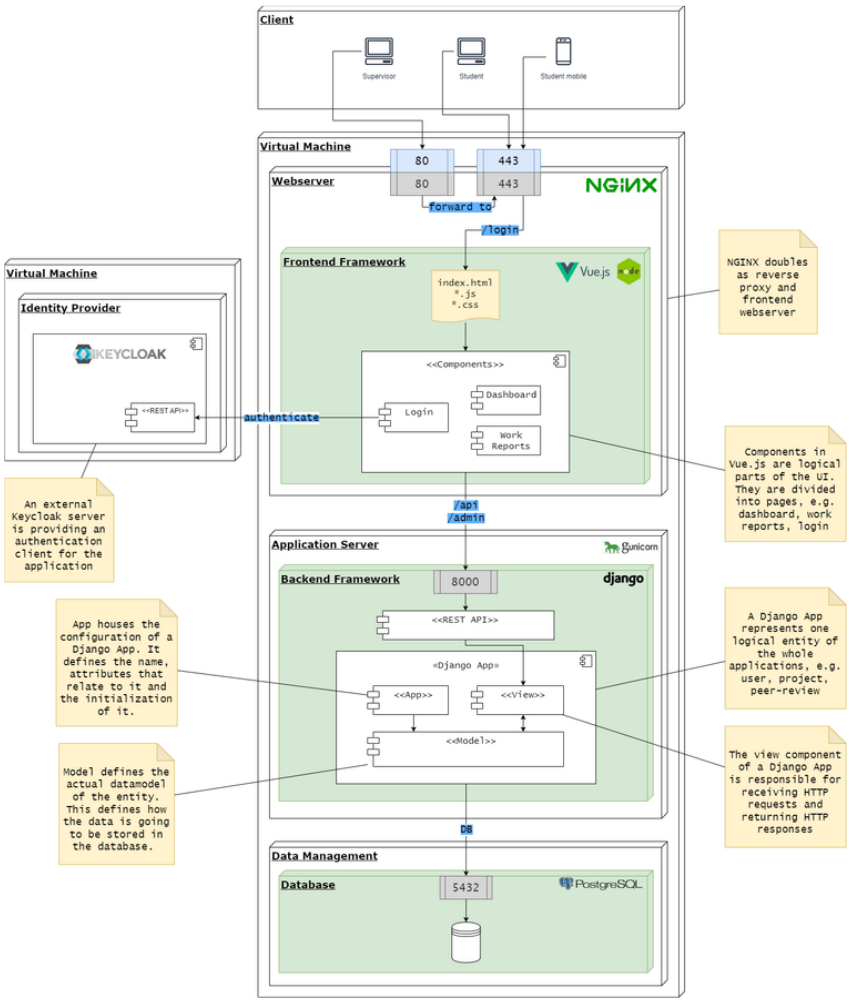
- *DevOps Pipeline:* we managed to automate the process of verifying the changes made to the code, having 4 different stages that assures the correct and efficient functionality of the application, and these are:
  - test-frontend: automated frontend unit and style tests.
  - build-server: database migrations and checks with django migrations.
  - test-server: automated backend unit and style tests.
  - check-dependencies: automated dependencies check and updates.
  - create-docker-image: creations of frontend and backend docker images.
  - deployment: deployment of frontend, backend and database containers.
- *Security:* we managed to include the necessary measurements to meet high level standards for protecting and accessing information throughout the portal. The security concept that was developed includes the following aspects:
  - Operational user on VM: limited privileges to avoid security breaches with unnecessary permissions.
  - Configuration of VM firewall: the ports that are needed are the only ones opened.
  - Configuration of NGINX as Reverse Proxy: established handling for incoming traffic in a production environment to accept connections over HTTPS and ensure that all HTTP traffic is redirected to HTTPS, so all data transmitted between client and server is encrypted.
  - Keycloak documentation: in this case, the recommendations and general steps for libraries are described by an investigation results to facilitate the authentication process.
- *Features:* finally we implemented some new functionalities to improve some workflows, the two prioritized ones with the customer are:
  - *Import / Export:* the capability of bulking and retrieving relevant data for administrators is enabled in backend, so they can choose and insert the information specific information important for them via django admin. This is made for some models, with specific filters and without UI access, because this was in accordance with the customer.
  - *Peer Review Commenting:* this was made in both frontend and backend, so there is the possibility that supervisors create, edit, delete or view comments related with peer reviews. But the implementation of this feature is partial, since there is no filtering to just get the comments of the peer review round of the student you are viewing.

**Members of the project team:** Alexander Seitz, Eric Weres, Erik Dwornik, Ioannis Theodosiadis, Jan Riesterer, Jose Luis Madrigal Sanchez
**Semester in which the project ran:** WS 23/24

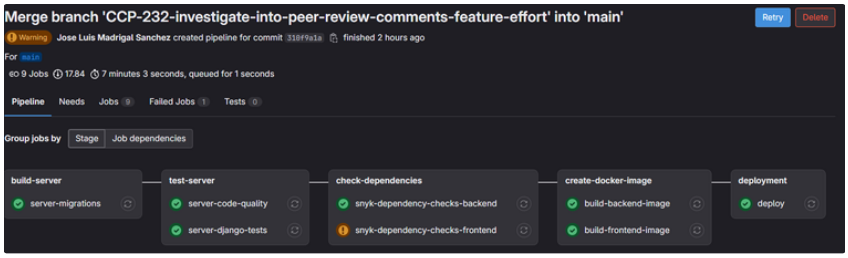
## Entity Relationship Diagram



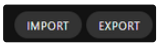
## Architecture Diagram

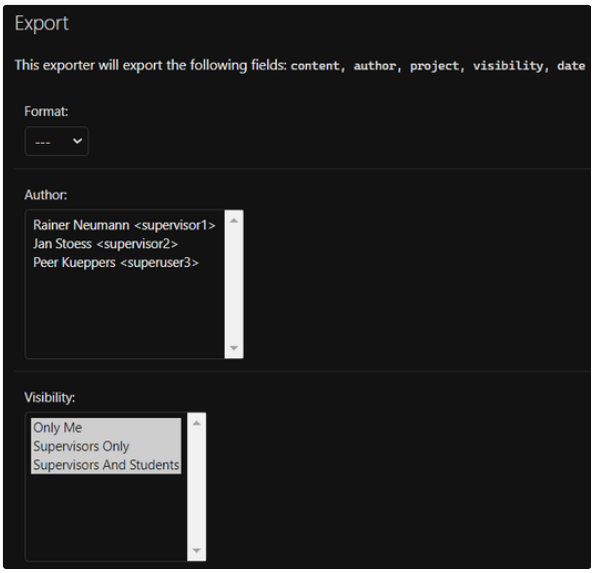


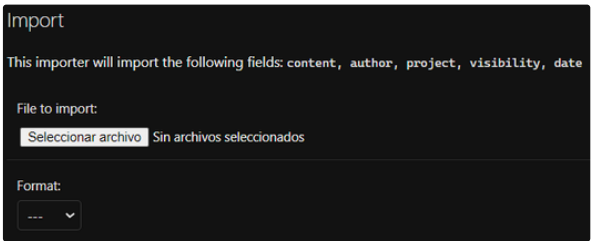

## Pipeline Stages Test with a Feature Branch Merge




### Export / Import Feature




### Peer Review Commenting Feature