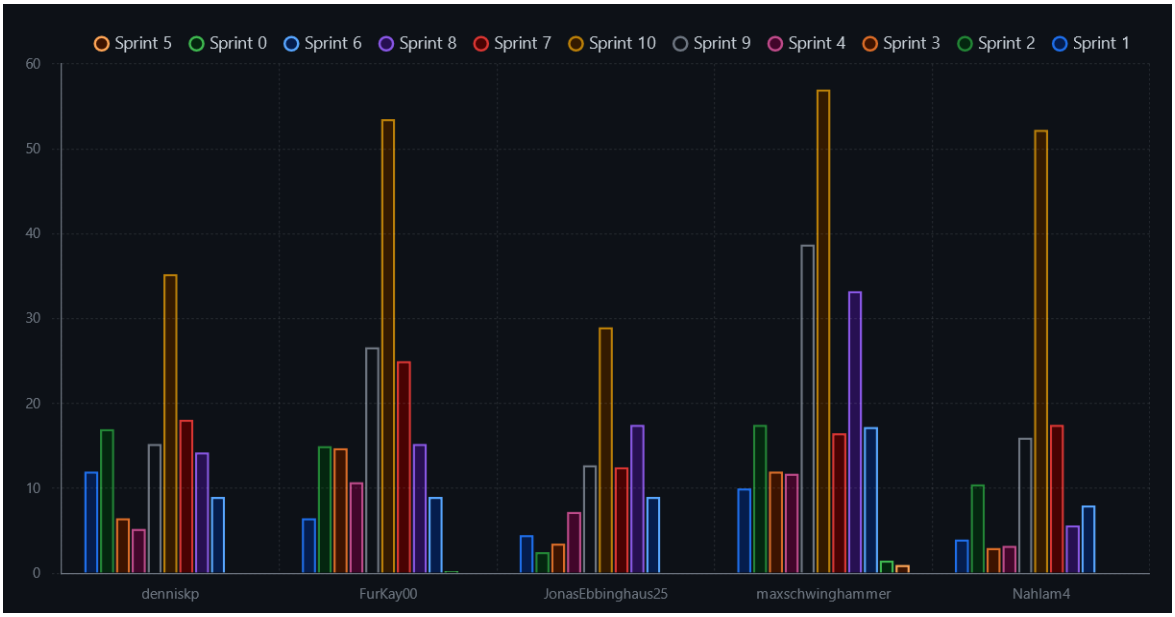
Lecture: Software Engineering
Lecturer: Prof. Nuo Li

Date: 12.06.2024

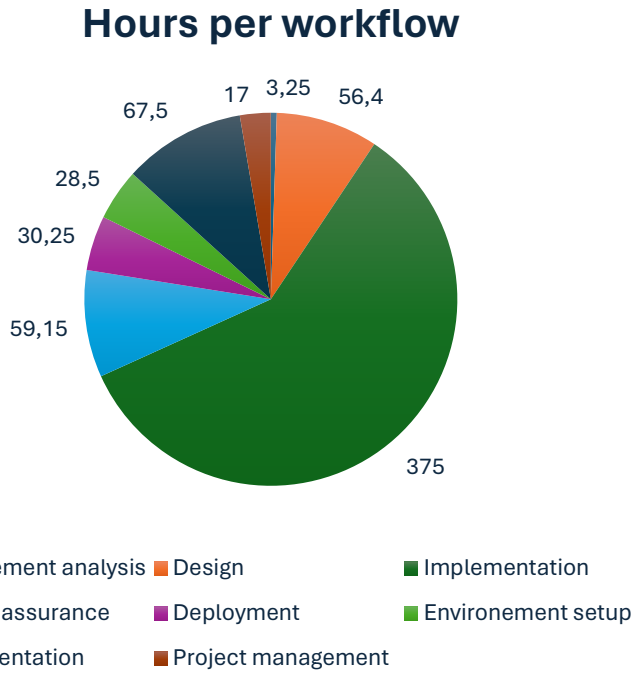
Final Handout – DHBWhub

Statistics Of Effort

1. Hours per person in each sprint



2. Hours per workflow



Team DHBWhub
– Dennis Kopp, Furkan Kay, Jonas Ebbinghaus, Josha Schmitt, Max Schwinghammer

3. Major contributions per Person and hours in total

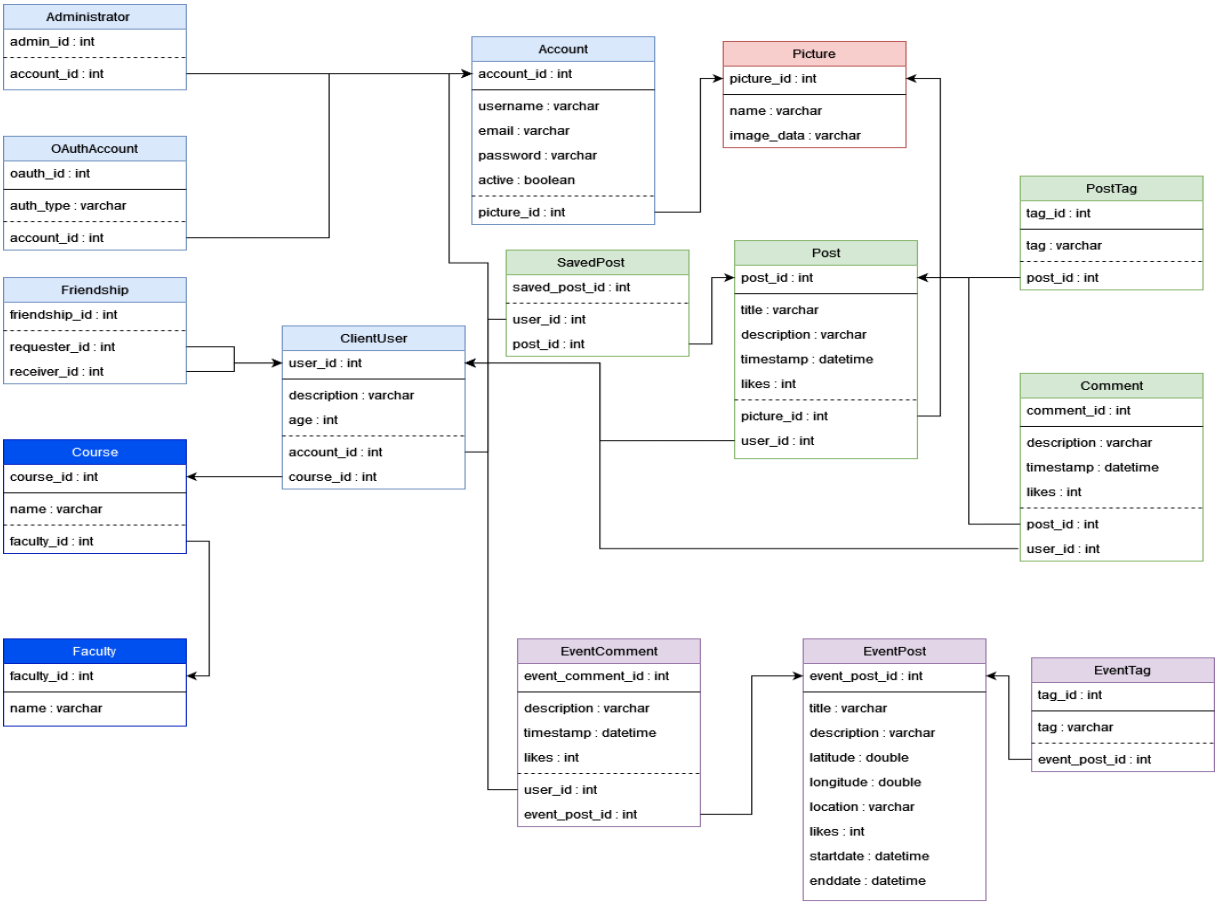
Member	Hours in total	Major contributions
Dennis	133	• Server maintenance • Backend implementation and testing • Bug fixing and utilities • CICD-Setup
Furkan	177	• Backend implementation and testing • Database design and maintenance • Documentation
Jonas	100	• Mock-up designs • Frontend implementation and testing • Authorization/Sign-up/Sign-in management
Josha	120	• Mock-up designs • Frontend implementation • Postman configuration
Max	217	• Mock-up designs • Frontend implementation deployment and testing • CICD-Setup

Demo Highlights

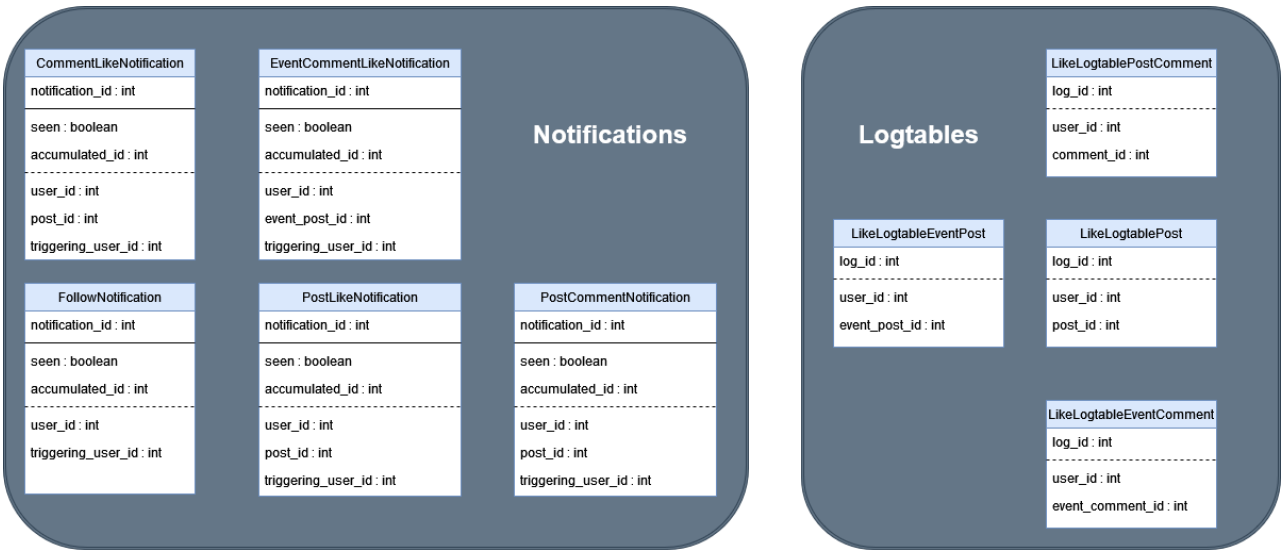
- Account creation and personalization!
- Create posts and comments!
- Notifications!
- View events in the calendar and in the thread format!
- Search for user and posts!
- Follow other users!
- Lots of shenanigans!

Database Design

1) Main relations in the database, mostly representing the identified entities in DHBWhub. First attribute is always the primary key; attributes below the dotted line are foreign keys.



2) Extra relations, simplifying more complex problems such as retrieving notifications or persisting user activity such as liked components. Foreign keys refer to the relations in the image above.



Lecture: Software Engineering
Lecturer: Prof. Nuo Li

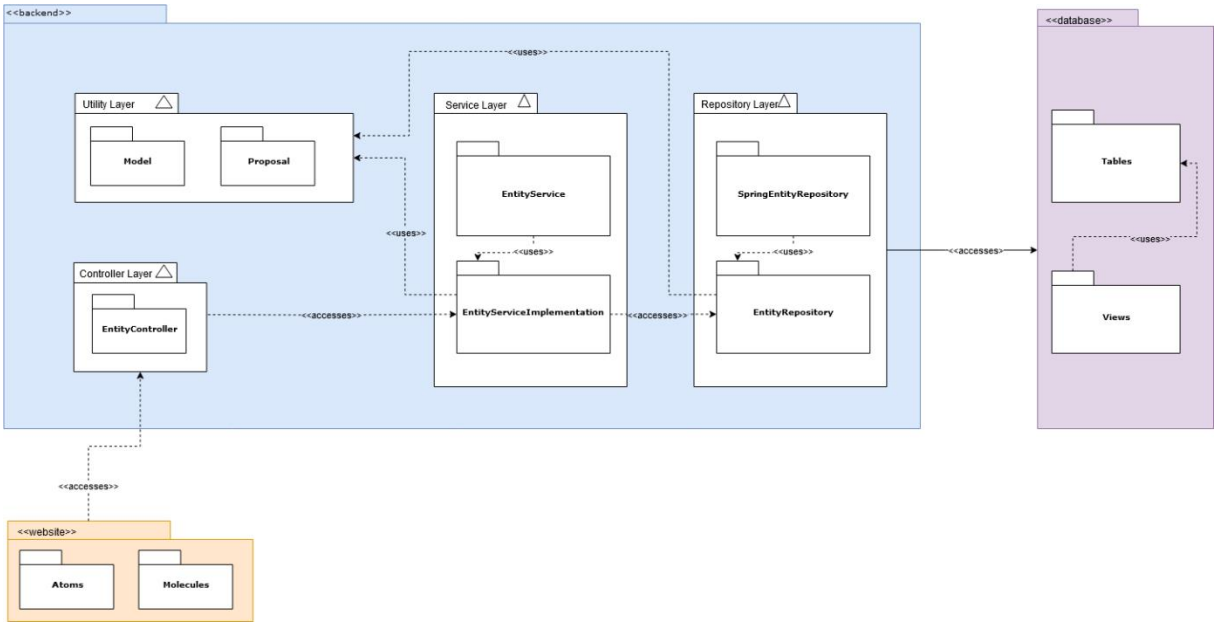
Date: 12.06.2024

Architecture Design and Decisions

Goals:

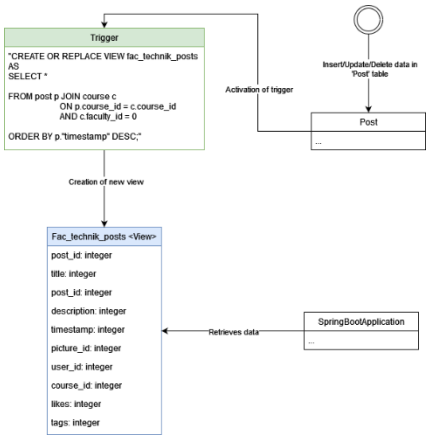
- Usability:
  - Our application is designed in such a way that the user intuitively knows what to do, where to navigate and what to expect after each action. Visual feedback also plays a huge role in this to notify or visualize the user, that an action functioned just as intended or has failed.
- Performance:
  - Since we are dealing with many data retrievals in such a short time, we put a lot of focus in minimizing not only the amount of data queries but also the amount of data transferred in each request.
- Modifiability:
  - The developers should spend less time developing huge complex constructs when creating new functions. Instead, they should implement smaller components that can be built together to bigger structures. This reduces the time to develop and maintain on the long term.

Decisions:



- REST-architecture and MVC:
  - Application divided in website, backend and database.
  - Backend provides specific endpoints for specific occasions in the controller layer.
  - Service layer transforms the needed data with the help of the utility layer.
  - Repository layer retrieves the data from the database.
- Component-architecture:
  - The website is split into several groups such as atoms, molecules, organisms and services.
  - Molecules are composed of atoms and organisms are composed of molecules => Establishment of modularity.

- External views and triggers in the database:
  - Used to shorten calculations of not user-specific data such as the number of comments of each post or the amount of followers of each user.
  - Views are created or update, when the corresponding relation is updated. Thus, the repository doesn't need to calculate the data but retrieves the already available data from the views.



Tech Stack

We used the following applications, platforms and frameworks to create DHBWhub:

- React
- Figma
- TypeScript
- Jest
- GitHub
- Postman
- Java
- JUnit5
- Postgre SQL
- Spring Boot
- Maven

CI/CD

- GitHub Actions triggers four workflows for every push or pull request:
  - CodeQL analysis for Java and Typescript.
  - Maven build (also runs all the tests).
  - React CI pipeline.
- Every developer used the following software or framework on their local environment:
  - Maven for automated testing.
  - React testing library and Jest for frontend testing.
  - JUnit5 and Mockito for backend tests.
  - Sonarlint/Eslint for static code analysis.