Tutorial for Advanced Software Engineering

Prof. Chunyang Chen

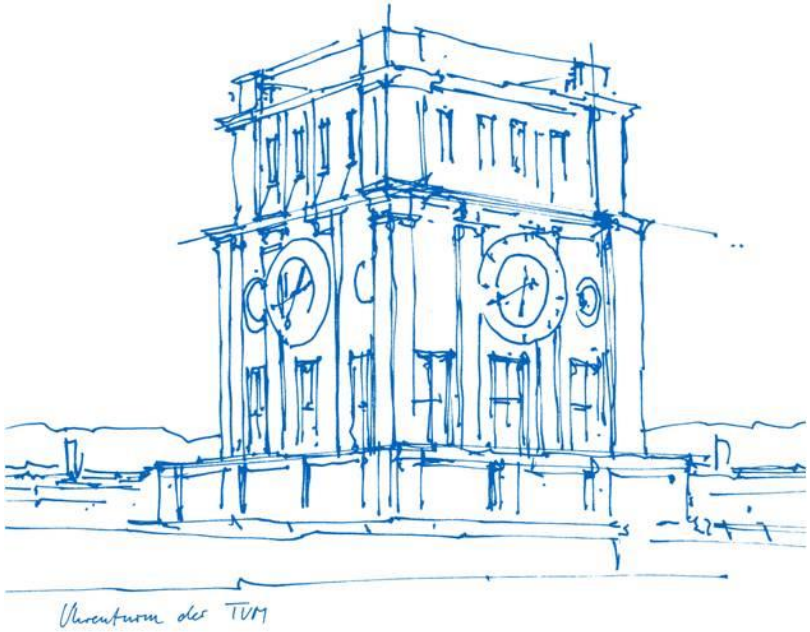
Yuetian Mao & Ludwig Felder

Technical University of Munich

School of Computation, Information and Technology

Chair of Software Engineering & AI

Heilbronn, 15. October 2024



Tutors

Yuetian Mao, M.Sc.

yuetian.mao@tum.de

Ludwig Felder, M.Sc.

ludwig.felder@tum.de

How we run tutorials

- Several exercises each week, related to what was taught in the course
- Discuss and finish the exercises in group
- Optimize the solutions based on reference answers

Team Projects

- **Optional, but highly recommended**
- Add up to 1.0 points to your overall grade
  - Final report and code (80%) + Mid-term and final Presentation (20%)
  - Use GitHub for team collaboration: Code and Documents (requirement analysis document, sprint plan, test report, ...)
  - **Develop both backend and frontend components for your project application**

Build teams

- 3-4 members, including 1 team lead
- Finish tutorial exercises in team
- Work on projects (optional) by team work

LLM-Assisted Application Evaluation at TUM

Students will develop an LLM-based tool to assist the process for evaluating master's program applications at the Technical University of Munich (TUM). This system will help streamline the Curricular Analysis process and provide recommendations based on applicants' essays and motivation letters.

Objectives & Key Components

1. Implement an LLM-based system to analyze and evaluate application materials.
2. Develop a user-friendly interface for the selection commission.
3. Ensure fairness, consistency, and efficiency in the application review process.

Backend:

- Implement LLM system for application analysis
- Develop fair evaluation algorithms

Frontend:

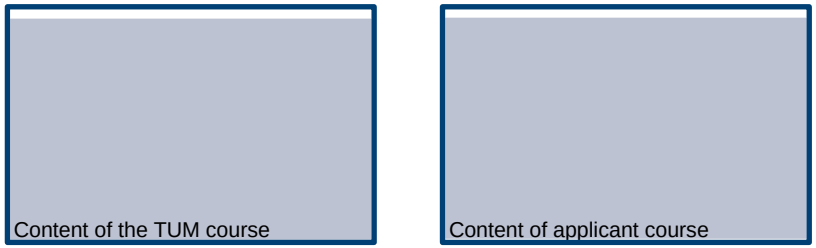
- Create user-friendly interface for selection commission
- Design clear visualization of recommendations

Curricular Analysis

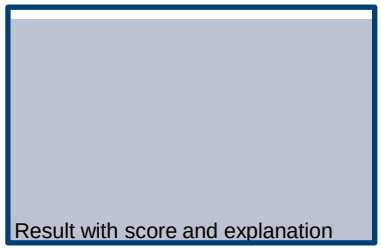
Task
- Read through mandatory course
- Read the max score achievable
- Read through student course(s)
- Compare them and reduce points if necessary
- Return for each comparison with a recommended score and explanation

Metrics
- The main metric is the comparison ”*on the basis of competencies*” the student has acquired
- “*Schematic comparison of the modules*” is generally not important
- The application can reduce the points of the comparison in 0.5 steps
- The ECTS of the student courses, and therefore the comparison should scale with the TUM course

Basic User Interface



for each course:



Core Requirements

- The application should contain an easy to use interface
- Evaluate the documents in a reasonable time
- The system should provide clear, justifiable recommendations to assist human decision-makers
- Ensure robust integration between backend and frontend components.
- Implement secure data handling and communication.
- Optimize overall system performance for both client-side and server-side.
- Think about user privacy and data security in all implementations.
- Design for scalability and future enhancements.

Tasks for the next tutorial

- Research on appropriate tools for your project
- Setup session in the next tutorial
- Present the plan for your projects
  - Planned functionalities
  - Timeline for the project
  - Feedback from tutors