Software Engineering Essentials

Introduction

Bernd Bruegge, Stephan Krusche, Andreas Seitz, Jan Knobloch
Chair for Applied Software Engineering — Faculty of Informatics

Course team

Bernd Bruegge

Stephan Krusche

Andreas Seitz

Jan Knobloch

Motivation

1) Complexity
2) Innovation
3) Flexibility
4) Recognition
5) Opportunities

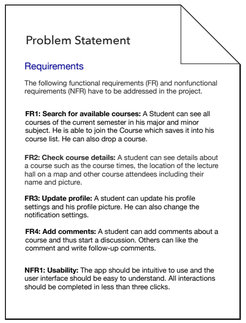
Learning goals

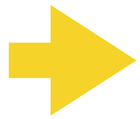
Learn and apply methodologies, techniques, workflows and tools to transform a problem statement given by the customer into a software system used by end users

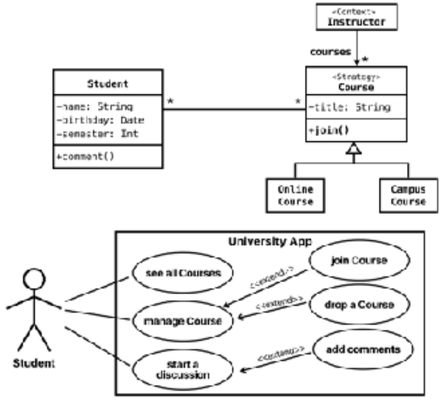
Customer

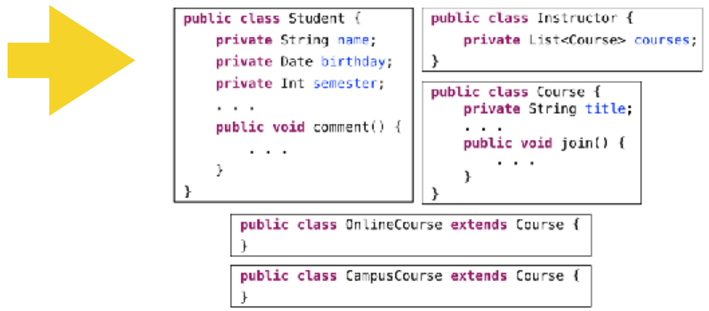
Development Team

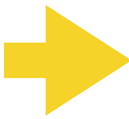
User

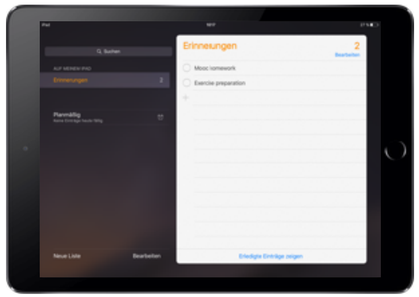












Problem statement

UML models

Source code

Software system

Assumptions and requirements for this course

Assumptions:

- You want to learn more about the development of complex software systems
- For TUM students: you take this course as a bridge course for your master studies

Requirements:

- You have basic experiences with a programming language, e.g. Java
- You must have a notebook to participate in online exercises

Beneficial:

- You have basic knowledge about UML modeling
- You have practical experience with software development

Course structure

The course is 8 weeks long

1) Project organization and management
2) Software configuration management
3) Object oriented programming
4) Requirements analysis
5) System design
6) Object design
7) Testing
8) Build and release management

Structure of a week

A week consists of multiple units. In each unit:

1) We teach knowledge and theories in a video lecture

2) You immediately practice the theories in online exercises

- Multiple choice and drag and drop quizzes
- Programming with automatic feedback
- UML modeling with peer reviews

➡ Earn exercise points

➡ Get immediate feedback

➡ Measure your progress

Communication with Slack

Communication is important in software engineering and education

➡ We want to get in touch with you!

Your first task: signup on https://www1.in.tum.de/slack

- Post questions in the Q&A channel
- Get into touch with other students

Required tools

1) Version control client

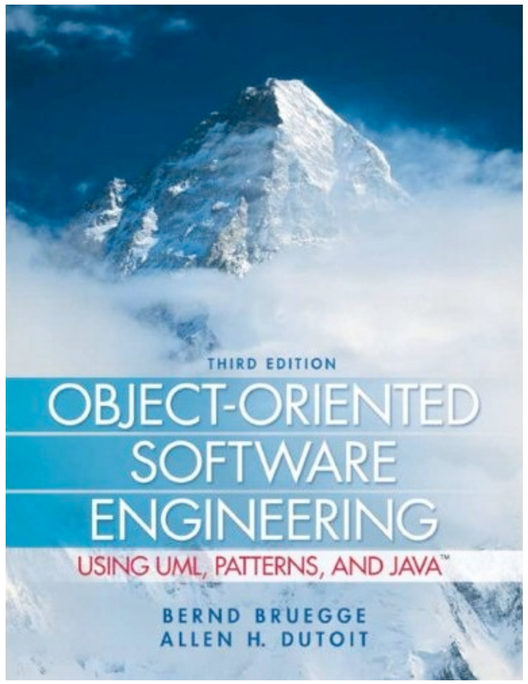
➡ More about it in the **2nd week** in software configuration management

2) Integrated development environment (IDE)

➡More about it in the **3rd week** in object oriented programming

Literature

This course is based on the following book



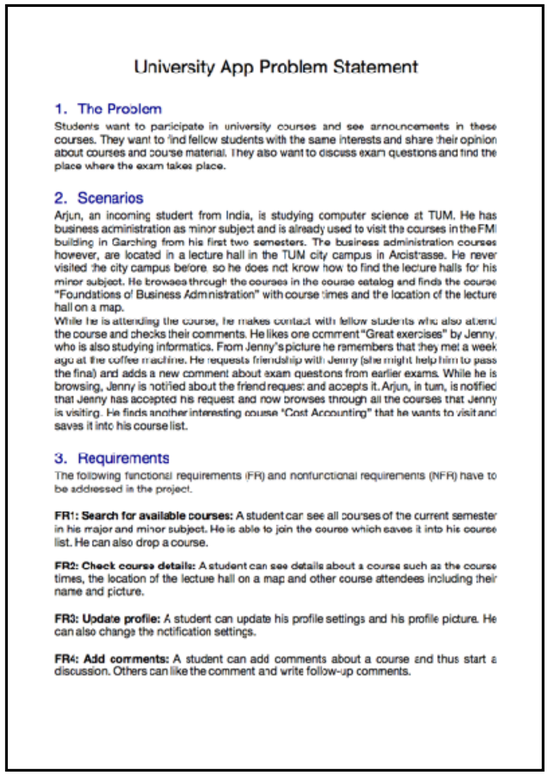
Bernd Bruegge, Allen Dutoit: **Object-Oriented Software Engineering Using UML, Patterns, and Java** (3rd edition)

Homework

1) Read the problem statement about the university app

➡ Discuss the problem statement on Slack

2) Participate in a quick survey about your background



University App Problem Statement

1. The Problem

Students want to participate in university courses and see announcements in these courses. They want to find fellow students with the same interests and share their opinion about courses and course material. They also want to discuss exam questions and find the place where the exam takes place.

2. Scenarios

Arjun, an incoming student from India, is studying computer science at TUM. He has business administration as minor subject and is already used to visit the courses in the FMI building in Garching from his first two semesters. The business administration courses however, are located in a lecture hall in the TUM city campus in Arcisstrasse. He never visited the city campus before so he does not know how to find the lecture halls for his minor subject. He browses through the courses in the course catalog and finds the course "Foundations of Business Administration" with course times and the location of the lecture hall on a map.
While he is attending the course, he makes contact with fellow students who also attend the course and checks their comments. He likes one comment "Great exercises" by Jenny, who is also studying informatics. From Jenny's picture he remembers that they met a week ago at the coffee machine. He requests friendship with Jenny (she might help him to pass the final) and adds a new comment about exam questions from earlier exams. While he is browsing, Jenny is notified about the friend reques and accepts it. Arjun, in turn, is notified that Jenny has accepted his request and now browses through all the courses that Jenny is visiting. He finds another interesting course "Cost Accounting" that he wants to visit and saves it into his course list.

3. Requirements

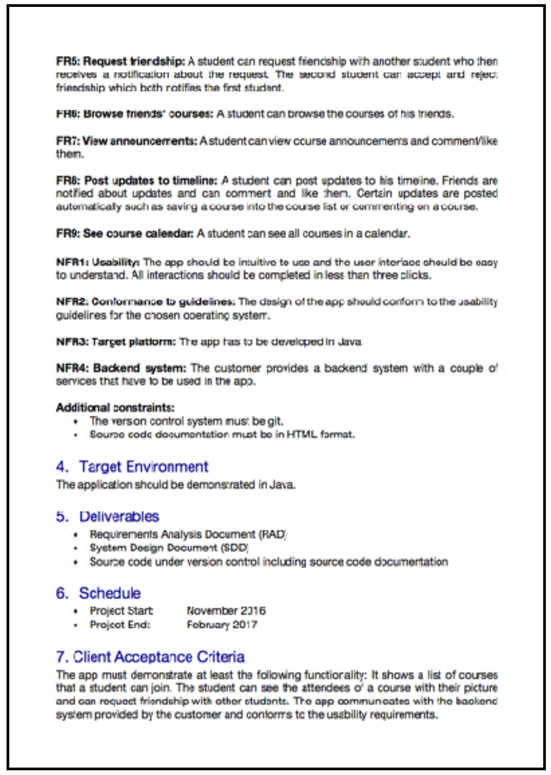
The following functional requirements (FR) and nonfunctional requirements (NFR) have to be addressed in the project.

FR1: Search for available courses: A student can see all courses of the current semester in his major and minor subject. He is able to join the course which saves it into his course list. He can also drop a course.

FR2: Check course details: A student can see details about a course such as the course times, the location of the lecture hall on a map and other course attendees including their name and picture.

FR3: Update profile: A student can update his profile settings and his profile picture. He can also change the notification settings.

FR4: Add comments: A student can add comments about a course and thus start a discussion. Others can like the comment and write follow-up comments.



FR5: Request friendship: A student can request friendship with another student who then receives a notification about the request. The second student can accept and reject friendship which both notifies the first student.

FR6: Browse friends' courses: A student can browse the courses of his friends.

FR7: View announcements: A student can view course announcements and comment/like them.

FR8: Post updates to timeline: A student can post updates to his timeline. Friends are notified about updates and can comment and like them. Certain updates are posted automatically such as saving a course into the course list or commenting on a course.

FR9: See course calendar: A student can see all courses in a calendar.

NFR1: Usability: The app should be intuitive to use and the user interface should be easy to understand. All interactions should be completed in less than three clicks.

NFR2: Conformance to guidelines: The design of the app should conform to the usability guidelines for the chosen operating system.

NFR3: Target platform: The app has to be developed in Java.

NFR4: Backend system: The customer provides a backend system with a couple of services that have to be used in the app.

Additional constraints:
- The version control system must be git.
- Source code documentation must be in HTML format.

4. Target Environment

The application should be demonstrated in Java.

5. Deliverables

- Requirements Analysis Document (RAD)
- System Design Document (SDD)
- Source code under version control including source code documentation

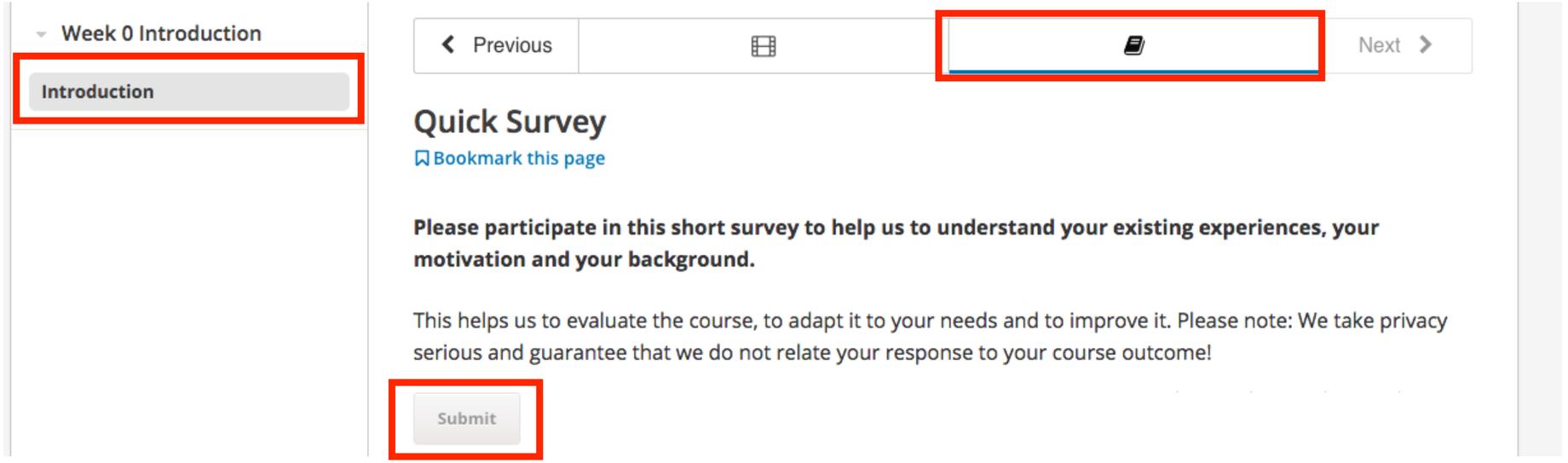
6. Schedule

- Project Start: November 2016
- Project End: February 2017

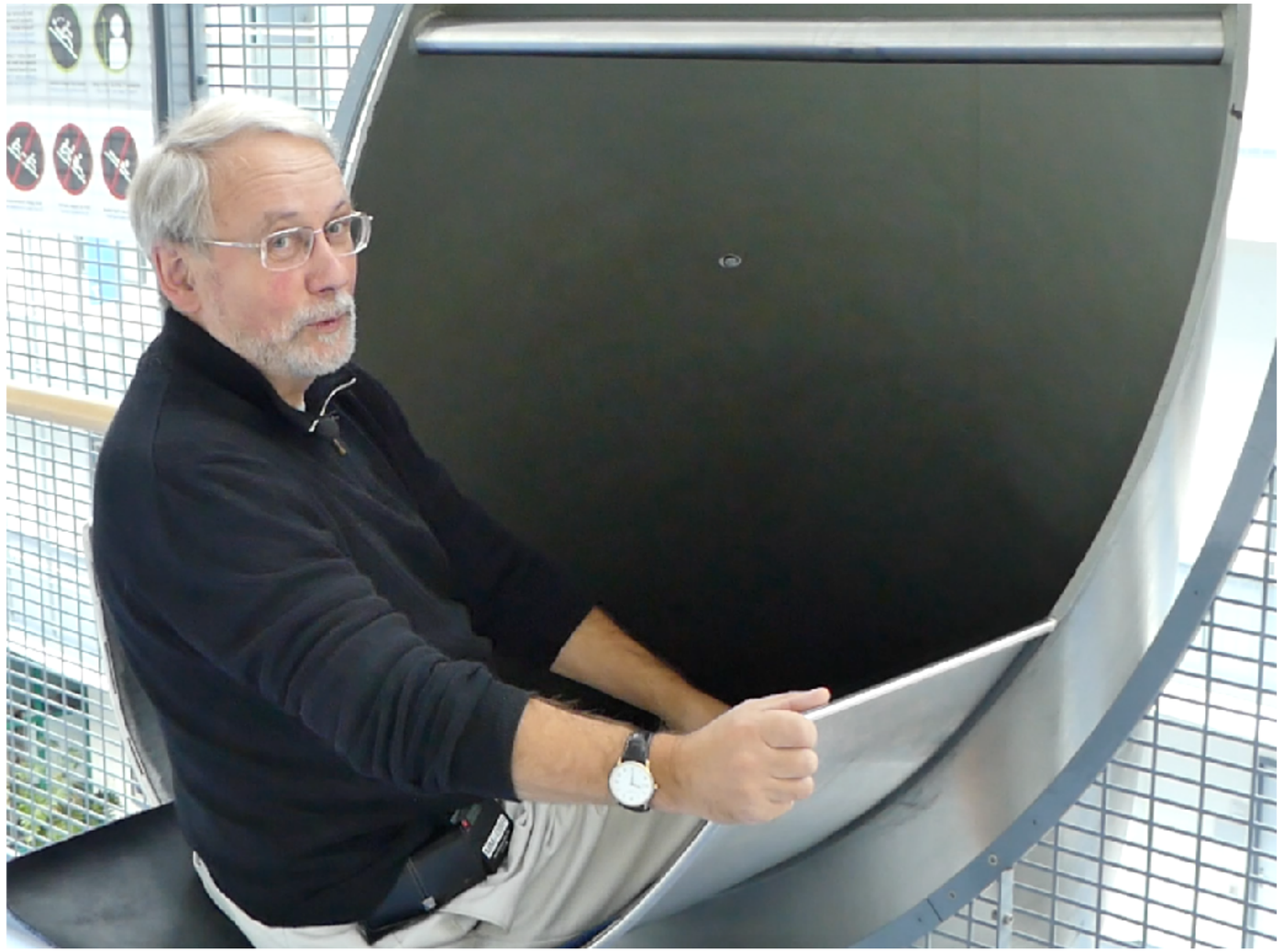
7. Client Acceptance Criteria

The app must demonstrate at least the following functionality: It shows a list of courses that a student can join. The student can see the attendees of a course with their picture and can request friendship with other students. The app communicates with the backend system provided by the customer and conforms to the usability requirements.

Participate in a quick survey



Let’s start and have fun!



Software Engineering Essentials

Introduction

Bernd Bruegge, Stephan Krusche, Andreas Seitz, Jan Knobloch
Chair for Applied Software Engineering — Faculty of Informatics