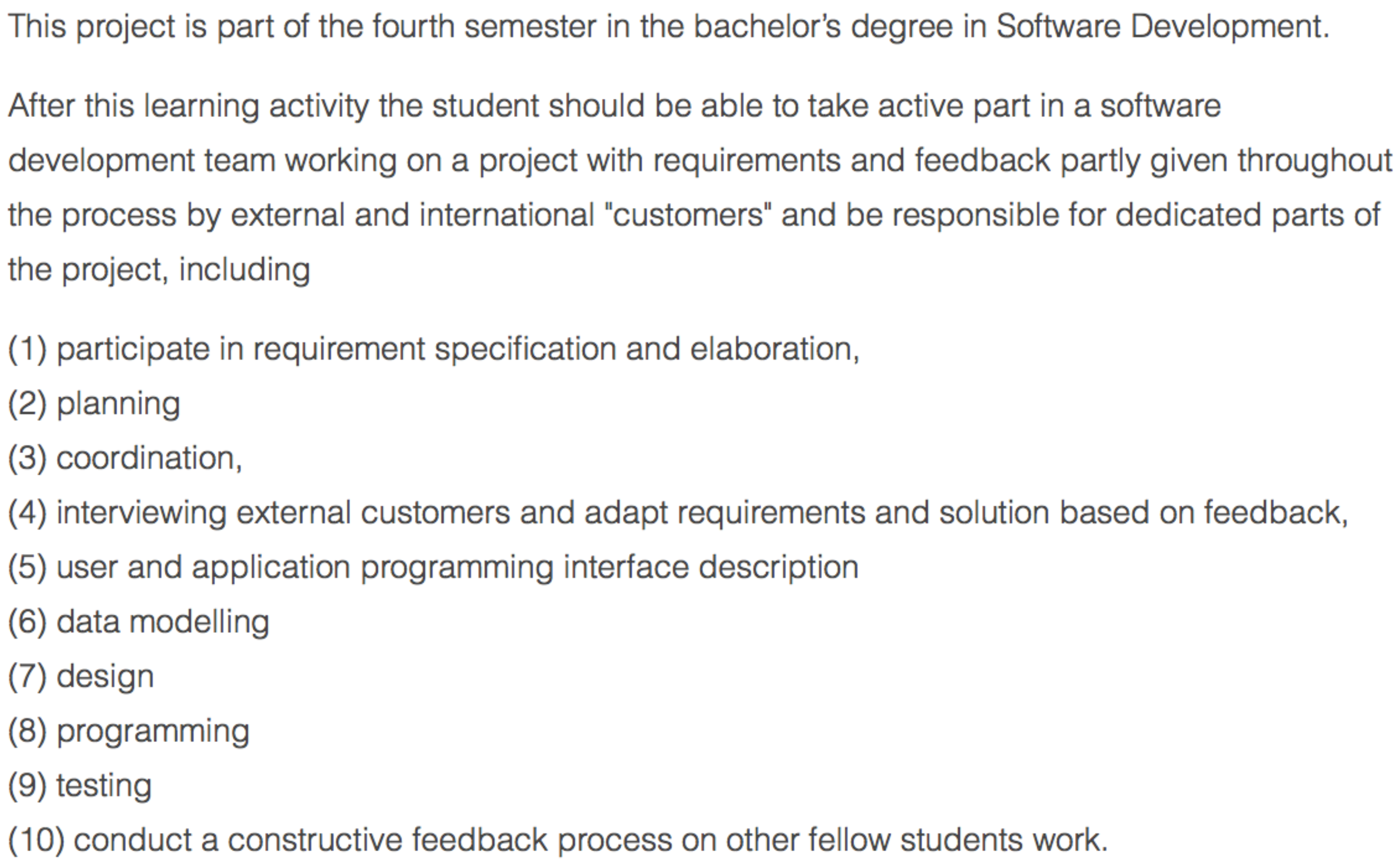
2nd Year Project: SW Development in Large Teams with International Collaboration

Lecture on evaluation & requirements for final hand in, 2015

IT UNIVERSITY OF COPENHAGEN

Learning Outcomes

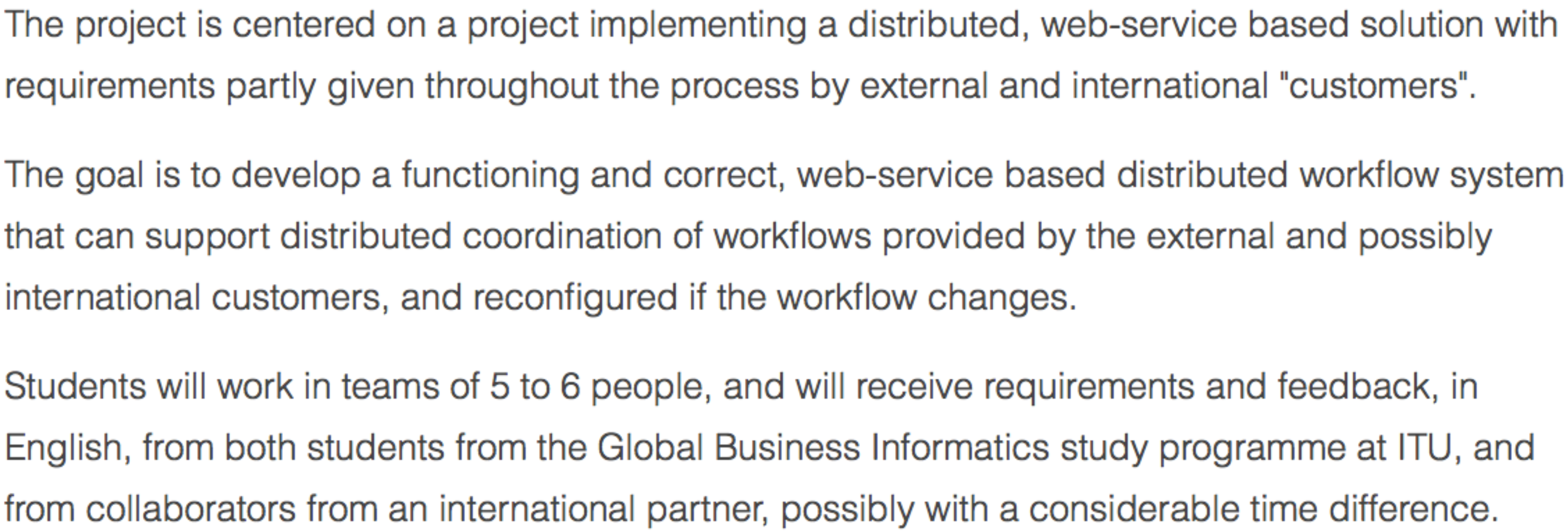


This project is part of the fourth semester in the bachelor's degree in Software Development.

After this learning activity the student should be able to take active part in a software development team working on a project with requirements and feedback partly given throughout the process by external and international "customers" and be responsible for dedicated parts of the project, including

(1) participate in requirement specification and elaboration,
(2) planning
(3) coordination,
(4) interviewing external customers and adapt requirements and solution based on feedback,
(5) user and application programming interface description
(6) data modelling
(7) design
(8) programming
(9) testing
(10) conduct a constructive feedback process on other fellow students work.

Contents



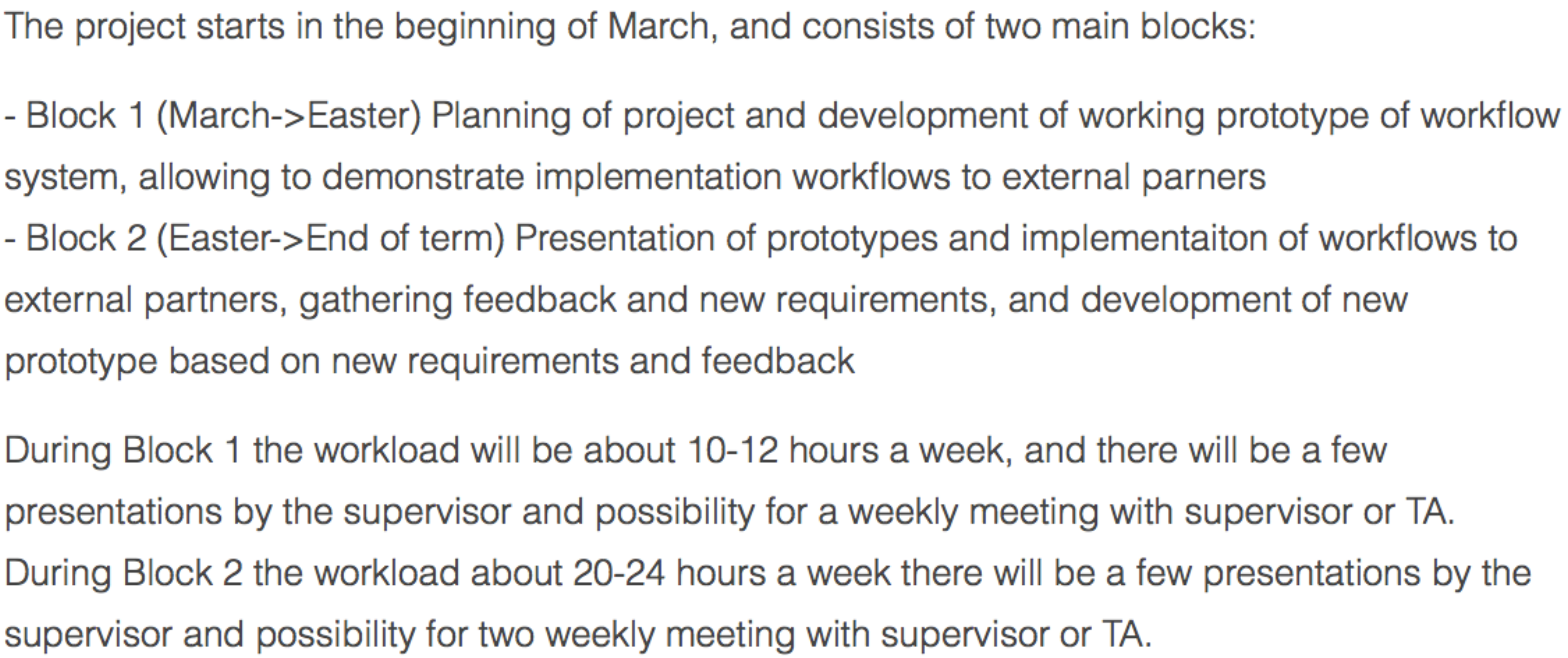
The project is centered on a project implementing a distributed, web-service based solution with requirements partly given throughout the process by external and international "customers".

The goal is to develop a functioning and correct, web-service based distributed workflow system that can support distributed coordination of workflows provided by the external and possibly international customers, and reconfigured if the workflow changes.

Students will work in teams of 5 to 6 people, and will receive requirements and feedback, in English, from both students from the Global Business Informatics study programme at ITU, and from collaborators from an international partner, possibly with a considerable time difference.

I assume you have formed project groups

Activities - overall form

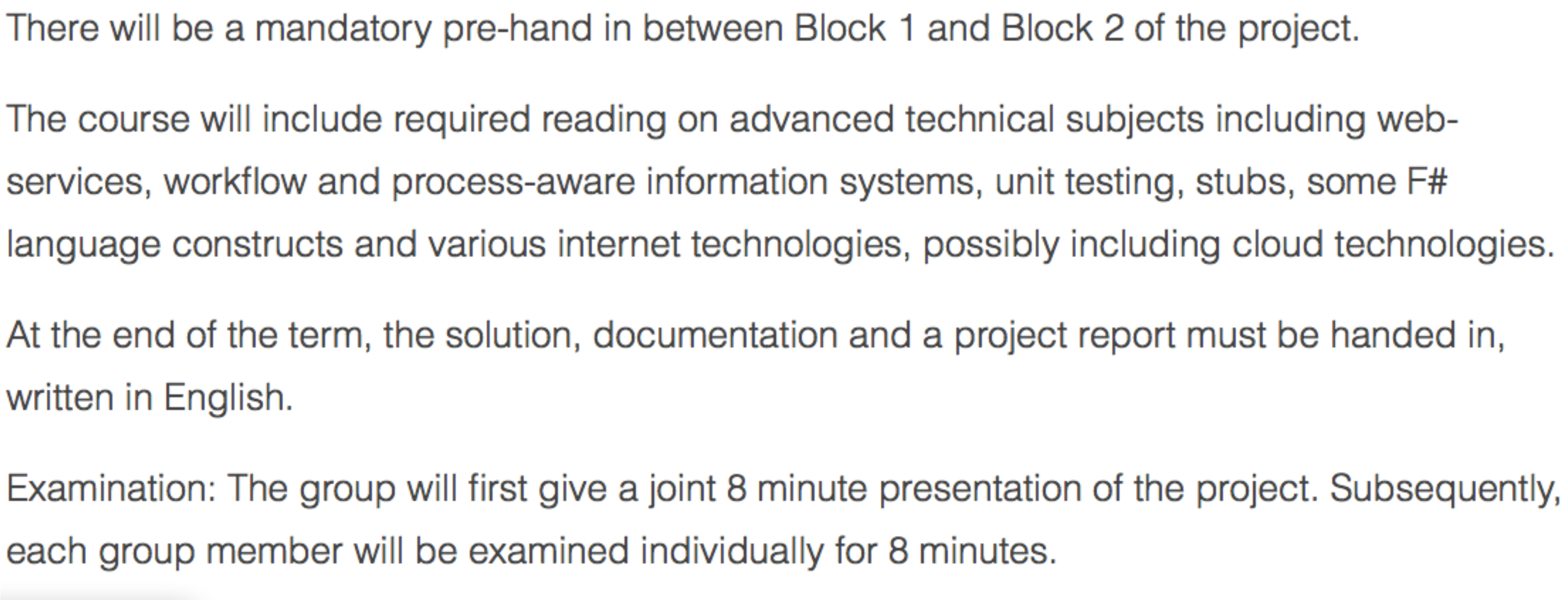


The project starts in the beginning of March, and consists of two main blocks:

- Block 1 (March->Easter) Planning of project and development of working prototype of workflow system, allowing to demonstrate implementation workflows to external parners
- Block 2 (Easter->End of term) Presentation of prototypes and implementaiton of workflows to external partners, gathering feedback and new requirements, and development of new prototype based on new requirements and feedback

During Block 1 the workload will be about 10-12 hours a week, and there will be a few presentations by the supervisor and possibility for a weekly meeting with supervisor or TA. During Block 2 the workload about 20-24 hours a week there will be a few presentations by the supervisor and possibility for two weekly meeting with supervisor or TA.

Activities - overall form



There will be a mandatory pre-hand in between Block 1 and Block 2 of the project.

The course will include required reading on advanced technical subjects including web-services, workflow and process-aware information systems, unit testing, stubs, some F# language constructs and various internet technologies, possibly including cloud technologies.

At the end of the term, the solution, documentation and a project report must be handed in, written in English.

Examination: The group will first give a joint 8 minute presentation of the project. Subsequently, each group member will be examined individually for 8 minutes.

Rough outline of course

- March (Tuesdays 8-12):
  Implement Workflow engine as REST services
  (I will give lectures the next two Tuesdays 11-12)

- April 7-May 12 (Tuesdays 8-12, Thursdays 10-14)
  Test on workflows, get feedback, improve, add features
  April 9: Pre-hand-in,
  April 14: Group presentations & feedback
  (lectures/new requirements other Tuesdays 11-12)

- May 15: Hand in of final project

Requirements for hand in

- Generic, distributed workflow engine based on REST implementation of core DCR Graphs (with roles) where every event offer the same interface to each other as described in Part 1 of the project description (like a P2P architecture as opposed to client-server, p63DS) that can be easily tested on server and local host DCR Graphs
- Must also implement: UI + event log, Concurrency control, Role based Access Control, Persistence

Report

- Background and requirements
- Elaboration of requirements to workflow
- Description of your implementation
- Testing
- Role Based Access control (ch 11)
- Concurrency control - problem and solution (ch 16)
- Reflection on project and conclusion
- Appendixes: Code & tests

Report

- Background and requirements:
  - Stating the requirements
  - Workflows, DCR Graphs, REST (~3 pages)
  - The hospital workflow
  - Your initial plan, including your division of the work in the group (~1 page)

Report

- Elaboration of workflow requirements:
  - Present and explain the DCR Graph you presented to Eduardo, his feedback and your changes based on that (~3 pages)
  - Present the additional workflow given from GBI students

Report

- Description of your implementation (~10-15 p)
  - Overview of implemented functionality and choices made
  - Architecture
  - Overview of classes/modules implementing your system, explaining key parts
  - Illustrative example runs & user guide

Report

- Testing (~2 pages)
  - Describe how you tested your implementation
  - (Include test code and outcome of tests as appendix)

Report

- Role Based Access Control (~3 pages)
  - Describe your approach to implementing RBAC
  - Discuss (relate to theory in ch 11)

Report

- Concurrency Control (~3-4 pages)
  - Describe concurrency problems and your approach
  - Discuss your solution
  - (relate to concepts in ch 16)

Rest of the term

- April 23: Rasmus lecture on Azure 10-12
- Tuesdays and Thursdays 10-12 from April 28-May 12: Work on implementation and report
- Friday May 15: Hand in via Learnit
- June 8,9, 11 (until 14:15) ,12 (until 14:15) Exams (individual responsibility of parts of implementation allowed - but everyone should have contributed proportionally and be able to answer question to the process and background)