Welcome to Software Engineering II! Thanks for joining!

We focus on Agile software development in Scrum teams

All details/slides on website (we'll update as fast as possible):

Introduction: Participants

Responsible
■ Dr. Michael Perscheid (michael.perscheid@hpi.de)

Teaching Team (swt2_20_orga@lists.myhpi.de)
■ Christoph Matthies (christoph.matthies@hpi.de)
■ Ralf Teusner (ralf.teusner@hpi.de)
■ Tutors: Wanda Baltzer, Luc Prestin, Gritta Weisheit

Students
■ You! Without your participation this course won't work!
Introduction and Organization — Software Engineering II
Introduction: EPIC Chair
Introduction: You

- 3rd semester? 5th semester? 1st semester?! Not HPI?
- What are your previous experiences...
  - concerning software development team work?
  - concerning web development?

- What are your expectations for this course?
  - What do you hope to learn?
  - What do you hope to experience?
  - What is your personal goal?

https://docs.google.com/document/d/1aT_sP-rr-tZV5BGI8cwrallE8aL0zaXRjleVbl3pins/edit?usp=sharing
What is SWT II?

This is a project course!
The focus is on practical work in teams
You will learn through experimentation and trying out collaboration techniques
Team meetings are vital, you need to make time for them

Learning Objectives

After this course, all students should have

1. Experience with Scrum and all of its artefacts and meetings
2. Learned how to scale Scrum over multiple collaborating teams
3. The ability to use Agile development practices, such as BDD and TDD, where appropriate
4. Confidence in using the full feature set of a source code management (SCM) system
5. Experienced the value of rapid release cycles and continuous integration (CI)
6. Learned to critically self-assess their role in a team
What is SWT II?

Lecture

- Scrum and Agile practices in large teams
- Guest lectures
- Agile methods beyond Scrum
- Requirements engineering
- Behavior Driven Development/Test Driven Development
- Development tools
- Continuous Integration
- DevOps
- Topics you really want?
What is SWT II?

**Software project**
- One single project for all participants, entire course
- Most major project roles facilitated by you!
- **You influence and organize the project!**
- Realistic (coordination) challenges and problems
- Open source on GitHub (MIT license), your contributions are public
- Application deployed and hosted publicly

*Have you contributed to open-source projects before?*
What is SWT II?

HPI Connect Jobportal
What is SWT II?

**Time management**
- Checking up on yourself part of the experience
- One fifth of the week
- Overtime discouraged
Prerequisites
- Undergraduate program
- Softwaretechnik I
- Interest in learning and working in project teams

Course
- 4 SWS (≈8h work per week including lectures)
  - Some lecture slots will be used for more group work time
  - 6 ECTS credit points (graded)
Grading

The final grading is determined by

- 30%: oral exam (date tbd)
- 45%: usage of methods and practices presented in the lecture: involvement in the development process, teamwork participation, Scrum meetings, BDD, TDD, SCM
- 25%: software development results (team mark)

Completion of the introductory exercise is mandatory for passing the course.

- Product Owners are exempt from the exercise (will work with the customer)
COVID-related Course Adaptations

Due to COVID19, the following course adaptations have been made for the winter term 20/21. These are subject to change as the situation evolves.

- Lectures will *not* be held in the lecture hall. They will be live streamed during regular lecture slots, enabling direct interaction and interactivity with the presenter. We rely on course participants engaging actively.
- We will be using a Discord instance as a central collaboration and communication platform.
- Participation is capped at a maximum of 30 participants.
- Team size is capped at a maximum of 8.
- Large meeting spaces (at Campus III) will be provided, which allow team meetings with adequate distances. Attendance sheets will be used.
- It will be possible to attend team meetings virtually as well.
Virtual lecture format

- Live, so that we can interact
- Please let us know what works (and what doesn't)
- We're learning too!
- Anything that you really liked in other lectures?
SWT II Setup

Scrum Team

- Product Owner
- Scrum Master
- Development Team
Participants form teams of **absolute max 8** participants

**Recommended: 5-6 team members**

**Each team consists of**

- 1 Product Owner (PO)
- 1 Scrum Master (SM), who is also ½ developer
- Development Team members
SWT II Setup

- Stakeholders
- Customer
- PO Coach
Teaching Team Roles

■ **Customer**
  □ Represents the clients & stakeholders of the developed software
  □ Has ideas and requirements for the software to be built

■ **Product Owner Coach**
  □ Main contact point for team POs
  □ Helps in dealing with customer requirements & process

■ **Tutors** as Scrum consultants
  □ Present during all major meetings
  □ Open for questions, advice, help & (crazy) ideas
  □ **Coordinate with them!**
Collaborative software development

- Web development
- Programming framework: Ruby on Rails (*who has used that?*)
- Minimal core will be provided
- Results will be open source on GitHub (*well-known at this point?!*)

Engineering focus

- Understanding of web (MVC) stack and components (*404 is error, ok is ...?*)
- Avoiding “patchwork” (UI, Workflows, Data)
- Maintainability of the code base (tests, quality, etc.)
- Functionality (*not on top of the list for a reason*)
Remote teams are now having "coffee raffles"
- Less physical interactions means fewer chance encounters at the coffee machine
- Less talking about stuff that isn't work-related, less Mensa
- More chance to lose contact
- **An application to bring some of that back**

And make it even better?!
- Social contacts are really important! Can an application help me remember things?
- What is the ideal remote "social recharge"?
- What is the digital equivalent of a "watering hole"?
### Project

**Sounds interesting?**
- Maybe you'd like to try out the role of Product Owner?!
- Connect with the customer and find out more!
- Do your own research!

**Change and Adaptation**
- Requirements Engineering is part of this course
- Discussion and meetings with the customer
- Requirements might change
Next Steps

- Mail your team choice to **swt2_20_orga@lists.myhpi.de**
  - Until **Sunday, November 15**
  - Who will be PO and SM?
  - Three suggestions for weekly meeting slot

- POs: mail us suggestions for **first customer meeting**
  - As soon as possible (please, gives you more time)
  - As close to November 16 as possible
Next Steps

Discord

- **Sign-up!** (link on the website)
- Team discussions
- Make a channel for your team
- **Coordinate meeting times**
- Strongly recommended as a single point of communication
To get started in Rails...

- Tutorial exercise based on GitHub issues
- Need a GitHub account

Accept the assignment —
Rails Exercise '20

Once you accept this assignment, you will be granted access to the rails-exercise-20-chrisma repository in the swt2-intro-exercise organization on GitHub.
Next Steps

Rough Schedule

■ 27.11. Project Kick-Off
■ Sprint 1 (2 weeks)
■ Sprint 2 (2 weeks, academic X-Mas break)
■ Sprint 3 (2 weeks)

No schedule survives contact with reality

■ Real teamwork brings real challenges
  □ Actually writing a software is vital
  □ External constraints may change
■ Schedule can adapt
  □ Also according to your suggestions

■ January: Intermediate Presentation
■ Last weeks: Kanban Iteration
■ End of semester in February: Final Presentation
General literature

- Verner, June M. et al. "In the 25 years since The Mythical Man-Month what have we learned about project management?." *Information and Software Technology* (1999)

If you can't find these items in the library or online, please send us an email. We might be able to help.
Next Steps

**Lectures**
- Friday, 09:15 – 10:45
- Friday, 11:00 – 12:30

*Check the website for major changes*
Next Steps

What to do next

■ Sign up for communication channels
■ **Form teams** *(how many are planning to take part in the course?)*
  ■ Discord
  ■ Email/message us if you'd like to be assigned
  ■ Discuss who takes on roles of Product Owner, Scrum Master and developers
■ **Find time slots** that work for all team members
  ■ If none can be found, rearrange teams
  ■ Time for collaborative work is crucial!
■ **Send team composition & meeting time slot proposals** to teaching team
■ Start Ruby on Rails Intro exercise
- Want to talk about team composition amongst yourselves?
- What is the best song that deals with computers?