Agile Software Development

L01 – Course organization

Ezequiel Scott
ezequiel.scott@ut.ee
Learning goals

To introduce some of the practices on agile software development, taking as an example the development of applications labelled as “software as a service”
Rationale of the course

Strong connection with
MTAT.03.229 – Enterprise System Integration

Great opportunity to introduce/recall some concepts:

- Software development practices
- Development of web-based applications
- Use of cloud-based tools
Approach

• Learn the bases (agile practices)
• Learn a technology stack (for SaaS)
• Put them into practice (warm up – labs)
• Develop a SaaS project from conception to deployment
  • Applying agile practices
  • Using a technology stack
Course Organization

• Lectures
  • Tuesdays 10:15-12:00, J. Liivi 2 - 111
    • Weeks 1-5: Ezequiel Scott
    • Weeks 6-15: Orlenys López-Pintado

• Practical sessions
  • Group 1, Fridays 14.15-16.00, J. Liivi 2 - 403
  • Group 2, Tuesdays 12.15-14.00, J. Liivi 2 – 404
    • Weeks 2-15 (Starts on 10.09)
<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>03.09</td>
<td><strong>Introduction</strong></td>
<td>15.10</td>
<td>Phoenix app’s architecture, a deep live</td>
</tr>
<tr>
<td>10.09</td>
<td>Test-Driven Development</td>
<td>22.10</td>
<td>Phoenix: Authentication and authorization</td>
</tr>
<tr>
<td>17.09</td>
<td>Requirements and Iterative Software Development</td>
<td>29.10</td>
<td>Front-end Development: Introduction to Vue.js</td>
</tr>
<tr>
<td>24.09</td>
<td>Code Smells, Code Refactoring, and Traceability</td>
<td>05.11</td>
<td>Phoenix channels and Vue.js Components</td>
</tr>
<tr>
<td>01.10</td>
<td>Agile Software Development with Scrum</td>
<td>12.11</td>
<td>Deployment and Continuous Integration</td>
</tr>
<tr>
<td>08.10</td>
<td>Behaviour Driven Development (BDD)</td>
<td>19.11</td>
<td>Project Consultation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10.12</td>
<td></td>
</tr>
</tbody>
</table>
Lab instructors

Orlenys López-Pintado
orlenyslp@ut.ee

Maksym Yerokhin
maksym.yerokhin@ut.ee
How to pass the course

- **Homeworks** (submitted in pairs)  10 points
  - 2 lab assignments x 5 points

- **Project** (team-based, 4 members)  35 points
  - Evidence of use of agile practices
  - Assessment of the delivered software (code + tests)
  - Sprint reviews (weekly)
    - (Commitment per sprint) * completeness
    - (5 sprints * 7 points) * completeness

- **Final exam**  55 points
  - You need a mark of at least 23 points out of 55 to pass the course
  - Structure of the exam: Theory (20p) / Practice (35p)
**Assessment**

- All members in a team receive equal grades in labs
- **BUT**: Exceptions from equal grade rule will be made, if individuals in a team don’t participate actively
- Lab assignments: team penalties apply for late delivery
  - 24h → -25%
  - 48h → -50%
  - >48h → -100%
- Sprint Reviews: Individual penalties apply for not attending the reviews
- Don’t plagiarize!
Communication channels

Messages / Feedback: Slack !!!