Seminar
Requirements Engineering in Software Evolution

Prof. Martin Glinz
Emitzá Guzmán, Eya Ben Charrada
Goal

Analyze and present scientific work in the field of requirements engineering
Organization

3 ECTS

Seminar Process

1. Write seminar paper (15 pages Springer style)
2. Peer-review (2 papers)
3. Present (20 min)
4. Participate in other presentations

Grading

2/3 seminar paper + reviews, 1/3 presentation + participation
<table>
<thead>
<tr>
<th>Day</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>20.02</td>
<td>First meeting</td>
</tr>
<tr>
<td>24.02</td>
<td>Choosing topic deadline</td>
</tr>
<tr>
<td>~15.03, by appointment or email</td>
<td>Q&amp;A seminar topic, paper structure, main storyline (voluntary)</td>
</tr>
<tr>
<td>~30.03, by appointment or email</td>
<td>Review and discussion of draft (voluntary)</td>
</tr>
<tr>
<td>20.04, 23:59 CET</td>
<td>Paper submission deadline</td>
</tr>
<tr>
<td>2.05, 23.59 CET</td>
<td>Peer-review submission deadline</td>
</tr>
<tr>
<td>16.05, 23:59 CET</td>
<td>Camera-ready submission deadline</td>
</tr>
<tr>
<td>~23.05, by appointment or email</td>
<td>Review and discussion of presentation (voluntary)</td>
</tr>
<tr>
<td>01.06</td>
<td>Final presentations</td>
</tr>
</tbody>
</table>
Q&A Seminar Topic

By appointment

Prepare the meeting
  – Draft an outline
  – For each planned section have a short description of what you plan to write

Send draft outline at least 2 days before the meeting
Review and Discussion of Draft

Read the material on writing before starting to write

Send draft 7 days *before* your appointment

Have a coherent version available (details may still be missing)
Peer Reviews

Review papers from two other students

Adhere to the reviewer forms (will be provided)

Be constructive, make concrete suggestions for improvement

Check for plagiarism, let us know if plagiarism is found!
Final Presentation Day

20 min. presentation + 10 min. Q&A and discussion

Each reviewer presents two questions in the discussion

Encouraged to participate in discussions where you are not reviewer!
Final Presentation Day

Participation on presentation day compulsory

Prepare presentation well, will be part of your grade!

Train your presentation with a peer beforehand
News and Information

Visit

http://www.ifi.uzh.ch/rerg/courses/fs17/sem-re-bsc.html

http://www.ifi.uzh.ch/rerg/courses/fs17/sem-re-msc.html
Seminar Theme

Requirements Engineering in Software Evolution
Main Themes

1. Elicitation and Automatic Analysis

1. Decision Making

1. Traceability and Management
Requirements Elicitation

- Which methods exist for eliciting requirements during software evolution?
- What are their advantages and limitations?
- **Focus on:** all methods except those that use social media/app stores OR only on app store/social media

**References:**

Requirements Prioritization

- Which methods exist for prioritizing requirements during software evolution?
- What are their advantages and limitations?
- **Focus on:** Agile projects OR other SE methodologies

References:
Release Planning

- Which release planning methods exist?
- In which context could these methods be applied?
- What are the limitations of these methods?
- **Focus on:** Agile projects OR other SE methodologies

References:
Requirements Traceability for Managing Software Evolution

- What is traceability?
- What are the benefits of traceability and what does it cost?
- How can traces be established and maintained?
- How can traceability support the evolution of software systems?

References:
Co-evolving Software Artifacts

- When software evolve, are the requirements and other software artifacts kept up-to-date in practice? If not why?
- What approaches are there to support the update of software artifacts?

References:
Classifying Requirements Change

• What are the types of software and requirements change?
• What are the main causes and triggers for changes in requirements?
• What are the costs and effects of those changes?

References:

Next Steps

Let us your topic preference after class or
Send your three most favorite topics to guzman@ifi.uzh.ch, topics will be assigned on a first come, first serve basis.
Explicit User Feedback Analysis

• What challenges arise when analyzing explicit user feedback?
• What are current strategies for solving these issues?
• What are the limitations of these strategies?
• How can explicit feedback be useful for software evolution and requirements engineering?

References:
Implicit User Feedback Analysis

- What challenges arise when analyzing implicit user feedback?
- What are current strategies for solving these issues?
- What are the limitations of these strategies?
- How can implicit feedback be useful for software evolution and requirements engineering?

References: