Software Evolution and Quality

Seminar in Advanced Software Engineering, FS 2012

Thomas Fritz
Harald Gall
Giacomo Ghezzi

s.e.a.l.
software evolution & architecture lab

Organizational Announcements

- 3rd year and up (prerequisite: Software Engineering)

- Language for report and presentation is English

- Work independently

- Don't forget to do the "Modulbuchung"!

Seminar Goals

- Introduce you to a part of research in software engineering

- Improve your ability to critically read and analyze scientific papers

- Strengthen your technical writing and your presentation skills

- Improve your ability to ask and answer critical questions

What you have to do

- **Do a literature review for assigned topic** (at least 7 to 10 relevant articles)

- **Write a report and refine it** (12 to 15 pages)

- **Review two reports**

- **Present your findings** (20mins + 10mins questions)

- **Actively participate in discussions**

3 credits / 90 hours: approx. 70 report & review, 20 presentation

Software Evolution and Quality – Topics

- Supporting Crosscutting Concerns

- Code Recommendation

- Tool Support for Tasks/Bugs

- Software Quality Tools

- Refactoring

- Software Architecture

- APIs

Topic – Supporting Crosscutting Concerns

- Aspect-Oriented Programming.

- N degrees of separation: multi-dimensional separation of concerns.

- Concern graphs: finding and describing concerns using structural program dependencies.

Topic – Code Recommendation

- Using structural context to recommend source code examples.

- Mining version histories to guide software changes.

Topic – Tool Support for Tasks/Bugs

- Using task context to improve programmer productivity.

- Who should fix this bug?

[Summarizing software artifacts: a case study of bug reports.]

Topic – Software Quality Tools

- Yesterday my program worked today it does not. Why?

- Change analysis with Evolizer and ChangeDistiller.

- Tools for continuously evaluating distributed-system qualities.

Topic – Refactoring

- A survey of software refactoring.

- How we refactor and how we know it.

- Automated testing of refactoring engines.

Topic – Software Architecture

- Using software evolution to focus architectural recovery.

- ArchJava: connecting software architecture to implementation.

- Software Reflexion Models: bridging the gap between source and high-level models.

Topic – APIs

- What makes APIs hard to learn? Answers from developers.

- How to design a good API and why it matters.
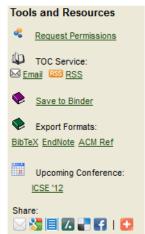
Finding Relevant Work

- Google scholar

- Proceedings of main conferences (ICSE, FSE, ESEM, ICGSE, ICPC, CSCW)

- References
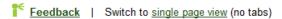
- Related work section

- Cited by

ACM DL DIGITAL LIBRARY

Consortium of the Swiss Academic
Libraries
Universität Zürich

Code bubbles: rethinking the user interface paradigm of integrated development environments

Full Text: PDF

Authors:
Andrew Bragdon	Brown University
Steven P. Reiss	Brown University
Robert Zeleznik	Brown University
Suman Karumuri	Brown University
William Cheung	Brown University
Joshua Kaplan	Brown University
Christopher Coleman	Brown University
Ferdi Adeputra	Brown University
Joseph J. LaViola, Jr.	University of Central Florida

2010 Article

Bibliometrics
· Downloads (6 Weeks): 13
· Downloads (12 Months): 258
· Citation Count: 11

Published in:

· Proceeding

ICSE '10 Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 1

ACM New York, NY, USA ©2010
table of contents ISBN: 978-1-60558-719-6   doi>10.1145/1806799.1806866

Tools and Resources

Request Permissions

TOC Service:
Email RSS RSS

Save to Binder

Export Formats:
BibTeX EndNote ACM Ref

Upcoming Conference:
ICSE '12

Share:

Tags: bubbles concurrent views debugging human factors human factors integrated development environments integrated environments navigation source code working set

Feedback | Switch to single page view (no tabs)

Abstract	Authors	References	**Cited By**	Index Terms	Publication	Reviews	Comments	Table of Contents

11 Citations

Robert DeLine , Gina Venolia , Kael Rowan, Software development with code maps, Communications of the ACM, v.53 n.8, August 2010

Robert DeLine , Gina Venolia , Kael Rowan, Software Development with Code Maps, Queue, v.8 n.7, July 2010

John Hardy , Christopher Bull , Gerald Kotonya , Jon Whittle, Digitally annexing desk space for software development: NIER track, Proceeding of the 33rd international conference on Software engineering, May 21-28, 2011, Waikiki, Honolulu, HI, USA

14

Expectations

- Find good structure / outline / categorization
- Find commonalities, specialties, differences,…
- Critical thinking (also constructive)
- Some reflection on your own
- Higher for master than for bachelor students

- Wikipedia is not an option!

Presentation

- Several guidelines on website
- Some more:
    - Don't exceed the time limit
    - Practice the talk
    - Don't ignore the audience
    - Avoid too many slides, too many bullets, fonts too small, too much text
    - Have a flow

- Scientifically Speaking:
  http://www.erp.wisc.edu/profdev/Scientifically_speaking.pdf

Deadlines

- 4th Mar – Email with 3 prefs to Giacomo

- 5th Mar – Topics assignment

- 6th Apr – Report submission (midnight hard deadline)

- 16th Apr – Review period end

- 17th Apr – Notification

- 27th Apr – Corrected report submission

- 15th May – Presentation Day (1pm, mandatory)
- 16th May – Second Presentation Day (1pm)

More Information

- See website: http://seal.ifi.uzh.ch/seminar
    (also includes presentation guidelines)

- Contact:
    - Thomas Fritz   fritz@ifi.uzh.ch
    - Harald Gall   gall@ifi.uzh.ch
    - Giacomo Ghezzi   ghezzi@ifi.uzh.ch

Lots of BSc and MSc Project…

- **Information interaction, tailoring and filtering using interactive displays and Kinect**
- Answering developers questions
- IDE perspective of a change
- software quality indicators
- mobile applications quality
- …

…contact us if you are interested