Group Project Assessment

December 2nd 2015

Outline

- Assessment Criteria
- Week 1 schedule of Spring Term
- Writing the Report

Assessment and Feedback

Marking Scheme

- Executive Summary 5%
- Presentation 10%
- Group Collaboration and Management 20%
- Report 30%
- Technical Difficulty and Achievement 35%

Feedback

- Comments from supervisors and moderators will be available on CATE

Pass Mark

- Recall: MEng3 need 40% to progress to year 4

Group Project Submission

- January 11th 2016 at 12pm (not midnight !!!)
  - Final report due (submit via CATE)
- January 12th 2016 at 6pm
  - Final WebPA assessment due - compulsory
- January 20th 2016 at 5pm
  - Final submission as an archive (all code and documents, including report and any presentation slides)
  - Return equipment to CSG/TFs → Important!

Week 1 Schedule

- Monday, January 11th
  - 14:00: How to present your project and give feedback to others
  - 14:30 - 16:30: ‘Supervised’ rehearsals with feedback (2 or 3 ‘volunteer’ groups)
- Tuesday + Wednesday, January 12th and 13th → ‘Peer-evaluated’ rehearsal presentations
  - Your chance to practice among friends (no staff present) → Peer feedback

Week 1 Schedule

- Wednesday morning (January 13th – MSc), Thursday (January 14th) and Friday (January 15th) ➔ Group Project Presentations
  - 25 minutes per slot, including questions
  - Include approximately 10 minutes on technical details and 10 minutes demonstration

How to write the report

Use LaTeX!

Suggested Report Contents

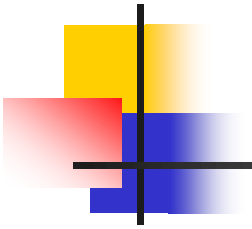
- Executive summary (COMPULSORY, max. 1 page) – see last slide
- Introduction
  - Set the scene (‘motivation’)
  - State the problem you are trying to solve (‘objective(s)’)
  - Summarise what you achieved (‘contributions’)
- Design & implementation
  - Detail your design (why did you do it this way?)
  - Summarise key implementation details (how did you do it? what tools did you use?)
- Evaluation
  - Summarise testing procedures (+ relevant testing results)
  - Evaluate your deliverables, e.g. in terms of performance, usability, usefulness… (how successful was the project?)
- Conclusion and future extensions
  - Say what you’ve concluded from doing the work and how you’d build on it
- Project management
  - Planning, group organisation, breakdown + task allocation etc.
- Bibliography
- Appendix (e.g. user guide)

General Tips

- Don’t leave it to the reader to guess what you mean, e.g.

The objective of this project is to embed the Flopsy system in Eclipse so that time scaling can be applied dynamically at run time.

The Flopsy system is a virtualised program execution environment that allows a programmer to explore the effects of speeding up or slowing down individual methods in a program. It is particularly useful when…<blah>. The objective of this project is to embed the Flopsy system in the Eclipse Integrated Development Environment (IDE) so that users can visualise the effect of such time scaling using existing Eclipse profiling plugins, such as Jprofiler.

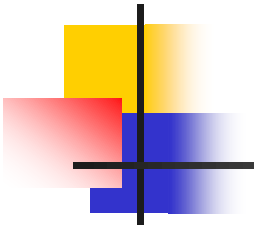


General Tips

- Detail your evaluation/experiments so someone else can understand what you did and reproduce your results, e.g.

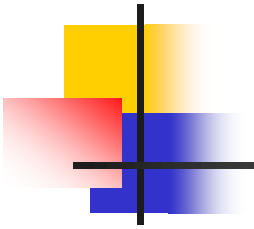
Fig 3 shows the results from a scalability experiment. This shows that…

To test scalability we ran the program on a varying number of cores (here 1-8), and measured both the throughput and execution time for a problem size of N=1024. The averages from 6 runs are shown in Fig. 3. This shows that…



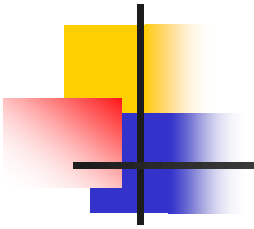
General Tips

- Use examples and diagrams extensively



Report

- Keep it short and to the point (e.g. circa 30-35 pages, excluding the Appendix)
- Project assessors may not have followed the project throughout and have a short time for presentation and demonstration, so impress them with your report ☺
- Top group project reports from last year:
  - http://www.doc.ic.ac.uk/lab/thirdyear/group-project/last-year



Executive summary

- A concise summary of your ‘product’
  - Your ‘elevator pitch’
- What is it? What does it do? Why would I want it? Why now? Etc.
  - No implementation details
  - No software engineering overview
  - No project management stuff