SAPM: Lecture 2
The Practical

Stuart Anderson

Overview

- Coursework is 25% of the final mark, exam is 75%
- The goal is to build a collection of software architecture case studies for the course.
- Groups of around 10 students will be responsible for developing an architecture case study.
- The goal is to develop a rich resource to support learning about SAPM

Contributions

- Peerwise questions (15%)
- Comments on other people's contributions (20%)
- Group participation (15%)
- Wiki page(s)(50%)
- Overall you should spend 25-30 hours on the practical.

Peerwise

Peerwise

Comments and Process

- You are expected to comment on other people's contributions. Comments will be graded and should be appropriate and helpful.

- Each group will develop a process to ensure the internal quality of their case study and links to the other case studies are appropriate. The quality of these processes will generate a group mark worth 15% of the overall grade.

Case Studies

- The idea of a case study is to demonstrate the use of the basic concepts of software architecture in a practical setting.

- This will involve you researching a deployed architecture and relating it to the concepts in SAPM and the other case studies as they are developed.

Case Studies

- Each person in the group will develop their own page(s) in the SAPM Wiki (You will have full access shortly).

- The main components are outlined on the next slide

Case Study Structure 1

- Page describing the internal and external review process (constructed by the whole group)
- Internal Review (overview of the case study plus a trail of review actions – mainly focussing on the coherence of the case study) – written by internal reviewer.
- External Review (how the case study relates to other case studies plus a trail of review actions mainly focussing on the relationship of the case study to other case studies. – written by external reviewer.

Case Study Structure 2

- Components and connectors used in the software architecture.
- The structure of the software architecture from different perspectives (static, dynamic, deployment…)
- The attributes (or non-functional properties the software architecture is intended to manage).
- Examples of the deployment of the software architecture in practical situations.
- Strengths and weaknesses of the software architecture.
- Comparison with other approaches to the problem the architecture is used to tackle.

Timeline

- Formative feedback deadline: 2 Mar 2017 – submissions before then will get feedback from me on the basis of the final grading scheme – focus is on how to improve. This is intended only to help you improve.

- Final deadline: 30 March 2017 – your contribution up to this deadline will be graded according to the grading scheme.

Length

- Wiki page contributions should be 500-1000 words roughly and can use diagrams or any other illustrative device (video, animation, audio, …). Think about accessibility of the material (e.g. Suppose someone is visually impaired – can they still access your material?)

Evaluation

- Each component (peerwise, comments, process, contribution) will be graded on four equally weighted factors (I will expand on this in the formal document):
- Relevance - to the course
- Understanding – does it illustrate your understanding?
- Quality – is it well-structured, accurate, comprehensible etc?
- Impact – how well you can relate your work to a practical context?