Case Study Presentation

Software Architecture

Semester 1, 2024

Richard Thomas & Brae Webb

Summary

In this assignment, you will be asked to demonstrate your ability to *understand*, *communicate*, and *critique* an architecture of an existing software project. You will

1. choose a suitable open source software project that has non-trivial functionality and architecture;
2. present the key information about the architecture of your selected project; and
3. critique the architecture.

1 Introduction

The digital world relies heavily on open source software, as seen by the log4j vulnerability[1]. Fortunately, open source developers often maintain high quality documentation for the users of their projects. Unfortunately however, many open source projects do not maintain the same high quality documentation for the architecture of their software projects. This can cause difficulty for developers who want to contribute to the project, but first need to understand it.

In your presentation, you have the chance to right this wrong. You are to find an open source software project with a sufficiently complex architecture and describe it. You may choose to share your results with the project developers. You are encouraged to do this, as the perspective of a newcomer to a project is often invaluable to the seasoned developers.

Before looking for projects, read some of the architecture documentation[2] written by students at TU Delft. You may also find it useful to read through one or more of the architecture descriptions in either volume of *The Architecture of Open Source Applications*[3].

You will give a presentation describing the architecture of the project you select. The intent is to give everyone in the course a broader view of how software architectures are used to solve problems. Your presentation should take advantage of what you learn throughout this course. You are to critique the architecture, discussing how well it meets the projects goals.

2 Selecting a Project

A list of potential open source software projects is available in the discussion board. You may propose another project to use for your presentation. It must still be an open source project and must be of sufficient size and complexity to have an interesting software architecture. It may not be a project for which a fairly complete set of architectural design documentation exists (e.g. one described by the TU Delft students or in one of the *The Architecture of Open Source Applications* books. A monolith is a possible choice, as long as the internal architecture is well structured and non-trivial.

[1] https://www.cisa.gov/news-events/news/apache-log4j-vulnerability-guidance
[2] https://delftswa.gitbooks.io/desosa2016/content/
[3] https://aosabook.org/en/index.html

3 Presentation Content

You are free to structure your presentation however you wish, though you should use some form of slides to support the delivery of information. Your presentation needs to deliver the following content.

Title Slide Name of the software project, and your name and student number.

Introduction Describe the software project, explaining the its key functionality and target users.

ASRs Describe the Architecturally Significant Requirements (ASR) of most importance to the project.

Context Provide an overview of the software system's context and its external dependencies.

Architecture Describe the software's architecture.

Critique Analyse the software's architecture, describing how well it delivers its ASRs.

Conclusion Highlight the key points or lessons learnt about the software's architecture.

Your presentation should introduce the software project. Give an elevator pitch style summary of what problem the project solves and its key features. Describe which ASRs and, in particular, the quality attributes you think are most important for the project, and why. Describe the project's software architecture using appropriate views [1]. Critique the software architecture, highlighting how well it supports delivering the project's architecturally significant requirements.

You must use the C4 modelling notation [1] [2] to describe the software architecture. You may supplement the C4 diagrams with other diagrams to help describe the architecture. For example, you may use UML use case, class, or sequence diagrams [1] [3] to describe system requirements or details of how the architectural design works. Other diagrams may also be used, if they clarify aspects of your C4 model. Any diagrams obtained from other sources (e.g. the project website) must be cited.

You should describe any security risks inherent in the software architecture. Your critique should evaluate and discuss what security design principles appear to have been followed in the design of the software and how well they guard against the security risks.

Your description of the software architecture should cover all of its important aspects. You are not expected to get down to the level of describing the detailed design of the software. You should not need to provide class or dynamic diagrams for the entire system. You may need to provide a small number of class or dynamic diagrams to highlight important features supported by the architecture. For example, a class diagram showing a plug-in API and a dynamic diagram of how the application uses a plug-in, may be informative.

Your audience is other students in this course. You may assume the audience has knowledge of the course content, though you should not assume they are familiar with the project you are describing.

3.1 Citations & References

You may use references in your presentation to support points you are making. These must be cited and referenced using the IEEE referencing style[4]. The final slide(s) of your presentation should include the references to any cited material. You should display the reference slide(s) for about 3 seconds at the end of your presentation. You are not required to speak to the reference slides, aside from possibly thanking your audience for listening and stating these are your references.

[4] https://libraryguides.vu.edu.au/ieeereferencing/gettingstarted

4 Presentation

Presentations will take place in your practical, and possibly case study, class sessions during weeks 10 to 13. You will have a **maximum** of eight minutes for your presentation, plus three minutes for questions. There is no minimum time required for your presentation, it is up to you to determine when you have described all relevant information about the software architecture within your eight minute limit.

If your presentation exceeds eight minutes, the marker will ask you to stop your presentation. No content of your presentation past eight minutes will be marked.

As a presenter, you should not read a script. You may wish to write a script to prepare for the presentation but should not read it during the presentation. You may make use of notes during the presentation but you should only quickly glance at your notes to keep yourself on track. You should not be constantly referring to notes. You should try to maintain eye contact with your audience, rather than focussing on your notes or slides.

A schedule of presentation time slots will be organised after the mid-semester break. You will be allocated a time and class session in which you will present. Please inform the course coordinator of any constraints you may have regarding presentation times ***before*** Easter.

5 Identity Verification

The presentation is an identity verified assignment. You must make your presentation in-person. At the start of your presentation you must show your UQ student card to one of the markers at your session. Like in an exam situation, if you have lost your student card you must obtain a temporary identity verification document from the UQ student centre *before* your presentation.

The marked result of your presentation will be used to determine any caps applied to your grade. (That means failing the presentation because you did not submit the required number of peer evaluations will ***not*** affect the mark used to determine a final grade cap.) The first slide of your presentation **must** contain your full name, as recorded in UQ's student enrolment system, and full 8-digit student number.

6 Submission

There are three components that make up your assessable content for the presentation. These are the slides you use for your presentation, the presentation itself, and your evaluation of other students' presentations.

6.1 Slides

The slides for your presentation are to be submitted as a PDF file to a link provided on BlackBoard. Your slides are due at 16:00 on Tuesday of the week in which you are scheduled to make your presentation. Late submission of your slides will result in a penalty of 1 grade per 24 hour period that they are late. Regardless of any penalty applied to the presentation, *even* if the penalty is a failing grade, you ***must*** still make your presentation in your allocated timeslot.

6.2 Presentation

The presentations will take place in the practical, and possibly case study, sessions during weeks 10 to 13. You will be allocated a week in which you are to make your presentation. Your presentation is to use the slides you submit to BlackBoard.

If you do not deliver your presentation, your final grade will be capped at a failing grade. If you are unable to attend your session to give your presentation due to exceptional circumstances, you may apply to defer your presentation to another date. You are not able to defer a deferred presentation.

6.3 Peer Evaluation

You are expected to attend all presentations. You are required to submit an evaluation of each presentation you observe. Submission of *meaningful* feedback for at least **75%** of the presentations in your class sessions is required to obtain a passing grade or higher for the presentation assessment.

An online form will be provided for you to submit your evaluation for each presentation. You must submit your evaluation of each presentation separately in order for the system to record all of your evaluations.

If you are unable to attend a practical session due to exceptional circumstances, and miss viewing several presentations, you may apply for a modified limit on the number of presentations you must evaluate.

7 Academic Integrity

As this is a higher-level course, you are expected to be familiar with the importance of academic integrity in general, and the details of UQ's rules. If you need a reminder, review the Academic Integrity Modules[5]. Submissions will be checked to ensure that the work submitted is not plagiarised. If you have quoted or paraphrased any material from another source, it must be correctly cited and referenced[6]. Use the IEEE referencing style[7] for citations and your bibliography.

Note that text generated by an AI tool, such as Chat GPT, is based on text from the Internet. Consequently all text, whether written on slides or spoken during a presentation, that was generated by an AI tool must be cited.

Uncited or unreferenced material will be treated as not being your own work. Extensive quotation or minor rephrasing of material from cited sources should be avoided. Significant amounts of cited material from other sources, even if paraphrased, will be considered to be of no academic merit. In all cases, any material that you cite must support the arguments and points that you are making in your presentation.

References

[1] R. Thomas and B. Webb, "Architectural views," February 2023. https://csse6400.uqcloud.net/handouts/views.pdf.

[2] S. Brown, *The C4 Model for Visualising Software Architecture*. Leanpub, Feb 2023. https://leanpub.com/visualising-software-architecture.

[3] *Unified Modeling Language*. OMG, 2.5.1 ed., December 2017. https://www.uml.org/.

[5] https://web.library.uq.edu.au/library-services/it/learnuq-blackboard-help/academic-integrity-modules

[6] https://guides.library.uq.edu.au/referencing

[7] https://libraryguides.vu.edu.au/ieeereferencing/gettingstarted

Presentation Criteria

Criteria	Standard						
	Exceptional (7)	**Advanced (6)**	**Proficient (5)**	**Functional (4)**	**Developing (3)**	**Little Evidence (2)**	**No Evidence (1)**
Context 10%	Project is introduced clearly and well situated within its context, providing an excellent starting point to understand the system.	Project is introduced clearly with good contextual information, providing a good starting point to understand the system.	Project is introduced well with a good overview of its context, providing a clear but basic overview of the system.	Project is introduced fairly well with some contextual information, providing a comprehensible overview of the system.	Project scope & general context are fairly clear, providing a general overview of the system.	Project scope & context are not clear, providing a poor overview of the system.	Project scope & context are confusing, providing an inaccurate overview of the system.
ASRs 10%	ASRs are clearly described, well justified, clearly of high importance, and all will influence architecture decisions.	ASRs are clearly described, fairly well justified, seemingly of high importance, and all are likely to influence architecture decisions.	Most ASRs are well described but a few justifications are a little weak. Most are important and likely to influence architecture decisions.	Some ASRs are well described but a few justifications are weak. Most are important and likely to influence architecture decisions.	Some ASRs are fairly well described but some justifications are weak. Some are important and likely to influence architecture decisions.	Most ASRs are poorly described or poorly justified. Few are important or likely to influence architecture decisions.	Most ASRs are poorly described and poorly justified. Very few are important or likely to influence architecture decisions.
Architecture Diagrams 20%	All diagrams are easy to comprehend, convey important information, and enhance the presentation.	Most diagrams are easy to comprehend, convey important information, and are used well in the presentation.	Most diagrams are comprehensible, convey useful information, and are used well in the presentation.	Most diagrams are comprehensible, convey useful information, and are connected to the presentation.	Most diagrams are comprehensible, convey some useful information, and are mostly connected to the presentation.	Some diagrams are incomprehensible, do not convey useful information, or are disconnected from the presentation.	Most diagrams are incomprehensible, do not convey useful information, or are disconnected from the presentation.
Architecture 25%	Description is clear, complete, concise, informative and at an appropriate level of detail, resulting in an excellent, coherent understanding of the entire architecture.	Description is clear, seemingly complete, informative and at an appropriate level of detail, resulting in a good coherent understanding of the entire architecture.	Description is mostly clear, informative and at an appropriate level of detail, resulting in a good understanding of the architecture structure.	Description is mostly clear, informative and at an appropriate level of detail, resulting in a good overview of the architecture structure.	At times the architecture description is not clear, informative or at an appropriate level of detail, resulting in a slightly vague overview of the architecture structure.	Architecture description is not clear, informative or at an appropriate level of detail, resulting in an incomplete understanding of the architecture structure.	Architecture description is not clear, informative or at an appropriate level of detail, resulting in an incorrect understanding of the architecture structure.
Critique 25%	Clear, accurate, insightful & concise critique, demonstrating in-depth knowledge of the entire architecture.	Clear, accurate & fairly insightful critique, demonstrating fairly in-depth knowledge of the entire architecture.	Mostly clear, accurate, and at times insightful critique, demonstrating good knowledge of the architecture structure.	Mostly clear and accurate critique, demonstrating fairly good knowledge of the architecture structure.	At times critique is not clear or is inaccurate, demonstrating some deficiencies in understanding the architecture structure.	Unclear or inaccurate critique, demonstrating incomplete understanding of the architecture structure.	Confusing or very inaccurate critique, demonstrating poor understanding of the architecture structure.
Presentation 10%	Presentation is well paced and delivered fluently. Information is logically sequenced, with clear objectives making it very easy to follow.	Presentation is well paced and delivered clearly. Information is logically sequenced, with some clear objectives making it easy to follow.	Presentation is mostly well paced and delivered clearly. Information is logically sequenced, with signposting guiding audience through presentation.	Presentation pace is a little inconsistent or delivery is occasionally unclear. Information is logically sequenced allowing audience to follow presentation fairly well.	Presentation pace is inconsistent or delivery is sometimes unclear. Information is not always logically sequenced, distracting audience from presentation flow.	Presentation pace is inconsistent or delivery is unclear. Information is not logically sequenced, and planned progression was not clear to audience.	Presentation pace is inconsistent and delivery is unclear. Information is poorly sequenced, confusing audience.