Project Revisions Overview

Initial Project Revision - preliminary insights over chosen open-source software architecture

- On Nov. 9th we will have our 1st project revision!
- Prepare a presentation:
    - 20 mins including questions
    - You will HAVE TO expose, at least:
        - (a) Preliminary thoughts and insights over SA, The architecture and organisational structure of the community and product;

        - (b) The properties and visible features of the architecture (Overview of Open Issues, focus on arch. Issues (if any);

        - (c) The limitations and shortcomings of that product, how to install/interact with it (e.g., Challenges in community participation);

    - You MAY want to use the concepts and notations we used up to this point;

Intermediate Project Revision - draft of software architecture documentation

- On Nov. 27th we will have our intermediate project revision!

- Prepare a presentation:
    - 20 mins including questions

- You will HAVE TO expose, at least:

    - The architectural structures that you could see – this step is supposed to be elaborated using diagrams, notations, models anything you can provide or produce using either support tooling or your own experience with respect to the community at hand;

    - The decisions that were taken over the time of your study and the properties that these decisions imply and your rationale over these – for example, if there are any relevant architecture or similar issues that were closed and what is the rationale and fixes that are connected to closing; your objective in this case is to document the decision and highlight the architectural change/refactoring;

    - The style and provide a rationale for that style or elaborate if/how other styles apply – provide an overview of possible other options for styles and other similarities with respect to related middleware;

    - An overview of the technologies involved and notations that describe the software architecture – for example you can focus on the infrastructure-as-code which comes with the middleware under your analysis as well as the details with which that middleware is deployed;

- You MAY need/want to use the concepts and notations we have studied up to this point;
- You MAY need/want to use an architectural reverse engineering tool (e.g., MoDisco, Architexa, VisualParadigm, etc.);

Advanced Project Revision - if it were my product… what would I do with it!

- On Monday Jan. 15th we will have our last project revision!

- Prepare a presentation:
    - **20-30 mins including questions**

- Applying all the insights that you learned across the course, You will HAVE TO at least:
    - Evaluate the architecture that you have studied, contributed to and described with respect to what you would do to it as a software architect (e.g., exercise ATAM or ADD);

    - What would you evolve/change and how (i.e., prepare an Architectural Improvement Plan);

    - What feature would you expose more prominently and how;

    - What property would you analyse and how;

    - What decisions would you revise and why;

    - Once revised, how would you change in the architecture to reflect change;

    - What other technologies would you adopt/embed in the product and why;

- You MUST use the concepts and notations we have studied up to this point and represent them in PDF hand-in of (at most) 10 pages, including diagrams, their description, views, viewpoints, notations, etc.;

NOTE: The hand-in will be evaluated against the criteria above for correctness (everything is correctly mapped in the hand-in and reasonably argued) and completeness (every item of the above list is addressed exhaustively)

- You MAY need/want to use an architectural reverse engineering tool (e.g., MoDisco, Architexa, VisualParadigm, etc.);

NOTE on Evaluation Extra’s

- *Fool-of-a-Too(l)k(it)!* ***+1****:* bug-hunting + patching of open-source project issue, documenting the process;
- *Bilbo Bug-gins!* ***+2****:* bug-hunting + patching of open-source project software architecture issue, documenting the process;
- *Gandalfeat!* ***+3****:* address a feature request, implement resolution, document the process;
- *AraGone!* ***+4****:* propose architecture change, implement & merge change, document the process;
- *FroDone!* ***+5****:* re-documentation of entire architecture through reverse engineering + document the process;

All of the above need to be properly documented in terms of:

- “Before-scenario” – describe how the SA was before you acted and how you addressed the extension (i.e., the process you followed);

- “Current-status” – report the status of your extension, e.g., you made a pull-request and you are pending approval;

- “End-status” – how did the SA change with respect to your extension;

The correct, complete, and well-argued documentation of the above will correspond to the increase in final mark equivalent to the performed extension.