

LSD Exam report

This report describes both the process of developing your system and the process of keeping it operational during the semester. It will form the basis for your oral exam, and will be a part of our assessment of your project.

The report contains three parts: a part for the description of your requirements, system architecture, system design and development process, a part for your experience with the maintenance and your SLAs and finally a discussion of the project as a whole. Apart from these three parts we expect you to write:

- A short introduction that briefly sets the stage for the project.
- A conclusion that summarises the two parts and evaluates on the project as a whole.

Hand-in

The report must be between 5 and 10 pages in total. One page is equal to 2400 characters (spaces included). You are welcome not to write the full 10 pages if you think the report covers the requirements.

The report must be handed in no later than the **16th of December at 23:59**.

1. Requirements, architecture, design and process

This section describes the requirements of the project, the architecture that you have chosen, the design of the actual components and the process you undertook to build them in the end.

1.1. System requirements

This section should contain an elaborate description of the requirements for the project. This includes the scope of the Hackernews clone (what should it be able to do / what should it not be able to do).

1.2. Development process

In this part you should show off by telling us all you know about software development processes and describe which concepts you used to structure your development.

1.3. Software architecture

In this section you illustrate and describe the architecture of your Hackernews clone. That is, you describe how your system is structured and how the different parts interact and communicate with each other.

1.4. Software design

Here you should sketch your thoughts on the software design *before* you started implementing the system. This includes describing the technical concerns you had about the system before you started development, together with all the technical components you came up with to fix these concerns and meet the requirements.

1.5. Software implementation

This section should describe your actual implementation. Mainly how well you followed the requirements, process and software design you began with. If your system changed during this phase you should summarise the unexpected events/problems and explain how you solved them.

2. Maintenance and SLA status

This section describes the process of maintaining the software over time, starting from the hand-over to the shutting down of your system. The section should be written from the viewpoint of the operator, *not* the developers.

2.1. Hand-over

In this part, you should describe the hand-over of the system you were operating. Specifically you should comment on the quality of the documentation you received and whether you felt well equipped to maintaining the system.

2.2. Service-level agreement

Here you should include the service level agreement you entered together with the group you were operating, including the metric itself and the threshold you agreed upon for that metric. If you had any disagreements about the SLA you should describe them and explain how you found an agreement.

2.3. Maintenance and reliability

This part should contain a description on how you experienced the actual operation. Explain how you actually monitored the system to ensure that the SLA was upheld, and describe any incidents you experienced that broke (or could potentially break) the SLA. Remember to include documentation for each incident! Finally you should conclude how well the developers responded to your issues and conclude on how reliable the system was overall.

3. Discussion

3.1. Technical discussion

This part summarises both the first and second part of the report by giving an overview of the good and bad parts of the whole semester project. Be critical and honest.

3.2. Group work reflection & Lessons learned

Give a short reflection on what were the three most important things you learned during the project. The lessons learned are with regards to both, what worked well and what worked not well. These reflections can cover anything from the sections above. That is, development process, architectural and design decisions, implementation, maintenance, etc. If you chose to use roles (project manager, architect, devops etc.) you should use those to reflect on whether they improved the process or not.

Additionally, focus on both, your work as developers as well as operators.

Hand-in procedure

- You create the project report as a Markdown (.md) file in a Github repository.
 - OBS You create a markdown document. No MS Word file, no PDF, no other format than .md
- At latest on 21 December 2018 23:59 your group will upload a link to the project report markdown document on Moodle.