0x19. Postmortem

DevOpsSysAdmin

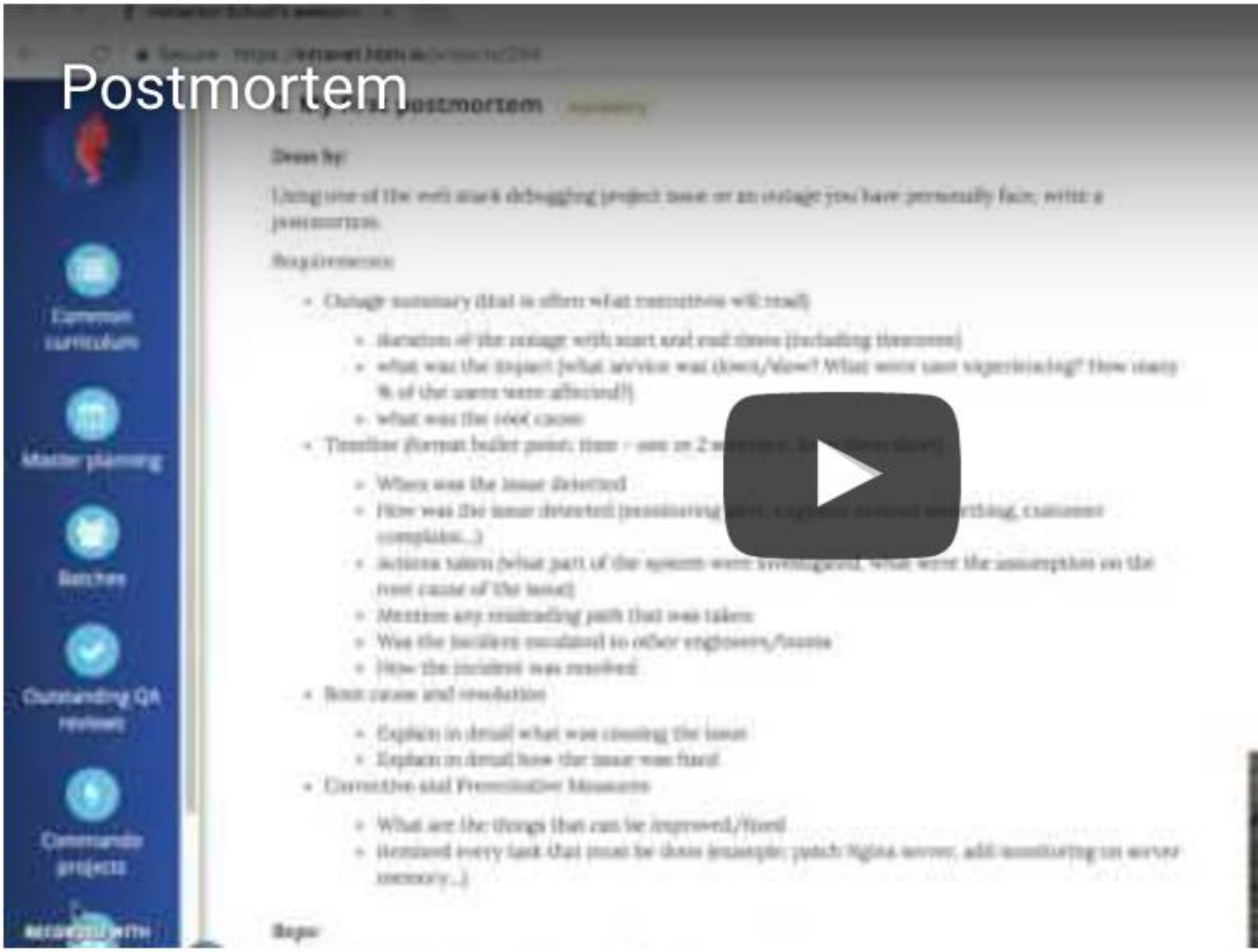
- By: Sylvain Kalache
- Weight: 1
- Ongoing second chance project - started Oct 31, 2022 6:00 AM, must end by Nov 12, 2022 6:00 AM
- **Manual QA review must be done** (request it when you are done with the project)

Concepts

For this project, we expect you to look at this concept:

- On-call

Background Context



Any software system will eventually fail, and that failure can come stem from a wide range of possible factors: bugs, traffic spikes, security issues, hardware failures, natural disasters, human error… Failing is normal and failing is actually a great opportunity to learn and improve. Any great Software Engineer must learn from his/her mistakes to make sure that they won't happen again. Failing is fine, but failing twice because of the same issue is not.

A postmortem is a tool widely used in the tech industry. After any outage, the team(s) in charge of the system will write a summary that has 2 main goals:

- To provide the rest of the company's employees easy access to information detailing the cause of the outage. Often outages can have a huge impact on a company, so managers and executives have to understand what happened and how it will impact their work.
- And to ensure that the root cause(s) of the outage has been discovered and that measures are taken to make sure it will be fixed.

Resources

Read or watch:

- Incident Report, also referred to as a Postmortem
- How to run a Postmortem

More Info

Manual QA Review

It is your responsibility to request a review for your postmortem from a peer before the project's deadline. If no peers have been reviewed, you should request a review from a TA or staff member.

Tasks

0. My first postmortem

mandatory

Using one of the web stack debugging project issue or an outage you have personally face, write a postmortem. Most of you will never have faced an outage, so just get creative and invent your own :)

Requirements:

- Issue Summary (that is often what executives will read) must contain:
  - duration of the outage with start and end times (including timezone)
  - what was the impact (what service was down/slow? What were user experiencing? How many % of the users were affected?)
  - what was the root cause
- Timeline (format bullet point, format: `time` - `keep it short, 1 or 2 sentences`) must contain:
  - when was the issue detected
  - how was the issue detected (monitoring alert, an engineer noticed something, a customer complained…)
  - actions taken (what parts of the system were investigated, what were the assumption on the root cause of the issue)
  - misleading investigation/debugging paths that were taken
  - which team/individuals was the incident escalated to
  - how the incident was resolved
- Root cause and resolution must contain:
  - explain in detail what was causing the issue
  - explain in detail how the issue was fixed
- Corrective and preventative measures must contain:
  - what are the things that can be improved/fixed (broadly speaking)
  - a list of tasks to address the issue (be very specific, like a TODO, example: patch Nginx server, add monitoring on server memory…)
- Be brief and straight to the point, between 400 to 600 words

While postmortem format can vary, stick to this one so that you can get properly reviewed by your peers.

Please, remember that these blogs must be written in English to further your technical ability in a variety of settings.

Add URLs here:

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x19-postmortem`
- File: `README.md`