

# DevOps 2025

---

IT University of Copenhagen

## Team Sad People

---

### People

- *Gábor Tódor* - gato@itu.dk
- *Kálmy Zalán* - zaka@itu.dk
- *Nicklas Koch Rasmussen* - nicra@itu.dk
- *Nicolai Grymer* - ngry@itu.dk
- *Sebastian Andersen* - seaa@itu.dk

### Subject

- **Course manager:** *Helge Pfeiffer* - ropf@itu.dk
- **Course Code:** KSDSESM1KU

## Introduction

---

## System

---

A description and illustration of the:

**[ seb/nick ] Design and architecture of your *ITU-MiniTwit* systems**

---

**[Nic] All dependencies of your *ITU-MiniTwit* systems on all levels of abstraction and development stages. That is, list and briefly describe all technologies and tools you applied and depend on.**

---

**[Nic] Important interactions of subsystems.**

---

For example, via an illustrative UML Sequence diagram that shows the flow of information through your system from user request in the browser, over all subsystems, hitting the database, and a response that is returned to the user.

Similarly, another illustrative sequence diagram that shows how requests from the simulator traverse your system.

**[G] Describe the current state of your systems, for example using results of static analysis and quality assessments.**

---

**[ALL] MSc students should argue for the choice of technologies and decisions for at least all cases for which we asked you to do so in the tasks at the end of each session.**

---

## **Process**

---

This perspective should clarify how code or other artifacts come from idea into the running system and everything that happens on the way.

In particular, the following descriptions should be included:

**[Nic] A complete description of stages and tools included in the CI/CD chains, including deployment and release of your systems.**

---

**[Z/G] How do you monitor your systems and what precisely do you monitor?**

---

**[Z/G] What do you log in your systems and how do you aggregate logs?**

---

**[Nic] Brief results of the security assessment and brief description of how did you harden the security of your system based on the analysis.**

---

**[Nic] TLS - 2FA - Port forwarding - DB encryption - ORM**

**[Seb/Nick] Applied strategy for scaling and upgrades.**

---

**[Nic] In case you have used AI-assistants during your project briefly explain which system(s) you used during the project and reflect how it supported or hindered your process.**

---

## **Reflection**

---

**Evolution and refactoring**

---

Finishing a sprint and adding a new feature

**[G/Z] ELK logging resource heavy + too many fields + all fields were indexed**

**[G/Z] Filebeat on all swarm nodes**

**[G/Z] Logging deployment: put config images in**

**[Nic] Migrating from SQLite to PostgreSQL**

**[Nic] Transition from docker compose to docker swarm (networking problems).**

**[Seb/Nick] Docker compose versioning problem (moving to stack)**

**[G/Z] scp files onto server and then deployment (Discuss ups/downs)**

- You can destroy the prod environment with wrong files/wrong docker compose

**[G/Z] Transition from config files to docker images (Tagging docker containers)**

**[Seb/Nick] Large amount of features clogging up in staging (Impossible to migrate to production)**

## Operation

---

Keep the system running

**[Nic] Database logical replication resulting in db crash**

**[G/Z] Log overflow problem. Access denied to machine. Massive clutch**

**[Nic] Backup strategy (cron job every three hours)**

## Maintenance

---

Keep system up to date and fix bugs

**[Seb/Nick] Stale ReadMe.md throughout project**

**[Seb/Nick] Returning wrong statuscode (Misalignment with simulation)**

- Thanks to running simulator in the CI/CD pipeline we found this

**[Nic] Upgrading to NGINX, setting up ufw, moving to domain**

**[Nic] Simulator IP protection stopped sim access (causing errors)**

## **Style of work**

---

Reflect and describe what was the "DevOps" style of your work.

**[Seb/Nick] Reflect on the workflow. Extensive Friday meeting. Split work into three groups**

**[Seb/Nick] Development environemnt: local => branch => staging => production**

**[Seb/Nick] Repo settings. Workflows on merge. Require 1 team member on pull requests.**

**[Seb/Nick] Running simulator in workflows**

That's it folks!

# **REMEMBER TO REMOVE THIS**

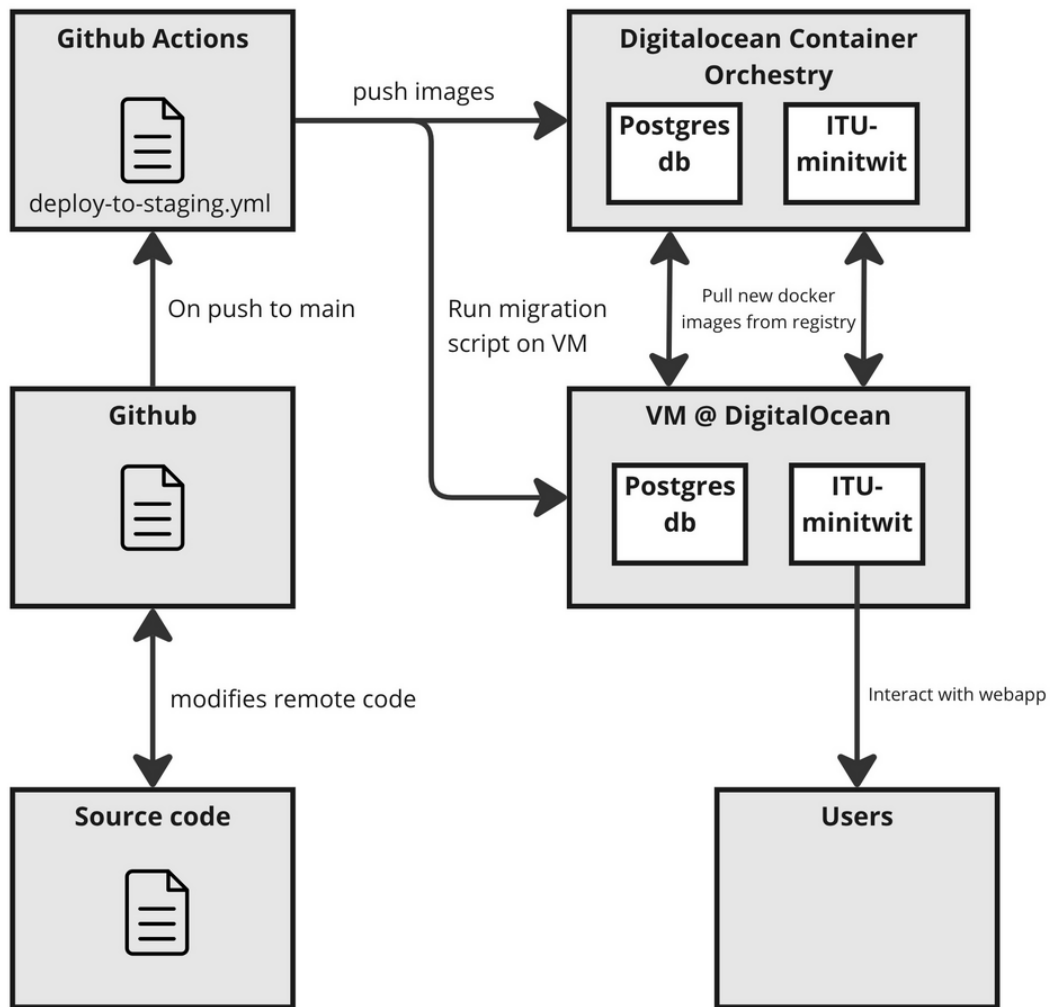
---

## **THIS IS JUST AN EXAMPLE**

---

## **THIS IS JUST AN EXAMPLE**

---



hello a LOL fdsa