

Week 01: Introduction to Seminars

Welcome to the course!

Agenda

- Tutor introduction
- Course info
- Git Basics
- Gitflow
- Setup Gitlab & IDE
- Hands on: Data modeling

Who (am/are) (I/we)?

Let me introduce myself

Course info: Basics

- Seminars are compulsory (max. 4 unexcused seminars)
- Attendance is checked using ROPOTs at the start of each seminar
- Special Lukáš Grolig demo seminars
- Source of truth: [Syllabus](#) and [Gitlab](#)
- Communication: [Discord](#)
 - Help support almost 24/7
 - We ❤️ to help you

Deadlines

For iterations: 72h hours before seminar (soft deadline – each tutor can have their own policy)

For team projects: Date of your presentation (more information later during the semester)

Course info: Evaluation

- Up to **33 points from iterations** (for completing assignments of the semestral project with the best effort and clean code).
- Up to **42 points for your team project** (for creating a complex solution, dividing work, and collaborating with others).
- Up to **25 points for exams** (the final ROPOT contains all the topics from the semester)
- Up to **5.2 soft points** for seminar ROPOTs - these apply after you gain at least 70 points from iterations, team project, and exams to help you get a better grade.

Grade	Points
A	100-94
B	93-88
C	87-82
D	81-76
E	75-70
F	69-0
Z	100-60
N	59-0

Let's revise data modeling in ERD

- Entity, Primary key, Foreign key, Relationship
- Types of ERD: Conceptual, Logical, Physical models

PlantUML

```
@startuml lab01-diagram

hide circle

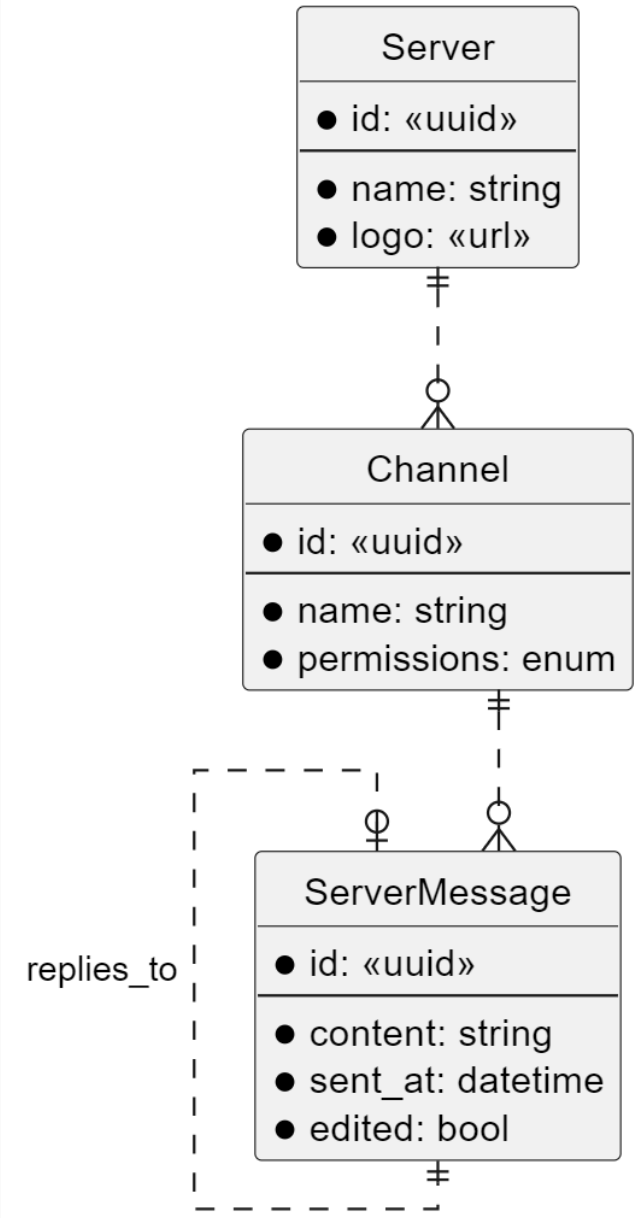
skinparam Linetype ortho

entity ServerMessage {
    * id: <<uuid>>
    ---
    * content: string
    * sent_at: datetime
    * edited: bool
}

entity Channel {
    * id: <<uuid>>
    ---
    * name: string
    * permissions: enum
}

entity Server {
    * id: <<uuid>>
    ---
    * name: string
    * logo: <<url>>
}

ServerMessage |o..|| ServerMessage: replies_to
Server ||..o{ Channel
Channel ||..o{ ServerMessage
@enduml
```



Git: Setup

1. Install git (Depends on your OS)

Using a package manager:

```
apt install git      # ubuntu/debian  
brew install git     # mac  
winget install Git   # windows package manager
```

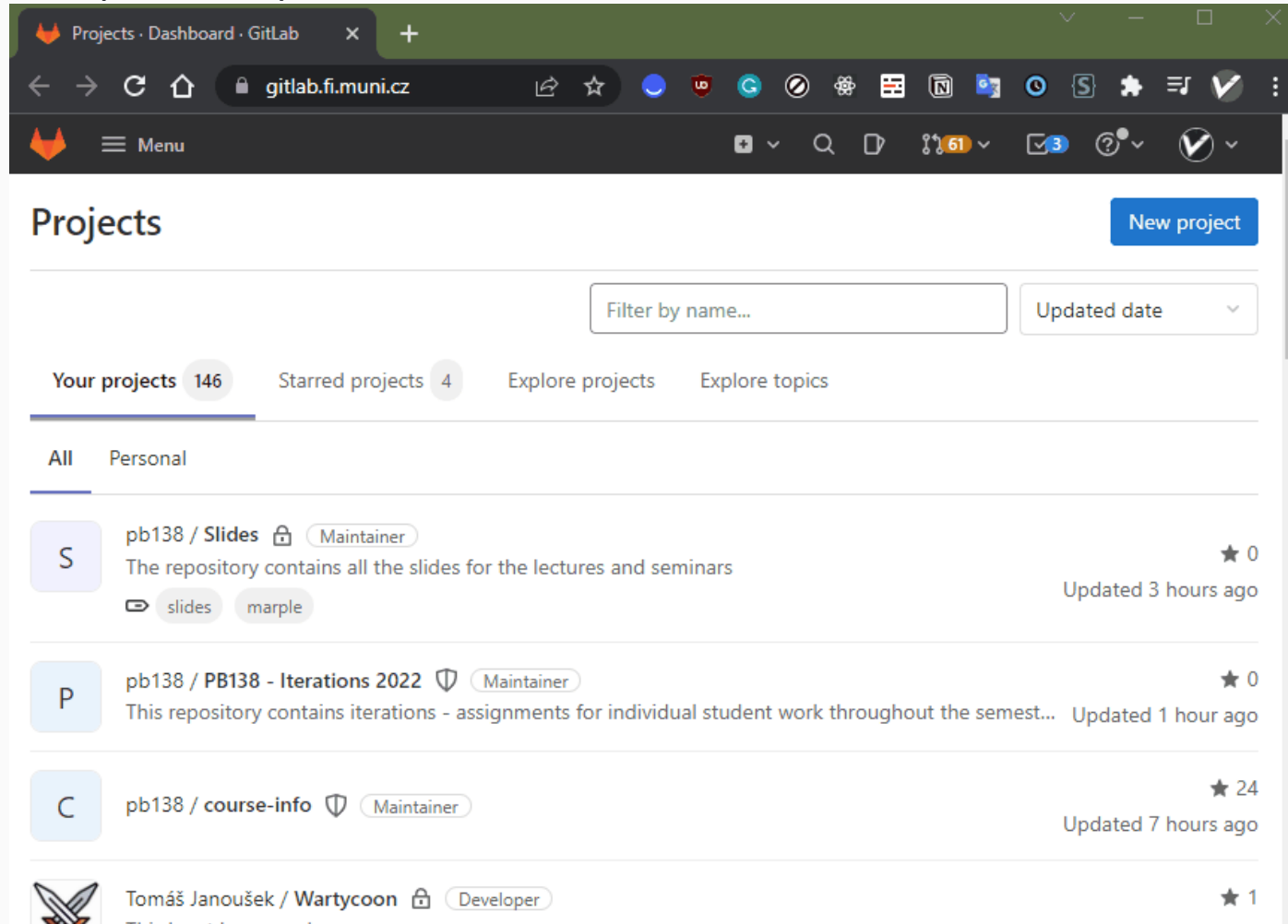
Or you can use the [Git website](#).

2. Setup keys

```
ssh-keygen -o -a 100 -t ed25519 -f ~/.ssh/id_muni -C "xuser@fi.muni.cz"
```

Git: Setup

3. Add public key to Gitlab



Git: Basics

Commit in [Conventional commits](#)

```
git config --global core.excludesFile "**/node_modules"  
git config --global user.name xuser  
git config --global user.email xuser@fi.muni.cz
```

```
git clone <url> # Clones the repository  
git status # Show status of added, removed files  
git checkout -b submit-00 # Switch to new branch  
git commit -am "feat: added erd" # Commit all tracked files  
git push origin submit-00 # Push committed to remote branch
```

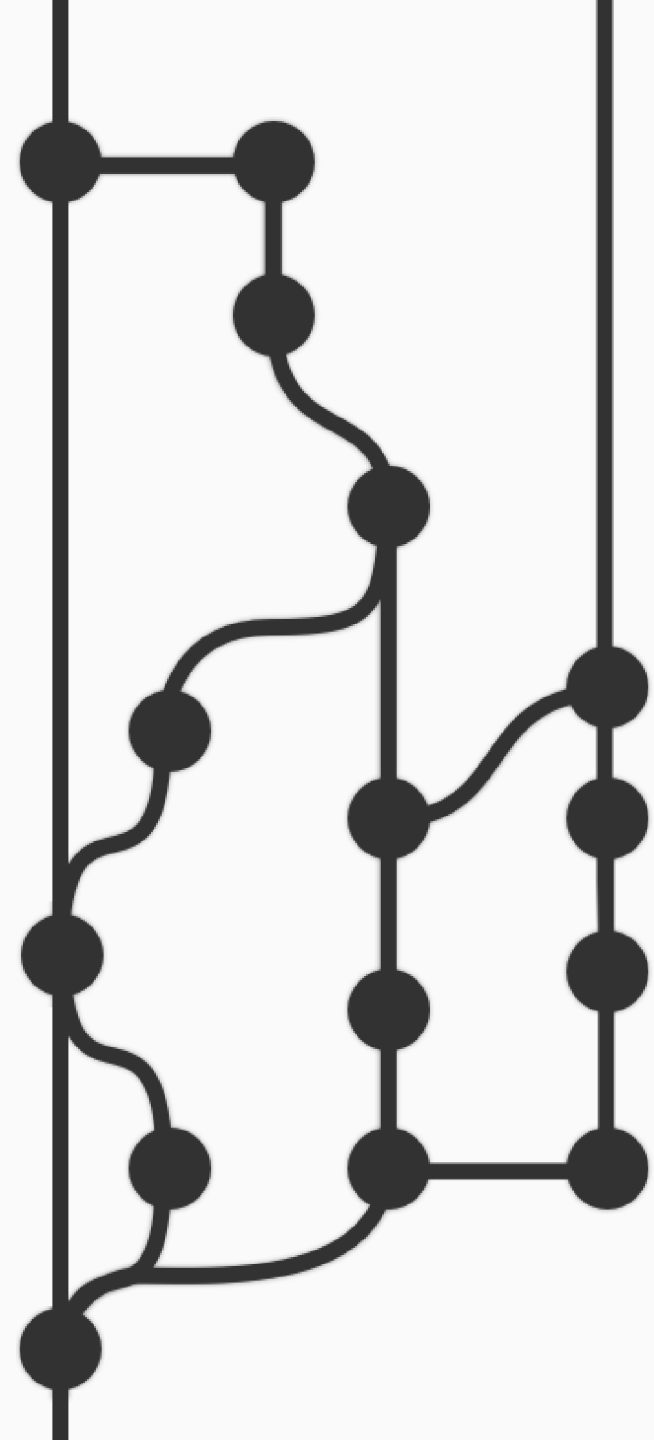
[Cheatsheet](#)

Git: Gitflow

- Starts from the main branch
- Feature branches contain new features, additions
- Master/Main is stable (tagged)

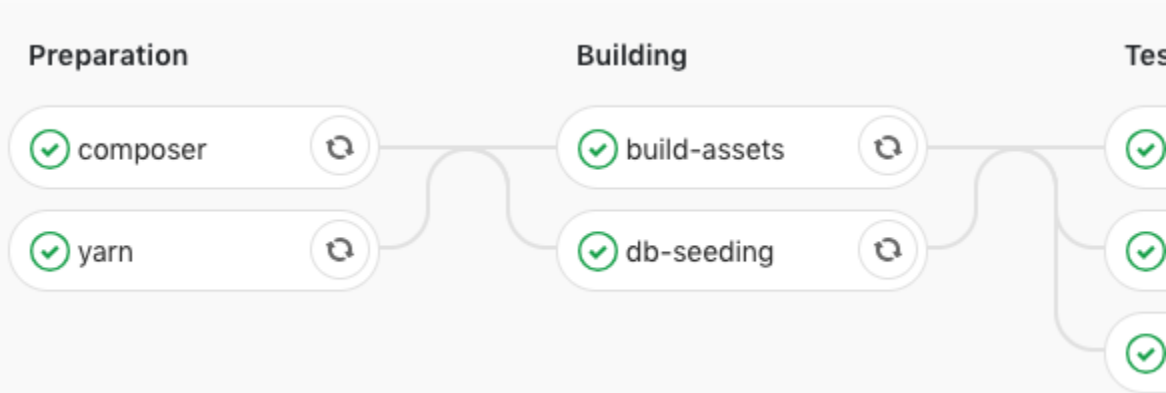
Note: You'll most likely work in trunk based development (it ignores the develop branch)

[In-depth explanation](#)



Git: Merge requests / Pull request

- MR/PR the way you check source code changes into a main branch
- Before pushing code, check [common mistakes](#)
- CI is not your enemy
- We require you to discuss changes



GitLab Community Edition

Project

Repository

Issues 11,511

Merge Requests 668

CI / CD

Operations

Snippets

Settings

Open Opened 44 minutes ago by Mayra Cabrera 2 of 9 tasks completed

Edit Close merge request

Resolve "Deploy Tokens failed to clone LFS repository"

What does this MR do?

Allow `DeployTokens` to be utilized to clone LFS repositories

Why was this MR needed?

When downloading LFS repositories through git cloning and deploy tokens as an authorization method, the cloning fails. This MR prevents that

Screenshots (if relevant)

Does this MR meet the acceptance criteria?

- Changelog entry added, if necessary
- Tests added for this feature/bug
 - Conforms to the code review guidelines
 - Has been reviewed by a Backend maintainer
 - Conforms to the merge request performance guidelines
 - Conforms to the style guides
 - Conforms to the database guides
 - If you have multiple commits, please combine them into a few logically organized commits by squashing them
 - Internationalization required/considered
 - End-to-end tests pass (`package-and-qd` manual pipeline job)

What are the relevant issue numbers?

Closes #46869

Edited 38 minutes ago by Mayra Cabrera

Request to merge 46869-deploy-tokens-failed-to-clone-lfs-repository into master

Open in Web IDE Check out branch

Pipeline #26130830 running for @a3350d9 on 46869-deploy-tokens-failed-to-clone-lfs-repository Coverage 56.42%

Approve Requires 1 more approval

Failed to load codeclimate report

Security scanning detected no vulnerabilities

Merge when pipeline succeeds Remove source branch Modify commit message

Closes #46869

You can merge this merge request manually using the command line

Discussion 3 Commits 1 Pipelines 2 Changes 5

- Mayra Cabrera @mayra-cabrera marked the task **Changelog entry added, if necessary** as completed · 34 minutes ago
- Mayra Cabrera @mayra-cabrera added 1 commit · 34 minutes ago
 - `@a3350d9` - Make DeployTokens compatible with LFS download access

Compare with previous version
- Mayra Cabrera @mayra-cabrera marked the task **Tests added for this feature/bug** as completed · 30 minutes ago

Mayra Cabrera @mayra-cabrera · 29 minutes ago

Maintainer

Todo Add todo

Assignee Edit

Kamil Trzcinski @ayufan

Milestone None Edit

Time tracking No estimate or time spent

Labels C/CD Edit

Lock merge request Unlocked Edit

3 participants

Notifications

Reference: gitlab-org/gitlab-ce...

Workspace setup

IDE: **Webstorm, VSCode, vim**

Git: **Gitkraken, Github desktop***

Extensions: [PlantUML](#), [GraphViz](#)

Extensions will be announced in every seminar session (We use many in this course)

**Note: ignore if you like solving conflicts on your own*

Activity

1. Make group of 3-4 students
2. Model: Heureka domain on paper
 - Product, Category, Store, Product Price, Product Photo
3. **Logical model:** focus on attributes, relations
4. Compare within groups in seminar
5. Code PlantUML (save it locally)
6. Git-Push to Gitlab (next slide)



Publishing the Heureka diagram as Submit 00

1. Fork the repository [pb138/pb138-iterations-2023](https://github.com/pb138/pb138-iterations-2023)
2. Clone the repository
3. Create the branch **submit-00**
4. Save **heureka.puml** to branch **submit-00**
5. Invite your seminar tutor(s) to the project as developer(s)
6. Open merge request from **submit-00** to **main**
7. Assign tutor as Assignee and Reviewer on Merge request

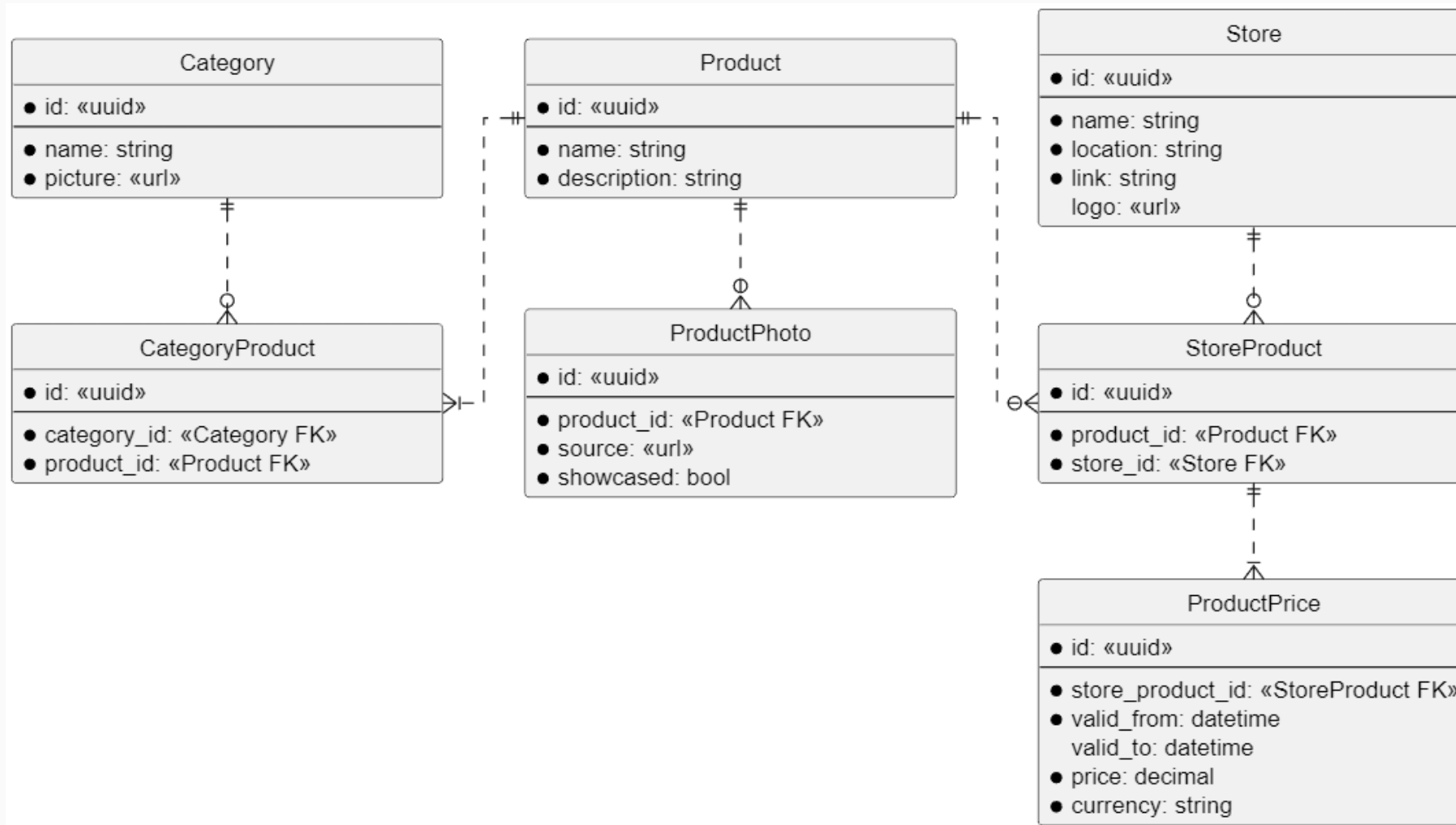
Publishing the Heureka diagram as Submit 00

And now wait... For the code review.



Demo

Demo



In case of fire



-  1. `git commit`
-  2. `git push`
-  3. `leave building`