

# Praktikum 1

Im Rahmen des Praktikums entwickeln wir eine Web-Anwendung, die wir Schritt für Schritt mit Anforderungen, Funktionen und Technologien erweitern.

Im ersten Praktikum bilden Sie Teams und erarbeiten die Rahmenaufgabe in Form einer Anforderungsanalyse sowie eines darauf aufbauenden Entwurfes.

## Aufgabe 1: Teambildung und Git-Repository

---

### 1. Teambildung

Bilden Sie Teams aus 2-3 Personen (bitte keine Einzel"teams"). Das Team arbeitet gemeinsam an der Gesamtlösung und bleibt daher das gesamte Semester über bestehen.

Denken Sie sich einen Namen für Ihr Team aus. Zur "Anmeldung" beim Praktikum schreibt bitte jedes Team eine E-Mail an [svn.joerges@fh-dortmund.de](mailto:svn.joerges@fh-dortmund.de) mit:

- dem Teamnamen
- Namen und E-Mail-Adressen aller Teammitglieder sowie
- der URL zum Git-Repository (siehe 2.).

**Wichtig:** Falls Sie auf der Suche nach einem Team sind, welchem Sie sich anschließen können, so schreiben Sie ebenfalls eine E-Mail an [svn.joerges@fh-dortmund.de](mailto:svn.joerges@fh-dortmund.de). Wir werden dann versuchen, Sie an ein noch nicht vollzähliges Team zu vermitteln.

### 2. Git-Repository

Zur Versionierung, gemeinsamen Arbeit und Bereitstellung der Abgabe am Semesterende verwenden wir in diesem Praktikum [Git](#), bzw. genauer den [GitLab](#)-Server des Fachbereiches.

**Hinweis:** Sie können sich frei entscheiden, mit welchen Tools sie programmieren und auf Git zugreifen möchten. Auf den Praktikumsrechnern ist als Entwicklungsumgebung [Visual Studio Code](#) sowie der Kommandozeilen-Client für Git ("git"-Kommando) vorinstalliert. Einen Schnelleinstieg in Git finden Sie [hier](#).

Richten Sie nun für Ihr Team ein Repository/Projekt auf GitLab ein:

1. Rufen Sie GitLab unter der URL <https://git.inf.fh-dortmund.de> auf.
2. Loggen Sie sich unter der Registerkarte "GitLab-01 FH-Kennung" mit Ihrer FH-Kennung in GitLab ein:

Sign in - GitLab

https://git.inf.fh-dortmund.de/01/users/sign\_in

## GitLab EE 01 - FHDo FB Informatik

Der hier angebotene GitLab-Dienst ermöglicht es Ihnen als **Studierender im Fachbereich Informatik** ein Softwareprojekt insbesondere mit GIT-Repository für Arbeiten im Rahmen der Lehre (Praktika und Projektarbeiten) anzulegen, zu verwalten bzw. zu planen und mit anderen Studierenden zu teilen. **Der Dienst dient ausschließlich zur Unterstützung der wissenschaftlichen Lehre, eine anderweitige, insbesondere kommerzielle Nutzung ist nicht gestattet.**

Das Erstellen von Pipelines und die Nutzung eines GitLab-Runner für CI/CD Workflows ist hier nicht vorgesehen.

Mit der ersten Anmeldung müssen Sie die [Nutzungsvereinbarung](#) anerkennen. Darüber hinaus gilt für dieses Angebot die allgemeine [Benutzungsordnung der Hochschul-IT](#).

GitLab-01 FH-Kennung

Standard

GitLab-01 FH-Kennung Username

tydur002

Password

.....

☐ Remember me

Sign in

Explore Help About GitLab

3. Vervollständigen Sie Ihr Profil mit Klarnamen und E-Mail-Adresse (z.B. über das runde Profilbild in der Menüleiste rechts oben → Menüpunkt "Einstellungen" → "Profil").
4. Wählen Sie in der Menüleiste die Schaltfläche "+" (1) und dort den Menüpunkt "Neues Projekt" (2), um ein neues Projekt zu erstellen:

Projects - Dashboard - GitLab

https://git.inf.fh-dortmund.de/01/?nav\_source=navbar

GitLab FB4

GitLab Projekte Gruppen Aktivität Meilensteine Codeausschnitte

1 +

Suche oder springe zu...

2 Neues Projekt

Neue Gruppe

Neuer Codeausschnitt

Neues Projekt

## Projekte

Deine Projekte 10 Markierte Projekte 1 Projekte entdecken

Alle Persönliche

	will-projekt / web-tech-slides	Owner	★ 0 🍴 0 📄 3	Updated vor 11 Stunden
	will-projekt / revealjs-styling	Owner	★ 0 🍴 0 📄 0	Updated vor 18 Stunden
	svjoe001 / pk-git-example	Maintainer	★ 0 🍴 1 📄 0	Updated vor 1 Tag
	svjoe001 / spring-mvc-todo-app	Maintainer	★ 0 🍴 0 📄 0	Updated vor 2 Tagen
	svjoe001 / vuejs-example	Maintainer	★ 0 🍴 0 📄 0	Updated vor 3 Tagen
	svjoe001 / express-example	Maintainer	★ 0 🍴 0 📄 0	Updated vor 3 Tagen

5. Im folgenden Formular sind vor Allem zwei Einstellungen wichtig:

1. Geben Sie Ihrem Projekt unter "Projektname" (1) einen Namen nach dem Muster `web2-ws19-$teamname`, also z.B. `web2-ws19-133tT34m`.
2. Stellen Sie die Sichtbarkeit Ihres Projektes unter "Visibility Level" (2) auf "Privat".

**Neues Projekt**

In einem Projekt speicherst du deine Dateien (Repository), planst deine Arbeit (Tickets) und veröffentlichst deine Dokumentation (Wiki), [unter anderem](#).

Alle Features sind für leere Projekte, Vorlagen, oder beim Import aktiviert. Du kannst sie aber danach in den Projekteinstellungen deaktivieren.

Um die CI/CD-Funktionen nur für ein externes Repository zu verwenden, wähle **CI/CD für externes Repo** aus.

Information about additional Pages templates and how to install them can be found in our [Pages getting started guide](#).

**Tipp:** Du kannst ein Projekt auch über die Kommandozeile erstellen. [Befehl anzeigen](#)

**Blank project** | Create from template | Import project | CI/CD für externes Repo

**1** Projektname  
pk-praktikum

Projekt-URL  
https://git.inf.fh-dortmund.de/svjoe001

Projekt-Slug  
pk-praktikum

Want to house several dependent projects under the same namespace? [Create a group](#).

Project description (optional)  
Description format

**2** Visibility Level  
☒ Privat  
Jedem/Jeder Benutzer(in) muss explizit der Zugriff auf das Projekt gewährt werden.  
☐ Intern  
Auf das Projekt kann jede(r) angemeldete Nutzer(in) zugreifen.

☐ Initialize repository with a README  
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

**3** Create project | Abbrechen

Nach Klick auf "Create project" (3) sieht Ihr neues leeres Projekt wie folgt aus:

svjoe001 / pk-praktikum · Details

Das Projekt 'pk-praktikum' wurde erfolgreich erstellt.

**pk-praktikum** Projekt ID: 75

[Lizenz hinzufügen](#) [Clone](#) [Markieren](#) 0

Das Repository für dieses Projekt ist leer

You can create files directly in GitLab using one of the following options.

[Neue Datei](#) [README hinzufügen](#) [CHANGELOG hinzufügen](#) [CONTRIBUTING hinzufügen](#)

**Command line instructions**

You can also upload existing files from your computer using the instructions below.

**Git global setup**

```
git config --global user.name "svjoe001"
git config --global user.email "sven.joerges@fh-dortmund.de"
```

**Create a new repository**

```
git clone https://git.inf.fh-dortmund.de/01/svjoe001/pk-praktikum.git
cd pk-praktikum
```

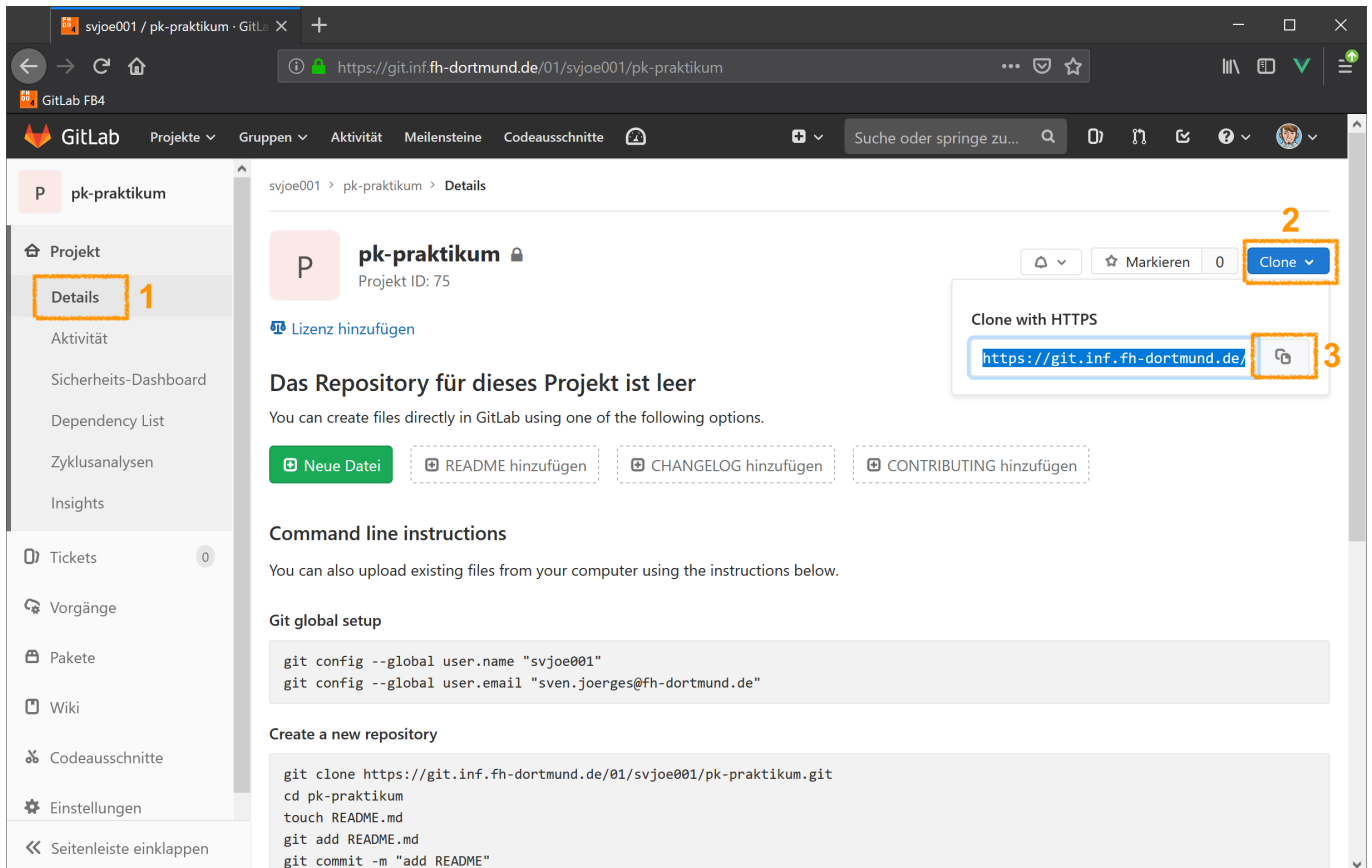
6. Wechseln Sie in Ihrem Projekt nun im Menü links in den Bereich "Einstellungen" → "Mitglieder" (1).

Wählen Sie rechts auf der Seite "Project members" die Registerkarte "Invite member" (2). Tragen Sie im Feld "Select members to invite" (3) den folgenden Benutzer ein: `svjoe001` (Sven Jörges).

Stellen Sie unter "Choose a role permission" (4) die Rolle "Developer" ein.

The screenshot shows the GitLab web interface for the 'pk-praktikum' project. The left sidebar contains navigation links: 'Projekt', 'Tickets', 'Vorgänge', 'Pakete', 'Wiki', 'Codeausschnitte', and 'Einstellungen'. Under 'Einstellungen', 'Mitglieder' is highlighted with an orange box and labeled '1'. The main content area is titled 'Project members' and includes the text 'You can invite a new member to **pk-praktikum** or invite another group.' Below this, there are two tabs: 'Invite member' (labeled '2' with an orange box) and 'Invite group'. The 'Invite member' tab is active, showing a form with two main sections. The first section, 'Select members to invite' (labeled '3' with an orange box), contains a text input field. The second section, 'Wähle eine Rollenberechtigung aus' (labeled '4' with an orange box), contains a dropdown menu with 'Developer' selected. Below these sections are links for 'Erfahre mehr über Rollenberechtigungen', an 'Expiration date' field, and buttons for 'Zum Projekt hinzufügen' and 'Importieren'. At the bottom, there is a section for 'Existing members and groups' showing 'Members of **pk-praktikum**' with a count of 1, and search filters for 'Find existing members by name' and 'Name, aufsteigend'.

7. Auf der Detailseite Ihres Projektes (links im Bereich "Projekt" → "Details" (1)) finden Sie durch Betätigen der Schaltfläche "Clone" (2) die URL zum Klonen Ihres Git-Repositories.



## Aufgabe 2: Anforderungsanalyse

Ihr Kunde ist ein privates Bildungsinstitut, welches mit dem Auftrag an Sie herantritt, eine Web-Anwendung zur Unterstützung bei der Bewertung von Klausuren zu entwickeln. Aktuell arbeitet jeder Dozent hier autark, es gibt also keine einheitliche Lösung. Großer Beliebtheit bei einigen Dozenten erfreut sich jedoch ein Excel-Sheet, das beim Eintragen von Punkten unterstützt, Noten berechnet und sogar einige Informationen wie z.B. den Notenschnitt liefert.

Diese Lösung hat jedoch einige Nachteile:

- Pro Klausur wird aktuell ein neues Excel-Sheet erstellt.
- Die Excel-Sheets liegen auf den Rechnern der Dozenten verstreut. Sie werden nicht zentral vorgehalten und - wichtiger - es kann somit auch keine wirksame und einheitliche Absicherung der personenbezogenen Daten erfolgen (Datenschutz!).
- Sind mehrere Dozenten an der Bewertung einer Klausur beteiligt, so muss das entsprechende Excel-Sheet aufwändig ausgetauscht werden (z.B. auf USB-Sticks).

Daher wünscht sich der Kunde eine Web-Anwendung, die eine zentrale und sichere Klausurbewertung unterstützt.

## 1. Anforderungserhebung durch Interview des Kunden (Simulation im Praktikum)

Zur Erhebung der Anforderungen treffen Sie sich mit einem Dozenten des Bildungsinstituts. Dieser demonstriert Ihnen den IST-Stand, d.h. das aktuell verwendete Excel-Sheet.

Stellen Sie dem Kunden Fragen, und erheben Sie so die Anforderungen sowie die Fachlichkeit der gewünschten Anwendung. Notieren Sie sich die Antworten!

## 2. Funktionale Anforderungen dokumentieren

Dokumentieren Sie die funktionalen Anforderungen der Web-Anwendung. Nutzen Sie dabei folgendes Schema zur Beschreibung einer einzelnen Anforderung:

- Eindeutiger Bezeichner (zur Identifikation der Anforderung)
- Name der Anforderung
- Kurzbeschreibung (2-3 Sätze)
- Priorität (hoch, mittel, niedrig, unklar)
- Akteure/Rollen

**Hinweis:** Stellen Sie die Dokumentation als PDF oder in einem Klartextformat (z.B. txt, [Markdown](#)) in Ihrem Git-Repository (siehe Aufgabe 1) zur Verfügung.

## Aufgabe 3: Entwurf

---

### 1. Fachmodell erstellen

Erstellen Sie ein angemessenes UML-Modell, welches die relevanten Fachobjekte (Entitäten), deren Attribute sowie Beziehungen beschreibt.

**Hinweis:** Als UML-Werkzeug können Sie z.B. [UMLet](#) (bzw. die Web-Variante [UMLetino](#)) oder [StarUML](#) verwenden - oder ein anderes Tool Ihrer Wahl. Bitte exportieren Sie Ihr fertiges UML-Modell in ein Bildformat wie z.B. PNG und stellen Sie dieses in Ihrem Git-Repository (siehe Aufgabe 1) zur Verfügung.

### 2. Plausibilitäten

Bestimmen Sie auf Basis der funktionalen Anforderungen und des Fachmodells die fachlichen *Plausibilitäten* der Daten. Betrachten Sie zu diesem Zweck die ermittelten Fachobjekte (Entitäten) im Fachmodell und deren Attribute, und beantworten Sie jeweils folgende Fragen:

1. Welche Wertebereiche werden für die Attribute benötigt? Gibt es Einschränkungen für die möglichen

Attributwerte (z.B. Ober- und Untergrenzen, bestimmte Formate etc.)?

2. Welche Attribute stellen "Mussfelder" dar, die unbedingt mit einem Wert versehen werden müssen?  
Welche Attribute repräsentieren optionale Angaben?

Dokumentieren Sie die ermittelten Plausibilitäten (d.h. Antworten auf die o.g. Fragen) und stellen Sie diese Dokumentation in Ihrem Git-Repository (siehe Aufgabe 1) zur Verfügung.