# **SWT Waterfall Project**

## **Tech Stack**

Javscript (Frontend)	J5
Flask (Backend)	Flask web development, one drop at a time
SQLite (DB)	SQLite

## Backend extensions:

Jinja (Template Engine)	Jinja
SQLAlchemy (ORM)?	<b>SQLAlchemy</b>

### Flask

Python version 3.10.0
-----------------------

- 1. Git Repository aus flask branch pullen: "git pull origin flask". Falls noch kein lokales Git Repository erstellt, bitte zu "Git einrichten" scrollen, dann hier weitermachen.
- Python installieren, Version 3.10.
   (Check if python is installed via: "python --version")
- 3. Python Virtual Environment erstellen. Geeignetes Verzeichnis dafür suchen.
  - a. "python -m venv .venv"
- 4. Python Virtual Environment aktivieren
  - a. Windows: ".venv\Scripts\activate.bat".
    Falls Skripte deaktiviert sind, zuerst: "Set-ExecutionPolicy
    Unrestricted -Scope Process".
  - b. Linux: "source .venv/bin/activate".
- 5. Notwendige Packages installieren: "python -m pip install -r requirements.txt".
- 6. IntelliJ öffnen, Projekt öffnen, main.py öffnen, rechtsklick, "Run 'main.py' ", URL in Konsole anlicken.
- 7. Ta-da!

### Git einrichten (Bash):

- 1. Git Bash installieren.
- 2. GitBash in geeignetem Verzeichnis öffnen.
- 3. Lokales git initalisieren: "git init".
- 4. Remote Repository hinzufügen: "git remote add origin https://github.com/kr1pt0n05/Softwaretechnik-Waterfall-Project.git".

## Änderungen auf GitHub pushen:

- 1. git add.
- 2. git commit -m "Deine Änderungen kurz beschreiben"
- 3. git push -u origin main

Beachte, zuerst immer Repositories synchronisieren: "git pull".

```
Andere Nützliche Befehle:
```

```
git status
git ls-files
```

git branch

git checkout -b BRANCH

git rm -r -cached.

## **SQLite**

#### Referenz:

## Jinja

Ist bereits in Flask integriert. Erlaubt Python Code in html-files auszuführen. Jina läuft im Backend, d.h. der Python Code wird serverseitig ausgeführt und nicht im Frontend des Nutzers.

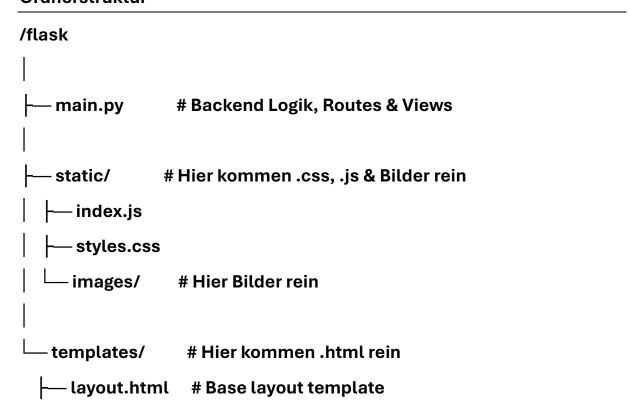
Beispiel siehe: index.html

## **SQLAlchemy**

Statt lästige SQL Befehle schreiben zu müssen, können wir unsere Daten auch in Python Klassen verpacken und dann in unsere Datenbank schieben.

Die Library übersetzt unsere Python Klasse quasi in SQL.

#### Ordnerstruktur



index.html # Example template