

Lab 4: First steps with Flask

Objective

The objective of this lab is to create a **back-end** in **Python** for the **social network** developed in the previous labs, using the **Flask** framework.

Activity description

To complete the lab, follow these steps:

Create a new HTML page "About us"

The page must contain static text and images that describe you as developers of the social network. For styling, continue using **Bootstrap**.

Develop a Flask application `app.py`

- Define a **decorator** (and its corresponding function) for the social network **homepage**.
- Define a **decorator** (and its corresponding function) for the "About us" page.

Manage posts on the homepage

- Create a **data structure** in **Python** to store posts.
- Each post can be represented as a **dictionary** containing: `username`, `publication date`, `main text`, `profile image path`, `post image path` (if present).
- Posts can be stored in a **list of dictionaries**, as in the following example:

```
posts = [{
    "username": "user123",
    "publication_date": "2025-03-25",
    "text": "This is an example post.",
    "profile_image": "/static/images/profile.jpg",
    "post_image": "/static/images/post.jpg"
}]
```

Use the Jinja templating engine to serve web pages, keeping the HTML structure separate from the application data:

- Organize HTML files inside the `templates` folder.
- Modify the **homepage** HTML file to remove static posts and replace them with **Jinja** expressions and statements.
- Add a link to the "**About us**" page in the **navbar**.
- Adapt the "**About us**" HTML file to use **Jinja** expressions for images and any links.