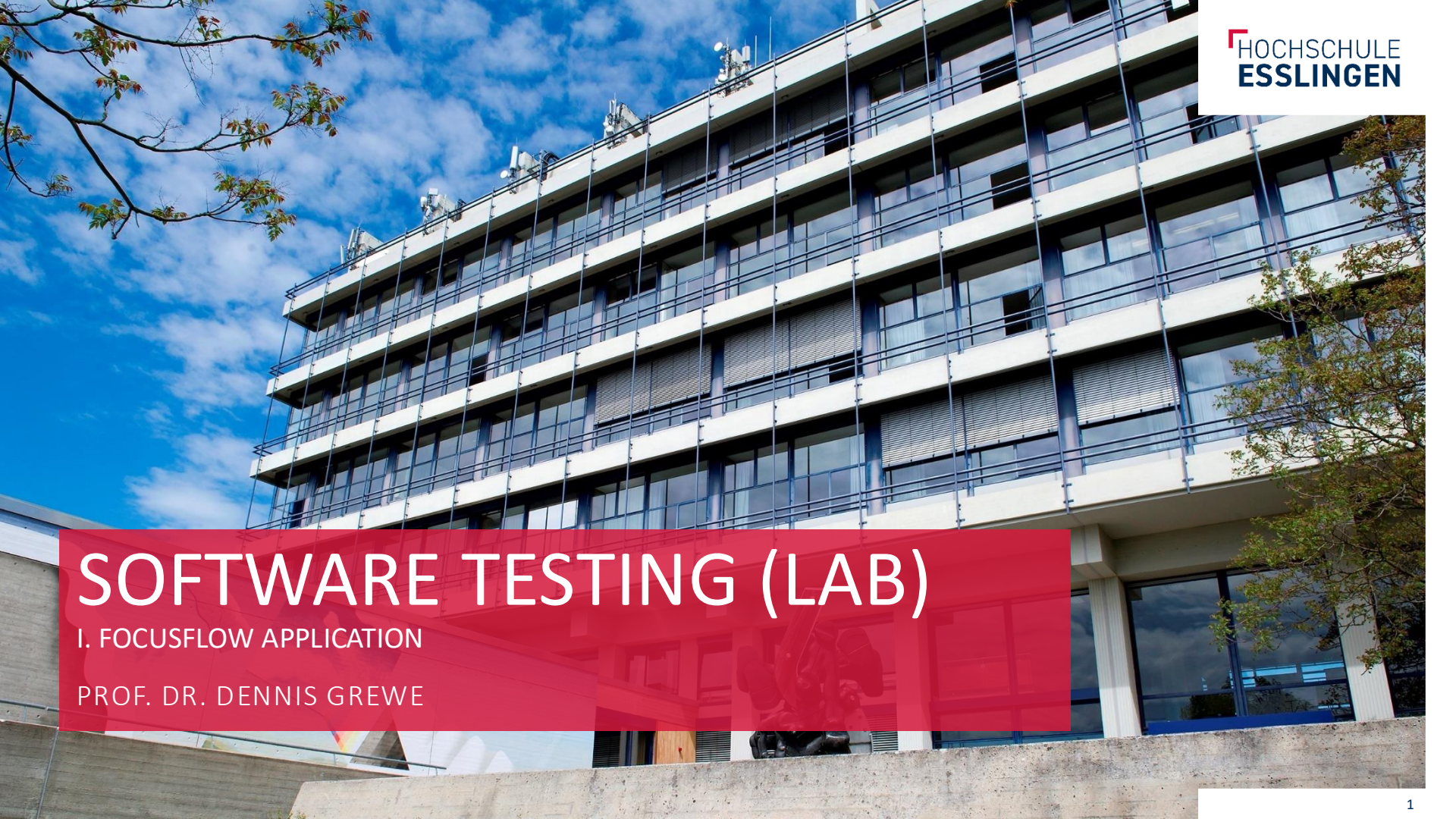

HOCHSCHULE ESSLINGEN

# SOFTWARE TESTING (LAB)

## I. FOCUSFLOW APPLICATION

PROF. DR. DENNIS GREWE

# LEARNING OBJECTIVES

- Lear about the fictious use case example, FocusFlow,  and its aspects
- Apply lessons learned from the lecture to the use case.
- Together we will develop the application using state of the art software quality tools and implementations.

# FOCUSFLOW

# USE CASE: FocusFlow

- **FocusFlow** is a lightweight web-based productivity app aimed at helping individuals and small teams organize their daily tasks effectively:
  - FocusFlow aims to streamline daily workflows by avoiding bloated functionality.
  - It provides a clear way to capture to-dos, prioritize them, track their progress, and keep everyone on the same page.
  - FocusFlow ensures both personal productivity and small-group collaboration remain intuitive and efficient.
  - Tasks are the central element in FocusFlow. Those can be grouped into categories or projects, helping users structure workloads across different domains.

# FocusFlow: SOME REQUIREMENTS

- **Task Creation & Editing:**
  - Users can create tasks with title, description, due data, and priority level.
  - Users can edit existing tasks to update status or modify details.
- **Task Status Management:**
  - Each task has a status, e.g., “Open”, “In Progress”, or “Done”.
  - The system automatically moves tasks to “Overdue” if the due date is passed and the task isn´t complete.
- **User Assignment:**
  - Users can assign tasks to themselves or to teammates (multi-user assignment optional).
  - Assigned users should be notified automatically of newly assigned tasks (e.g., by email or in-app notifications).

# FocusFlow: SYSTEM REQUIREMENTS

- **FocusFlow** will be implemented as a 3-tier architecture comprising of
  - **Database** – to persist tasks and user data
  - **API Backend** – to access task and user data
  - **Frontend** – to visualize task and user data

- It is up to the team to select the programming framework / technology to bring FocusFlow to live.
  - Please document your decision what kind of technologies are used and why!!

# TECHNOLOGIES WE WILL USE DURING THE COURSE

- Despite the technologies selected to develop FocusFlow within the teams, we will use the following set during the course:










# USEAGE OF CODING AIs

- During the course, I will use coding AIs to improve my productivity during live coding session.
- I encourage you to test and interact with code generating AI to increase your awareness of capabilities and limitations of such tools.
- Most of the big(ger) companies in and around Stuttgart do provide access to coding AIs for their developers.
- Example AI tools include (but not limit) Anthropic Claude 3.5/3.7 Sonnet, Microsoft/GitHub CoPilot, OpenAI Codex, … (**caution**: some of them demand for paid plan)
- **Hint**: GitHub CoPilot provides free plans
  - About GitHub CoPilot Free and Free Access to CoPilot Pro for Students)

# FORMATION OF DEV TEAMS

- Next, let us now create development teams (max. 4 people per team):
  - Group 1:
  - Group 2:
  - Group 3:
  - Group 4:
  - Group 5:

# LAB EXERCISE 1: PROJECT SETUP & TECH SELECTION

- After team formation please fulfill the following task:
  - We are organizing our projects using GitHub.
    - Each team member should create a GitHub account to get access: https://github.com/signup
    - Create a fork of https://github.com/dgrewe-hse/focusflow
  - Ensure all team members do have access to your team project
  - Critically think about the base technologies and programming languages you want to use to implement the FocusFlow application.
  - Critically think about how you want to organize the development tasks in your team during the semester.
  - Create a “doc” folder in your project and document your decisions properly.
- Submit a link to your documentation to the first assignment in the Moodle course.
  - Group submission will be enable -> only one person of the group need to assign in Moodle