GitHub

Modern engineering teams use version control systems to manage complex designs, coordinate work between many contributors, and maintain a complete, traceable history of changes. GitHub is one of the most widely used platforms for this purpose in industry and research.

In this course, GitHub is used as the official submission, verification, and integration system for all design work. It allows:

1. Each participant is to work independently without interfering with others
2. Automated checks to verify files before they are accepted
3. Designs from many contributors are to be safely merged into a single layout for peer review and fabrication
4. A complete, permanent record of who submitted what and when

The workflow you will practice here—forking a repository, committing files, running automated checks, and creating a pull request—matches how collaborative engineering projects are managed in practice.

This Week-1 assignment is a low-stakes introduction to that workflow. The file content is unimportant; the goal is to ensure that Week-3 design submission and peer review proceed smoothly.

Step 1 — Create a GitHub account

- Create a free GitHub account

Step 2 — Fork the course repository

- Open the course repository:
  https://github.com/SiEPIC/openEBL-2026-02
- Click Fork (top right)

A fork is your personal copy of the repository where you can upload files without affecting anyone else's work.

Step 3 — Prepare your PDF file

- Create a short PDF file.
- The PDF may be blank or contain minimal text (for example, your name or username).

This file is not evaluated for content.
It is only an exercise to practice the GitHub submission, verification, and pull-request workflow that will be required later in the course.

File naming (IMPORTANT)

Calculator

Your PDF must follow the same naming convention used later for GDS layout submissions.

EBeam_<Username>.pdf

Do not include version numbers (such as _v1, _v2, or _final) and do not add extra text to the filename.

Why no version numbers?
GitHub automatically tracks every change you make. When you upload a file with the same name again, GitHub replaces the file while preserving the full version history internally. In professional workflows, the correct approach is to keep one consistent filename and let the version control system manage restoration.

Step 4 — Upload the PDF to your fork

- In your fork of the repository, open the reports folder
- Click Add file and select Upload files
- Upload your correctly named PDF
- Click Commit changes

Step 5 — Create a Pull Request and review verification results

- Click Pull requests in your fork
- Click New pull request
- Open a pull request from your fork to the main course repository
- After the pull request is created, automated verification will run on the pull request itself as part of the main repository workflow.

IMPORTANT:
If verification fails, click on the verification checks and read the error messages carefully. The checks will tell you exactly why the submission failed.

- Later in the course (Week 3), verification may fail due to:
- Incorrect file naming
- Files placed in the wrong folder
- Invalid layout floor plans
- Missing or incorrect labels
- Other design rules or consistency checks

Reading the verification output is essential. The purpose of the automated checks is to help you identify and correct problems efficiently. You are expected to modify your submission and re-run verification until it passes.

Step 6 — Create a Pull Reques

Calculator

- Click Pull req
- Click New pull request
- Open a pull request from your fork to the main course repository

This step mirrors the process used later for design aggregation, peer review, and fabrication.

Step 7 — Submit your link in edX

Previous Next