10. Project 2

Sustainable Software Engineering
CS4575

Luís Cruz
L.Cruz@tudelft.nl

Carolin Brandt
C.E.Brandt@tudelft.nl

Enrique Barba Roque
E.BarbaRoque@tudelft.nl

1. Goal/assignment
2. Deliverables
3. Strategy
4. Ideas

Assignment

- **Goal:** Solve a Sustainable Software Engineering problem.
  - **Identify 1 problem** that should be fixed to help enabling sustainability in the software engineering industry/community.
  - **Propose a solution**. A tool, framework, guidelines, etc.
  - **Implementation**.
  - **Validation**. (Depending on the idea)
  - **Dissemination**/social impact. (Solution should be open source, welcome contributors, post on social media? Tool website? Available in a package manager?)

Deliverables

- Paper-like **article**. (Min 4 pages, max 10)
- Online **git repo** with open source codebase and/or replication package.
- **Presentation**: 5 min + 5min Q&A

Article

- Different projects will have different expectations -> Make agreements with your coach.
  - Some projects are more technical and some projects more theoretical.
- Common requirements:
  - **Motivation**, formulation of the **problem** being addressed, etc.
  - Description of the **solution**.
  - **Validation** of the solution (if applicable -> discuss with coach)
  - **Discussion** of the solution. (Including limitations)

Strategy

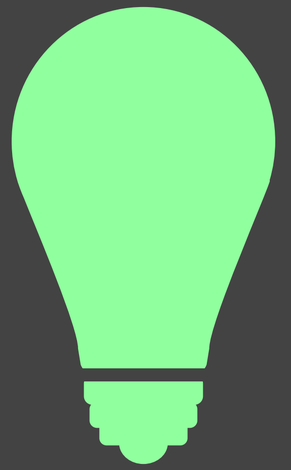
- Starting next week, there are no lectures
- Steering meetings from week 5 till week 9 (either online or in person).
  - 1 steering meeting per week. (**4+1 sprints**)
- Every week, you need to plan different tasks and assignments.
- Deadline **April 4**.

Strategy

- **Week 0**
  - Decide project idea (today)
  - Define steering meeting schedule
  - Create working document of the article: Problem statement and solution proposal!
  - Define and assign tasks for each week.
- **Week 1**
  - Implementation
  - Agreements with supervisor.
- **Week 2**
  - Implementation
- **Week 3**
  - Implementation, Full draft of article, dissemination.
- **Week 4**
  - Final refinements
  - Prepare presentation

Project ideas

- A1. Prototype **cross-machine comparable** benchmarks
- A2. Add energy metrics to **LMstudio/Ollama**
- A3. **Visualizations** built-in with EnergiBridge
- A4. **Service-based** version of EnergiBridge



Project ideas

- B1. Measure energy consumption of single **JUnit tests**
- B2. Study **test generation** energy consumption
- B3. Energy profiling of **static analysis** tools
- B4. Detailed energy profiling of **build pipelines**

Quality Assurance & Testing

- B5. Tool supporting **SusAF** workshop / process
- B6. Dataset of **government-developed** OS software
- B7. Queue - but **better for** the student / TA **society**

Social & Individual Sustainability

Project ideas

- C1. Compare energy consumption of **docker images** for ML workloads.
- C2. Create a plugin to visualize **Hugging Face** carbon emissions in detail.
- C3. Plugin for **ChatGPT** (footprint per chat window)
- C4. **NutriScore** for software libraries.
- C5. Add energy-awareness to existing software
- C6. Green **Shift Left**
- C7. **Education** for Sustainable SE

A1. Prototype cross-machine comparable benchmarks

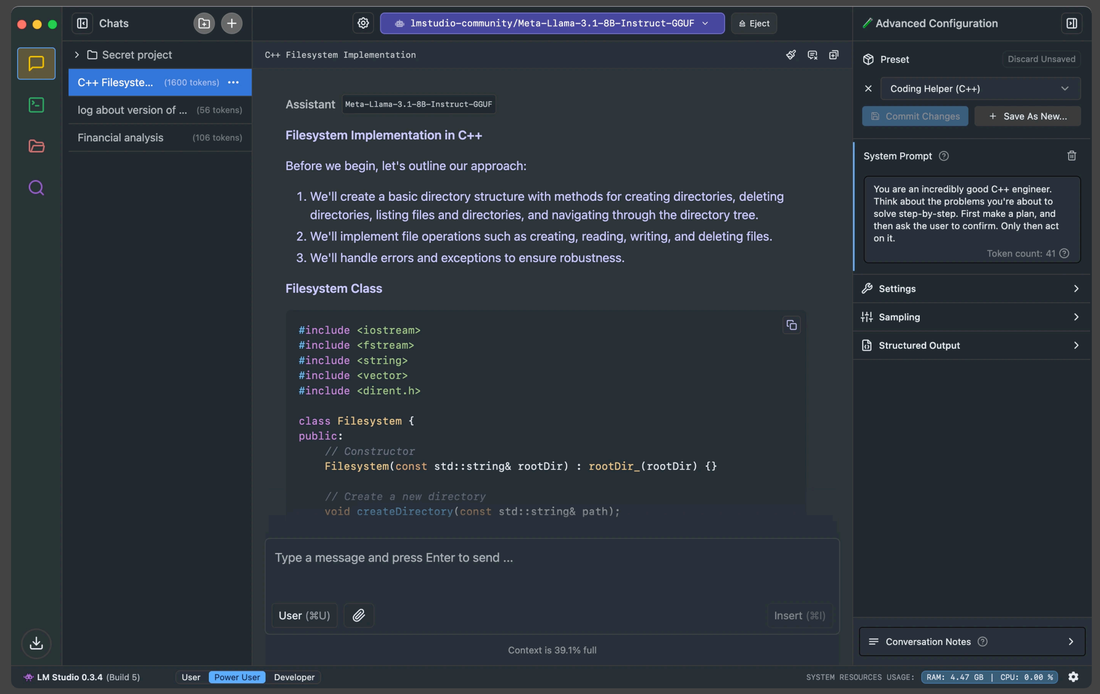
- Energy-usage comparisons require running both baseline + software on the same machine → limits how extensive our experiments can be
- Research community is in need of benchmarks that make energy measurements comparable even if executed on different machines

- Focus on a single task or model (i.e. computing vision, classification)

A2. Adding energy metrics to LMstudio/Ollama

- Make energy consumption visible to users within local chat-interface for LLMs
- LMstudio/Ollama are tools for easy deployment of LLMs
  - Do not show energy metrics
- Add energy metrics to LMstudio-python or Ollama

https://lmstudio.ai/



A3. Visualizations built-in with EnergiBridge

- EnergiBridge simplifies energy measurement, but analysis & visualization is still left to the user
- Extend the tool with well-chosen visualizations and analyses directly from the provided data

A4. Service-based version of EnergiBridge

- Simplify interaction & setup with EnergiBridge
- Service that runs independently, start/stop signals over RPC to manage experiments
- Potential: create EnergiBridge interface for other prog. lang

B1. Measure energy consumption of single JUnit tests

- We’d like to identify energy anti patterns in unit tests
- As a first step, we need tooling to measure and compare the energy consumption of single unit tests
- Ideally including preliminary analysis looking at potential reasons for high-energy-consuming tests

B2. Study test generation energy consumption

- Automatic test generation mainly focuses on making strong test suites
- Do different techniques and configurations impact the energy consumption during generation?
- Preferably focus on non-LLM test generation methods (EvoSuite, Pyguin, DSpot)

B3. Detailed energy profiling of build pipelines

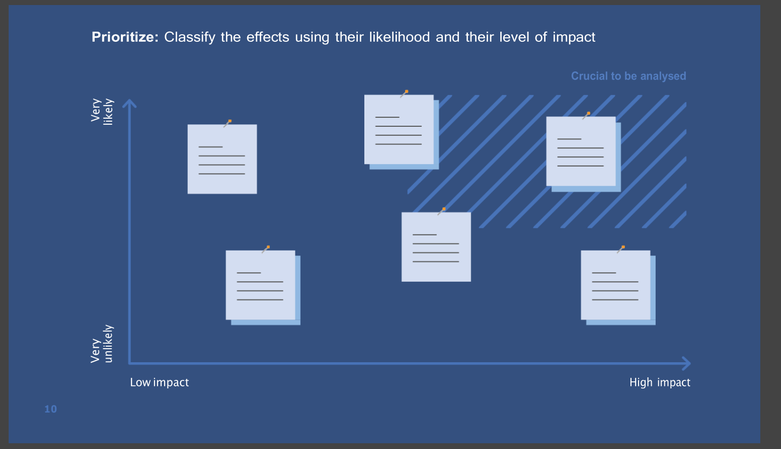
- Automatic builds have become a cornerstone of quality assurance. But how much energy do they even consume?
- Create a tool that reports on the energy consumed during the (different stages of the) **whole build** (compile, build, test, package, …)
- Should be integrated with build system(s), making setup for developers easy
- For local setup (to enable true energy measurements)

B4. Energy profiling of static analysis tools

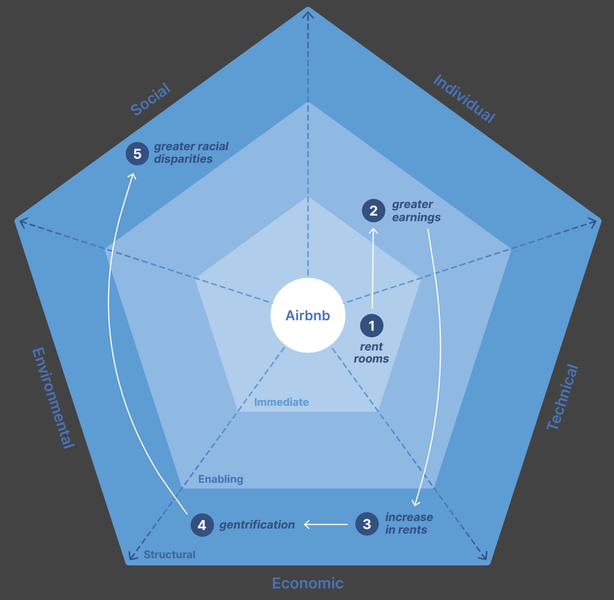
- What is the energy consumption of a "typical run" for a few OSS projects?
- Does the type of analysis matter? Are certain analysis more expensive? Does the number of rules that are activated in a static analysis tool important for the energy consumption?
- Differences between static analysis tools [lower priority]

B5. Tool supporting SusAF workshop / process

- Lead engineers through process & questions
- Interface to create & document the two diagrams
- (!) Should be easy to start using & set up



- You may also create a simplified version / your favorite sustainability framework



B6. Dataset of government-developed OS software

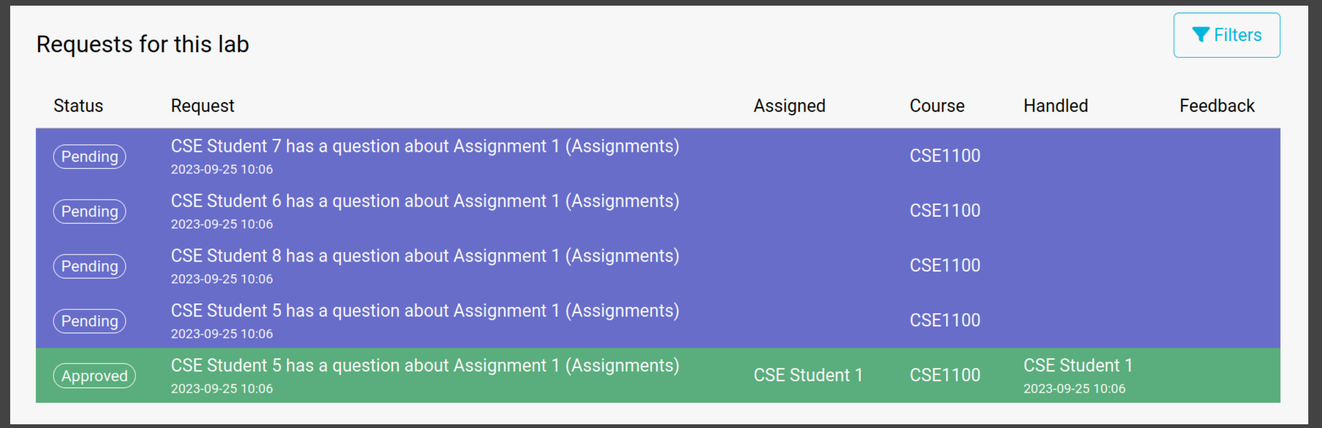
- Governments develop & use software for **supporting society**
  Open-source development & policies are on the rise
- But studying government software is difficult b/c we don't know what is out there
  → Lack of incentive to make popular
  → Language barriers
- Create a comprehensive dataset, **incl. data to understand state** of software: buildable?, open dev. history?, requirements documentation?
- Could start with NL, but including your / other countries greatly appreciated!

B7. Queue - but better for the student / TA society

- Requirements analysis regarding **social and individual sustainability effects** of Queue
- Other EIP / TUD used software also possible: e.g., Answers EWI

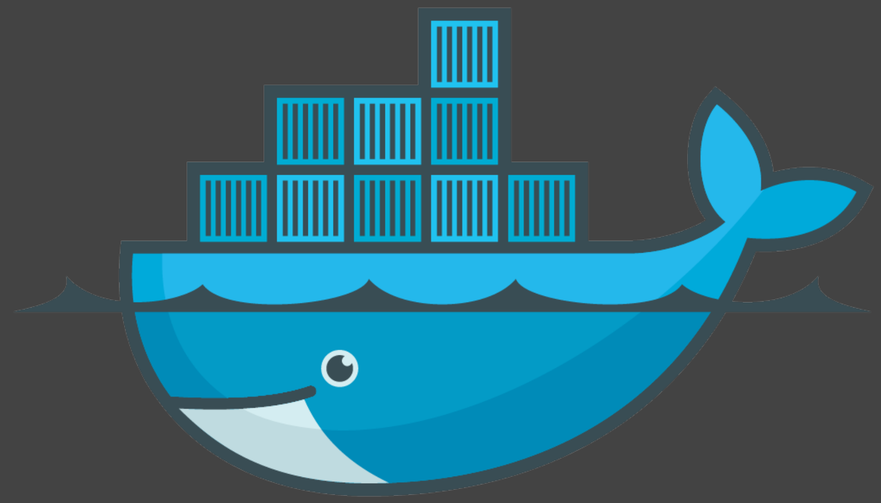
https://eip.pages.ewi.tudelft.nl/eip-website/queue.html

- Non-technical project → proper process (workshops? Interviews?) & rich description of outcomes focus of grading



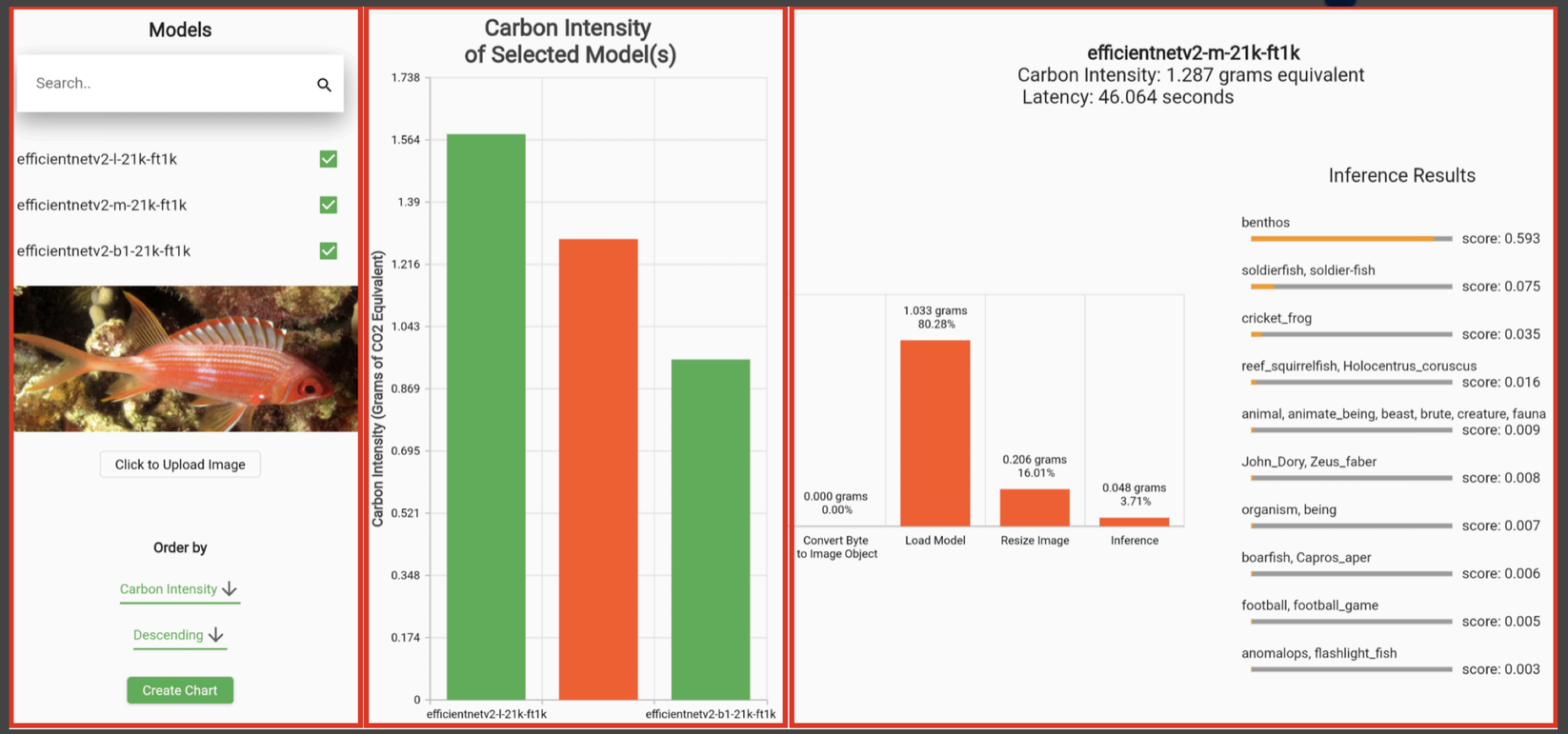
C1. Compare energy consumption of docker images for ML workloads.

- Similar to what we have seen in the lectures but for **ML-specific workloads**.
- We can reuse **existing** experiment **replication packages**.



C2. Plugin to visualize Hugging Face carbon emissions.

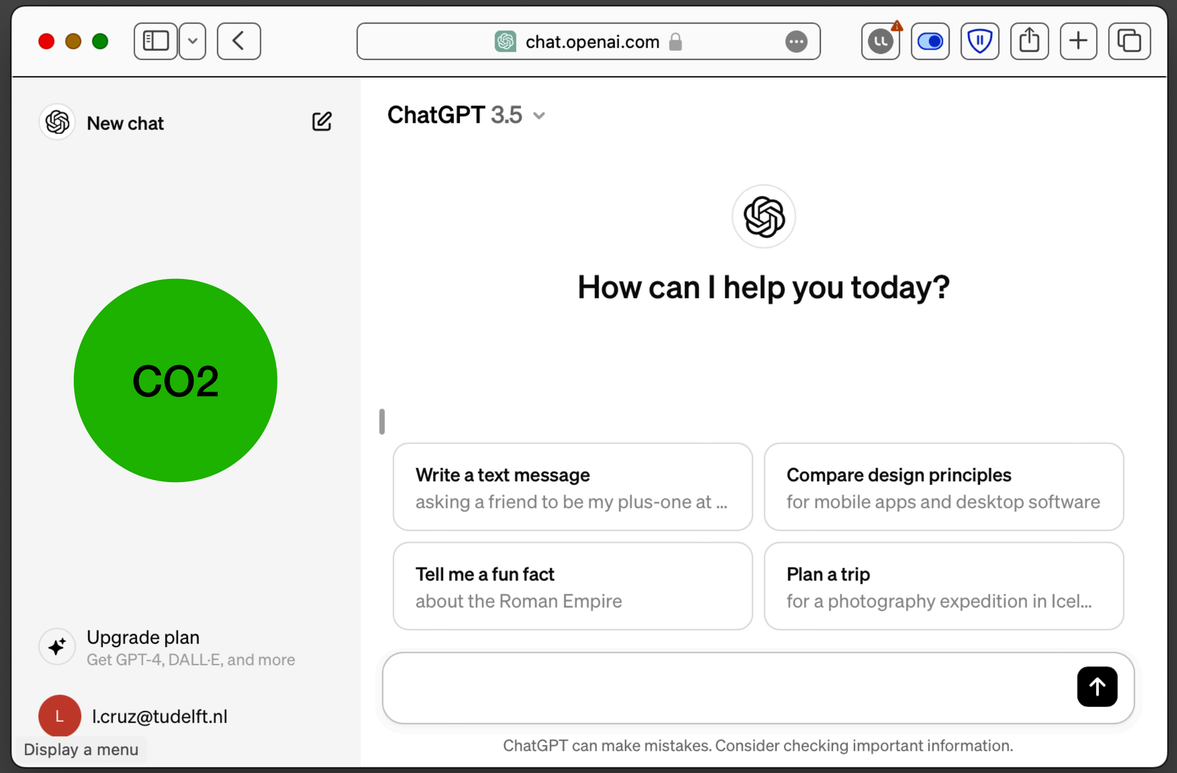
- https://huggingface.co/blog/leaderboard-emissions-analysis
-



Yoo, Taewon, et al. "Visualizing the Carbon Intensity of Machine Learning Inference for Image Analysis on TensorFlow Hub."

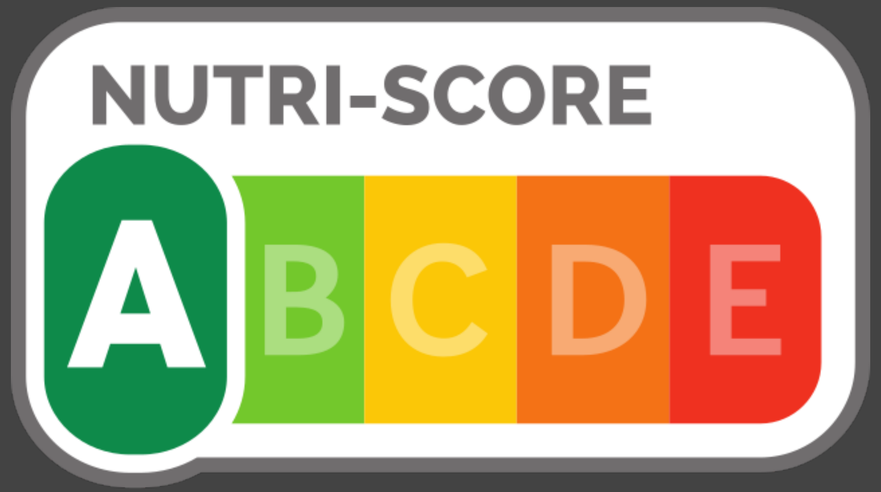
C3. Plugin for ChatGPT

- Users seldom know how much carbon they are emitting when they interact with chat GPT.
- Let’s make it transparent to the users. Browser plugin?



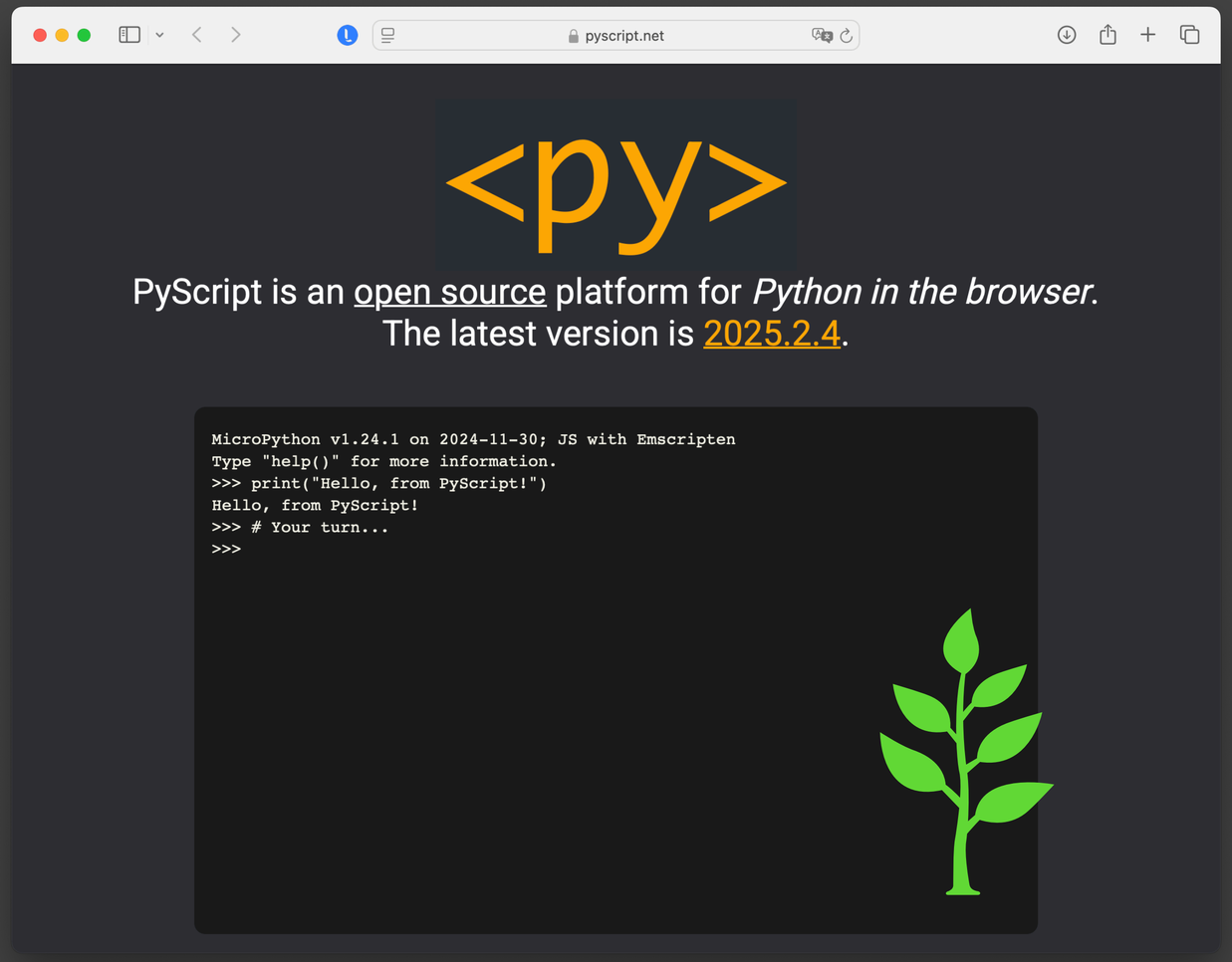
C4. NutriScore for software libraries

- **NutriScore labels** are not perfect but they are a good **starting point**!
- What if we could do the same thing for the **energy efficiency of software**.
- (Also open to **individual** or **social** sustainability)
- This work can be **scoped** in particular domains/ecossystems/use cases.
    - Libraries for stats? ML? Web Dev? Cloud?



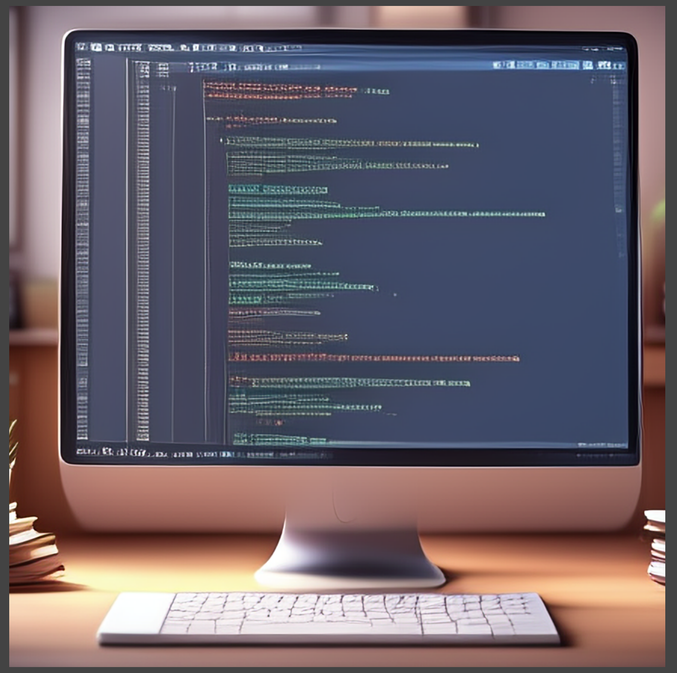
C5. Add energy-awareness to existing software

- **Streamlit**, **Notebooks**, **JSFiddle**, **PyScript**, **zsh**, etc.
-



C6 - Green Shift Left

- Estimate energy efficiency using **static code analysis**.
- We don’t need an accurate value.
- It is useful to know which **code** is more likely to introduce **energy hotspots** and that should be **reviewed** with more attention.
- Can be scoped to a particular domain (react, php, data science, web, etc., etc.)



C7 - Education

- **Educational game** for Software Sustainability practices
- Purpose: use within software teams to discuss or learn about different sustainable IT practices: at the organisation level, software, etc.

https://github.com/OttoKaaij/Ticket-To-Sustainability/?tab=readme-ov-file

Project ideas (old)

- Plugin from **EnergiBridge** (GUI, report generator, python library, etc.)
- Plugin for **ChatGPT** (carbon emissions per chat window)
- **Seamless measurements** for AI libraries
- Energy **patterns** for Green AI
- Sustainable SW dev **gamification**
- Sustainability **auditor for AI** projects
- Energy Profiling of **screen colour filter tools** (or display settings)
- … you can also propose yours!

edu.nl/64gpk

https://edu.nl/64gpk