Friedrich-Alexander-Universität
Erlangen-Nürnberg

ProgLLM

Status Report

Lisa Rebecca Schmidt

Friedrich-Alexander-Universität Erlangen-Nürnberg

9. Feb. 2024

Problems Encountered

- Obsolete libraries/APIs/dependencies
  - remedy: ask for the version!
  - very annoying: it even had old versions of the openai apis...
- "This task would require a team of developers .../ "This task is highly complex .../ Ï can provide a basic implementation which needs to be extended"
  - remedy: Prompt: "You are a team of experienced [XYZ-] engineers".
- Systems design can be overwhelming, as it will present many different technologies which you may not yet know.
- Configurations are always the same! Especially when you design microservices, it will always use the same ports.
- The interfaces which are built are not actually used! -> you have to know how apis/sockets/etc work

Insights

- It still makes sense to learn new technologies
  - If you already know the technologies, it will speed you up even more.
  - You will have a much easier time debugging.
  - You know ’best’ practices.
  - You know what you are doing ;)
  - I failed miserably when I trusted GPT-4 to use React (I don’t know React).
- LLMS can speed up coding significantly.
- The LLM can actually handle simple systems-architecture problems.
- You are very much forced to build decoupled and modular code (which is good) because the LLM will have "forgotten"your specs.
  - Bonus: Let the LLM generate interface specs, these you can then put in the prompt if you need a specific module.
- You are also very much forced to use Infrastructure-as-Code and templates, because LLMS can only generate text.
- You will lose a lot of time if you switch back and forth between Google/Stack Overflow and your LLM - stick to one!