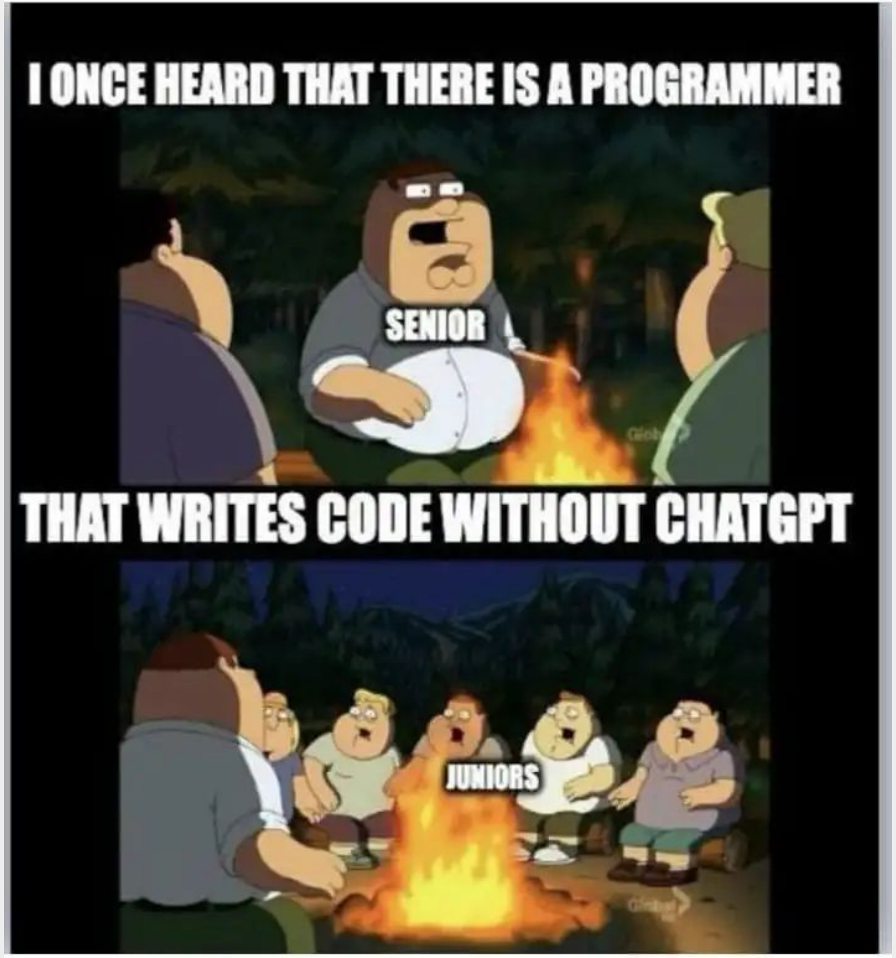
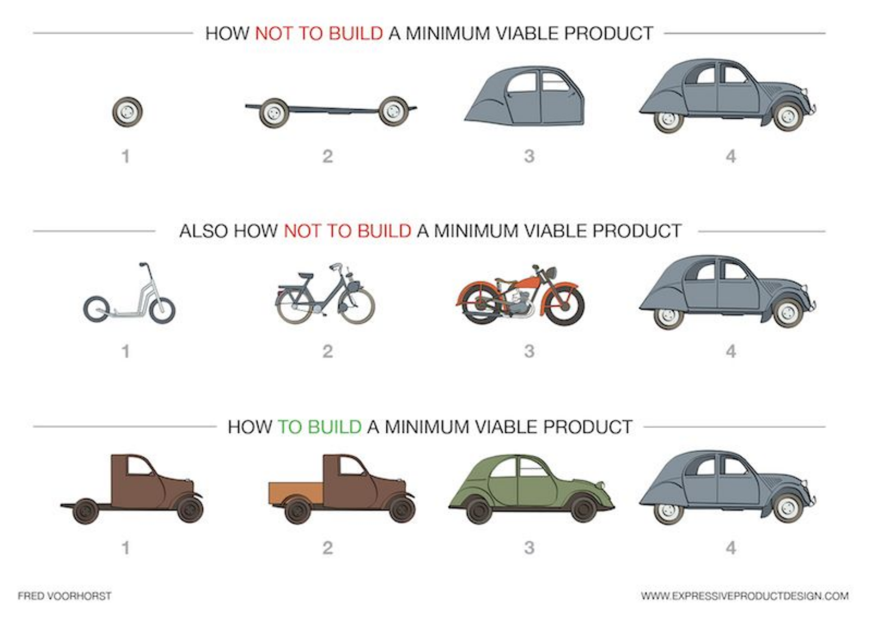
Prompt driven security

when Vibe Coding goes into production


I ONCE HEARD THAT THERE IS A PROGRAMMER
SENIOR
THAT WRITES CODE WITHOUT CHATGPT
JUNIORS

VIBE CODERS ANONYMOUS

Why?


HOW NOT TO BUILD A MINIMUM VIABLE PRODUCT
1
2
3
4
ALSO HOW NOT TO BUILD A MINIMUM VIABLE PRODUCT
1
2
3
4
HOW TO BUILD A MINIMUM VIABLE PRODUCT
1
2
3
4
FRED VOORHORST
WWW.EXPRESSIVEPRODUCTDESIGN.COM

Deploy to production

You have to take care of production

So what?

Injection flaws

IDORs and Authorization flaws

Hardcoded credentials / Credentials in source

Exposed infrastructure, db servers, data buckets

Supply chain attacks

- Old vulnerable packages
- non-existent packages, then added as malware

Ad-hoc rewriting

Architectural Blindness

Inconsistent results

Profit chasing

Data theft (legal or illegal)

Enshittification (save money)

Unaffordability

Retraining on AI slop

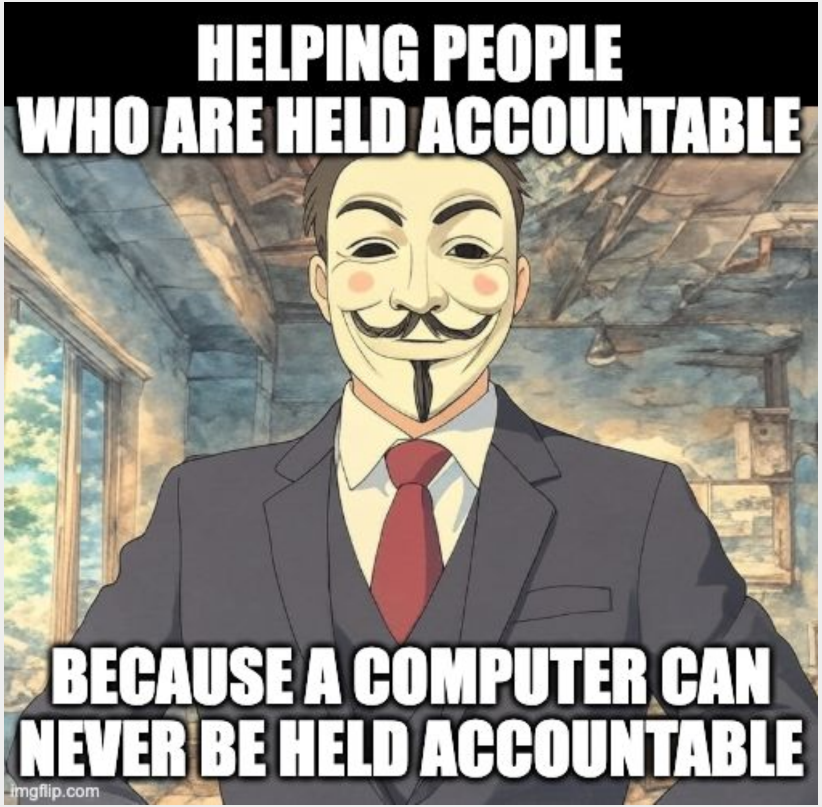
Vulnerable to well known attacks

Too much power and reach

/usr/bin/whoami

- CISO of Blue dot and Sourcico
- Trying to get people to do the right thing for better part of two decades
- Terrified of people chasing deadlines and being pressured into MVPs
- Many mistakes, incidents and a lot of stress

Email: b.spirovski@beyondmachines.net
Youtube: https://www.youtube.com/@spirovskib
LinkedIn: https://www.linkedin.com/in/spirovskibozidar/



Is this session for you?

Take accountability, or not walk away now

What my lawyers make me say*

Not a silver bullet: The guidelines in this presentation are not the ultimate solution for secure development

Not all that terrible: For a larger organization with proper processes, using LLMs is more nuanced

- Context-appropriate caution - not all AI coding carries equal risk
- Human oversight should scale with risk - more critical code = more human validation
- Defense in depth - always use multiple security layers, not single points of failure
- Continuous monitoring - security doesn't end at deployment
- Use professional judgment - experienced developers can assess context appropriately

You make the final decisions: You are the final decision maker for your code

* if I had lawyers

Context is king

Give the machine enough context

Tell it how to make a PB&J

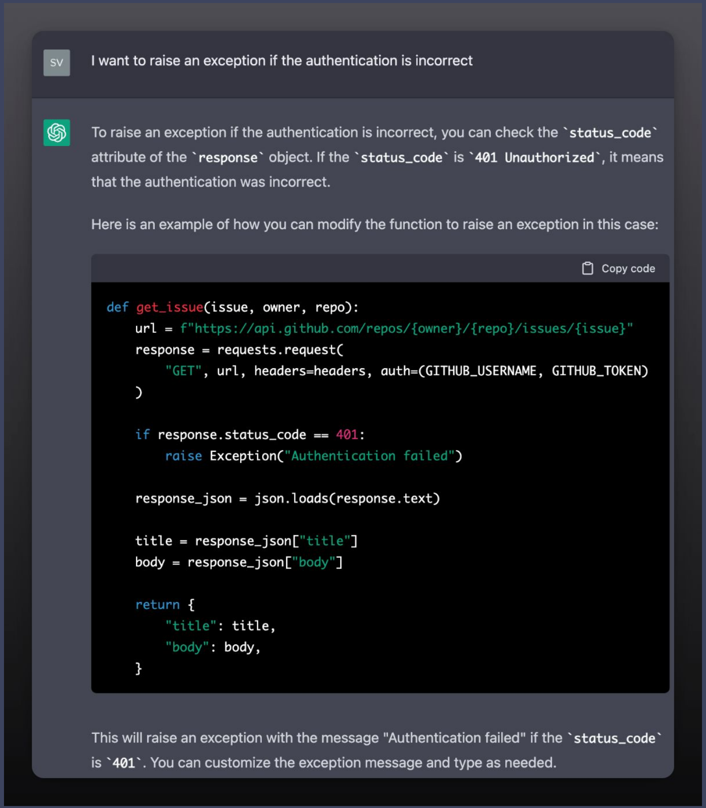
- Don’t be lazy in explaining. The LLM can’t read your mind
- Ask one thing from the LLM
- Before asking it, tell it any constraints that already exist

If you can’t tell it exactly, ask it how it thinks it should be made

- Explain the context, but ask it for summary assumptions or unclear elements
- Correct them, and input the result

Use this prompt: context_prompt.md

Find the difference between these two images



I want to raise an exception if the authentication is incorrect

To raise an exception if the authentication is incorrect, you can check the `status_code` attribute of the `response` object. If the `status_code` is `401 Unauthorized`, it means that the authentication was incorrect.

Here is an example of how you can modify the function to raise an exception in this case:

```
def get_issue(issue, owner, repo):
    url = f"https://api.github.com/repos/{owner}/{repo}/issues/{issue}"
    response = requests.request(
        "GET", url, headers=headers, auth=(GITHUB_USERNAME, GITHUB_TOKEN)
    )

    if response.status_code == 401:
        raise Exception("Authentication failed")

    response_json = json.loads(response.text)

    title = response_json["title"]
    body = response_json["body"]

    return {
        "title": title,
        "body": body,
    }
```

This will raise an exception with the message "Authentication failed" if the `status_code` is `401`. You can customize the exception message and type as needed.

Stop the bad habits

Let the tools and AI tell you:

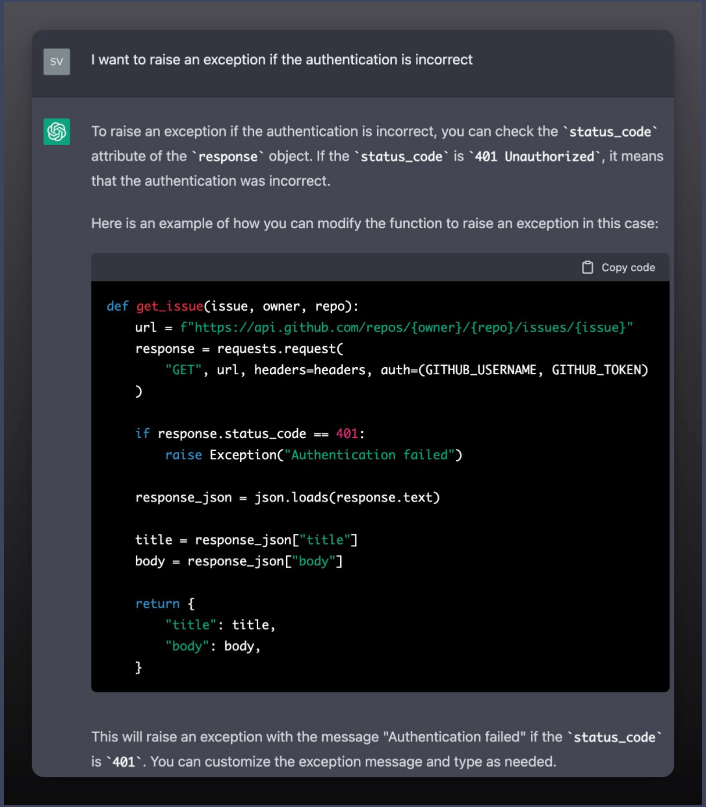
what the STACK?

- Pre-commit hook for secrets and code security and infrastructure as code
- Ask it for review on results on alerts
- Insist on line by line corrections!
- Check Dependabot!

Use this system prompt:
secure_code_prompt.md

- AI command amnesia
- Reiterate or start a new session

Find the differences between these two images



SV I want to raise an exception if the authentication is incorrect

To raise an exception if the authentication is incorrect, you can check the `` `status_code` `` attribute of the `` `response` `` object. If the `` `status_code` `` is `` `401 Unauthorized` ``, it means that the authentication was incorrect.

Here is an example of how you can modify the function to raise an exception in this case:

```
def get_issue(issue, owner, repo):
    url = f"https://api.github.com/repos/{owner}/{repo}/issues/{issue}"
    response = requests.request(
        "GET", url, headers=headers, auth=(GITHUB_USERNAME, GITHUB_TOKEN)
    )

    if response.status_code == 401:
        raise Exception("Authentication failed")

    response_json = json.loads(response.text)

    title = response_json["title"]
    body = response_json["body"]

    return {
        "title": title,
        "body": body,
    }
```

This will raise an exception with the message "Authentication failed" if the `` `status_code` `` is `` `401` ``. You can customize the exception message and type as needed.

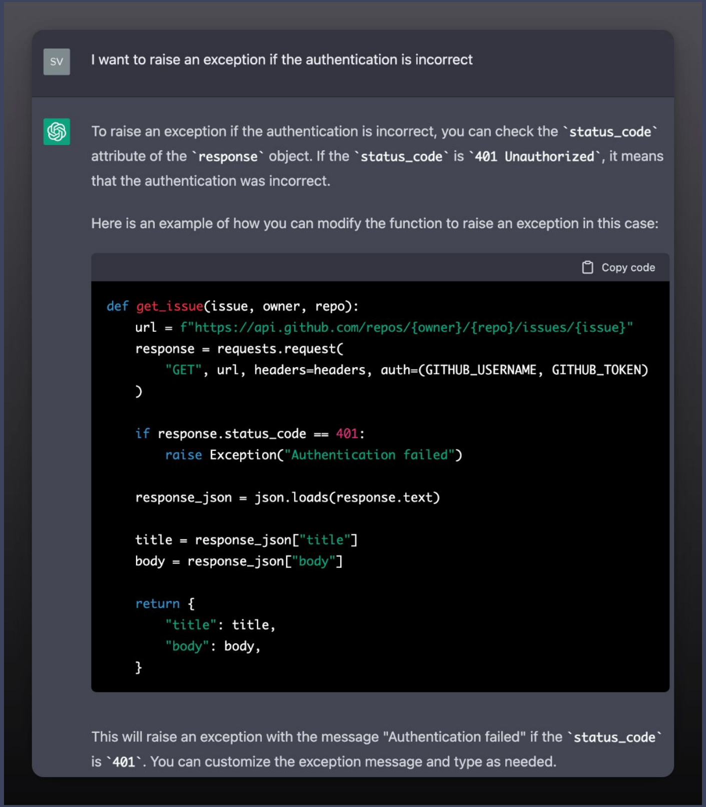
Stop the project from becoming an avalanche

- Ask for one item at a time
- Keep various features separate, so you can ask for one item
- Write part of the changes yourself
- Test the new features AND old features
- Never deploy immediately to production
- Don't give AI powers for deployment
  - If you do, MAKE SURE TO CHECK before and after
  - Remove permissions

Use this system prompt:
code_avalanche_prompt.md

- AI command amnesia
- Reiterate or start a new session

Find the differences between these two images



SV

I want to raise an exception if the authentication is incorrect

To raise an exception if the authentication is incorrect, you can check the `status_code` attribute of the `response` object. If the `status_code` is `401 Unauthorized`, it means that the authentication was incorrect.

Here is an example of how you can modify the function to raise an exception in this case:

```
def get_issue(issue, owner, repo):
    url = f"https://api.github.com/repos/{owner}/{repo}/issues/{issue}"
    response = requests.request(
        "GET", url, headers=headers, auth=(GITHUB_USERNAME, GITHUB_TOKEN)
    )

    if response.status_code == 401:
        raise Exception("Authentication failed")

    response_json = json.loads(response.text)

    title = response_json["title"]
    body = response_json["body"]

    return {
        "title": title,
        "body": body,
    }
```

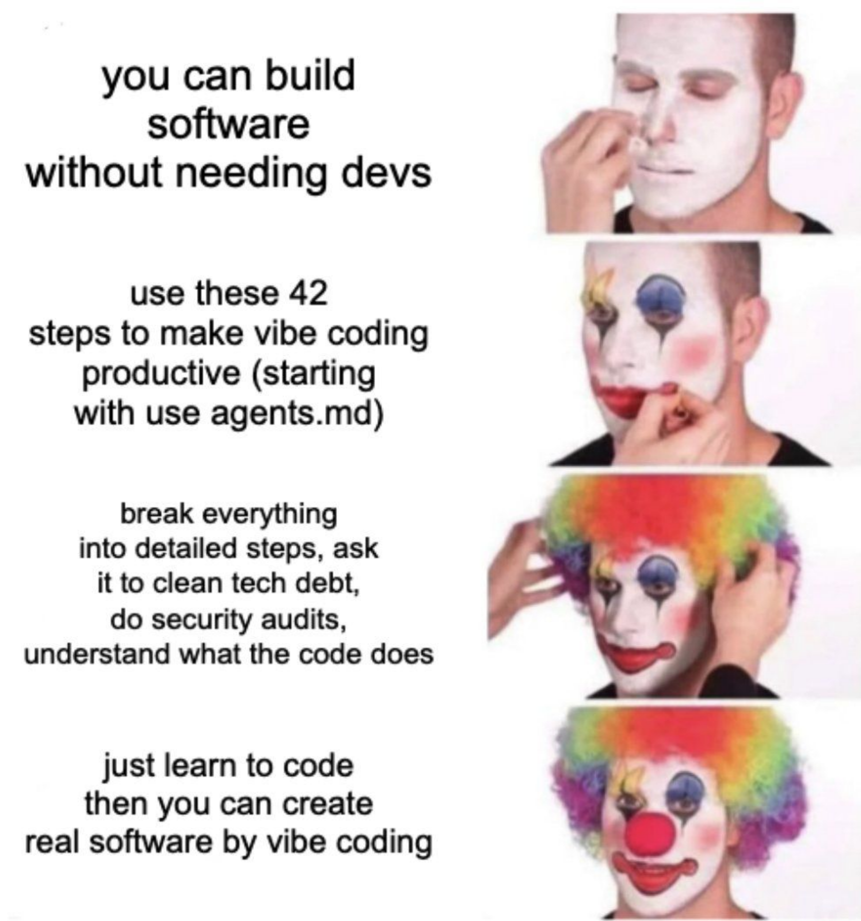
This will raise an exception with the message "Authentication failed" if the `status_code` is `401`. You can customize the exception message and type as needed.

Be aware this is a tool, dangerous and can fail

- NEVER give real API keys, passwords, credit cards, PII
- NEVER grant access over your entire data
- Check the Terms and Conditions at least quarterly. Use another AI if needed!
- What if you just licensed your company data or patent to them?
- Use local LLM

There is no prompt against the greed of the AI company

What you know and what you can do yourself can’t be taken away


you can build software without needing devs
use these 42 steps to make vibe coding productive (starting with use agents.md)
break everything into detailed steps, ask it to clean tech debt, do security audits, understand what the code does
just learn to code then you can create real software by vibe coding

Thank you
for coming to my
TED talk.

Let’s repeat

The machine can’t be held accountable

Help yourself so you aren’t

- **Context is king**
- **Trust but verify security**
- **Prevent code avalanche**
- **Protect sensitive data**
- **Be ready for random changes by vendors**
- **Check your exposed infrastructure AGAIN!!!**
- **Have a continuity plan**
- **Build up your skills**

Contact

Repo