

SFSCON 2024

Designing open source tools for Citizen Coders

Empower non-professional developers
to drive innovation!

Luca Rainone
Software Architect

7 November 2024
Bolzano, IT



The Citizen Coder: Profile

Citizen coder - Citizen developer

- 01** A citizen coder is someone with little experience in formal coding, IT, or computer science
- 02** Citizen coders work as hobbyist or as employee of non-tech companies.
- 03** The citizen coder is able to use a **low code** tool (and of course no code tools) in order to met specific business requirements



Beginner's All Purpose Symbolic Instruction Code

- 01 Long time ago, in the 1960s, John George Kemeny and Thomas Eugene Kurtz set out to make computer programming accessible.
- 02 They created a language simple enough for non-engineers to learn and use: **BASIC**
- 03 Their idea: anyone with interest should be able to instruct a computer.



8 Principles of BASIC

1. Be **easy for beginners** to use
2. Be a **general-purpose** programming language
3. **Allow extra features** to be added for experienced programmers.
4. Be **interactive**
5. Provide **clear and friendly error messages**
6. **Fast results** for small programs
7. Does **not require knowledge** about **computer hardware**
8. **Stop** the user from **editing the operating system**



Who am I and Why This Matters

Luca Rainone

- Software Architect, 20+ years of experience.
- Co-host of an Italian developer podcast. GITBAR
- 1,500+ member developer community.



The Citizen Coder Awakening

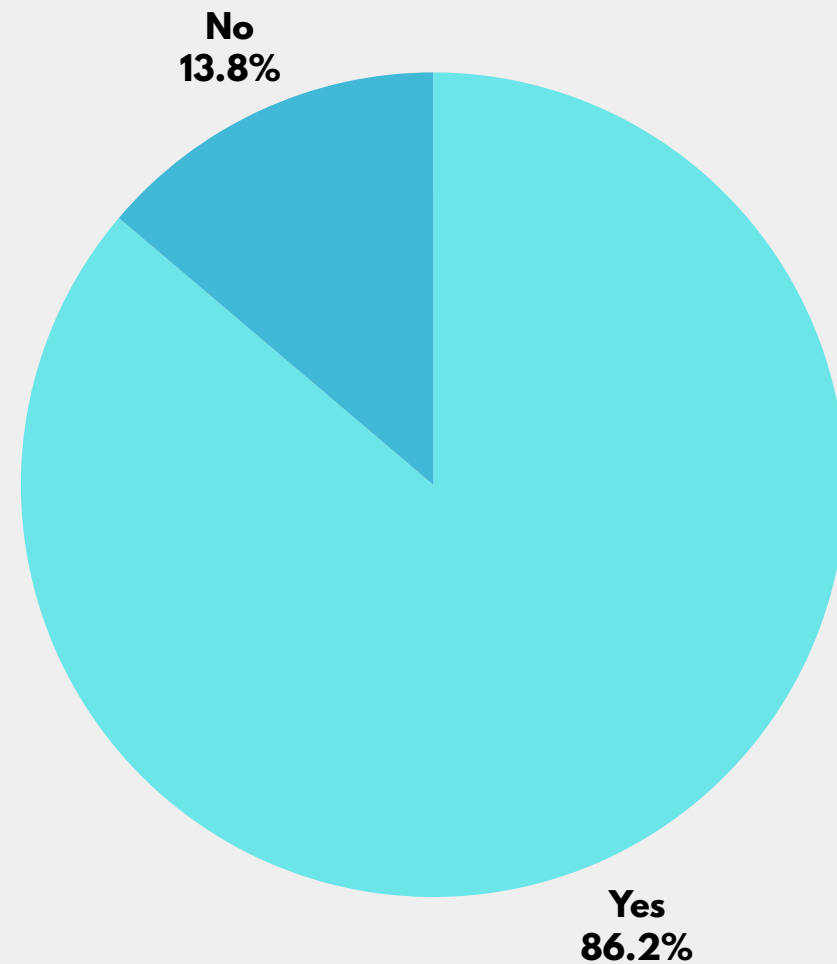
After a low-code podcast episode, I was contacted by an archaeologist. She asked precise technical questions, developing a tool for her work. Surprised to discover she wasn't a developer.

Sparked curiosity: How many “silent” citizen coders are out there?

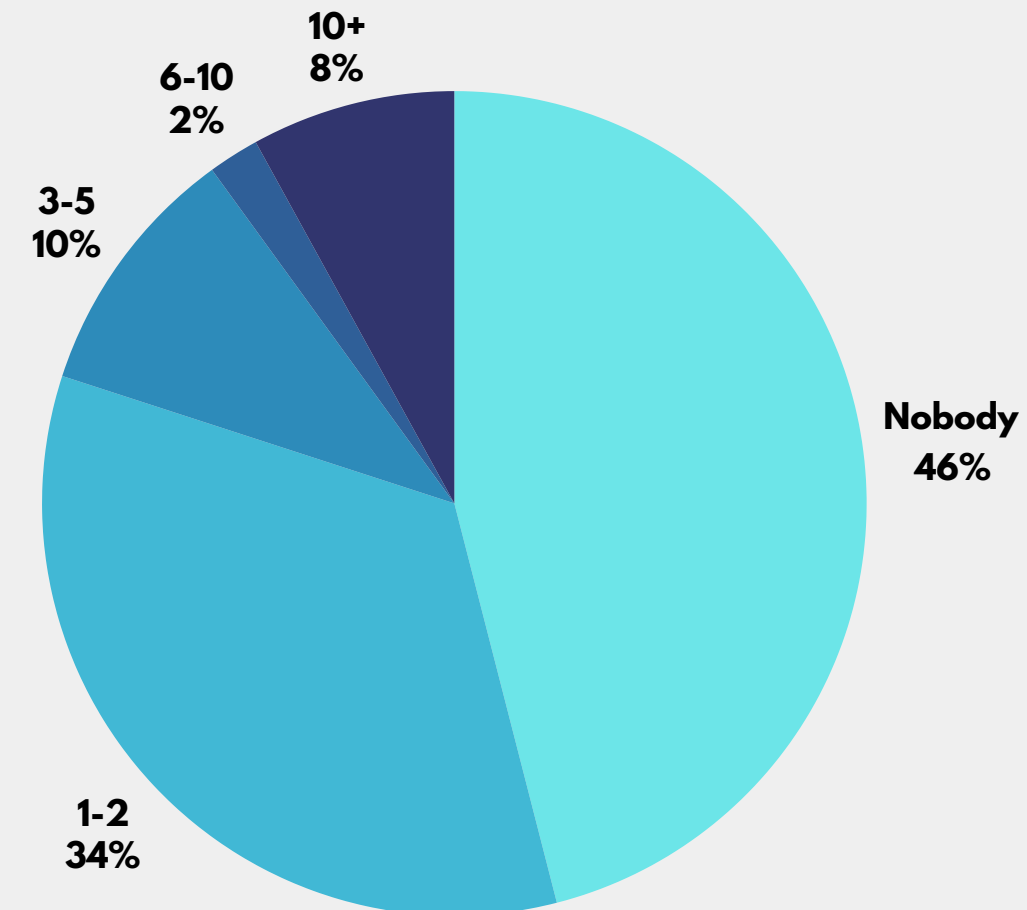


Survey in Gitbar community

1) Are you a professional developer?

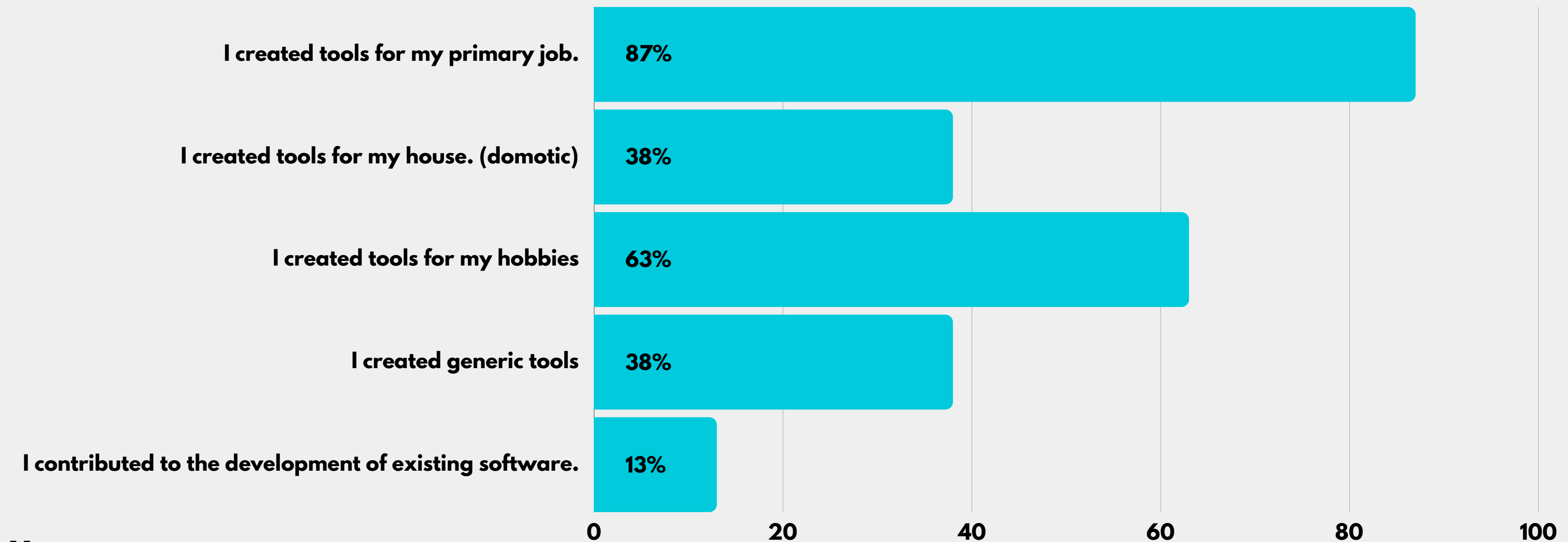


2) How many developers do you know who are not professionals and develop only in their free time or for utility in their primary job?



Survey in Gitbar community/3

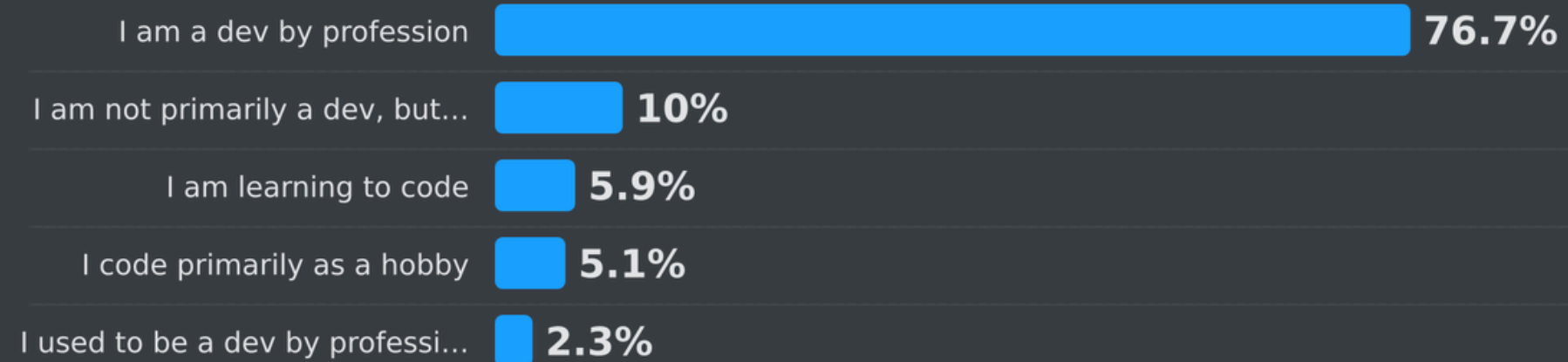
4) Thanks to my non professional IT skills (multiple choice):



2024 Developer Survey

Participants / undefined

Who participated in this survey



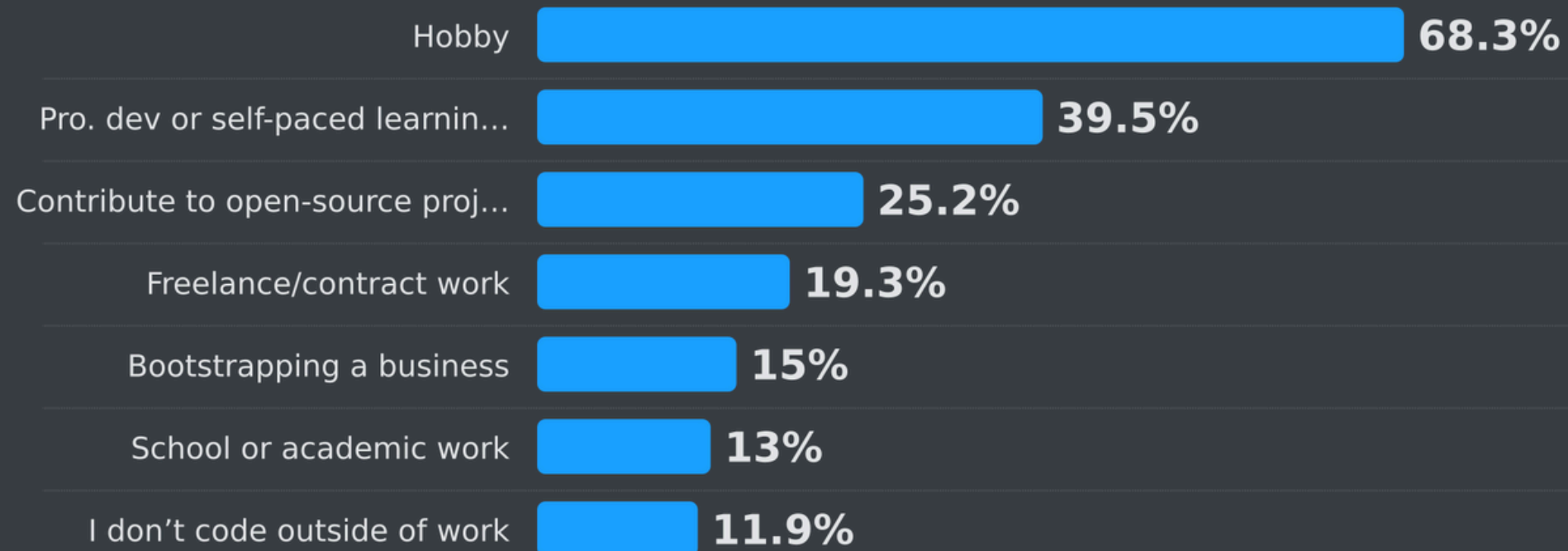
Source: survey.stackoverflow.co/2024
Data licensed under Open Database License (ODbL)



2024 Developer Survey

Coding outside of work / All Respondents

Coding outside of work



Source: survey.stackoverflow.co/2024
Data licensed under Open Database License (ODbL)



What we can do?



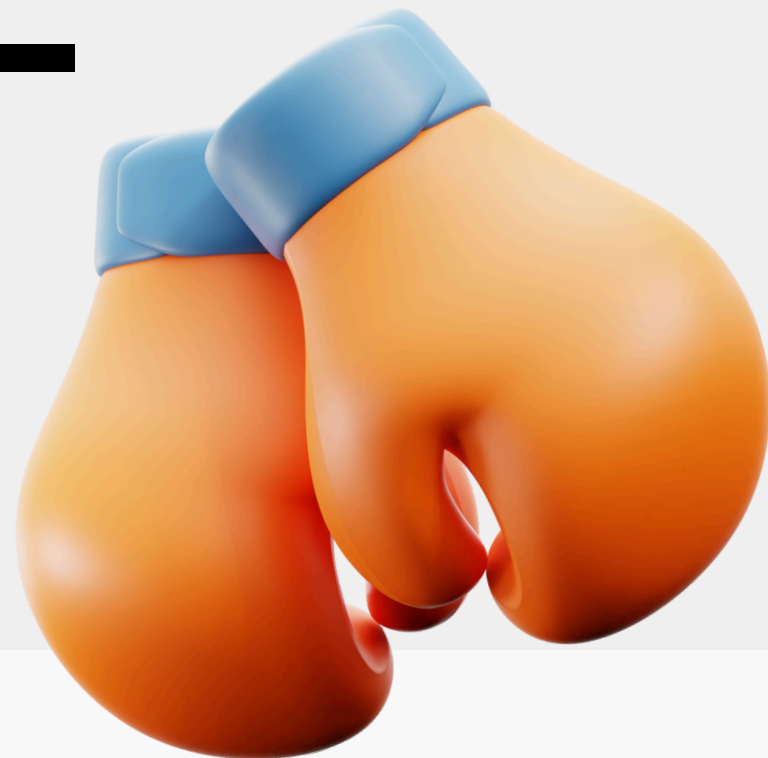
- **Recognizing Their Existence:**
We know they exist and are many.

What are our options? Three choices:

- Ignore Them
- Fight Them
- Value Them



Fighting Citizen Coders



- **When It Makes Sense:**

In complex libraries where high control and quality assurance are essential.

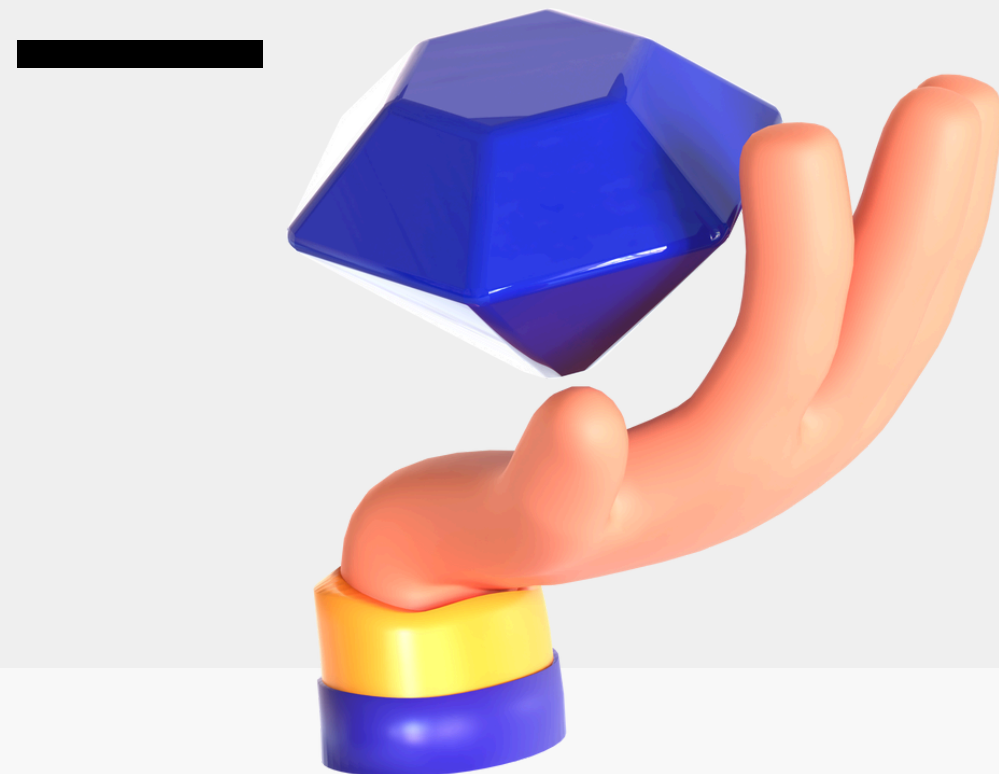
For projects requiring specialized expertise and rigorous standards.

- **How:**

Raising Barriers and Implementing strict controls: linters, conventional commits, strict pull requests.



Valuing Citizen Coders



- **Making Products Accessible:**
Design your projects to be welcoming and inclusive.
- **Engagement Strategies:**
Provide clear documentation, tutorials, and resources.
Encourage contributions by simplifying the onboarding process.



Ignoring Citizen Coders



What Does It Mean?

- Simply not engaging with or considering citizen coders in your projects.

Implications:

- Missed opportunities for innovation and collaboration.
- Loss of valuable insights and diverse perspectives.
- Potential time loss due to irrelevant or inadequate pull requests (PRs) that do not meet project standards.

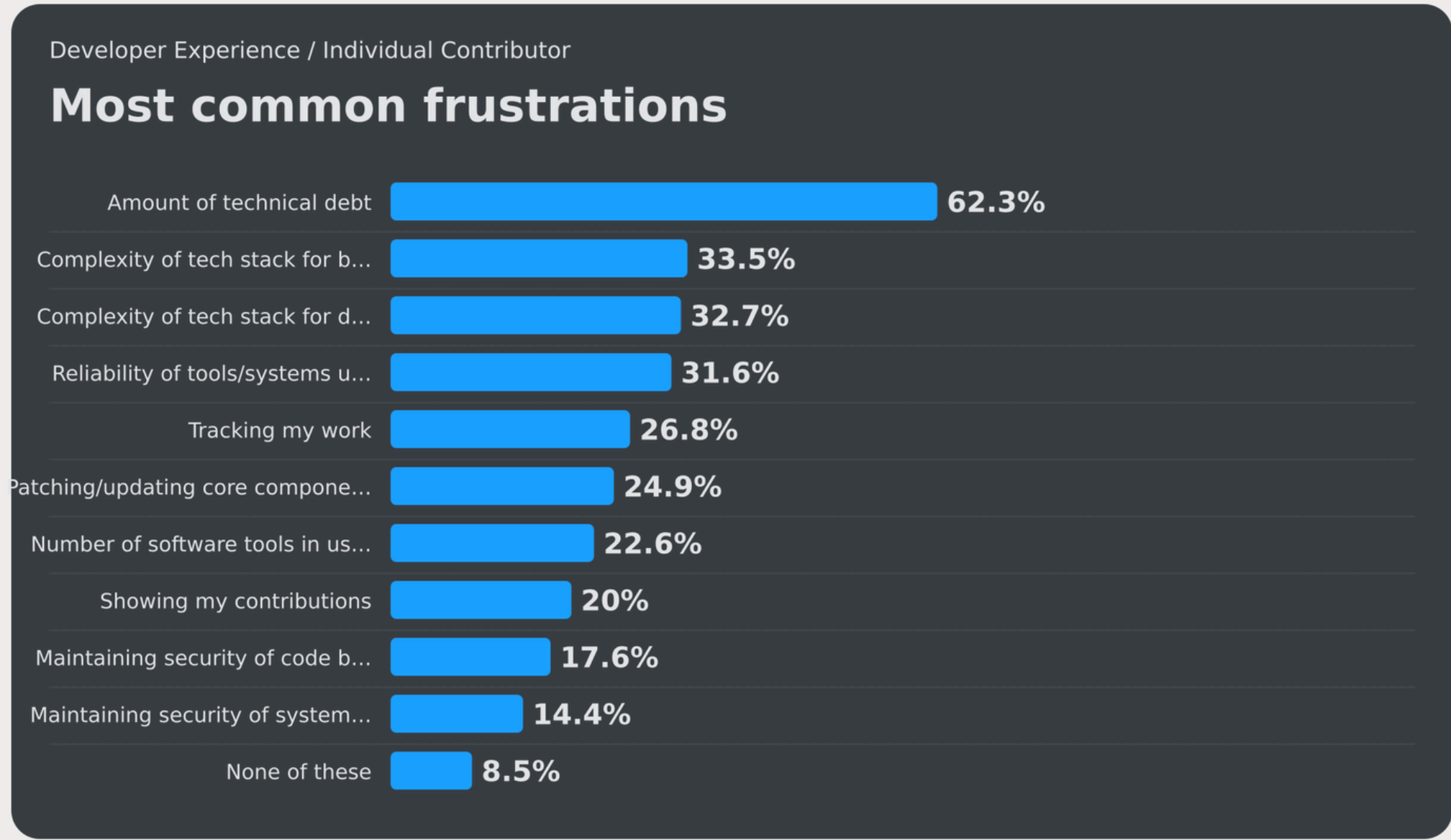


The Great Opportunity

By improving accessibility
for citizen coders, you enhance
the overall **D**eveloper **eX**perience.



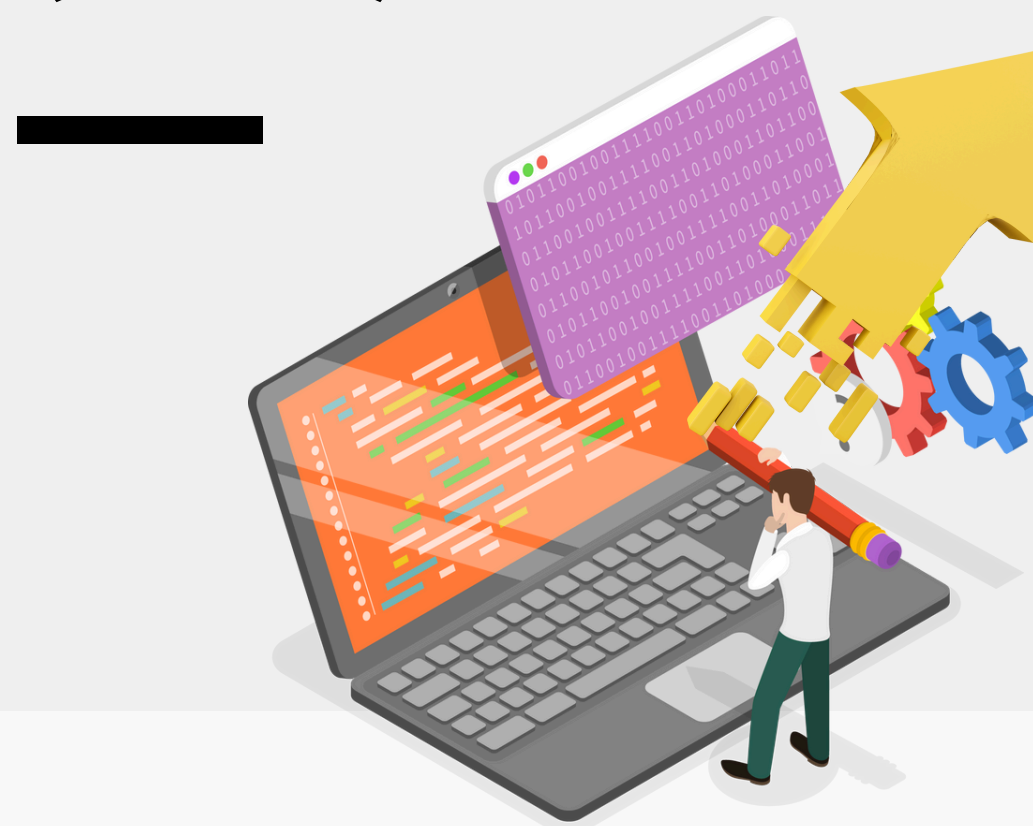
2024 Developer Survey



Source: survey.stackoverflow.co/2024
Data licensed under Open Database License (ODbL)



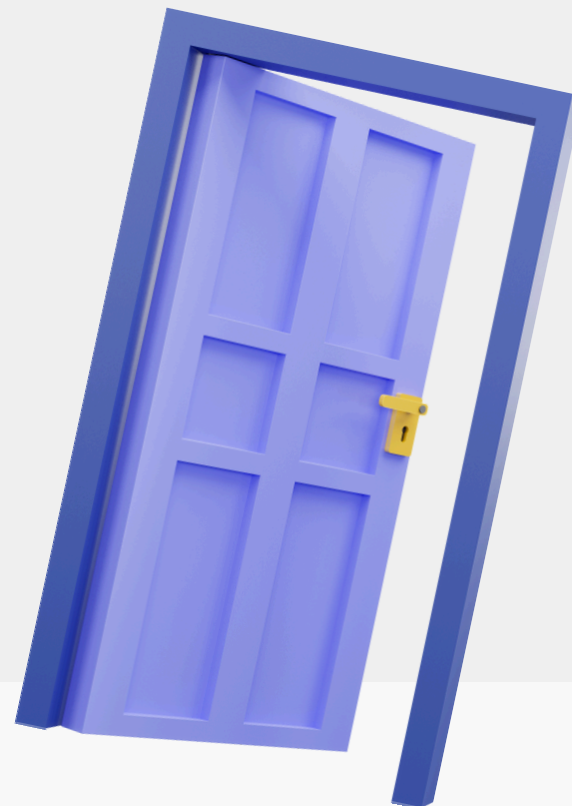
Developer Experience (DX)



- **What is Developer Experience (DX)?**
 - DX refers to the overall experience developers have when using a particular tool or contributing to a project.
- **Importance of Maximizing DX:**
 - Enhancing DX allows citizen coders to contribute effectively to open-source projects.
- **Inclusion of Professional Developers:**
 - Engage developers from various tech stacks to enrich collaboration and knowledge sharing.



Making Your Project Accessible



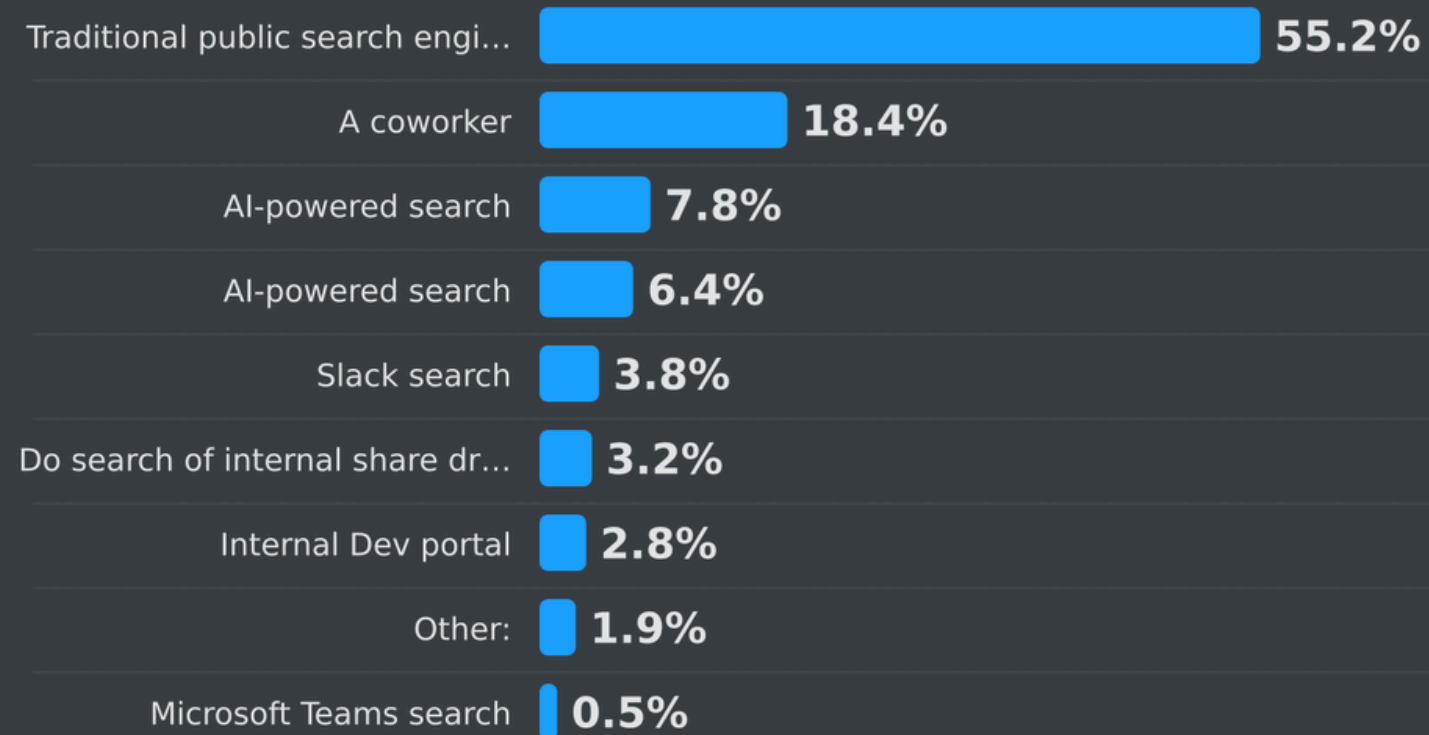
- **Separate Complexity:** Break down tasks into manageable components.
- **Apply BASIC Principles:** Keep interfaces simple and user-friendly.
- **Use a Plugin/Component Architecture:** Allow **easy integration** and customization.
- Provide **Comprehensive README Files:** Offer clear documentation to guide users through the project.



2024 Developer Survey

Developer Experience / Individual Contributor

Resources for technical questions at work



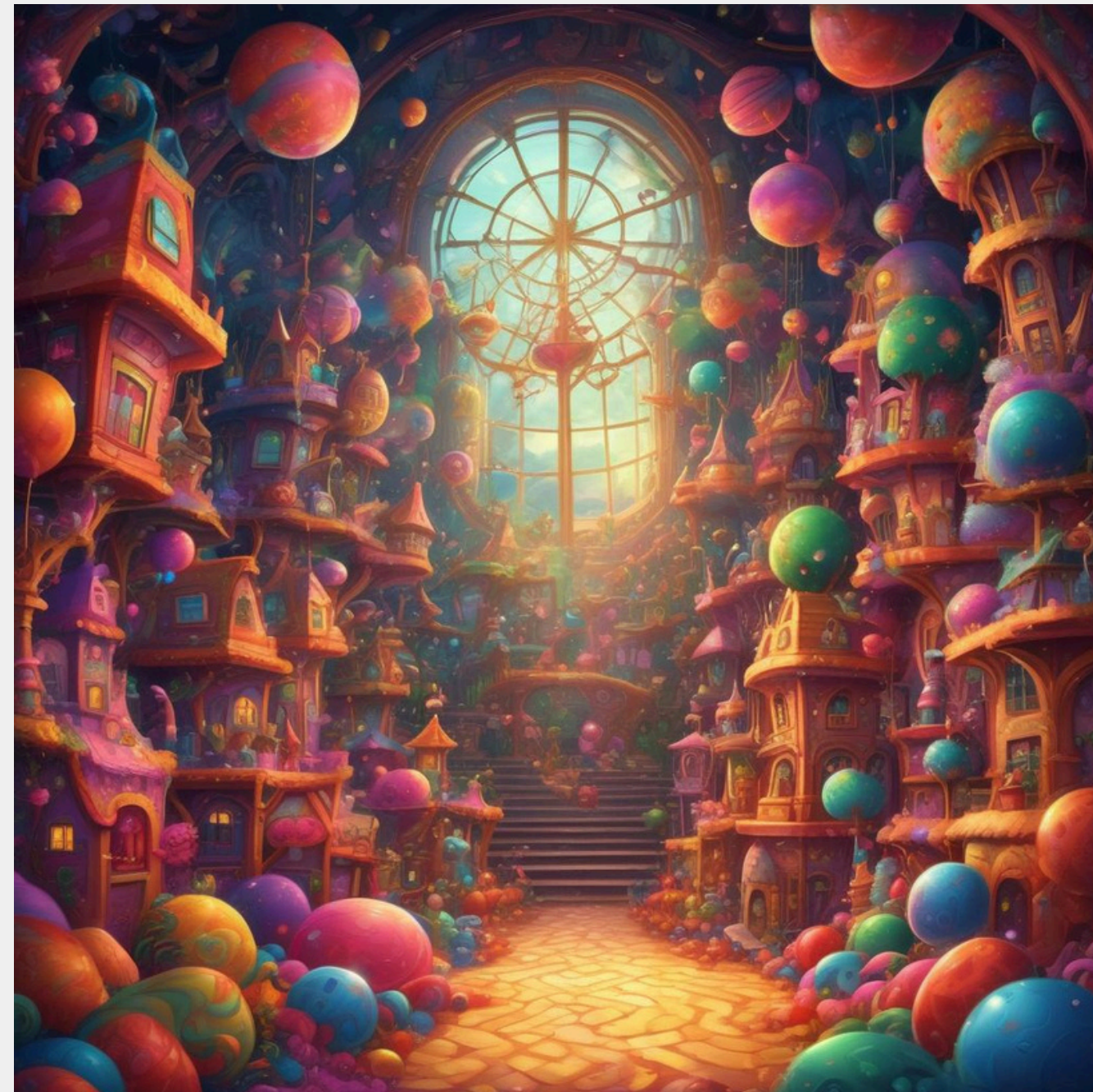
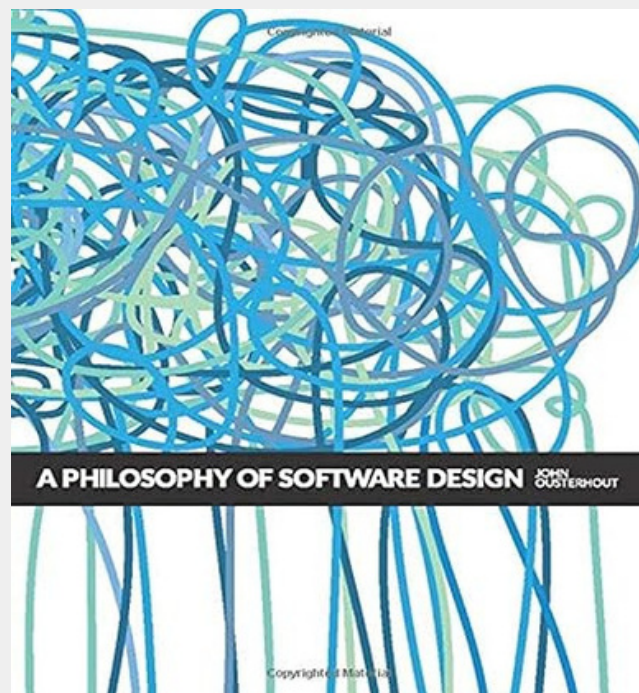
Source: survey.stackoverflow.co/2024
Data licensed under Open Database License (ODbL)



Designing open source tools for Citizen Coders

SFSCON 2024

Toyland / Paese dei Balocchi



<https://davinci.ai>


 othneildrew / Best-README-Template

 Fork 23k

 Star 14.2k

 firstcontributions / first-contributions

 Fork 79.6k

 Star 45.4k

 plantuml-stdlib / C4-PlantUML



Build your Readme



othneildrew / Best-README-Template



Fork 23k



Star 14.2k



firstcontributions / first-contributions



Fork 79.6k



Star 45.4k

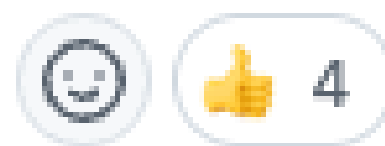


Involve people

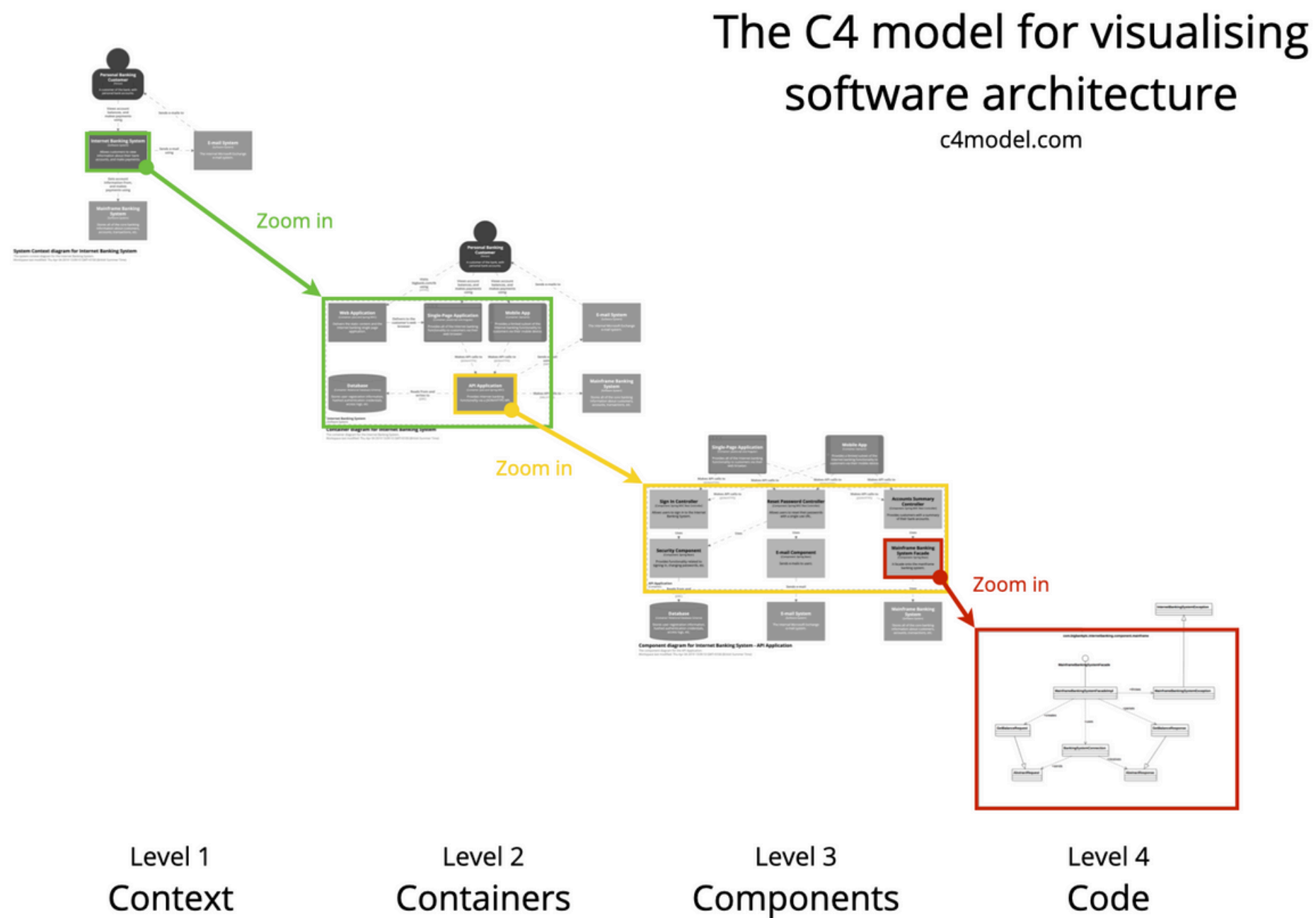


- **“Would you you like to work on this and send a pull request?”**
- **“Improvements to docs PRs are always welcome.”**
- **“A PR would be likely welcomed.”**
- **“Thanks for reporting! Would you like to send a Pull Request to address this issue? Remember to add unit tests.”**

Sounds good, yeah I can work on it!



Explain Your Project



plantuml-stdlib / C4-PlantUML



C4-PlantUML

index.puml x

```

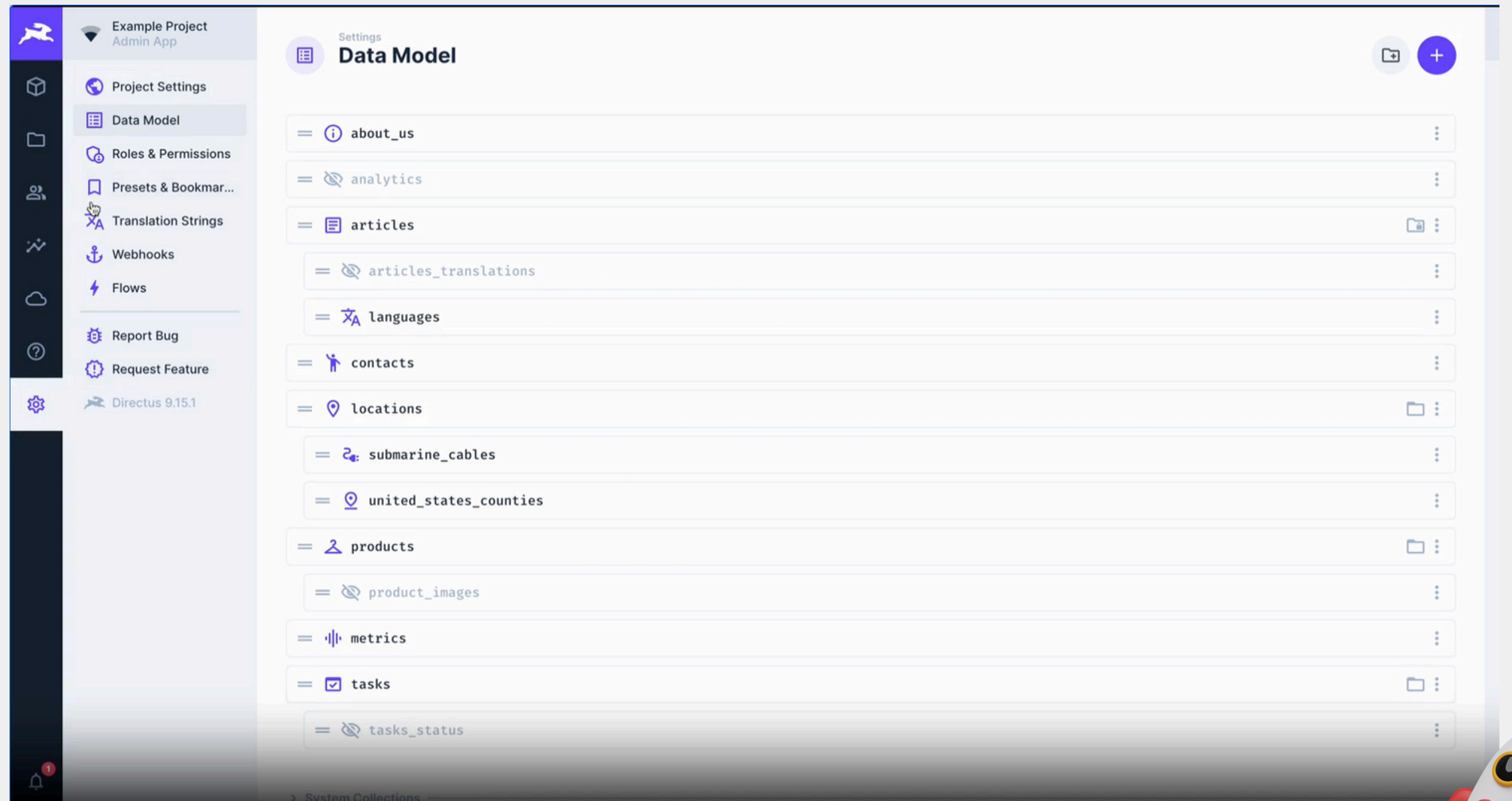
1 @startuml C4/index
2 !include C4-PlantUML/C4_Container.puml
3 HIDE_STEREOTYPE()
4
5
6 Container(client, "client", "App")
7 Container(gateway, "API GATEWAY", "aws")
8 Container(auth, "Auth lambda", "custom Authorizer")
9 Container(lambdas, "Lambdas\nCoupon", "get /network\n /sa
10 ContainerDb(dynamo, "DynamoDB", "es: tables Cards-prod-*")
11
12 client->gateway:1\nauthorization
13 client<-gateway:6
14 auth<--gateway:2
15 auth-->gateway:3
16 auth->dynamo:satellite\nsecret\nkey
17 gateway->lambdas:4
18 gateway<-lambdas:5
19 dynamo<-->lambdas
20 @enduml
                
```

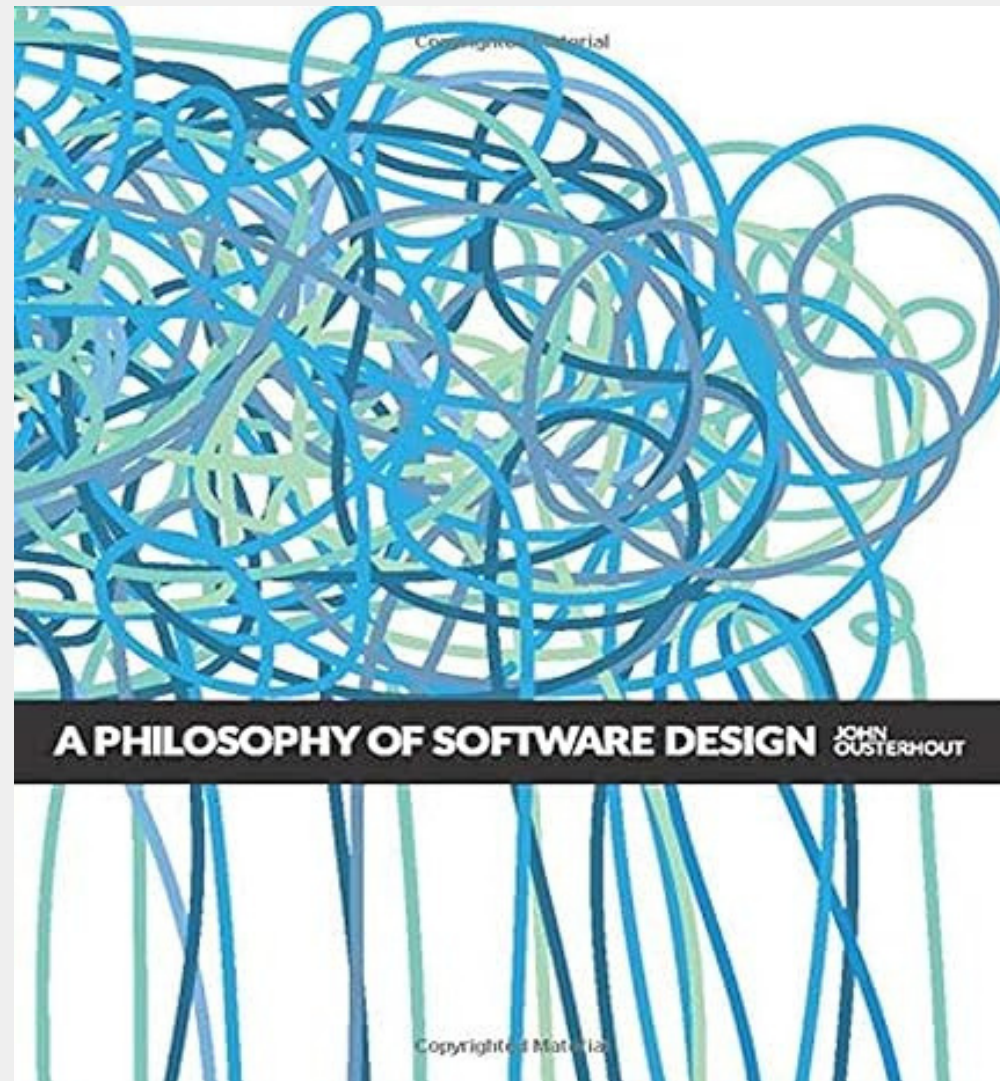
```

graph TD
    client[client [App]] -- 1 authorization --> gateway[API GATEWAY [aws]]
    gateway -- 6 --> client
    gateway -- 2 --> auth[Auth lambda [custom Authorizer]]
    auth -- 3 --> gateway
    auth -- "satellite secret key" --> dynamo[(DynamoDB [es: tables Cards-prod-*])]
    dynamo <--> lambdas[Lambdas Coupon [get /network /satellite /card]]
    gateway -- 4 --> lambdas
    lambdas -- 5 --> gateway
                
```



A good approach to low-code





A Philosophy of Software Design

John Ousterhout

It's all about complexity!

“Complexity is caused by two things:
dependencies and **obscurity**”



SFSCON 2024

Thank You

Feel Free to keep in touch with me and have some chat =)

Luca Rainone

