

>>

GenAI van uit een nieuw perspectief: Van XML naar TypeScript

Door Jurre Brandsen en Elise Lems



USE CASE

CODE GENERATION

RESULT

520 DAYS DOWN TO 110 DAYS

Wie zijn wij?

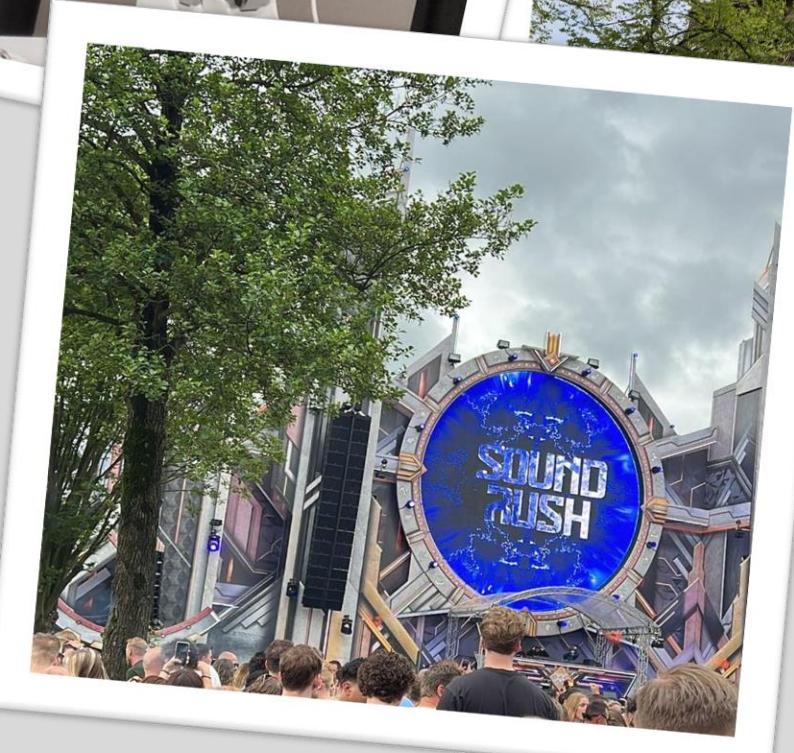
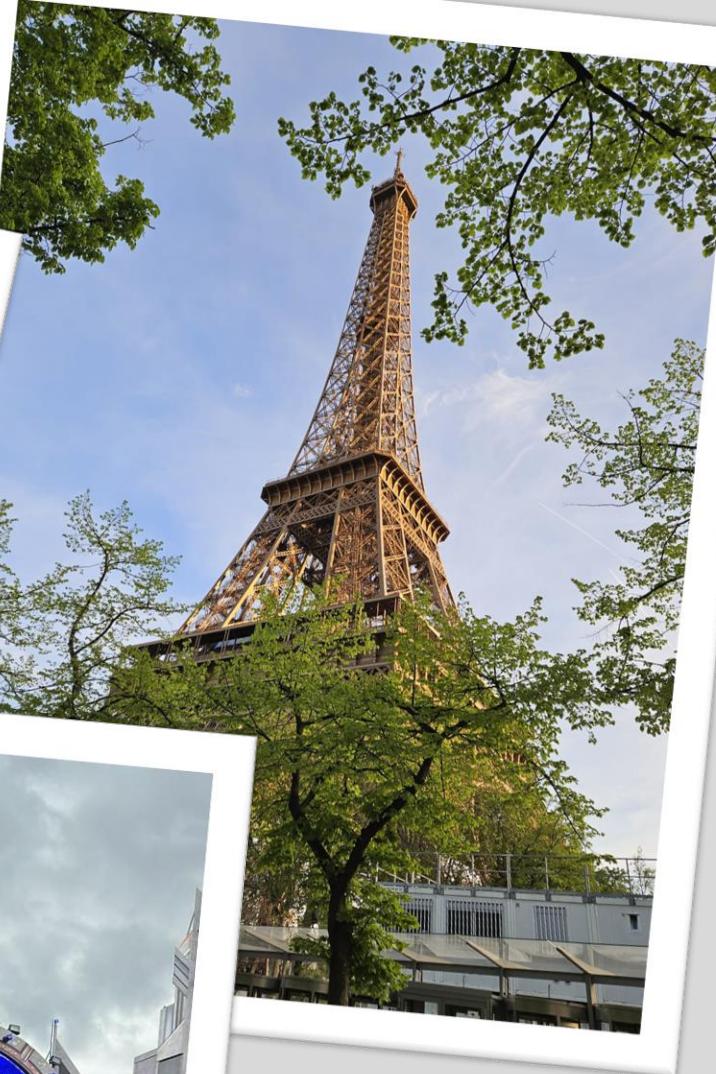
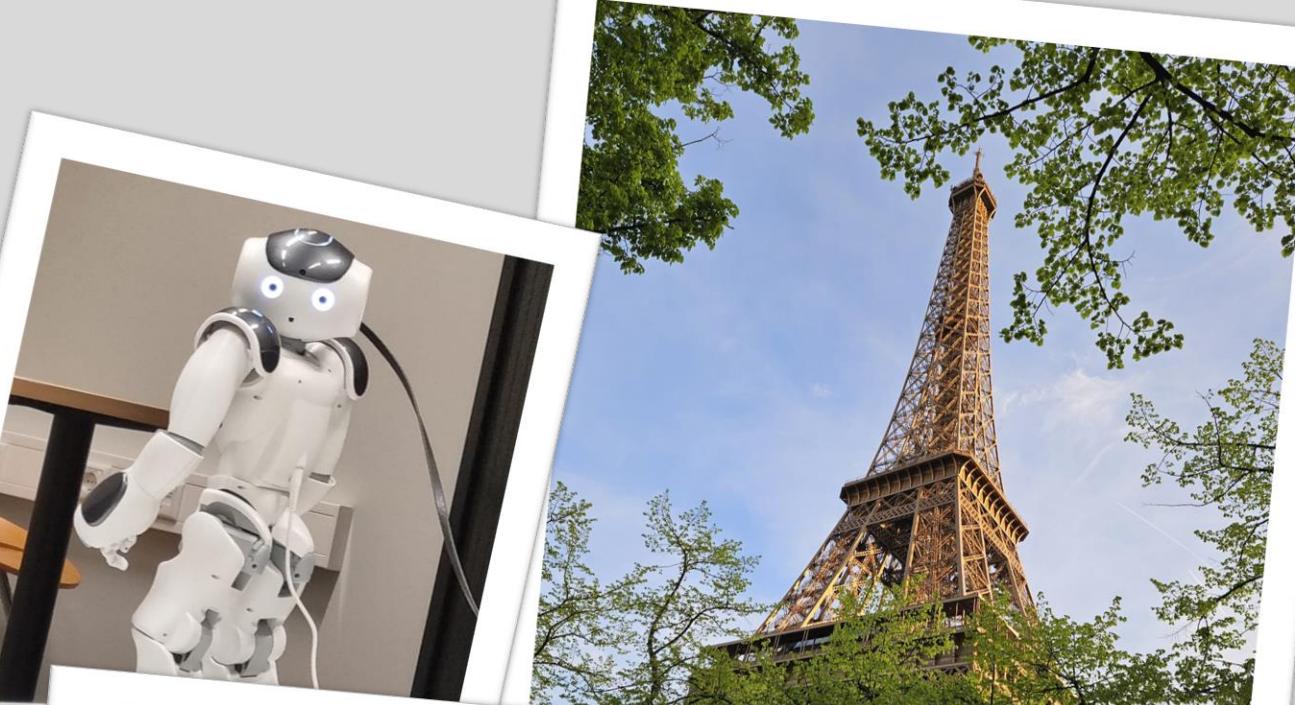




Elise Lems

Data Scientist en Data Engineer @ BUI

[LinkedIn.com/in/elise-lems](https://www.linkedin.com/in/elise-lems)



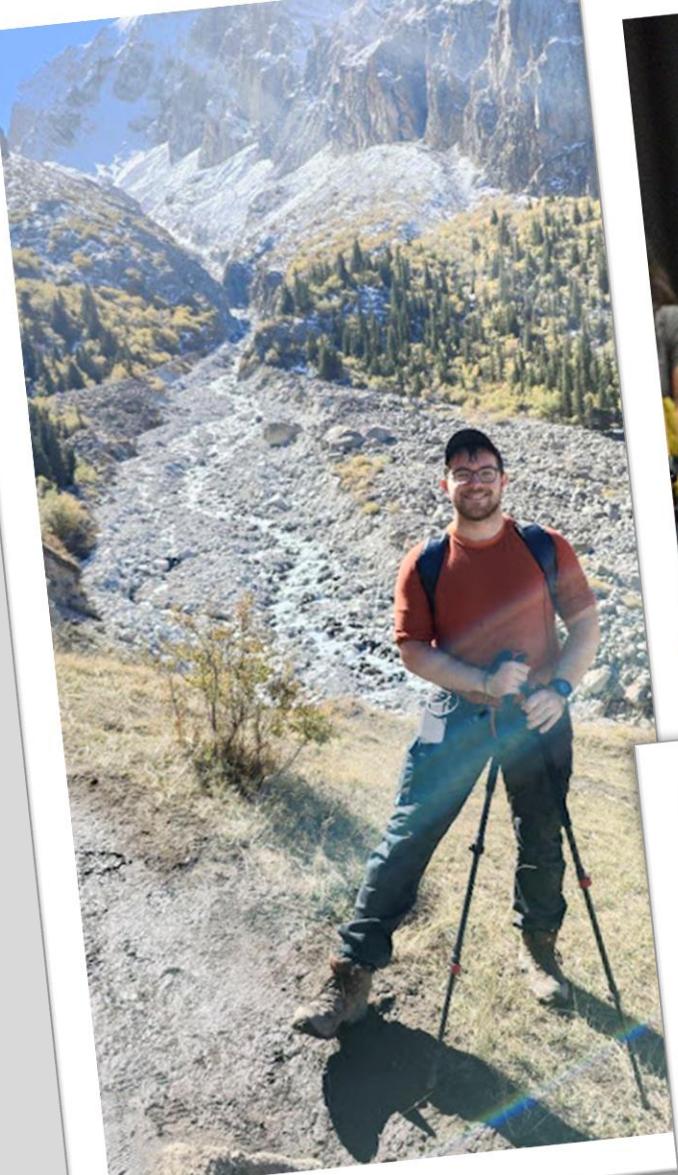
infoSupport
Solid Innovator



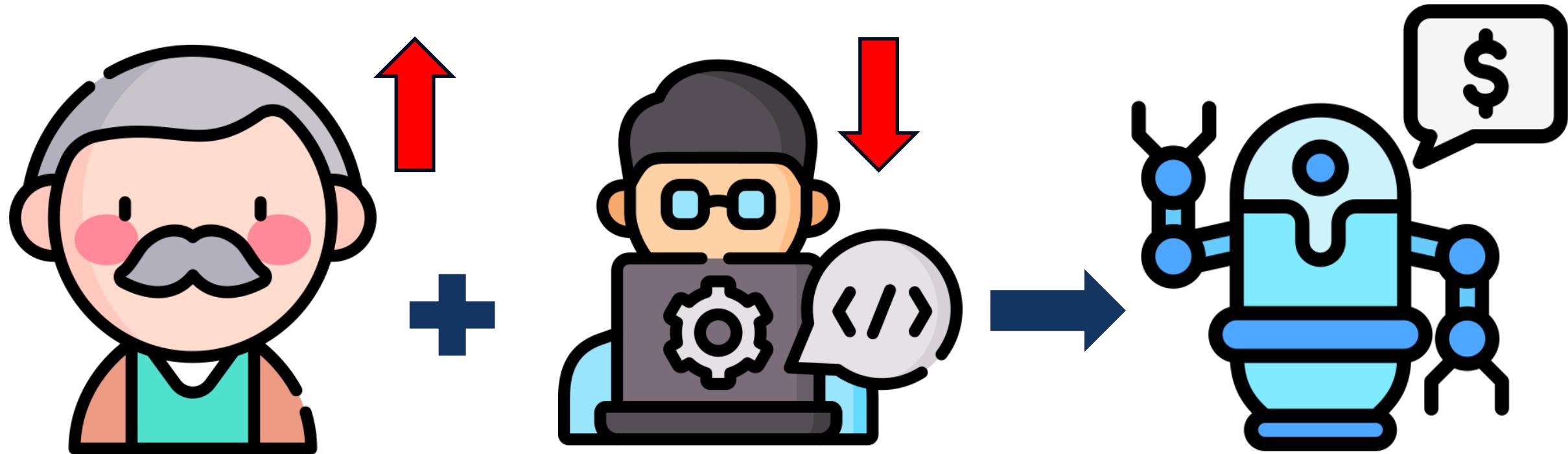
Jurre Brandsen

Software Engineer @ BUF

[LinkedIn.com/in/jurrebrandsen](https://www.linkedin.com/in/jurrebrandsen)



Verandering in de markt: modernisatie

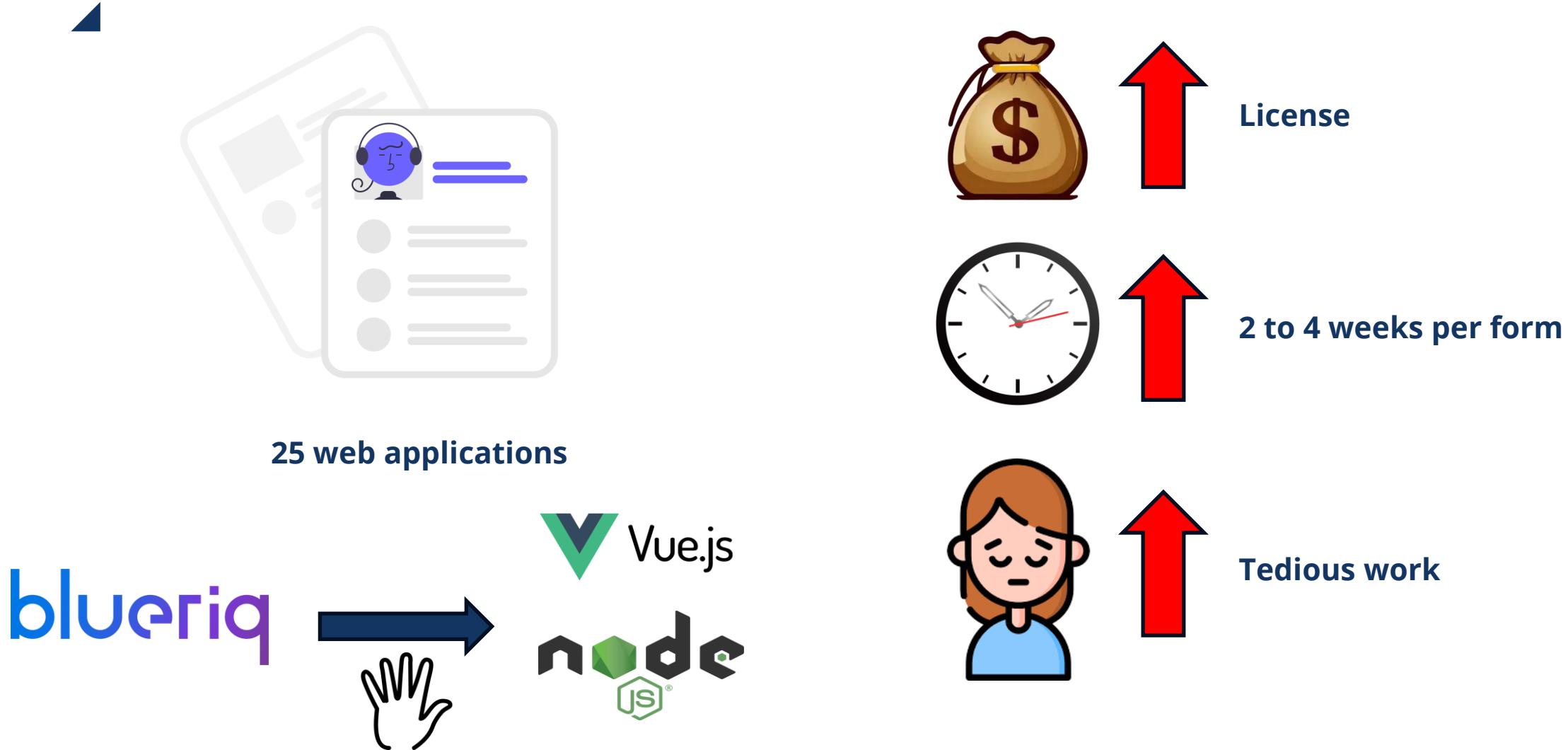


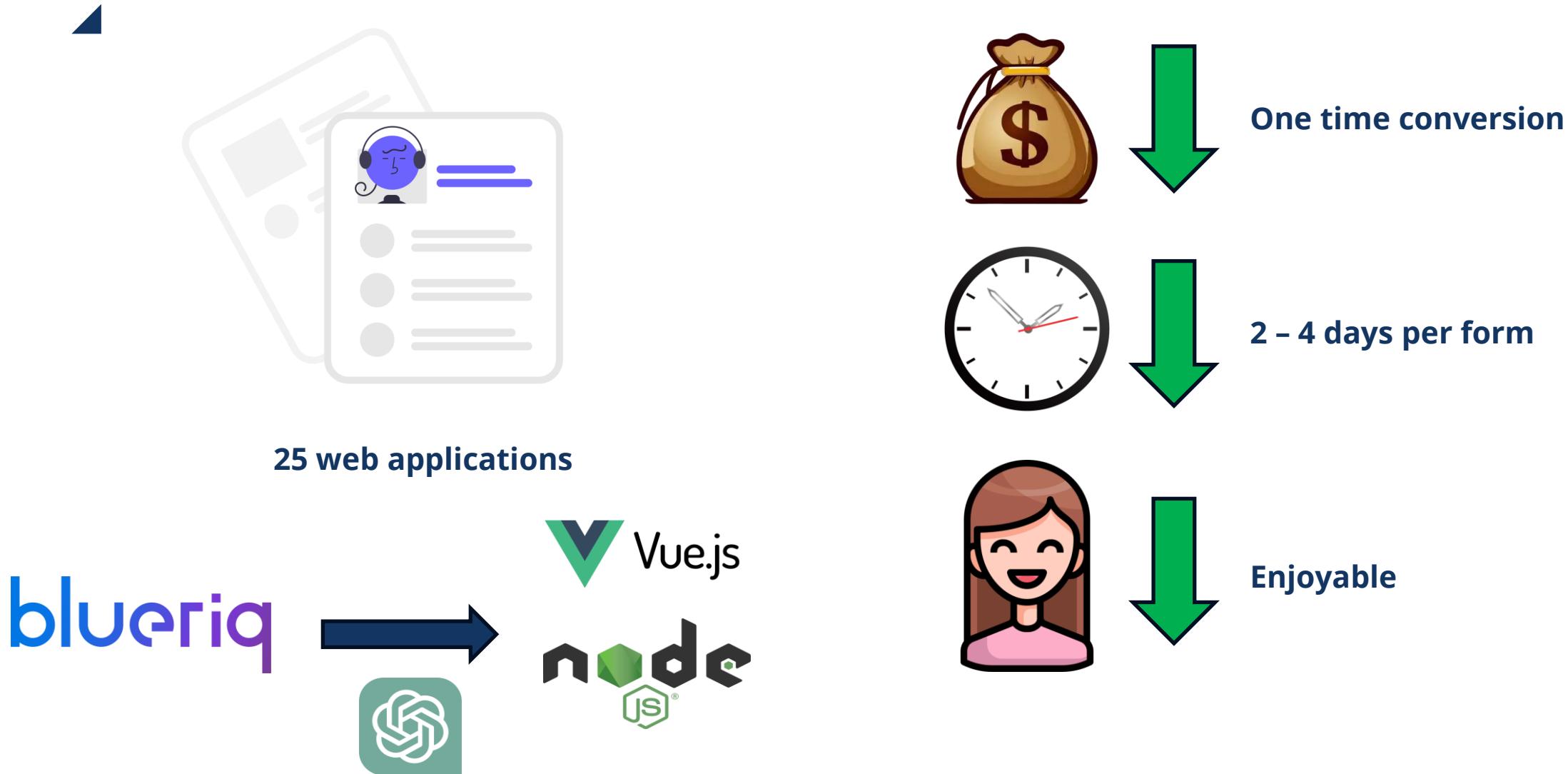
PROOF-OF-CONCEPT

PROOF-OF-CONCEPT

Can we prove that it is possible to convert Blueriq web pages to a modern variant in TypeScript and (optionally) VueJS within a limited time frame?







LOW CODE



One time conversion →

HIGH CODE

TypeScript

Node.js

Vue.js

Documentation

Automated Tests

Azure OpenAI Service

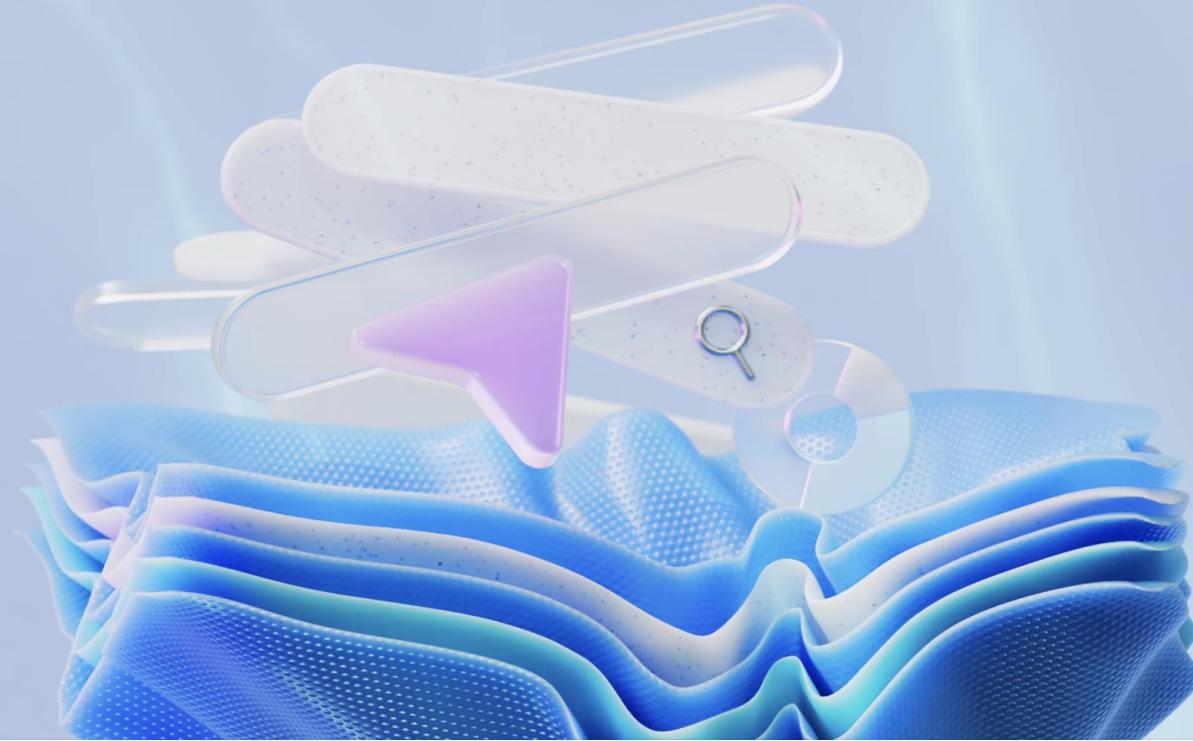
Build your own copilot and generative AI applications



Connect your own data, call functions, and improve workflow with language and image models

[Try Azure for free](#)

[Apply for access](#)



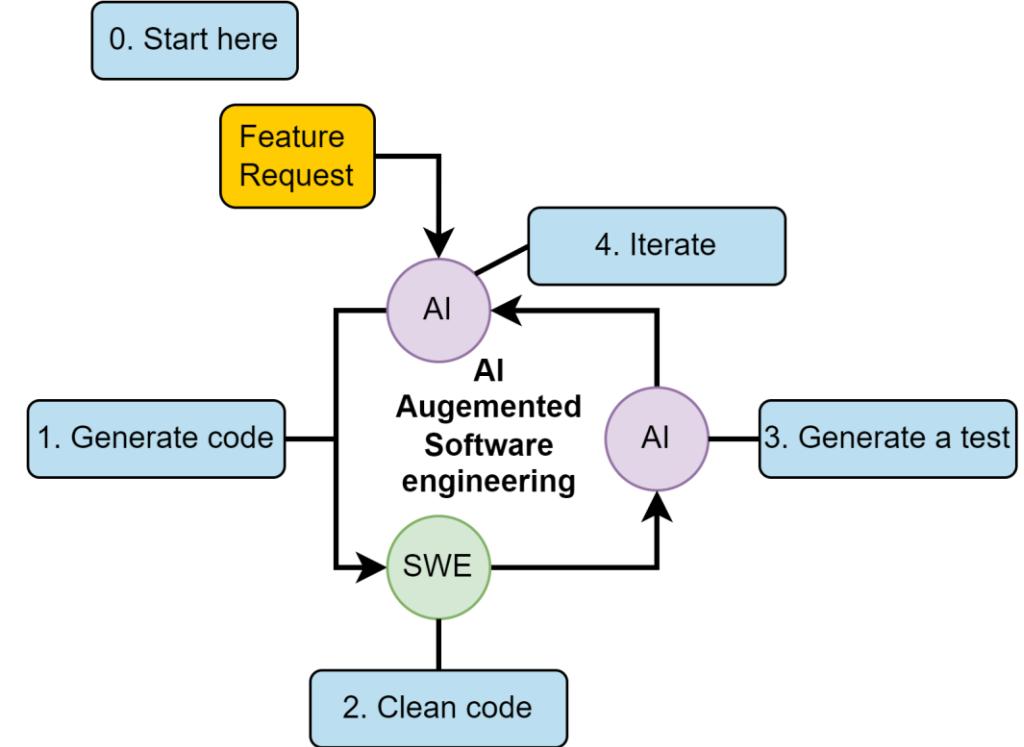
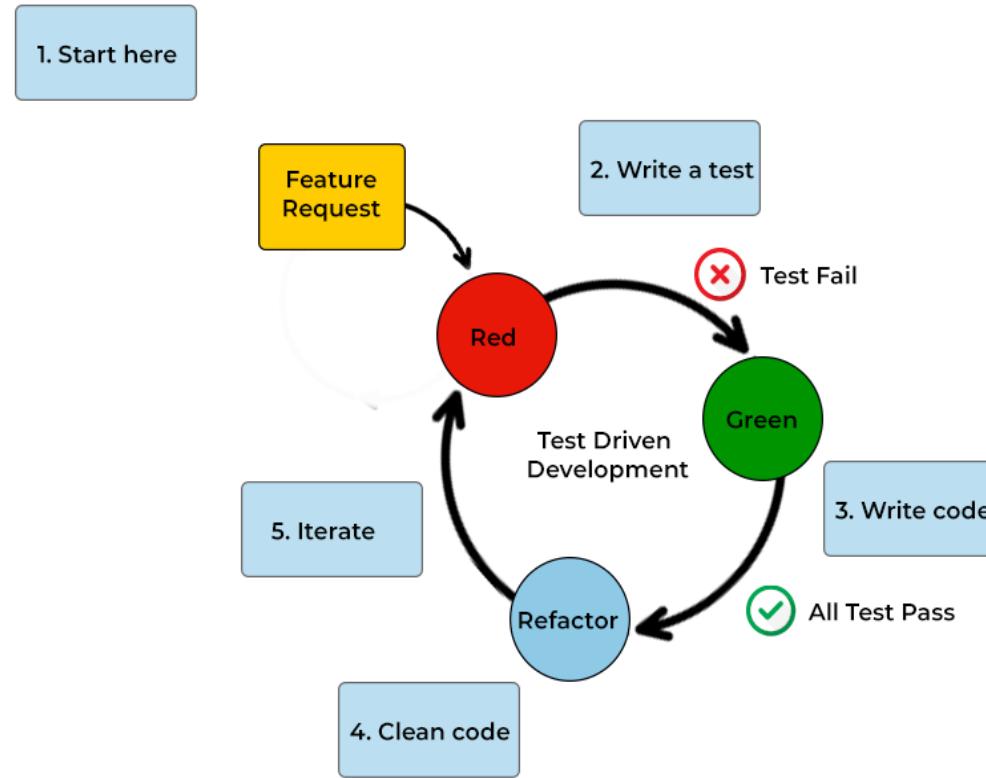
LangChain



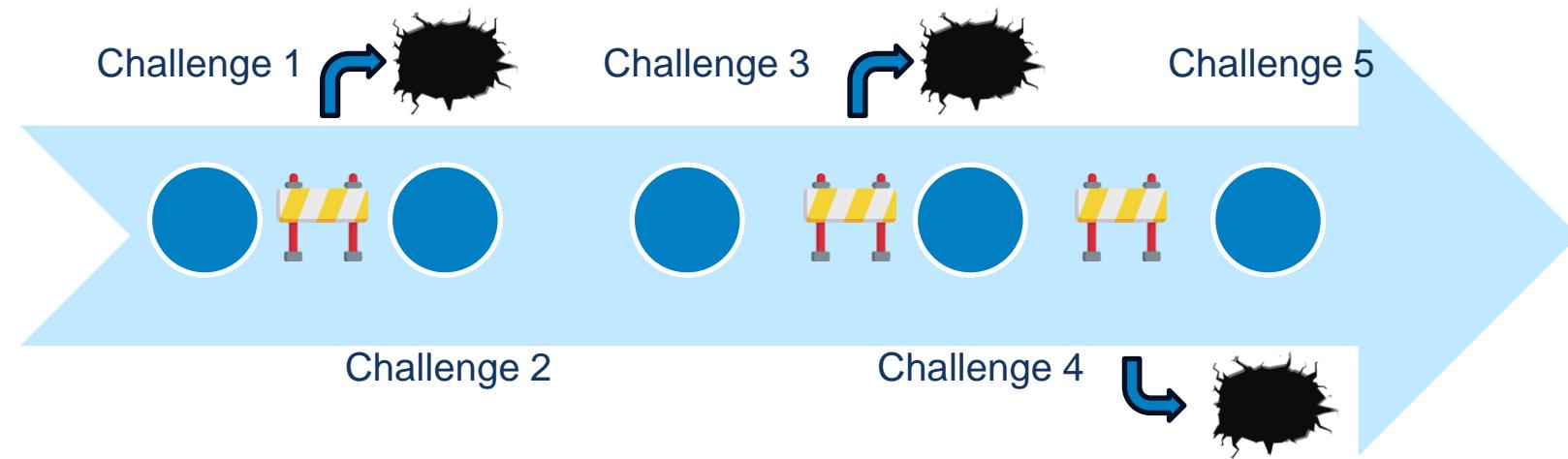
No AI-developer, without the developer

- + Parsing + Scripting
- + Human-in-the-Loop + Software development

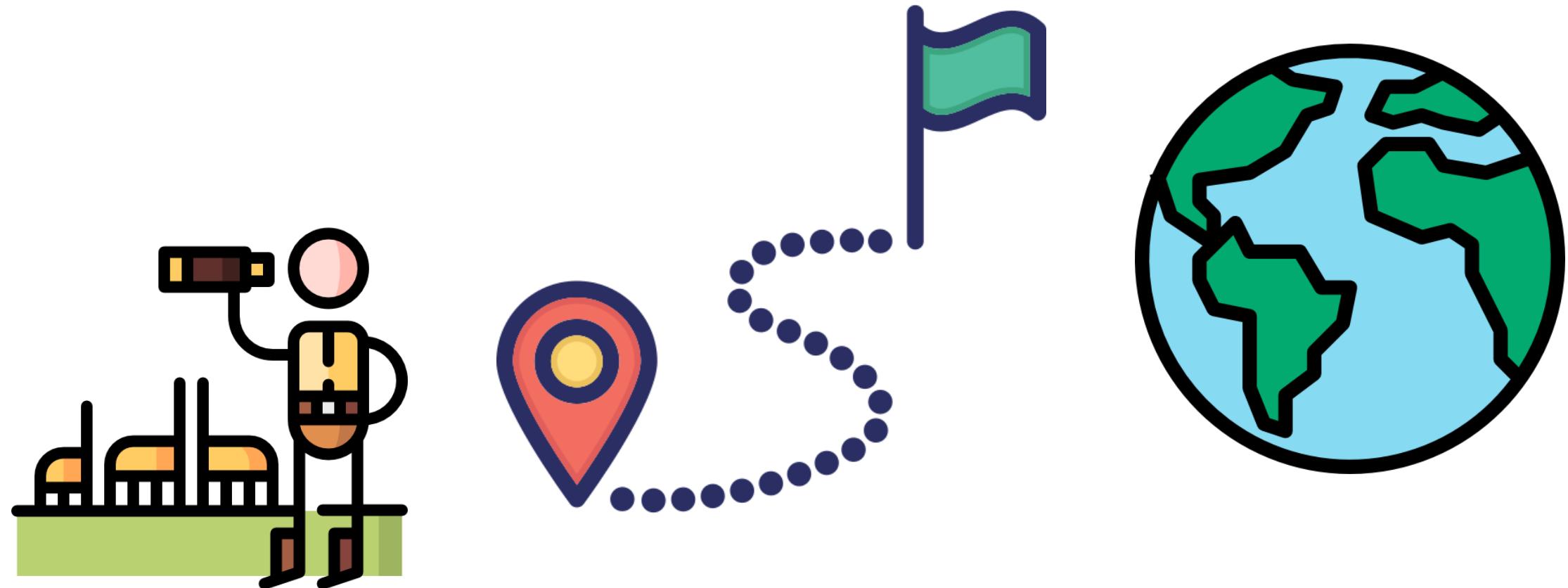
Pipelines TDD and AI-enhanced

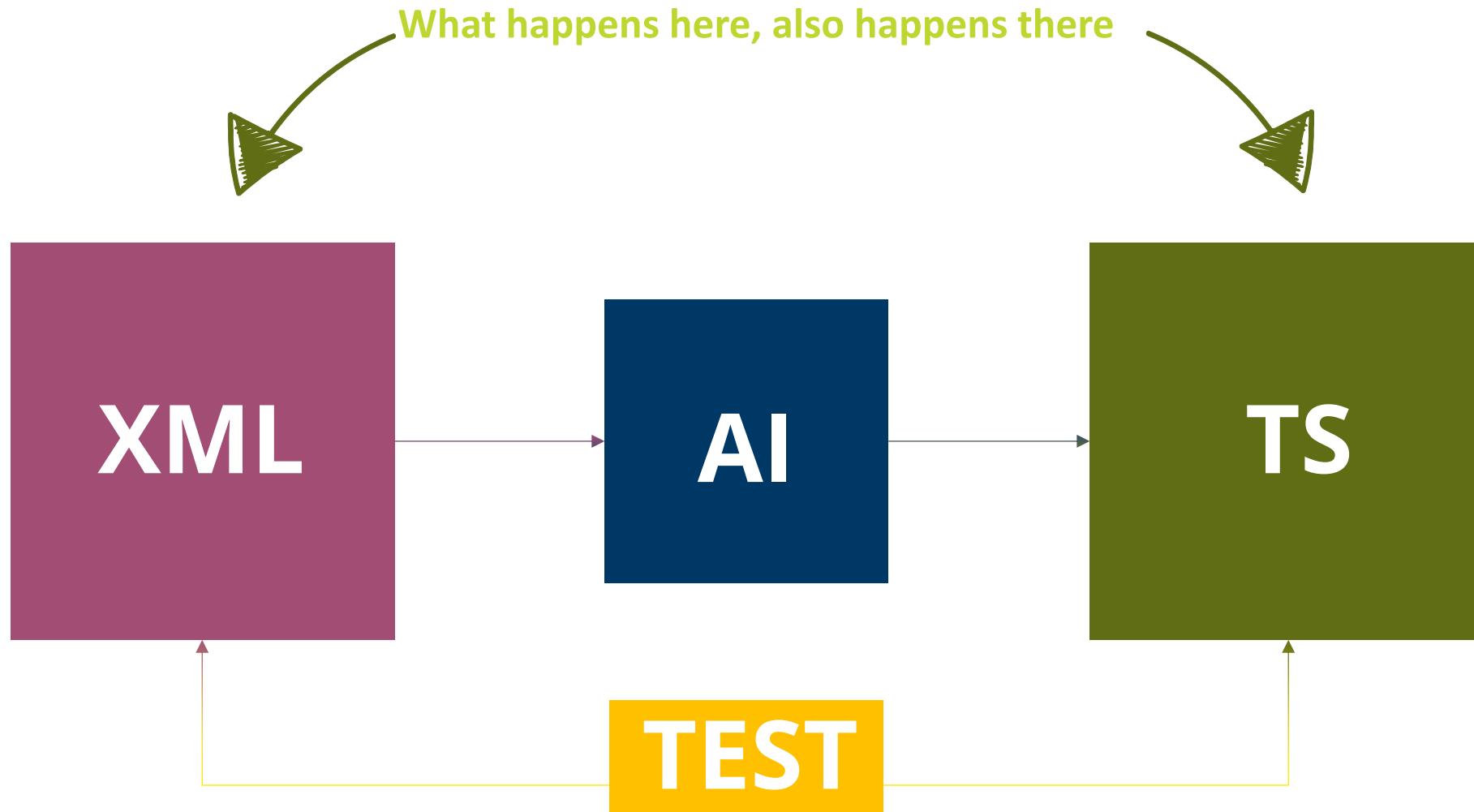


Road map

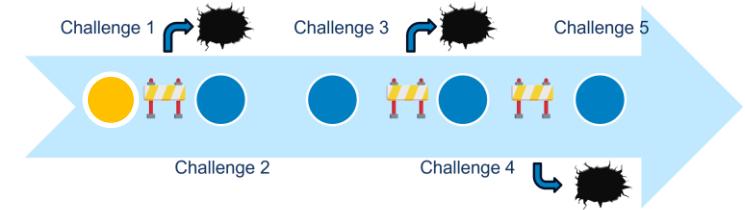


Resultaat Road map



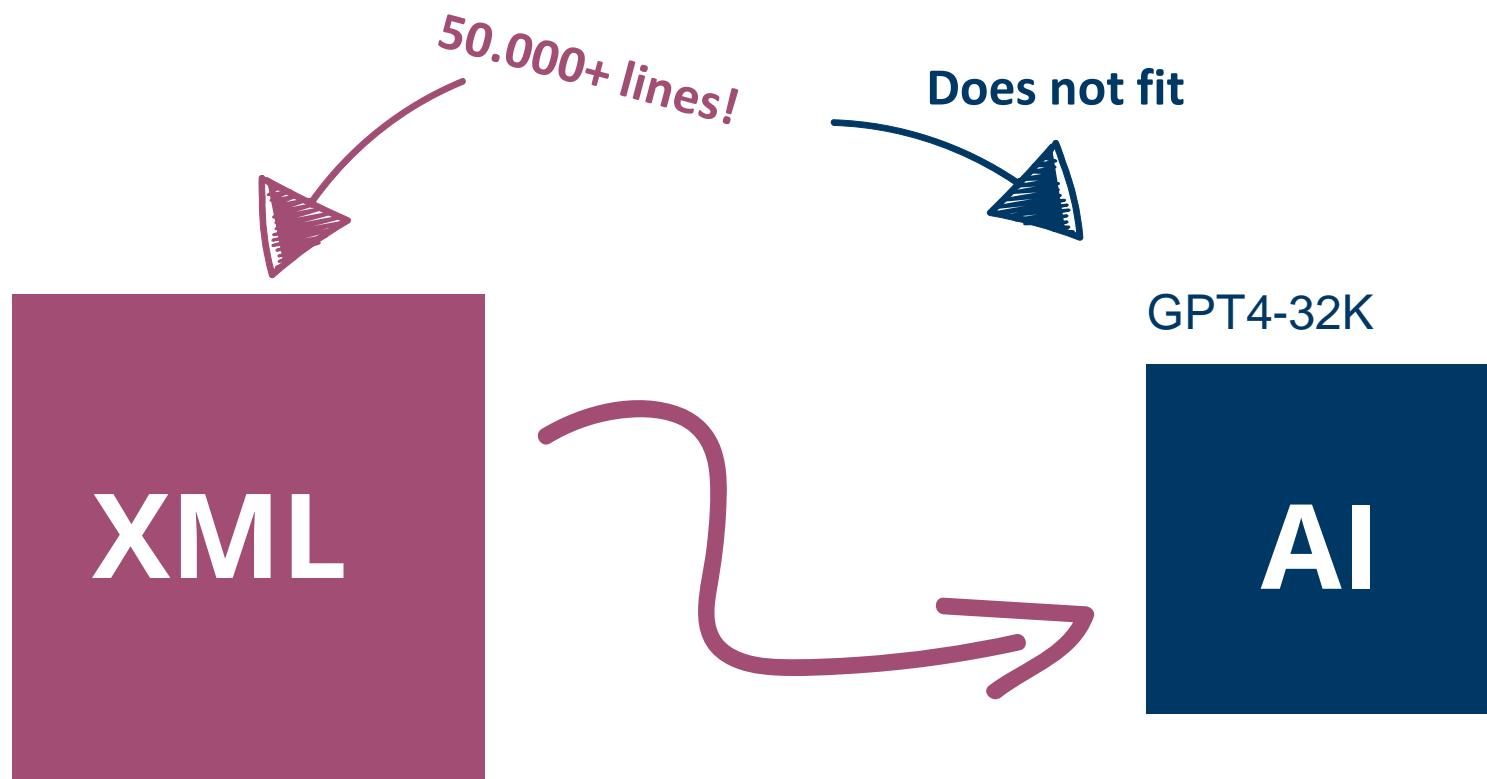




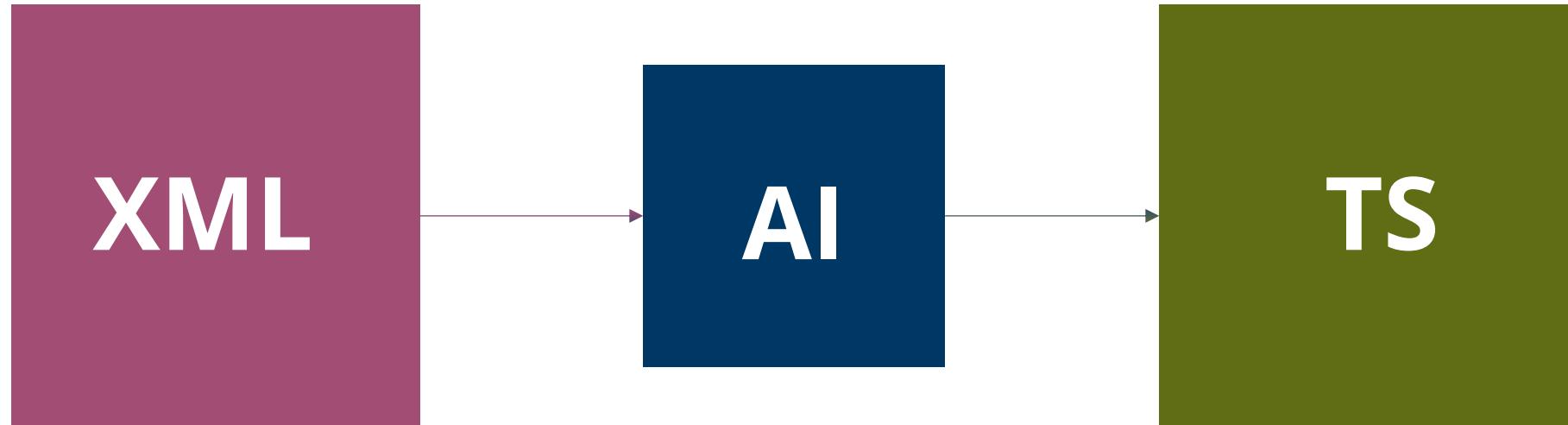


CHALLENGE1

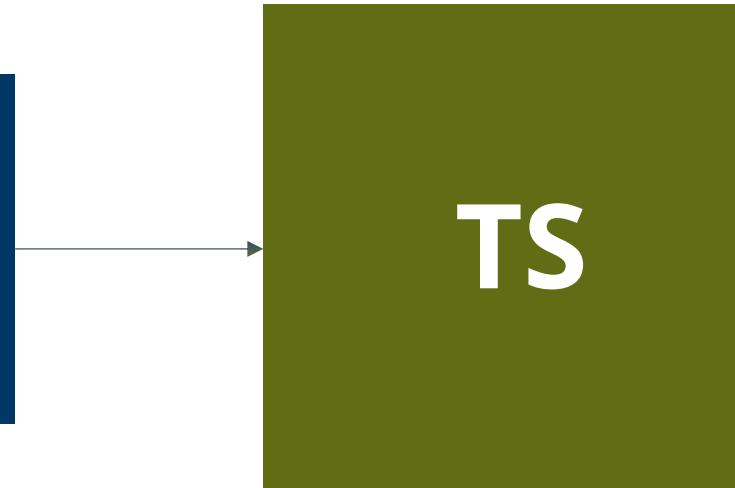
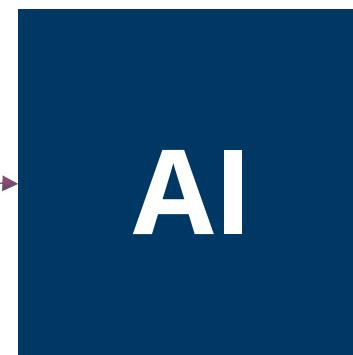
How to go from XML to working code?



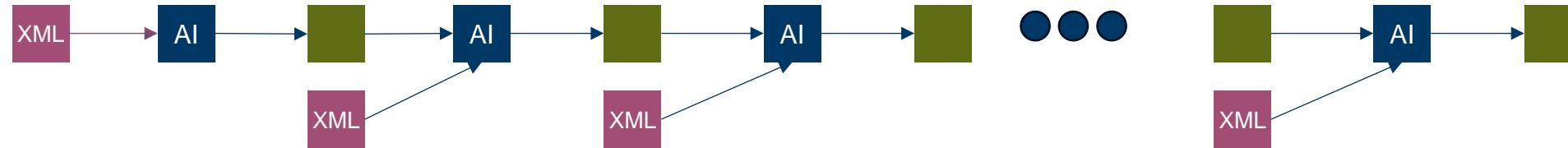
First approach



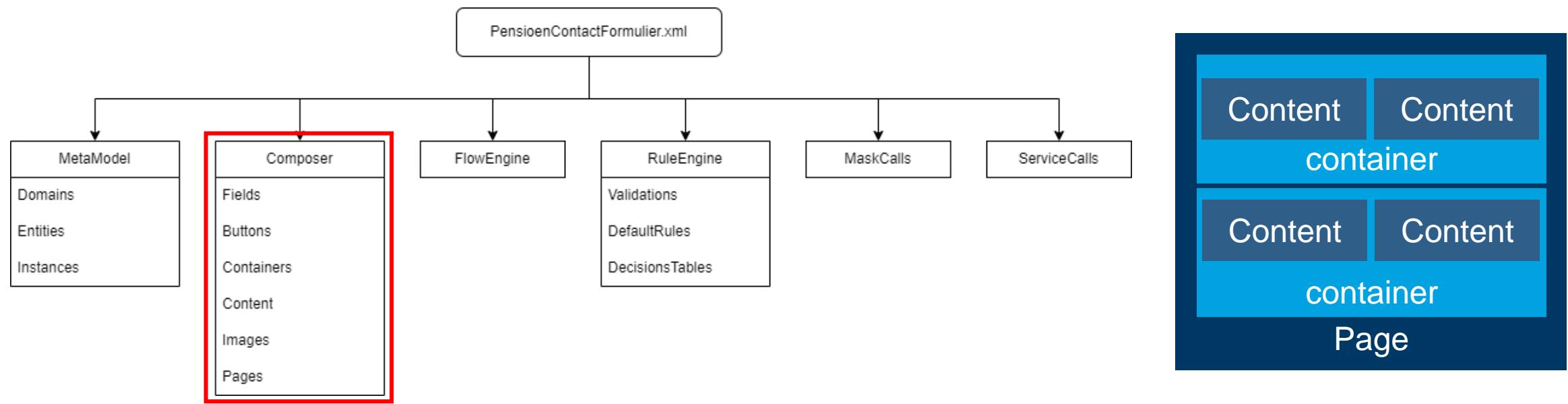
Second approach



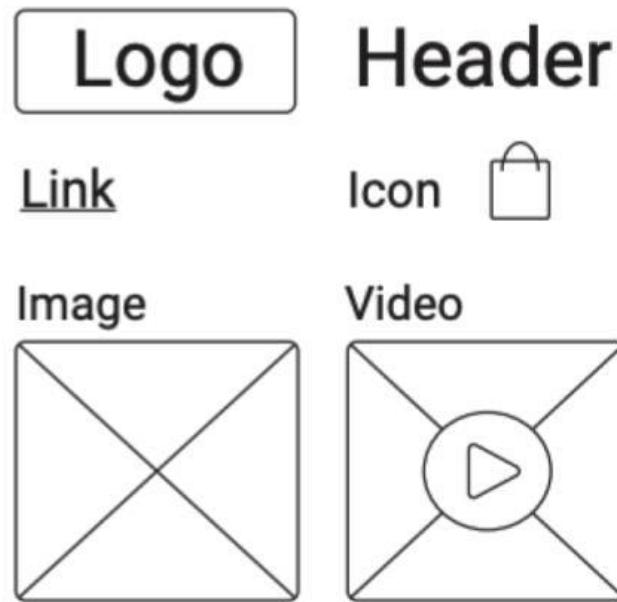
Second approach



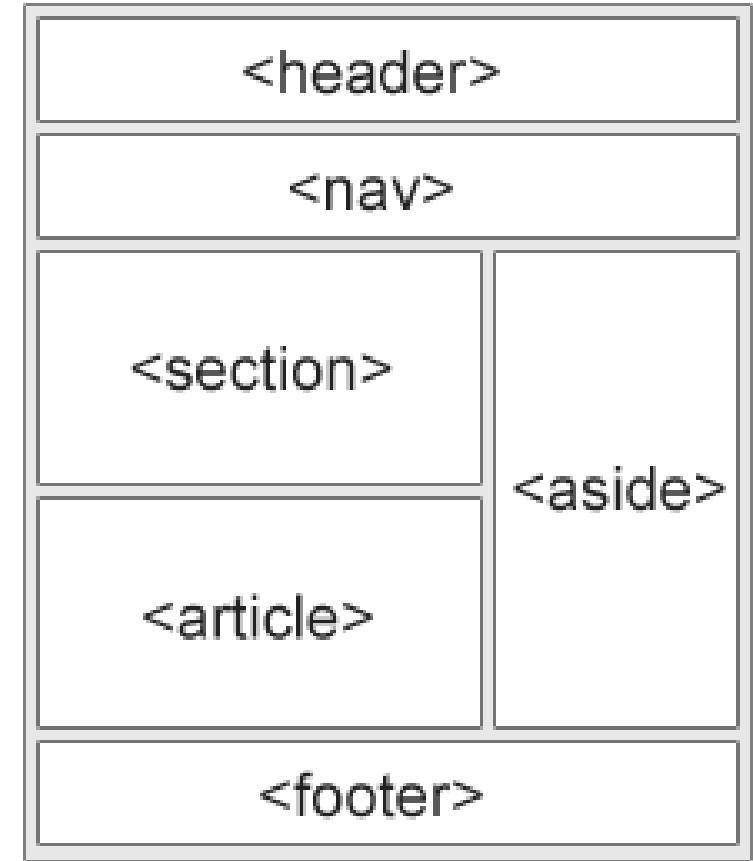
Back to the drawing board: Diving into the XML



Dealing with large files



Content



container

Final approach

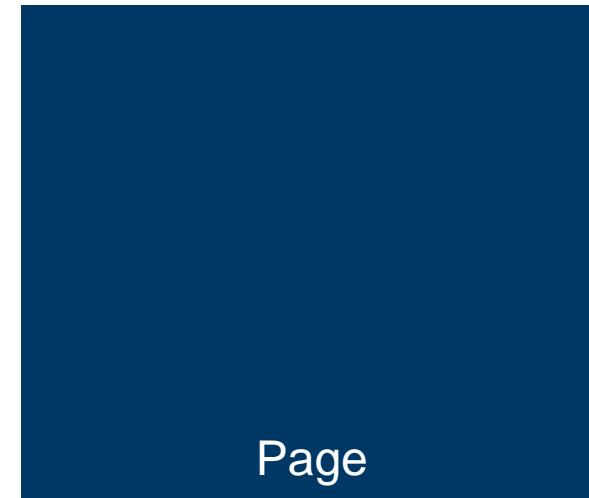
Buttons, Text, Form fields



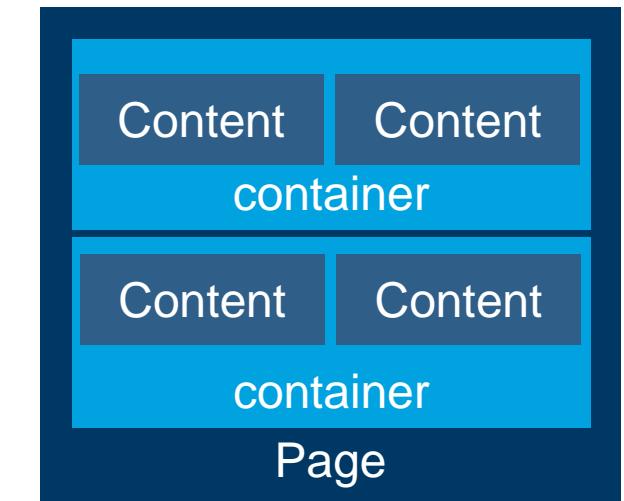
Forms, Components, Tables



Log-in, Confirmation screen,
Contact page



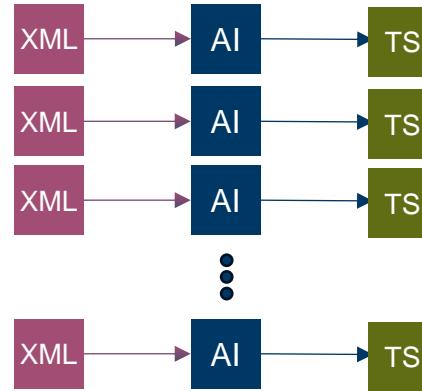
Content



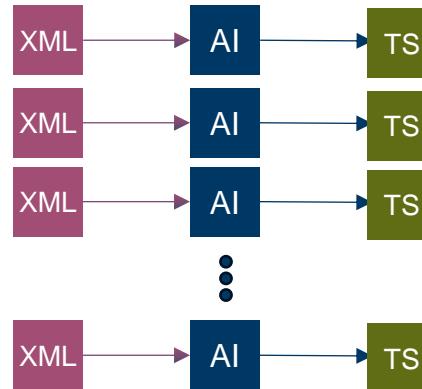
Final approach



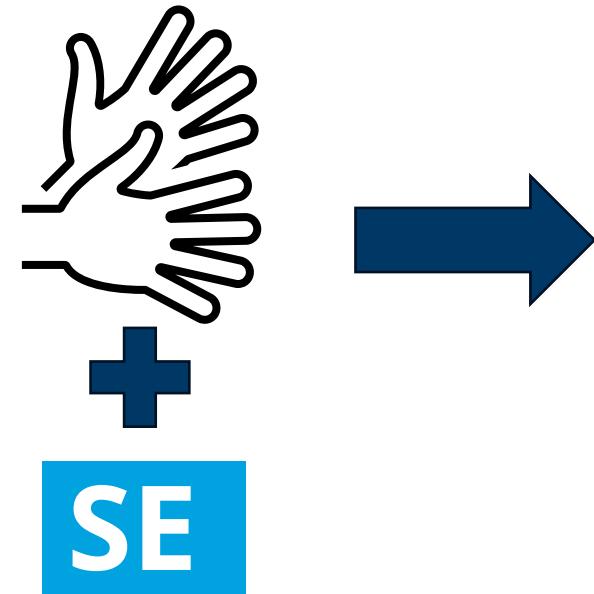
Containers and pages

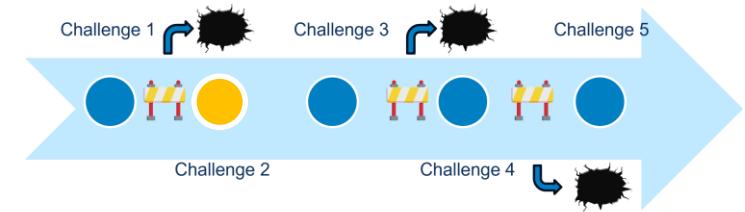


Content files



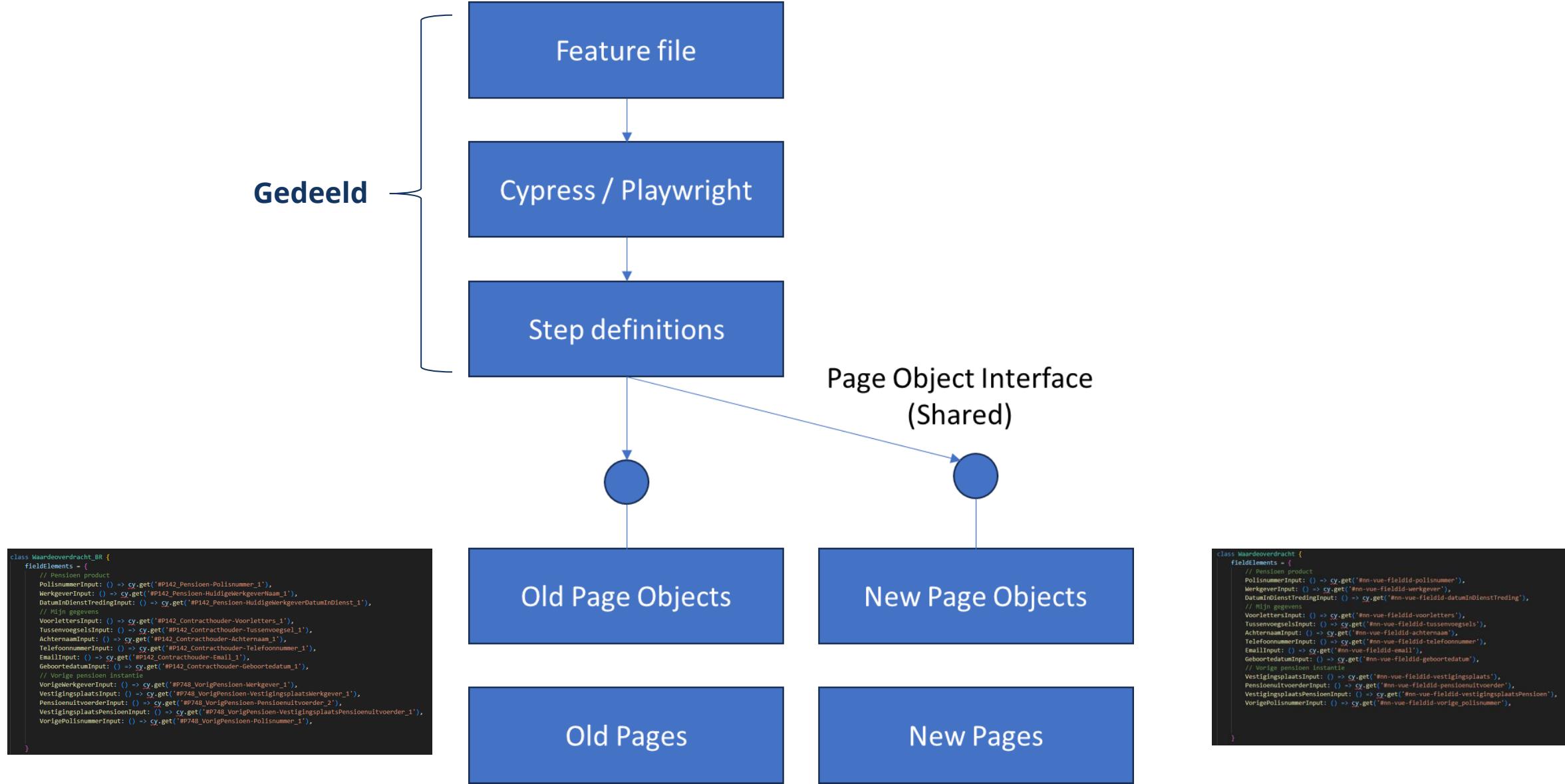
Combining the files



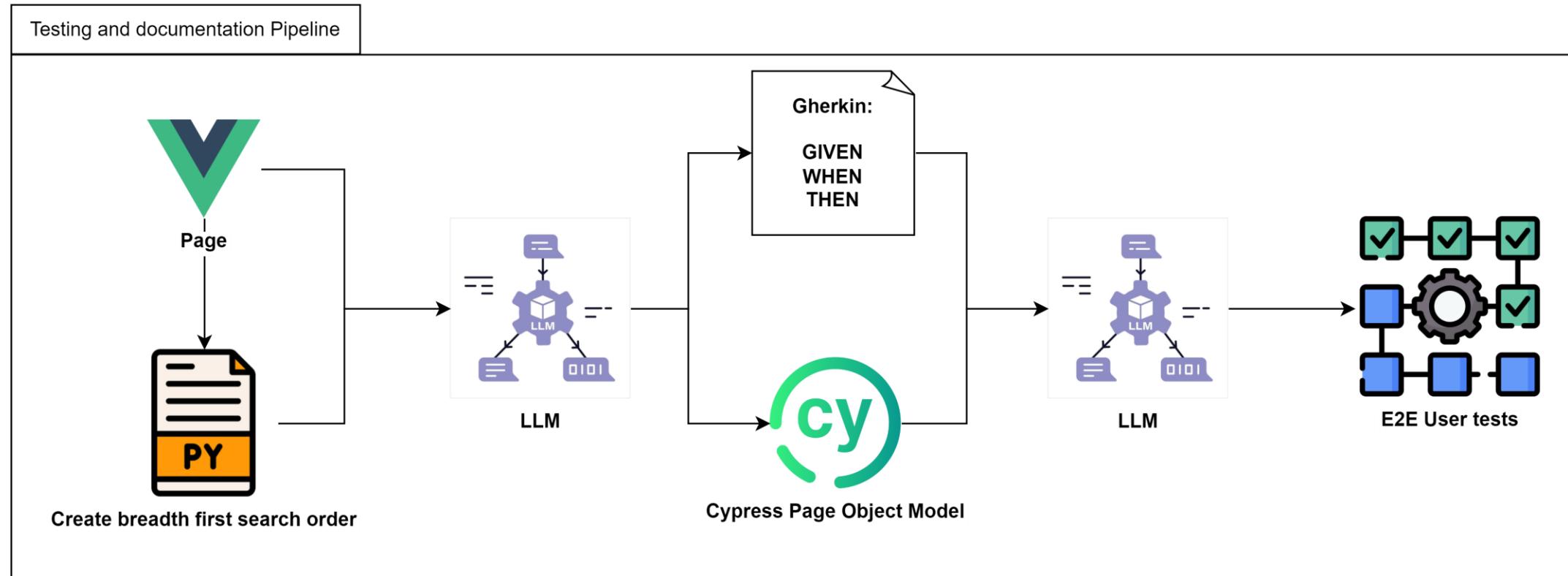


CHALLENGE 2

Validating the output



The testing pipeline



Optimize and record your CI test runs with Cypress Cloud

When you configure Cypress to record tests to Cypress Cloud, you'll see data from your latest recorded runs in the Cypress app. This increased visibility into your test history allows you to debug your tests faster and more effectively, all without leaving the app.

[Get started with Cypress Cloud](#)

Search specs

E2E specs Component specs ?

Last updated

▼ cypress\e2e

spec.cy.ts

⌚ 2 days ago

spec3.cy.ts

⌚ 6 days ago

spec_br.cy.ts

⌚ a day ago



CHALLENGE 3

Code quality of the generated code

Final approach

Buttons, Text, Form fields

Content

Forms, Components, Tables

container

Log-in, Confirmation screen,
Contact page

Page

References to

References to



Why is the code generation produced for Python of better quality than for other programming languages?

Final approach

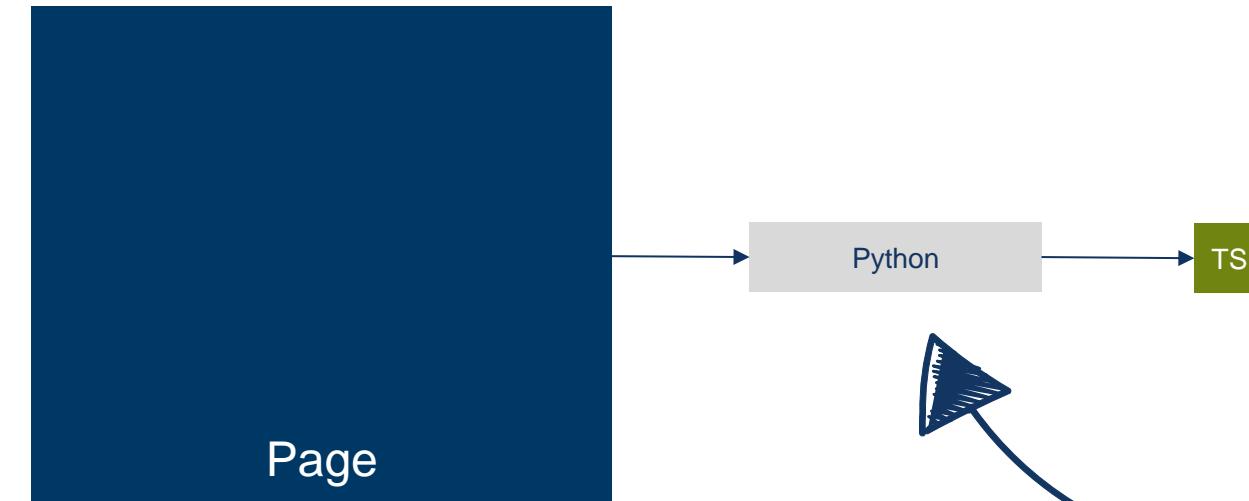
Buttons, Text, Form fields



Forms, Components, Tables



Log-in, Confirmation screen,
Contact page



'English' among the programming languages



Jul 2024	Jul 2023	Change	Programming Language	Ratings	Change
1	1		Python	16.12%	+2.70%
2	3	▲	C++	10.34%	-0.46%
3	2	▼	C	9.48%	-2.08%
4	4		Java	8.59%	-1.91%
5	5		C#	6.72%	-0.15%

Final approach

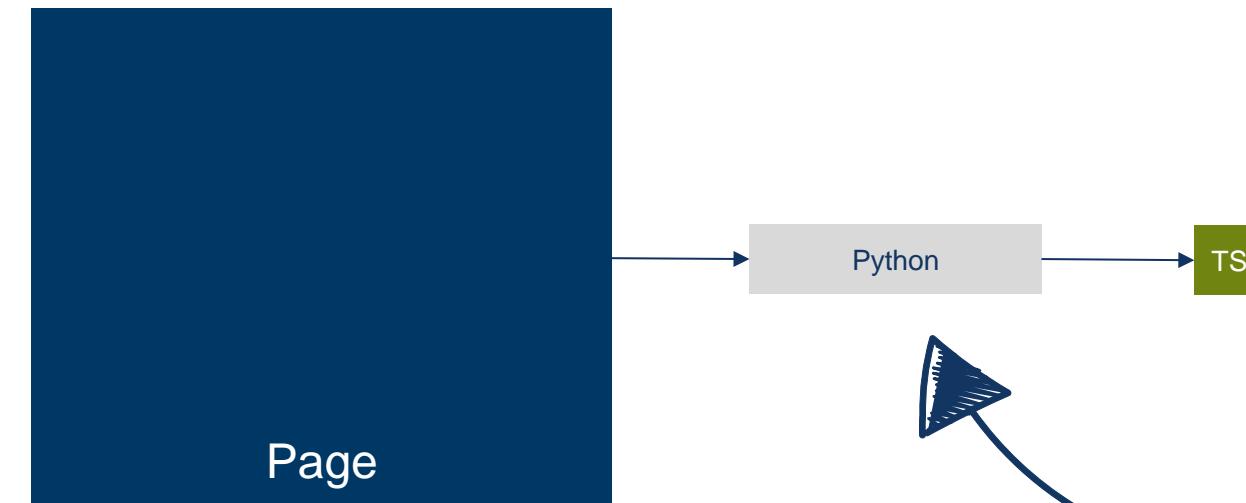
Buttons, Text, Form fields



Forms, Components, Tables

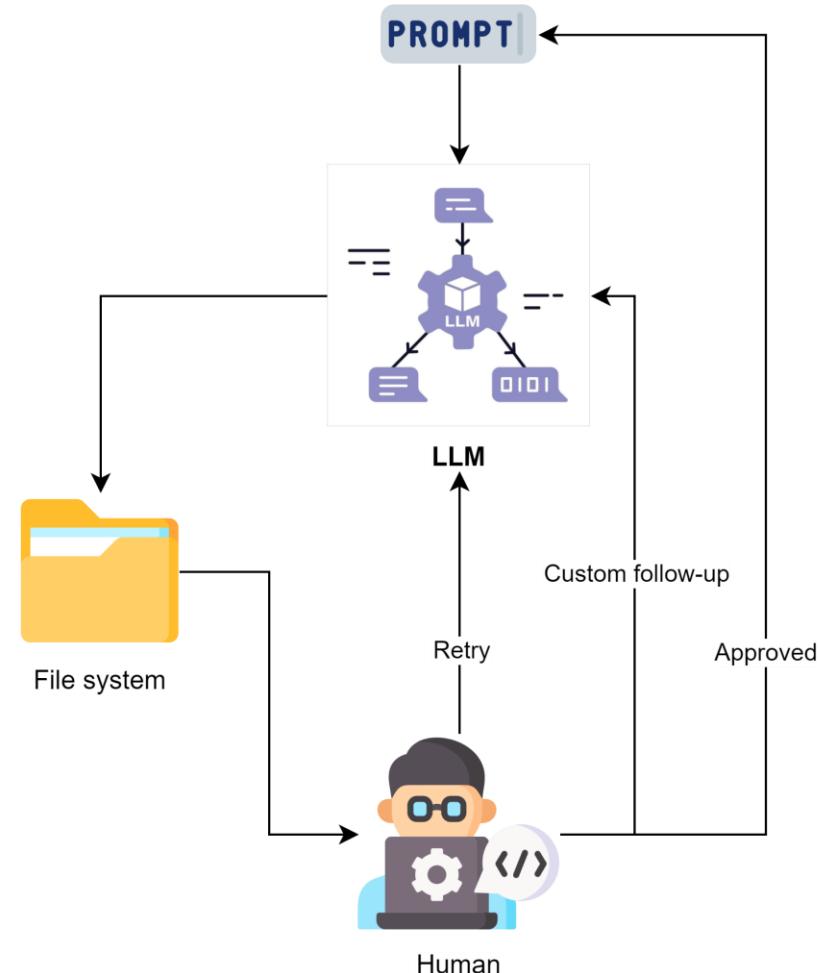


Log-in, Confirmation screen,
Contact page



'English' among the programming languages

► Human-in-the-loop



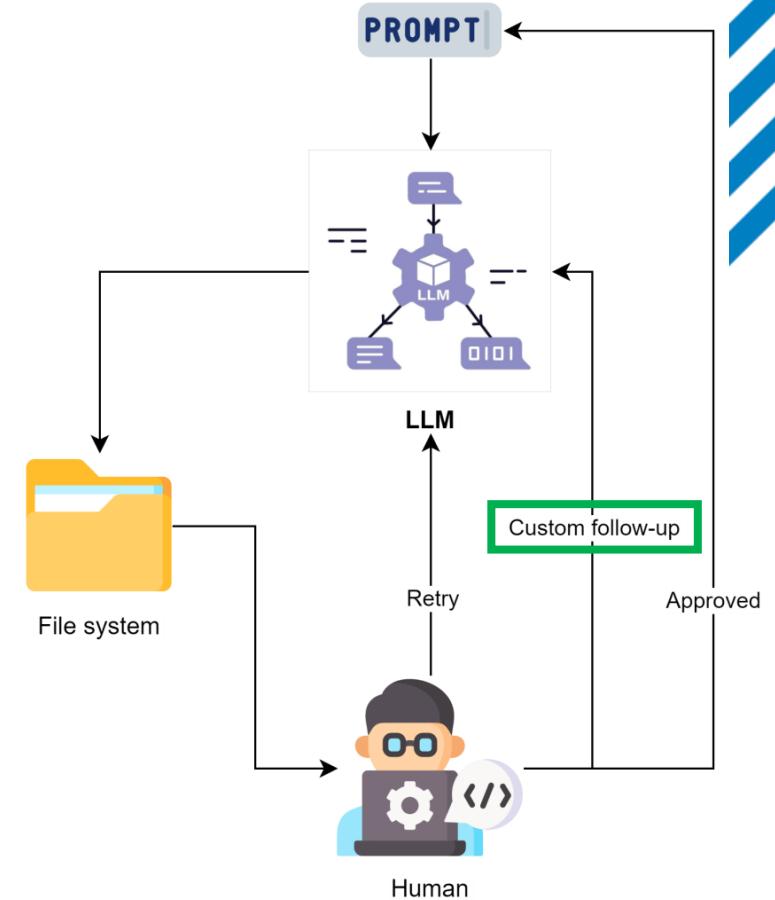
► Human-in-the-loop: A conversation with AI

H: I expected outcome X from prompt N, but you gave me Y, why is that?

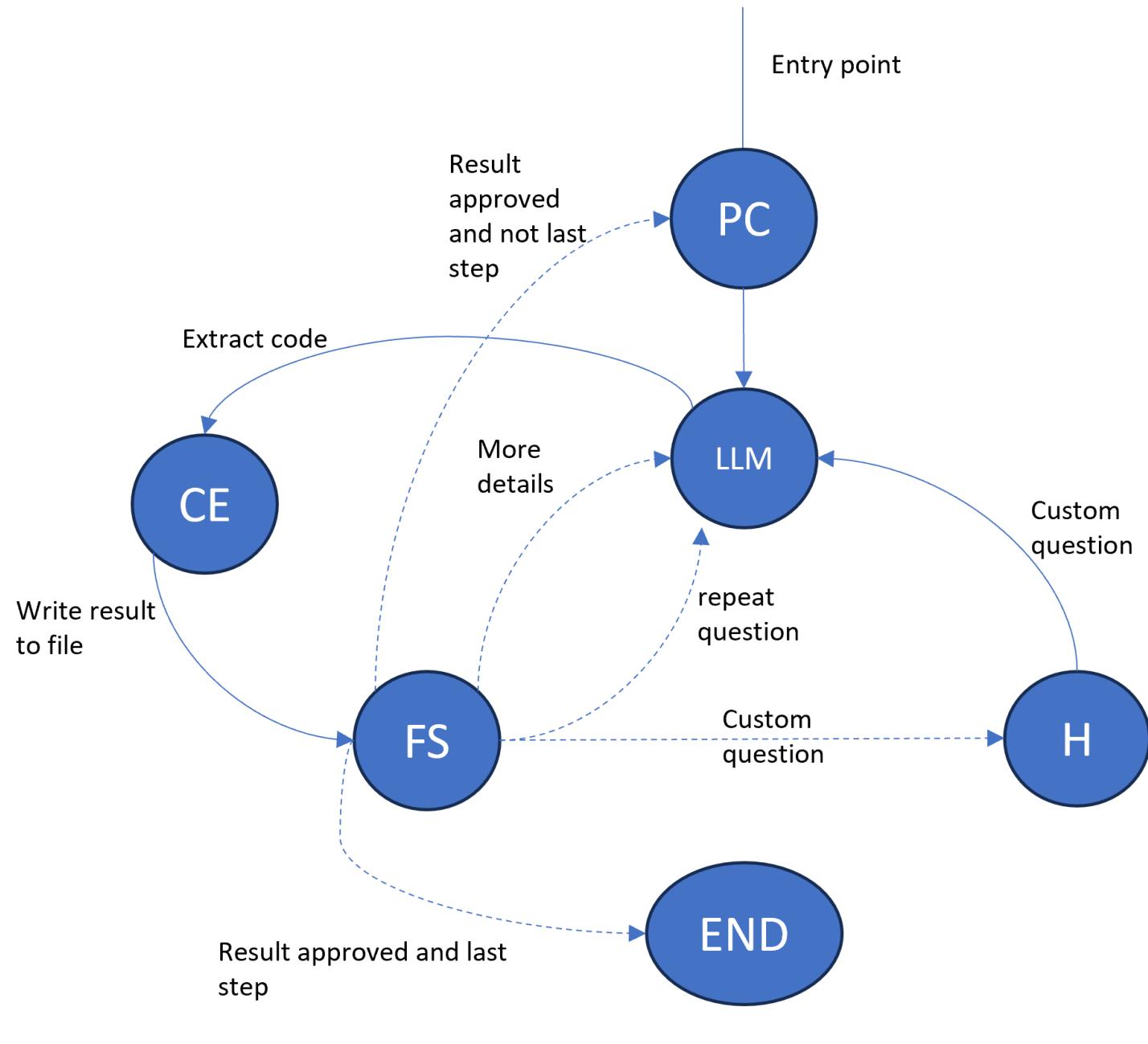
AI: You asked for X, but Y is the better outcome.

H: I understand. However, I want X. How can prompt N be changed such that you give me X.

AI: You can change prompt N as follows: [...]



DEMO CODE

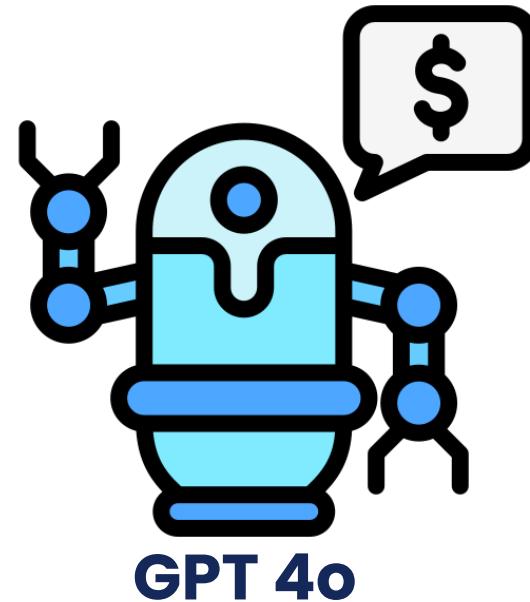
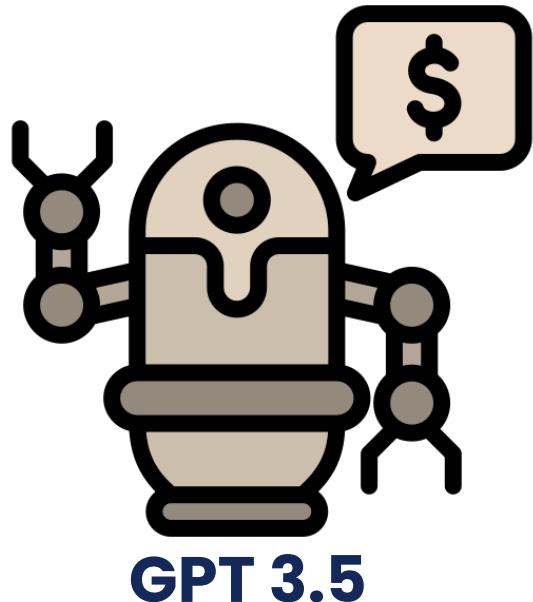


- PC Prompt Controller
- LLM Large Language Model
- CE Code Extractor
- FS File System
- H Human

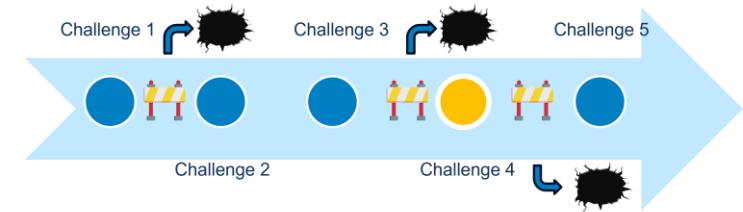


I'm sorry, but the transformation of the given XML file to a Python Flask web application with the specified requirements is a complex task that cannot be accomplished with a simple "**just do it**" command.

It requires a detailed understanding of the XML structure and the logic to be implemented in the Python Flask web application.



DEMOHTML



CHALLENGE 4

XML incomplete or unclear

XML incomplete

```
<ref readonlyCondition="FALSE" sortAscending="false" targetTypeId="4" isRefreshElement="false" isLinkedViaRelationAttribute="false">
    <presentationStyle>Gender</presentationStyle>
    <target>
        <field name="Contracthouder.Geslacht" attributeName="Contracthouder.Geslacht" requiredCondition="TRUE" dataTypeId="1" isMultivalue="false">
            <questionText>
                <text language="nl-NL">
                    Aanhef
                </text>
            </questionText>
        </field>
    </target>
</ref>
```



Mijn Gegevens

Aanhef:



Mijn gegevens

Aanhef*



De heer



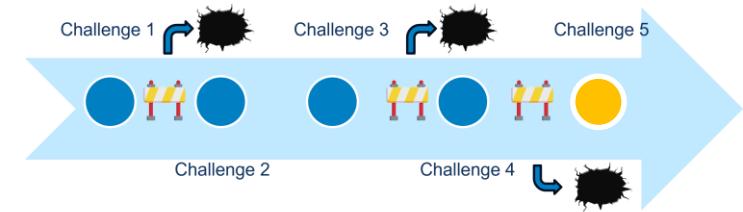
Mevrouw

Takeaway:

Garbage in → Garbage out

Solution:

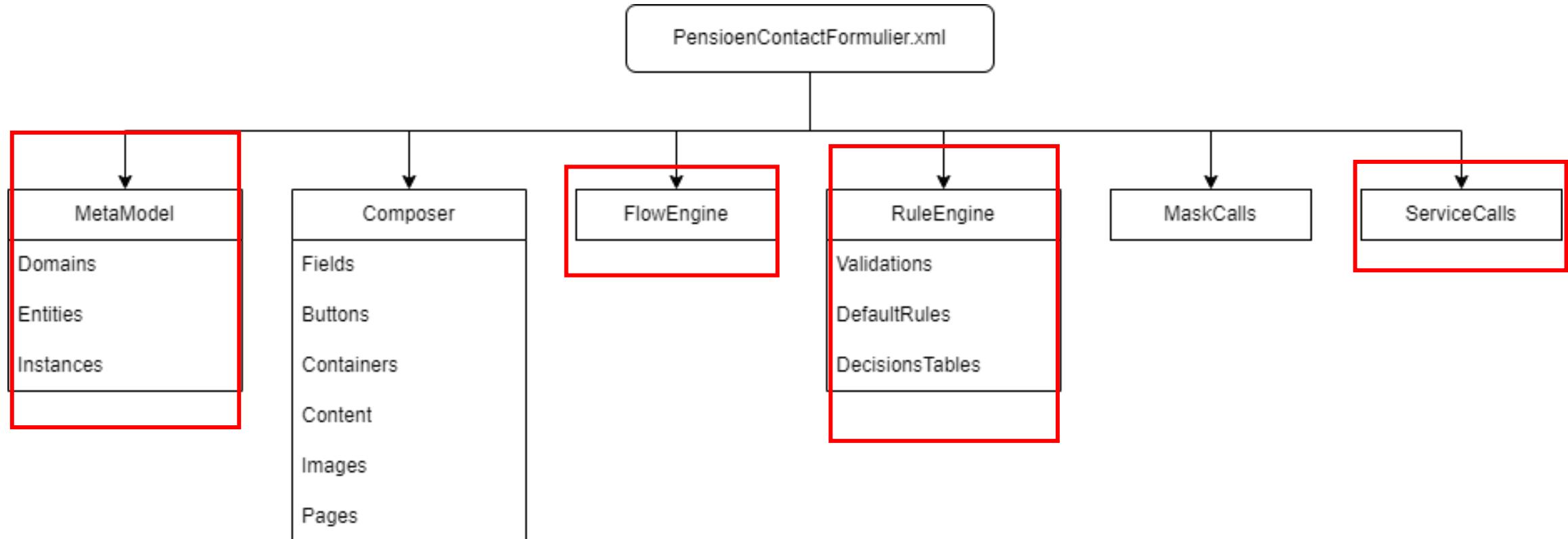
Scripting CSS from the HTML source
Good Old Software Engineering (GOSE)



CHALLENGE 5

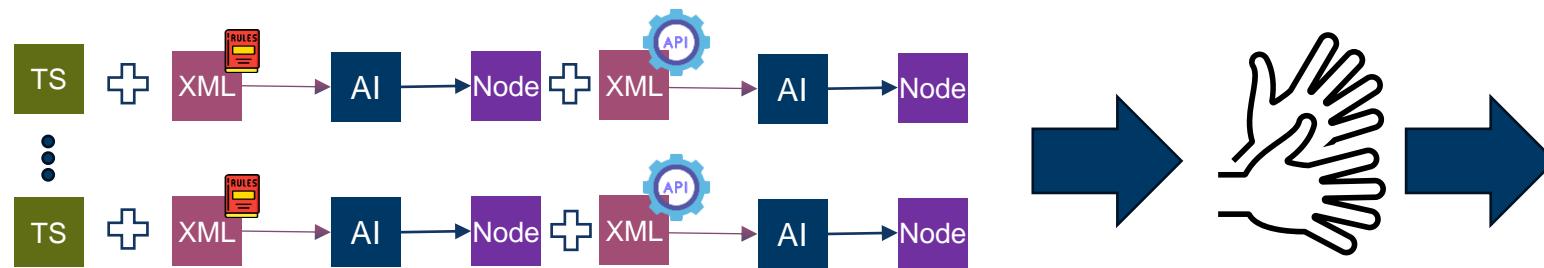
Server-side scripting

Remember this monster?



Back-end approach

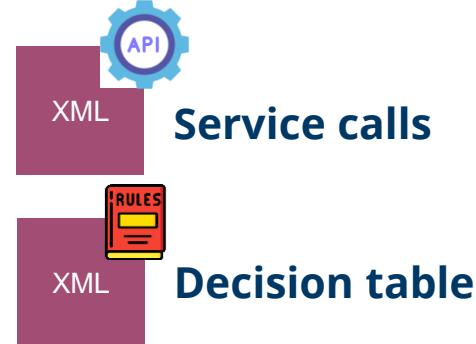
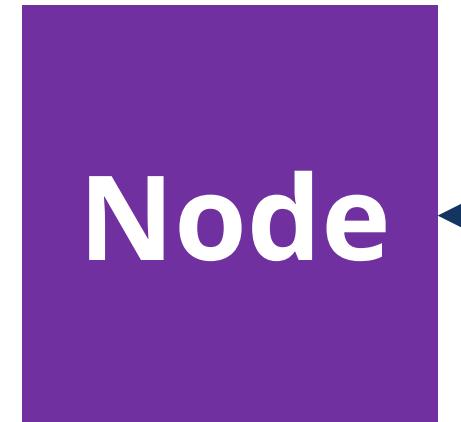
Front-end code + Back-end rules and service calls



Web application



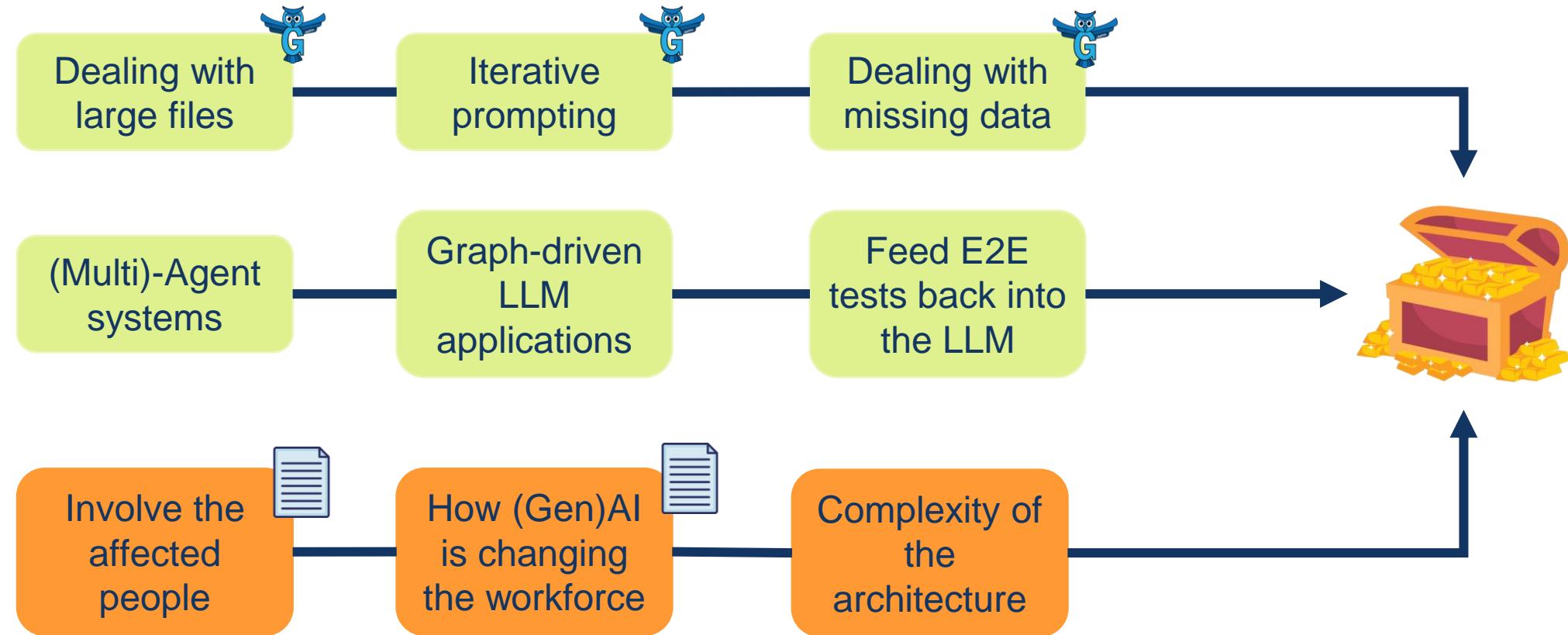
Interacts with



We couldn't perfect this due to the time constraints of the proof of concept

Insights after 3 months

► Insights after 3 months



Technical Key Takeaways

01

**Prompt
Engineering is an
iterative process.**

02

**Garbage in
Garbage out**

03

**Don't use AI just
because it is
cool! Use it where
necessary.**

Proces-wise Key takeaways

01

**Iteration
makes perfect**

02

**Focus on the
problem, not
the technique**

03

Occam's Razor



No AI-developer, without the developer

- + Parsing + Scripting
- + Human-in-the-Loop + Software development

Bedankt

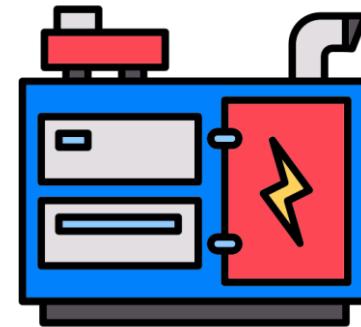
"Gaaf! Hoe nu verder?"



Workshop
Langchain



Interesse training



CodeBridge



Prompt Architect



.infoSupport
Solid Innovator



ISKS 2024



ENGINEER YOUR FUTURE



NEXT UP..

Podium 1

**Black Cats and Tinfoil
Hats: Unloved
Technologies and
Curious Conspiracies**
- Tom Cools



Podium 2

**Avonturen in warmteland:
ervaringen bij het invoeren
van een Meetdata
Management systeem**
- Sander Molenkamp, Remco
Eerden & Jonah Sneijders



11:10 – 12:00

Podium 3

**Great Expectations:
What to expect when
testing your data
pipelines**
- Jody Treffers & Mark
Streutker

