

WYKORZYSTANIE NARZĘDZI AI W ROLI DEVOPSA

OD AUTOCOMPLETE DO AI-NATIVE ENGINEERING



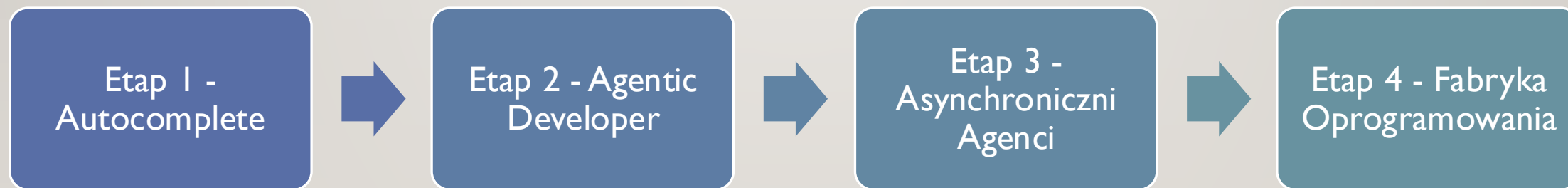
MICHAŁ SMYK, TECHNICAL DEVOPS LEAD



- Od ponad 12 lat pracuję w obszarze rozwoju oprogramowania, chmury oraz automatyzacji procesów IT.
- Prowadzę zespół specjalistów DevOps i Cloud Engineering pracujących w środowiskach Azure i Google Cloud Platform oraz wspieram organizacje w budowaniu efektywnych platform technologicznych.
- Interesują mnie praktyczne sposoby wykorzystywania nowych narzędzi — od automatyzacji po AI — do rozwiązywania realnych problemów technicznych i biznesowych.
- Kontakt: [LinkedIn](https://www.linkedin.com/in/michal-smyk/) (https://www.linkedin.com/in/michal-smyk/)

DOJRZAŁOŚĆ ORGANIZACJI WE WDRAŻANIU AI

- Oparte na bootcampie Cursora dla “Forward Deployed Engineer”
- Istotne nie jest, na jakim poziomie biegłości jesteś Ty jako pojedyncza jednostka w zespole/dziale/firmie – ale jak dojrzała jest cała struktura
- Na jakim etapie jest aktualnie Twój zespół? 😊



CO CHCĘ WAM DZISIAJ PRZEKAZAĆ

Gdzie aktualnie jesteście?	Następny krok
Etap 1	Budowanie podstawowych struktur dla agentów (scaffolding)
Etap 2	Praca asynchroniczna
Etap 3	‘Fabryka oprogramowania’

Nie możesz omijać etapów – korzystanie z Agentów bez podstaw nie da tak dobrych rezultatów, na jakie możesz liczyć robiąc wszystkie te rzeczy.

SCAFFOLDING – DAJ KONTEKST SWOIM AGENTOM

- Pracując z Visual Studio Code/Cursorem, zamiast otwierać pojedyncze repozytorium, możesz tworzyć własne „workspace” pod dany temat
- Zalecany przeze mnie schemat to posiadanie pojedynczego workspace na każdy projekt przy którym pracujesz.
 - Takie ustawienie pozwala używać wielokrotnie niektórych materiałów pomiędzy różnymi projektami (np. shared config repo o którym będzie mowa później)
 - Jeżeli dobrze opiszesz co masz dostępne agentom nawet poza workspace’em, zysk w porównaniu do ilości przepalonych tokenów będzie wyższy 😊

TRAKTUJ AI JAKO NOWEGO CZŁONKA ZESPOŁU

- Traktuj swój komputer jak repozytorium wiedzy, gdzie poszczególne obszary połączone są ze sobą odniesieniami (np. wzmianka w plikach AGENTS.md gdzie szukać konkretnych informacji).
- Sprowadzaj wiedzę do jak najprostszych plików, najlepiej tekstowych – dzięki temu AI szybciej przetworzy twoje dane i będzie w stanie zmaksymalizować ich użyteczność.



WORKFLOW OPARTY NA PLAN MODE

Intencja

- Opisuję co agent ma zrobić
- Jako output oczekuję pliku .md gdzie agent opisze, co chce zrobić

Agent

- Tworzy plik .md

Sprawdzenie

- Weryfikuję zawartość pliku .md
- Nanoszę poprawki samodzielnie lub każę agentowi poprawić na bazie moich uwag

Akcja

- Każę agentowi zaimplementować zmiany wg. instrukcji z pliku .md

PIERWSZE KROKI WE WDRAŻANIU AGENTA

- Plik startowy dla agenta – AGENTS.md – zwykle to tutaj zacznij myśleć nad tym, co jest do zrobienia
- Im lepiej opiszesz co posiadasz w projekcie, tym bardziej będzie pomocny.
- Warto też opisać, jaki masz styl pracy, rolę w projekcie, kryteria skończenia zadania (definition of done)
- Najważniejsze – nie pisz tego samodzielnie, opowiedz co tam ma się znajdować Agentowi i **niech on napisze to za Ciebie.**

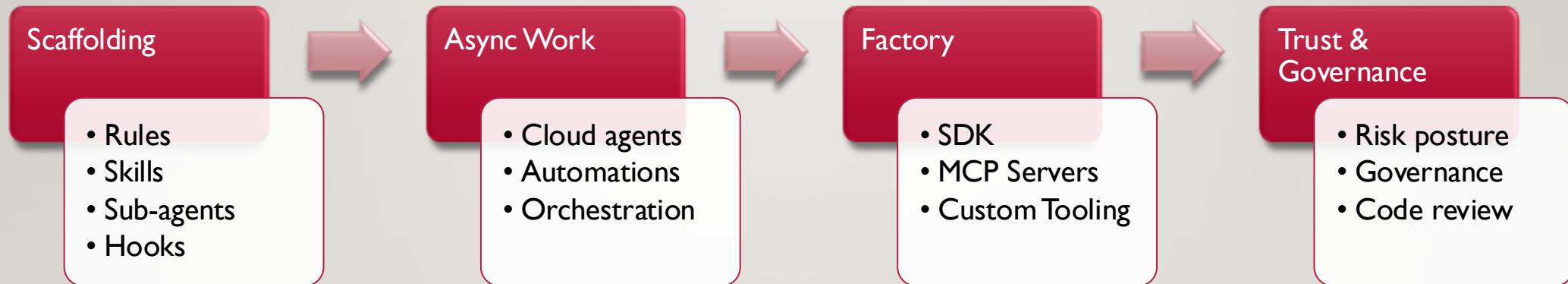
WIEDZA, KTÓRĄ WYKORZYSTASZ WIELOKROTNI

- Przygotuj proste pliki tekstowe, opisujące jak agent ma się zachowywać w określonych sytuacjach
 - Instrukcje – zasady, których agent powinien przestrzegać podczas pracy
 - Umiejętności – krótki opis jak robić rzeczy wg. standardów projektu, zwykle do ponownego użycia
 - Prompty – konkretne zapytanie, które chcemy zlecić agentowi do wykonania. Może np. składać się z wykorzystania kilku umiejętności po sobie.

IT TO SPORT DRUŻYNOWY – RÓWNIEŻ Z AI

- Zamiast trzymać swoje prompty, opisy jak wszystko działa lokalnie – stwórzcie w ramach swojego zespołu repozytorium, gdzie wszyscy będą mogli wrzucać najbardziej przydatne (lub wymagane!) instrukcje
- Pomaga to zarówno utrzymanie spójności na poziomie zespołu, jak i propagowanie wiedzy między poszczególnymi osobami
- W ramach VS Code Workspace po prostu dodaj to repo i wpis w AGENTS.md, żeby z tego korzystał 😊

CO JEST MI POTRZEBNE, BY WEJŚĆ NA WYŻSZY POZIOM W UŻYWANIU AI?



Oparte na materiałach bootcampu Cursora.

AI TO NIE TYLKO CHATBOT – CLI W PRAKTYCE

- [Copilot CLI](#) (lub podobne narzędzia)
- Świetnie nadaje się np. do uruchamiania analizy na zdalnych serwerach
- Osobiście wykorzystuję po prostu w Terminalu, gdy kontekst nad którym pracuję nie opiera się na kodzie – dokumenty, arkusze, praca koncepcyjna.
- Pracując w IDE, opieram się na kolaboracji z narzędziami przy pisaniu kodu – używając CLI, deleguję na nie zadania.

MOJE BŁĘDY Z PRZESZŁOŚCI – JAK OPTYMALIZOWAĆ WYKORZYSTANIE TOKENÓW

- Wybrany model - "Auto" twoim przyjacielem. Nie dać się ponieść hype'owi – to, że każdy kolejny model bije benchmarki, jeszcze nie znaczy, że faktycznie jest mocniejszy – kolejne modele są szkolone pod te benchmarki, żeby firmy na nich zarabiały, nie koniecznie pod lepszy output dla użytkownika.
- Kontekst modeli – sprawdzaj regularnie, na co właściwie przepalasz tokeny – więcej nie znaczy lepiej
- AI uwielbia tworzyć od nowa te same skrypty pomiędzy różnymi uruchomieniami agentów – jeżeli wiesz, że z jakiegoś prompta będziesz korzystać wielokrotnie, poproś agenta by zapisał skrypt na dysku na potrzeby przyszłych uruchomień
- Nie każ agentom za każdym razem skanować tych samych dokumentów – zapisz w instrukcji, by po analizie dodał plik .md opisujący co zawierają przeskanowane pliki i dodał do AGENTS.md informację, że taki plik istnieje



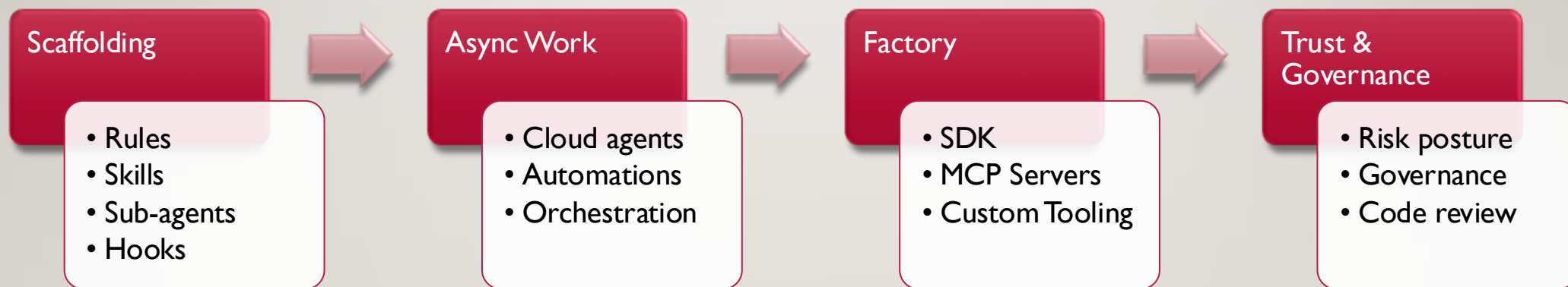
EKSPERYMENTUJ BEZ PRZEPALANIA BUDŻETU FIRMY

- Odpalanie LLMów lokalnie – jaki model najlepiej zadziała na Twojej maszynie?
<https://www.canirun.ai/>
- Wypróbuj pracę z agentami w warunkach domowych - <https://mistral.ai/products/vibe/>
(zawiera również agenta w wersji CLI!)
- Materiały community - agenci, umiejętności, instrukcje, pluginy -
<https://github.com/github/awesome-copilot>

POZWÓL AI FAKTYCZNIE PRACOWAĆ KORZYSTAJĄC Z TWOICH NARZĘDZI

- Serwery MCP pozwalają na integrację Agentów z narzędziami, z których Ty sam musisz korzystać na co dzień
- Bez narzędzi AI zgaduje, jak wszystko działa. Z MCP jest w stanie wchodzić w interakcję z całym środowiskiem.
- Wykorzystywane przeze mnie:
 - Azure DevOps MCP Server ([aktualna](#) oraz [nowa wersja, jeszcze w preview](#))
 - Azure MCP Server [azure/mcp](#)
 - Terraform - [hashicorp/terraform-mcp-server](#) – redukuje halucynacje modeli w kontekście Terraforma + operacje na Terraform Enterprise (jeżeli używacie)
- Pierwsze dwa serwery wystawiają również flagę "—readonly", na wypadek gdyby Wasz pracodawca obawiał się, co LLMy będą wyprawiać na Waszych zasobach 😊
- Znajdź najlepsze serwery dla Twojego zastosowania:
<https://github.com/punkpeye/awesome-mcp-servers>

JAK WEJŚĆ NA NOWY POZIOM WYKORZYSTYWANIA AI WEWNĄTRZ SWOJEGO ZESPOŁU?



MATERIAŁY



Kod źródłowy: <https://github.com/Azkel/AiNowPolska-June2026>

Q&A

DZIĘKUJĘ ZA UWAGĘ! 😊



ZADANIE DOMOWE 😊

- Jeżeli jeszcze nie korzystasz z pliku AGENTS.md – stwórz taki plik
 - Podziel się nim z resztą zespołu, zacznijcie budować wspólne repozytorium wiedzy przy użyciu Gita
- Zacznij pracować z wykorzystaniem "Plan Mode"
- Stwórz jeden proces wykorzystujący agentów, który będzie mógł wykorzystywać cały zespół