Od nápadu k aplikaci

Kdo vybírá další krok?

Vibe coding a agentní programování



Mgr. Markéta Trnečková, Ph.D.

Minule krok. Dnes cíl.

Minule

Zadám úpravu, přečtu návrh a spustím program. Další krok vybírám já.

Dnes

Předám cíl. Systém může upravovat, spouštět a volit další kroky.

Troje dveře. Jedna výhra.



Vyberete dveře. Moderátor otevře jiné a ukáže kozu.

Zůstat, nebo přejít k posledním zavřeným dveřím?

Co ví moderátor?

Auto je rovnoměrně za jedněmi ze tří dveří.

Moderátor ví, kde auto je.

Vždy otevře jiné dveře s kozou a vždy nabídne změnu.

Má-li na výběr dvě kozy, vybírá mezi nimi rovnoměrně.

Svůj odhad a důvod si zapište před použitím AI.

Postavte simulátor

Aplikace má spolužákovi pomoci prozkoumat, zda se vyplatí změnit volbu.

Vlastní prompt. Pravidla hry máte; podobu aplikace navrhněte.

Uložte první verzi. Vyzkoušejte ji jako uživatel.

V této fázi nemusíte číst každý řádek. Ještě nepřebíráte ověřený model.

Stačí prohlížeč

Chat + vlastní běh

Univerzitní chat vytvoří jeden HTML soubor.

Spustíte jej v prohlížeči nebo v připraveném Colab notebooku.

Dostupný agent

Například Replit:
https://replit.com

Pracuje přímo v projektu.

Při omezení účtu sledujte demonstraci vyučující.

Bez instalace a bez nákupu předplatného. Používáme náhled, nic nemusíme publikovat.

Co vás přesvědčilo?

Vypadá to dobře. Ale co víme?

Tlačítka reagují? Výsledky se mění? Vysvětlení zní odborně?

Navrhněte bez AI dvě vlastní kontroly.

Nejdříve očekávání a důvod. Potom pokus.

Pokud problém nenajdete, popište přesně, co jste ověřili.

Co zde nazýváme vibe codingem?

Iterujeme podle zamýšleného chování a viditelného výsledku.

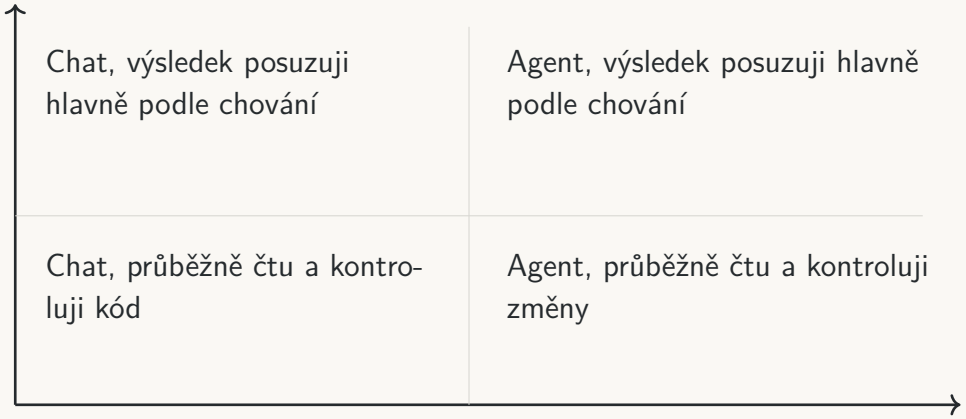
Podrobnosti implementace nemusíme průběžně číst.

Prototyp tak může vzniknout rychle.

Při převzetí musíme doplnit kontrolu toho, co jsme přeskočili.

Pracovní definice pro kurz; ne každé programování s AI je vibe coding.

Způsob práce a autonomie



Doprava: více kroků volí AI. Nahoru: méně průběžné kontroly implementace.

Co musí jedna hra zachovat?

Moderátor neotevře hráčovy dveře ani dveře s autem.

Změna vede k posledním zavřeným dveřím.

Auto se během jedné hry nepřesouvá.

Výsledek odpovídá stavu hry, ne předem napsanému procentu.

Dokážeme jednu hru ověřit bez AI a bez náhody?

Tři polohy auta stačí

Hráč vždy nejprve volí dveře 1. Každá poloha auta má stejnou váhu.

AUTO	ZŮSTAT	ZMĚNIT
Dveře 1	výhra	prohra
Dveře 2	prohra	výhra
Dveře 3	prohra	výhra

Zůstat: $1/3$**. Změnit:** $2/3$**. Platí pro naše pravidla.**

Více běhů neopraví jinou hru

Výběrová nejistota

Konečný počet her kolísá. Více běhů zpřesní odhad pravděpodobnosti v implementovaném modelu.

Chyba modelu

Moderátor může v kódu jednat jinak, než jsme zadali. Milion běhů tuto chybu neodstraní.

Shoda s očekávaným procentem sama o sobě neověřuje mechanismus.

Kdy už jde o agenta?

Systém dostane cíl a přístup k nástrojům.

Volí další akce podle průběžných výsledků.

Může číst soubor, změnit kód, spustit kontrolu a reagovat.

Samotná dlouhá odpověď ani vygenerovaný plán nestačí.

Vždy sledujeme skutečné akce dostupné v konkrétním prostředí.

Model není celý agent

Cíl + kontext + dosavadní výsledky nástrojů

Model → akce → nástroj

↑ nový kontext ← výsledek ←

Aplikace vykonává volání nástrojů a vynucuje oprávnění.

Zpětná vazba mění kontext dalšího kroku; není to automatické přetrénování vah.

Jedna změna, jasné převzetí

Rozšiřte simulátor o dávku her. Pro každou vygenerovanou hru vyhodnoťte zůstat i změnit a ukažte počty a podíly výher.

Pokud dávky už máte: přidejte audit jedné hry s pevně zadaným autem a první volbou.

Nejdříve sami napište, podle čeho změnu převezmete.

Uložte verzi před změnou. Omezte úkol na jednu funkční úpravu.

Dohoda s agentem

Co má vzniknout?

Cíl a ověřitelný výsledek.

Dostupné soubory a pravidla hry.

Doklad kontroly.

Co smí udělat?

Povolené změny a spuštění.

Kdy se má zeptat.

Kdy má skončit.

Vy navrhujete hranice; agent vybírá kroky uvnitř nich.

Sledujte stopu práce

Jaký další krok systém zvolil?

Který soubor skutečně změnil?

Jakou kontrolu spustil a co vrátila?

Reagoval na výsledek, nebo jen napsal „hotovo“?

Zapisujte viditelné akce a výstupy. Nepotřebujeme domněnky o skrytém uvažování.

Kdo kontroluje kontrolora?

Stejný předpoklad

AI může vytvořit implementaci i test, které spolu souhlasí, ale oba popisují jinou hru.

Nezávislá opora

Ruční případy, pravidla moderátora a log jedné hry.

Teprve potom náhodné dávky.

Zelený test říká, co prošlo. Musíme vědět, co test skutečně ověřuje.

Přijmout, vrátit, nebo doplnit?

Porovnejte původní a novou verzi.

Zopakujte vlastní kontrolu a jednu společnou.

Oddělte doložené výsledky od tvrzení asistenta.

Vyslovte rozhodnutí a konkrétní důvod.

Při nejasnosti se vraťte k uložené verzi a zúžte zadání.

Co když moderátor neví?

Náhodně otevře jedny ze dvou nevybraných dveří.

Někdy odhalí auto. Tyto hry zvlášť zaznamenáme.

Strategie porovnáme pouze mezi hrami, kde odhalil kozu.

Před použitím AI: zůstane výhoda změny stejná?

Nejde o opravu původní hry. Měníme pravidla a podmínění.

Jiná informace, jiný výsledek

Při náhodném otevření a odhalené koze:

$$P(\text{výhra zůstat} \mid \text{koza}) = 1/2$$

$$P(\text{výhra změnit} \mid \text{koza}) = 1/2$$

Jmenovatel je počet her s odhalenou kozou, ne všechny zahájené hry.

Předpoklady: rovnoměrné umístění auta a rovnoměrné otevření nevybraných dveří.

Co bych AI svěřil příště?

Snadno ověřitelné

Změna popisku, vzhledu nebo malého izolovaného výpočtu.

Uložená verze a rychlá kontrola.

S vyššími důsledky

Změna modelu, práce s reálnými daty, publikování výsledku.

Více kontroly a rozhodnutí člověka.

Rozsah autonomie závisí na ověřitelnosti, důsledcích a možnosti návratu.

Co jste dnes skutečně delegovali?

Jedna věc, kterou AI urychlila.

Jedno rozhodnutí, které jste si ponechali.

Jeden důkaz, díky němuž jste změnu převzali.

Kdy bych příště předal méně autonomie?

Důležitý úkol: pošlete vyučující screenshot vytvořeného webu.

Odevzdejte verze nebo rozdíl, vlastní kontroly a zdůvodnění převzetí.

K dalšímu čtení

Yao et al.: ReAct (2022/2023). Propojení plánování, akcí a pozorování.

Anthropic: Building effective agents (2024). Rozdíl předem daného workflow a agenta.

P. B. Stark: Monty Hall, SticiGui. Pravidla a podmínění.

Replit Starter: omezený Lite build; Full build a Plan Mode nejsou bezplatnou podmínkou semináře.

Prostředí ověřeno v dokumentaci 12. 10. 2026; přístup konkrétních účtů je třeba vyzkoušet.