

LLM開発環境の構築

本日のアジェンダ

1. オープニング、自己紹介
2. 本日のゴール
3. AI VibeCodingの環境全体像
4. おすすめツール等の紹介
5. AI Vibe Coding のコツ
6. まとめ
7. Q & A

はじめに：講師紹介

Haruki

UNCHAIN Admin

- Web3エンジニア
- ETH Tokyo'24 finalist

X: @haruki_web3

GitHub: mashharuki



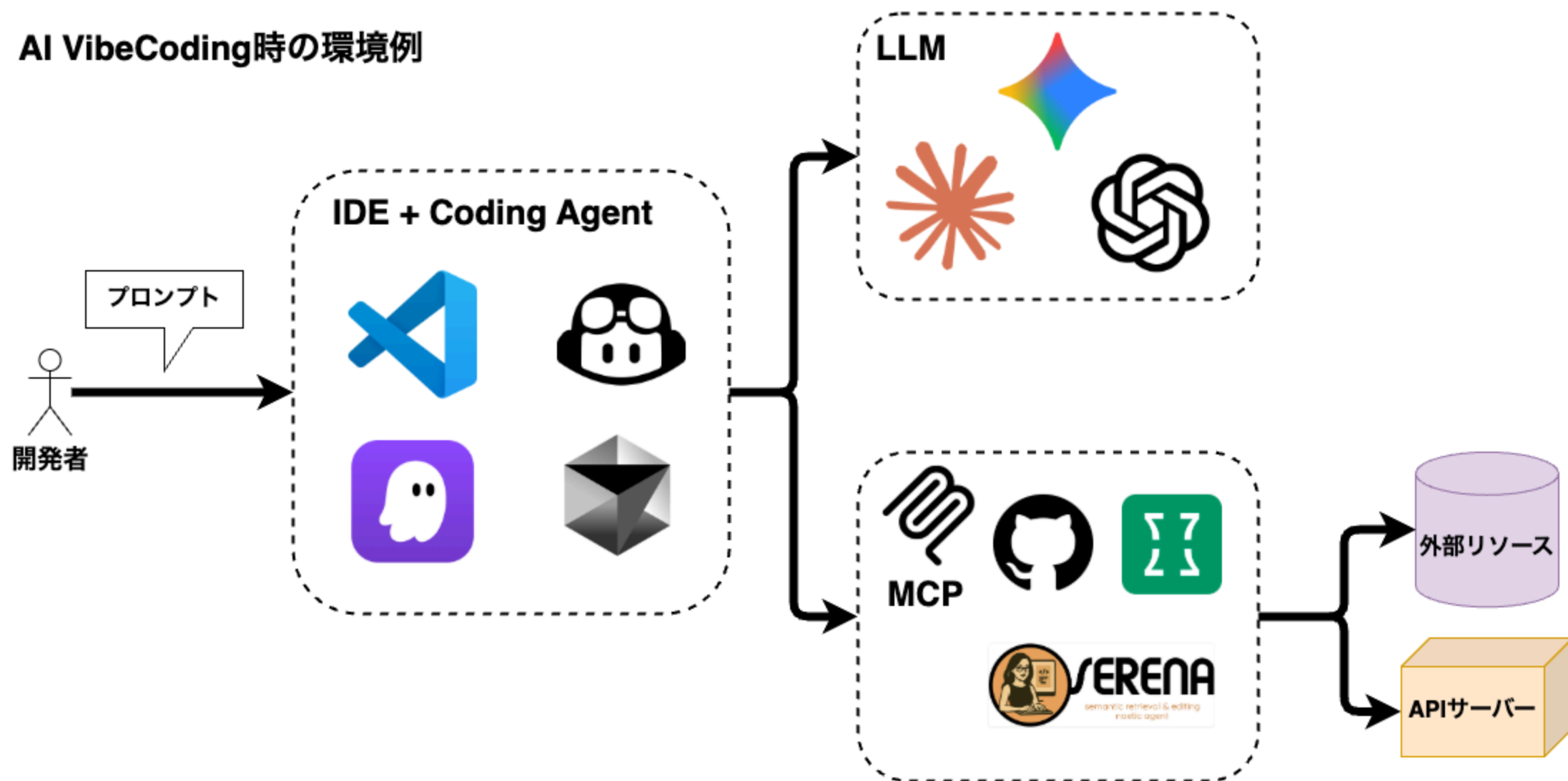
本日のゴール

このワークショップが終わる頃、あなたは...

- **LLM開発環境の構築**方法をゲット
- AIの力を最大限に引き出す **「コツ」** が手に入ります

AI VibeCodingの環境全体像

AI VibeCoding時の環境例



おすすめツール等の紹介

IDE

- **VS Code(GitHub Copilotとの併用前提)**
- **Kiro**

Coding Agent

- **GitHub Copilot Agent Mode**
- **Gemini CLI**
- **Kiroに内蔵されたCoding Agent**
- **Claude Code**

MCP

- **context7**
- **serena MCP**
- **GitHub MCP**
- **sequential-thinking**

AI Vibe Coding のコツ

これを知っているだけで、
AI駆動開発体験は劇的に向上します！
特に重要な4つのポイントをご紹介します。

コツ①：複数のモデルを使い分ける

「できること・得意なこと」はモデルで異なります

- コーディングが得意なモデル
(例: Claude)
- マルチモーダル処理ができるモデル
(例: Gemini)

コツ②：MCPでAIの機能を強化する

MCPを使うとAIはあなたのプロジェクトの
強力な「**専属アシスタント**」になります！

コツ②：MCPでAIの機能を強化する

- ファイル構造やコーディング規約をAIに直接教え込むことができます。
- これによりAIはプロジェクトの文脈を深く理解し、よりの確なコードを生成してくれるようになります。

コツ③：段階的に進める

一度に完璧を目指す必要はありません。

「小さく作って、次に進める」が成功への近道です。

コツ③：段階的に進める

- 要件定義書と設計書とタスクリストを作成する
- プロジェクトの全体をセットアップする
- スマートコントラクトを実装する
- フロントエンドを実装する

コツ④：カスタムインストラクション

AIへの「指示の出し方」も重要です。

カスタムインストラクションファイルを
事前に準備しましょう。

明確な指示が、AIの力を最大化します！

見本のGitHub

GitHub -Web3AIVibeCodingStarterKit

見本のドキュメント

以下のドキュメントを参考にしてください！

プロンプト

要件定義書

タスクリスト

見本の設定ファイル

Claude Codeの設定ファイル

Gemini CLIの設定ファイル





GitHub Copilotの設定ファイル

Kiroの設定ファイル

本日のまとめ

本日のまとめ

成功の鍵となる4つのポイント

-  複数のモデルを使い分ける
-  MCPを使ってAIの機能を強化する
-  段階的に進める
-  カスタムインストラクションのセットアップ

Thank You!