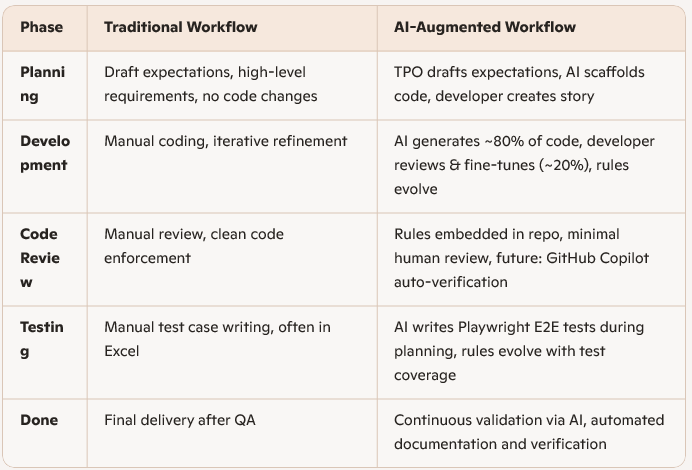
🧠 1. Journey of AI in Development

NVIDIA (Hardware Acceleration) → OpenAI ChatGPT (LLM Foundation) → RAG (Retrieval-Augmented Generation) → Agentic Planning (Autonomous Tasking) → RAG (Contextual Execution) → External Integration → Fix & Iterate → Repeat

2. Development Lifecycle Comparison  


💻3. Install & Cursor IDE Walktrough

✨ Cursor Capabilities Showcase

* **Codebase Indexing (RAG)**  
  Semantic search and context-aware generation across your repo.
* **MCP Integration**  
  Connect external tools like GitHub, Jira, Confluence, Playwright, etc.
* **Rules & Temporary Memory**

🧑‍💻 3. Prompt & Context Practices

📌 Pre-Template Rules

* **Pin rules into Cursor prompt context**  
  Always include relevant .cursorrules, agent.md, or guideline files in your prompt to guide AI behavior.
* **Frame requirements as “Don’t” rather than “Do”**  
  Because LLMs tend to overgeneralize, it's safer to specify what should be avoided.  
  **Example**: Instead of saying “Use async calls,” say “Don’t use blocking synchronous calls.”
* **Predefine .cursorrules and agent.md in each repo**  
  These files act as behavioral contracts for AI generation and should be maintained per project.

🛠️ During Coding

* **AI is not 100% bug-free or requirement-perfect**  
  Treat AI output as a draft. Expect refinement.
* **Iterate by refining context**  
  If the code doesn’t meet expectations, re-prompt with clearer requirements or corrections.  
  Example: “Fix the login redirect issue and ensure it works with AWS Cognito.”

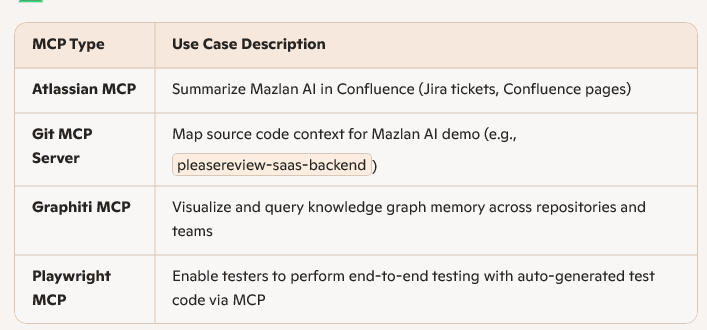
✅ Completed Code Generation

* **Verify build passes without errors**  
  Always run a build/test cycle after AI generation.
* **Summarize AI-generated changes**  
  Ask AI to explain what it changed and validate that it aligns with your expectations.
* **Optional: Share learnings via rules or guidance**  
  If a fix or pattern is valuable for others, ask AI to update .cursorrules or guideline.readme so the team avoids repeating mistakes.

🌐 4. What Is MCP? + Use Cases

🧩 MCP (Model Context Protocol)

MCP connects AI tools to external systems like GitHub, Jira, Confluence, databases, and test frameworks. It transforms AI tools into **central hubs** for development, enabling seamless integration, context sharing, and automation.

🏠 Homework Use Cases  
  
\*Git MCP Server and Graphiti are optional

|  |  |
| --- | --- |
|  |  |
|  |  |

🧪 First Demo: Atlassian MCP

**Goal**: Use Atlassian MCP to summarize Mazlan AI in Confluence  
  
**Server**:

"atlassian-mcp-server": { "url": "https://mcp.atlassian.com/v1/sse" }

**Steps**:

1. Define your goals - Implement Ideagen Hub Notification Feature
2. Connect Cursor to Atlassian MCP via mcp.json
3. Prompt to checkout repository or manual download from Bitbucket  
   plz checkout repo as below repository : @https://bitbucket.org/ideagendevelopment/pleasereview-saas-backend.git Branch : feature/pr-35985-hub-integration   
   Targeted folder : D:\AI\PleaseReview-services  
   #Use git-mc-server
4. A complete prompt for code generation.

🧾 Example Prompt

Task: Implement Ideagen Hub Notification Feature  
Objective:  
Enhance the PleaseReview-services codebase by integrating a new feature called Ideagen Hub Notification, following existing code patterns and ensuring minimal code introduction.  
  
🧩 Requirements  
- Codebase & Documentation Review  
- Scan the PleaseReview-services codebase to understand current architecture and notification handling.  
- Review Confluence documentation on Ideagen Hub integration for notifications to identify required parameters and behavior.  
- OAuth Integration  
- Implement Client Credentials OAuth flow for Ideagen Hub Notification.  
- Clarify OAuth usage in Confluence documentation.  
- Ensure support for sliding expiration tokens and token refresh via AWS Cognito (Hub).  
- API Endpoint  
- Create a new API endpoint in Ideagen Hub services to trigger notifications.  
- Ensure the endpoint adheres to existing routing and controller patterns.  
- Parameters for the endpoint should be based on Confluence documentation.  
- Notification Trigger  
- The notification should be triggered by the new Ideagen Hub API endpoint.  
- Ensure secure and reliable communication between services.  
  
✅ Development Guidelines  
- Follow existing code structure and conventions.  
- Avoid introducing unnecessary new components.  
- Add unit tests for all new logic.  
- Ensure build and test pipelines pass successfully.  
- Summaries what code changes completed in checklist.

🧪 5. Playwright MCP Discovery  
Prerequisites setup refer Playwright-MCP\readme.

🧱 Traditional Playwright Workflow

* Static code generation without real use case context
* Requires heavy manual edits:
  + Element selectors often fail
  + Timeout issues
  + Hardcoded navigation endpoints
* Limited adaptability to dynamic environments

**Start Run**:

npx playwright codegen https://tigerlily-hub.staging-0-1-0.ideagendev.com/

**Test Run**:

npx playwright test test.spec.ts --headed

🚀 Playwright MCP via Cursor IDE

Cursor enhances Playwright automation by integrating MCP for context-aware, rule-driven test generation.

**MCP Configuration (mcp.json)**{

"mcpServers": {

"playwright": {

"command": "npx",

"args": ["@playwright/mcp"],

"env": {

"PLAYWRIGHT\_HEADLESS": "false",

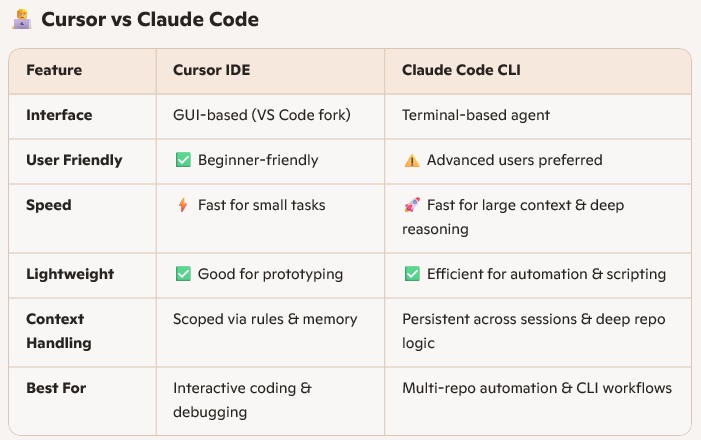
"PLAYWRIGHT\_BROWSER": "chromium"

}

}

}

}  
  
Prompt:  
@.cursorrules   
\*\*Requirements:\*\*   
1. Follow .cursorrules ruleset strictly   
2. Use Playwright MCP   
  
Steps   
1. Target URL: https://tigerlily-hub.staging-0-1-0.ideagendev.com/   
2. Fill Credentials Email textbox = xxxxxx and Password textbox = xxxx   
  
After steps done.   
1.Generate code for all steps above based on your Playwright MCP walkthrough testing. Your code need to fill on visible element only.   
2. Create code in d:\AI\Playwright-MCP\tests\   
3.Test the script and fix it if failures

Question: Cursor VS Claude Code?  
  
  
References:  
- [getzep/graphiti: Build Real-Time Knowledge Graphs for AI Agents](https://github.com/getzep/graphiti)  
- [MCP Server Directory: 5990+ updated daily | PulseMCP](https://www.pulsemcp.com/servers)  
- https://cursor.com/