

# BR 1: The "Perfect Prompt" Crafting

Ready to communicate effectively with your AI coding partner? This checklist will help you craft prompts that get great results.

## The "Perfect Prompt" Crafting Checklist

Use this checklist to make your AI prompts more effective and get better code!



### Clear Goal

Is the main purpose of the code I want clear in my mind?



### Specific Request

Am I telling the AI exactly what I want it to do, not just a vague idea?



### Context Provided

Have I mentioned the programming language, framework, or relevant existing code/files?



### Key Elements Mentioned

Have I included specific variable names, HTML IDs/classes, or text content I want to see?



### Step-by-Step (for complex tasks)

Could I break this down into smaller, simpler prompts instead of one giant one?



### Example Given (if helpful)

Would a small example of input/output or desired format help the AI?



### Plain Language

Am I using simple, direct language the AI is likely to understand?



# BR 1: The "Perfect Prompt" Crafting

Ready to communicate effectively with your AI coding partner? This checklist will help you craft prompts that get great results.



## Reviewed for Ambiguity

Could my prompt be misunderstood in any way?



## Iterative Mindset

Am I prepared to refine this prompt if the first result isn't perfect?

Note: Clear, specific, and contextual prompts are the secret to unlocking your AI's full potential and avoiding "Garbage In, Garbage Out."



# BR 2: My Vibe Coding Project Planner

This exercise will help you systematically analyze key business processes, identify current operational pain points, and uncover high-potential areas where AI can deliver significant improvements or new capabilities.

## My Vibe Coding Project Planner

Got a great idea for your next Vibe Coding project? Use this planner to map it out before you start prompting your AI!

**Project Idea/Name:**

---

---

**Core Problem this Project Solves (or Creative Goal):**

---

---

**My Target User(s) (Even if it's just me!):**

---

---

**Key MVP Features (Max 3-5 for a first version):**

(User Story Format: "As a user, I want to [action] so that [benefit].")

1. 

---
2. 

---
3. 

---
4. 

---
5. 

---

**Chosen AI Tool(s) for this Project (e.g., Cursor, Replit AI, Copilot, ChatGPT):**

---

---



# BR 2: My Vibe Coding Project Planner

This exercise will help you systematically analyze key business processes, identify current operational pain points, and uncover high-potential areas where AI can deliver significant improvements or new capabilities.

## Initial High-Level Prompts for Core Features:

(Think about your first prompt for each feature listed above)

- Feature 1 Prompt Idea: \_\_\_\_\_
- Feature 2 Prompt Idea: \_\_\_\_\_
- Feature 3 Prompt Idea: \_\_\_\_\_
- (Add more as needed)

## Success Looks Like (What's the minimum for it to be a "win"?):

---

---

Note: A little planning upfront helps you create focused prompts and guides your AI towards building exactly what you envision.



# BR 3: AI Coding Tool Comparison

Choosing the right AI coding assistant can make all the difference. This chart offers a quick comparison to help you decide.

AI Coding Tool Quick Comparison (2025 Snapshot)

Feature	GitHub Copilot	Cursor	Replit AI (Ghostwriter/Agent)	ChatGPT (for code)
Primary Use	Inline AI suggestions in editor	AI-first code editor	Browser-based IDE with AI	General chat & code generation
Interaction Style	Autocomplete, Comment-based	Chat panel, Inline AI, Contextual Edits	Chat, Inline AI, Full Project Gen.	Conversational Chat
Environment	VS Code, JetBrains, Neovim (Plugin)	Standalone Editor (VS Code based)	Web Browser	Web Interface / API
Good for Beginners?	Yes, if using a supported editor	Yes, very intuitive chat	Yes, very easy setup	Yes, very accessible
Best For...	Quick code completion, In-flow assist	Deep AI collaboration, Multi-file context	Zero setup, Rapid web prototypes, Deploy	Brainstorming, Explanations, Snippets



# BR 3: AI Coding Tool Comparison

Choosing the right AI coding assistant can make all the difference. This chart offers a quick comparison to help you decide.

Feature	GitHub Copilot	Cursor	Replit AI (Ghostwriter/Agent)	ChatGPT (for code)
Typical Cost	Subscription	Free tier + Subscription	Free Replit + AI Subscription	Free tier + Subscription (Plus)
Key AI Feature	"Ghost text" code suggestions	Integrated AI Chat, Edit with AI	Full project scaffolding, In-browser AI	Versatile text & code generation

Note: The best tool often depends on your project needs and personal workflow preferences. Don't be afraid to try a few!



# BR 4: Starter Prompt Library

Choosing the right AI coding assistant can make all the difference. This chart offers a quick comparison to help you decide.

## Starter Prompt Library: Common Web Tasks (HTML, CSS, JavaScript)

Need some inspiration for your AI prompts? Here are some ready-to-use examples for common web development tasks.

### 1. Basic HTML5 Page:

"Create a standard HTML5 boilerplate document. The page title should be '[Your Page Title]'. Include meta tags for charset and viewport. Link to an external 'style.css' file in the head, and an external 'script.js' file just before the closing body tag."

### 2. Simple Centered Heading (CSS):

"In style.css, write CSS to make the h1 heading on my page centered, with a font size of 28px and a color of navy blue."

### 3. Button Click Alert (JavaScript):

"In script.js, write JavaScript to: 1. Get a reference to an HTML button with the id 'myButton'. 2. Add an event listener so that when 'myButton' is clicked, an alert pops up saying 'Button clicked!'. Make sure this script runs after the HTML is loaded." (You'd need to add `<button id="myButton">Click Me</button>` to your HTML).

### 4. Create a Basic Navigation Bar (HTML & CSS):

"Generate the HTML for a simple navigation bar with three links: 'Home', 'About', and 'Contact'. Each link should point to '#' for now. Then, provide basic CSS in style.css to make this navigation bar horizontal, with a light gray background, and some padding around the links."

### 5. Simple Python Flask "Hello World" Route:

"Write a basic Python Flask app in a file named 'app.py'. It should have one route for '/' that returns the text 'Hello, Vibe Coder!' when accessed in a browser. Ensure the app runs when the script is executed."

Note: Adapt these prompts by changing details to fit your specific project. Use them as a springboard for your own creative prompting!



# BR 5: Beginner's Debugging Checklist

Ready to communicate effectively with your AI coding partner? This checklist will help you craft prompts that get great results.

## Debugging AI Code: Your First Aid Checklist

AI-generated code isn't always perfect. Use this checklist as your first line of defense when things don't work as expected.



### Read the Error Message Carefully

If there's an error in your browser console or terminal, copy it exactly.



### Ask the AI About the Error

Paste the error message AND the relevant code snippet into your AI assistant and ask: "What does this error mean and how can I fix this code?"



### Check for Typos (Yours or AI's)

- Are variable names spelled consistently?
- Are HTML IDs and class names in your HTML, CSS, and JavaScript perfectly matched (case-sensitive)?
- Are brackets {}, braces {}, and quotes "" correctly paired?



### Ask the AI to Explain the Code

Select the confusing or non-working code section and ask your AI: "Explain this piece of code to me step by step." Understanding what it's trying to do can help you see why it's failing.



### Add "Print" / "Console.log" Statements

- Ask the AI: "Help me add console.log statements in this JavaScript function to see the value of variableX and check if this if condition is true."
- Run the code and check the browser console/terminal for these logs. Are the values what you expect? Is the code even reaching that point?



### Simplify the Problem

- If a large block of AI code isn't working, can you comment out parts of it to see if a smaller section works?
- Can you test a single function in isolation?





# BR 5: Beginner's Debugging Checklist

Ready to communicate effectively with your AI coding partner? This checklist will help you craft prompts that get great results.



## Review Your Prompt

Did you clearly explain what you wanted? Could the AI have misunderstood? Try rephrasing your prompt with more detail or an example.



## Test in a Different Browser (Web Only)

Rarely, some issues are browser-specific.



## Take a Short Break

Sometimes, stepping away and coming back with fresh eyes helps you spot the obvious!

Note: Debugging is a normal part of coding. Use your AI assistant as a partner in this process to learn and fix issues faster.



# BR 6: GitHub "First Steps" Cheatsheet

Choosing the right AI coding assistant can make all the difference. This chart offers a quick comparison to help you decide.

## Git & GitHub: Super Simple Cheatsheet for Your AI Projects

Version control helps you save your work and collaborate. Here are the very basics! (Your AI assistant or code editor's Git integration can often help you run these!)

- **Goal: Start Tracking a New Project Locally**
  - Command (in your project folder via terminal): `git init`
  - What it does: Creates a new hidden `.git` folder in your project to store its history. (Do this once per project).
- **Goal: Prepare Changes for Saving (Staging)**
  - Command: `git add .` (the dot means "all changed files in this folder and subfolders")
  - Or: `git add specific_file.html` (to add just one file)
  - What it does: Tells Git which modified files you want to include in your next "snapshot."
- **Goal: Save Your Staged Changes (Committing)**
  - Command: `git commit -m "Your descriptive message here"`
  - Example: `git commit -m "AI generated initial HTML for homepage"`
  - What it does: Takes a snapshot of your staged files and saves it to your project's local history with your message. Commit often!
- **Goal: Connect Your Local Project to a Remote GitHub Repository**
  - (First, create an empty repository on GitHub.com)
  - Command: `git remote add origin https://github.com/YOUR_USERNAME/YOUR_REPOSITORY_NAME.git`
  - What it does: Tells your local Git where your project lives on GitHub. (Do this once per project).
- **Goal: Send Your Local Commits to GitHub (Sharing/Backup)**
  - Command: `git push origin main` (or `master` for older repos)
  - What it does: Uploads your committed snapshots to GitHub.



# BR 6: GitHub "First Steps" Cheatsheet

Choosing the right AI coding assistant can make all the difference. This chart offers a quick comparison to help you decide.

- **Goal: Get the Latest Changes from GitHub (If a Teammate Pushed)**
  - Command: `git pull origin main`
  - What it does: Downloads and merges the latest changes from GitHub into your local project.

Remember: This is super simplified! Ask your AI, "How do I use the Git features in VS Code/Cursor/Replit?" for easier ways to do this without typing commands.



# BR 7: Ethical Vibe Coding Self-Check

Ready to communicate effectively with your AI coding partner? This checklist will help you craft prompts that get great results.

## Ethical Vibe Coding: A Quick Self-Check

Vibe Coding is powerful! Use this questionnaire to ensure you're creating responsibly with AI.



### Code Origin & Licensing

- Does any of the AI-generated code look very similar to existing open-source code I've seen?
- Have I checked the terms of service for my AI coding tool regarding code ownership and usage rights?
- If building a commercial product, am I comfortable with the potential licensing implications of the AI-generated code? (Consider consulting an expert if unsure for large projects).



### Bias & Fairness

- Could any text, examples, or logic generated by the AI (or prompted by me) unintentionally reflect or perpetuate harmful biases (e.g., related to gender, race, age, etc.)?
- If my application makes decisions or categorizations, have I tested it with diverse inputs to look for unfair outcomes?
- Is the language used inclusive and respectful?



### Security & Privacy (Especially for web apps or apps handling user data)

- If my app takes user input, have I asked the AI how to protect against common web vulnerabilities (like XSS or ensuring data is handled securely on the backend if applicable)?
- If I'm storing any user data, am I doing it securely (e.g., hashing passwords, not storing sensitive info unnecessarily)?
- Have I considered who can access the data my app collects or generates?



### Transparency

- If my project's output is heavily AI-driven (e.g., AI-generated art or text for users), should I consider disclosing the use of AI?



### Intended Use & Impact

- Am I building this project for a positive purpose?
- Could it be easily misused for harmful activities? (My AI tool's terms of service likely prohibit this anyway).

Note: This isn't about having all the answers, but about fostering a mindset of responsible creation!



# BR 8: Communities & Further Learning

Choosing the right AI coding assistant can make all the difference. This chart offers a quick comparison to help you decide.

## Keep Learning & Connect!

Your Vibe Coding journey doesn't end here! Connect with communities and explore these resources to keep growing.

- **Reddit**
  - r/learnprogramming (Great for beginner questions)
  - r/webdev (For web-specific projects)
  - r/Python or r/JavaScript (Language-specific help)
  - r/SideProject or r/indiehackers (For sharing projects & getting feedback)
- **DEV Community (dev.to)**: Articles, project showcases, and discussions.
- **Stack Overflow**: For specific, well-researched coding questions.
- **Official Forums/Discord for Your AI Tools**: Check the websites for Cursor, Replit, GitHub Copilot for links to their communities.

## Recommended Beginner Resources (If Bridging to Traditional Skills - Ask AI for more!)

- **freeCodeCamp.org**: Interactive tutorials for web development (HTML, CSS, JS, and more).
- **MDN Web Docs (Mozilla Developer Network)**: Excellent, detailed documentation for all web technologies.
- **Khan Academy (Computer Programming)**: Beginner-friendly introductions to programming concepts.
- **Codecademy (some free courses)**: Interactive lessons for various languages.
- **Python.org (Official Python Tutorial)**: For those wanting to dive deeper into Python.

Note: The Vibe Coding community is growing. Sharing your experiences and learning from others is a great way to stay inspired!



# BR 9: "VC Projects" Idea Catcher

This exercise will help you systematically analyze key business processes, identify current operational pain points, and uncover high-potential areas where AI can deliver significant improvements or new capabilities.

## My Next 3 Vibe Coding Projects!

Feeling inspired? Capture your next brilliant Vibe Coding project ideas here and start planning!

### Project Idea 1:

- Brief Description/Core Vibe:\_\_\_\_\_
- Key Features (1-3):\_\_\_\_\_
- Potential AI Tools:\_\_\_\_\_
- First Smallest Step/Prompt:\_\_\_\_\_

### Project Idea 2:

- Brief Description/Core Vibe:\_\_\_\_\_
- Key Features (1-3):\_\_\_\_\_
- Potential AI Tools:\_\_\_\_\_
- First Smallest Step/Prompt:\_\_\_\_\_

### Project Idea 3:

- Brief Description/Core Vibe:\_\_\_\_\_
- Key Features (1-3):\_\_\_\_\_
- Potential AI Tools:\_\_\_\_\_
- First Smallest Step/Prompt:\_\_\_\_\_

Note: Keep that creative spark alive! Breaking down big ideas into small, AI-promptable steps is the heart of Vibe Coding.



# BR 10: Vibe Coding Glossary

Getting familiar with new terms is part of the journey. This glossary covers some key Vibe Coding concepts.

## Vibe Coding Glossary: Key Terms Made Simple

- **AI Coding Assistant:** A tool (like Copilot, Cursor, Replit AI) that uses Artificial Intelligence to help you write, debug, and understand code.
- **API (Application Programming Interface):** A way for different software programs to talk to each other and exchange data.
- **Backend:** The "behind-the-scenes" part of an application that handles logic, databases, and server operations (e.g., Python/Flask code).
- **Debugging:** The process of finding and fixing errors (bugs) in your code.
- **Deployment:** Making your application live and accessible on the internet.
- **Frontend:** The part of an application that the user sees and interacts with (e.g., HTML, CSS, JavaScript in a web browser).
- **Git:** A version control system (software) that tracks changes to your code over time.
- **GitHub:** A popular website for hosting Git repositories, sharing code, and collaborating.
- **HTML (HyperText Markup Language):** The standard language for creating the structure and content of web pages.
- **CSS (Cascading Style Sheets):** Used to describe the presentation and styling (look and feel) of web pages.
- **JavaScript (JS):** A programming language that makes web pages interactive and dynamic.
- **IDE (Integrated Development Environment):** Software that provides comprehensive tools for coding (e.g., VS Code, Cursor, Replit's editor).



# BR 10: Vibe Coding Glossary

Getting familiar with new terms is part of the journey. This glossary covers some key Vibe Coding concepts.

- **LLM (Large Language Model):** A type of AI (like GPT) trained on vast amounts of text and code, capable of understanding and generating human-like text and code.
- **MVP (Minimum Viable Product):** The simplest version of a product with just enough features to be usable by early customers and provide feedback for future development.
- **Prompt (for AI):** The instruction, question, or piece of text you give to an AI to get a response or code.
- **Prompt Engineering:** The skill of crafting effective prompts to get the best possible output from an AI.
- **Repository (Repo):** A place where your project's code and its history are stored (often on GitHub).
- **Scaffolding (Code):** Generating the basic structure or framework for a new project or feature.
- **UI (User Interface):** The visual part of an application that the user interacts with.
- **UX (User Experience):** The overall experience a user has when interacting with an application.

Note: Understanding these terms will help you communicate more effectively about your Vibe Coding projects and learn new concepts.

