

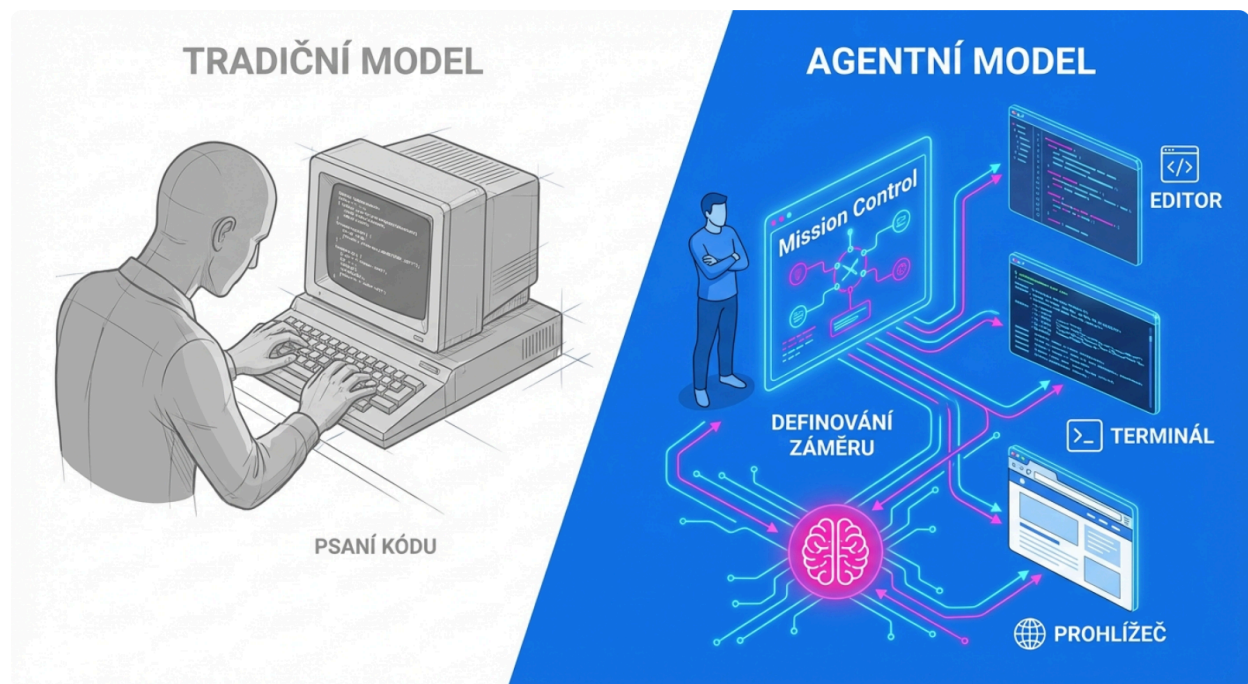
# Google Antigravity: Komplexní Architektura Transformace Uživatele na Agntního Operátora

## Úvod: Přejchod od Konzumace Obsahu k Orchestraci Vývoje

Vstupujeme do historického momentu v oblasti softwarového inženýrství, který je často označován jako éra "Agent-First" vývoje. Pro uživatele, jehož dosavadní digitální kompetence končily schopností spustit webový prohlížeč, představuje platforma Google Antigravity paradoxní, avšak obrovskou příležitost. Tradiční bariéry vstupu do světa vývoje softwaru – nutnost memorovat syntaxi jazyků, chápat kompilátory či spravovat závislosti – jsou zde nahrazeny kompetencemi nového typu: schopností definovat záměr, spravovat autonomní agenty a verifikovat výsledky. Google Antigravity není pouhým nástrojem pro doplňování kódu; je to komplexní vývojová platforma, která redefinuje roli člověka v procesu tvorby softwaru z "pisatele" (coder) na "architekta" (architect).<sup>1</sup>

Tato zpráva slouží jako vyčerpávající, technicky podložená metodika, která má za cíl provést naprostého laika procesem transformace v "Power Usera" platformy Antigravity. Cílem není naučit uživatele programovat v tradičním slova smyslu, nýbrž naučit ho efektivně řídit flotilu umělé inteligence, která programuje za něj. Tento posun vyžaduje změnu mentálního modelu: uživatel přestává být operátorem klávesnice a stává se manažerem "Mission Control" centra.<sup>3</sup>

## Evoluce Vývojového Paradigmatu: Od Syntax k Záměru



Vlevo tradiční model: člověk píše každý řádek kódu. Vpravo model Antigraivity: člověk definuje záměr (Mission Control) a AI agenti vykonávají práci v Editoru, Terminálu a Prohlížeči.

Základní premisou Antigraivity je integrace tří klíčových povrchů (surfaces) – editoru, terminálu a prohlížeče – do jednoho koherentního celku ovládaného modely jako Gemini 3 Pro. Zatímco v minulosti musel vývojář přepínat mezi psaním kódu, spouštěním příkazů a testováním v prohlížeči, Antigraivity deleguje tuto smyčku na AI agenty. Uživatel zadává úkoly na vysoké úrovni abstrakce (např. "Vytvoř přihlašovací formulář") a agent autonomně plánuje kroky, píše kód, spouští terminálové příkazy a vizuálně ověřuje výsledek v integrovaném prohlížeči.<sup>2</sup>

Tento dokument je strukturován do logických celků, které kopírují křivku učení od absolutních základů až po pokročilé techniky orchestrace více agentů. Každá sekce jde do hloubky mechanismů, které běží na pozadí, aby uživatel chápal nejen "co" má dělat, ale "proč" to funguje, což je definičním znakem Power Usera.

---

## Část 1: Technická Příprava Prostředí a Infrastruktura

Pro uživatele, který dosud využíval počítač pouze jako terminál pro webový prohlížeč, je prvním a nejdůležitějším krokem správné nastavení "strojovny". Antigraivity, ačkoliv abstrahuje složitost kódu, stále běží na standardních vývojářských technologiích. Pochopení a správná

instalace těchto prerekvizit je nezbytná pro hladký chod agentů. Agenti totiž nefungují ve vakuu; využívají nástroje nainstalované v hostitelském systému k vykonávání své práce.

## 1.1 Hardwarové a Systémové Nároky

Antigravity je robustní aplikace postavená na základech Visual Studio Code (VS Code), což zajišťuje její rozšiřitelnost a stabilitu, ale zároveň to klade určité nároky na systém. Pro plynulý běh modelů a vykreslování rozhraní je doporučeno následující vybavení:

- **Operační systém:**
  - **Windows:** Vyžaduje se Windows 10 nebo 11 (architektura x64). Pro uživatele Windows je kritické zajistit, aby měli oprávnění instalovat software do systémových složek (Administrator privileges), jelikož Antigravity bude instalovat a spouštět další podprocesy.<sup>6</sup>
  - **macOS:** Podporovány jsou verze Monterey 12 a novější. Je důležité poznamenat, že pro uživatele na platformě Apple Silicon (čipy M1/M2/M3) je k dispozici optimalizovaná verze, která by měla být preferována před Intel verzí pro lepší výkon a energetickou efektivitu.<sup>6</sup>
  - **Linux:** Podporovány jsou distribuce založené na Debianu (Ubuntu 20.04+) a Red Hatu (Fedora 36+, RHEL 8+). Linux vyžaduje specifické knihovny (glibc >= 2.28), což může být pro začátečníka bariéra, proto se pro netechnické uživatele doporučuje spíše Windows nebo macOS.<sup>7</sup>

## 1.2 Instalace Kritických Závislostí (The Dependency Chain)

Než nainstalujeme samotné Antigravity, musíme připravit půdu. Agenti v Antigravity často potřebují spouštět skripty, stahovat balíčky nebo spravovat verze souborů. K tomu využívají standardní nástroje, které musí být v systému přítomny. Absence těchto nástrojů by vedla k tomu, že agent bude hlásit chyby typu "Command not found", kterým by začátečník nerozuměl.

### 1. Google Chrome (Runtime pro Browser Agenta)

Antigravity využívá protokol Chrome DevTools Protocol pro ovládání prohlížeče. Ačkoliv má vlastní integrované okno, často spoléhá na instalaci systémového Chromu pro pokročilé renderování a debugování. Ujistěte se, že máte nainstalovanou nejnovější stabilní verzi Google Chrome.<sup>8</sup>

### 2. Node.js a NPM (Motor pro Webové Aplikace)

Toto je nejdůležitější komponenta pro moderní webový vývoj. Node.js je prostředí, které umožňuje spouštět JavaScript mimo prohlížeč (například na serveru nebo ve vašem počítači). Součástí Node.js je **npm** (Node Package Manager), což je obrovská knihovna hotových kódů.

- **Proč to potřebujete:** Když požádáte agenta "Vytvoř mi aplikaci v Reactu", agent použije příkaz `npm install react`, aby stáhl potřebné soubory. Bez Node.js by agent nemohl nic

"postavit".

- **Instalace:** Stáhněte verzi **LTS (Long Term Support)**, aktuálně verze 20 nebo vyšší, z oficiálních stránek Node.js. Vyhněte se verzím "Current", které mohou být nestabilní.<sup>8</sup>

### 3. Git (Paměť Projektu)

Git je systém pro správu verzí. Umožňuje ukládat historii změn souborů.

- **Proč to potřebujete:** I když nebudete psát příkazy jako git commit ručně, Antigravity je používá na pozadí k ukládání "bodů obnovy" (snapshots). Pokud agent udělá chybu, díky Gitu se můžete vrátit zpět. Také je nutný pro stahování šablon projektů (tzv. klonování).<sup>8</sup>

### 4. Python (Volitelné, ale Doporučené)

Pro pokročilejší analýzu dat nebo backendové skripty agenti často volí jazyk Python. Instalace verze 3.10+ zajistí, že pokud agenta požádáte o "skript na třídění souborů", bude ho schopen okamžitě spustit.<sup>8</sup>

## 1.3 Proces Instalace Antigravity a Prvotní Konfigurace

Jakmile jsou závislosti na místě, můžeme přistoupit k instalaci samotné platformy.

1. **Stahování Binárních Souborů:** Navštivte oficiální portál [antigravity.google/download](https://antigravity.google.com/download). Web automaticky detekuje váš operační systém. Stáhněte instalátor.<sup>6</sup>
2. **Instalační Wizard:** Spustěte stažený soubor. Na Windows a macOS je proces přímočarý (Next -> Next -> Finish). Na Linuxu může být nutné přidat repozitář do správce balíčků, což je pokročilejší úkon popsáný v dokumentaci.<sup>7</sup>
3. **Autorizace a Přihlášení:** Při prvním spuštění vás Antigravity vyzve k přihlášení. V současné fázi "Public Preview" je nutné použít osobní Google účet (Gmail). Toto přihlášení je klíčové, protože zpřístupňuje cloudové modely Gemini a synchronizuje vaše nastavení.<sup>4</sup>
4. **Výběr Modelu:** Budete vyzváni k výběru výchozího AI modelu. Pro začátečníka je absolutně kritické zvolit **Gemini 3 Pro**. Tento model má nejlepší schopnosti "reasoning" (uvažování) a plánování. Modely jako "Flash" jsou sice rychlejší, ale dělají více logických chyb, což by začátečníka mátl. Gemini 3 Pro poskytuje nejlepší poměr mezi rychlostí a kvalitou výstupu pro komplexní úkoly.<sup>5</sup>

---

## Část 2: Anatomie Rozhraní "Mission Control" a Navigace

Po úspěšné instalaci se uživatel ocitne v prostředí, které se radikálně liší od běžných textových editorů. Antigravity nezahajuje práci pohledem na prázdný textový soubor, ale pohledem na tzv. **Agent Manager**, často nazývaný "Mission Control". Pochopení této filozofie je klíčové pro

přechod na Power Usera.

## 2.1 Koncept Dvou Povrchů (Surfaces)

Antigravity operuje na dvou hlavních úrovních rozhraní, mezi kterými uživatel přepíná. Schopnost plynule přecházet mezi těmito režimy je základní dovedností.<sup>12</sup>

### 1. Agent Manager (Mission Control):

- Toto je strategická vrstva. Zde se neprogramuje, zde se řídí.
- Uživatel zde vidí přehled všech aktivních agentů, jejich úkoly a stav.
- Je to místo pro zadávání komplexních, více krokových instrukcí (např. "Refaktoruj celou databázovou vrstvu").
- Obsahuje vizualizaci **Artefaktů** (plány, seznamy úkolů, screenshoty), které slouží ke kontrole práce agenta.<sup>3</sup>

### 2. Editor View (Taktická Vrstva):

- Toto rozhraní vychází z Visual Studio Code.
- Slouží pro detailní pohled na kód, pokud je to potřeba, nebo pro rychlé, lokální úpravy.
- Obsahuje integrovaný **Terminál** a panel pro **Chat s Agentem** pro kontextuální dotazy nad konkrétním souborem.<sup>1</sup>

## Anatomie 'Mission Control' (Agent Manager)

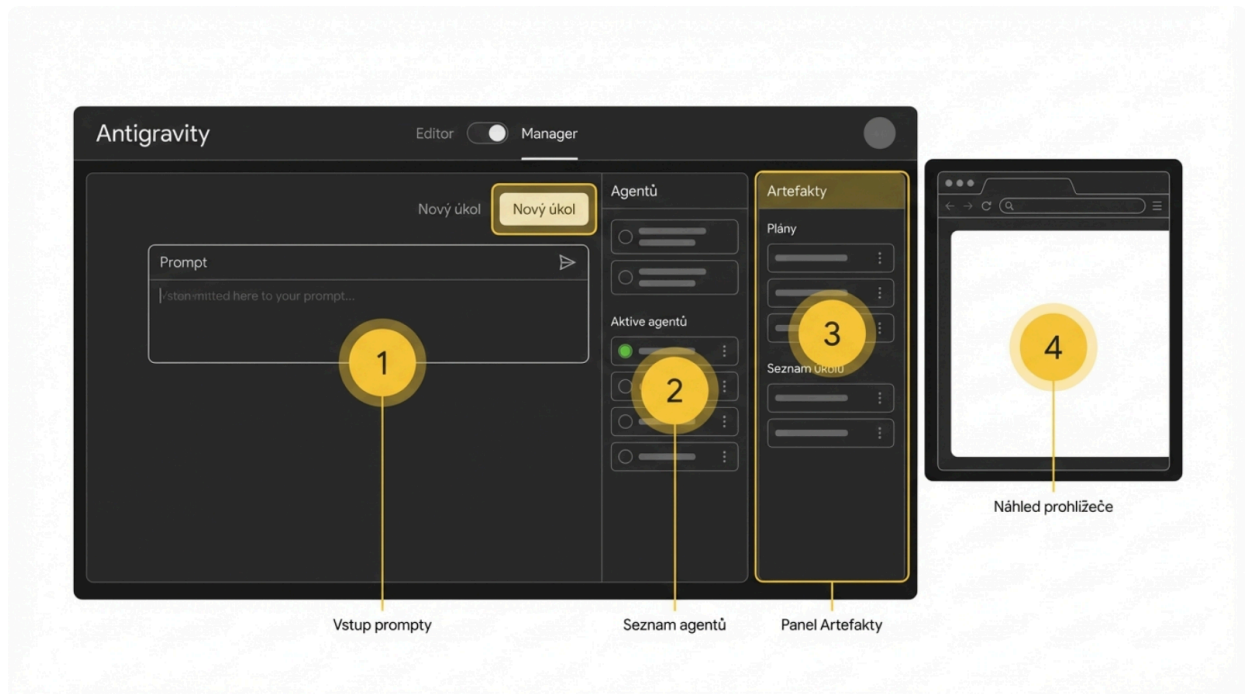


Schéma rozhraní Agent Manageru. 1: Vstupní pole pro prompty (zadávání úkolů). 2: Seznam aktivních agentů. 3: Panel Artefaktů (Plány, Task Listy). 4: Náhled Browser Agentu.

## 2.2 Režimy Interakce: Planning Mode vs. Fast Mode

Jedním z nejčastějších bodů zmatení pro začátečníky je volba správného režimu práce agenta. AntigraVity nabízí dva fundamentálně odlišné módy, které ovlivňují, jak agent přistupuje k řešení problému.<sup>14</sup>

- **Planning Mode (Plánovací režim):**

- **Charakteristika:** Toto je výchozí a bezpečnější režim pro začátečníky. Agent v tomto módu **nezačne psát kód okamžitě**. Nejprve analyzuje zadání, prozkoumá existující soubory a vytvoří dokument zvaný **Implementation Plan** (Implementační plán).
- **Proces:** Uživatel musí tento plán zkontrolovat a schválit. Teprve po schválení agent začne vykonávat změny.
- **Vhodné pro:** Tvorbu nových funkcí, komplexní změny napříč mnoha soubory, nebo když si uživatel není jistý, jak přesně má výsledek vypadat. Je to ekvivalent diskuze s architektem před stavbou domu.

- **Fast Mode (Rychlý režim):**

- **Charakteristika:** Agent přeskakuje fázi explicitního plánování a okamžitě provádí akce (editace souborů, spouštění příkazů).
- **Proces:** Zadání -> Okamžitá exekuce.



- **Vhodné pro:** Rychlé opravy chyb (bugfixy), malé změny (např. "Změň barvu tohoto tlačítka na modrou"), nebo jednoduché refaktoringy.
- **Riziko:** Pokud je zadání nejednoznačné, agent může v Fast Mode "rozbit" více věcí, než opraví, protože nemá fázi schválení plánu.

**Strategie pro začátečníka:** Vždy začínějte v **Planning Mode**. Donutí vás to zkontrolovat, zda AI pochopila váš záměr, ještě než začne nevratně měnit projekt. Považujte Fast Mode za nástroj pro pokročilé, kteří přesně vědí, co chtějí.<sup>16</sup>

---

## Část 3: První Projekt – Metodika

### "Prompt-Review-Verify"

Teorie je nezbytná, ale praxe je nenahraditelná. Následující sekce provede uživatele vytvořením první reálné webové aplikace – Pomodoro časovače. Tento projekt byl vybrán proto, že demonstruje interakci všech tří povrchů: Editoru (HTML/CSS), Terminálu (spuštění serveru) a Prohlížeče (testování UI). Postupujeme metodikou **Prompt-Review-Verify**, která je základním cyklem práce v Antigravity.<sup>17</sup>

#### Krok 1: Inicializace a Strukturování Promptu (Zadání)

Otevřete **Agent Manager** a vytvořte novou složku projektu (např. moje-pomodoro). Do chatovacího okna zadejte prompt. Pro uživatele, který neumí programovat, je kvalita promptu rozhodující. Prompt musí být deskriptivní, nikoliv technický.

##### Doporučený Prompt:

*"Create a functional Pomodoro timer web app using HTML, CSS, and JavaScript. It should have a large 25-minute default timer display, distinct start/pause/reset buttons, and a visual progress ring that animates as time decreases. The design should be modern and clean."*<sup>17</sup>

Všimněte si klíčových slov: "functional" (funkční), "visual progress ring" (vizuální prvek), "HTML, CSS, JavaScript" (technologie). Tím dáváte agentovi jasné mantinely.

#### Krok 2: Review (Schválení Plánu)

Po odeslání promptu agent (v Planning Mode) vygeneruje **Implementation Plan**. Toto je kritický moment. Agent vám předloží seznam kroků:

1. Vytvořit index.html s strukturou pro časovač a tlačítka.
2. Vytvořit style.css pro kruhový progress bar.
3. Vytvořit script.js pro logiku odpočítávání.

Jako uživatel musíte tento plán přečíst. Dává smysl? Nechybí tam tlačítko Reset? Pokud je vše v pořádku, klikněte na tlačítko **"Approve"** (Schválit).<sup>2</sup>

### Krok 3: Exekuce a Verifikace v Prohlížeči (Verify)

Po schválení agent začne generovat kód. Následně se stane něco, co odlišuje Antigraivity od jiných nástrojů: **Agent si vyžádá přístup k prohlížeči.**

1. **Povolení:** Potvrďte dialogové okno, které umožňuje agentovi spustit tzv. **Browser Agent**.
2. **Autonomní Testování:** Agent sám spustí lokální server (např. pomocí Pythonu nebo Node.js), otevře integrované okno prohlížeče a načte vaši aplikaci.
3. **Vizuální Kontrola:** Uvidíte, jak agent pohybuje kurzorem, kliká na tlačítko "Start", čeká, zda se časovač rozběhne, a kliká na "Reset".
4. **Výstup:** Agent vygeneruje **Walkthrough Artifact** – sadu screenshotů nebo videozáznam z testování, spolu s reportem "Timer counts down correctly".<sup>4</sup>

### Krok 4: Iterace a Zpětná Vazba (Feedback Loop)

První verze je málokdy dokonalá. Power User ví, jak iterovat. Řekněme, že chceme přidat zvuk. Místo hledání, kam napsat kód pro zvuk, jednoduše napíšeme do chatu:

*"Update the code to add a sound effect notification when the timer hits zero. Also, add an input field in the HTML so the user can set a custom time in minutes instead of the default 25."*<sup>17</sup>

Agent opět navrhne plán úprav, vy ho schválíte a cyklus se opakuje. Tímto způsobem "socháte" software, aniž byste se dotkli dláta.

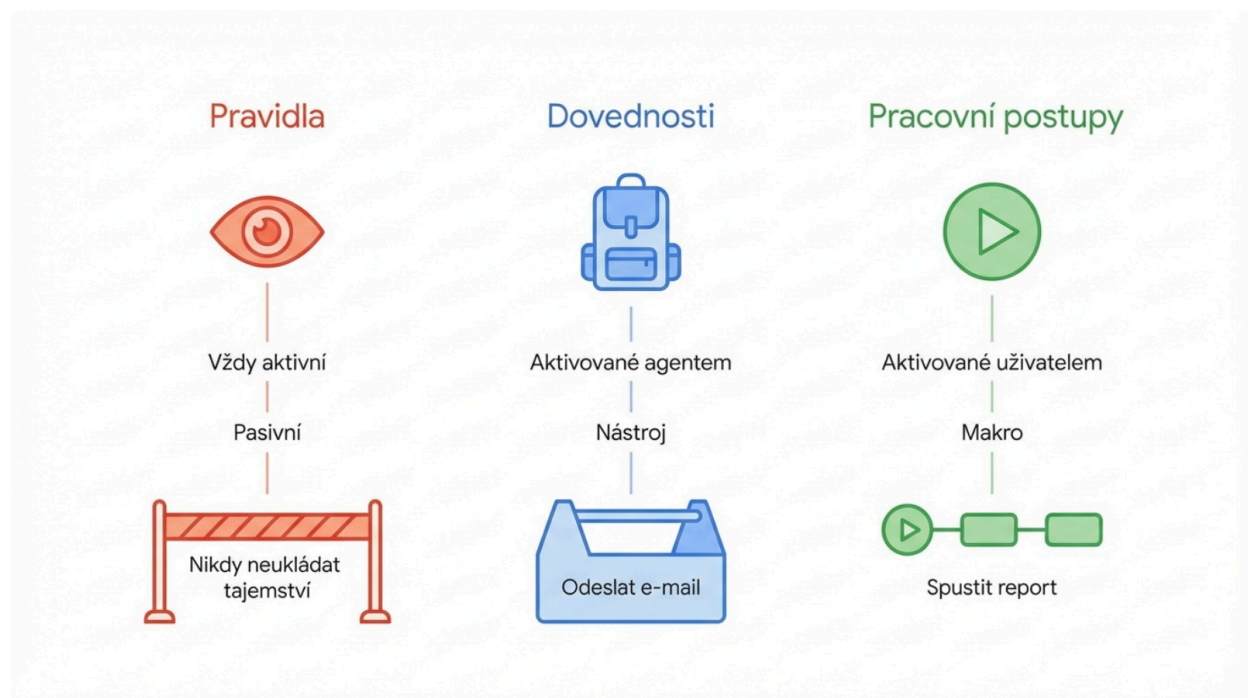
---

## Část 4: Pokročilá Architektura – Skills, Rules a Workflows

Aby se z uživatele stal skutečný Power User, musí přestat zadávat repetitivní příkazy. Musí naučit Antigraivity své preference a standardy. K tomu slouží hierarchie automatizace: **Skills (Dovednosti)**, **Rules (Pravidla)** a **Workflows (Pracovní postupy)**. Tyto mechanismy transformují agenta z obecného pomocníka na specializovaného experta pro vaše potřeby.<sup>19</sup>



# Hierarchie Automatizace: Rules, Skills a Workflows



Rozdíl mezi nástroji automatizace. Rules jsou mantinely (Guardrails), Skills jsou nástroje v batohu agenta, Workflows jsou makra spouštěná uživatelem.

## 4.1 Skills (Dovednosti): Modulární Schopnosti

Skills jsou balíčky znalostí a instrukcí, které agent "vybalí" pouze tehdy, když jsou relevantní pro daný úkol. Představte si je jako aplikace, které si agent instaluje do svého mozku. Toto zabraňuje "přehlcení kontextu" (Context Saturation) – agent nemusí držet v paměti všechno najednou, ale načte si instrukce pro práci s databází jen tehdy, když po něm chcete práci s databází.<sup>19</sup>

- **Struktura:** Skill je definován adresářem obsahujícím soubor SKILL.md.
- **Umístění:**
  - **Lokální (Workspace):** .agent/skills/ – platí jen pro konkrétní projekt.
  - **Globální:** ~/.gemini/antigravity/skills/ – platí pro všechny vaše projekty.<sup>19</sup>
- **Syntaxe SKILL.md:** Soubor se skládá ze dvou částí. **YAML Frontmatter** (hlavička s metadaty) a **Markdown Body** (tělo s instrukcemi).

## Příklad Skillu "Git Commit Formatter"

<sup>21</sup>: Tento skill naučí agenta, jak správně psát zprávy při ukládání kódu (commity).

## **name: git-commit-formatter description: Use this skill when the user asks to commit changes or save work to git.**

Goal: Create a standardized commit message following Conventional Commits.

Instructions:

1. Analyze the changes in the codebase.
2. Determine the type (feat, fix, docs, style, refactor).
3. Write a concise subject line under 50 characters.
4. Do not execute the commit command, only propose the message.  
Jakmile tento soubor uložíte, kdykoliv řeknete agentovi "Commitni to", on automaticky aktivuje tento skill a vytvoří perfektní zprávu, aniž byste mu museli vysvětlovat pravidla.

### **4.2 Rules (Pravidla): Pasivní Mantinely**

Pravidla jsou "guardrails" – bezpečnostní zábradlí. Jsou vždy aktivní a agent je nesmí porušit. Jsou injektována do systémového promptu každého agenta.<sup>19</sup>

- **Umístění:** .agent/rules/.
- **Použití:** Pro vynucení bezpečnosti nebo striktních konvencí.
- **Příklady:**
  - "Nikdy neukládej hesla nebo API klíče do kódu."
  - "Vždy používej TypeScript místo čistého JavaScriptu."
  - "Při úpravě UI vždy zkontroluj responzivitu pro mobilní zařízení."

### **4.3 Workflows (Pracovní Postupy): Aktivní Makra**

Zatímco Skills se aktivují automaticky na základě kontextu, Workflows jsou sekvence kroků, které spouští uživatel explicitním příkazem (např. /test nebo /deploy). Jsou to "recepty" na složité operace.<sup>22</sup>

- **Umístění:** .agent/workflows/.
- **Příklad Workflow pro Code Review:**  
Můžete definovat workflow, které po zadání příkazu /review:
  1. Spustí všechny testy.
  2. Provede analýzu kódu (linting).
  3. Vygeneruje souhrnný report chyb.
  4. Navrhne opravy pro nalezené chyby.

---

## **Část 5: Integrace Externí Dokumentace a MCP**

Skutečný expert se nespolehá jen na to, co se AI naučila během tréninku (což může být zastaralé). Power User propojuje Antigravity s aktuální dokumentací nástrojů a knihoven pomocí **Model Context Protocol (MCP)** a standardu `llms.txt`.

## 5.1 Problém: Halucinace o Knihovnách

Když vyjde nová verze knihovny (např. React 19), AI model o ní nemusí vědět a bude používat staré postupy z verze 18. To vede k nefunkčnímu kódu. Řešením je "nakrmit" agenta aktuální dokumentací.

## 5.2 Řešení: `llms.txt` a `mcpdoc`

Standard `llms.txt` je způsob, jakým moderní dokumentace prezentuje informace pro AI – v čistém textu, bez vizuálního balastu. Nástroj `mcpdoc` funguje jako překladač, který umožní agentovi v Antigravity "číst" tyto soubory přímo z internetu.<sup>23</sup>

## 5.3 Postup Integrace MCP Serveru

Pro laika to může znít složitě, ale v Antigravity je to otázka konfigurace:

1. **Otevření Konfigurace:** V Editoru klikněte na panel Agent -> tři tečky (...) -> **MCP Servers** -> **Manage MCP Server** -> **Edit Config**.
2. **Vložení Konfigurace:** Do souboru `mcp_config.json` vložte definici serveru `mcpdoc`.

JSON

```
{
  "mcpServers": {
    "docs-server": {
      "command": "uvx",
      "args":
    }
  }
}
```

*Vysvětlení:* Příkaz `uvx` (součást balíčku `uv`, který je moderní náhradou `pipu` pro Python) stáhne a spustí nástroj `mcpdoc`. Parametr `--urls` říká, které dokumentace má mít agent k dispozici.<sup>23</sup>

3. **Použití:** Nyní můžete v chatu napsat: *"Check the ADK docs regarding initialization and write the setup code."* Agent si na pozadí stáhne aktuální informace z definované URL a použije je. Tím zajistíte, že váš kód bude vždy aktuální.

---

## Část 6: Strategie Efektivity a Best Practices

Být Power Userem není jen o znalosti funkcí, ale o rychlosti a efektivitě práce (Flow).

## 6.1 Práce s Artefakty a Verifikace

Nejčastější chybou začátečníků je slepá důvěra v textový výstup chatu. Power User ignoruje "povídání" agenta a soustředí se na **Artefakty**.

- **Task List:** Sledujte tento dynamický seznam. Pokud agent začne "bloudit" (řeší úkoly, které jste nezadali), okamžitě zasáhněte. Task List je obrazem myšlenkového procesu agenta.<sup>2</sup>
- **Diff Views:** Při schvalování změn (Review Changes) se naučte číst "diffy" (rozdílové pohledy). Červená barva znamená smazaný řádek, zelená přidaný. I bez znalosti kódu můžete poznat podezřelé změny, např. když agent smaže obrovský blok textu, který měl zůstat.<sup>4</sup>

## 6.2 Strategické Přepínání Modelů (Model Toggling)

Antigravity umožňuje měnit "mozek" agenta za běhu. Každý model má jiné silné stránky a cenu (v tokenové spotřebě/rychlosti).

- **Gemini 3 Pro:** "Těžká váha". Používejte pro úvodní plánování, architekturu, řešení složitých logických problémů a debugování, kdy si nevíte rady.
- **Gemini 3 Flash:** "Sprinter". Přepněte na něj pro rychlé iterace (změna textů, barev, přidání jednoduché funkce). Je levnější, má nižší latenci a pro jednoduché úkoly je stejně efektivní jako Pro.<sup>10</sup>
- **Claude Sonnet 4.5:** Alternativa. Pokud se Gemini zasekne v kruhu a nemůže vyřešit problém, přepnutí na Claude často pomůže, protože má jiný způsob "uvažování".<sup>2</sup>

## 6.3 Klávesové Zkratky pro Rychlost

Myš je pomalá. Naučte se tři klíčové zkratky, které pokryjí 90 % navigace:

- Cmd/Ctrl + L: Okamžité otevření/zavření panelu chatu s agentem (Agent Panel). Umožňuje rychlé zadání příkazu a návrat k práci.<sup>4</sup>
- Cmd/Ctrl + E: Přepínání fokusu mezi Editorem a Agent Managerem. Rychlý skok z "taktického" do "strategického" pohledu.<sup>26</sup>
- Cmd/Ctrl + ,: Rychlý přístup k nastavení (Settings), např. pro úpravu pravidel nebo MCP serverů.<sup>27</sup>

---

## Závěr a 30denní Plán Růstu

Transformace z pasivního uživatele prohlížeče na architekta softwaru pomocí Google Antigravity je cesta, která nevede přes memorování syntaxe, ale přes zvládnutí **záměru (Intent)**, **kontextu (Context)** a **verifikace (Verification)**.

Níže je navržený intenzivní učební plán pro dosažení úrovně Power Usera:

Týden	Zaměření	Cíle a Aktivity
1.	<b>Základy &amp; Prompting</b>	Instalace prostředí. Vytvoření aplikace Pomodoro. Pochopení cyklu Prompt-Plan-Verify. Experimentování s různými formulacemi promptů.
2.	<b>Browser Agent &amp; Iterace</b>	Práce s Browser Agentem. Tvorba složitější aplikace (např. To-Do list s ukládáním do paměti prohlížeče). Naučit se iterovat pomocí zpětné vazby ("Make the buttons bigger").
3.	<b>Automatizace (Skills &amp; Rules)</b>	Vytvoření prvního vlastního Skillu (např. pro formátování CSS). Nastavení globálních pravidel (Rules). Pochopení adresářové struktury .agent.
4.	<b>Integrace &amp; MCP</b>	Napojení externí dokumentace přes mcpdoc. Vytvoření projektu využívajícího externí knihovnu (např. grafy pomocí Chart.js) s využitím načtené dokumentace.

Google Antigravity není jen IDE; je to multiplikátor lidského potenciálu. Dává uživateli do rukou tým virtuálních inženýrů. Vaším úkolem je stát se jejich schopným manažerem.

### Citovaná díla

1. Google Antigravity Tool (IDE): What It Is and How Developers ..., použito ledna 22, 2026,

- <https://medium.com/@expertappdevs/google-antigravity-tool-ide-what-it-is-and-how-developers-benefit-50119f8d886c>
2. Build with Google Antigravity, our new agentic development platform, použito ledna 22, 2026,  
<https://developers.googleblog.com/build-with-google-antigravity-our-new-agentic-development-platform/>
  3. použito ledna 22, 2026,  
<https://codelabs.developers.google.com/getting-started-google-antigravity#:~:text=This%20interface%20acts%20as%20a.developer%20acts%20as%20an%20architect.>
  4. Getting Started with Google Antigravity, použito ledna 22, 2026,  
<https://codelabs.developers.google.com/getting-started-google-antigravity>
  5. Introducing Google Antigravity, a New Era in AI-Assisted Software Development, použito ledna 22, 2026,  
<https://antigravity.google/blog/introducing-google-antigravity>
  6. Getting Started with Google Antigravity - Setup Tutorial & Guide 2026, použito ledna 22, 2026, <https://antigravity.im/getting-started>
  7. 🚀 Google Antigravity — Set Up Guide, použito ledna 22, 2026,  
<https://mayur-ds.medium.com/google-antigravity-set-up-guide-c15f8df80c6c>
  8. použito ledna 22, 2026, <https://antigravity.codes/tutorial>
  9. How to Use Antigravity: Complete Step-by-Step Tutorial - Skywork.ai, použito ledna 22, 2026, <https://skywork.ai/blog/how-to-use-antigravity/>
  10. použito ledna 22, 2026,  
<https://medium.com/ai-software-engineer/i-tested-google-antigravity-gemini-3-flash-upgrade-everyones-missing-this-044efafcba16#:~:text=Gemini%203%20Flash%20scores%2090.4,series%20and%20Gemini%203%20Pro.>
  11. Gemini 3 Flash in Google Antigravity, agentic coding at the speed of thought, použito ledna 22, 2026,  
<https://antigravity.google/blog/gemini-3-flash-in-google-antigravity>
  12. použito ledna 22, 2026,  
<https://pub.towardsai.net/why-everyone-is-talking-about-google-antigravity-the-era-of-agentic-ai-coding-bad5e9e78326#:~:text=Antigravity%20forks%20the%20open%2Dsource.Editor%20and%20the%20Agent%20Manager%20.>
  13. Google Antigravity Documentation, použito ledna 22, 2026,  
<https://antigravity.google/docs/home>
  14. použito ledna 22, 2026,  
<https://sdh.global/blog/ai-ml/how-google-antigravity-makes-ai-assisted-coding-actually-work-for-developers/#:~:text=Workspace%20Configuration%20and%20Agent%20Settings,-After%20completing%20the&text=First%2C%20select%20the%20appropriate%20planning,simpler%2C%20localized%20tasks%20more%20efficiently.>
  15. Google Antigravity AI IDE: New Era of Coding - Royal Cyber, použito ledna 22, 2026,  
<https://www.royalcyber.com/blogs/ai-services/google-antigravity-ai-ide-shaping-new-era/>

16. Sometimes, Fast mode is better than Planning mode! : r/GoogleAntigravityIDE - Reddit, použito ledna 22, 2026,  
[https://www.reddit.com/r/GoogleAntigravityIDE/comments/1pvib7d/sometimes\\_fast\\_mode\\_is\\_better\\_than\\_planning\\_mode/](https://www.reddit.com/r/GoogleAntigravityIDE/comments/1pvib7d/sometimes_fast_mode_is_better_than_planning_mode/)
17. Google Antigravity Tutorial for Beginners: Build Your First App (Step-by-Step) - YouTube, použito ledna 22, 2026,  
<https://www.youtube.com/watch?v=-0lrz8G0PEE>
18. An Honest Review of Google Antigravity - DEV Community, použito ledna 22, 2026,  
<https://dev.to/fabianfrankwerner/an-honest-review-of-google-antigravity-4g6f>
19. Tutorial : Getting Started with Google Antigravity Skills - Medium, použito ledna 22, 2026,  
<https://medium.com/google-cloud/tutorial-getting-started-with-antigravity-skills-864041811e0d>
20. Authoring Google Antigravity Skills, použito ledna 22, 2026,  
<https://codelabs.developers.google.com/getting-started-with-antigravity-skills>
21. rominirani/antigravity-skills: Sample Google Antigravity Skills - GitHub, použito ledna 22, 2026, <https://github.com/rominirani/antigravity-skills>
22. Rules / Workflows - Google Antigravity Documentation, použito ledna 22, 2026,  
<https://antigravity.google/docs/rules-workflows>
23. The Modern Developer's Blueprint: Integrating Documentation ..., použito ledna 22, 2026,  
<https://medium.com/google-cloud/the-modern-developers-blueprint-integrating-documentation-aware-agents-in-google-antigravity-016b87327867>
24. Tutorial : Getting Started with Google Antigravity | by Romin Irani - Medium, použito ledna 22, 2026,  
<https://medium.com/google-cloud/tutorial-getting-started-with-google-antigravity-b5cc74c103c2>
25. Gemini 3 Flash: frontier intelligence built for speed - Google Blog, použito ledna 22, 2026,  
<https://blog.google/products-and-platforms/products/gemini/gemini-3-flash/>
26. Google Antigravity Documentation, použito ledna 22, 2026,  
<https://antigravity.google/docs>
27. Settings - Google Antigravity Documentation, použito ledna 22, 2026,  
<https://antigravity.google/docs/settings>