

無程式碼遊戲開發

基於多代理協同與擴散模型之生成式 AI 解決方案

No-Code Game Development

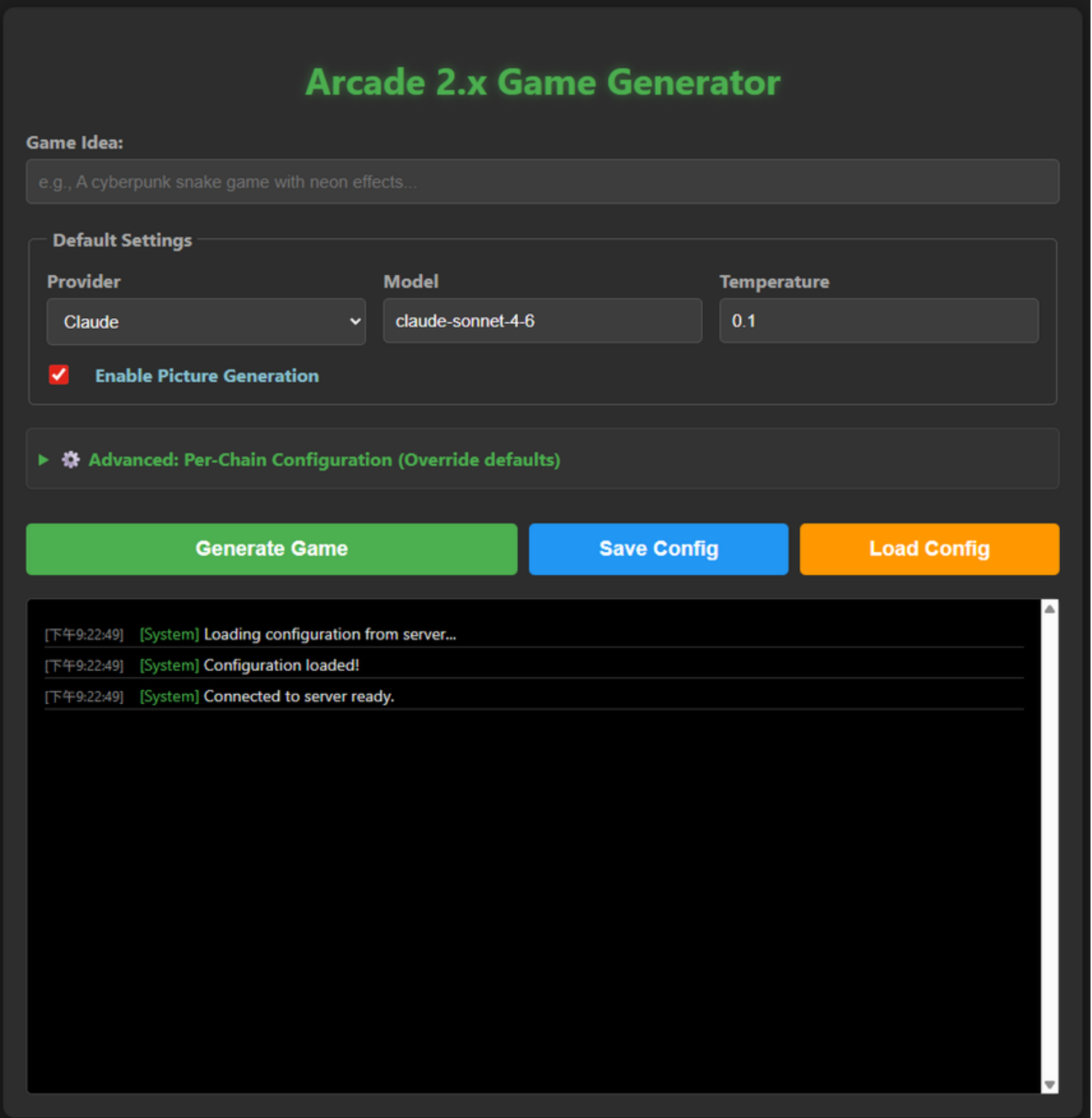
Generative AI via Multi-Agent and Diffusion Models

指導教授: 李同益

專題成員: 丁泓元、余嘉憲、曾翊庭

簡介

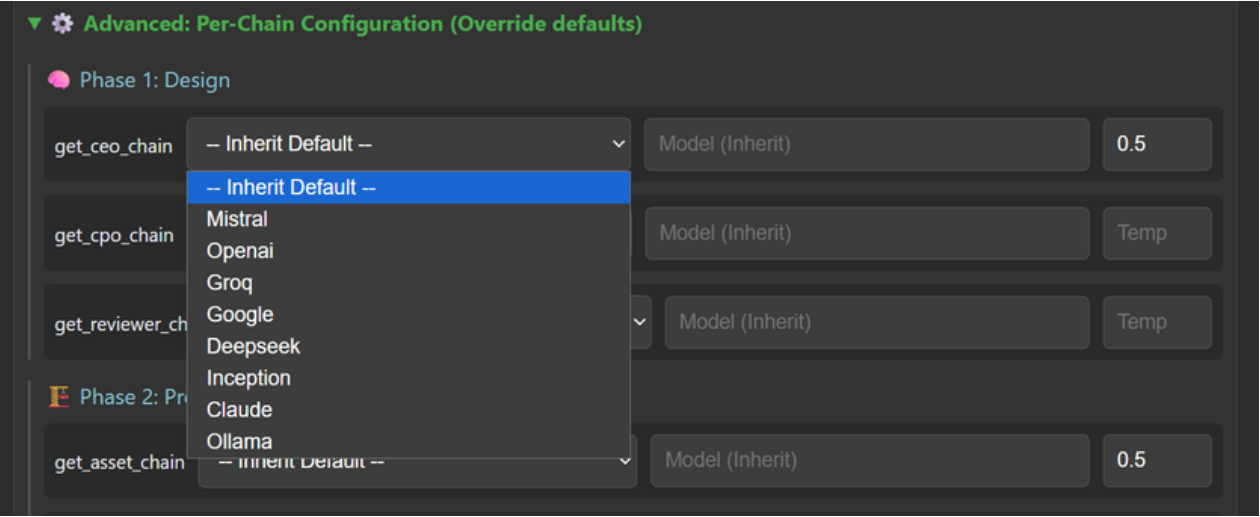
- 僅簡短一句 prompt 即可自動生成遊戲
- LLM + Diffusion 構建遊戲程式碼及素材
- 使用本地模型壓縮 token 用量
- Diffusion 減少繪製素材的程式碼占比
- 提供最小可行方案降低前期開發成本



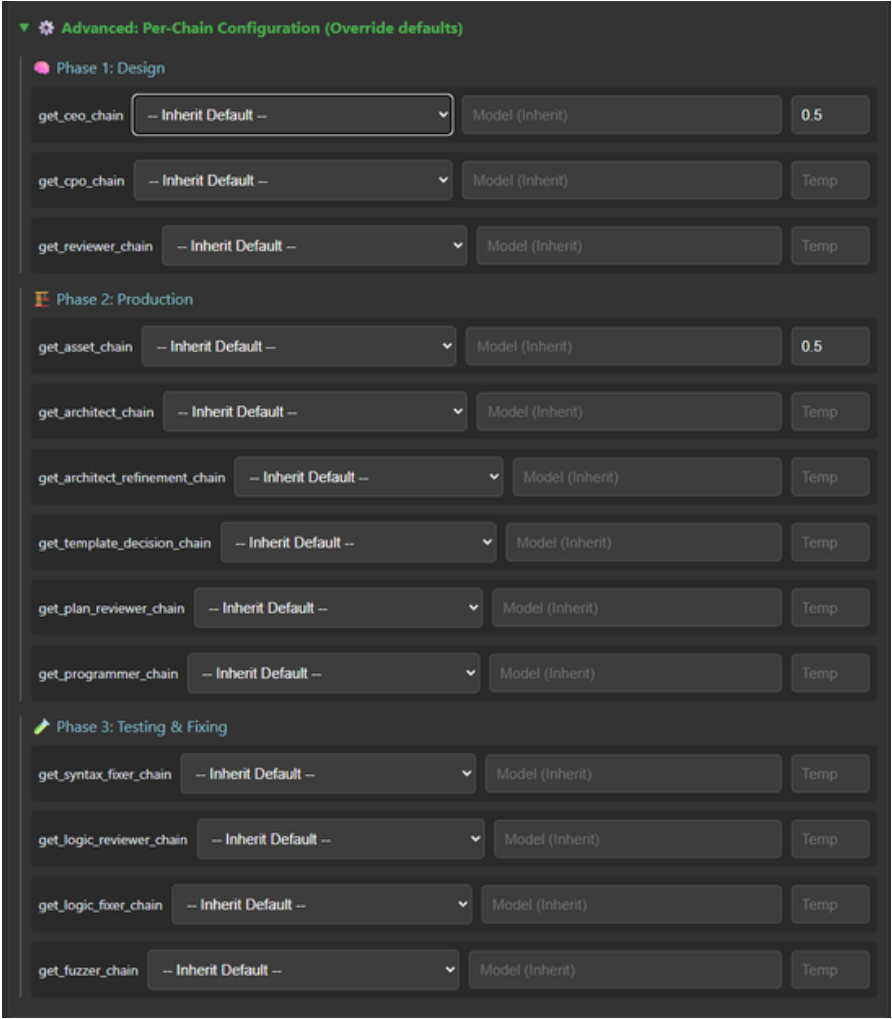
The screenshot shows the 'Arcade 2.x Game Generator' web interface. At the top, the title 'Arcade 2.x Game Generator' is displayed in green. Below it is a 'Game Idea:' section with a text input field containing the example prompt: 'e.g., A cyberpunk snake game with neon effects...'. Underneath is the 'Default Settings' section, which includes three input fields: 'Provider' (set to 'Claude'), 'Model' (set to 'claude-sonnet-4-6'), and 'Temperature' (set to '0.1'). There is also a checkbox labeled 'Enable Picture Generation' which is checked. Below these settings is a section for 'Advanced: Per-Chain Configuration (Override defaults)' with a gear icon and a right-pointing arrow. At the bottom of the settings are three buttons: 'Generate Game' (green), 'Save Config' (blue), and 'Load Config' (orange). The bottom half of the interface is a large black area for logs, showing three system messages: '[System] Loading configuration from server...', '[System] Configuration loaded!', and '[System] Connected to server ready.'.

圖一、遊戲生成器介面

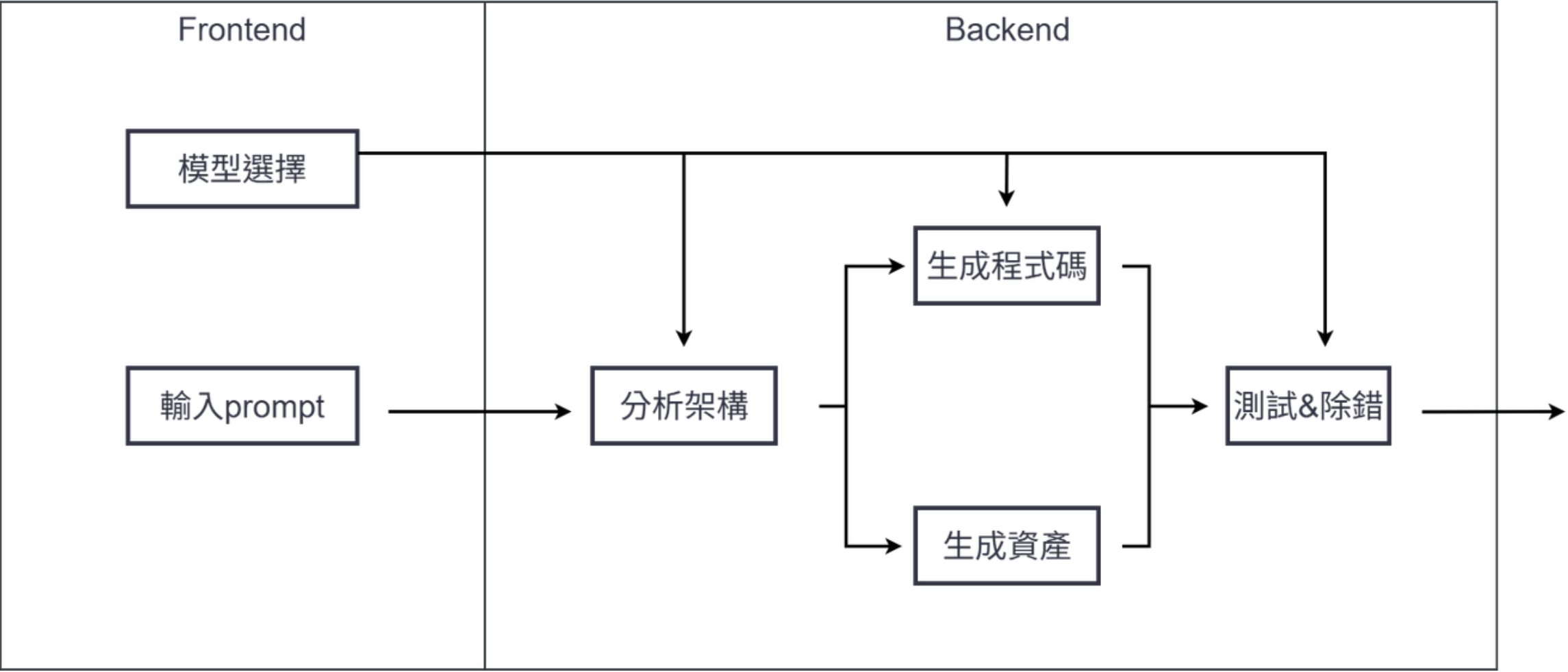
系統架構圖



圖三、模型選擇



圖四、模型設定

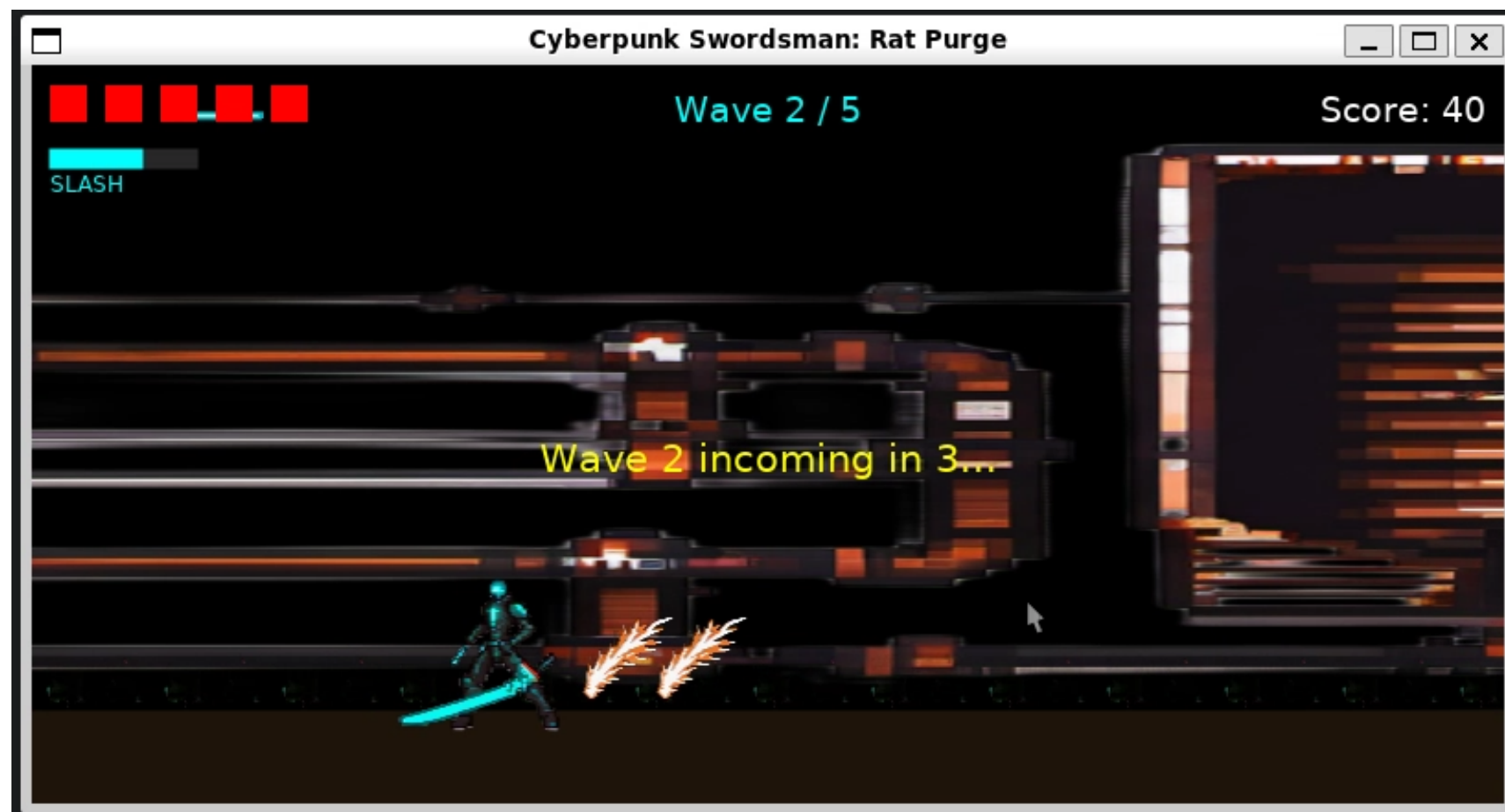


圖二、本專案系統架構圖

表一、使用模型

功能用途	模型名稱
主要內容生成	claude-sonnet-4-6
提示詞壓縮	google/gemma-4-e4b
基礎圖像構建	stable-diffusion-xl-
風格效果強化	pixel-art-xl

成果



圖五、本專案生成遊戲



圖六、ChatDev生成遊戲

結論

表二、本專案與 ChatDev 單輪生成數據比較

Game Name	Token(total)			Time(second)			Lines of Code		
	chatdev	nocode	chatdev/nocode	chatdev	nocode	chatdev/nocode	chatdev	nocode	chatdev/nocode
Cyberpunk Swordsman	737768	78781	9.36	2220	572	3.88	1600	1131	1.41
8 ball pool game	131794	126892	1.04	682	903	0.76	796	1444	0.55
a tank game	204897	92783	2.21	940	581	1.62	1101	982	1.12
Space Shooter Game	111413	105757	1.05	661	580	1.14	804	1212	0.66
Plants vs Zombies	106338	115277	0.92	823	776	1.06	935	1421	0.66
Average	138610.5	110177.25	1.26	776.5	710	1.09	909	1264.75	0.72
Average Efficiency Compared to ChatDev(%)	-20.51%			-8.56%			39.14%		

- 與 ChatDev 相比：
- 減少 20.51 % token 使用量
- 減少 8.56% 生成時間
- 提升 39.14% 的生成程式碼行數
- 因使用 Diffusion 模型，可減少程式碼中繪製素材程式碼比例

未來展望

- Diffusion 生成圖片品質可加強
- 生成遊戲的邏輯稍嫌簡單，需使用更多輪生成逐步加強
- 強化遊戲修復和自動測試腳本的品質

感謝聆聽