บทที่ 1 — นี่ไง เว็บแอปของคุณ!



ในบทแรกนี้ เราจะมาเริ่มสร้างเว็บแอปของเรากันเลยครับ ไม่รออะไรแล้ว!

อย่างที่ผมเกริ่นไปในบทนำ ตลอดหนังสือเล่มนี้เราจะใช้เว็บแอปบันทึกการเงินส่วนตัวชื่อ **ละมุนเงิน** เป็นตัวอย่างในการลงมือทำครับ เป็นเว็บแอปที่ช่วยจัดการรายรับรายจ่าย เงินเก็บ และเงินลงทุน พร้อมสรุปสถานะการเงินผ่านรายงานและกราฟบนหน้าเว็บตามที่เราต้องการ

ภาพที่คุณเห็นอยู่ด้านบนก็คือเว็บแอปที่เรากำลังจะสร้างกันนั่นเองครับ

แต่เราไม่ได้จะสร้างทั้งหมดให้เสร็จภายในบทเดียวนะครับ เพราะผมแบ่งงานออกเป็นส่วน ๆ เพื่อให้เราเข้าใจว่าแต่ละส่วนทำหน้าที่อะไร และค่อย ๆ ประกอบเข้าด้วยกันจนกลายเป็นเว็บแอปที่ใช้งานได้จริง

โดยในบทนี้ เราจะเริ่มจากการสร้าง Web App Mockup ซึ่งเป็นจิ๊กซอว์ชิ้นแรกของเราครับ

Web App Mockup คืออะไร?

Web App Mockup ก็คือ **ต้นแบบหน้าตาของเว็บแอปที่เราต้องการสร้าง** ครับ

ก่อนที่เราจะลงมือพัฒนาเว็บแอปให้ใช้งานได้จริง เราจะให้ AI ช่วยออกแบบหน้าตาและส่วนประกอบต่าง ๆ ขึ้นมาก่อน ไม่ว่าจะเป็นหน้าเข้าสู่ระบบ หน้าหลักแสดงข้อมูล ปุ่ม เมนู ฟอร์ม หรือกราฟต่าง ๆ

แต่ Mockup ของเราจะไม่ได้มีไว้แค่ดูหน้าตาสวย ๆ นะครับ

เราจะสร้างให้เปิดดูและกดสำรวจส่วนต่าง ๆ ได้ด้วย เช่น กดเข้าสู่หน้าหลัก เปิดรายการข้อมูล หรือเปิดฟอร์มเพิ่มและแก้ไขรายการ เพื่อให้เราตรวจสอบได้ว่า **AI สร้างส่วนประกอบต่าง ๆ ครบตามที่ต้องการแล้วหรือยัง**

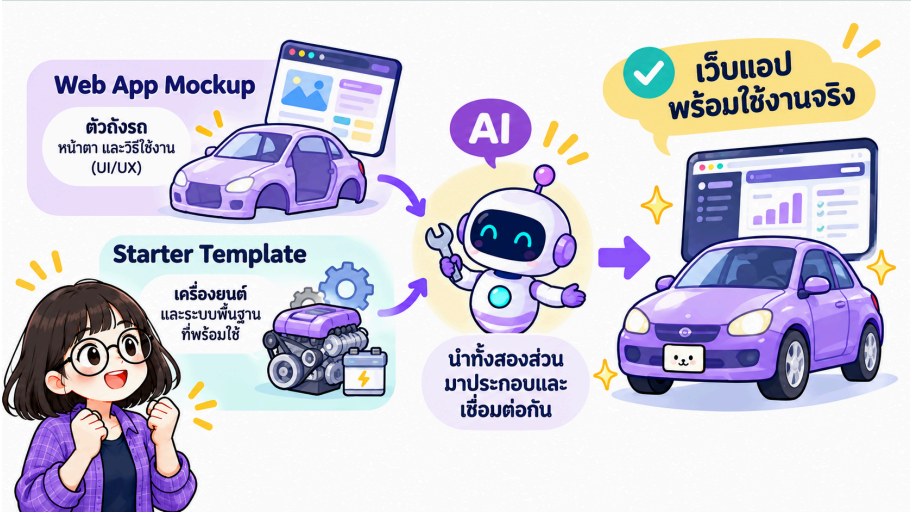
เพียงแต่ตอนนี้เราจะยังใช้ข้อมูลตัวอย่างอยู่ครับ ปุ่มและฟอร์มต่าง ๆ ยังไม่ต้องบันทึกหรือเปลี่ยนข้อมูลจริง และยังไม่มีการเชื่อมต่อฐานข้อมูลหรือระบบบัญชีผู้ใช้จริง

แล้วทำไมเราไม่ให้ AI สร้างเว็บแอปที่ใช้งานได้จริงไปเลยตั้งแต่แรก?

ลองนึกถึงการประกอบรถยนต์ดูครับ ถ้าเรามีเครื่องยนต์ที่พร้อมใช้งานอยู่แล้ว เราก็ไม่จำเป็นต้องเริ่มสร้างเครื่องยนต์ใหม่ทุกครั้งที่อยากได้รถอีกคัน

ในหนังสือเล่มนี้ ผมจึงแบ่งการสร้างเว็บแอปออกเป็นสองส่วนหลัก ๆ คือ **Web App Mockup ที่เป็นส่วนหน้าตาและวิธีใช้งาน** กับ **Starter Template ซึ่งเป็นชุดไฟล์แม่แบบที่เตรียมโครงสร้างและระบบพื้นฐานของเว็บแอปไว้แล้ว** โดยผมมี Starter Template แจกให้พร้อมใช้งาน หรือคุณจะให้ AI ช่วยสร้างเองตามขั้นตอนในบทที่ 3 ก็ได้ครับ

ถ้าเปรียบเป็นรถยนต์ Starter Template ก็คือเครื่องยนต์ที่ผมเตรียมไว้ให้แล้ว ส่วน Web App Mockup ก็คือตัวถัง หน้าตา และการออกแบบวิธีใช้งานของรถคันนั้นครับ



แทนที่จะให้ AI สร้างทั้งเครื่องยนต์และตัวรถไปพร้อมกัน ซึ่งเป็นงานใหญ่และมีหลายส่วนที่ต้องจัดการ หากเกิดปัญหาขึ้นมาก็อาจตรวจสอบได้ยากว่าเกิดจากส่วนไหน **ผมจึงเลือกแยกงานออกจากกันให้ชัดเจนตั้งแต่ต้นครับ**

สร้างหน้าตาเว็บแอปเสร็จเมื่อไหร่ ก็ยกเครื่องยนต์ Starter Template มาใส่ แล้วให้ AI ช่วยเชื่อมระบบเข้าด้วยกัน เท่านี้ก็พร้อมซิ่งแล้วครับ!

ในบทนี้ เราจะมุ่งไปที่การออกแบบหน้าตาและวิธีใช้งาน (UI/UX) ให้ได้ตามที่ต้องการก่อน โดยยังไม่ต้องทำอะไรกับระบบการทำงานเบื้องหลังเลย

นี่เป็นหนึ่งในขั้นตอนสำคัญตามแนวทางการสร้างเว็บแอปที่ผมสรุปไว้ในหัวข้อ **“ภาพรวมการสร้างเว็บแอปของเราทั้งเล่ม”** ในบทนำครับ

เราจะค่อย ๆ ประกอบแต่ละส่วนเข้าด้วยกัน โดยมี AI เป็นคนลงมือสร้างและแก้ไข ส่วนเราเป็นคนกำหนดความต้องการ ตรวจสอบผลงาน และตัดสินใจว่าจะให้ AI ทำอะไรต่อ

จบบทนี้ เราจะได้อะไร?

เมื่อทำตามบทนี้จนจบ คุณจะมี **Web App Mockup ที่ออกแบบร่วมกับ AI ตามความต้องการของตัวเอง** สามารถเปิดดู กดสำรวจหน้าจอและฟอร์มต่าง ๆ ได้ พร้อมชุดไฟล์ ZIP สำหรับนำไปพัฒนาต่อในบทถัดไป

และไม่จำเป็นต้องสร้างเว็บแอปบันทึกการเงินเหมือนตัวอย่างของผมก็ได้นะครับ

ถ้าคุณอยากสร้างเว็บแอปจดบันทึก จัดการสินค้า หรือติดตามงาน ก็เปลี่ยนคำอธิบายให้เป็นสิ่งที่คุณต้องการได้เลย เราจะใช้หลักการสร้าง Mockup แบบเดียวกัน แล้วลองทำไปพร้อมกันเลยก็ได้ครับ!

แต่ก่อนจะลงมือ ผมขอเล่าถึงวิธีที่เราจะใช้สร้าง Mockup กันสักนิดครับ

จากผู้เขียน — บทนี้เราจะไปกันแบบจืด ๆ หน่อยนะครับ :D

ทุกวันนี้มีวิธีสร้างหน้าเว็บด้วย AI ให้เลือกเยอะมาก(ก.ไก่ยาวๆ) เลยครับ ทั้งเครื่องมือ AI ที่ออกแบบมาเพื่อสร้างหน้าเว็บโดยเฉพาะ ชุด Prompt สำเร็จรูปที่มีคนแจกให้ใช้ หรือแม้แต่การเอาภาพหน้าเว็บที่ชอบมาให้ AI ช่วยสร้างตาม แต่ละวิธีก็มีลูกเล่นและข้อดีของตัวเอง บางวิธีสร้างหน้าเว็บออกมาได้สวยมาก แถมยังรวดเร็วอีกด้วย

แต่ในบทนี้ ผมตั้งใจจะพาคุณทำด้วยวิธีพื้นฐานที่สุดครับ

เราจะแค่เปิด ChatGPT แล้วพิมพ์บอกตรง ๆ ว่าอยากได้เว็บแอปอะไร ไม่ใช้เครื่องมือเสริม ไม่ต้องไปหาชุด Prompt เทพ ๆ ที่ไหน

เพราะผมอยากให้คุณเข้าใจ **Concept ของงานสร้างหน้าเว็บก่อน** ว่าเรากำลังทำอะไร มีองค์ประกอบสำคัญอะไรบ้าง ต้องการผลลัพธ์แบบไหน และจะตรวจงานที่ AI สร้างให้ได้อย่างไร โดยไม่ให้ลูกเล่นของเครื่องมือมารบกวนงานจริง ๆ ที่เรากำลังทำอยู่

เมื่อเข้าใจ Concept แล้ว คุณจะเลือกใช้วิธีอื่น ๆ ได้อีกมากมายครับ ซึ่งผมจะเตรียมบทเสริม **“ขยายจักรวาลเครื่องมือสร้างหน้าเว็บ”** เอาไว้พาไปสำรวจโลกของการสร้างหน้าเว็บด้วย AI ว่ายังมีแนวทางและเครื่องมืออะไรน่าสนใจอีกบ้าง ในปัจจุบัน

เอาละครับ เก็บลูกเล่นเหล่านั้นไว้ก่อน มาดูขั้นตอนที่เราจะใช้ทำงานกับ AI กันครับ

Concept — สั่งงาน ตรวจงาน แล้วรับผลงาน



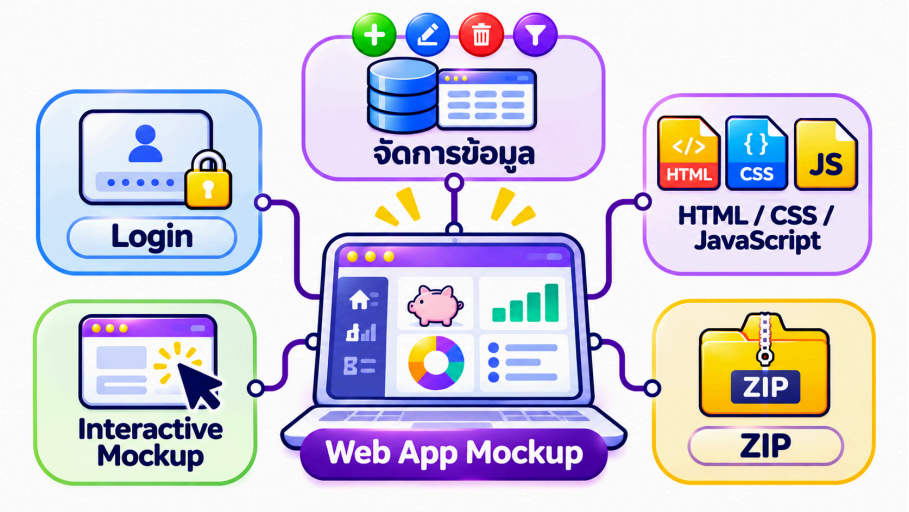
สั่งงาน → ตรวจ / ปรับ / แก้ → รับงาน (ZIP)

นี่คือวิธีทำงานหลักของเราครับ เริ่มจากบอก AI ว่าต้องการเว็บแอปแบบไหน แล้วเปิดดูผลงานจริง ตรวจว่ามีส่วนประกอบครบหรือไม่ หากยังไม่ตรงใจก็บอกจุดที่ต้องปรับให้ AI แก้ไข จากนั้นตรวจซ้ำจนพอใจ แล้วรับชุดไฟล์ ZIP เก็บไว้พัฒนาต่อ

สิ่งสำคัญไม่ใช่แค่การสั่งให้ AI สร้างหน้าเว็บสวย ๆ ออกมา แต่เป็นการรู้ว่า **เราต้องตรวจอะไร และจะบอก AI ให้แก้ตรงไหน** ครับ

ก่อนสั่งงานจริง ผมอยากพาคุณรู้จักส่วนประกอบสำคัญ 5 อย่างที่เราจะเตรียมไว้ใน Mockup กันก่อน

องค์ประกอบหลักทั้ง 5 ของ Web App Mockup



เราเข้าใจแล้วว่า Mockup ของเราคืออะไร คราวนี้มาดูกันครับว่าต้องเตรียมส่วนประกอบอะไรไว้บ้าง เพื่อให้ตรวจงานได้ครบและพร้อมนำไปพัฒนาต่อในบทถัด ๆ ไป

1. Login — หน้าเข้าสู่ระบบ

ในบทนี้ เราจะเตรียมหน้า Login จำลองไว้ก่อนครับ ยังไม่ต้องใช้บัญชีจริงหรือเชื่อมระบบเข้าสู่ระบบ แต่ผมอยากให้มีหน้าจอนี้อยู่ใน Mockup ตั้งแต่ต้น เพราะเราจะนำไปพัฒนาเป็นระบบ Login จริงในบทต่อ ๆ ไป

ผมต้องการให้เว็บแอปของเรามีแค่ผมที่เข้าถึงข้อมูลของตัวเองได้ครับ ไม่ว่าจะเปิดจากคอม มือถือ หรือ iPad ก็แค่ Login ด้วยบัญชีของผมเอง ข้อมูลทั้งหมดก็จะเป็นข้อมูลชุดเดียวกัน และคนอื่นจะเข้ามาดูหรือแก้ไขข้อมูลของผมไม่ได้

ที่สำคัญ เราจะทำให้ Login ค้างไว้ได้ด้วยครับ หากยัง Login ค้างไว้อยู่ เว็บของเราจะตรวจสอบและเปิดเข้าใช้งานอัตโนมัติ ไม่ต้องห่วงว่าจะต้องมา Login ใหม่ทุกครั้ง และถ้าต้องการหยุดใช้งานบนอุปกรณ์ไหน ก็แค่ Sign out ออกจากอุปกรณ์นั้น ก็เรียบร้อยครับ

นี่เป็นข้อดีของ Web App ที่ผมชอบครับ เราสามารถเปิดใช้งานจากอุปกรณ์ต่าง ๆ ได้โดยไม่ต้องติดตั้งแอปแยกในแต่ละเครื่อง

ข้อดีขึ้นไปอีกก็คือ

- จะแชร์เว็บให้เพื่อนใช้งานด้วยก็ได้ครับ เมื่อเราเชื่อมต่อระบบบัญชีเสร็จแล้ว บัญชีใครบัญชีมัน ข้อมูลแยกกัน
- สามารถพัฒนาเป็น Web App ที่เปิดบริการให้บุคคลภายนอกสมัครสมาชิกใช้งานได้ในอนาคตเลยครับ! (ยังอยู่นอกเหนือขอบเขตของเนื้อหาหลักในเล่มนี้ แต่อยู่ในแผนเนื้อหาเสริมที่ผมจะเขียนอัปเดตให้ฟรี สำหรับทุกคนที่ซื้อหนังสือเล่มนี้ครับ)

การเริ่มพัฒนาระบบบัญชี Login ไว้ตั้งแต่ต้น จึงเป็นข้อดีสุด ๆ ที่ผมทำเองมาตลอด และอยากให้คุณทำด้วยครับ โดยไม่ได้เหนื่อยอะไรเพิ่มเลย เพราะเดี๋ยว AI จะทำงานทั้งหมดนี้ให้เราเอง

ส่วนระบบ Login ที่ใช้งานได้จริงและการป้องกันไม่ให้คนอื่นเข้าถึงข้อมูล เราจะทำกันในบทต่อ ๆ ไปครับ

2. HTML / CSS / JavaScript — โครงสร้างพื้นฐานของเว็บแอป

ผมเลือกให้เว็บแอปของเราสร้างจากไฟล์พื้นฐาน 3 ชนิดที่ทำงานร่วมกันครับ ได้แก่ HTML, CSS และ JavaScript

เนื้อหาทางเทคนิคหน่อยนะครับ

- **HTML** — ใช้สร้างโครงสร้างและเนื้อหาของหน้าเว็บ
- **CSS** — ใช้จัดหน้าตา สีสัน และการแสดงผล

- **JavaScript** — ใช้ควบคุมการทำงานของปุ่ม ฟอร์ม และการโต้ตอบต่าง ๆ บนหน้าเว็บ

จริง ๆ แล้วทั้งสามอย่างนี้เป็นเทคโนโลยีพื้นฐานของการสร้างเว็บไซต์ครับ ไม่ใช่แค่ชื่อชนิดไฟล์ แต่เราไม่จำเป็นต้องลงลึกไปกว่านี้ในตอนนี้

สิ่งที่ผมต้องการคือให้ AI สร้างหน้าเว็บโดยใช้ HTML, CSS และ JavaScript เป็นหลัก ไม่ต้องเพิ่ม Framework หรือเครื่องมือสร้างหน้าเว็บอื่นเข้ามา เพราะแค่สามอย่างนี้ก็เพียงพอสำหรับสร้างหน้าเว็บตามแนวทางในหนังสือเล่มนี้แล้วครับ

และคุณไม่จำเป็นต้องเข้าใจหรืออ่านโค้ดทั้งสามอย่างนี้เป็นด้วยนะครับ แค่บอก AI Programmer ของเราว่าต้องการใช้เทคโนโลยีอะไร ก็ถือว่าเรากำหนดแนวทางการทำงานให้ AI ได้แล้ว ส่วนการเขียนไฟล์และประกอบให้ทำงานร่วมกัน ปล่อยให้ AI จัดการได้เลยครับ

3. Interactive Mockup — เปิดดูและตรวจงานได้

เราเกริ่นเรื่องการกดสำรวจ Mockup กันไปแล้ว ทีนี้ลองดูว่าทำไมการเปิดตรวจหน้าจอแต่ละส่วนถึงสำคัญครับ

ยกตัวอย่างเช่น เราสั่งให้มีหน้าแก้ไขข้อมูล แต่หน้าแก้ไขจะปรากฏก็ต่อเมื่อเรากดเข้าไปที่รายการข้อมูลก่อน **ถ้ากดเข้าไปดูไม่ได้ เราจะรู้ได้อย่างไรว่า AI สร้างหน้าแก้ไขข้อมูลให้เราจริง ๆ แล้ว?**

เพราะฉะนั้น เราจะให้ AI ทำ Mockup ที่กดปุ่มเพื่อเปิดหน้าต่าง ๆ และฟอร์มได้ เพื่อให้เราตรวจงานได้ครบทุกส่วนครับ ส่วนการทำให้ฟังก์ชันเหล่านี้จัดการข้อมูลจริงได้ เราจะไปทำกันในบทต่อ ๆ ไป

4. ฟังก์ชันจัดการข้อมูล — เพิ่ม แก้ไข ลบ และกรองรายการ

นี่เป็นอีกหนึ่ง "หัวใจสำคัญ" ของเว็บแอปที่เราจะสร้างในหนังสือเล่มนี้ครับ

ไม่ว่าคุณจะสร้างเว็บแอปเพื่อทำงานอะไรก็ตาม การจัดการข้อมูลในเว็บแอปต้องสามารถ

- **เพิ่มข้อมูลใหม่**
- **ดูข้อมูลที่มีอยู่**
- **แก้ไขข้อมูลเดิม**
- **ลบข้อมูลที่ไม่ต้องการ**
- **กรองหรือแสดงผลข้อมูลตามที่ต้องการได้**

ฟังก์ชันเหล่านี้เป็นพื้นฐานที่ครอบคลุมงานจัดการข้อมูลพื้นฐานได้ครบถ้วน และสามารถนำไปประยุกต์สร้างเว็บแอปได้อีกสารพัดประเภทครับ

เพราะฉะนั้น **เราจะเตรียมปุ่ม ฟอร์ม และหน้าจอสำหรับฟังก์ชันเหล่านี้ไว้ตั้งแต่ Web App Mockup เลย** แม้ตอนนี้ยังไม่ต้องจัดการข้อมูลจริง แต่เมื่อถึงขั้นตอนเชื่อมฐานข้อมูล เราจะมีส่วนประกอบพร้อมให้ AI นำไปทำให้ใช้งานได้จริง โดยไม่ต้องกลับมาเริ่มออกแบบกันใหม่

5. ZIP — ชุดไฟล์พร้อมนำไปพัฒนาต่อ

สุดท้าย **เราต้องได้ไฟล์ผลงานกลับมาเป็นชุดครับ ไม่ใช่แค่หน้าเว็บที่เปิดดูได้ในแชต** ผมจึงกำหนดให้ AI จัดไฟล์ทั้งหมดเป็น ZIP เพื่อให้เราดาวน์โหลด เก็บสำรอง และนำไปเปิดทำงานต่อบนคอมพิวเตอร์ของตัวเองได้

ไฟล์ ZIP นี้จะเป็น **จิ๊กซอว์ชิ้นแรกของเว็บแอปเรา** ภายในมีไฟล์หน้าเว็บและส่วนประกอบที่เตรียมไว้ตามสูตรทั้งหมด เมื่อถึงบทต่อๆไป เราจะนำชุดไฟล์นี้ไปประกอบกับ Starter Template แล้วค่อยเชื่อมระบบต่าง ๆ ให้กลายเป็นเว็บแอปที่ใช้งานได้จริงครับ

ตอนนี้เรารู้แล้วครับว่าต้องเตรียมอะไรไว้ใน Mockup บ้าง

ต่อไปเราจะลองสั่ง AI ให้สร้างทั้งหมดนี้ออกมาจริง ๆ ด้วย Prompt เดียวไปเลยครับ!

Prompt แรก: สร้างหน้าตาเว็บแอป

คัดลอกข้อความด้านล่างใส่ในแชต แล้วกดส่งได้เลยครับ **Prompt คือข้อความที่เราพิมพ์บอก AI ว่าอยากให้ช่วยทำอะไร** นั่นเอง

PROMPT · คัดลอกไปใช้

ผมต้องการ Web App Mockup

การใช้งาน:
แอปช่วยบันทึกรายรับรายจ่าย รวมถึงเงินเก็บและเงินลงทุน แยกหมวดหมู่ได้ มีรายงานเป็นกราฟ แยกรายการที่เกิดขึ้นจริงกับรายการประมาณการ เพื่อดูสถานะการเงินตอนนี้และคาดการณ์เงินเหลือสิ้นเดือน
แยกเงินที่แบ่งไปเก็บหรือลงทุนออกจากรายจ่าย เพื่อไม่ให้นับซ้ำเป็นค่าใช้จ่าย
ใช้ภาษาไทยและสกุลเงินบาท

หน้าตา:
สวย ทันสมัย อ่านง่าย บรรยากาศเป็นมิตร ไม่ดูเหมือนโปรแกรมบัญชีที่ซับซ้อน

ข้อกำหนด:
- สร้างด้วย HTML, CSS และ JavaScript ใช้ข้อมูลตัวอย่าง ยังไม่เชื่อมฐานข้อมูลหรือระบบบัญชีผู้ใช้จริง
- รองรับคอมพิวเตอร์และมือถือ
- มีหน้าเข้าสู่ระบบจำลอง พร้อมช่องอีเมล รหัสผ่าน และปุ่มเข้าสู่ระบบ ให้เริ่มที่หน้านี้และกดเข้าหน้าหลักได้โดยไม่ใช้บัญชีจริง
- เตรียมปุ่มและฟอร์มสำหรับเพิ่ม ลบ แก้ไข และกรองรายการ
- กดสำรวจหน้าเว็บ เปิดฟอร์ม และปิดหน้าต่างได้ แต่ยังไม่ต้องบันทึกหรือเปลี่ยนข้อมูลจริง
- ให้ตัวเลขสรุปและกราฟสอดคล้องกับข้อมูลตัวอย่าง
- ช่วยออกแบบหน้าจอและฟังก์ชันที่เหมาะสมจากคำอธิบายนี้ให้ด้วย

ผลลัพธ์ที่ต้องการ:
แสดงตัวอย่างหน้าเว็บแบบโต้ตอบในแชตถ้ารองรับ และจัดชุดไฟล์ให้ดาวน์โหลดไปเปิดดูได้ พร้อมบอกว่าต้องเปิดไฟล์ใดก่อน

Prompt แรก: ส่วนไหนแก้ได้ ส่วนไหนควรเก็บไว้

Prompt แรกของเราประกอบด้วยเนื้อหาสองกลุ่มใหญ่ครับ

เรื่องแรกคือ **เว็บแอปที่เราอยากได้** เช่น เว็บแอปใช้ทำอะไร มีฟังก์ชันอะไร และอยากได้หน้าตาแบบไหน ส่วนนี้เป็นพื้นที่ของคุณเต็มที่ หากต้องการสร้างเว็บแอปชนิดอื่นก็เปลี่ยนข้อความได้เลย

อีกเรื่องคือ **วิธีสร้าง Mockup ในบทนี้** เช่น ให้สร้างด้วย HTML, CSS และ JavaScript ใช้ข้อมูลตัวอย่าง และยังไม่เชื่อมฐานข้อมูลจริง ข้อความส่วนนี้ควรเก็บไว้ก่อน เพื่อให้ผลลัพธ์อยู่ในรูปแบบที่เรานำไปใช้ต่อในบทถัดไปได้ครับ

ผมจะแบ่งเนื้อหาสองกลุ่มนี้ออกเป็นสามส่วนให้เห็นชัดเจนดังนี้

ส่วนที่ 1 — เขียนตามเว็บแอปที่คุณต้องการ

ส่วนนี้บอกว่าเว็บแอปใช้ทำอะไรและต้องการหน้าตาแบบไหน คุณสามารถเขียนใหม่ทั้งหมดได้ครับ

PROMPT · คัดลอกไปใช้

ผมต้องการ Web App Mockup

การใช้งาน:
แอปช่วยบันทึกรายรับรายจ่าย รวมถึงเงินเก็บและเงินลงทุน แยกหมวดหมู่ได้ มีรายงานเป็นกราฟ แยกรายการที่เกิดขึ้นจริงกับรายการประมาณการ เพื่อดูสถานะการเงินตอนนี้และคาดการณ์เงินเหลือสิ้นเดือน

แยกเงินที่แบ่งไปเก็บหรือลงทุนออกจากรายจ่าย เพื่อไม่ให้นับซ้ำเป็นค่าใช้จ่าย

ใช้ภาษาไทยและใช้สกุลเงินบาท

หน้าตา:
สวย ทันสมัย อ่านง่าย บรรยากาศเป็นมิตร ไม่ดูเหมือนโปรแกรมบัญชีที่ซับซ้อน

ถ้าต้องการสร้างเว็บแอปอื่น ให้เปลี่ยนข้อความตั้งแต่ **การใช้งาน:** จนถึง **หน้าตา:** เช่น เปลี่ยนเป็นเว็บแอปจดบันทึก เว็บแอปจัดการสินค้า หรือเว็บแอปติดตามงานได้เลย

ไม่ต้องใช้ศัพท์เทคนิคครับ บอก AI ด้วยภาษาปกติว่าอยากใช้เว็บแอปทำอะไร อยากเห็นข้อมูลอะไร และอยากให้บรรยากาศของเว็บแอปเป็นแบบไหน

ส่วนที่ 2 — เลือกฟังก์ชันที่ต้องการ

ส่วนนี้กำหนดหน้าจอและสิ่งที่อยากลองกดดูใน Mockup สามารถเพิ่ม ลด หรือปรับให้เหมาะกับเว็บแอปของคุณได้ครับ

> **PROMPT · คัดลอกไปใช้**
>
> ฟังก์ชันและหน้าจอที่ต้องการ:
> - มีหน้าเข้าสู่ระบบจำลอง พร้อมช่องอีเมล รหัสผ่าน และปุ่มเข้าสู่ระบบ ให้เริ่มที่หน้านี้และกดเข้าหน้าหลักได้โดยไม่ใช้บัญชีจริง
> - เตรียมปุ่มและฟอร์มสำหรับเพิ่ม ลบ แก้ไข และกรองรายการ
> - ให้ตัวเลขสรุปและกราฟสอดคล้องกับข้อมูลตัวอย่าง
> - ช่วยออกแบบหน้าจอและฟังก์ชันอื่นที่เหมาะกับการใช้งานข้างต้นให้ด้วย

หากกำลังทำตามตัวอย่าง **ละมุนเงิน** ให้ใช้รายการนี้ได้เลยครับ ส่วนใครสร้างเว็บแอปชนิดอื่นสามารถเปลี่ยนชื่อข้อมูล ฟอร์ม และส่วนแสดงผลให้ตรงกับงานได้

อย่างไรก็ตาม ผมแนะนำให้เก็บการ **เพิ่ม แก้ไข ลบ และกรองรายการ** ไว้ก่อน เพราะเป็นฟังก์ชันพื้นฐานที่นำไปประยุกต์กับเว็บแอปได้หลายชนิด และเราจะใช้ฝึกทำงานกับ AI ในบทต่อ ๆ ไปครับ

ส่วนที่ 3 — เก็บไว้ตามเดิม

ส่วนนี้กำหนดว่า AI ต้องสร้าง Mockup ออกมาในรูปแบบใด เพื่อให้เราเปิดดูดาวน์โหลด และนำไปประกอบกับ Starter Template ภายหลังได้ จึงควรคงไว้ตามหนังสือก่อนครับ

> **PROMPT · คัดลอกไปใช้**
>
> ข้อกำหนดการสร้าง:
> - สร้างด้วย HTML, CSS และ JavaScript
> - ใช้ข้อมูลตัวอย่าง ยังไม่เชื่อมฐานข้อมูลหรือระบบบัญชีผู้ใช้จริง
> - รองรับทั้งคอมพิวเตอร์และมือถือ
> - กดสำรวจหน้าเว็บ เปิดฟอร์ม สลับหน้า และปิดหน้าต่างได้
> - ปุ่มต่าง ๆ ใช้สำหรับสำรวจหน้าตาของ Mockup แต่ยังไม่ต้องบันทึกหรือเปลี่ยนข้อมูลจริง
>
> ผลลัพธ์ที่ต้องการ:
> - แสดงตัวอย่างหน้าเว็บแบบโต้ตอบในแชต หากระบบรองรับ
> - แนบชุดไฟล์ทั้งหมดให้ดาวน์โหลดไปเปิดดูได้
> - บอกว่าต้องเปิดไฟล์ใดก่อน

ส่วนนี้ช่วยกำหนดขอบเขตให้ AI เข้าใจว่า ตอนนี้เราต้องการเพียง **ต้นแบบที่กดสำรวจได้** ยังไม่ได้ต้องการเว็บแอปที่เก็บข้อมูลจริงครับ

จำง่าย ๆ ว่า:

> **PROMPT · คัดลอกไปใช้**
>
> **เปลี่ยน “แอปทำอะไร หน้าตาแบบไหน และมีฟังก์ชันอะไร”**
> **เก็บ “วิธีสร้าง Mockup และรูปแบบไฟล์” ไว้ตามเดิม**

แบบนี้คุณจะนำโครง Prompt ไปสร้างเว็บแอปอื่นได้ โดยยังได้ไฟล์ในรูปแบบเดียวกับที่หนังสือใช้สอนต่อครับ

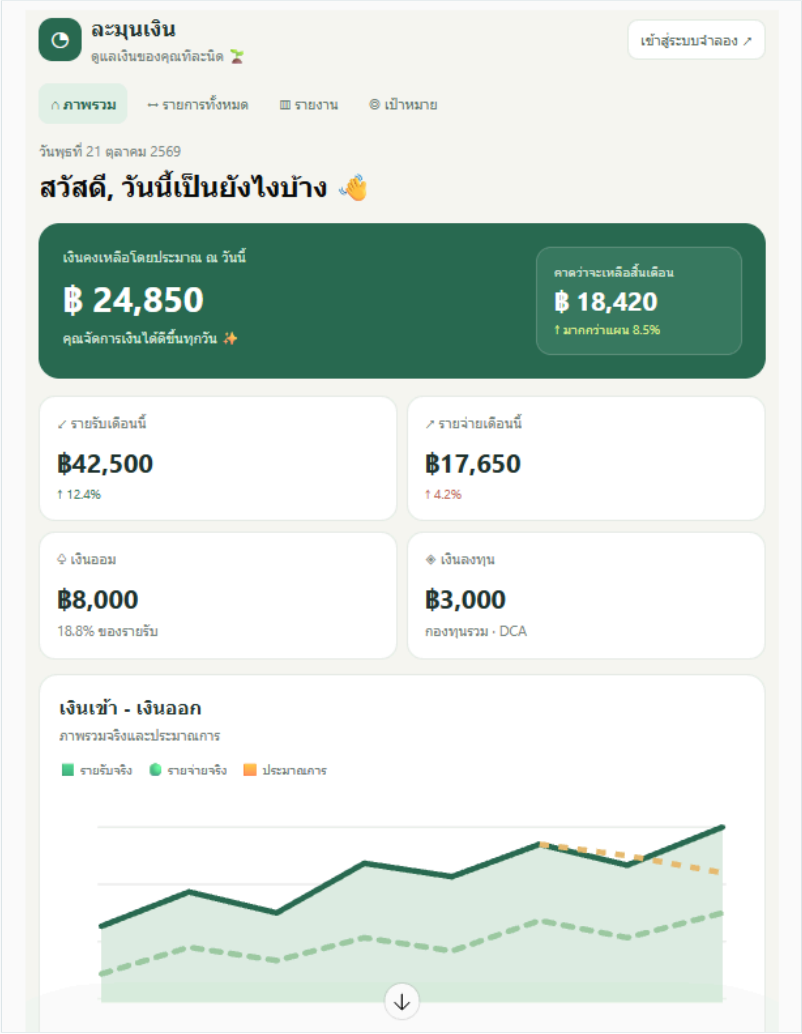
รอสักครู่ แล้วเปิดดูผลงาน

หลังส่งข้อความ ให้รอ ChatGPT สร้างงานจนเสร็จก่อนครับ ในการทดลองด้วยบัญชี ChatGPT Free รอบที่ผมเก็บภาพนี้ ใช้เวลาเกือบ 3 นาที ของคุณอาจเร็วหรือช้ากว่านี้ได้ครับ

ใครตั้งใจจะใช้ ChatGPT Plus ตลอดเล่มอยู่แล้ว ก็เริ่มจากบัญชีนั้นได้เลยครับ สำหรับบทนี้ให้ดูผลลัพธ์ที่ได้และตรวจตามรายการของเราเป็นหลัก



ได้แล้ว! ลองดูว่า ChatGPT ส่งอะไรกลับมาบ้างครับ ในตัวอย่างของผมมีทั้งคำอธิบายและตัวอย่างหน้าเว็บแอป



ตัวอย่างหน้าเว็บแอปหลังสั่งสร้างต้นแบบ

ช่วงท้ายของคำตอบมีไฟล์ให้ดาวน์โหลดด้วยครับ ก่อนเริ่มสำรวจและขอแก้ เรามาจัดที่เก็บ ZIP สำรองไว้ก่อน เผื่อภายหลังมีอะไรผิดพลาดจะได้ย้อนกลับมาดูชุดก่อนหน้าได้

- สร้างโฟลเดอร์สำหรับเก็บ ZIP ของงานนี้ไว้ในที่หาเจอง่าย เช่น `Documents\ละมุนเงิน-ไฟล์สำรอง`
- ดาวน์โหลด ZIP ชุดแรก แล้วบันทึกไว้ในโฟลเดอร์นี้ ตั้งชื่อให้รู้ว่าเป็นชุดเริ่มต้น เช่น `ละมุนเงิน-01-ต้นแบบแรก.zip`
- เมื่อมี ZIP ชุดใหม่ ให้เก็บไว้ในโฟลเดอร์เดียวกัน ตั้งเลขต่อไป เช่น `ละมุนเงิน-02-ส่วนประกอบครบ.zip` และอย่าเขียนทับชุดก่อนหน้า

ใช้เลขสองหลัก เช่น `01`, `02` แทน `1`, `2` นะครับ เพราะถ้ามีถึง 10 ชุด การเรียงตามชื่อไฟล์ก็ยังเรียงลำดับได้ถูกต้อง และมองออกว่าชุดไหนใหม่กว่า ตัวเลขในตัวอย่างต่อจากนี้เป็นเพียงแนวทาง ให้ใช้เลขถัดไปที่ยังไม่เคยใช้ทุกครั้งที่ได้ ZIP ฉบับใหม่ หากมีการแก้หลายรอบก็เพิ่มเลขใหม่ทุกครั้งครับ

> **หน้าจอของคุณอาจไม่เหมือนภาพในหนังสือครับ** แต่หลัง AI ทำงานเสร็จ เราควรได้รับของสองอย่างตามที่ขอไว้ใน Prompt ก่อนหน้า:

1. **หน้าเว็บที่กดสำรวจได้** — อาจเปิดให้ลองใช้ในแชตได้เลย หรือเป็นลิงก์ที่กดเปิดหน้าเว็บก็ได้
2. **ชุดไฟล์สำหรับดาวน์โหลด** — พร้อมคำแนะนำว่าต้องเปิดไฟล์ใดก่อน

บางครั้ง AI อาจส่งมาให้ไม่ครบ แม้เราจะระบุไว้ใน Prompt แล้วก็ตาม ให้ตรวจดูคำตอบก่อนครับ ถ้าขาดอย่างใดอย่างหนึ่ง ให้พิมพ์ขอต่อในแชตเดิมได้เลย

ถ้ายังไม่มีหน้าเว็บให้กดสำรวจและไม่มีลิงก์สำหรับเปิดดู พิมพ์ว่า:

> **PROMPT · คัดลอกไปใช้**
>
> ช่วยเปิดตัวอย่างหน้าเว็บที่กดสำรวจได้ให้ดูในแชตด้วยครับ หากระบบไม่รองรับ ช่วยสร้างลิงก์ที่กดเปิดหน้าเว็บเพื่อทดลองใช้งานได้แทน

ถ้ายังไม่มีชุดไฟล์สำหรับดาวน์โหลด พิมพ์ว่า:

> **PROMPT · คัดลอกไปใช้**
>
> ช่วยจัดงานนี้เป็นชุดไฟล์ทั้งหมดสำหรับดาวน์โหลดไปเปิดดูได้ และบอกด้วยว่าต้องเปิดไฟล์ใดก่อน

หากยังขาดทั้งสองอย่าง จะรวมคำขอเป็นข้อความเดียวก็ได้ครับ:

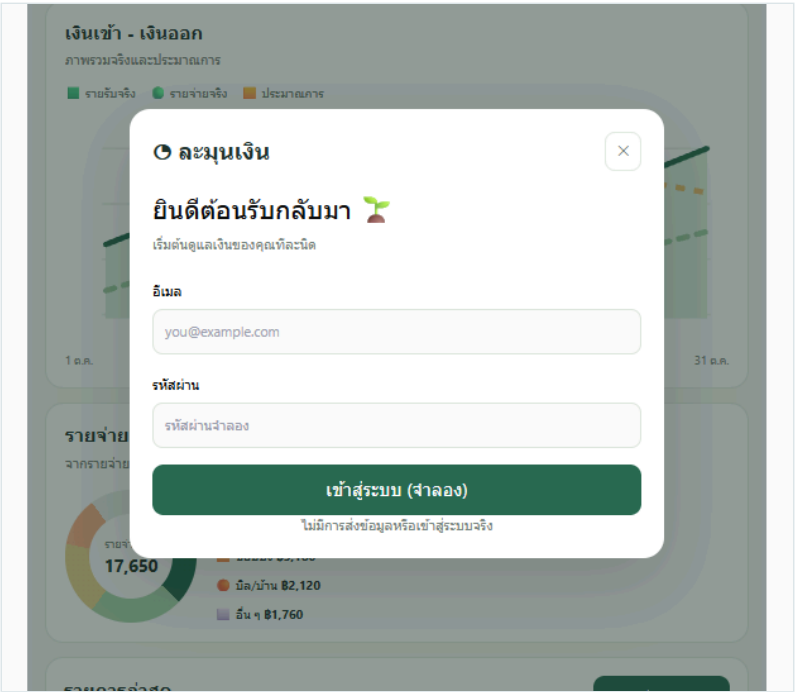
> **PROMPT · คัดลอกไปใช้**
>
> ช่วยเปิดตัวอย่างหน้าเว็บที่กดสำรวจได้ให้ดูในแชต หรือสร้างลิงก์สำหรับเปิดดู พร้อมจัดชุดไฟล์ทั้งหมดให้ดาวน์โหลด และบอกด้วยว่าต้องเปิดไฟล์ใดก่อน

ลองกดสำรวจเว็บแอป

เริ่มจากหน้าเข้าสู่ระบบจำลอง แล้วกดเข้าสู่หน้าหลัก จากนั้นลองปุ่มเพิ่มรายการ ดูว่ามีฟอร์มเปิดขึ้นมาหรือไม่ และปิดกลับได้ไหมครับ

ต่อด้วยปุ่มแก้ไขและปุ่มลบ เป้าหมายตอนนี้คือเดินดูต้นแบบได้ ส่วนปุ่มบันทึกยังไม่ต้องบันทึกหรือเปลี่ยนข้อมูลจริง แค่มีปุ่มอยู่ก็พอครับ

ภาพชุดต่อไปเป็นตัวอย่างระหว่างสำรวจเว็บแอปที่ผมได้มาครับ ลองเทียบกับของคุณว่ามีส่วนประกอบตามที่สั่งให้ AI ทำอยู่ตรงไหนบ้าง ตำแหน่งและหน้าตาอาจต่างกันได้



ตัวอย่างหน้าเข้าสู่ระบบจำลอง

ภาพขยาย 01 · กล่องยืนยันการลบ



ตัวอย่างกล่องยืนยันการลบรายการ

เมื่อกดปุ่มลบ ให้ตรวจว่ากล่องยืนยันเปิดขึ้นมา และกดปุ่มยกเลิกเพื่อกลับไปยังรายการได้

ตอนนี้เราได้หน้าตาเว็บแอปแรกแล้วครับ เร็วจัด!

ทิป · ลองทำดู

เคล็ดลับ: ถ้าหน้าตาชุดแรกยังไม่ถูกใจ และยังไม่ได้เริ่มแก้รายละเอียด ลองเปิดแชตใหม่แล้วใช้ Prompt เดิมอีกสักรอบเพื่อสร้างหน้าเว็บอีกแบบ แล้วเปรียบเทียบจนเจอแบบที่พอใจครับ

PROMPT · คัดลอกไปใช้

แต่เมื่อพอใจกับต้นแบบตัวนี้แล้ว — สำหรับระยะเริ่มต้นยังไม่ต้องสวยเวอร์ แค่รู้สึกว่า “อืม อารมณ์ประมาณนี้แหละ” — ขั้นตอนต่อไปให้สั่งปรับส่วนที่ต้องการในแชตเดิมได้เลยครับ

ตรวจสอบความครบถ้วนก่อนไปต่อ

หน้าตาเว็บแอปของเราอาจต่างกัน แต่เราตรวจจากสิ่งที่ต้องการได้เหมือนกันครับ รอบนี้ยังไม่ต้องไล่ขยับทุกปุ่มหรือปรับทุกกรอบ ขอให้ส่วนประกอบครบและกดสำรวจได้ก่อน

เกณฑ์ของเรามีสองข้อ:

1. **ปุ่มสำหรับเดินดูต้นแบบต้องทำงาน** เช่น เข้าหน้าหลัก เปิดฟอร์ม และปิดหน้าต่าง
2. **ระบบจัดการข้อมูลจริงยังไม่ต้องทำงาน** เช่น กดบันทึกแล้วเพิ่มรายการ ลบข้อมูล หรือกรองผลลัพธ์ พวกนี้ยังใช้ไม่ได้ครับ (เพราะเรายังไม่มีฐานข้อมูลจริง)

ใช้รายการด้านล่างตรวจเว็บแอปของคุณ แล้วติ๊กเมื่อผ่านครับ ถ้าสร้างเว็บแอปชนิดอื่น ให้ปรับรายการตรวจให้ตรงกับสิ่งที่คุณขอไว้

- ☐ มีหน้าเข้าสู่ระบบพร้อมช่องอีเมล รหัสผ่าน และปุ่มที่กดเข้าหน้าหลักได้โดยไม่ใช้บัญชีจริง
- ☐ มีหน้าหลักแสดงรายรับ รายจ่าย เงินเหลือใช้ปัจจุบัน เงินเก็บ/ลงทุน และเงินเหลือใช้คาดการณ์สิ้นเดือน
- ☐ มีกราฟตัวอย่าง และตัวเลขสอดคล้องกับรายการที่แสดง
- ☐ มีรายการตัวอย่างแสดงชื่อ จำนวนเงิน วันที่ ประเภท หมวดหมู่ และสถานะจริง/ประมาณการ
- ☐ แยกเงินที่แบ่งไปเก็บ/ลงทุนออกจากรายจ่าย ไม่ได้นับซ้ำเป็นค่าใช้จ่าย
- ☐ มีปุ่มเพิ่มรายการ เปิดฟอร์มและปิดกลับได้
- ☐ ในฟอร์มมีช่องประเภท ชื่อรายการ จำนวนเงิน วันที่ หมวดหมู่ และสถานะจริง/ประมาณการ
- ☐ มีปุ่มแก้ไข เปิดฟอร์มที่แสดงข้อมูลเดิมและปิดกลับได้
- ☐ มีปุ่มลบ เปิดกล่องยืนยันและกดยกเลิกได้

- ☐ มีตัวเลือกเดือน ประเภทรายการ หมวดหมู่ และสถานะจริง/ประมาณการสำหรับเตรียมกรองรายการ
- ☐ ดาวน์โหลดชุดไฟล์และเปิดดูบนเครื่องได้

> **เรากำลังตรวจต้นแบบครับ ยังไม่ได้ทดสอบระบบบันทึกข้อมูลจริง** ไม่ต้องใส่อีเมล รหัสผ่าน หรือข้อมูลการเงินจริงของตัวเองในขั้นนี้

ในขั้นตอนนี้ ถ้าพบว่าเว็บแอปขาดส่วนไหนไป หรือทำงานไม่เป็นไปตามที่ต้องการ ให้พิมพ์ต่อในแชตเดิมได้เลย ค่อย ๆ เก็บทีละเรื่อง แล้วตรวจส่วนที่แก้พร้อมส่วนเดิมอีกครั้ง

ตัวอย่าง: เปิดมาเจอหน้าหลักเลย ไม่เจอหน้าเข้าสู่ระบบ

> **PROMPT · คัดลอกไปใช้**
>
> ตอนนี้ตัวอย่างเปิดเข้าหน้าหลักเลย ฉันต้องการให้เริ่มที่หน้าเข้าสู่ระบบจำลอง แล้วกดเข้าสู่ระบบเพื่อดูหน้าหลักได้โดยไม่ต้องใช้บัญชีจริง ช่วยปรับทั้งตัวอย่างและชุดไฟล์ดาวน์โหลดให้ตรงกันด้วย

ตัวอย่าง: มีรายการแล้ว แต่ยังไม่มีส่วนแก้ไข

> **PROMPT · คัดลอกไปใช้**
>
> ตอนนี้รายการที่บันทึกไว้ยังไม่มีปุ่มแก้ไข ช่วยเพิ่มปุ่มแก้ไขในแต่ละรายการ เพื่อให้เปิดหน้าจอและแก้รายละเอียดของรายการนั้น ๆ ได้

เมื่อ ChatGPT แก้เสร็จ ลองเปิดฟอร์มดูว่าข้อมูลเดิมแสดงขึ้นมาหรือยัง และกดยกเลิกกลับได้ไหมครับ

ถ้าแก้แล้วของเดิมหายล่ะ?

บอกอาการที่เห็นให้ชัด เช่น:

> **PROMPT · คัดลอกไปใช้**
>
> หลังเพิ่มปุ่มแก้ไข ปุ่มเพิ่มรายการหายไป ช่วยนำปุ่มเพิ่มรายการกลับมาให้เปิดฟอร์มได้เหมือนเดิม โดยยังเก็บปุ่มแก้ไขไว้

เราจะวนแบบนี้ทีละเรื่อง:

เจอสิ่งที่ขาด → บอกสิ่งที่ต้องการ → ให้ ChatGPT แก้ → เปิดสำรวจ → ติ๊กผ่าน

ด่านนี้ผ่านแล้วครับ! เรามีต้นแบบที่ส่วนประกอบหลักครบ และรู้วิธีตรวจสิ่งที่ AI สร้างให้แล้ว ต่อไปเราจะลองปรับหน้าตาให้ถูกใจขึ้นครับ



บันทึก ZIP สำรอง: เมื่อแก้ส่วนประกอบที่ขาดหรือเรียกคืนของเดิมจน AI ส่ง ZIP ฉบับใหม่มา ให้ดาวน์โหลดเก็บในโฟลเดอร์สำรองที่สร้างไว้ โดยเพิ่มเลขลำดับต่อจากชุดก่อน เช่น `ละมุนเงิน-02-ส่วนประกอบครบ.zip` เก็บชุดเดิมไว้ด้วย เผื่อชุดใหม่มีอะไรผิดพลาดจะได้กลับไปใช้ชุดก่อนหน้าครับ

เปลี่ยน Prompt ให้เป็นเว็บแอปของคุณ

ถ้าอยากสร้างเว็บแอปอื่น ให้เปลี่ยนรายละเอียดใน **ส่วนที่ 1 — เขียนตามเว็บแอปที่คุณต้องการ** และปรับฟังก์ชันใน **ส่วนที่ 2 — เลือกฟังก์ชันที่ต้องการ** ให้เหมาะกับเว็บแอปของคุณ แต่ต้องรักษาฟังก์ชันพื้นฐานอย่าง **เพิ่ม แก้ไข ลบ และกรองรายการ** ไว้ด้วยนะครับ เพราะเราจะใช้ฝึกสร้างระบบให้ทำงานจริงในบทต่อ ๆ ไป ส่วนข้อกำหนดการสร้างและรูปแบบไฟล์ในส่วนที่ 3 ให้เก็บไว้ตามเดิมครับ

โครงประโยคต่อไปนี้ใช้เขียนแทนข้อความใต้หัวข้อ **การใช้งาน:** ในส่วนที่ 1 ได้ คุณไม่จำเป็นต้องรู้ว่าต้องมีปุ่มกี่ปุ่ม ใช้ภาษาที่อธิบายให้เพื่อนฟังได้เลยครับ:

> **PROMPT · คัดลอกไปใช้**
>
> แอปช่วย [ทำอะไร] บันทึก [ข้อมูลอะไร] แยกตาม [หมวดหมู่หรือสถานะ] และแสดง [ผลสรุปที่อยากเห็น] เพื่อให้ฉัน [เป้าหมายของการใช้งาน]

ไม่ต้องเติมครบทุกช่อง เลือกเฉพาะสิ่งที่ต้องใช้ ถ้ามีเงื่อนไขสำคัญ เช่น ภาษา สกุลเงิน หรือหน่วยวัด ให้บอกเพิ่มด้วย

ถ้ายังคิดออกแค่ว่า “เว็บแอปจดโน้ต” ก็ลองเริ่มจากตรงนั้นได้ครับ เรามีประโยค **“ช่วยออกแบบหน้าจอและฟังก์ชันที่เหมาะสมจากคำอธิบายนี้ให้ด้วย”** ให้ AI ช่วยเสนอรายละเอียด เมื่อเห็นต้นแบบแล้วค่อยตรวจและสั่งเพิ่มส่วนที่ขาด

ตัวอย่างอีก 5 เว็บแอป

ข้อความต่อไปนี้ใช้แทนเฉพาะส่วน **การใช้งาน:** ของ Prompt แรกได้ครับ ถ้าจะทดลองเว็บแอปอีกชนิด ให้เปิดแชตใหม่เพื่อแยกจากงานเดิม ส่วนใครทำเว็บแอปการเงินตามผมอยู่ อ่านไว้เป็นไอเดียแล้วไปเลือกธีมสีต่อได้เลย

1. เว็บแอปจดโน้ตส่วนตัว

PROMPT · คัดลอกไปใช้

การใช้งาน:
แอปช่วยจดโน้ตส่วนตัว บันทึกหัวข้อ เนื้อหา และวันที่ แยกหมวดหมู่ได้ ค้นหาโน้ตจากคำที่จำได้ และปักหมุดเรื่องสำคัญไว้ด้านบน เพื่อให้กลับมาหาข้อมูลได้ง่าย ใช้ภาษาไทย

2. เว็บแอปดูแลต้นไม้

PROMPT · คัดลอกไปใช้

การใช้งาน:
แอปช่วยดูแลต้นไม้ที่บ้าน บันทึกชื่อต้นไม้ ตำแหน่งที่ปลูก และวันที่รดน้ำหรือใส่ปุ๋ย กำหนดวันดูแลครั้งถัดไป และแสดงรายการที่ถึงกำหนดวันนี้ เพื่อให้รู้ว่าต้นไหนต้องดูแลอะไร ใช้ภาษาไทย

3. เว็บแอปติดตามการออกกำลังกาย

PROMPT · คัดลอกไปใช้

การใช้งาน:
แอปบันทึกการออกกำลังกาย บันทึกวันที่ ท่าออกกำลังกาย น้ำหนัก จำนวนครั้ง และจำนวนเซต แยกประเภทการฝึกได้ มีกราฟดูพัฒนาการและสรุปจำนวนวันที่ฝึกในแต่ละเดือน เพื่อให้ติดตามความสม่ำเสมอของตัวเอง ใช้ภาษาไทยและหน่วยกิโลกรัม

4. เว็บแอปจัดการออเดอร์ขนม

PROMPT · คัดลอกไปใช้

การใช้งาน:
แอปช่วยจัดการออเดอร์ขนม บันทึกชื่อลูกค้า เมนู จำนวน ราคา และวันนัดรับ แยกสถานะรอทำ ทำเสร็จ และส่งมอบแล้ว พร้อมสถานะการชำระเงิน สรุปยอดขายกับยอดค้างรับ เพื่อให้รู้ว่าแต่ละวันต้องเตรียมอะไรบ้าง ใช้ภาษาไทยและสกุลเงินบาท

5. เว็บแอปติดตามงานซ่อมบ้าน

PROMPT · คัดลอกไปใช้

การใช้งาน:
แอปช่วยติดตามงานซ่อมบ้าน บันทึกชื่องาน จุดที่ต้องซ่อม งบประมาณ ค่าใช้จ่ายจริง และวันที่ตั้งใจทำ แยกสถานะยังไม่เริ่ม กำลังทำ และเสร็จแล้ว มีสรุปงบที่ใช้ไป เพื่อให้วางแผนทำงานทีละจุดได้ ใช้ภาษาไทยและสกุลเงินบาท

รอบแรกเขียนเท่าที่นึกออกก่อนครับ พอเห็นหน้าตาเว็บแอปแล้ว คุณจะเริ่มรู้เองว่าอยากเพิ่มหรือเปลี่ยนอะไร ไม่ต้องคิดทุกอย่างให้ครบก่อนเริ่มเสก!

ส่วนประกอบครบแล้ว มาลองปรับหน้าตาเว็บแอปกัน

ตอนนี้เราได้ต้นแบบที่มีส่วนประกอบหลักครบแล้วครับ กลับมาที่แชตของเว็บแอปที่คุณเลือกทำต่อ เราจะลองปรับหน้าตาให้เป็นแบบที่ชอบ และฝึกสั่งให้ AI แก้งานจากของเดิมไปด้วย

คุณไม่จำเป็นต้องทำเว็บแอปการเงินเหมือนผมครับ สิ่งที่เราจะฝึกคือการมองหน้าจอของตัวเอง เลือกจุดที่อยากเปลี่ยน แล้วอธิบายให้ AI เข้าใจว่าต้องการให้ต่างจากเดิมอย่างไร วิธีนี้นำไปใช้ได้ทั้งกับเว็บแอปจดโน้ต เว็บแอปจัดการออเดอร์ หรือเว็บแอปอื่นที่คุณกำลังทำอยู่

รอบนี้แก้แค่ไหน?

ในบทนี้เราจะปรับ **หน้าตาของต้นแบบ** เช่น สีพื้นหลัง สีปุ่ม ข้อความบนหน้าจอ หรือส่วนที่มองเห็นแล้วอ่านยาก เพื่อให้เห็นภาพว่าเว็บแอปของเราจะออกมาประมาณไหนครับ

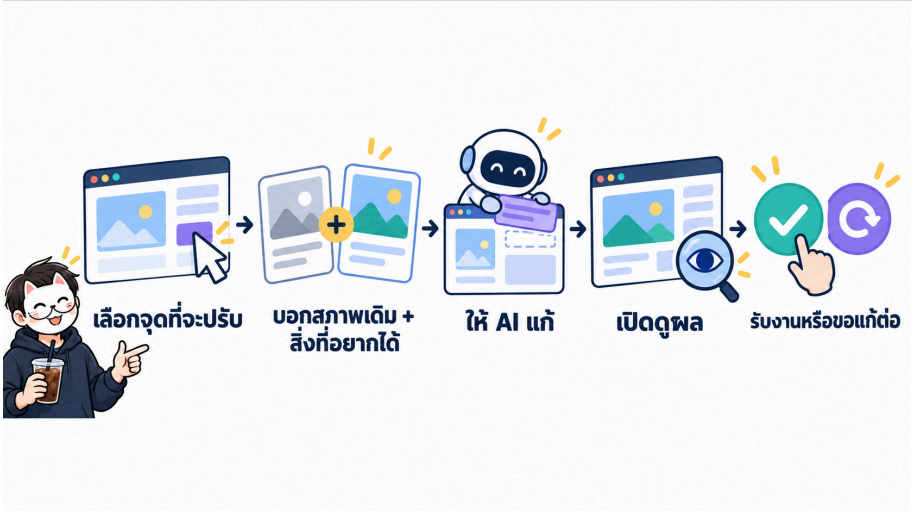
ส่วนการบันทึกข้อมูล การเข้าสู่ระบบด้วยบัญชีจริง หรือการคำนวณจากข้อมูลที่เราเพิ่มเข้าไป ยังเป็นงานในบทต่อ ๆ ไป ตอนนี้ถ้าเปิดฟอร์มได้ แต่กดบันทึกแล้วรายการไม่เพิ่ม ก็ยังไม่ต้องสั่งแก้เรื่องนั้น เพราะเราตั้งใจให้เป็นต้นแบบสำหรับกดสำรวจอยู่แล้ว

แต่ถ้ากดเปิดฟอร์มแล้วหาปุ่มปิดไม่เจอ หรือมีส่วนบังจนกดต่อไม่ได้ ควรแก้ให้กลับมาเดินดูต้นแบบได้ก่อนครับ

วิธีสั่งแก้ที่ใช้ได้กับเว็บแอปของคุณ

เริ่มจากเลือกสิ่งที่อยากปรับสักหนึ่งเรื่อง แล้วใช้ลำดับนี้ครับ:

เลือกจุดที่จะปรับ → บอกสภาพเดิมและสิ่งที่อยากได้ → ให้ AI แก้ → เปิดดูผล → รับงานหรือขอแก้ต่อ



เวลาพิมพ์ขอให้ AI แก้งาน ลองอธิบายให้ครบสามส่วนนี้ครับ:

1. **จุดที่ต้องการแก้และปัญหาที่พบ** — บอกว่าเป็นส่วนไหนของหน้าจอ และตอนนี้มีปัญหาอย่างไร เช่น “ปุ่มเพิ่มรายการบนหน้าหลักมีสีใกล้กับพื้นหลังจนมองหาได้ยาก”
2. **อยากให้เปลี่ยนอย่างไร** — บอกผลที่ต้องการให้ชัด เช่น “ช่วยปรับสีปุ่มให้เด่นขึ้น โดยยังเข้ากับธีมเดิม”
3. **เหตุผลหรือบริบทเพิ่มเติม** — อธิบายว่าทำไมจึงอยากเปลี่ยน เช่น “ปุ่มนี้ต้องใช้บ่อย อยากให้เปิดหน้าเว็บแอปมาแล้วมองเห็นได้ทันที”

การบอกเหตุผลช่วยให้ AI เข้าใจเป้าหมายของเรา และเลือกวิธีปรับได้ตรงใจมากขึ้นครับ



ไม่ต้องรู้ชื่อส่วนประกอบหรือใช้ศัพท์เทคนิคครับ ถ้าอธิบายตำแหน่งไม่ถูก ให้แคปภาพหน้าจอแล้ววงจุดที่ต้องการ ส่งพร้อมข้อความบอกสิ่งที่อยากเปลี่ยนได้เลย

ตัวอย่างเช่น หากปุ่มเพิ่มข้อมูลในเว็บแอปของคุณดูจางจนหาไม่เจอ คุณอาจพิมพ์ว่า:

> **PROMPT · คัดลอกไปใช้**
>
> ปุ่มเพิ่มรายการบนหน้าหลักสีกลืนกับพื้นหลังจนมองหาได้ยาก ช่วยเปลี่ยนสีปุ่มนี้ให้เด่นขึ้น โดยคงสีพื้นหลัง ข้อความ และตำแหน่งส่วนอื่นไว้เหมือนเดิม รอบนี้แก้เฉพาะหน้าตา ยังไม่เพิ่มระบบบันทึกข้อมูลจริง ขอให้อัปเดตทั้งตัวอย่างและชุดไฟล์ดาวน์โหลดด้วย

หากเว็บแอปของคุณเป็นเว็บแอปจดโน้ต ก็เปลี่ยนจาก “ปุ่มเพิ่มรายการ” เป็น “ปุ่มสร้างโน้ต” ได้ครับ หลักคือบอกให้ชัดว่าเรากำลังพูดถึงส่วนไหนและอยากได้ผลแบบใด

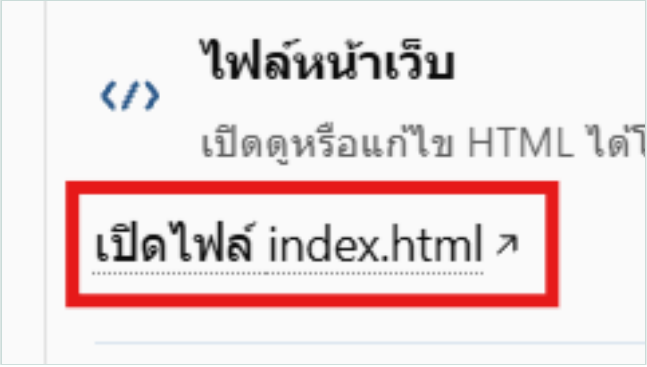
เริ่มแก้ทีละเรื่องก่อน จะตรวจได้ง่ายกว่าการสั่งเปลี่ยนหลายอย่างพร้อมกัน เมื่อ AI ทำเสร็จ ให้เปิดตัวอย่างดูด้วยตัวเองว่าจุดนั้นเปลี่ยนตรงตามที่ขอหรือยัง แล้วลองกดสำรวจส่วนที่เกี่ยวข้องด้วย เช่น ถ้าแก้ปุ่มเปิดฟอร์ม ก็ลองกดดูว่าฟอร์มยังเปิดและปิดได้เหมือนเดิม

ถ้ายังไม่ตรง ให้บอกสิ่งที่เห็นเพิ่มเติม เช่น “ปุ่มเด่นขึ้นแล้ว แต่ตัวหนังสือบนปุ่มยังอ่านยาก” เราจึงมีข้อมูลให้ AI แก้ต่อได้ชัดกว่าการบอกเพียงว่า “ยังไม่สวย” ครับ

ตัวอย่างที่ผมแก้จริง

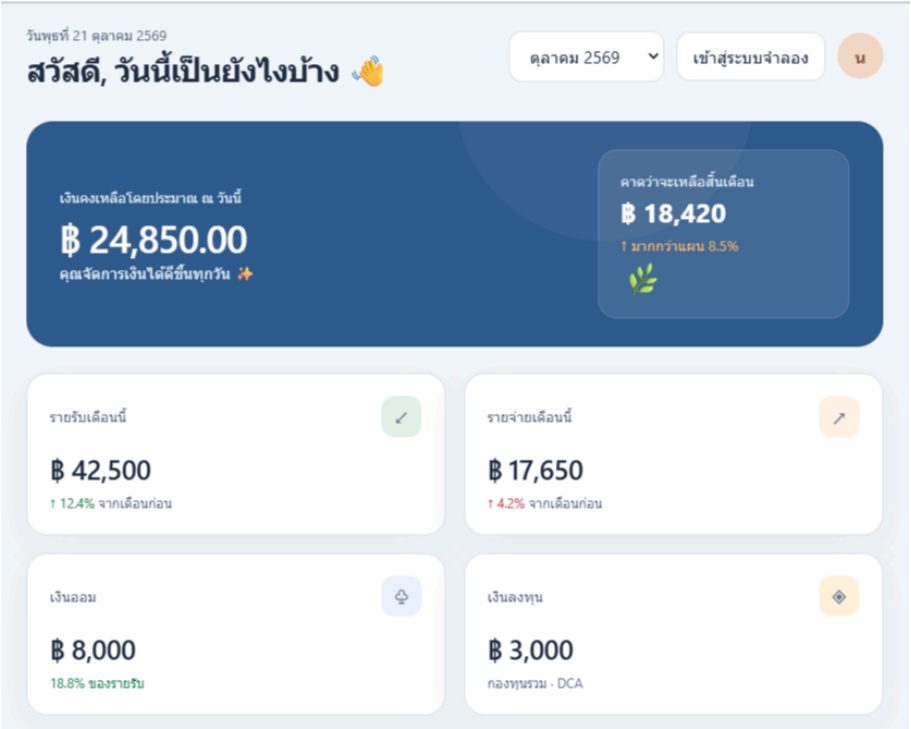
เปิดไฟล์หน้าเว็บตามที่ AI แนะนำเพื่อดูต้นแบบครับ ในตัวอย่างนี้ใช้ไฟล์ `index.html` ซึ่งเป็นชื่อไฟล์เริ่มต้นที่พบได้บ่อยในเว็บแอป แต่ไม่ใช่ทุกโปรเจกต์จะต้องเปิดไฟล์ชื่อนี้เสมอไป

ภาพขยาย 03 · ลิงก์เปิดไฟล์ index.html

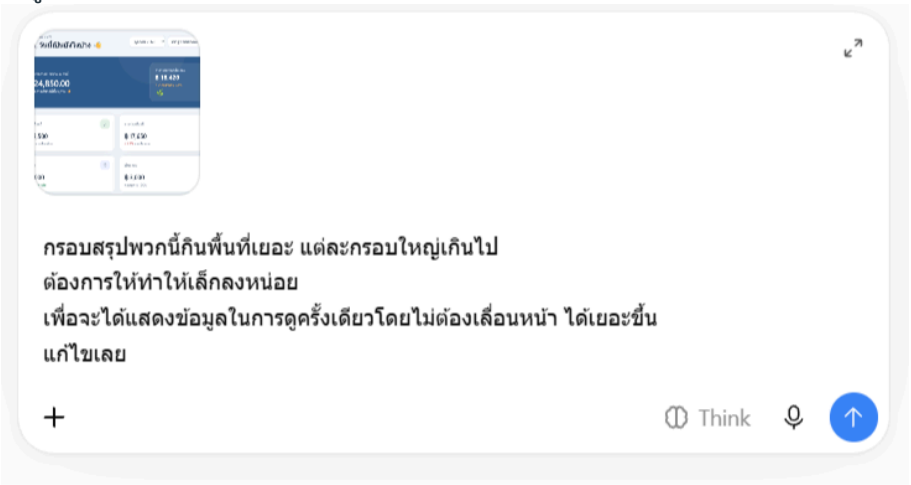


> กดลิงก์เปิดไฟล์ index.html ในการ์ดไฟล์เว็บไซต์ เพื่อเปิดต้นแบบล่าสุดขึ้นมาตรวจบนหน้าจอ

แคปรูปส่วนที่ต้องการแก้



ส่งรูปพร้อมแชต



โดยแชตจะใช้ Prompt ดังนี้

จุดที่ต้องแก้ และปัญหา

> **PROMPT · คัดลอกไปใช้**
>
> กรอบสรุปพวกนี้กินพื้นที่เยอะ แต่ละกรอบใหญ่เกินไป ต้องการให้ทำให้เล็กลงหน่อย เพื่อจะได้แสดงข้อมูลในการดูครั้งเดียวโดยไม่ต้องเลื่อนหน้า ได้เยอะขึ้น แก้ไขเลย

อยากให้เปลี่ยนอย่างไร

> **PROMPT · คัดลอกไปใช้**
>
> ต้องการให้ทำให้เล็กลงหน่อย

เหตุผล เพื่อจะได้แสดงข้อมูลในการดูครั้งเดียวโดยไม่ต้องเลื่อนหน้า ได้เยอะขึ้น

เน้นย้ำให้เริ่มทำงานเลย

> **PROMPT · คัดลอกไปใช้**
>
> แก้ไขเลย

รวมเป็น Prompt

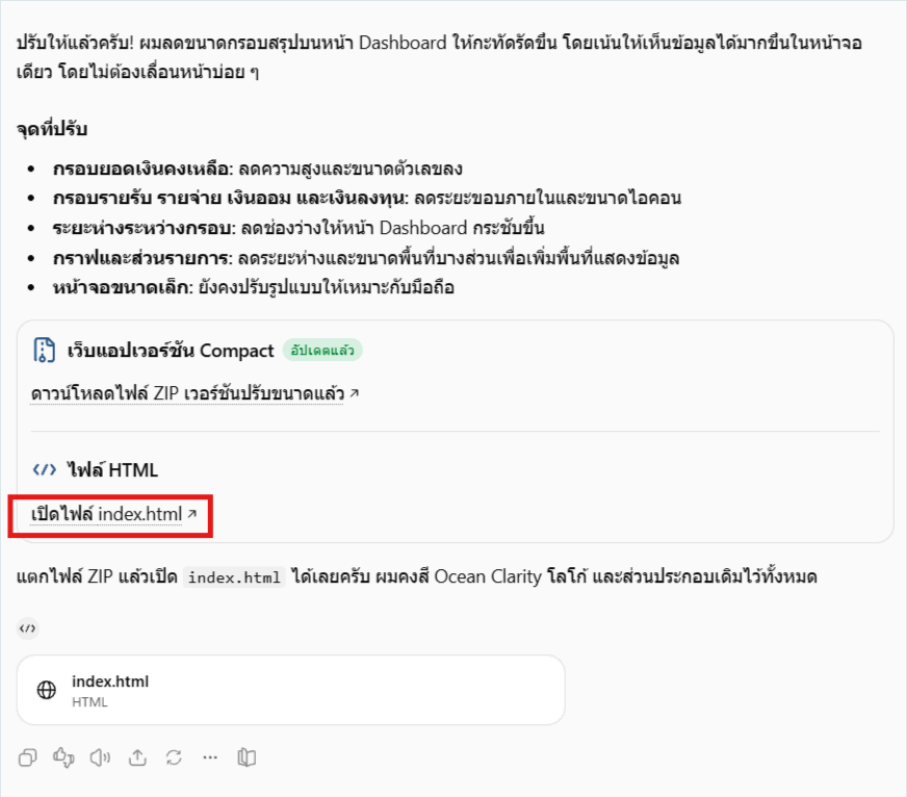
> **PROMPT · คัดลอกไปใช้**
>
> กรอบสรุปพวกนี้กินพื้นที่เยอะ แต่ละกรอบใหญ่เกินไป ต้องการให้ทำให้เล็กลงหน่อย เพื่อจะได้แสดงข้อมูลในการดูครั้งเดียวโดยไม่ต้องเลื่อนหน้า ได้เยอะขึ้น แก้ไขเลย

ผมแค่บอกว่ากรอบใหญ่เกินไป อยากให้เล็กลง และบอกเหตุผลว่าต้องการเห็นข้อมูลบนหน้าจอมากขึ้น

เท่านี้ AI ก็มีข้อมูลเพียงพอที่จะเริ่มแก้งานแล้วครับ

หลังแก้

AI ตอบกลับว่าปรับกรอบสรุปบนหน้า Dashboard ให้เล็กลงแล้ว พร้อมสรุปจุดที่แก้ และส่งชุดไฟล์ฉบับใหม่มาให้ดาวน์โหลดครับ ในคำตอบยังมีลิงก์ **เปิดไฟล์ `index.html`** ให้กดดูผลได้ทันทีด้วย



AI ตอบกลับพร้อมสรุปสิ่งที่แก้และไฟล์ฉบับใหม่

เมื่อเปิดดู จะเห็นว่ากรอบรายรับ รายจ่าย เงินออม และเงินลงทุนเตี้ยและกะทัดรัดขึ้น ทำให้มองเห็นกราฟกับข้อมูลส่วนถัดไปได้มากกว่าเดิมโดยไม่ต้องเลื่อนหน้าบ่อย ๆ ส่วนธีมสีและองค์ประกอบเดิมยังอยู่ครบครับ

ภาพขยาย 02 · กรอบสรุปและกราฟหลังปรับ Dashboard



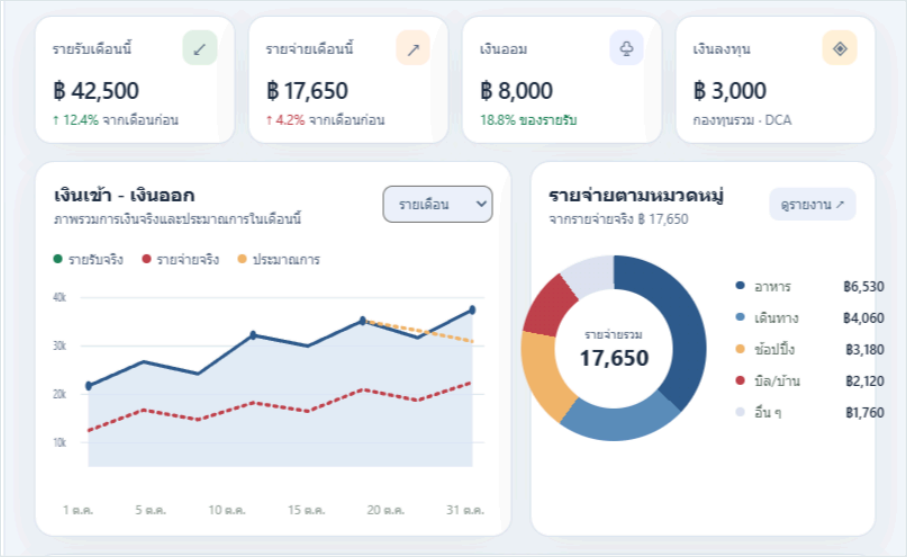
หน้า Dashboard หลังย่อกรอบสรุปให้กะทัดรัดขึ้น

ตรวจว่ากรอบยอดคงเหลือ รายรับ รายจ่าย เงินออม และเงินลงทุนเตี้ยลงแต่ตัวเลขยังอ่านได้ครบ และกราฟเส้นกับรายจ่ายตามหมวดหมู่ปรากฏในหน้าจอเดียวกันได้มากขึ้น

PROMPT · คัดลอกไปใช้

บันทึก ZIP สำรอง: หลัง AI ส่ง ZIP ที่ย่อกรอบสรุปแล้ว ให้ดาวน์โหลดเก็บเป็นชุดใหม่ในโฟลเดอร์สำรอง เช่น ละมุนเงิน-03-ย่อกรอบสรุป.zip โดยเปลี่ยนเลขเป็นลำดับถัดไปที่ยังไม่ใช้ อย่าเขียนทับ ZIP ก่อนหน้า เผื่อการแก้รอบถัดไปมีจุดอื่นคลาดเคลื่อน

คำขอแรกถือว่าทำได้ตามเป้าหมายแล้ว แต่ยังไม่ควรรีบรับงานทันทีครับ ลองมองส่วนอื่นที่อยู่ใกล้กับจุดที่แก้ด้วย ในรอบนี้ผมพบว่ากรอบ **รายจ่ายตามหมวดหมู่** กะทัดรัดขึ้นก็จริง แต่แผนภูมิวงกลมกับข้อความรายการหมวดหมู่เริ่มเบียดกันจนดูแน่นเกินไป



จุดที่พบหลังแก้ — แผนภูมิวงกลมและข้อความในกรอบรายจ่ายเบียดกัน

ผมจึงถ่ายภาพหน้าจอเฉพาะจุดที่มีปัญหา ส่งกลับไปในแชตเดิม แล้วบอกทั้งสิ่งที่เห็นและข้อจำกัดที่ต้องการรักษาไว้ คือให้จัดเนื้อหาภายในกรอบใหม่ แต่ไม่ขยายกรอบให้ใหญ่กลับไปเหมือนเดิม

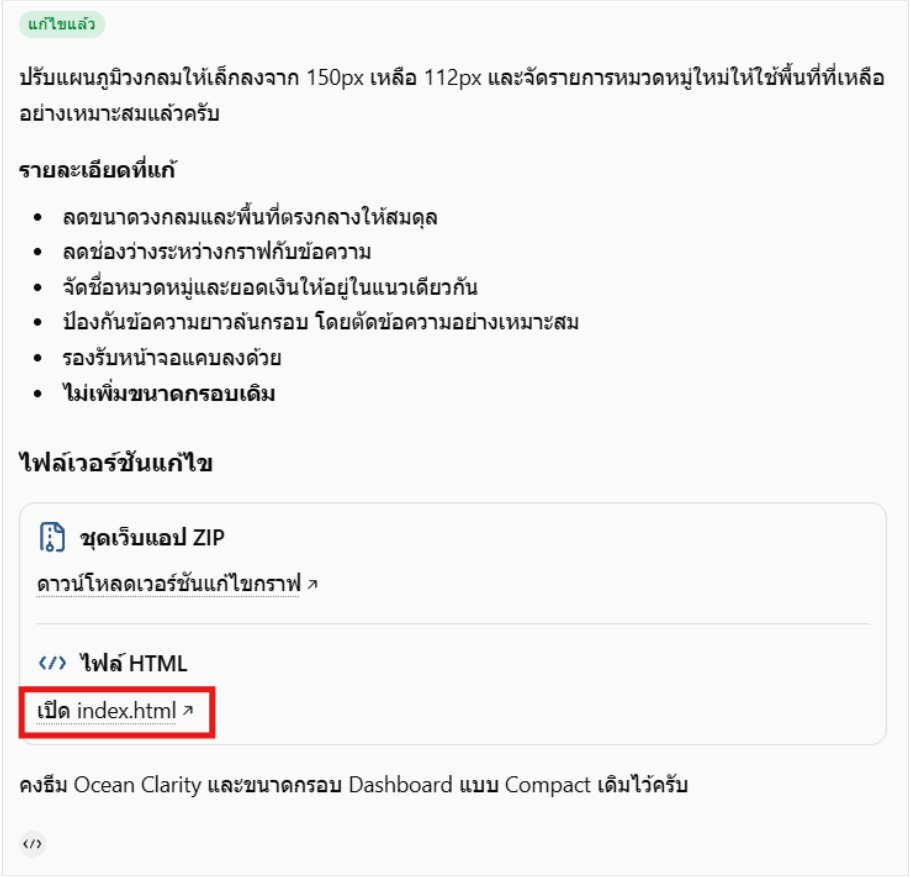
ข้อความที่ใช้สั่งต่อมีประมาณนี้ครับ:

> **PROMPT · คัดลอกไปใช้**
>
> กรอบรายจ่ายตามหมวดหมู่มีแผนภูมิวงกลมและข้อความรายการหมวดหมู่เบียดกัน ช่วยจัดให้ทุกอย่างแสดงพอดีอยู่ภายในกรอบ โดยคงขนาดกรอบปัจจุบันไว้ ไม่ต้องขยายกรอบให้ใหญ่ขึ้น แก้ไขเลย



ส่งภาพจุดที่พบพร้อมข้อความขอให้แก้ต่อ

AI ปรับขนาดแผนภูมิและจัดวางรายการหมวดหมู่ใหม่ โดยยืนยันว่าคงขนาดกรอบเดิมไว้ พร้อมส่งชุดไฟล์ที่แก้แล้วมาให้อีกครั้งครับ



AI ตอบกลับพร้อมรายละเอียดและชุดไฟล์หลังแก้กราฟ

เมื่อได้ไฟล์รอบใหม่ ให้เปิด `index.html` แล้วตรวจอีกครั้งว่าแผนภูมิและข้อความไม่เบียดหรือล้นกรอบ และส่วนที่แก้ผ่านรอบแรกยังคงอยู่ครบ วิธีทำงานจึงไม่ได้จบแค่ AI บอกว่า “แก้แล้ว” แต่จบเมื่อเราเปิดดูและยืนยันผลด้วยตัวเองครับ

PROMPT · คัดลอกไปใช้

บันทึก ZIP สำรอง: เมื่อได้ ZIP หลังจัดกราฟแล้ว ให้เก็บเป็นอีกชุด เช่น ละมุนเงิน-04-จัดกราฟ.zip โดยเปลี่ยนเลขเป็นลำดับถัดไปที่ยังไม่ใช้ เพื่อให้ย้อนกลับไปยังฉบับก่อนแก้ได้



ผ่านครับ!

ลองฝึกด้วยการเลือกธีมสี

คราวนี้มาลองใช้วิธีข้างต้นกับการเปลี่ยนสีเว็บแอปกันครับ **ธีมสี** คือชุดสีที่ใช้ร่วมกันบนหน้าจอ เช่น สีพื้นหลัง สีตัวหนังสือ สีปุ่ม และสีส่วนแสดงข้อมูล การเลือกเป็นชุดช่วยให้ส่วนต่าง ๆ ของเว็บแอปดูเข้ากัน

ถ้ามีชุดสีที่ชอบอยู่แล้ว ส่งภาพชุดสีนั้นให้ ChatGPT แล้วบอกให้นำมาใช้กับเว็บแอปได้เลย ถ้ามีรหัสสีติดมาด้วย ก็แนบไปพร้อมกันได้ครับ

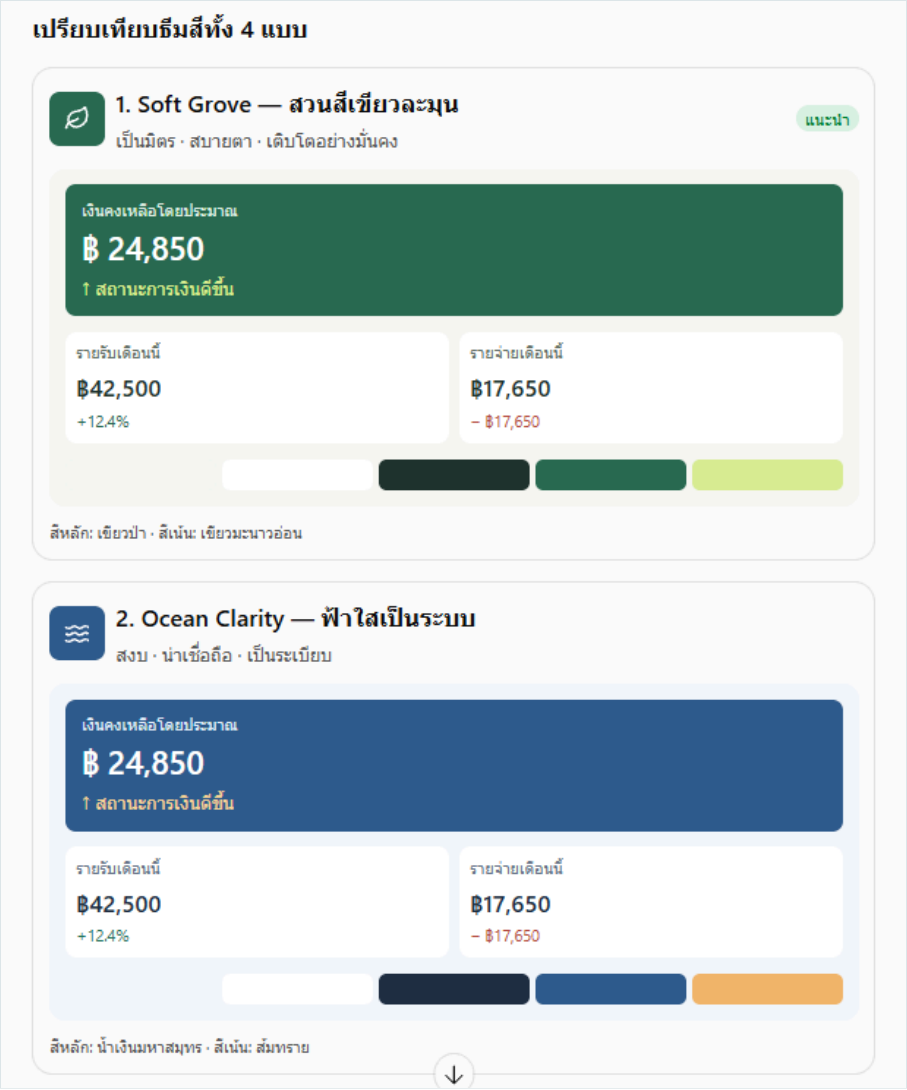
แต่ถ้ายังนึกไม่ออกว่าจะใช้สีอะไร ให้ AI เสนอทางเลือกก่อน รอบนี้เรายังไม่ต้องสั่งเปลี่ยนหน้าเว็บแอปทันที ลองดูชุดสีและคำอธิบายว่าแต่ละแบบให้อารมณ์อย่างไรก่อนครับ

พิมพ์ต่อในแชตเดิมว่า:

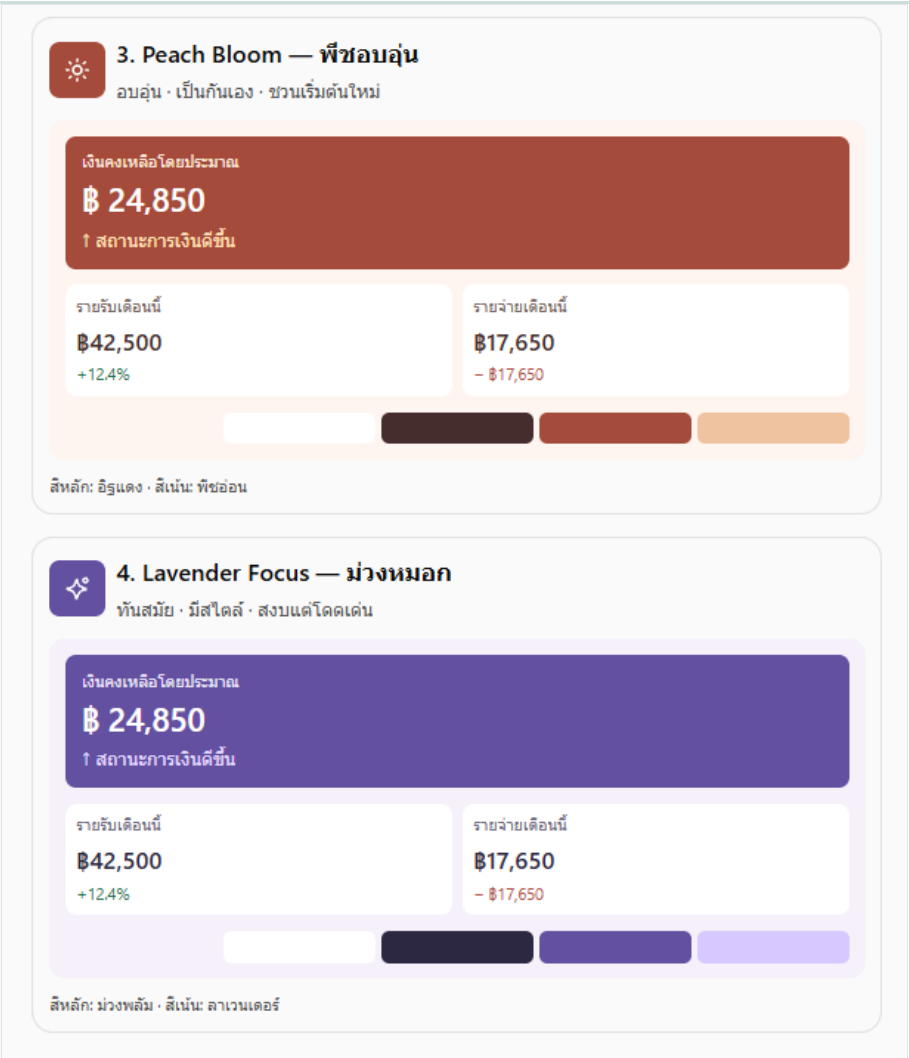
> **PROMPT · คัดลอกไปใช้**
>
> ช่วยเสนอธีมสีสำหรับแอปบันทึกรายรับรายจ่ายนี้ 4 แบบ แต่ละแบบให้มีบุคลิกต่างกัน พร้อมตั้งชื่อธีมและอธิบายว่าจะใช้แต่ละสีกับส่วนใดของเว็บแอป ขอให้แสดงเป็นตัวอย่างชุดสีให้เปรียบเทียบได้ง่าย โดยเน้นอ่านสบายและข้อความชัดเจน รอบนี้เสนอให้เลือกก่อน ยังไม่ต้องแก้ไฟล์แอป

ถ้าคุณสร้างเว็บแอปชนิดอื่น ให้เปลี่ยนชื่อเว็บแอปในคำสั่งให้ตรงกับของคุณนะครับ

นี่คือตัวอย่างธีมสีที่ AI เสนอให้ผมครับ:



ตัวอย่างข้อเสนอธีมสี — ภาพที่ 1



ตัวอย่างข้อเสนอธีมสี — ภาพที่ 2

ลองดูทั้งชุดสีและคำอธิบายประกอบ เช่น พื้นหลังเป็นสีอะไร ข้อความใช้สีอะไร และปุ่มสำคัญจะใช้สีไหน ถ้าชอบสีบางชุดแต่ยังนึกภาพไม่ออก ก็ถาม AI ให้ช่วยอธิบายเพิ่มก่อนเลือกได้ครับ

เลือกชุดที่เห็นแล้วอยากเปิดเว็บแอปใช้ทุกวัน ไม่ต้องหาสีที่ถูกใจคนทั้งโลกครับ เอาใจเจ้าของก่อนหนึ่งคน

เลือกแล้ว สั่งให้นำมาใช้กับเว็บแอป

การได้ชุดสีมาให้เลือกยังไม่ได้หมายความว่าไฟล์เว็บเปลี่ยนตามแล้วนะครับ เราต้องบอก AI อีกครั้งว่าจะเลือกชุดไหน และให้นำไปใช้กับต้นแบบเดิม

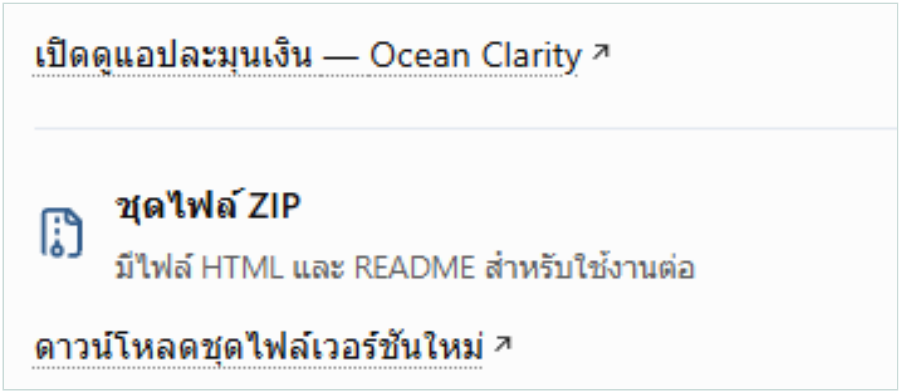
ตัวอย่างของผมเลือกธีมหมายเลข 2 จึงสั่งต่อได้ว่า:

> **PROMPT · คัดลอกไปใช้**
>
> ช่วยนำชุดสีของธีมหมายเลข 2 ที่เสนอเมื่อกี้มาใช้กับเว็บแอปเดิม ช่วยเน้นให้สีรายรับกับรายจ่ายแตกต่างกันชัดเจน รอบนี้ปรับเฉพาะสี โดยคงหน้าจอ ปุ่ม ช่องข้อมูล และการกดสำรวจเดิมไว้ ยังไม่เพิ่มระบบบันทึกข้อมูลจริง ขอให้อัปเดตทั้งตัวอย่างและชุดไฟล์ดาวน์โหลดด้วย

เปลี่ยนหมายเลขให้ตรงกับธีมที่คุณเลือกครับ ถ้า AI ตั้งชื่อธีมไว้ จะระบุชื่อแทนหมายเลขก็ได้ ส่วนเงื่อนไขเรื่องสีรายรับกับรายจ่ายเป็นความต้องการของเว็บแอปการเงิน ใครทำเว็บแอปอื่นก็เปลี่ยนหรือตัดข้อความส่วนนี้ได้

ที่ขอให้อัปเดตทั้งสองอย่าง เพราะเราต้องการเปิดดูต้นแบบที่แก้แล้ว และเก็บไฟล์ฉบับเดียวกันไปทำต่อบนเครื่องด้วย ถ้าได้ตัวอย่างใหม่แต่ดาวน์โหลดไฟล์ชุดเก่า หน้าตาที่เปิดบนเครื่องอาจยังเป็นแบบเดิมครับ

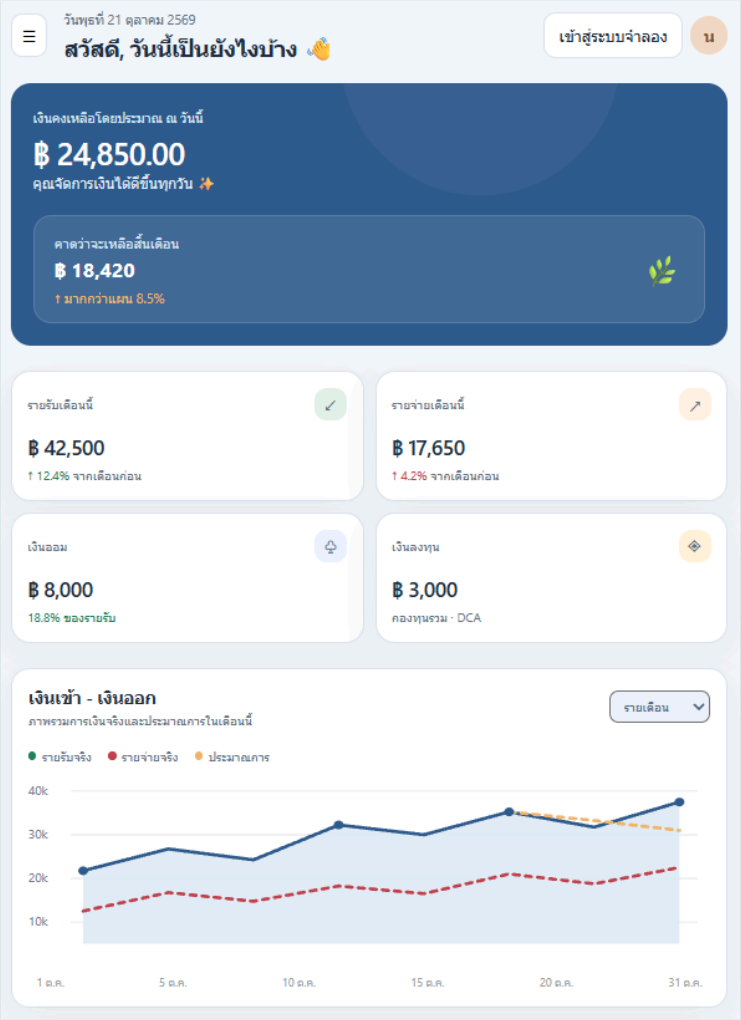
ภาพขยาย 04 · ลิงก์เปิดแอปและดาวน์โหลดชุดไฟล์



> ตัวอย่างคำตอบหลังสั่งเปลี่ยนธีม

> ใช้ลิงก์เปิดแอปเวอร์ชันใหม่เพื่อตรวจผล แล้วดาวน์โหลดชุดไฟล์เวอร์ชันใหม่เพื่อเก็บแยกจากชุดเดิม

รอบนี้ของผมไม่ได้แสดงหน้าเว็บให้ดูแบบเดิม แต่มีลิงก์ชื่อ **“เปิดดูแอปละมุนเงิน — Ocean Clarity”** ให้กดเปิดแทนครับ ของคุณอาจใช้ชื่ออื่น ให้ดูจากคำตอบที่ได้รับ



ตัวอย่างหน้าเว็บแอปหลังเปลี่ยนธีม

PROMPT · คัดลอกไปใช้

บันทึก ZIP สำรอง: หลัง AI นำธีมที่เลือกไปใช้และส่ง ZIP ฉบับใหม่แล้ว ให้ดาวน์โหลดเก็บเป็นชุดถัดไป เช่น ละมุนเงิน-05-ใช้ธีมสี.zip โดยใช้เลขลำดับถัดไปที่ยังไม่เคยใช้ อย่าเก็บทับชุดก่อนหน้า

เปิดดูผล แล้วค่อยตัดสินใจ

หลัง AI แก้เสร็จ อย่าเพิ่งดูแค่คำตอบว่าแก้แล้วครับ ให้เปิดหน้าเว็บแอปที่อัปเดตขึ้นมาดู เริ่มจากหน้าหลักแล้วลองเปิดฟอร์มสักครั้ง เพื่อดูว่าสีใหม่ไปถึงส่วนที่เราต้องใช้หรือยัง

ลองอ่านชื่อรายการและจำนวนเงินตามปกติ ถ้าต้องเพ่งเพราะตัวหนังสือจาง ก็จดไว้ว่าเป็นข้อความตรงไหน จากนั้นมองหาปุ่มเพิ่มรายการว่าหาเจอง่ายหรือไม่ และถ้าเป็นเว็บแอปการเงิน ลองดูว่าสีรายรับกับรายจ่ายแยกกันได้ชัดหรือยัง

หากชอบภาพรวมแล้ว แต่ยังมีจุดอ่านยาก ให้ขอแก้เฉพาะจุดนั้นได้ครับ ไม่จำเป็นต้องเริ่มเลือกธีมใหม่ทั้งหมด เช่น:

PROMPT · คัดลอกไปใช้

ชอบชุดสีนี้แล้ว แต่ข้อความสีเทาในกล่องสรุปยอดบนหน้าหลักยังจางจนอ่านยาก ช่วยเพิ่มความเข้มของข้อความตรงนี้ โดยคงสีพื้นหลัง สีปุ่ม และส่วนอื่นไว้ตามเดิม ขอให้อัปเดตทั้งตัวอย่างและชุดไฟล์ดาวน์โหลดด้วย

เมื่อได้ตัวอย่างใหม่ ให้กลับไปดูจุดเดิมว่าดีขึ้นตามที่ขอไหม หากยังไม่ตรงก็บอกอาการที่เหลืออยู่ หากตรงแล้วก็รับผลและไปต่อครับ นี่คือวงจรเดียวกับที่เราฝึกไว้: เลือกจุด บอกสิ่งที่อยากได้ ให้ AI แก้ แล้วเปิดดูผล

> **PROMPT · คัดลอกไปใช้**
>
> **บันทึก ZIP สำรอง:** ถ้าการแก้จุดอ่านยากทำให้ AI ส่ง ZIP ฉบับใหม่มา ให้ดาวน์โหลดเก็บเป็นชุดถัดไปด้วย เช่น ละมุนเงิน-06-ปรับข้อความ.zip โดยใช้เลขลำดับถัดไปที่ยังไม่เคยใช้ เพื่อเก็บผลของแต่ละรอบไว้เทียบหรือย้อนกลับได้

พอแค่ไหนถึงไปต่อได้?

บทนี้เราต้องการต้นแบบที่ใช้เป็นแนวทางทำเว็บแอปต่อครับ รายละเอียดอย่างระยะห่าง ขนาดตัวหนังสือ และการจัดวางบนอุปกรณ์ต่าง ๆ เราจะกลับมาเก็บอย่างละเอียดใน VS Code ช่วงท้ายของเล่ม

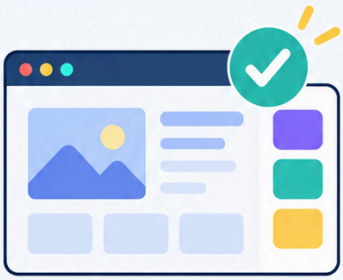
สำหรับรอบนี้ ให้ตรวจสามข้อก่อน:

- ☐ **ทิศทางหน้าตาใช่แล้ว** — สีและบรรยากาศโดยรวมเป็นแบบที่อยากทำต่อ ไม่จำเป็นต้องลองทุกธีมจนแน่ใจว่าเป็นชุดที่ดีที่สุด
- ☐ **ส่วนประกอบหลักยังครบ** — หน้าจอ ปุ่ม และช่องข้อมูลที่ตรวจไว้ก่อนหน้ายังอยู่หลังการปรับหน้าตา
- ☐ **อ่านและกดสำรวจได้** — ข้อความหลักอ่านออก ปุ่มสำคัญหาเจอ เปิดฟอร์มและปิดกลับได้ ไม่มีส่วนบังจนไปต่อไม่ได้

ตรวจว่า Web App Mockup ผ่าน นำไปใช้ต่อได้

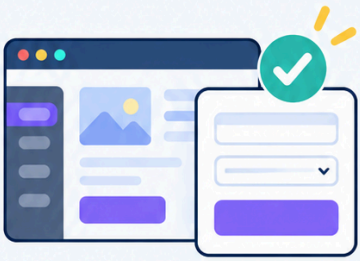
ทิศทางหน้าตาใช่แล้ว

สีและบรรยากาศโดยรวมเป็นแบบที่อยากทำต่อ ไม่จำเป็นต้องลองทุกธีมจนแน่ใจว่าเป็นชุดที่ดีที่สุด



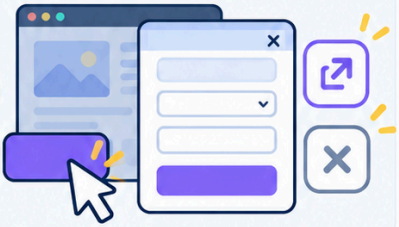
ส่วนประกอบหลักยังครบ

หน้าจอ ปุ่ม และช่องข้อมูลที่ตรวจไว้ก่อนหน้ายังอยู่หลังการปรับหน้าตา



อ่านและกดสำรวจได้

ข้อความหลักอ่านออก ปุ่มสำคัญหาเจอ เปิดฟอร์มและปิดกลับได้ ไม่มีส่วนบังจนไปต่อไม่ได้



ถ้าครบทั้งสามข้อก็ไปต่อได้ครับ แม้จะยังรู้สึกว่าปุ่มมนกว่านี้อีกนิดน่าจะสวย หรือกล่องบางใบเว้นระยะได้ดีกว่านี้ จดสิ่งที่อยากปรับไว้ก่อนได้ แล้วนำไปเก็บรายละเอียดภายหลัง

แต่ถ้าข้อความจางจนอ่านไม่ออก หรือปุ่มปิดอยู่นอกจอจนปิดฟอร์มไม่ได้ ให้แก้จุดนั้นก่อน เพราะเป็นอุปสรรคต่อการสำรวจต้นแบบครับ

เมื่อผ่านแล้ว ถ้ามี ZIP ฉบับใหม่จากการปรับในช่วงนี้ ให้ดาวน์โหลดเก็บเป็นชุดถัดไปในโฟลเดอร์สำรองด้วยครับ จากนั้นเราจะขอรวมไฟล์ฉบับล่าสุดอีกครั้งก่อนจบบท

เติมโลโก้ ให้เว็บแอปมีหน้าเป็นของตัวเอง

ขั้นตอนนี้เป็นของเสริม ข้ามได้ครับ เราจะลองสร้างโลโก้แยกเก็บไว้เป็นรูปสำหรับนำไปใช้กับเว็บแอปภายหลัง ยังไม่ใส่รูปลงในไฟล์เว็บในบทนี้

เริ่มจากบอกว่าอยากได้รูปอะไร ลักษณะแบบไหน และใช้สีอย่างไร เช่น เว็บแอปการเงินของผมอาจใช้กระเป๋าเงินเป็นสัญลักษณ์ ใครทำเว็บแอปดูแลต้นไม้อาจเลือกกระถางเล็ก ๆ ก็ได้ ไม่จำเป็นต้องใช้รูปเหมือนกันครับ

ลองพิมพ์ต่อในแชตเดิมว่า:

> **PROMPT · คัดลอกไปใช้**
>
> ช่วยออกแบบโลโก้น่ารักเรียบง่ายให้แอปนี้หน่อย เป็นรูปกระเป๋าเงินใบเล็กมีหน้าอมยิ้ม ใช้สีเข้ากับธีมที่เลือก และไม่ต้องมีข้อความ



ตัวอย่างโลโก้ที่ได้จากคำสั่งแรก

ได้รูปมาแล้ว ให้ลองดูว่ารูปสื่อถึงสิ่งที่ต้องการหรือไม่ และสีเข้ากับเว็บแอปไหมครับ ถ้ายังไม่ถูกใจ ใช้วิธีเดียวกับการแก้หน้าตาเว็บ: บอกจุดที่อยากเปลี่ยนและส่วนที่อยากเก็บไว้

ตัวอย่างเช่น หากชอบรูปกระเป๋าและสีแล้ว แต่รู้สึกว่ารายละเอียดเยอะไป อาจขอว่า:

> **PROMPT · คัดลอกไปใช้**
>
> ชอบรูปกระเป๋าเงินและชุดสีนี้แล้ว ช่วยลดรายละเอียดเล็ก ๆ ให้รูปเรียบง่ายขึ้น โดยคงรูปทรงหลักและหน้าอมยิ้มไว้ ไม่ต้องเพิ่มข้อความ

ข้อความนี้เป็นตัวอย่างวิธีขอแก้นะครับ คุณควรบอกตามสิ่งที่เห็นในรูปของตัวเอง ส่วนภาพถัดไปเป็นตัวอย่างโลโก้หลังผมลองขอแก้อีกครั้ง:



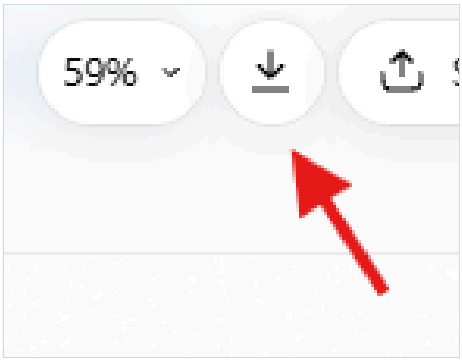
> ตัวอย่างโลโก้หลังขอปรับแก้

โอเค! สมมติว่าผมถูกใจแล้ว 555+

รอบนี้แค่ได้รูปที่ชอบพอจะนำไปใช้ต่อก็พอครับ ไม่ต้องวนออกแบบจนสมบูรณ์ที่สุด และถ้าบัญชีที่ใช้ติดข้อจำกัดการสร้างรูป ข้ามไปทำเว็บต่อได้เลย โลโก้กลับมาเพิ่มภายหลังได้

เมื่อเลือกได้แล้ว ให้กดเปิดรูปเต็มและใช้ปุ่มดาวน์โหลดหรือบันทึกรูป เก็บเป็นไฟล์รูปบนเครื่อง แยกจากชุดไฟล์เว็บต้นแบบไว้ก่อนครับ

ภาพขยาย 05 · ปุ่มดาวน์โหลดรูปโลโก้



> ตัวอย่างการบันทึกไฟล์โลโก้

> เปิดรูปเต็มแล้วกดสัญลักษณ์ลูกศรลงบนแถบด้านบน ลูกศรสีแดงในภาพเดิมชี้ตำแหน่งปุ่มนี้ไว้

การเห็นโลโก้ในแชตยังไม่ได้หมายความว่ารูปถูกใส่ในไฟล์เว็บแล้ว ตอนนี้เรามีไฟล์รูปเตรียมไว้ ส่วนการนำไปติดบนหน้าเว็บหรือใช้เป็นไอคอนจะทำภายหลังครับ

เสร็จครับ คุณได้ต้นแบบเว็บแอปของตัวเองแล้ว!

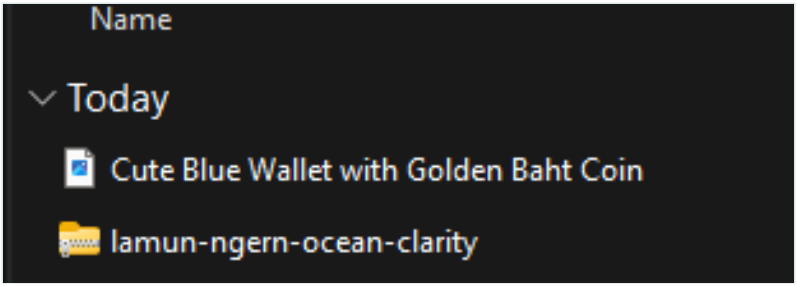
ก่อนปิดแชต เราจะรวบรวมงานฉบับล่าสุดเก็บไว้ครับ โดยเฉพาะถ้าแก้หลายรอบจนไม่แน่ใจว่าลิงก์ดาวน์โหลดใดเป็นชุดปัจจุบัน ให้ขอชุดใหม่ที่รวมการแก้ทั้งหมดอีกครั้ง

1. ขอชุดไฟล์ล่าสุด

พิมพ์ในแชตเดิมว่า:

PROMPT · คัดลอกไปใช้

รวมงานเว็บแอปฉบับล่าสุดที่มีหน้าจอและส่วนประกอบครบ พร้อมธีมสีที่เลือก เป็นชุดไฟล์ ZIP สำหรับดาวน์โหลด ขอรายการไฟล์ทั้งหมดและบอกด้วยว่าหลังแตก ZIP แล้วต้องเปิดไฟล์ใดเพื่อดูต้นแบบบนเครื่อง โลโก้ที่สร้างแยกไว้ยังไม่ต้องนำมาใส่ในเว็บ



ตัวอย่างชุดไฟล์สำหรับดาวน์โหลดเมื่อจบบท

2. ดาวน์โหลดและแยกเก็บให้หาเจอ

ดาวน์โหลด ZIP ฉบับสุดท้ายไว้ในโฟลเดอร์สำรองเดิม แล้วตั้งเลขถัดไปจากชุดก่อนหน้า เช่น `ละมุนเงิน-07-ฉบับจบบท.zip` โดยใช้เลขลำดับถัดไปที่ยังไม่เคยใช้ เก็บ ZIP ระหว่างทางไว้ด้วยครับ เลขลำดับจะช่วยให้เห็นว่าไฟล์ใดใหม่สุด และยังมีชุดก่อนหน้าให้ย้อนกลับหากพบปัญหาภายหลัง

ถ้าทำโลโก้ด้วย ให้นำไฟล์รูปมาเก็บใกล้กัน แต่ยังไม่ต้องแทรกเข้าไปในโฟลเดอร์ย่อยของเว็บเอง ตอนนี้ขอให้รู้ว่าไฟล์เว็บอยู่ที่ไหนและไฟล์โลโก้อยู่ที่ไหนก็พอ

3. เปิดต้นแบบจากชุดที่ดาวน์โหลด

แตกไฟล์ ZIP แล้วเทียบกับรายการไฟล์ที่ AI แจ้งไว้ จากนั้นเปิดไฟล์เริ่มต้นตามคำแนะนำของ AI ชื่อไฟล์ของแต่ละคนอาจต่างกัน จึงให้ยึดคำแนะนำที่มากับงานของคุณครับ

คราวนี้เรากำลังตรวจชุดไฟล์ที่เก็บบนเครื่อง ให้ลองดูว่าหน้าตาเป็นธีมล่าสุดหรือไม่ มีส่วนประกอบที่ขอแก้ครบไหม และยังกดเข้าหน้าหลัก เปิดฟอร์ม และปิดกลับได้หรือเปล่า

ถ้าเปิดไม่ได้หรือไม่แน่ใจว่าต้องเปิดไฟล์ไหน ให้กลับไปถามในแชตเดิมพร้อมภาพหรือข้อความที่พบ เช่น:

> **PROMPT · คัดลอกไปใช้**
>
> ฉันดาวน์โหลดและแตก ZIP แล้ว แต่ยังไม่แน่ใจว่าต้องเปิดไฟล์ใดเพื่อดูต้นแบบบนเครื่อง นี่คือภาพรายการไฟล์ที่ได้ ช่วยบอกขั้นตอนเปิดจากชุดนี้ให้หน่อย

ถ้าเปิดได้แต่ยังเป็นสีเดิม ให้บอกสิ่งที่พบโดยตรง:

> **PROMPT · คัดลอกไปใช้**
>
> ตัวอย่างล่าสุดในแชตเปลี่ยนเป็นธีมที่เลือกแล้ว แต่ชุด ZIP ที่เพิ่งดาวน์โหลด เมื่อแตกไฟล์และเปิดตามคำแนะนำยังเป็นสีเดิม ช่วยตรวจและส่งชุดไฟล์ที่ตรงกับตัวอย่างล่าสุดให้ใหม่ด้วย

ไม่ต้องเดาสาเหตุเองครับ บอกว่าเปิดไฟล์ใดแล้วเห็นอะไร ให้ AI ช่วยตรวจต่อ และเมื่อได้ชุดแก้ไขแล้วให้ลองเปิดตรวจอีกครั้ง

เช็กของก่อนจบบท

- ☐ มีชุดไฟล์ ZIP ต้นแบบฉบับล่าสุด เก็บไว้ในที่ที่หาเจอ
- ☐ แตกไฟล์และเปิดสำรวจบนเครื่องตามคำแนะนำได้ หน้าตาตรงกับฉบับล่าสุด
- ☐ ส่วนประกอบหลักครบ อ่านข้อความหลักได้ และกดสำรวจหน้าจอได้
- ☐ มีไฟล์รูปโลโก้เก็บแยกไว้ ถ้าเลือกทำขั้นตอนเสริมนี้

เยี่ยมครับ! ตอนนี้เว็บแอปแรกไม่ได้อยู่แค่ในแชตแล้ว แต่มีชุดไฟล์อยู่บนเครื่องของคุณเองด้วย



ตอนนี้เรามีต้นแบบที่เห็นหน้าตาและกดเดินดูได้ พร้อมวิธีสั่งแก้จากสิ่งที่เห็นบนหน้าจอแล้วครับ ส่วนการเข้าสู่ระบบและบันทึกข้อมูลจริงยังเป็นงานในบทต่อ ๆ ไป

เมื่อ AI แก้จนงานมั่ว จะกลับมาเริ่มใหม่อย่างไร?

ก่อนจบบทนี้ ผมมีอีกเทคนิคหนึ่งที่อยากฝากไว้ครับ เป็นเรื่องที่ผมเจออยู่บ่อย ๆ เวลาทำงานกับ AI

สมมติว่าเรากำลังแก้หน้าตาเว็บแอปกันอย่างสนุกสนาน สั่งเพิ่มตรงนั้น ปรับตรงนี้ แล้วก็แก้อีกนิด...

จนเริ่มรู้สึกว่า เอ๊ะ ทำไมยิ่งแก้ยิ่งแย่?

สั่งเพิ่มปุ่มใหม่ แต่ปุ่มเก่าหาย สั่งแก้สี แต่หน้าจอกลับเปลี่ยนไปทั้งหน้า พอขอให้เอาของเดิมกลับมา ก็มีอย่างอื่นหายไปอีก

แก้ไปแก้มา จากที่ตอนแรกเกือบเสร็จ กลายเป็นไม่รู้จะเริ่มแก้ตรงไหนก่อนแล้วครับ 555+

ถ้าเจอสถานการณ์แบบนี้ **ไม่จำเป็นต้องฝืนแก้ต่อไปเรื่อย ๆ ครับ**

เราสามารถกลับไปเริ่มจากผลงานชุดที่ยังทำงานได้ดี แล้วค่อยเดินหน้าต่อจากตรงนั้นได้

นี่แหละครับเหตุผลที่ผมให้คุณเก็บ ZIP สำรองไว้ทุกครั้งที่ได้ผลงานชุดใหม่

ผมมีวิธีที่ใช้แก้สถานการณ์แบบนี้อยู่สองวิธีครับ

วิธีที่ 1 — กลับไปเริ่มจาก ZIP ชุดที่ดี ในแชตเดิม

วิธีนี้เหมาะกับกรณีที่แชตเดิมยังใช้งานได้ตามปกติ แต่การแก้ไขช่วงหลังเริ่มออกนอกทางไปแล้ว

เริ่มจากเปิดโฟลเดอร์สำรอง แล้วเลือก ZIP ชุดล่าสุดที่เรายังพอใจครับ

สมมติว่า `ละมุนเงิน-04-จัดกราฟ.zip` ยังทำงานได้ดี แต่หลังจากนั้นเราสั่งแก้อีกหลายรอบจนส่วนประกอบเริ่มหายไป

ให้กลับไปที่แชตเดิม แนบ ZIP ชุดที่ 04 แล้วบอก AI ว่าเราต้องการเริ่มใหม่จากไฟล์ชุดนี้

ลองใช้ข้อความประมาณนี้ครับ

> **PROMPT · คัดลอกไปใช้**
>
> ตอนนี้งานที่เราแก้ไขกันมาหลายรอบเริ่มมีปัญหา บางส่วนที่เคยทำงานได้กลับหายไป และการแก้ไขรอบล่าสุดก็ยังไม่ตรงตามที่ต้องการ
>
> ผมต้องการกลับไปเริ่มจาก ZIP ที่แนบมานี้แทน ให้ถือว่าไฟล์ชุดนี้เป็นต้นฉบับสำหรับทำงานต่อ ไม่ต้องนำการแก้ไขที่เกิดขึ้นหลังจากไฟล์ชุดนี้กลับมารวมเอง
>
> ในไฟล์ชุดนี้มีหน้าเข้าสู่ระบบจำลอง หน้าหลัก กราฟสรุป และฟอร์มเพิ่ม แก้ไข ลบรายการอยู่แล้ว โดยใช้ข้อมูลตัวอย่าง ยังไม่เชื่อมฐานข้อมูลจริง
>
> ช่วยตรวจดูไฟล์ที่แนบมาก่อน แล้วสรุปว่าตอนนี้มีส่วนประกอบอะไรอยู่บ้าง จากนั้นเราจะเริ่มแก้ไขงานต่อจากจุดนี้ครับ

สิ่งสำคัญคือ **ต้องบอกให้ชัดว่า ZIP ชุดไหนเป็นต้นฉบับที่เราจะใช้ต่อ และในชุดนั้นมีอะไรอยู่แล้วบ้าง**

ไม่ควรบอกเพียงว่า “เอากลับไปเหมือนเดิม” เพราะเมื่อแก้ไขกันมาหลายรอบ AI อาจไม่รู้ว่าเราหมายถึงงานช่วงไหน

เมื่อ AI ตรวจไฟล์และสรุปกลับมา ให้ดูว่าตรงกับชุดที่เราเลือกหรือไม่ ถ้าถูกต้องแล้วค่อยสั่งแก้ไขงานใหม่ทีละเรื่องครับ

วิธีที่ 2 — เปิดแชตใหม่ แล้วเริ่มต่อจาก ZIP ชุดที่ดี

วิธีนี้เป็นวิธีที่ผมใช้บ่อยครับ โดยเฉพาะเวลาคุยกับ AI มานานมากแล้ว มีทั้งคำสั่งเก่า คำสั่งใหม่ และการแก้ไขซ้อนกันหลายรอบ

บางครั้งการเปิดแชตใหม่ก็สะดวกกว่าการพยายามแก้ต่อในแชตเดิมครับ

แต่เราไม่ได้เริ่มสร้างเว็บแอปใหม่ตั้งแต่ศูนย์นะครับ เราจะนำงานที่ทำสำเร็จแล้วติดตัวไปด้วย

วิธีทำมีดังนี้

1. **เปิดแชตใหม่** กับ AI
2. **นำ Prompt แรกมาใส่** เพื่ออธิบายว่าเรากำลังสร้างเว็บแอปอะไร และต้องการให้ทำงานอย่างไร
3. **สรุปสิ่งที่แก้ไขเพิ่มเติม** จาก Prompt แรก ว่าตอนนี้เว็บแอปมีหน้าจอหรือฟังก์ชันอะไรเพิ่มขึ้นแล้วบ้าง
4. **แนบ ZIP ชุดที่ยังทำงานได้ดี** แล้วบอกให้ AI ใช้ไฟล์ชุดนี้เป็นจุดเริ่มต้นสำหรับทำงานต่อ

ลองดูตัวอย่างครับ

> **PROMPT · คัดลอกไปใช้**
>
> ผมกำลังสร้างเว็บแอปด้วย AI แต่แชตก่อนหน้าได้แก้ไขงานกันมาหลายรอบจนเริ่มมีปัญหา จึงต้องการย้ายมาทำงานต่อในแชตใหม่นี้
>
> ด้านล่างคือ Prompt แรกที่ผมใช้สร้างแอป เพื่อให้คุณเข้าใจเป้าหมายของงาน
>
> [วาง Prompt แรกทั้งหมดที่นี่]
>
> หลังจากสร้างต้นแบบครั้งแรก ผมได้ปรับแก้เพิ่มเติมจนมีหน้าเข้าสู่ระบบจำลอง หน้าหลักแสดงข้อมูล กราฟสรุป ฟอร์มเพิ่ม แก้ไข ลบรายการ และเปลี่ยนธีมสีตามที่ต้องการแล้ว
>
> ไฟล์ ZIP ที่แนบมาคือผลงานชุดที่ผมตรวจแล้วว่ายังใช้งานได้ดี ให้ถือว่าไฟล์นี้เป็นต้นฉบับปัจจุบันสำหรับทำงานต่อ ส่วน Prompt แรกใช้เพื่ออธิบายเป้าหมายของแอป ไม่ใช่ให้สร้างงานใหม่ทับไฟล์ที่แนบมา
>
> ช่วยตรวจไฟล์ ZIP แล้วสรุปให้ผมทราบก่อนว่าตอนนี้แอปมีอะไรอยู่บ้าง และมีส่วนไหนที่ยังเป็น Mockup จากนั้นเราจะเริ่มพัฒนาต่อจากไฟล์ชุดนี้ครับ

ตรง `[วาง Prompt แรกทั้งหมดที่นี่]` ให้นำ Prompt แรกของเว็บแอปคุณมาใส่จริง ๆ นะครับ ส่วนรายการหน้าจอและฟังก์ชันเพิ่มเติม ก็เปลี่ยนให้ตรงกับงานที่คุณทำมาแล้ว

วิธีนี้ช่วยให้ AI เริ่มทำความเข้าใจงานจากข้อมูลที่เราจัดเตรียมให้ใหม่ โดยไม่ต้องไล่ตามการแก้ไขที่สับสนในแชตเก่าทั้งหมด

แล้วควรเลือกวิธีไหนดี?

ถ้าเพิ่งแก้ผิดไปไม่กี่รอบ และแชตเดิมยังทำงานต่อได้ดี ใช้ **วิธีที่ 1** ก็เพียงพอครับ

แต่ถ้าแก้กันมานานจนเริ่มสับสนว่าอะไรเป็นอะไรแล้ว ผมมักเลือก **วิธีที่ 2** เปิดแชตใหม่ แล้วนำ Prompt แรกกับ ZIP ชุดที่ดีไปเริ่มต่อ

ไม่ว่าจะใช้วิธีไหน ให้จำหลักสำคัญสามข้อครับ

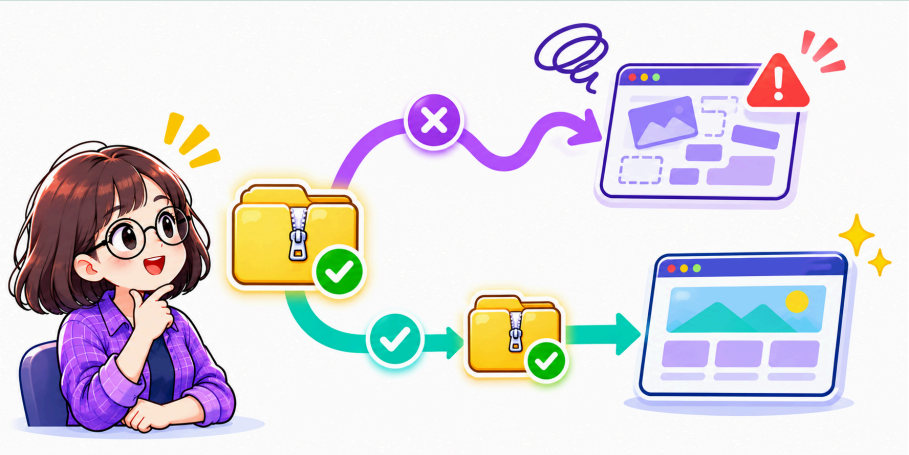
- **เลือก ZIP ที่ตรวจแล้วว่าดีจริง** ไม่ใช่เลือกเพียงเพราะเป็นไฟล์ใหม่ที่สุด
- **บอกให้ชัดว่าไฟล์นี้คือจุดเริ่มต้นปัจจุบัน** และอะไรเป็นเพียงข้อมูลประกอบจากงานเก่า
- **ให้ AI ตรวจและสรุปไฟล์ก่อนเริ่มแก้** เพื่อให้แน่ใจว่าเรากำลังทำงานจากจุดเดียวกัน

ถ้า AI เปิดอ่าน ZIP ไม่ได้ ให้แตกไฟล์แล้วส่งไฟล์ที่เกี่ยวข้องตามที่ระบบรองรับแทนครับ อย่าให้ AI เดาหรือสร้างงานขึ้นมาใหม่โดยยังไม่ได้ดูไฟล์ของเรา

และอย่าเขียนทับ ZIP ชุดที่ดีเด็ดขาดครับ เก็บไว้เป็นจุดสำรองเผื่อเราต้องกลับมาใช้อีกครั้ง

การกลับไปเริ่มจากจุดที่ดีกว่า ไม่ได้หมายความว่าเราต้องเริ่มทำทุกอย่างใหม่

เรากำลังเลือกจุดที่งานยังถูกต้อง แล้วเดินหน้าต่อจากตรงนั้นต่างหาก



วิธีนี้ใช้ได้ทั้งตอนสร้าง Mockup และตอนพัฒนาเว็บแอปจริงในบทต่อ ๆ ไปครับ เพียงแต่ภายหลังเราจะมีเครื่องมือช่วยจัดการเวอร์ชันของไฟล์ที่สะดวกขึ้น

สำหรับตอนนี้ แค่มี ZIP สำรองที่ดี และรู้วิธีบอก AI ให้เริ่มงานจากไฟล์นั้น คุณก็มีทางออกเวลา AI แก้จนงานยุ่งเหยิงแล้วครับ

เมื่อทำเป็นแล้ว เครื่องมือทั้งโลกก็เป็นของคุณ

ตลอดบทที่ผ่านมา เราใช้ ChatGPT เป็นเครื่องมือหลักในการสร้าง Mockup ครับ เพราะผมอยากให้ทุกคนเริ่มจากเครื่องมือเดียวกัน จะได้ทำตามขั้นตอนในหนังสือได้ง่าย

แต่บนโลกอินเทอร์เน็ตยังมี AI และเครื่องมืออีกมากมายให้เราเลือกใช้ครับ ไม่จำเป็นต้องยึดติดกับ ChatGPT เพียงตัวเดียว

ผมเคยได้ยินคนพูดกันว่า **DeepSeek สร้างเว็บแอปได้สวยมาก**

จริงหรือเปล่า? ผมก็อยากให้คุณลองไปพิสูจน์ด้วยตัวเองครับ!

ลองเปิด DeepSeek แล้วนำ Prompt แรกที่เราใช้สร้าง Mockup ในบทนี้ไปสั่งดูเลย ว่าจะได้หน้าตาเว็บแอปออกมาแบบไหน สวยถูกใจกว่าเดิมหรือเปล่า

เพราะตอนนี้ **คุณสร้าง Web App Mockup เป็นแล้วนี่ครับ!**

คุณรู้แล้วว่าต้องอธิบายความต้องการอย่างไร ต้องตรวจอะไรบ้าง และถ้าผลงานยังไม่ตรงใจก็รู้วิธีสั่งแก้ไขต่อ

จะใช้ AI ตัวไหนก็ลองได้ครับ

เพียงแต่ AI แต่ละตัวอาจมีความสามารถต่างกัน บางตัวแสดงตัวอย่างหน้าเว็บให้กดดูได้ บางตัวส่งเป็นโค้ดมาให้เรานำไปเปิดเอง หรือบางตัวอาจไม่มีระบบสร้าง ZIP ให้ดาวน์โหลด

ไม่จำเป็นต้องคาดหวังว่าทุกเครื่องมือจะทำงานเหมือน ChatGPT ทุกขั้นตอนครับ ขอแค่เราเข้าใจว่าเครื่องมือนั้นทำอะไรได้ และนำมาใช้ให้เหมาะกับงานของเรา

ไม่จำเป็นต้องใช้ AI ตัวเดียวทำทุกอย่าง

อย่างเรื่องโลโก้ที่เราลองสร้างกันก่อนหน้านี้ ก็เป็นอีกตัวอย่างหนึ่งครับ

ถ้าคุณรู้สึกว่าโลโก้ที่ได้จาก ChatGPT ยังไม่ถูกใจ จะลองใช้ AI สำหรับสร้างภาพโดยเฉพาะ หรือใช้เครื่องมือออกแบบโลโก้บนอินเทอร์เน็ตก็ได้

บางเครื่องมืออาจสร้างภาพได้ตรงสไตล์ที่คุณชอบมากกว่า บางเครื่องมืออาจเหมาะกับการปรับสี จัดรูปทรง หรือใส่ข้อความเพิ่มเติม

เมื่อได้โลโก้ที่ถูกใจแล้ว ก็บันทึกเป็นไฟล์รูปไว้ เพื่อนำไปใช้กับเว็บแอปของเราภายหลังครับ

เราไม่จำเป็นต้องใช้เครื่องมือเดียวตั้งแต่เริ่มจนจบ จะใช้ ChatGPT สร้าง Mockup แล้วใช้เครื่องมืออื่นทำโลโก้ก็ได้ หรือจะใช้ AI ตัวอื่นสร้าง Mockup ตั้งแต่แรกก็ได้เช่นกัน

เครื่องมือมีไว้ให้เราเลือกใช้ตามความเหมาะสมของงานครับ



สิ่งที่ติดตัวเราไป ไม่ใช่ชื่อเครื่องมือ

สิ่งที่ผมอยากให้คุณได้จากบทนี้ ไม่ใช่แค่การจำว่าต้องกดปุ่มไหนใน ChatGPT ครับ

แต่เป็นความเข้าใจว่าเรากำลังทำอะไร ต้องสั่ง AI อย่างไร และจะตรวจผลงานได้อย่างไรว่าตรงกับที่ต้องการ

นี่คือหัวใจของ **Concept** ที่ผมใช้เป็นหลักในการสอนตลอดทั้งเล่มครับ

เพราะเมื่อเข้าใจหลักการทำงานแล้ว ต่อให้เปลี่ยน AI เปลี่ยนโปรแกรม หรือหน้าตาของเครื่องมือไม่เหมือนในหนังสือ เราก็ยังรู้ว่าต้องทำอะไรต่อ

วันนี้คุณอาจใช้ ChatGPT สร้างเว็บแอปแรก วันหน้าอาจพบเครื่องมือใหม่ที่ถูกใจกว่า ก็สามารถนำความรู้ที่ได้ไปลองใช้ต่อได้ครับ

ส่วนใครอยากใช้ ChatGPT ตามหนังสือไปจนจบก่อน ก็ทำได้เลย ไม่จำเป็นต้องเปลี่ยนเครื่องมือระหว่างทาง

เอาละครับ ตอนนี้เรามีต้นแบบเว็บแอปของตัวเองแล้ว

บทต่อไป เราจะย้ายจากการสร้าง Mockup ในแชต มาเตรียมพื้นที่ทำงานบนคอมพิวเตอร์กัน และค่อย ๆ เปลี่ยนต้นแบบนี้ให้กลายเป็นเว็บแอปที่ใช้งานได้จริง

ไปต่อกันเลยครับ!