

ก้าวสู่ยุค No-Code สร้าง App และ Web โดย “ไม่ต้องเขียนโปรแกรม”



ดร.จักรกฤษณ์ สิริริน

ผู้อำนวยการสถาบันการศึกษา ฝึกอบรม
และให้คำปรึกษาสถานประกอบการ
สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น)

นั่ง Time Machine ย้อนกลับไปในปี ค.ศ. 1980 จุดเริ่มต้นสำคัญของ 4GL (Fourth-Generation Language)

ที่มีการตั้งเป้าหมายเอาไว้ว่า การเขียนโปรแกรม นับจากนี้ จะมีการลดทอนเรื่องการ Coding ให้น้อยลงเรื่อย ๆ สำหรับการพัฒนาระบบงานคอมพิวเตอร์

แน่นอนว่า การเขียน Code ที่น้อยลง ย่อมทำให้ระยะเวลาในการสร้างระบบงานสั้นลงตามไปด้วย โดยเฉพาะอย่างยิ่ง เป็นการส่งเสริมให้ผู้ที่ไม่มีความรู้ หรือประสบการณ์ด้าน Programming สามารถสร้างระบบได้ด้วยตนเอง

ซึ่งถือเป็นจุดเริ่มต้นที่ยิ่งใหญ่ **ภายใต้แนวคิด No-Code Development Platform ที่ทุก ๆ คนไม่จำเป็นต้องมีความรู้เรื่องเทคนิคด้านการเขียนโปรแกรม แต่สามารถสร้าง Web และ App ด้วยตัวเองได้**

กำจัดจุดอ่อน และแก้ Pain Point ของคนที่ไม่ได้จบคอมฯ ที่ต้องการสร้าง Web และ App ของตัวเองแบบไม่ต้องไปจ้างใคร

แนวคิด No-Code จึงเกิดขึ้น เปลี่ยนมุมมอง และการปฏิบัติ จากการพัฒนาระบบงานคอมพิวเตอร์ผ่านการเขียนโปรแกรม หรือ Coding มาเป็นการใช้ Graphical User Interface (GUI) เพียงแค่ลาก “ปุ่ม” คลิก เลื่อน วาง

โดยมี Template และคำสั่ง หรือ Library รวมถึงเครื่องมือ เครื่องมือ Plugin ต่าง ๆ ที่เตรียมมาไว้ให้พร้อมสรรพ

No-Code Platform จึงเป็นแนวคิดที่เป็นมิตรกับคนทำธุรกิจ มากกว่า Developer หรือโปรแกรมเมอร์

เพราะ No-Code เป็น Platform ที่ใช้งานง่าย เหมาะสำหรับทุกคน โดยเฉพาะอย่างยิ่ง ในทางกลับกัน เหล่า Developer ยิ่งชอบใจ เพราะประหยัดเวลา และแรงกายแรงใจไปได้มาก

เนื่องจาก Developer มี Logic ในการเขียนโปรแกรมที่แน่นอยู่แล้ว เมื่อมาเจอเครื่องมือแบบนี้นี่ยิ่งดีใจกันยกใหญ่

ในส่วนของบุคคลทั่วไปที่ไม่ได้จบคอมฯ ผมคิดว่า No-Code Platform อาจต้องอาศัย “ทักษะทางศิลปะ” เป็นตัวนำมากสักหน่อย

เพราะ No-Code นั้น ผู้ใช้งานไม่จำเป็นต้องมีความรู้เกี่ยวกับ Syntax ภาษาคอมพิวเตอร์อะไรเลย

ทว่า ต้องมีทักษะด้านการปรับแต่ง User Interface (UI) หรือเลือก Layout สำเร็จรูป ให้มีรูปร่างหน้าตาที่สวยงาม และใช้งานง่าย ๆ เท่านั้น

นั่ง Time Machine เดินหน้ามาในปี ค.ศ. 2014 ที่ Forrester Research ได้ให้คำจำกัดความของคำว่า No-Code ว่าเป็นเครื่องมือสำคัญอันหนึ่ง ซึ่งนำมาใช้ในการพัฒนาระบบงานคอมพิวเตอร์ที่มุ่งเน้นให้ผู้ที่ไม่มีความรู้ด้านการเขียนโปรแกรม สามารถสร้างสรรค์ Web และ App ใหม่ ๆ ได้ด้วยตนเอง

ลองไปดูความแตกต่างระหว่างการ Coding แบบเดิม กับ แนวคิด No-Code และ Low-Code กันสักเล็กน้อย

การ Coding รูปแบบเก่า หรือการพัฒนากระบวนการคอมพิวเตอร์ดั้งเดิม (Traditional Coding) คือการสร้างระบบโดยการเขียน Code โดยโปรแกรมเมอร์ทั้งหมด

ส่วน Low-Code คือการเขียนโปรแกรมผ่าน Visual Modelling ในการสร้างระบบโดยลดการเขียน Code ลง

และ No-Code คือการสร้างระบบงานคอมพิวเตอร์ เช่น Web และ App โดยไม่มีการเขียน Code เลยแบบ 100%



ข้อดีของ No-Code Platform

1. รวดเร็ว สามารถสร้าง Solution หลาย Platform ในคราวเดียว ทำ Prototype ให้ Stakeholder ทดสอบได้อย่างรวดเร็ว และสมบูรณ์แบบภายในระยะเวลาอันสั้น

2. ประหยัด สามารถลดเวลาในการดำเนินโครงการ หรือ Project โดยเฉพาะอย่างยิ่งโครงการใหญ่ ๆ โดยไม่ต้องลองผิดลองถูก หรือเข้าไปในขั้นตอนของจริงต่าง ๆ ด้วยตัวเอง แต่ผ่าน No-Code Prototype จัดการกับไอเดียใหม่ ๆ โดยไม่จำเป็นต้องรอให้โครงการสำเร็จ นอกจากประหยัดเวลาแล้ว ยังประหยัดทรัพยากรขององค์กร ด้วยต้นทุนที่ต่ำกว่า เมื่อเทียบกับวิธีการ Coding แบบเดิม

3. ความเสี่ยงต่ำ และ ROI สูง เนื่องจาก No-Code ได้รับการสนับสนุนด้านมาตรฐานอุตสาหกรรมจึงมีความปลอดภัยสูง และมีการพัฒนาอยู่ตลอดเวลา ดังนั้น เราจึงสามารถมุ่งความสนใจไปที่การสร้างรายได้ของธุรกิจที่สำคัญกว่าได้ เพราะ Workflow ทั้งหมดของ No-Code ได้รับการปกป้องเป็นอย่างดีนั่นเอง

4. ประยุกต์ใช้งานง่าย การเผชิญหน้ากับปัญหาการพัฒนากระบวนการคอมพิวเตอร์โดยเฉพาะอย่างยิ่ง Bug ที่ No-Code Platform เปิดให้เราสามารถตั้งค่าการเปลี่ยนแปลงในเวอร์ชัน

ปัจจุบัน กลับไปเป็นเวอร์ชันก่อนหน้าที่เราได้ด้วยการคลิกเพียงครั้งเดียว ทำให้เราแก้ไขปัญหาได้ตั้งแต่เนิ่น ๆ

5. การติดตั้งที่สะดวก และการ Support ในระยะยาว

ที่ดี Platform LC/NC (No-Code/Low-Code) สนับสนุนวงจรการพัฒนากระบวนการคอมพิวเตอร์ (System Development Life Cycle: SDLC) ตั้งแต่เริ่มต้นจนถึงสิ้นสุด แน่นอนว่า ย่อมรวมถึง DevOps และ CI/CD นอกจากนี้ API ของ LC/NC ยังช่วยประสานงานกับเครื่องมือภายนอก เพื่อสนับสนุนการบริหารโครงการ, DevOps, การทดสอบ และ CICD Pipeline อีกด้วย

ส่วนข้อเสียของ No-Code Platform

1. ข้อจำกัดระยะยาว ยิ่งธุรกิจมีความซับซ้อนมากขึ้นจากเวลาที่เปลี่ยนแปลงไปเรื่อย ๆ ในระยะยาว การใช้ LC/NC อาจไม่ตอบโจทย์ เพราะเราไม่สามารถปรับระบบให้เหมาะกับเป้าหมายทางธุรกิจได้

2. ฟังก์ชันที่จำกัด การใช้ LC/NC Platform จะมาพร้อมกับจำนวนฟังก์ชันที่จำกัด เนื่องจากเราได้เลือกแล้วว่า ต้องการพัฒนาระบบงานคอมพิวเตอร์ในเวลาอันรวดเร็ว ดังนั้นอาจจะต้องแลกมาด้วยฟังก์ชันที่จำกัดนั่นเอง โดยหากเราอยากเพิ่มเติมส่วนของ Code ที่เราต้องการจะกำหนดเอง

อาจจะมีค่าใช้จ่ายที่แพงกว่าระบบที่เราสร้างเองขึ้นมาใหม่ทั้งหมด

3. ข้อจำกัดด้านความปลอดภัย Solution แบบ LC/NC Platform แม้จะมีความปลอดภัยในตัวของมันเอง แต่ก็ยังไม่สามารถให้คำยืนยันเกี่ยวกับระดับของความปลอดภัย เมื่อเทียบกับเทคโนโลยีการพัฒนาแบบ Stand Alone เนื่องจากเราไม่สามารถควบคุมความปลอดภัยของข้อมูล หรือการเข้าถึง Source Code ได้อย่างเต็มที่ ดังนั้น เราจึงไม่สามารถค้นพบช่องโหว่ในโปรแกรมได้เลย

ทั้งนี้ Gartner มีการคาดการณ์ว่า ภายในปี ค.ศ.2024 แนวคิดการนำ No-Code มาใช้ในการพัฒนา App หรือ Web มีความเป็นไปได้สูงถึง 65%

ดังนั้น การก้าวเข้าสู่ยุคทองของ No-Code ที่เรา ๆ ท่าน ๆ สามารถสร้าง Application Software และ Website ได้เองโดยที่ “ไม่ต้องเขียนโปรแกรม”

คืออนาคตของการสร้างสรรค์ระบบที่เราควรให้ความสำคัญในการเรียนรู้ เพื่อให้เท่าทันกับการเปลี่ยนแปลงที่กำลังเกิดขึ้นนั่นเองครับ