
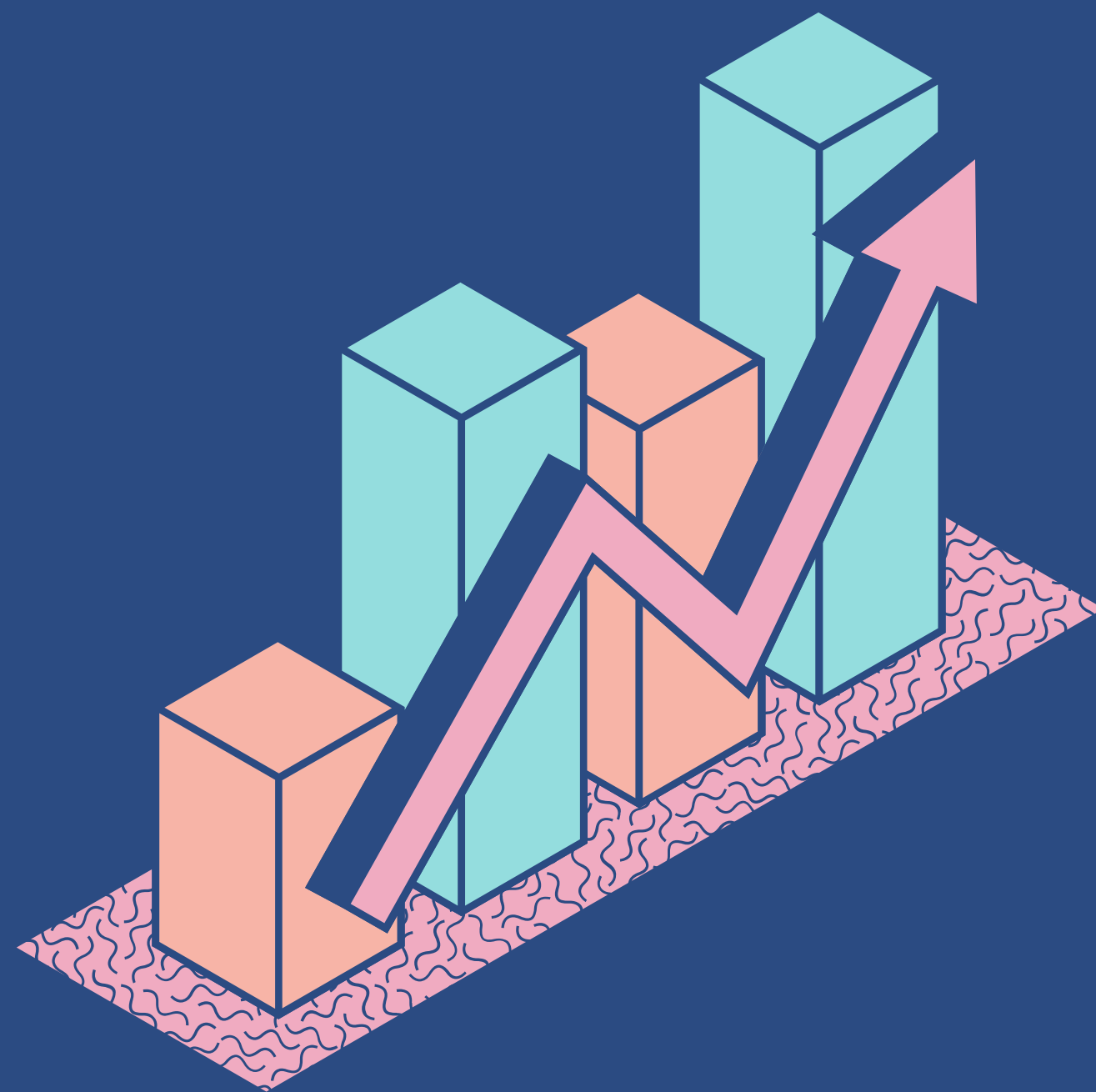


- 
- **crossplatform**
 - **hybrid Application**
 - **native app**

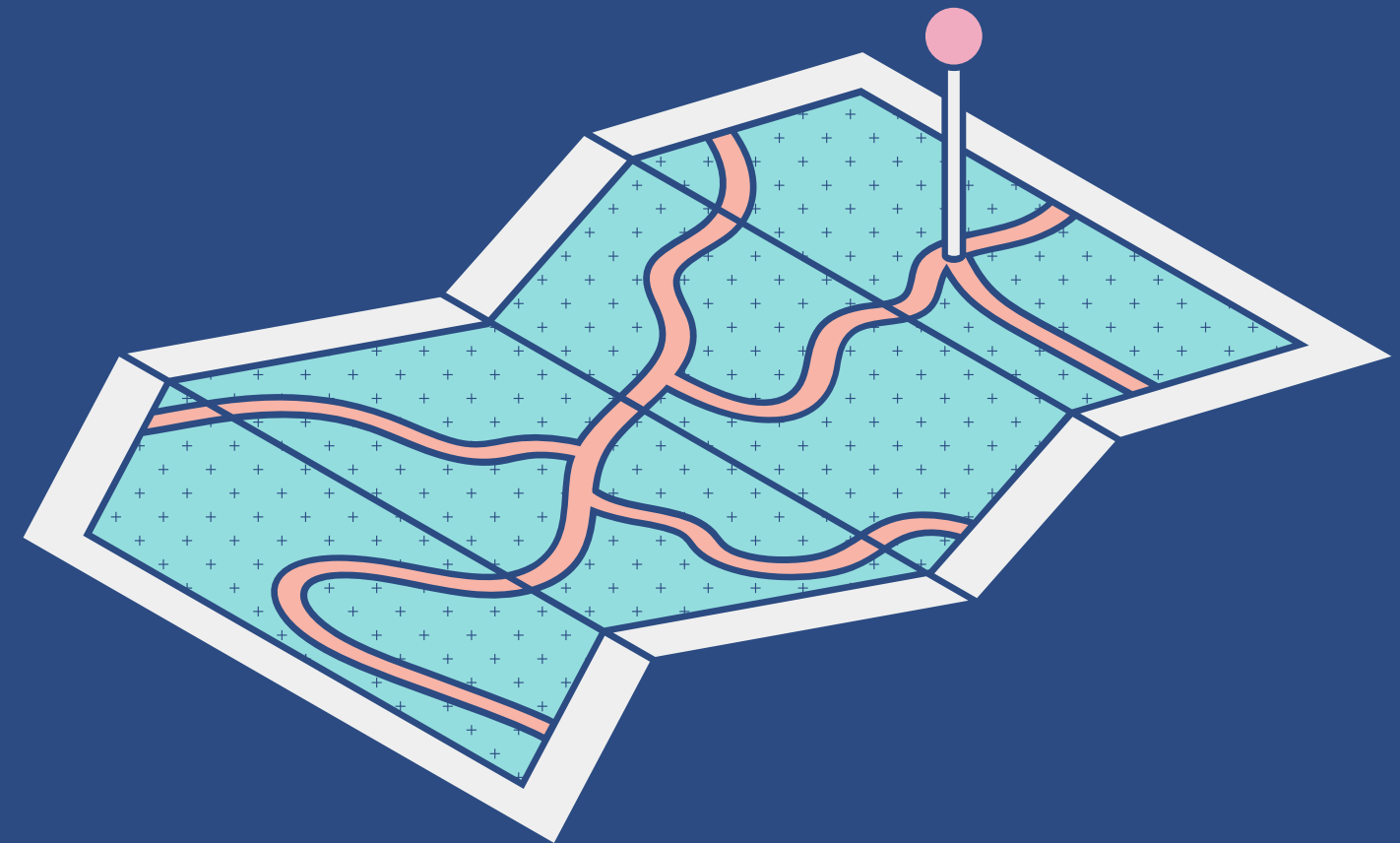
crossplatform

Cross platform application คือแอปพลิเคชัน, โปรแกรม, หรือซอฟต์แวร์ที่ออกแบบมาเพื่อทำงานบนระบบปฏิบัติการหรือแพลตฟอร์มหลายระบบ เช่น ระบบปฏิบัติการที่แตกต่างกัน (เช่น Windows, macOS, Linux) หรือมือถือต่างๆ (เช่น iOS, Android) โดยแอปฯ ชนิดนี้ได้รับการพัฒนาให้สามารถใช้งานบนอุปกรณ์และระบบปฏิบัติการต่างๆ ได้โดยไม่จำเป็นต้องแก้ไขเพิ่มเติมหรือสร้างแอปฯ สำหรับแต่ละแพลตฟอร์มขึ้นมาต่างหากแต่อย่างใด



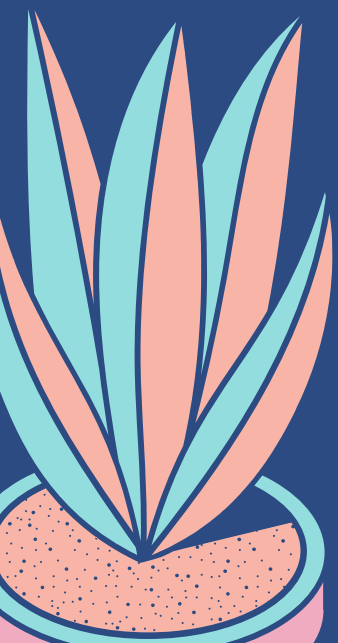
เครื่องมือเขียนและสร้างแอปฯ แบบ cross platform application

1. React Native
2. Flutter
3. .NET MAUI
4. Kotlin Multiplatform



• จุดเด่น

ประหยัดต้นทุน: การพัฒนาแอปฯ แบบ cross-platform ช่วยให้ธุรกิจสามารถสร้างแอปพลิเคชันสำหรับหลายแพลตฟอร์มโดยใช้โค้ดเพียงชุดเดียว ซึ่งช่วยลดต้นทุนในการพัฒนาได้เยอะมาก เนื่องจาก developer ไม่จำเป็นต้องมาเขียนและคอยดูแลโค้ดสำหรับแต่ละแพลตฟอร์มนั่นเอง



• จุดด้อย

แอปแบบ cross-platform นั้นไม่สามารถใช้ประโยชน์ได้โดยตรงจาก native component หรือ component ที่สามารถใช้งานบน native app ได้โดยตรง จึงไม่สามารถที่จะทำให้แอปนั้นทำให้ ผู้ใช้นั้นมีประสบการณ์ที่เหมือนกันได้ บนสองแพลตฟอร์ม การตัดสินใจเพื่อเลือกการพัฒนาที่เหมาะสม

Hybrid Application

Hybrid Application คือแอปพลิเคชันมือถือที่มีการพัฒนาารองรับแพลตฟอร์ม Android, iOS และ Windows ซึ่งไม่เหมือนกับ Native Application ที่ถูกพัฒนาภาษาเฉพาะให้รองรับระบบปฏิบัติการที่แตกต่างกันเพื่อให้งานได้ทุกแพลตฟอร์ม ตัวอย่างภาษาที่ใช้พัฒนา Native Application ที่ใช้พัฒนาระบบปฏิบัติการ iOS คือภาษา Swift หรือ Objective-C ส่วนระบบปฏิบัติการ Android นั้นจะใช้ภาษา Kotlin หรือ Java ในการพัฒนา ซึ่งต่างกับ Hybrid Application ที่จะใช้ HTML, JavaScript, CSS, Dart และอื่น ๆ



เครื่องมือเขียนและสร้างแอปฯ แบบ Hybrid

- Ionic

คุณสมบัติ: ใช้ HTML, CSS และ JavaScript ในการพัฒนา

ประโยชน์: มีประสิทธิภาพสูง, สามารถทำให้แอปมีความรวดเร็ว และมีพื้นฐานของการออกแบบที่เรียบง่าย

- React Native

คุณสมบัติ: พัฒนาด้วย JavaScript และใช้ส่วนประกอบแบบ Native

ประโยชน์: ประสิทธิภาพสูง, สามารถเข้าถึงฟีเจอร์ของมือถือได้เต็มที่ และมีความยืดหยุ่นในการพัฒนา

- Flutter

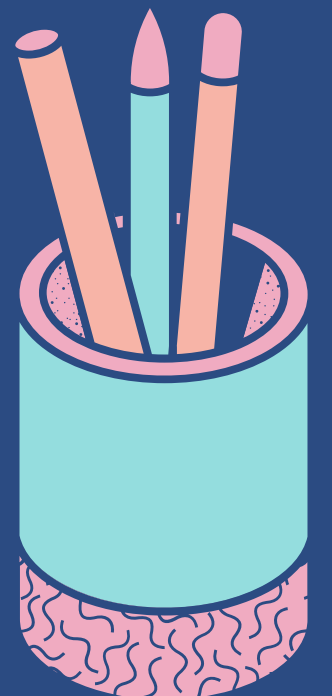
คุณสมบัติ: ใช้ภาษา Dart ในการพัฒนา

ประโยชน์: มีความเร็วในการทำงาน, ออกแบบ UI ที่สวยงามและยืดหยุ่น และสามารถใช้ได้ทั้ง iOS และ Android

PhoneGap/Cordova

คุณสมบัติ: ใช้ HTML, CSS, และ JavaScript

ประโยชน์: สามารถเข้าถึง API ของมือถือได้ง่าย และมีปลั๊กอินหลายตัวที่ช่วยเสริมฟังก์ชัน

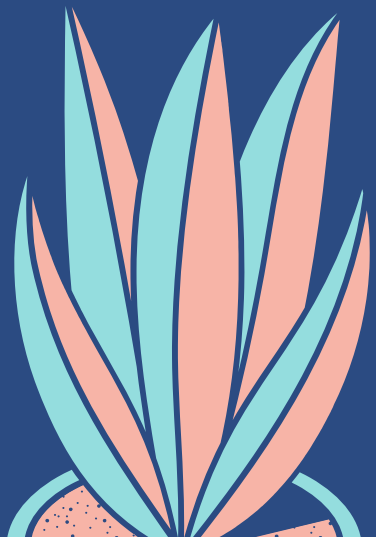


• จุดเด่น

พัฒนาด้วยภาษา HTML, CSS และ JavaScript ทำให้ง่ายและเรียนรู้ได้อย่างรวดเร็ว

พัฒนาครั้งเดียวสามารถใช้ได้หลาย Platform ทั้ง iOS, Android และ Window Phone

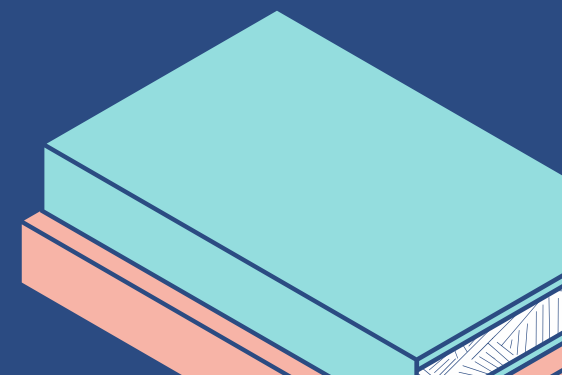
ใช้ต้นทุนในการพัฒนาน้อยกว่า Native App



• จุดด้อย

ประสิทธิภาพการทำงานจะด้อยกว่า Native App

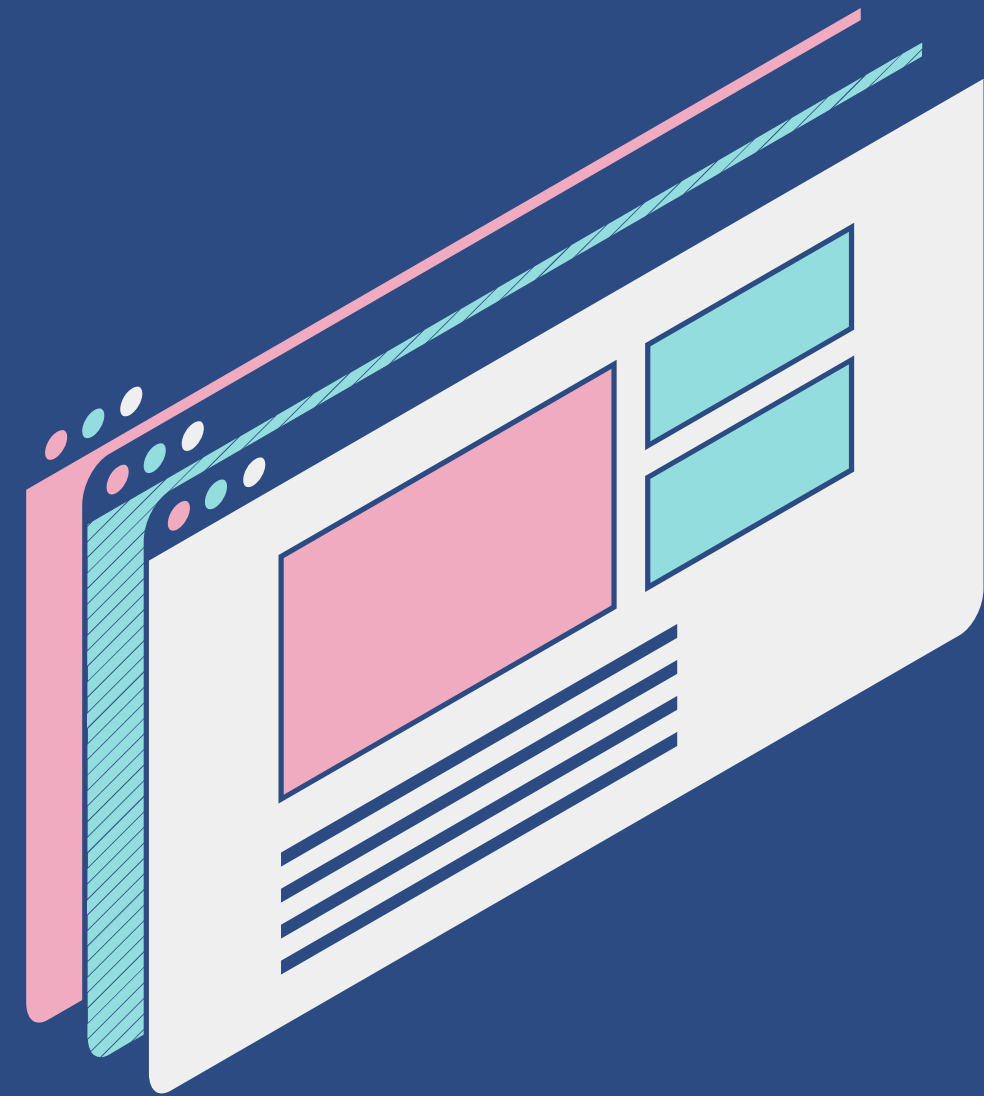
ในบางกรณีอาจจะใช้ความสามารถของอุปกรณ์ได้ไม่เต็มที่ เนื่องจากต้องขึ้นอยู่กับ Framework ที่เลือกในการพัฒนานั้นมี Component ที่ต้องการหรือไม่ ดังนั้น Hybrid App จึงมีจุดเด่นในเรื่องความง่ายและพัฒนาได้รวดเร็ว



Native app

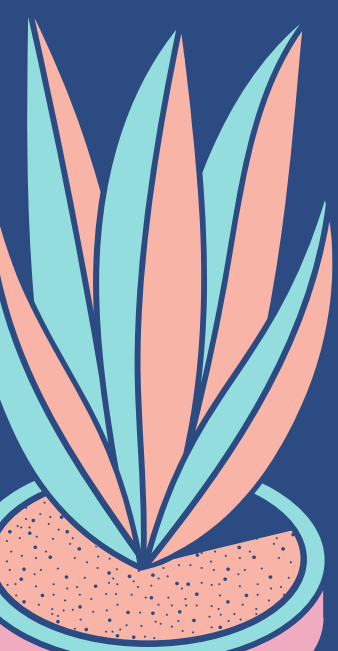
Native App คือ การพัฒนาแอปพลิเคชันที่ใช้รูปแบบการพัฒนาและชุดคำสั่งต่าง ๆ ตามที่ผู้พัฒนาอุปกรณ์ได้จัดทำขึ้น เช่น

- **iOS สำหรับ iPhone, iPad, Apple Watch จะใช้ภาษา Object C หรือ Swift โดยการพัฒนาจะต้องใช้โปรแกรม XCode**
- **Android จะใช้ภาษา Java และใช้โปรแกรม Android Studio ในการพัฒนา**
- **Window Phone ใช้ภาษา C# และใช้โปรแกรม Visual Studio ในการพัฒนา**



• จุดเด่น

- สามารถใช้งานชุดคำสั่งต่าง ๆ สำหรับ Platform นั้น ๆ ได้อย่างครบถ้วน
- สามารถใช้งานความสามารถของอุปกรณ์ ได้เต็มประสิทธิภาพ เช่นการเรียกใช้งาน กล้อง เบิรมติศ เป็นต้น
- สามารถนำขึ้นให้ผู้ใช่โหลดจาก AppStore หรือ PlayStore ได้



• จุดด้อย

- ต้องพัฒนาแยกต่างแต่ละ Platform คือ ถ้าต้องการมี App บน iOS, Android, Window Phone จะต้องพัฒนาแยกกัน
- มีต้นทุนในการพัฒนาสูง เนื่องจากต้องใช้ จำนวนคนและระยะเวลาในการพัฒนามาก



เครื่องมือเขียนและสร้างแอปฯ แบบ Native app

Native App

