สถาบันไอทีจีเนียส
IT GENIUS INSTITUTE

COURSE ID
MBC-19
MOBILE / REACT NATIVE

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

Advanced React Native CLI

ต่อยอดสาย React Native CLI สู่โค้ดเนทีฟ ประสิทธิภาพ และการส่งแอปขึ้น Store แบบอัตโนมัติ

- 18 ชั่วโมง
- 3 วัน
- พื้นฐาน **สร้างแอป React Native CLI ได้แล้ว**
- ระดับ **Advanced**



ระยะเวลา: 3 วัน
18 ชั่วโมง (3 วัน วันละ 6 ชม.) · 09:00-16:00 น.

รูปแบบ: Classroom / Hybrid
Onsite หรือ Online ผ่าน Microsoft Teams · Public / In-house

รายละเอียดหลักสูตร
www.itgenius.co.th

แอป React Native ที่ใช้งานจริงในองค์กรมักต้องทำมากกว่าหน้าจอและการเรียก API เช่น เรียกความสามารถของเครื่องที่ยังไม่มีไลบรารีรองรับ ห่อ View เนทีฟเดิมของทีม Android และ iOS มาใช้ในแอป แก้ปัญหาแอปช้าหรือเปิดนาน ทำงานได้แม้ไม่มีอินเทอร์เน็ต รับ Push Notification เปิดแอปจากลิงก์ และออกเวอร์ชันใหม่ขึ้น Store อย่างสม่ำเสมอโดยไม่ต้อง Build ด้วยมือ งานเหล่านี้ต้องเข้าใจทั้งฝั่ง JavaScript และโปรเจกต์ Android กับ iOS ที่อยู่เบื้องหลัง

หลักสูตรนี้เป็นฉบับปรับปรุงใหม่ที่ต่อยอดสาย React Native CLI แบบไม่ใช้ Expo บน React Native 0.87 ซึ่งทำงานบน New Architecture เท่านั้น พาผู้เรียนเขียน Turbo Native Module และ Fabric Native Component ผ่าน Codegen วัดและปรับประสิทธิภาพด้วย Hermes V1 และเครื่องมือ Profiler ทำข้อมูลแบบ Offline-first ทำ Push Notification ผ่าน Firebase Cloud Messaging และ Deep Linking เขียนชุดทดสอบ แล้วสร้าง Signed Release และ Pipeline ส่งแอปขึ้น Google Play และ App Store ด้วย Gradle, Xcode และ fastlane (อบรม 3 วัน วันละ 6 ชั่วโมง รวม 18 ชั่วโมง ระดับ Advanced)

เครื่องมือ

- RN React Native 0.87
- Kt Kotlin และ Swift
- H Hermes V1
- FCM Firebase Messaging
- fl fastlane

ราคาค่าอบรม · PUBLIC / IN-HOUSE

8,500 บาท / ท่าน

สมัครผ่านเว็บไซต์ ลด 10%

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม Lab Guide ซอร์สโค้ดตัวอย่างครบทุก Lab ไฟล์ตั้งค่า fastlane และ CI ตัวอย่าง และ Certificate · รับจัด In-house ขอใบเสนอราคาได้

เพิ่มเพื่อน LINE

02-570-8449 · 088-807-9770 · @itgenius · www.itgenius.co.th

01 วัตถุประสงค์ Objectives

- อธิบาย New Architecture ได้แก่ JSI, Turbo Native Modules, Fabric และ Codegen และอัปเกรดโปรเจกต์ได้
- เขียน Turbo Native Module และ Fabric Native Component ด้วย Kotlin และ Objective-C++ หรือ Swift ได้
- วัดและปรับประสิทธิภาพแอปด้วย React Native DevTools, Android Studio Profiler และ Instruments ได้
- ทำแอปแบบ Offline-first, Push Notification และ Deep Linking บนโปรเจกต์ React Native CLI ได้
- เขียน Unit Test และ E2E Test และตั้งค่า Build Variant แยกตาม Environment ได้
- สร้าง Signed Release และ Pipeline ส่งแอปขึ้น Google Play และ App Store ด้วย fastlane ได้

02 กลุ่มเป้าหมาย Target Audience

- นักพัฒนา React Native ที่สร้างแอปด้วย React Native CLI ได้แล้วและต้องการทำแอประดับ Production
- ทีมที่ต้องเชื่อมแอป React Native กับโค้ด Kotlin หรือ Swift ที่มีอยู่เดิม
- นักพัฒนา Android หรือ iOS ที่ทำงานร่วมกับทีม React Native
- ทีมที่ต้องการระบบ Build และส่งแอปขึ้น Store แบบอัตโนมัติ

03 จุดเด่นของหลักสูตร Highlights

- **สาย React Native CLI** ทำงานกับโปรเจกต์เนทีฟโดยตรง
- **เขียนโค้ดเนทีฟเองได้** Turbo Native Modules และ Fabric
- **อัปเดตเป็น React Native 0.87** New Architecture และ Hermes V1
- **ฟีเจอร์ของแอปจริง** Offline-first, Push และ Deep Link
- **ออกเวอร์ชันได้อัตโนมัติ** Signed Release, CI และ fastlane

04 ความรู้พื้นฐาน Prerequisites

- สร้างแอปด้วย React Native CLI ได้ ทั้ง Navigation, State และการเรียก API
- เขียน TypeScript ได้คล่อง
- เคยเปิดโปรเจกต์ใน Android Studio หรือ Xcode และอ่านโค้ด Kotlin หรือ Swift เบื้องต้นได้
- ใช้ Git และ Command Line ได้

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
อุปกรณ์	Notebook แรมแนะนำ 16 GB ขึ้นไป macOS จำเป็นสำหรับ Lab ฝั่ง iOS และมือถือ Android หรือ iPhone พร้อมสาย USB
ซอฟต์แวร์	Node.js 24 LTS, JDK 17, Android Studio พร้อม Android SDK, Xcode และ CocoaPods (เฉพาะ macOS), Ruby สำหรับ fastlane และ VS Code
เฟรมเวิร์ก	React Native 0.87, TypeScript, React Navigation 7 และ create-react-native-library
ไลบรารี	react-native-mmkv, op-sqlite, NetInfo, React Native Firebase Messaging, react-native-permissions, Detox และ React Native Testing Library
บริการ	Firebase, Google Play Console, Apple Developer และ Sentry ใช้บัญชีของผู้เรียน

หมายเหตุ: ปรับปรุงใหม่ปี 2026 จากหลักสูตร Advanced React Native เดิม: เปลี่ยนมาใช้ React Native 0.87 ซึ่งทำงานบน New Architecture เท่านั้นและใช้ Hermes V1 เป็นค่าเริ่มต้น ตัด Facebook Login, OTP ผ่าน Cloud Functions และ State Management ด้วย Context ที่สอนไปแล้วในหลักสูตรก่อนหน้า เพิ่ม Turbo Native Modules, Fabric Native Components และ Codegen เปลี่ยน Offline Storage จาก AsyncStorage มาเป็น MMKV และ SQLite ใช้ React Native Firebase รุ่นที่รองรับ New Architecture และเพิ่ม Testing, CI และ fastlane หลักสูตรนี้ต่างจาก Advanced React Native with Expo ที่ใช้ Expo, EAS และ Expo Modules API เพราะเป็นสาย React Native CLI ที่ต่อจาก Basic React Native for React Developers ทำงานกับโปรเจกต์ Android และ iOS โดยตรง เขียนโค้ดเนทีฟเอง และ Build ด้วย Gradle, Xcode และ fastlane โดยไม่ใช้ Expo Lab ฝั่ง iOS ต้องใช้ macOS และ Xcode การทดสอบ Push Notification ต้องใช้มือถือจริง การส่งแอปขึ้น Store ต้องมีบัญชี Apple Developer และ Play Console ของผู้เรียน Firebase และ Sentry ใช้บัญชีของผู้เรียนซึ่งมีโควตาแบบฟรีจำกัด React Native ออกรุ่นใหม่หลายครั้งต่อปี เนื้อหาอิงเวอร์ชัน stable ล่าสุด ณ วันอบรม

06 ภาพรวมการเรียน 3 วัน Big Picture

วัน	หัวข้อหลัก
Day 1 Section 1-5	New Architecture เชิงลึก, เขียน Turbo Native Module ด้วย Kotlin และ Objective-C++ ผ่าน Codegen, ส่ง Event และ Promise กลับ JavaScript, สร้าง Fabric Native Component และแพ็กโมดูลเป็นไลบรารี
Day 2 Section 6-10	วัดและปรับ Performance ด้วย Hermes V1 และเครื่องมือ Profiler, ข้อมูลแบบ Offline-first ด้วย MMKV และ SQLite, Push Notification ผ่าน FCM และ APNs, Local Notification และ Deep Linking
Day 3 Section 11-15	Unit Test และ E2E ด้วย Detox, Build Variant และ Environment, Signed Release สำหรับ Google Play และ App Store, CI และ fastlane, ติดตาม Crash แล้วปิดท้ายด้วย Workshop แอปพร้อมขึ้น Store

DAY 1

New Architecture และโค้ดเนทีฟ

เนื้อหาการอบรม · Section 1-5

รวม
6 ชั่วโมง

1 Lab: New Architecture เชิงลึกและการอัปเกรดโปรเจกต์

- JSI, Turbo Native Modules, Fabric และ Codegen ทำงานร่วมกันอย่างไร
- ตั้งแต่ 0.82 ทำงานบน New Architecture เท่านั้น และผลต่อไลบรารีเดิม
- Strict TypeScript API ของ 0.87 และการเลิกใช้ Deep Import
- อัปเกรดเวอร์ชันด้วย React Native Upgrade Helper อย่างเป็นขั้นตอน
- Lab: อัปเกรดโปรเจกต์ตัวอย่างและแก้ไลบรารีที่ไม่รองรับ

2 Lab: Turbo Native Module ตัวแรก

- เขียน Spec ใน TypeScript ด้วยชื่อขึ้นต้น Native และ TurboModuleRegistry
- ตั้งค่า codegenConfig ใน package.json และรัน Codegen
- Android: เขียนคลาส Kotlin ที่สืบทอด Spec ที่ Codegen สร้างให้
- iOS: เขียนด้วย Objective-C++ และลงทะเบียนผ่าน modulesProvider
- Lab: โมดูลอ่านข้อมูลอุปกรณ์และพื้นที่เก็บข้อมูลของเครื่อง

3 Lab: Native Module ขั้นสูง

- Promise, Callback และการส่ง Error จากโค้ดเนทีฟ
- ส่ง Event จากเนทีฟกลับมา JavaScript
- งานหนักบน Background Thread โดยไม่บล็อก UI
- ภาพรวม Turbo Module แบบ C++ ที่ใช้โค้ดเดียวกันทั้งสองแพลตฟอร์ม
- Lab: โมดูลติดตามสถานะแบตเตอรี่ที่ส่ง Event แบบเรียลไทม์

4 Lab: Fabric Native Component

- Spec ด้วย codegenNativeComponent และไฟล์ชื่อลงท้าย NativeComponent
- Android: SimpleViewManager และ Interface ที่ Codegen สร้างให้
- iOS: RCTViewComponentView และการอัปเดต Props
- ส่ง Event จาก View เนทีฟกลับมา JavaScript
- Lab: ห่อ View เนทีฟที่ทีมมีอยู่มาใช้ใน React Native

5 Lab: แพ็กโมดูลเนทีฟเป็นไลบรารี

- สร้างโครงไลบรารีด้วย create-react-native-library
- Autolinking และการใช้ไลบรารีภายในองค์กรผ่าน Private Registry
- ภาพรวม Swift Package Manager ที่ยังอยู่ในขั้นทดลอง
- Lab: ย้ายโมดูลจาก Lab ก่อนหน้าเป็นไลบรารีและติดตั้งในแอปใหม่

DAY 2

Performance, ข้อมูล และการแจ้งเตือน

เนื้อหาการอบรม · Section 6-10

รวม
6 ชั่วโมง

6 Lab: วัดและปรับประสิทธิภาพ

- Hermes V1 และ Bytecode ใน Release Build
- Performance และ Network Panel ใน React Native DevTools
- Android Studio Profiler, Xcode Instruments และ Perfetto
- ลดเวลาเปิดแอป ปรับ FlatList และลดการ Render ซ้ำ
- Lab: หาและแก้จุดที่ทำให้หน้ารายการสินค้ากระตุก

7 Lab: ข้อมูลในเครื่องแบบ Offline-first

- เลือกที่เก็บข้อมูล: MMKV สำหรับค่าตั้งค่า และ SQLite ด้วย op-sqlite สำหรับข้อมูลจำนวนมาก
- ตรวจสถานะเครือข่ายด้วย NetInfo และคิวรายการที่รอส่ง
- ซิงก์ข้อมูลเมื่อกลับมาออนไลน์และจัดการข้อมูลชนกัน
- เก็บ Token อย่างปลอดภัยด้วย Keychain และ Keystore
- Lab: แอปจดคำสั่งซื้อที่ทำงานได้ตอนออฟไลน์

8 Lab: Push Notification ผ่าน FCM และ APNs

- ตั้งค่า Firebase และ APNs Key สำหรับ iOS
- ขอสิทธิ์ด้วย react-native-permissions และ POST_NOTIFICATIONS บน Android 13 ขึ้นไป
- รับ Token และจัดการข้อความตอนเปิดแอปและเบื้องหลังด้วย React Native Firebase
- ทดสอบบนมือถือจริงเพราะ iOS Simulator ไม่รับ Push จริง
- Lab: แจ้งเตือนเมื่อสถานะคำสั่งซื้อเปลี่ยน

9 Lab: Local Notification ด้วย Native Module

- Android: Notification Channel และ NotificationManager
- iOS: UNUserNotificationCenter และการตั้งเวลาแจ้งเตือน
- ห่อทั้งสองแพลตฟอร์มเป็น Turbo Native Module เดียว
- Lab: แจ้งเตือนนัดหมายที่ตั้งเวลาไว้ในเครื่อง

10 Lab: Deep Linking, Universal Links และ App Links

- Custom URL Scheme กับลิงก์ https ที่เปิดแอปได้
- Android App Links: Intent Filter และไฟล์ assetlinks.json
- iOS Universal Links: Associated Domains และ RCTLinkingManager
- ตั้งค่า Linking ของ React Navigation 7 และเปิดหน้าจากการแจ้งเตือน
- Lab: ลิงก์ในอีเมลที่เปิดหน้ารายละเอียดคำสั่งซื้อในแอป

DAY 3

Testing, Release และ CI

เนื้อหาการอบรม · Section 11-15

รวม
6 ชั่วโมง

11 Lab: Unit Test และ E2E Test

- Jest กับ @react-native/jest-preset และ React Native Testing Library
- Mock Native Module ที่เขียนเองในการทดสอบ
- E2E Test ด้วย Detox บน Emulator และ Simulator
- Lab: ชุดทดสอบขั้นตอนสั่งซื้อทั้ง Unit และ E2E

12 Lab: Build Variant และ Environment

- Android productFlavors และ iOS Scheme กับ Build Configuration
- แยก Application ID ชื่อแอป และไอคอนของ Dev, Staging และ Production
- ส่งค่า Environment เข้าแอปโดยไม่ฝังความลับใน Bundle
- versionCode, versionName และ Build Number
- Lab: ติดตั้งแอป Staging และ Production คู่กันบนเครื่องเดียว

13 Lab: Signed Release สำหรับ Google Play และ App Store

- Upload Key ด้วย keytool และ signingConfigs ใน Gradle
- สร้าง AAB และใช้ Play App Signing กับ Internal Testing
- iOS: Release Scheme, Archive และ Distribute App ไปยัง App Store Connect
- ทดสอบผ่าน TestFlight และเตรียมข้อมูลหน้า Store
- Lab: Build รุ่น Release และส่งขึ้นช่องทางทดสอบ

14 Lab: CI, fastlane และ Crash Monitoring

- fastlane lane สำหรับ Android ด้วย gradle และ upload_to_play_store
- fastlane lane สำหรับ iOS ด้วย match, build_app และ upload_to_testflight
- รัน Pipeline บน GitHub Actions พร้อมเก็บความลับอย่างปลอดภัย
- ติดตาม Crash ด้วย Sentry และอัปโหลด Source Map ของ Hermes
- Lab: Pipeline ที่ทดสอบ Build และส่งแอปขึ้นช่องทางทดสอบอัตโนมัติ

15 Workshop: แอปพร้อมขึ้น Store

- รวม Native Module, ข้อมูลออฟไลน์, Push และ Deep Link ในแอปเดียว
- วัดประสิทธิภาพและแก้ปัญหาที่พบบนมือถือจริง
- รันชุดทดสอบและ Pipeline จนได้ Build สำหรับทดสอบ
- นำเสนอผลงาน รีวิวโค้ด และวางแผนออกเวอร์ชันต่อไป

07 เรียนจบแล้วได้อะไร Outcomes

- **เข้าใจ New Architecture** JSI, Codegen และ Fabric
- **เขียนโค้ดเนทีฟได้** Turbo Module และ Native Component
- **ปรับประสิทธิภาพได้** Hermes V1 และเครื่องมือ Profiler
- **ทำฟีเจอร์แอปจริงได้** Offline, Push และ Deep Link
- **ทดสอบแอปได้** Jest และ Detox
- **ออกเวอร์ชันอัตโนมัติได้** Signed Release และ fastlane

08 สิ่งที่ผู้เรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมครบทั้ง 3 วัน และ Lab Guide ทีละขั้นตอน
- ซอร์สโค้ดตัวอย่างครบทุก Lab รวม Native Module และไลบรารีที่สร้างขึ้น
- ไฟล์ Fastfile และ Workflow CI ตัวอย่าง พร้อม Checklist ก่อนส่งแอปขึ้น Store
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สามิตร โกยม ผู้สอนและนักพัฒนาที่ดูแลแอป React Native ในงานจริงทั้งฝั่ง JavaScript และฝั่งเนทีฟ สอนโดยพาเปิดโปรเจกต์ Android และ iOS ดูทีละส่วนว่าโค้ดที่เขียนทำงานอย่างไร แล้วลงมือสร้างฟีเจอร์ที่แอปจริงต้องมีจนถึงการส่งขึ้น Store เพื่อให้ผู้เรียนแก้ปัญหาเองได้เมื่อไลบรารีที่มีอยู่ไม่ตอบโจทย์

ราคาค่าอบรม 8,500 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

8,500 บาท / ท่าน (ยังไม่รวม VAT 7%)

สมัครผ่านเว็บไซต์ ลด 10%

02-570-8449 · 088-807-9770

@itgenius · contact@itgenius.co.th

เพิ่มเพื่อน LINE