



Advanced React Native with Expo



หลักสูตร "Advanced React Native with Expo" นี้ได้รับการออกแบบมาเพื่อยกระดับความสามารถของนักพัฒนาที่มีพื้นฐาน React Native แล้ว ให้สามารถสร้างแอปพลิเคชันระดับมืออาชีพที่พร้อมใช้งานในธุรกิจจริง โดยเน้นการเรียนรู้เทคนิคขั้นสูงในการจัดการ State Management, การทำงานกับ API แบบซับซ้อน การเพิ่มประสิทธิภาพแอป และการจัดการข้อมูลขนาดใหญ่

วัตถุประสงค์:

- ให้ผู้เรียนเชี่ยวชาญการใช้งาน State Management แบบขั้นสูงด้วย Redux Toolkit และ Zustand
- สอนเทคนิคการเชื่อมต่อและจัดการ API แบบซับซ้อน รวมถึงการใช้งาน GraphQL และ Real-time Subscriptions
- ฝึกการออกแบบ Architecture แอปแบบ Scalable และการจัดการ Code Structure ระดับมืออาชีพ
- แนะนำการใช้งาน Performance Optimization เพื่อให้แอปทำงานได้อย่างราบรื่นแม้จะมีข้อมูลขนาดใหญ่
- สอนการพัฒนาฟีเจอร์ขั้นสูง เช่น การทำงานแบบ Offline, Push Notifications

กลุ่มเป้าหมาย:

- นักพัฒนาที่ผ่านการอบรมหลักสูตร Basic React Native with Expo หรือมีประสบการณ์เทียบเท่า
- Mobile Developer ที่ต้องการยกระดับทักษะการพัฒนาแอปสู่ระดับมืออาชีพ
- Software Engineer ที่ต้องการเรียนรู้การพัฒนาแอปพลิเคชันระดับองค์กร
- Tech Lead หรือ Senior Developer ที่ต้องการเข้าใจ Architecture และ Best Practices ของ React Native
- ผู้ประกอบการที่ต้องการพัฒนาแอปธุรกิจด้วยตนเองในระดับที่ซับซ้อนขึ้น

ความรู้พื้นฐาน:

- ผ่านการอบรมหลักสูตร Basic React Native with Expo หรือมีประสบการณ์พัฒนา React Native
- มีความเข้าใจเกี่ยวกับ ES6+, Async/Await, Promises และ TypeScript Fundamentals
- เข้าใจ React Hooks, Context API, Component Lifecycle และ State Management พื้นฐาน
- มีประสบการณ์การทำงานกับ REST API, HTTP Methods และการจัดการ Error Handling
- เข้าใจพื้นฐานการทำงานของฐานข้อมูล, CRUD Operations และ Authentication concepts
- คำนึงเกี่ยวกับ Git, Package Management (npm/yarn) และ Command Line Interface
- เข้าใจ Mobile UI/UX Patterns และ Platform-specific considerations (iOS/Android)

ระยะเวลาในการอบรม:

- 18 ชั่วโมง (3 วัน)

วิทยากรผู้สอน:

- อาจารย์สามิตร ไทยม



เนื้อหาการอบรม:

วันที่ 1 Architecture & State Management

Section 1: Advanced Project Architecture

- Clean Architecture principles สำหรับ React Native
- Folder Structure และ Code Organization แบบ Scalable
- TypeScript Configuration ขั้นสูงสำหรับ Large-scale Projects
- Design Patterns: Repository, Factory และ Observer Pattern

Section 2: Advanced State Management

- Redux Toolkit: Store setup, Slices และ RTK Query
- Zustand: เปรียบเทียบดีกว่า Redux
- การเลือกใช้ Local vs Global State อย่างเหมาะสม
- State Persistence และ Hydration Strategies

Section 3 Performance Optimization Fundamentals

- React Native Performance Profiling
- Memory Management และ Garbage Collection
- การใช้ React.memo, useMemo และ useCallback อย่างถูกต้อง
- Lazy Loading และ Code Splitting

Section 4 Workshop Setup - AuraMarket Project

- Project initialization ด้วย Advanced Template
- Development Environment setup
- Initial Architecture implementation



วันที่ 2 Advanced Features & Integration

Section 5: Advanced API Integration

- GraphQL with Apollo Client
- Real-time Data ด้วย WebSocket และ Server-Sent Events
- Advanced HTTP Client configuration (Interceptors, Retry logic)
- API Caching Strategies และ Offline-first Architecture

Section 6: Authentication & Security

- Multi-provider Authentication (OAuth, Social Login)
- JWT Token Management และ Refresh Token handling
- Biometric Authentication integration
- Security Best Practices: Keychain, Encryption

Section 7: Advanced UI/UX Implementation

- Custom Animations ด้วย Ranimated 3
- Gesture Handler สำหรับ Complex Interactions
- Advanced Layout Techniques
- Dark Mode และ Theme Management

Section 8: Workshop - Core Features Implementation

- User Authentication system
- Product catalog ด้วย Advanced filtering
- Shopping cart ด้วย Persistent state



วันที่ 3 Production-Ready Features

Section 9: Push Notifications & Real-time Features

- Firebase Cloud Messaging setup
- Local Notifications และ Scheduling
- Real-time Chat implementation
- Background Tasks และ App State Management

Section 10: Testing & Quality Assurance

- Unit Testing ด้วย Jest และ React Native Testing Library
- Integration Testing strategies
- E2E Testing ด้วย Detox
- Code Quality tools: ESLint, Prettier, TypeScript

Section 11: Deployment & Monitoring

- Advanced EAS Build Configuration
- Environment Management (Development, Staging, Production)
- App Store Optimization (ASO) strategies
- Crash Reporting ด้วย Sentry
- Analytics integration และ Performance Monitoring