

# Đào Tạo Lập Trình Ứng Dụng Trên Thiết Bị Di Động Nâng Cao

## I. Tổng quan

Khóa học này đi sâu vào các kỹ thuật và khái niệm nâng cao trong phát triển ứng dụng di động đa nền tảng sử dụng React Native. Dựa trên nền tảng vững chắc về ReactJS, khóa học tập trung vào tối ưu hóa hiệu năng, tích hợp sâu với nền tảng gốc, quản lý trạng thái phức tạp, và các kiến trúc ứng dụng hiện đại.

## II. Thời lượng

40 giờ (5 ngày)

## III. Mục tiêu khóa học

- Nắm vững các kỹ thuật tối ưu hóa hiệu năng cho ứng dụng React Native.
- Thành thạo các phương pháp quản lý trạng thái nâng cao (Redux Toolkit, Zustand, Jotai).
- Tìm hiểu cách tích hợp Native Modules và Native UI Components.
- Xây dựng và triển khai các ứng dụng React Native phức tạp với kiến trúc vững chắc.
- Áp dụng các quy trình kiểm thử tự động (Unit, Integration, E2E) cho ứng dụng React Native.
- Khám phá các tính năng nâng cao và hệ sinh thái của React Native

## IV. Đối tượng tham gia:

- Các nhà phát triển React Native muốn nâng cao kỹ năng và xây dựng các ứng dụng phức tạp, hiệu suất cao
- Lập trình viên đã hoàn thành khóa học cơ bản về React Native hoặc có kinh nghiệm tương đương
- Các nhà phát triển muốn tìm hiểu sâu về kiến trúc và tối ưu hóa trong React Native.

## V. Điều kiện tiên quyết

- Kiến thức vững chắc về JavaScript (ES6+) và ReactJS (bao gồm Hooks, Context API)
- Đã có kinh nghiệm xây dựng ứng dụng cơ bản với React Native
- Hiểu biết về các khái niệm cơ bản trong phát triển di động (iOS và Android) là một lợi thế.

## VI. Nội dung khóa học

### 1. Module 1: React Native Advanced Concepts & Performance

- Deep Dive into the React Native Bridge and New Architecture (JSI, Fabric, TurboModules)
- Profiling React Native Applications (CPU, Memory, UI Rendering)
- Performance Optimization Techniques (Memoization, FlatList Optimization, Code Splitting, RAM Bundles)
- Handling Large Lists and Complex UIs Efficiently
- Understanding and Optimizing Image Loading and Caching
- Debugging Performance Issues with Flipper and Native Tools

#### Lab 1: Profiling and Debugging

- Sử dụng Flipper để phân tích hiệu năng của một ứng dụng mẫu.
- Xác định và khắc phục các điểm nghẽn về hiệu năng (bottlenecks).

#### Lab 2: Optimizing FlatList

- Áp dụng các kỹ thuật getItemLayout, keyExtractor, windowSize để tối ưu hóa FlatList.
- Implement conditional logic based on user input or data manipulation.

#### Lab 3: Image Optimization

- Triển khai chiến lược tải và cache ảnh hiệu quả.
- Sử dụng các thư viện tối ưu hóa ảnh.

### 2. Module 2: Advanced State Management & Architecture:

- Advanced Redux Patterns (Redux Toolkit Query, Middleware, Selectors Optimization)
- Exploring Alternatives: Zustand, Jotai for Global and Local State
- Managing Complex Application State and Data Flow
- Structuring React Native Projects for Scalability (Monorepos, Feature-based structure)
- Navigation Patterns for Complex Apps (Nested Navigators, Custom Transitions)
- Dependency Injection and Inversion of Control in React Native

#### Lab 1: Implementing RTK Query

- Tích hợp RTK Query để quản lý trạng thái API và caching.
- Xử lý loading, error states và data fetching.

### Lab 2: Exploring Zustand/Jotai

- Xây dựng một tính năng sử dụng Zustand hoặc Jotai để quản lý trạng thái.
- So sánh với cách tiếp cận Redux hoặc Context API.

## 3. Module 3: Native Modules & Native UI Components Integration

- Understanding Native Modules Communication (Bridge vs. JSI)
- Creating Custom Native Modules for iOS (Swift/Objective-C) and Android (Kotlin/Java)
- Integrating Existing Native SDKs into React Native
- Creating Custom Native UI Components and Bridging Them
- Working with Platform-Specific APIs (Permissions, Device Info, Sensors)
- Troubleshooting Native Integration Issues

### Lab 1: Creating a Custom Native Module

- Xây dựng một Native Module đơn giản (ví dụ: lấy thông tin thiết bị) cho cả iOS và Android.
- Gọi Native Module từ JavaScript.

### Lab 2: Bridging a Native UI Component

- Tạo một Native UI Component đơn giản (ví dụ: custom button) và tích hợp vào React Native.
- Xử lý props và events giữa Native và JavaScript.

### Lab 3: Integrating a Native SDK

- Tích hợp một SDK native bên thứ ba (ví dụ: bản đồ, camera) vào ứng dụng React Native.
- Xử lý cài đặt và gọi các API của SDK.

## 4. Module 4: Advanced Testing & Deployment Strategies

- Unit Testing React Native Components (Jest, React Native Testing Library)
- Integration Testing Navigation, State Management, and Native Modules
- End-to-End (E2E) Testing with Detox or Maestro

- Setting up CI/CD Pipelines for React Native (GitHub Actions, Codemagic, App Center)
- Over-the-Air (OTA) Updates with CodePush / EAS Updates
- Store Deployment Process (App Store Connect, Google Play Console) including Beta Testing (TestFlight, Internal Testing)

### Lab 1: Unit & Integration Testing

- Viết unit test cho các components và logic nghiệp vụ.
- Viết integration test cho luồng navigation và state management.

### Lab 2: E2E Testing Setup

- Cấu hình Detox hoặc Maestro cho một ứng dụng mẫu.
- Viết kịch bản E2E test cho một luồng người dùng chính.

## 5. Module 5: Cutting-Edge React Native & Ecosystem

- Exploring Expo Ecosystem (EAS Build, EAS Submit, Expo Go, Development Clients)
- Server Components in React Native (Experimental)
- Advanced Animation Techniques (Reanimated 3, Skia)
- Working with Offline First Applications (WatermelonDB, AsyncStorage)
- Security Best Practices in React Native (Secure Storage, SSL Pinning)
- Contributing to the React Native Community and Open Source

### Lab 1: Building with EAS

- Sử dụng EAS Build để tạo bản build cho development và production.
- Sử dụng EAS Submit để gửi ứng dụng lên store.

### Lab 2: Advanced Animations with Reanimated

- Triển khai các hiệu ứng animation phức tạp sử dụng Reanimated 3.
- Tạo các gesture-based animations.