

# React Native 4-Week Project Plan

## 4-Week Iterative React Native Mastery Schedule (Project-Based + Spaced Repetition)

### Daily Format:

- 4 hours coding project work
- 1 hour spaced repetition (from question list)

### Week 1: Foundations & Fundamentals (Build: Task Tracker)

#### Day 1: Setup & JavaScript Fundamentals

- Setup RN CLI + VSCode extensions
- Practice: variables, types, let/const, arrow functions, imports/exports
- Project: Build a static TaskList screen with sample task data

#### Day 2: Classes, Promises & Arrays

- Practice: classes, spread/rest, destructuring, array functions, async/promises
- Project: Add buttons to filter/combine task arrays using ES6

#### Day 3: JSX & Components

- JSX syntax, stateful/stateless components, props
- Project: Refactor task list into reusable components (TaskItem, TaskList)

#### Day 4: Core Components & HOCs

- Practice: TextInput, Button, View, ScrollView
- Project: Add task creation form and an HOC to handle validation

#### Day 5: Flexbox & Navigation (Stacks)

- Flexbox basics, react-navigation setup
- Project: Convert to multi-screen app (Home, Add Task, Task Details)

### Weekend Project: Mini CRUD Task Manager

- Add, view, delete tasks
- Persistent state via component state (no backend yet)

## React Native 4-Week Project Plan

Week 2: Navigation, State, Hooks (Build: NoteKeeper)

Day 6: Navigation Deep Dive

- navigate vs push, goBack, useEffect, route params
- Project: Login Dashboard Note View

Day 7: Tabs, Drawers & Modals

- Add tab bar (Notes, Profile), drawer nav, image modals
- Project: Allow switching between Notes/Profile, show note attachments

Day 8: Lifecycle & useState

- Component lifecycle, side effects, useState
- Project: Show notes that auto-refresh on screen focus

Day 9: useMemo, useCallback, AsyncStorage

- AsyncStorage helpers, memoization hooks
- Project: Persist notes locally with caching & performance optimizations

Day 10: Context API

- Context setup, limitations, Provider pattern
- Project: Replace prop drilling in NoteKeeper using Context

Weekend Project: Context-Driven NoteKeeper

- Add logout, dashboard state via Context
- Add offline note saving using AsyncStorage

Week 3: Data & Redux (Build: Shopping List)

Day 11: Error Handling & API

- Debugging, Flipper, useRef, fetch
- Project: Load shopping items from mock API, show network error banner

Day 12: Redux & Redux Toolkit

## React Native 4-Week Project Plan

- Setup store, slices, actions
- Project: Add shopping items using Redux + Toolkit

### Day 13: Redux Persist & Auth Flow

- Redux persist, guest vs auth states
- Project: Add login/register screens and protect screens

### Day 14: Storage + Crashlytics

- Keychain, MMKV, Firebase Crashlytics
- Project: Store tokens securely and log user actions/crashes

### Day 15: Redux Saga + Auth API

- Redux Saga intro, token-based auth, profile screen
- Project: Replace Redux async with Sagas, use refresh tokens

### Weekend Project: Authenticated Shopping App

- Redux Saga for auth
- Full offline and online support using Redux persist

## Week 4: Real-World Features & Scaling (Build: Event Planner)

### Day 16: Lists, Animations & Toolkit Query

- FlatList, VirtualizedList, RN Animatable
- Project: Event list with shimmer/loading animations

### Day 17: Location & Maps

- Geolocation, permissions, maps
- Project: Add venue picker + display user location

### Day 18: Push Notifications & Analytics

- FCM, RN Firebase Analytics
- Project: Push notification integration + screen tracking

## **React Native 4-Week Project Plan**

Day 19: API Helpers, SSL, Security

- Write API helper, SSL pinning, secure practices
- Project: Securely call event APIs and encrypt user info

Day 20: Distributions & Code Review

- useRef, Firebase distribution, store upload
- Project: Submit signed build to Firebase

Weekend Project: Event Planner Pro

- Combine all features: auth, maps, notifications, analytics
- Secure and scalable

Week 5+: Polishing, Final Project, iOS Specific

You can begin a full-scale app with all covered concepts (e.g. chat app, booking app, or social app).