

Navigation in React Native

Lesson 5

Lesson Objectives

Navigate between screens

Create multiple screens

Use Stack Navigator

Pass data between screens

Understand navigation structure

What is Navigation?

Navigation allows users to move between different screens.

Examples:

Home Screen → Profile Screen

Login Screen → Dashboard

Product List → Product Details

Expo Router: Stack + Tabs + Modal

File-based routes from `src/app`

Tabs group for main app flow

In Expo Router, routes come from files. We combine a root Stack (for app-level routes like modals) with Tabs (for main screens).

File-based routing (Expo Router)

File = Route

Routes live inside `src/app/`

Folders can group routes (e.g., (tabs)), without appearing in the URL

`_layout.tsx` defines a navigator for that folder

In React Navigation

In React Navigation where you register screens manually, Expo Router reads your folder structure and generates routes automatically.

`src/app/_layout.tsx`

`src/app/(tabs)/`

`src/app/modal.tsx`

Tabs: `src/app/(tabs)/_layout.tsx`

Tabs navigator via `<Tabs />`

Each file inside (tabs) becomes a tab screen

Example tabs:

`index` → Home

`explore` → Explore

Shared tab options (icons, styling)

Home Screen: `src/app/(tabs)/index.tsx`

`index.tsx` maps to Home tab

Displays UI components (ex: CounterButton)

Navigation can be added from here (Link / router)

Explore Screen: `src/app/(tabs)/explore.tsx`

`explore.tsx` maps to Explore tab

FlatList

cards layout

search/filter later

Modal Screen: `src/app/modal.tsx`

Separate stack route: `/modal`

Presented on top of tabs

Dismiss/back behavior (Link or router back)

Modals are not tabs. They are pushed by the stack above everything. We use modals for “Create”, “Quick actions”, “Edit”, “Settings”, etc.

How users move in the app

App starts in (tabs) by default

Tabs switch: Home ↔ Explore

Modal is opened from anywhere → sits on top → dismiss returns