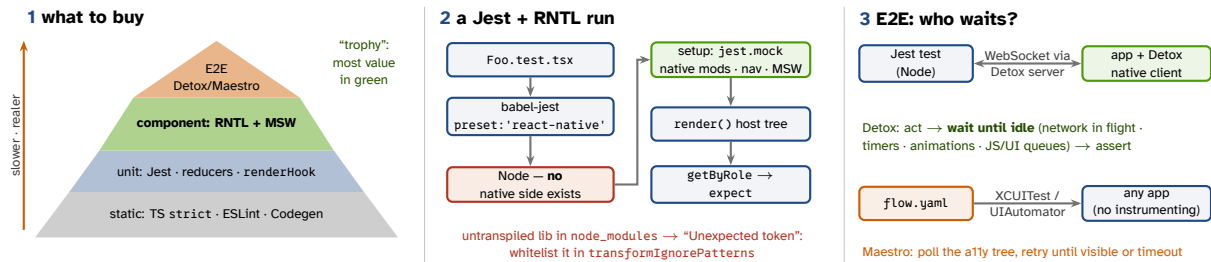


In one line: Buy **confidence** per **second**: TS strict for free, Jest for logic, the **bulk** in **RNTL** component tests that query *like a user* against mocked network, and a **few** E2E journeys — **Detox** (grey-box) or **Maestro** (black-box).



How it works — the tools

- Jest** runs in **Node**: the preset stubs core native bits; every *third-party* native module is `jest.mocked` (many ship a mock, e.g. `react-native-gesture-handler/jestSetup`). Fake timers: `jest.useFakeTimers() + advanceTimersByTime`.
- RNTL** renders the React tree (no pixels, no layout), queries *host* elements: `ByRole('button', {name}) > ByLabelText > ByText > ByTestId` (last resort). Role needs an *accessible* element — tests force real a11y. `getBy` throws · `queryBy` → null (absence) · `findBy` = async wait.
- fireEvent.press** just calls the nearest `onPress`. **userEvent** (async) replays `pressIn/Out`, `focus`, `per-key changeText`, `blur`, and respects `disabled / editable={false}`.
- Hooks**: `renderHook(fn, {wrapper}) → result.current`; or test *through* the component.
- Network**: `MSW` `setupServer(msw/node)` — real client + parsing, `per-test server.use()` for errors. Cheaper seam: mock the API module. **TanStack Query**: **new QueryClient per test**, `retry:false`.
- Navigation**: real `NavigationContainer` + navigator (integration), or mock `useNavigation` and assert `navigate` (unit). `TurboModule`: mock its `Native<X>.ts` spec file.
- Detox** (Wix): instrumented build, `by.id(testID)`, no sleeps; costs native builds + upgrade breakage; a never-idle app (looping animation, polling) hangs it.
- Maestro**: `YAML` (`tapOn`, `assertVisible`), no app changes, also native/Flutter; slower, fuzzier waits.

Example — a user-level test

```
it('shows the error, then retries', async () => {
  server.use(http.get('/me', () => HttpResponse.error()));
  const user = userEvent.setup();
  render(<Profile />, { wrapper: Providers }); // new client
  expect(await screen.findByText(/could not load/i))
    .toBeOnTheScreen();
  server.resetHandlers(); // next call succeeds
  await user.press(screen.getByRole('button', { name: 'Retry' }));
  expect(await screen.findByText('Ada')).toBeOnTheScreen();
});
```

TypeScript for RN seniors

```
type Load<T> = { status: 'idle' } | { status: 'loading' }
  | { status: 'ok'; data: T } | { status: 'err'; error: Error };
type Props = PressableProps & { title: string;
  style?: StyleProp<ViewStyle>; children?: React.ReactNode };
function List<T>(p: { items: readonly T[];
  key: (t: T) => string; row: (t: T) => React.ReactElement }) {}
type Root = { Home: undefined; Profile: { userId: string } };
type P = NativeStackScreenProps<Root, 'Profile'>;
const gap = { s: 4, m: 8 } satisfies Record<string, number>;
```

```
// NativeCrypto.ts -- Codegen reads this spec
export interface Spec extends TurboModule {
  hash(input: string): Promise<string>; }
export default
  TurboModuleRegistry.getEnforcing<Spec>('NativeCrypto');
```

- Unions**: no loading + error at once; `const x: never = s` in the `default = exhaustive`.
- Skip `React.FC` (implicit children dropped in React 18 types); extend `host props` via `ComponentProps<typeof X>`.
- satisfies** (TS 4.9) checks *without widening* — literals and keys stay precise; as just silences the checker.
- Nav**: global `ReactNavigation.RootParamList` types `useNavigation()`; deep-link params arrive as *strings* — validate.
- Flags**: `strict`, `noUncheckedIndexedAccess` (`arr[i]: T | undefined`), `exactOptionalPropertyTypes`.
- Codegen**: `Native<Name>.ts` + `codegenConfig` (`package.json`) → C++/ObjC++/Java glue: the TS is the ABI, in a subset (number, `Int32`, `string`, `boolean`, `objects`, `arrays`, `Promise`, `callbacks`).

Interview traps

- Types vanish at runtime: `res.json()` as `User` is a lie — validate at the boundary (`zod/valibot`) or type it unknown.
- Flakes: un-awaited `findBy/userEvent`; real timers; leaked state (singletons, shared `QueryClient`); live network; E2E permission dialogs.
- "not wrapped in `act(...)`" = an update you didn't wait for: `findBy`, not more `act`.
- Whole-screen snapshots = change detectors; asserting state or child internals tests the implementation.
- Jest green ≠ works: no native code, layout or threads ran.

Remember

Query like a user · mock at the edge · wait, don't sleep · type the boundary.

Likely questions

- Detox vs Maestro? — Detox: fast, deterministic idle-sync, heavy setup; Maestro: easy, black-box, slower waits.
- Why `getByRole` over `testID`? — it tests what users and screen readers see, and survives refactors.
- `fireEvent` vs `userEvent`? — a direct handler call vs the realistic event sequence.
- Component calling a native module? — `jest.mock` the module/spec; cover the real path in E2E.
- What to E2E? — only the money journeys (login, checkout, onboarding), not every branch.