**Seminarul 11 — Redux Toolkit — Partea 3 (Proiecte/teme, extins)**

Această Parte 3 propune \*\*45 de proiecte\*\* (15 × L1, 15 × L2, 15 × L3) pentru consolidarea \*\*Redux Toolkit\*\* într‑un mod progresiv. Fiecare proiect vine cu:  
• \*\*specificații\*\*, \*\*learning goals\*\*, \*\*sugestii AI‑assist (VSL)\*\*,  
• un \*\*starter\*\* (în care testele pot pica) și o \*\*solution\*\* (minimă, verde),  
• \*\*teste duale\*\* (Vitest & Jest) și \*\*README.md\*\* dedicat,  
• integrare de bază cu \*\*TypeScript\*\* și structură RTK standard.  
  
Nivelurile acoperă: L1 — \*fundamente (slice, entityAdapter, selectors, wiring UI)\*; L2 — \*asincron (thunks + MSW), persistență, middleware, Reselect perf, code‑splitting, RTK Query intro\*; L3 — \*modele avansate (multi‑entity, Query avansat, optimistic robust, WS middleware, SSR/hydration, cross‑slice, Zod, profilare, migrare, securitate)\*.

# L1 (Fundamental) — 15 proiecte

## L1-P01 — Slice Basics: Counter state

\*\*Tema:\*\* Slice Basics: Counter state — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P02 — Entity Adapter Intro: Clubs list

\*\*Tema:\*\* Entity Adapter Intro: Clubs list — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P03 — Selectors Basics: Visible clubs by query

\*\*Tema:\*\* Selectors Basics: Visible clubs by query — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P04 — UI Wiring: useAppSelector/useAppDispatch

\*\*Tema:\*\* UI Wiring: useAppSelector/useAppDispatch — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P05 — Controlled Forms: Add Club

\*\*Tema:\*\* Controlled Forms: Add Club — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P06 — Filtering by Category

\*\*Tema:\*\* Filtering by Category — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P07 — Persist Middleware (intro)

\*\*Tema:\*\* Persist Middleware (intro) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P08 — Reselect Memoization Basics

\*\*Tema:\*\* Reselect Memoization Basics — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P09 — Feature Foldering

\*\*Tema:\*\* Feature Foldering — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P10 — DevTools Sanity

\*\*Tema:\*\* DevTools Sanity — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P11 — Error Handling (sync)

\*\*Tema:\*\* Error Handling (sync) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P12 — Type‑safe Actions (PayloadAction)

\*\*Tema:\*\* Type‑safe Actions (PayloadAction) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P13 — Remove/Update Entities

\*\*Tema:\*\* Remove/Update Entities — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P14 — Empty State & A11y

\*\*Tema:\*\* Empty State & A11y — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L1-P15 — Integration Smoke

\*\*Tema:\*\* Integration Smoke — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

# L2 (Intermediar) — 15 proiecte

## L2-P01 — Async Thunk: fetchClubs + MSW

\*\*Tema:\*\* Async Thunk: fetchClubs + MSW — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P02 — CRUD Thunks (add/update/delete)

\*\*Tema:\*\* CRUD Thunks (add/update/delete) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P03 — Optimistic Update (basic)

\*\*Tema:\*\* Optimistic Update (basic) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P04 — Pagination slice + selector

\*\*Tema:\*\* Pagination slice + selector — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P05 — Search + Category chaining

\*\*Tema:\*\* Search + Category chaining — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P06 — Logger Middleware

\*\*Tema:\*\* Logger Middleware — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P07 — Persist Subtree + Rehydrate

\*\*Tema:\*\* Persist Subtree + Rehydrate — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P08 — Undo/Redo (history in reducer)

\*\*Tema:\*\* Undo/Redo (history in reducer) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P09 — RTK Query Intro (createApi)

\*\*Tema:\*\* RTK Query Intro (createApi) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P10 — Mutations & Invalidation

\*\*Tema:\*\* Mutations & Invalidation — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P11 — Error Boundary integration (UI)

\*\*Tema:\*\* Error Boundary integration (UI) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P12 — Selective Subscription (shallowEqual)

\*\*Tema:\*\* Selective Subscription (shallowEqual) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P13 — Reselect Perf scenario

\*\*Tema:\*\* Reselect Perf scenario — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P14 — Code splitting (injectReducer)

\*\*Tema:\*\* Code splitting (injectReducer) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L2-P15 — Testing Thunks advanced

\*\*Tema:\*\* Testing Thunks advanced — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

# L3 (Avansat) — 15 proiecte

## L3-P01 — Multi‑Entity Graph (clubs/members/events)

\*\*Tema:\*\* Multi‑Entity Graph (clubs/members/events) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P02 — RTK Query Advanced patterns

\*\*Tema:\*\* RTK Query Advanced patterns — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P03 — Optimistic UI robust (rollback)

\*\*Tema:\*\* Optimistic UI robust (rollback) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P04 — Infinite Scroll / Cursor Pagination

\*\*Tema:\*\* Infinite Scroll / Cursor Pagination — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P05 — WebSocket middleware → entities

\*\*Tema:\*\* WebSocket middleware → entities — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P06 — Permissions & Feature Flags (selectors)

\*\*Tema:\*\* Permissions & Feature Flags (selectors) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P07 — Hydration (SSR) skeleton

\*\*Tema:\*\* Hydration (SSR) skeleton — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P08 — State Machines (XState) integration — concept

\*\*Tema:\*\* State Machines (XState) integration — concept — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P09 — Wizard Form orchestrated

\*\*Tema:\*\* Wizard Form orchestrated — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P10 — Cross‑Slice Actions (extraReducers)

\*\*Tema:\*\* Cross‑Slice Actions (extraReducers) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P11 — Schema Validation (Zod) in thunks

\*\*Tema:\*\* Schema Validation (Zod) in thunks — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P12 — Non‑serializable guard & serialization

\*\*Tema:\*\* Non‑serializable guard & serialization — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P13 — Profiling & renders (why-did-you-render pattern)

\*\*Tema:\*\* Profiling & renders (why-did-you-render pattern) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P14 — Migration Legacy Redux → RTK

\*\*Tema:\*\* Migration Legacy Redux → RTK — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

## L3-P15 — Security & Resilience (logout reset, circuit breaker)

\*\*Tema:\*\* Security & Resilience (logout reset, circuit breaker) — implementați comportamentul esențial folosind Redux Toolkit 2.x (TypeScript) și treceți testele Vitest & Jest.

\*\*Learning goals:\*\* Consolidarea principiilor RTK pe nivelul curent.; Exersarea tipării stricte (TS) și a testării duale (Vitest/Jest).; Aplicarea bunelor practici (entityAdapter, selectors, thunks/Query).

\*\*Cerințe:\*\* pornește de la `starter/`, rulează `npm test`, apoi implementează logica cerută până când testele sunt verzi (Vitest & Jest).

\*\*Soluție (sumar):\*\* în `solution/` ai implementarea minimă (RTK 2.x) care trece testele. Compară diff‑ul pentru înțelegere.

\*\*AI‑assist (VSL):\*\* prompts scurte pentru generarea/ajustarea reducerelor, \*selectors\*, \*thunks\* și a serviciului `createApi`.

---  
\*\*Fragmente comune de cod (starter vs. solution) — ilustrative:\*\*

// L1 — selectVisibleClubs (starter vs. solution)

starter: createSelector([selectAll], list => list)  
solution: createSelector([selectAll, selectQuery, selectCategory], ...)

// L2 — createAsyncThunk fetchClubs (starter returnează [] pentru a forța eșecul testului; solution face fetch real către /api/clubs)

// L3 — RTK Query createApi: reducerPath='clubsApi', endpoints getClubs/addClub + hooks React