**Ghid de utilizare — arhive Partea 2 (Laborator)**

Acest ghid descrie conținutul și modul de utilizare pentru arhivele livrate în Partea 2 (Laborator) — varianta \*\*Standalone (npm)\*\* și varianta \*\*Monorepo (PNPM workspaces)\*\*. Include comenzi de rulare, structură de directoare și notițe de depanare.

---

## 1) Standalone — s4-p2-lab-standalone.zip

\*\*Structură:\*\* `src/` (module ESM), `public/` (HTML/CSS/JSON), `tests/` (Vitest/Jest).

\*\*Comenzi:\*\*  
```bash  
npm i  
npm run test  
npm run serve # http://localhost:5173/public/index.html  
```

\*\*Fișiere cheie:\*\*  
- `src/main.js` (bootstrap + cleanup cu AbortController + lazy import)  
- `src/events.js` (delegation + listenWithSignal)  
- `src/dom.js` (qs/qsa cu invariante)  
- `src/services/data.js` (loadEvents)  
- `src/filters.js` (filterItems)  
- testele oglindă în `tests/vitest` și `tests/jest`

---

## 2) Monorepo — s4-p2-monorepo.zip (PNPM workspaces)

\*\*Structură:\*\* `packages/lab-s4/\*`, `.github/workflows/unit-vitest.yml`, `.github/workflows/unit-jest.yml`.

\*\*Comenzi:\*\*  
```bash  
pnpm i  
pnpm -r run test  
pnpm -F @s4/lab-s4 run serve  
```

\*\*Scop:\*\* rularea testelor per pachet, izolarea dependențelor, pregătirea pentru extinderi (ex.: e2e, PWA) în Partea 3.

---

## 3) Depanare rapidă

- ESM + Jest: necesită `babel-jest` și `extensionsToTreatAsEsm`.  
- Căi HTML→module: folosiți `/src/...` din `public/index.html` când serverul servește rădăcina proiectului.  
- `fetch` în teste: \*mock\* în Vitest/Jest pentru a elimina dependența de rețea.