

## ACE Platform - One-Page App Summary

### What it is

ACE Platform is a monorepo web app with a React frontend and a Fastify backend. It supports trainee operations with role-aware flows and a DDD/clean-architecture backend structure.

### Who it's for

Primary users: trainees/students tracking daily activity, and admins managing trainee performance and operations.

### What it does

- Login flow with JWT issuance and client-side token storage.
- Trainee dashboard UI with stats, weekly hours, worklogs, and meetings.
- Admin panel UI with tabs for students, worklogs, LeetCode, meetings, concerns, and requests.
- Sidebar navigation for attendance, worklogs, reports, blog posts, LeetCode, meetings, concerns, and requests.
- Backend user APIs: health check, create user, get user by ID.
- MongoDB persistence via Mongoose schema + repository mapping.
- Not found in repo: implemented backend endpoints for most admin/user dashboard data routes used by frontend services.

### How it works (repo-evidenced architecture)

#### Components/services

- Frontend: React + Vite + React Router + React Query + Redux Provider + AuthContext + Axios service layer.
- Backend: Fastify app, auth/user routes, controllers, application use cases, domain entities/errors, Mongo repo.
- Data store: MongoDB via Mongoose (``UserModel``).

#### Data flow

1. User signs in from frontend ``Login`` page.
2. ``authService.login`` posts to ``/api/auth/login``.
3. ``AuthController`` calls ``LoginUser`` use case.
4. Use case checks user via ``MongoUserRepository``, validates password, then issues JWT.
5. Frontend stores token in ``localStorage``; Axios interceptor adds ``Authorization: Bearer <token>``.
6. Protected UI pages render; backend user APIs serve health/user data.

### How to run (minimal)

1. From repo root: ``npm install``
2. Start frontend: ``npm run dev:frontend``
3. Start backend: ``npm run dev:backend``
4. Optional env vars from code: ``PORT``, ``HOST``, ``DATABASE\_URI``, ``JWT\_SECRET``, ``JWT\_EXPIRES\_IN``, ``VITE\_API\_URL``
5. Not found in repo: documented ``.env.example`` and production deployment/run instructions.