

PROBLEM

Teams that run events, registrations and rosters live in a gap. Consumer form tools give a nice builder but a thin, generic results view: still exporting a CSV and rebuilding the same pivot tables every time. Enterprise platforms do the analytics but demand a full account setup and a budget. What people want is narrow: build the form fast, share a link, and have the answers they care about appear on their own.

APPROACH

The frontend talks only to a small, stable HTTP contract for forms, submissions and analytics. That contract is implemented twice: once as a Cloudflare Worker over D1, once as a Node/Express server over SQLite, so the client is blind to which backend serves it. On top sit 14+ field types with branching logic, role-based accounts with session auth, and a live dashboard of KPIs, attendance heat-matrices and capacity-aware roster auto-fill.

OUTCOME

14+

field types with branching logic

2

backends, both in production

1

prompt to a working form

AdvanceForm takes a team from "we need to collect this" to a live, analysed dataset without touching code or a spreadsheet. Describe the form or click it together, share the link, and the dashboard fills itself.

Vanilla JS SPA

Cloudflare Workers

Hono + D1

Node + Express

better-sqlite3

Google Gemini