MARKETPLACE APP · PLATFORM

App Builder

Describe the app you need and AI builds it — custom trackers, logs, and check-ins that run like first-party apps, with mobile, analytics, and AI built in.

Updated August 2026

MODEL	TIER	CAPABILITIES	INTELLIGENCE
Pay-as-you-go	**Platform**	**42+**	**AI-powered**

OVERVIEW

Every operations team carries a list of small apps nobody will ever build for them — the vehicle-inspection log, the visitor tracker, the daily check-in that lives in a spreadsheet because the internal-tools backlog is six months deep.

App Builder closes that gap: an administrator describes the app in plain language and an AI builder agent generates the whole thing — record types with list, detail, and form screens, notifications and SLA/overdue logic, approval chains and event automations, QR and public submission links, CSV import with automatic column matching, demo data, and mobile rendering — deployed as its own app at /apps/ext/<slug>, enabled per business like any first-party app.

The builder plans first: it drafts the plan and pauses for approval before writing any code, and the plan can be revised, edited, or declined. Every build passes deterministic fit checks, an AST-level security scan, and an AI QA review before it can deploy.

Built apps run inside the MangoApps SDK sandbox with a declared-capability model and full tenant isolation, so generated code can only use what the platform grants and only see the business's own data.

What separates a built app from a standalone no-code tool is what it can reach: 79 platform APIs, each gated on a capability the app declares. A generated app reads your real people, locations, departments and schedules, creates platform Tasks, routes approvals through the tenant's approval workflow, requests native e-signatures, and projects records onto the unified calendar and News Feed.

Admins start from a curated system template gallery or save their own builds as team templates, hand-edit generated files, restore prior versions, and reopen any live app with "Fix with AI." Every AI token the builder spends is metered and attributed, and the app's Analytics tab shows deployed apps, build outcomes, data volume, and spend.

Billed by usage per deployed app — no per-employee charge and no floor — and off by default until an admin opts the business in, because it deploys code into your tenant.

WHY TEAMS CHOOSE IT

- Describe the app you need — a maintenance log, a visitor tracker, a daily check-in — and AI builds the whole thing: records, screens, notifications, and mobile.
- Built apps aren't embeds or forms: each one deploys as its own app with its own pages, data, automations, and per-business enablement.
- Generated apps plug into the platform — they read your people and locations, create real Tasks, route approvals, request e-signatures, and post to the calendar and feed.
- Builds are planned first and pause for your approval, so you see what will be built — and can revise it — before any code is written.
- Every build passes deterministic fit checks, a static security scan, and an AI QA review before it can deploy.
- Usage-based: you pay per deployed app, so a tenant that ships two trackers pays for two — not for every employee on the roster.

CAPABILITIES

Describe-to-Build

- ✓ Describe an app in plain language; AI generates the complete app
- ✓ Live build narration — watch the plan, files, and checks stream in
- ✓ Plan approval before any code is written — Revise, edit, or decline the plan; "Fix with AI" repairs skip straight to the fix.
- ✓ Attach screenshots or documents as build input
- ✓ "Fix with AI" on a deployed app that misbehaves
- ✓ Hand-editing generated files before deploy
- ✓ Version history with one-click restore

What a Built App Includes

- ✓ Record types with list, detail, create, and edit screens
- ✓ Deploys as its own app at /apps/ext/<slug>, enabled per business
- ✓ Mobile rendering for built apps
- ✓ Notifications, SLA/overdue logic, and scheduled automations
- ✓ Approval chains and event-driven workflow automations
- ✓ QR codes and unauthenticated public submission links
- ✓ Record comments, report-a-problem, and demo data
- ✓ Sensitive fields with masked display and audited reveal
- ✓ CSV import with automatic column matching, plus export
- ✓ Inbound webhooks with per-endpoint signed secrets
- ✓ Semantic search over the app's own records

Platform Integration

- ✓ Reads live workforce data — people, locations, departments, shifts, schedules
- ✓ Creates real platform Tasks assigned to business members
- ✓ Routes approvals through the tenant's approval workflow
- ✓ Sends records for native e-signature with a generated PDF
- ✓ Projects records onto the unified calendar, ICS feed, and Outlook/Google sync
- ✓ Posts to the company News Feed through the governed pipeline
- ✓ Audience-targeted notifications by department, location, role, or group
- ✓ Geocoding and geofence checks against your business locations

Quality & Safety Gates

- ✓ Deterministic fit checks on every build (schema, paths, security scan)
- ✓ AI QA reviewer inspects the generated app before deploy
- ✓ Static security scan of generated code (AST-level)
- ✓ Per-session token budget so a runaway build can't burn unbounded AI spend
- ✓ Tenant isolation — a built app sees only its own business's data

Templates & Developer Access

- ✓ Curated system template gallery, importable as an editable draft
- ✓ Save any build as a reusable team template
- ✓ Scoped, revocable API keys for external build tooling
- ✓ MCP endpoint for building from Cursor, Claude Code, or the CLI

Management & Governance

- ✓ Your apps register — every built app with status, health, and data volume
- ✓ Recover a deployed app that fails to load, without leaving App Builder
- ✓ Per-app health monitoring with rollback to a previous version
- ✓ Lifecycle alerts target the app's builder and deployer, not every admin — Health alerts file as Inbox action requests; both alert types have tenant opt-out toggles.
- ✓ CSV export of your apps, builds, and AI-usage analytics
- ✓ Business-scoped AI usage analytics (builder tokens and build volume)
- ✓ Publish a built app to the MangoApps community directory

Limits & Specs

Platform APIs available to a built app	**79**
Declared capabilities an app can request	**16**
Built-app address	**/apps/ext/<slug>**
Ask AI agent tools	**6 (2 write actions, confirmation-gated)**

USE CASES

- **The tracker IT never had time to build**
  An operations admin describes a fleet-vehicle inspection log — fields, overdue rules, who gets notified — and deploys it the same afternoon, instead of joining a six-month internal-tools backlog.

- **Replacing a spreadsheet that outgrew itself**
  A shared spreadsheet of equipment loans becomes a real app: CSV import with automatic column matching brings the history in, and the app adds due-date notifications and a mobile view the spreadsheet never had.
- **A public intake form with a real backend**
  A facilities team publishes a QR-coded public submission link for hazard reports; submissions land as records with SLA timers, comments, and an overdue escalation instead of an unwatched inbox.
- **An app that knows who works where**
  A site-safety check-in reads the business's real locations and departments, assigns a follow-up Task to the responsible supervisor, and puts the next inspection date on their calendar — none of which a standalone no-code tool can reach.
- **Iterating on a live app safely**
  After two weeks of real use, an admin asks the builder to add a field and tighten a notification rule. The change goes through the same QA gates, and version history means the previous behavior is one restore away.

PRICING & LICENSING

Pricing	**Usage-based, billed per deployed app**

Pricing is tailored to your organization — **talk to sales for pricing & a live demo**. Final commercial terms per your agreement.

FAQ

How is App Builder different from Power Apps, Airtable, or Retool?
Those build apps next to your systems; App Builder builds them inside one. A generated app reads your real people, locations, departments, and schedules, creates platform Tasks, routes approvals through your existing approval workflow, requests e-signatures, and projects records onto the company calendar — 79 platform APIs in total, each gated on a capability the app declares. It also inherits your roles, permissions, tenant isolation, and audit trail rather than reimplementing them.

Who can use App Builder?
Building, editing, and deploying apps — plus Settings — are restricted to global administrators. A delegated App Builder app-admin can monitor: Your apps, Builds, Templates, and Analytics. Regular members never see the builder at all; they just use the apps it produces, on web and mobile.

What does a built app actually consist of?
A real app: record types with list/detail/create/edit screens, its own page at /apps/ext/<slug>, notifications and SLA/overdue logic, optional approval chains and event automations, QR and public submission links, CSV import/export, inbound webhooks with signed secrets, semantic search over its records, demo data, and mobile rendering. It is registered like any other app and enabled per business.

How do you keep AI-generated code from doing something unsafe?
Built apps run inside the MangoApps SDK sandbox with a declared-capability model — they can only call APIs the platform grants, from a fixed list of 16 capabilities, and their data access is tenant-isolated. On top of that, every build passes deterministic fit checks, an AST-level static security scan, and an AI QA review before deploy, and a per-session token budget caps runaway builds.

Do I have to accept whatever the AI produces?
No. The builder drafts a plan and pauses for your approval before writing code, and you can revise, edit, or decline that plan. After generation you can inspect and hand-edit every file, and after deploy there's version history with restore plus per-app rollback. "Fix with AI" reopens the build conversation against the live code at any time.

What happens if a built app breaks after it's deployed?
App Builder watches for it. Health problems — a quarantined app, a failed load, a suspended event handler, a rolled-back auto-update — raise a lifecycle alert to the people who built and deployed that app, filed as an Inbox action request rather than mailed to every admin in the tenant. From Your apps you can retry the load, roll back to a previous version, or reopen the build with "Fix with AI".

Can we build from our own tools instead of the browser?
Yes. App Builder issues scoped, revocable API keys and exposes an MCP endpoint, so external tools — Cursor, Claude Code, a CLI — can list tools, call them, and run read-only platform queries against your tenant. Keys track last use and can be revoked from the builder at any time.

How is it priced?
Usage-based, billed per deployed app on an annual interval — no per-employee charge and no floor, so a tenant with no deployed apps pays nothing. The AI tokens the builder spends are not billed separately here; they stay on the shared platform AI-credits meter, and the app's Analytics tab shows the spend. App Builder is off by default and an administrator opts the business in, because it deploys code into your tenant.