

Forge Primitives Demo (Jira, UI Kit latest) - One Page Summary

What it is

A Jira Forge app demo showcasing UI Kit latest primitives and common Jira integration patterns (JQL search, presets, admin utilities). It ships as a global page and an admin page with a Forge resolver backend.

Who it is for

Primary user/persona: Not found in repo (inferred: Jira admins and Forge developers evaluating UI Kit and admin tooling).

What it does (key features)

- JQL search UI with a results table (key, summary, status, assignee, updated).
- Saved JQL presets per detected project (feature flag).
- CSV export of search results (feature flag).
- Catalog of Jira Forge extension points with links to docs.
- Admin page for RBAC, feature flags, and secret management.
- App audit trail viewer with refresh and clear actions.
- Optional read-only Jira audit count fetch (requires admin scope).

How it works (architecture overview)

- UI Kit frontends: `src/frontend/index.jsx` (global page) and `src/frontend/admin.jsx` (admin page) render UI and call backend via `@forge/bridge`.
- Backend: `index.js` Forge resolver defines handlers for access control, presets, audit, secrets, and JQL search.
- Data storage: Forge storage holds config, presets, and audit log; Forge secret store holds secrets.
- External services: Jira REST API via `api.asUser().requestJira` for JQL search and audit records.
- Manifest wiring: `jira:globalPage`, `jira:adminPage`, a resolver function, and resources for UI Kit and a bundled Custom UI JQL editor.

How to run (minimal getting started)

- Install Forge CLI: `npm install -g @forge/cli@latest`
- Register: `forge register`
- Deploy: `forge deploy`
- Install to site: `forge install --site=<your-site>.atlassian.net --product=jira`
- Open in Jira: Apps - Forge Primitives Demo

Optional: Custom UI JQL editor build steps are in README (Not required for minimal run).