

# EmbedOps

Platform Overview and Security Brief

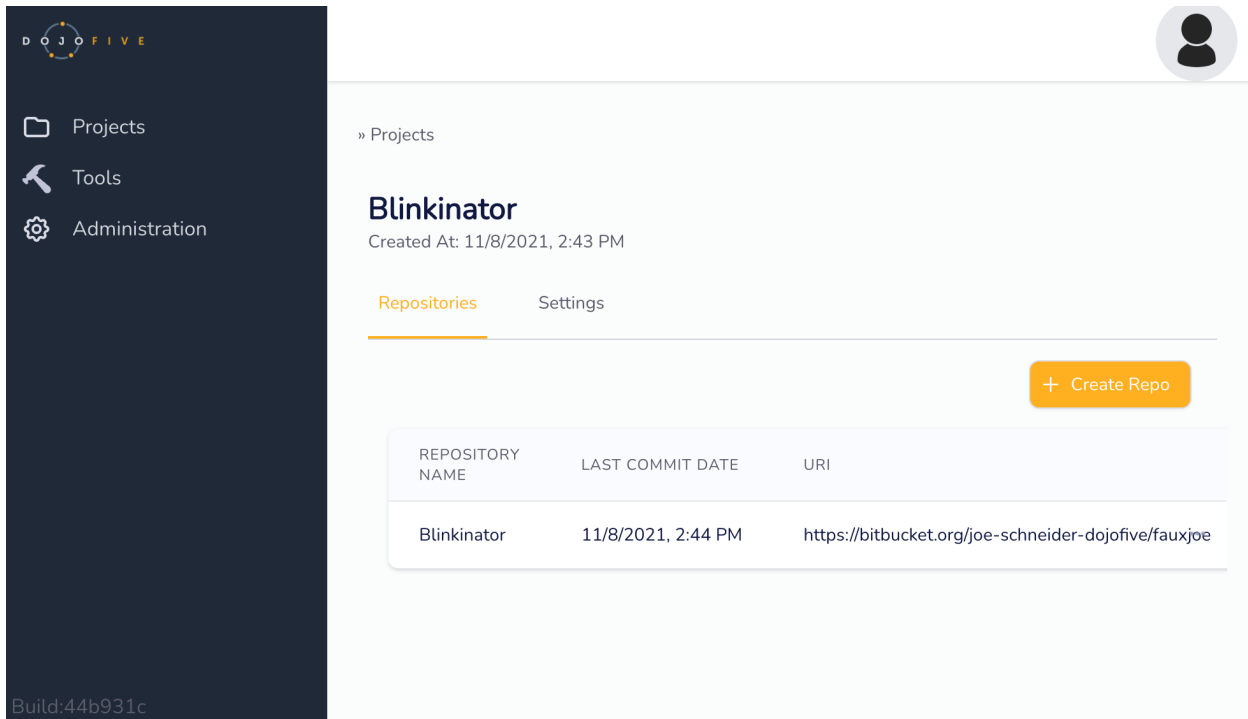


## Platform Overview

EmbedOps is a continuous build, automation, and visibility acceleration framework designed specifically for firmware development. The system is composed of a Dojo Five Docker registry, a CLI tool, a Web Dashboard, and a customer-supplied CI engine (which can be hosted on-prem or in the cloud).

## Web Dashboard

The Web Dashboard provides basic status, pipeline configuration, and debugging information.



The screenshot shows the Dojo Five Web Dashboard interface. On the left is a dark sidebar with the 'DOJO FIVE' logo and navigation links for 'Projects', 'Tools', and 'Administration'. The main content area shows the 'Blinkinator' project page, which includes a 'Created At' timestamp of '11/8/2021, 2:43 PM'. Below this are tabs for 'Repositories' (selected) and 'Settings'. A '+ Create Repo' button is visible. A table lists repository details:

REPOSITORY NAME	LAST COMMIT DATE	URI
Blinkinator	11/8/2021, 2:44 PM	https://bitbucket.org/joe-schneider-dojofive/fauxjoe

Build:44b931c

## Docker Registry

The Dojo Five docker registry is the location firmware developers and customer CI systems will pull controlled build environments for use when building development and product assets.

## CLI tool

The Dojo Five CLI tool provides developers with manual and scriptable access to local build and test environments. It requires a recent version of Python and a local Docker installation to operate correctly.

## EmbedOps Security Stance

Dojo Five's system interfaces with customer environments via three controlled access points:

- A backend API, secured with TLS 1.3 over HTTPS

- A web interface, secured with Auth0 user authentication and authorization
- A docker registry, secured with a unique per-customer token

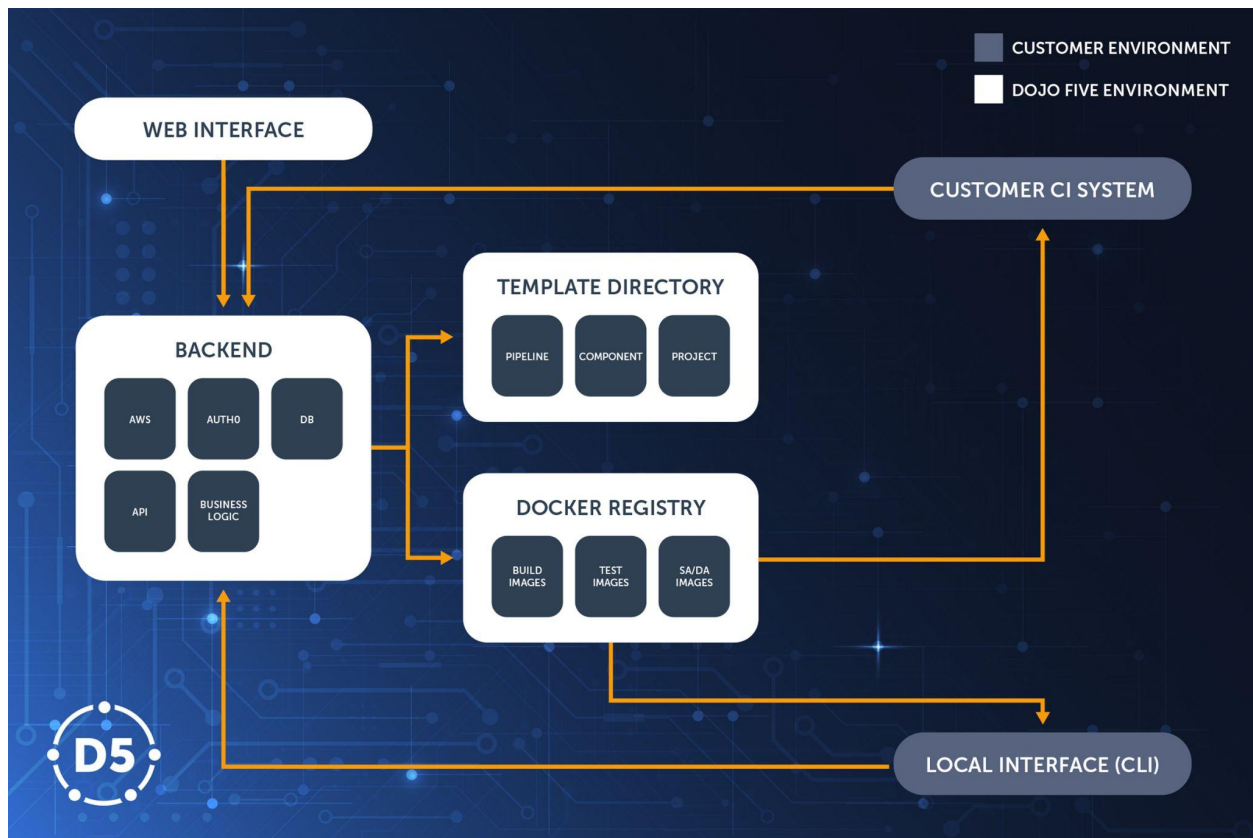
The system has been architected so that client source code / IP never is stored at-rest in Dojo Five's servers. Metadata, such as build information, logs, telemetry is stored in the backend database in order to provide dashboard functionality. This data is segregated from other customers via our backend API and database structure.

## EmbedOps Customer Expectations

Dojo Five provides on-call support according to the terms of our SLA, negotiated with each contract, in order to support and maintain the EmbedOps instance for each of our customers.

In order to provide continuous service, Dojo Five requires remote access to on-prem or cloud-hosted customer CI systems. Each customer has a primary technical contact and a primary IT contact which we use in case of system status updates, planned outages, or issues with remote connectivity.

## System Architecture



## Security Policies and Practices

The following is a tabular list of our security practices across the system.

Encryption	<ul style="list-style-type: none"><li>• All in-flight data is secured by TLS 1.3 or greater</li><li>• All at-rest metadata (build and test telemetry) is not encrypted.</li><li>• The system does not store customer source code at-rest.</li></ul>
Architecture	<ul style="list-style-type: none"><li>• React, Node.JS, PostgreSQL, Python, Docker. AWS, Auth0</li></ul>
Code Reviews	<ul style="list-style-type: none"><li>• All production code is reviewed by a secondary engineer who is not the person who wrote the code.</li></ul>
Code Testing	<ul style="list-style-type: none"><li>• All production code is unit tested according to our internal testing standards.</li></ul>
Workstation protection	<ul style="list-style-type: none"><li>• All workstations are monitored via Jamf Now.</li><li>• All workstations are protected with Bitdefender Endpoint Security.</li></ul>
System patching	<ul style="list-style-type: none"><li>• All workstation patch levels are monitored and upgraded as soon as updates are available.</li></ul>
Logging	<ul style="list-style-type: none"><li>• Failures on the web or in infrastructure are monitored using Sentry.io and reported to the team.</li></ul>