

# DevOps & Releases

Last updated by | Daniels, Steve | Sep 30, 2025 at 8:51 PM GMT+5:30

## Contents

- [Controls](#)
  - [Change Control Process](#)
  - [Deployment Pipeline Controls](#)
  - [Pull Request Controls](#)
- [Pipelines](#)
  - [Vulnerability Scanning \(Dependabot\)](#)
  - [Quality Checks](#)
  - [Scheduled Health & E2E Tests](#)
  - [Deployment pipelines](#)
  - [Release processes](#)
    - [iHub Release Process](#)
    - [MEA Release Process](#)
    - [Pricing APIs \(Dataiku\) Release Process](#)
    - [Python APIs Release Process](#)
    - [Referrals Release Process](#)
    - [Q Web App / API Release Process](#)
    - [Workbench Release Process](#)
    - [ODS Release Process](#)

## Controls

### Change Control Process

- [IQUW IT Change Management Policy v3.6.docx](#)

### Deployment Pipeline Controls

All deployment pipelines must implement the following controls:

- Prior to deploying to any environment the app must check:
  - Build successfully
  - All Tests pass
  - All Linting pass
  - All Code scanning checks pass
- Deployment to non-production environments must be approved by at least one engineer

- Deployment to pre-production/production environments cannot be approved by the user who initiated the deployment
- Deployments to production require cross team approval (i.e. Tester + Developer + Platform) with at least 1 approval from a minimum of 2 teams

## Pull Request Controls

All code repositories must implement the following controls

- All changes to any `main` branch must be made via a Pull Request
- Pull Request cannot be merged unless peer reviewed by at least one other engineer
- Pull Requests cannot be merged until all Security & Testing checks pass
- Pull Requests cannot be merged if any outstanding comments or discussions made on a given Pull Request have been resolved
- Pull Requests must have a work item linked
- All changes must meet the relevant Coding Standards

[#58031] Policy Admin API V2 - Transition Policy Section, Declined

Completed! 18633 Squire, Jonny proposes to merge [feature/58031-policy-transition-declined](#) into [main](#) All comments resolved

Overview Files Updates Commits Conflicts

Daniels, Steve completed this pull request 2 Apr

Merged PR 8633: [ #58031 ] Policy Admin API V2 - Transition Policy Section, Declined

Show details

All required checks succeeded  
Optional check succeeded

View 3 checks

2 reviewers approved

No merge conflicts  
Last checked 2 Apr

**Reviewers**

Required

No required reviewers

Optional

- ✓ IH Integration Hub Approved via Ilyas, Farooq
- ✓ DS Daniels, Steve Approved
- ✓ IF Ilyas, Farooq Approved
- RN Rayan, Nour No review yet

Tags

# Checks



The displayed list of checks may be truncated for optimal performance. For a smoother experience, please maintain a reasonable number of policies (fewer than 100).

## Required



**policy-admin-service PR**

Build succeeded



**Work items must be linked**

Succeeded

## Optional



**[Policy Admin Service (v1)] Quality Gate passed**

Succeeded



**[Policy Admin Service (v2)] Quality Gate passed**

Succeeded



**Comments must be resolved**

Succeeded

# Pipelines

We use Azure DevOps pipelines to handle releases and other automated tasks such as PR requests or health checks

## Vulnerability Scanning (Dependabot)

See: [Dependabot Pipelines](#)

Work items:

- [ ] 48313 Vulnerabilities scanning (iHub) | Done
- [ ] 70762 Vulnerabilities Scanning (Python) | Done
- [ ] 67525 Vulnerabilities Scanning (Azure Functions) | Done
- [ ] 70763 Vulnerabilities Scanning (React) | Done
- [ ] 81543 Vulnerability Scanning (Dataiku) | New

We scan our repos for vulnerabilities using Dependabot periodically.

PR are created automatically by the tool with updated packages which are trigger [Quality Checks](#) and are then peer-reviewed by Developers.

The screenshot shows the Azure DevOps interface for a specific run. On the left, there's a sidebar titled "Jobs in run #20250..." with a list of tasks: Dependabot (3m 45s), Initialize job (<1s), Checkout user-se... (5s), GoTool (<1s), Dependabot (3m 39s), Post-job: Check... (<1s), Finalize Job (<1s), and Report build sta... (<1s). The main area is titled "Dependabot" and displays a log of its operations. The log entries are numbered 7731 through 7754. It shows Dependabot processing pull requests for various dependencies like dotenv, express, and express-openapi-validator, pushing changes, creating commits, and merging them into the main branch. Each entry includes a brief description of the action taken.

## Quality Checks

See: [Quality Check Pipelines](#)

Work items:

- [ ] 65980 Add 100% test coverage for new code to all PRs (iHub) | Done
- [ ] 69394 Add 100% test coverage for new code to all PRs (Python) | Done
- [ ] 70764 Add 100% test coverage for new code to all PRs (Azure Functions) | Done
- [ ] 70765 Add code coverage reports and ensure 100% test coverage to all PRs (React) | Done
- [ ] 81544 Add 100% test coverage for new code to all PRs (Dataiku) | New
- [ ] 55027 Sonar Cloud pipeline scanning (iHub) | Done
- [ ] 70759 Sonar Cloud pipeline scanning (Python) | Done
- [ ] 70760 Sonar Cloud pipeline scanning (Azure Functions) | Done

- | 70761 Sonar Cloud pipeline scanning (React) | Done
- | 81545 Sonar Cloud pipeline scanning (Dataiku) | New
- | 70767 Automated tests for React Apps (unit/integration test) | Done

We check the code of our repos before and after merging PRs.

Quality checks include:

- Linting
- Building
- Unit Testing
- BDD Testing
- Code Coverage
- Code Scanning (SonarCloud)
- Terraform Coding Testing

← **Jobs in run #20250530.8**  
ihub-service Quality Check

Quality Checks		
✓ CI	49s	
Initialize job	<1s	
Checkout ihub-service@refs/pull/...	1s	
UseNode	<1s	
Audit production dependencies	13s	
Install dependencies	13s	
Linting	3s	
Build	16s	
Post-job: Checkout ihub-service...	<1s	
Finalize Job	<1s	
> Unit Tests	29s	
> BDD Acceptance tests	28s	
> SonarQube	33s	
PR Terraform Check		
> Plan to dev	58s	
Finalize build		
Report build status	<1s	

✓ **Finalize Job**

- Starting: Finalize Job
- Cleaning up task key
- Start cleaning up orphan processes.
- Finishing: Finalize Job

## Scheduled Health & E2E Tests

See: [Scheduled Health & E2E Test Pipelines](#)

Work items:

- 70862 API Healthchecks (iHub Endpoints) | Done
- 61700 API Healthchecks (Dataiku Pricing Endpoints) | Done
- 70861 API Healthchecks (Python Endpoints) | Done

We periodically run checks on our systems to ensure that they are working as expected.

Scheduled Health & E2E Checks include:

- Calling health endpoints on our APIs
- Running automated E2E tests on our front end applications

## Deployment pipelines

See: [Deployment Pipelines \(iHub/Python/MEA\)](#)

See: [Deployment Pipelines \(Dataiku\)](#)

See: [Deployment Pipelines \(Q\)](#)

Work items:

- TBC

We have a diverse tech stack at 1856, with a variety of in-house and SaaS/IaaS applications.

Deployments for our internal applications and APIs are handled independently and co-ordinated across teams.

## Release processes

### iHub Release Process

[iHub \(Integration Hub\)](#) is deployed as follows:

1. Create a [Semantic Versioning](#) tag in the format `vMAJOR.MINOR.PATCH` where `MAJOR` is APIM API version (i.e. `v1.2.1`), on the `main` branch for a given microservice
2. A deployment pipeline will be triggered (see: [Deployment Pipelines \(iHub/Python/MEA\)](#))
3. All checks will need to pass and the deployment will need to be approved to be progressed to dev/test environments
4. Any deployment to production environments needs approval and have further checks (see: [Change Control Process](#) & [Deployment Pipeline Controls](#))

### MEA Release Process

[Mea - Overview](#) is deployed as follows:

1. Trigger deployment for [Pipelines - Runs for mea-ingestion Deploy](#) and/or [Pipelines - Runs for mea-processing Deploy](#) as required
2. All checks will need to pass and the deployment will need to be approved to be progressed to dev/test environments
3. Any deployment to production environments needs approval and have further checks (see: [Change Control Process](#) & [Deployment Pipeline Controls](#))

### Pricing APIs (Dataiku) Release Process

For the [Dataiku Pricing API](#):

- All merges to `main` for a given Pricing API repo will deploy to the dev environment (for Dataiku Devs only)

To proceed to deploy to higher environment that proxied via iHub's [Dataiku Gateway - Overview](#):

1. Trigger [Dataiku Service Deployment Pipeline](#)
2. All checks will need to pass and the deployment will need to be approved to be progressed to dev/test environments
3. Any deployment to production environments needs approval and have further checks (see: [Change Control Process](#) & [Deployment Pipeline Controls](#))

See [Dataiku Pipelines - Overview](#) for more details

## Python APIs Release Process

1. Create a [Semantic Versioning](#) tag in the format `vMAJOR.MINOR.PATCH` where `MAJOR` is APIM API version (i.e. `v1.2.1`), on the `main` branch for a given Python App built using the [Python Starter Kit - Overview](#)
2. A deployment pipeline will be triggered (see: [Deployment Pipelines \(iHub/Python/MEA\)](#))
3. All checks will need to pass and the deployment will need to be approved to be progressed to dev/test environments
4. Any deployment to production environments needs approval and have further checks (see: [Change Control Process](#) & [Deployment Pipeline Controls](#))

## Referrals Release Process

- Referrals code changes follow [Dataiku Release Process](#)
- Referrals dataset changes are released using the following procedure:
  1. Non-prod releases
    - o Releases to Dev and Staging envs are described [here](#)
  2. Production releases
    1. In Azure navigate to Storage browser -> Subscription: Service Model NonProd -> [Account: stpricingdluksouthdev](#) -> Blob container: `uwauthority` (Make sure your ZScaler connection is authenticated).
    2. Go into the target environment folder (PreProd or Prod), select the master `underwriting_matrix_processed.parquet` file and create a backup, by clicking on the ... menu and choosing the `clone` option. Backups are identified by a suffix containing a timestamp in the `-YY-MM-`

DD-HH:mm:ss format in UTC time.

Authentication method: Microsoft Entra user account (Switch to Access key)

Search blobs by prefix (case-sensitive) Only show active objects

Showing all 12 items

<input type="checkbox"/> Name	Last modified	Access tier	Blob type	Size	Lease state
<input type="checkbox"/> [..]					
<input type="checkbox"/> 2024	2/27/2025, 4:09:10 PM				
<input type="checkbox"/> underwriting_matrix_processed-2025-01-23-150700.parquet	1/8/2025, 4:57:42 PM	Hot (Inferred)	Block blob	25.66 KiB	
<input type="checkbox"/> underwriting_matrix_processed-2025-02-19-143400.parquet	1/23/2025, 5:19:51 PM	Hot (Inferred)	Block blob	20.11 KiB	
<input type="checkbox"/> underwriting_matrix_processed-2025-02-27-150700.parquet	2/19/2025, 3:35:26 PM	Hot (Inferred)	Block blob	20.15 KiB	
<input type="checkbox"/> underwriting_matrix_processed-2025-04-16-031200.parquet	2/27/2025, 4:08:48 PM	Hot (Inferred)	Block blob	20.17 KiB	
<input type="checkbox"/> underwriting_matrix_processed-2025-04-30-111300.parquet	4/16/2025, 2:11:37 PM	Hot (Inferred)	Block blob	20.69 KiB	
<input type="checkbox"/> underwriting_matrix_processed-2025-05-13-140700.parquet	5/13/2025, 3:07:10 PM	Hot (Inferred)	Block blob	20.69 KiB	
<input type="checkbox"/> underwriting_matrix_processed-2025-06-05-150400.parquet	6/5/2025, 4:04:45 PM	Hot (Inferred)	Block blob	21.01 KiB	
<input type="checkbox"/> underwriting_matrix_processed-2025-06-10-140500.parquet	6/10/2025, 3:05:30 PM	Hot (Inferred)	Block blob	21.02 KiB	
<input type="checkbox"/> underwriting_matrix_processed-2025-07-09-141700.parquet	7/9/2025, 3:17:44 PM	Hot (Inferred)	Block blob	21.13 KiB	
<input type="checkbox"/> underwriting_matrix_processed-rollback-2025-04-30-113000....	4/30/2025, 12:13:48 PM	Hot (Inferred)	Block blob	20.99 KiB	
<input checked="" type="checkbox"/> underwriting_matrix_processed.parquet	7/9/2025, 3:18:01 PM	Hot (Inferred)	Block blob	21.21 KiB	Available

3. Go back to uwauthority blob storage and click into one environment lower (Staging -> PreProd, PreProd -> Prod). Make sure the lower environment has been deployed previously. Select the master underwriting\_matrix\_processed.parquet file and click the Copy button in the top bar.

Add Directory Upload Refresh | Delete Copy Paste Rename Acquire lease Break lease Edit columns

Blob containers > uwauthority > Preprod

Authentication method: Microsoft Entra user account (Switch to Access key)

Search blobs by prefix (case-sensitive) Only show active objects

Showing all 14 items

<input type="checkbox"/> Name	Last modified	Access tier	Blob type	Size	Lease state
<input type="checkbox"/> [..]					
<input type="checkbox"/> 2024	2/26/2025, 5:18:33 PM				
<input checked="" type="checkbox"/> underwriting_matrix_processed.parquet	7/2/2025, 3:08:06 PM	Hot (Inferred)	Block blob	21.21 KiB	Available
<input type="checkbox"/> underwriting_matrix_processed_2025-01-17-125900.parquet	12/18/2024, 2:43:51 PM	Hot (Inferred)	Block blob	25.66 KiB	Available
<input type="checkbox"/> underwriting_matrix_processed_2025-01-21-170000.parquet	1/17/2025, 1:59:20 PM	Hot (Inferred)	Block blob	20.99 KiB	Available
<input type="checkbox"/> underwriting_matrix_processed_2025-02-11-175600.parquet	1/21/2025, 6:00:17 PM	Hot (Inferred)	Block blob	20.11 KiB	Available

4. Once again go into the target environment folder and click the Paste button in the top bar. When prompted, select Replace from the pop up dialog. The updated master file will be automatically picked up by the API code.

Copy blob conflict

A blob named with the name 'underwriting\_matrix\_processed.parquet' already exists at the destination. How would you like to resolve the conflict?

Replace Keep Both Cancel

Show 14 items

Notes:

- Code for release automation exists in several shapes and forms, but is not used for Prod releases due to authorization concerns, since the solution relies on Dataiku functionality, and the control mechanisms are not ideal. It shouldn't be too hard to replace the manual procedure with a ADO pipeline, if need be.

## Q Web App / API Release Process

(TBC)

### Workbench Release Process

Administrator->UDLC Toolkit-> Release Management

Click-> Create New Release

Version-> Enter the version of the release based on Sprint. Eg: 12.5

Name-> Enter release name. Eg: "release\_12.5 Prod"

Notes-> This is optional.

Client-> Select "ers"

Environment-> Select "production"

Select Workspace applications modules. Full and partial selection can be done. In very rare cases partial releases are done.

After all selection of Workspace applications modules click on Create new Release

From Release dashboard select newly created release.

Click on Promote button.

Once all the Workspace applications status turned into green,click on Finish button.

More information on Release Management can be read [here](#)

### ODS Release Process

(TBC)