DIBBs eCR Viewer

Technical Acceptance Testing Guide - Azure

Release 6.0.16.2-GA

Created Date: October 29, 2024

Last Updated: December 3, 2024

Prepared by: U.S. Centers for Disease Control and Prevention

Version History

Version	Author	Reviewed By	Revision Date	Revision Notes
1.0	Emma Stephenson	Matt Goldberg, Mary Yeh	10/29/2024	Draft document created in support of 6.0.16.2 beta release
2.0	Emma Stephenson	Rebecca Fisher	11/14/2024	Revisions to support the 6.0.16.2 GA release
3.0	Emma Stephenson	Michael Wodajo	12/3/2024	Final revision for 6.0.16.2 GA release

Table of Contents

1   Overview

After you've installed DIBBs services in your jurisdiction, this guide walks you through the steps needed to verify the installation.

This will ensure that the system has been properly installed in your environment and that it's ready for your public health department to interact with.

2   Prerequisites

- ☐ You've installed DIBBs services using the DIBBs Azure Terraform repository.
- ☐ You've checked that the following environment variables have been set within the eCR Viewer container App in Azure:
  - ☐ AZURE_STORAGE_CONNECTION_STRING
  - ☐ AZURE_CONTAINER_NAME
  - ☐ SOURCE
  - ☐ NEXT_PUBLIC_NON_INTEGRATED_VIEWER
  - ☐ APP_ENV
  - ☐ NODE_ENV
  - ☐ NEXT_PUBLIC_BASEPATH
- ☐ You have access to the sample eCR files.

3   Acceptance Checklist

The following instructions will guide you through this list. Mark the checklist as you complete each step.

- ☐ Verified that the Orchestration health check is available
- ☐ Verified that the Orchestration pipeline run was successful
- ☐ Verified that the eCR FHIR bundle is available in blob storage
- ☐ Verified that the eCR Viewer health check is available
- ☐ Verified that the processed eCR is visible in the viewer

4   Installation Testing

To test the installation, we'll run a series of API requests to the DIBBs services. These requests can also be found in a Postman collection.

{{dibbs-url}} will be used throughout the document to indicate the URL your services are available on — use your own URL instead.

4.1   Orchestration

4.1.1   Health Check

Verify that the Orchestration service is available to receive requests.

Run the Orchestration Health Check request:

```
curl --location 'http://{{dibbs-url}}/orchestration'
```

You should receive the following response:

```
{ "status": "OK" }
```

4.1.2   Pipeline Run

Now, we need to verify that an eCR can run through the DIBBs pipeline.

Run the Process eCR request:

```
curl --location 'http://{{dibbs-url}}/orchestration/process-zip' \
--form 'message_type="ecr"' \
--form 'include_error_types="[errors]"' \
--form 'config_file_name="sample-orchestration-azure-config.json"' \
--form 'data_type="zip"' \
--form 'upload_file=@"{{link-to-eCR-zip=file}}";type=application/zip'
```

If running through Postman, please upload your own sample eCR (zip) under Body → upload_file.

You should see the following response:

```
{
      "message": "Processing succeeded!",
      "processed_values": {
            "message": "Parsing succeeded!”,
            “parsed_values”: {
                  [parsed values]
            }
      }
}
```

4.1.3   Check FHIR Bundle Storage

To confirm that the eCR processed through the DIBBs pipeline correctly, we'll manually check the output FHIR bundles in blob storage.

In Azure, navigate to the DIBBs storage account, go into the data storage container, and view all records. You should see the eCR that was just processed stored in blob storage.

Take note of the eCR file name `{{ecr-id}}.json` in blob storage — you'll need it for the next step.

4.2   eCR Viewer

4.2.1   Health Check

Verify that the eCR Viewer service is available to receive requests.

Run the eCR Viewer Health Check request:

```
curl --location 'http://{{dibbs-url}}/ecr-viewer/api/health-check'
```

You should receive the following response:

```
{ "status": "OK" }
```

4.2.2   View Processed eCR

Verify that the eCR Viewer is able to display the eCR that was processed.

To access that eCR, run the following request:

```
http://{{dibbs-url}}/ecr-viewer/view-data?id={{ecr-id}}
```

where `{{ecr-id}}` should be replaced with the eCR ID you'd like to view. You can find the `ecr-id` by navigating to the blob storage container and copying the file name, which should be `{{ecr-id}}.json`

4.3   Debugging

If you run into any issues during testing, please reach out to the DIBBs team! We want to know if there are changes needed to the core pipeline. You can email us at dibbs@cdc.gov.

If your eCR contains invalid data, it will be rejected by the pipeline. You'll see a message like:

```
{
        "message": "Processing failed",
        { ...more details about the failure...}
}
```
Or:

```
Internal Server Error
```

Generally speaking, you can get around this error by re-running or choosing a different eCR bundle. You'll also need to make sure your eCRs are named as they are coming out of AIMs — `CDA_eICR.xml` and `CDA_RR.xml`.

5   Next Steps

Congratulations! You've completed the verification process for the DIBBs installation. Please reach to your contact within your public health department and dibbs@cdc.gov to confirm that the installation was successful.