

GitHub Migration Summary Report

<Product Name>

<Subcontractor Name >

<Performance period>

<Name of submitter>

<Email address of submitter>

<Date submitted>

1. Personnel Who Worked on Migration

List all the team members who worked on the migration effort.

| Name | Title/Role | Contact Information |
|------|------------|---------------------|
| | | |
| | | |
| | | |

Insert additional rows as needed.

2. Migration Overview

Insert a brief summary of the migration of the product and current status. Include a description of the structure and contents of the repository or repositories created in GitHub. Report on the current status of the build and the performance status of any automated build and test sequences.

3. Description of Work Completed

List all the activities that were performed to complete the migration (preferably in the order they were performed). Account for any automated tools that were created and/or used in support of this process.

| ID | Activity | Description | Time Spent |
|----|----------|-------------|------------|
| | | | |
| | | | |
| | | | |

Insert additional rows as needed.

4. Migration Artifacts

List all the artifacts of the product with appropriate locations.

| ID | Artifact | Location(s) |
|----|--|-------------|
| 1 | Source code | |
| 2 | Documentation (including repository-based, wiki, other) | |
| 3 | Data (test data and seed data) | |
| 4 | Issue Tracker | |
| 5 | Binaries and installation packages (If there are any binaries and installation packages, please include these links in README.txt) | |
| 6 | Release notes | |
| 7 | Production / Development /QA URLs if deployed at NCI | |
| 8 | Other useful links, if any | |

Insert additional rows as needed

5. Issues

List all the issues that were encountered during the migration and the steps taken to resolve them.

| ID | Issue | Status | Resolution/Proposed Resolution |
|----|-------|--------|--------------------------------|
| | | | |
| | | | |
| | | | |

Insert additional rows as needed.

6. Success factors

List or describe what went right during the entire migration process.

| ID | Success factor |
|----|----------------|
| | |
| | |
| | |

Insert additional rows as needed.

7. Lessons learned / Future Improvements

List the lessons learned during the migration process including things that you would do differently if you have to do migration of other tool and suggestions to other teams

| ID | Lessons Learned/Enhancements/Improvements |
|----|---|
| | |
| | |
| | |
| | |
| | |

Insert additional rows as needed.

8. Verification (Please refer to <https://github.com/NCIP/ncip.github.com/wiki/Migration-Checklists#wiki-quality-assurance> for more details)

| ID | Verification Check | Result |
|----|--|--------|
| 1 | Check if a 1-2 line description is added to the repository | |
| 2 | Check that the repo clones correctly and leaves "master" checked out. <code>\$ git clone https://github.com/NCIP/<PROJECTNAME>.git</code> | |

| | | |
|----|---|--|
| | <pre>\$ cd <PROJECTNAME></pre> <pre>\$ git status</pre> | |
| 3 | Check that the LICENSE, NOTICE, and README files appear in the master branch | |
| 4 | Check that the LICENSE.txt file also appears in the gh-pages branch | |
| 5 | Check that the LICENSE, NOTICE, and README template placeholders have been replaced | |
| 6 | Check that all links in the README file work | |
| 7 | Check that the text in LICENSE, NOTICE, and README files matches the text provided in the appropriate templates at https://github.com/NCIP/ncip.github.com/wiki/Templates | |
| 8 | Check files of several source code types to ensure the appropriate copyright header is placed and template placeholders have been replaced | |
| 9 | Check that the URL to the LICENSE.txt file works | |
| 10 | Check that the URL to the LICENSE.txt file is of the form: <pre>http://ncip.github.com/<PROJECTNAME>/LICENSE.txt</pre> | |
| 11 | Do a web search for the tool and check that the appropriate links are found for User Documentation Reference Web Sites Wiki Documentation GitHub repository Knowledge Center | |
| 12 | Check that Subversion revision numbers are mentioned in the commit messages for future reference. Run: <pre>\$ git log</pre> and look for commit message lines containing the corresponding Subversion revision numbers. For example: SVN-Revision: 1234 | |
| 13 | Check that a README file appears in the corresponding docs and data | |

| | | |
|----|---|--|
| | directories. | |
| 14 | Check that the format of README file in the corresponding docs repository match the template at https://github.com/NCIP/ncip.github.com/wiki/Templates#wiki-readme-for-documents and the template tokens are replaced. | |
| 15 | Check that the format of README file in the corresponding data repository matches the template at https://github.com/NCIP/ncip.github.com/wiki/Templates#wiki-readme-for-data and the tokens are replaced. | |
| 16 | Check that format of CONTRIBUTING.rst file in the main repository matches the template at https://github.com/NCIP/ncip.github.com/wiki/Templates#wiki-contributing | |
| 17 | Inform the PO to create a helpdesk ticket to remove write access to all the user of the repository. | |

Insert additional rows as needed.