

Project Idea

The Jenkins Pipeline Step reference is difficult to read, even to the initiated, mostly due to its format. This project aims to improve the way the documentation is formatted.

For example, when you read the [readFile step documentation](#), the documentation format makes it difficult to see that the obvious usage is `: readFile(String file, String encoding)`. When you read the [checkout documentation](#), you are shown a single web page that when printed would be 62 pages long. That is too large a page for usable navigation. We should consider logical partitions of that page, possibly with a single page per SCM provider (like checkout-git, checkout-p4, checkout-svn, checkout-cvs).

For this project, the student is expected to study the [documentation generator](#) code base, the [documentation feedback](#) from readers, and to propose enhancements to the documentation generator code to improve the format, layout, and readability of the documentation published on-line. The student is not expected to contribute to the content of the documentation, however, technical additions that enhance the discoverability of the documentation via annotations or other special markup is considered valid for a coding project.

The student is also expected to study [Stapler](#) and [Jelly](#), as these technologies are used to capture documentation as well. Documentation can be extracted from Stapler and Jelly.

A potential stretch goal could be something like an embedded pipeline steps generator, though that might be outside the scope of the website.

Links

- Pipeline documentation: <https://jenkins.io/doc/book/pipeline/>
- Example of documentation for [readFile](#) step
- A related proposal was made some time ago: [Former proposal](#).
- <https://github.com/jenkins-infra/pipeline-steps-doc-generator/>
- <https://github.com/jenkins-infra/pipeline-library>
- Examples of remote documentation:
 - Pipeline: AWS steps adds documentation to the [README](#)
 - JIRA Pipeline steps keeps their documentation in their repo, and displays it as a [weblink](#).

Newbie Friendly issues

- Fix bugs like [JENKINS-21311](#).
- List of issues related to [documentation generation](#).
- [INFRA-1717](#) - Javadoc landing page
- <https://github.com/jenkins-infra/pipeline-library/pull/22> Review and help merge the PR to generate Javadoc for Pipeline steps
 - Component Java Doc: <https://javadoc.jenkins.io/component/>
 - Core Java doc: <https://javadoc.jenkins.io/component/>
 - Plugin java docs: <https://javadoc.jenkins.io/plugin/>
 - The java docs cannot be searched easily (e.g. try to find `org.jenkinsci.Symbol`)

Skills to study/improve

- Java
- Jenkins Pipeline
- HTML
- CSS
- AsciiDoc
- JavaScript

Project Metadata

Created on: January 2018

Title: Jenkins Pipeline Step Documentation Generator Improvements

Goal: Enhance the Jenkins Pipeline documentation generator to produce better documentation for thousands of Pipeline developers

Champion: Martin d'Anjou

Champion Github Id and link: <https://github.com/martinda>

Champion Time Zone: UTC-4

Champion Role: I am a making this proposal as a **mentor**

Technical advisor: TBD

Advisor Github Id and link: TBD

Advisor Time Zone: TBD

Project Category: Dev Tools

SIG/Subproject: pipeline-authoring

Project Gitter chat room: <https://gitter.im/jenkinsci/pipeline-authoring-sig>

Potential Mentors

The potential mentors are:

- 1) Kristin Whetstone (<https://github.com/kwhetstone>)
- 2) Mark Waite (<https://github.com/MarkEWaite>)
- 3) Tasneem Koushar (<https://github.com/koushartasneem>)