

Online UI/UX Hackfest

Kick-Off Session
(May 25, 2020)



#jenkinsIsTheWay

Agenda

- Hackfest Introduction
- How to contribute? How to record contributions?
- Track introductions by track leaders
- Q&A

Slides: See the Gitter chat
<https://gitter.im/jenkinsci/hackfest>

Jenkins Online Meetup

Anything about Jenkins

- <https://www.meetup.com/Jenkins-online-meetup>
- <https://jenkins.io/events/online-meetup/>

“By contributors for contributors”

- Show and tell
- Minimum slides
- Live demos!
- Everyone can participate



Sponsors



CD.FOUNDATION



CloudBees®



Jenkins

Questions and Feedback

Online

- Zoom Q&A
- Hackfest office hours

Offline

- Gitter: [jenkinsci/hackfest](https://gitter.im/jenkinsci/hackfest)
- Mailing lists



JENKINS IS THE WAY

—— UI/UX Hackfest ——

- Improve user interface
- Improve user documentation
- Spread the word
- Get swag and prizes!

<https://www.jenkins.io/events/online-hackfest/2020-uiux/>

#jenkinsIsTheWay

What is “Hackfest”?

- Hackfest **!=** Hackathon
- Spend as much time as you can/want
- Focus on learning and sharing experiences
- Focus on resolving your own UX issues
- Newcomers are welcome!



Thanks to the org team and contributors!



Oleg Nenashev



Mark Waite



Tracy Miranda



Tim Jacomb



Marky Jackson



Alyssat

+ SIG members

- [User Experience](#)
- [Documentation](#)
- [Advocacy and Outreach](#)



Thanks to the sponsors!



CD.FOUNDATION



CloudBees®



Jenkins

How to participate?



**Jenkins
needs you!**

How to get started?

1. [Register to the event](#) if you have not done it already.
2. Join [our Gitter channel](#) - Q&A and discussions
3. Join/watch the [kick-off meeting](#)
 - a. Join in person: May 25, 8AM UTC or 1PM UTC
 - b. OR: Watch the recording
4. Review and choose projects
5. You are all set, start contributing!



Participate and Contribute

There are many ways to engage with the Jenkins project and community. Here are the most common ways to get you started. Welcome aboard!

MEET

Meet other Jenkins users and share your experiences by organizing and attending events and meetups.

[Read More](#)

CODE

Do you enjoy writing code? There are numerous plugins and components for you to contribute to.

[Read More](#)

HELP

As an experienced user, you can help others get the most out of Jenkins.

[Read More](#)

TRANSLATE

If you're fluent in languages other than English, consider improving support for those languages.

[Read More](#)

TEST

You can help prevent regressions by contributing automated tests.

[Read More](#)

DOCUMENT

Improve the documentation to make it easier for others to get started.

[Read More](#)

DESIGN

As it is intended for daily use by finicky web developers, design is essential

[Read More](#)

REVIEW

Help review changes to code or documentation.

[Read More](#)



Jenkins

<https://jenkins.io/participate>

UI/UX Hackfest: What can you do?

- Development: Java, JavaScript, Go, ...
- Documentation: Markdown and AsciiDoc
- UX Testing
- Code and Documentation reviews
- Creating content for users: blogs, videos, etc.
- Help others
- ...



**Jenkins
needs you!**

Tracks

User Interface

- Look & Feel
- UI Accessibility
- Read-only UI (JEP-224)
- UI Themes
- Pipeline visualization
- ...

User Documentation

- User Guides, Tutorials
- Admin Guides
- Installation Guides
- Solution pages
- Reviews
- ...

Spread the Word

- Share your stories
- Create user content: blogs, videos, etc.
- ...

<https://www.jenkins.io/events/online-hackfest/2020-uiux/#tracks-and-project-ideas>

Off-track UX contributions

- It is possible!

Reporting contributions during the hackfest

GitHub Issues in [jenkinsci/ui-ux-hackfest-2020](https://github.com/jenkinsci/ui-ux-hackfest-2020)



UI/UX Enhancement

Let us know about a UI/UX enhancement in Jenkins you've submitted!

[Get started](#)

UI/UX Bug Fix

Let us know about a UI/UX bug in Jenkins you have fixed!

[Get started](#)

Documentation

Let us know about a documentation contribution (optional - we will track GitHub)

[Get started](#)

Spread The Word

Let us know about your contributions: user stories, posts and videos about Jenkins

[Get started](#)

UX Testing

Let us know about a User Experience testing you have done for Jenkins UI and User Documentation projects

[Get started](#)

Reviews

Helped to review code or documentation changes? It is a contribution as well!

[Get started](#)

Other UX Contributions

Report other contributions related to the Jenkins User Experience stories

[Get started](#)

Communication channels

- Chat: [UI/UX Hackfest Gitter channel](#)
- Mailing lists
 - a. [User Experience](#), [Documentation](#), [Advocacy & Outreach](#) - for tracks
 - b. [Developer mailing list](#) for common matters
- Office Hours
 - a. Daily at 8AM UTC
 - b. 2PM UTC on Tuesday, Wednesday, and Thursday

Events

Mon 5/25	Tue 5/26	Wed 5/27	Thu 5/28	Fri 5/29
08:00 – 09:00 UI/UX Hackfest: Early Bird https://zoom.us/j/92352421942?	08:00 – 09:00 UI/UX Hackfest Office Hours 09:00 – 10:00 Migrating plugin documentation to Documentation-as-Code	08:00 – 09:00 UI/UX Hackfest Office Hours 10:00 – 11:00 Hands-on: How to use and create Jenkins themes?	08:00 – 09:00 UI/UX Hackfest Office Hours 14:00 – 15:00 UI/UX Hackfest Office Hours	08:00 – 09:00 UI/UX Hackfest Office Hours 13:00 – 16:00 RESERVED: Closing Hackfest Sessions
13:00 – 14:30 Jenkins UI/UX Online Hackfest: Kick-off Zoom Webinar, RSVP to get the link	13:00 – 14:00 SystemRead permission demo, and how to contribute 14:00 – 15:00 UI/UX hackfest office hours	 15:00 – 16:00 Mid-term updates: UX SIG meeting		
15:00 – 16:00 Hands On: Beautify the user Zoom Webinar, RSVP to see the link	15:00 – 16:00 Migrating documentation to jenkins.io			
16:00 – 16:45 Jenkins Is The Way: Program Intro				

[Calendar Link](#)



Jenkins

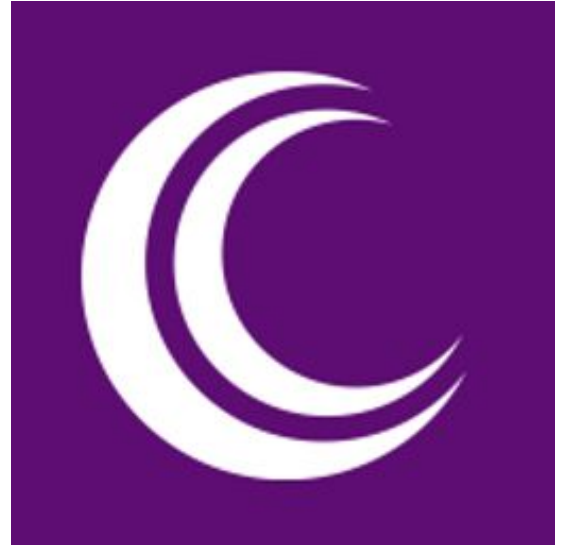
Online meetups

- All key sessions will be hosted as online meetups
 - Zoom Webinar or Zoom
 - <https://www.meetup.com/Jenkins-online-meetup/>
- More sessions will be announced soon!
 - Subscribe to the online meetup!



Code of Conduct

- <https://jenkins.io/conduct>
- Based on Contributor Covenant 1.3
- TL;DR: Be nice!



Quick links

- [Code of Conduct](#)
- [Community Spaces](#)
- [Reporting](#)
- [Handling of violations](#)

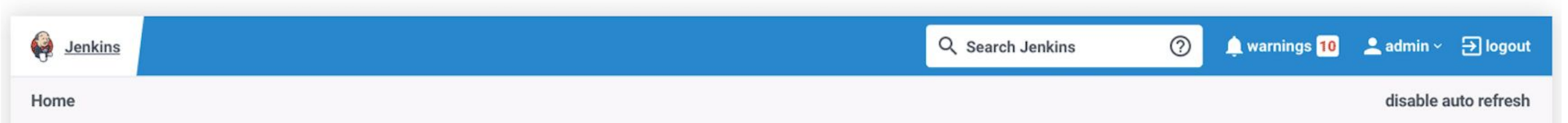
Track and project introductions



User Interface

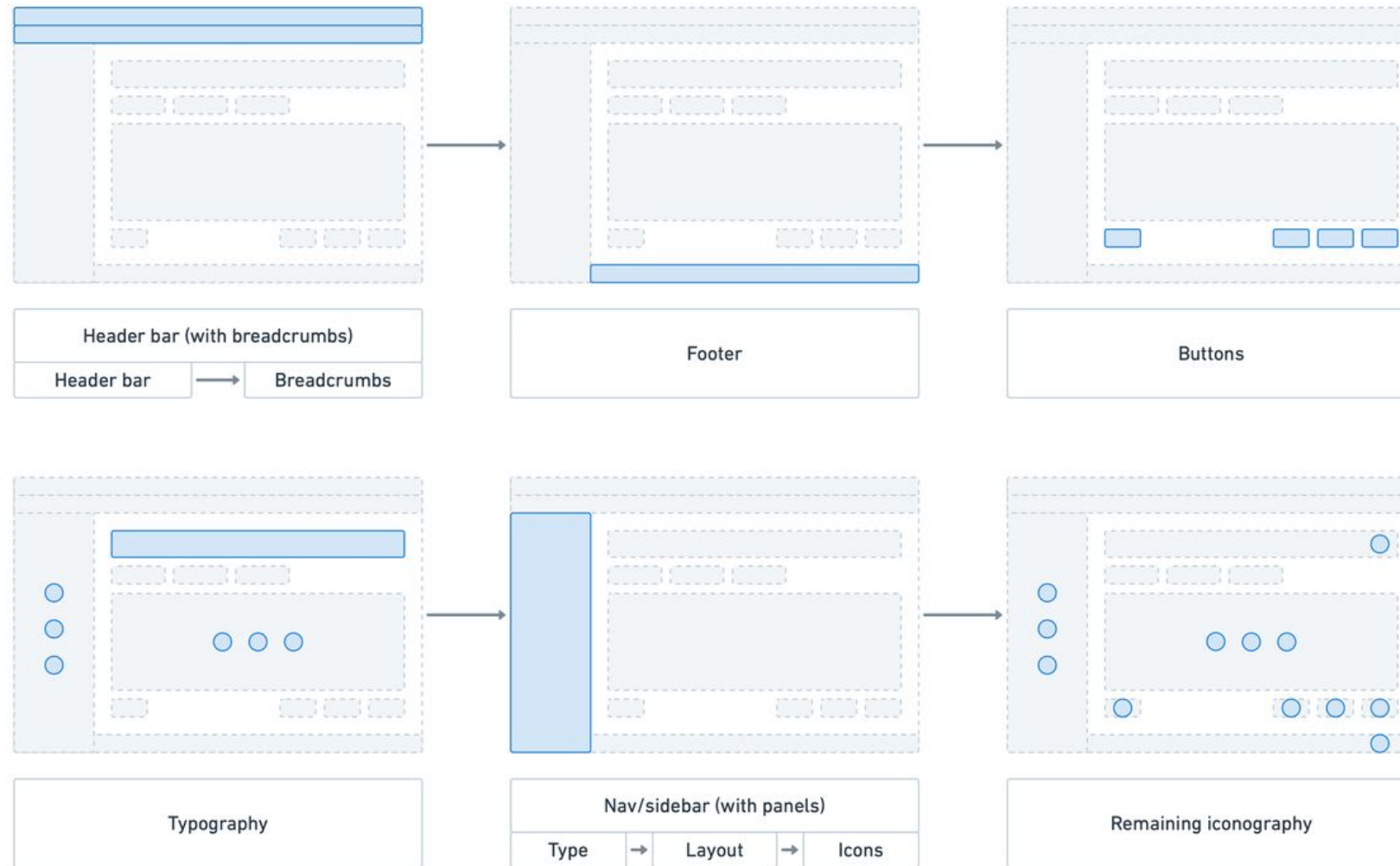
<https://www.jenkins.io/events/online-hackfest/2020-uiux/#user-interface>

Work on New Jenkins UI



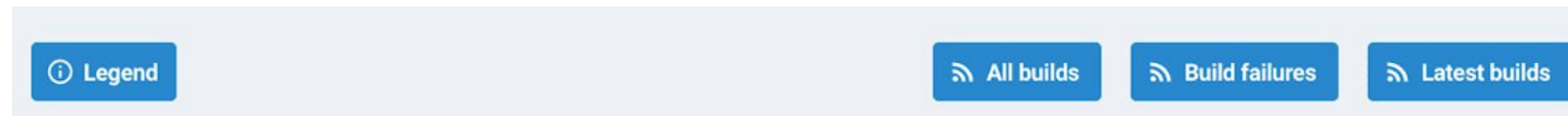
- Work on new Jenkins UI commenced at end of 2019
- Goals
 - Learn from Blue Ocean
 - Be as inclusive of existing plugins as possible
 - Start with overhauling the look and feel of the existing UI

Work During 2020 So Far



Jenkins UX SIG

- Special Interest Group (SIG) formed last year
 - Meet every two weeks on Wednesday 3PM UTC
 - About 10 people join meeting on average
 - Looking for more contributors
 - Find out more: <https://www.jenkins.io/sigs/ux/>



Hackfest project ideas

<https://www.jenkins.io/events/online-hackfest/2020-uiux/#user-interface>

User Documentation

<https://www.jenkins.io/events/online-hackfest/2020-uiux/#user-documentation>

Improve User and Administrator Documentation

- Review & renew user documentation
- Improve navigation, menus, tables of contents
- Improve look and feel (CSS, rendering, etc.)
- Improve experience on mobile

Docs Wiki Migration

- Much useful content still in the wiki to rework & reuse
- Wiki exporter - <https://jenkins-wiki-exporter.jenkins.io/>
- GitHub issues - <https://bit.ly/wiki-migration>


Pipeline CPS method mismatches

Created by Unknown User (jglick), last modified by Unknown User (indifire) on Aug 27, 2019

Introduction

Jenkins Pipeline uses a library called Groovy CPS to run Pipeline scripts. While Pipeline uses the Groovy parser and compiler, unlike a regular Groovy environment it runs most of the program inside a special interpreter. This uses a continuation-passing style (CPS) transform to turn your code into a version that can save its current state to disk (a file called `program.dat` inside your build directory) and continue running even after Jenkins has restarted. (You can get some more technical background on the plugin page and the library page.)

While the CPS transform is usually transparent to users, there are limitations to what Groovy language constructs can be supported, and in some circumstances it can lead to counterintuitive behavior.

 [JENKINS-34344](#) - Running asynchronous code inside a `@NonCPS` method should fail cleanly **RESOLVED**
makes the runtime try to detect the most common mistake: calling CPS-transformed code from non-CPS-transformed code. The following kinds of things are CPS-transformed:

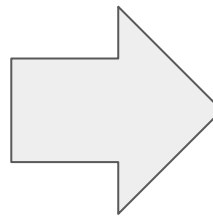
- Almost all of the Pipeline script you write (including in libraries).
- Most Pipeline steps, including all those which take a block.

The following kinds of things are *not* CPS-transformed:

- Compiled Java bytecode, including
 - the Java Platform
 - Jenkins core and plugins
 - the runtime for the Groovy language

Table of Contents

- 1 Introduction
 - 1.1 Table of Contents
- 2 Common problems and solutions
 - 2.1 Use of Pipeline steps from `@NonCPS`
 - 2.2 Calling non-CPS-transformed methods with CPS-transformed arguments
 - 2.3 Constructors
 - 2.4 Overrides of non-CPS-transformed methods
 - 2.5 Closures inside `GString`
- 3 False Positives
 - 3.1 Direct invocation of closures stored in object fields or maps



User Handbook

- User Handbook overview
- Installing Jenkins
- Using Jenkins
- Pipeline
 - Getting started with Pipeline
 - Using a Jenkinsfile
 - Running Pipelines
 - Branches and Pull Requests
 - Using Docker with Pipeline
 - Extending with Shared Libraries
 - Pipeline Development Tools
 - Pipeline Syntax
 - Pipeline Best Practices
 - Scaling Pipelines
- Blue Ocean
- Managing Jenkins
- System Administration
- Scaling Jenkins
- Appendix
- Glossary

Tutorials

- Guided Tour
- Jenkins Pipeline
- Using Build Tools

Resources

- Pipeline Syntax reference
- Pipeline Steps reference

Pipeline

This chapter covers all recommended aspects of Jenkins Pipeline functionality, including how to:

- [get started with Pipeline](#) - covers how to define a Jenkins Pipeline (i.e. your `Pipeline`) through Blue Ocean, through the classic UI or in SCM,
- [create and use a Jenkinsfile](#) - covers use-case scenarios on how to craft and construct your `Jenkinsfile`,
- [work with branches and pull requests](#),
- [use Docker with Pipeline](#) - covers how Jenkins can invoke Docker containers on agents/nodes (from a `Jenkinsfile`) to build your Pipeline projects,
- [extend Pipeline with shared libraries](#),
- [use different development tools](#) to facilitate the creation of your Pipeline, and
- [work with Pipeline syntax](#) - this page is a comprehensive reference of all Declarative Pipeline syntax.

For an overview of content in the Jenkins User Handbook, see [User Handbook overview](#).

What is Jenkins Pipeline?

Jenkins Pipeline (or simply "Pipeline" with a capital "P") is a suite of plugins which supports implementing and integrating *continuous delivery pipelines* into Jenkins.

Chapter Sub-Sections

[Getting started with Pipeline](#)
[Using a Jenkinsfile](#)
[Running Pipelines](#)
[Branches and Pull Requests](#)
[Using Docker with Pipeline](#)
[Extending with Shared Libraries](#)
[Pipeline Development Tools](#)
[Pipeline Syntax](#)
[Pipeline Best Practices](#)
[Scaling Pipelines](#)

Table of Contents

[What is Jenkins Pipeline?](#)
 [Declarative versus Scripted Pipeline syntax](#)
[Why Pipeline?](#)
[Pipeline concepts](#)
 [Pipeline](#)
 [Node](#)
 [Stage](#)
 [Step](#)
[Pipeline syntax overview](#)
 [Declarative Pipeline fundamentals](#)
 [Scripted Pipeline fundamentals](#)
[Pipeline example](#)



Jenkins

Installation Guidelines

- User experience testing and improvement
 - Installation guidelines
 - Tutorials
- Windows installer
 - Use the new Windows installer
<https://www.jenkins.io/blog/2019/02/01/windows-installers/>
- **Advanced:** Docker use jenkins/jenkins image

Improve Plugin Docs

- Wiki conversion - <https://bit.ly/plugin-docs>
- Plugins - <https://jenkins-wiki-exporter.jenkins.io/progress>

ChuckNorris ^{1.2}

Minimum Jenkins requirement: 1.651.3
ID: chucknorris
Installs: 4606
GitHub →
Last released: 2 years ago

Maintainers
Baptiste Mathus

Dependencies
Pipeline: Step API ≥ 1.14
bouncycastle API v.2.16.0 (implied)(what's this?)
Command Agent Launcher v.1.0 (implied)(what's this?)
Oracle Java SE Development Kit Installer v.1.0 (implied)(what's this?)
JAXB v.2.3.0 (implied)(what's this?)
Trilead API v.1.0.4 (implied)(what's this?)

Displays a picture of Chuck Norris (instead of Jenkins the butler) and a random Chuck Norris 'The Programmer' fact on each build page.



Download & Installation

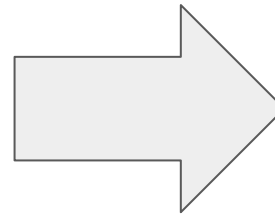
You can preferably install this plugin directly from the Plugins Update Center, or if need be by downloading the latest .hpi and install it from the Manage Plugins menu.

Usage

1. After installing the plugin, go to job configuration page, a checkbox with the text 'Activate ChuckNorris' should appear.
2. Tick the checkbox and save the configuration. For freestyle job, Chuck Norris image (along with a random quote) will appear on the job page.
3. Run a build. After it's completed, Chuck Norris image (and the random quote) should also appear on the build page. This is applicable for both freestyle and Maven jobs.

Screenshots

 **Build #13 (Sep 10, 2015)** Started 18 min ago Took 10 sec
12:02:44 AM
No changes.
Started by user anonymous





jenkinsci / chucknorris-plugin

Watch 123 Star 74 Fork 76

Code Pull requests 11 Actions Projects 0 Security 0 Insights

Branch: master chucknorris-plugin / README.adoc Find file Copy path

 batmat Add the typical README 1a2e7b7 on Nov 1, 2015

1 contributor

10 lines (7 sloc) 553 Bytes Raw Blame History

Chuck Norris Jenkins Plugin

This plugin adds an absolutely delightful feature to Jenkins: depending if your build succeeds, fails, or is unstable, it will show a picture of Chuck Norris **auto-adapting** (that's right, computer science it is!) to the build result!

Seeing that, already many thousands (yes, **thousands!**) of people have decided to take the plunge and install this plugin to finally see the light. Why not you?

For more details, please see the [associated wiki page](#).



User Documentation Track. Links

- All project ideas
 - <https://www.jenkins.io/events/online-hackfest/2020-uiux/#user-documentation>
- jenkins.io
 - Issues: [here](#)
 - Newcomer-friendly Issues: [here](#)
- Plugin issue links
 - All issues: [Jira](#), [GitHub Issues](#)
 - Newcomer-friendly issues: [Jira](#), [GitHub Issues](#)

Spread the Word

<https://www.jenkins.io/events/online-hackfest/2020-uiux/#spread-the-word>

Spread the Word!

- Write user stories for the [Jenkins Is The Way](#) site!
- Write case studies and feature overviews
 - Jenkins blog
 - Other resources
- Post about your Jenkins experience and new features, use the *#jenkinsIsTheWay* hashtag
- Record videos and publish them



<https://www.jenkins.io/events/online-hackfest/2020-uiux/#spread-the-word>



Questions?



Have fun!

#JenkinsIsTheWay