

# 2.0 UX improvements

Please use [epic ticket](#) for comments, feedbacks, etc.  
This is a draft document for [Jenkins 2.0](#). Many of this is real (yet).

A more powerful plugin manager. A revamped UI for configuring jobs, views, slaves, etc. An easier to use "New Item/View/Slave" dialog.

## Overview

Over the past year and a half or so, Tom and I have been cobbling together fragments of Javascript, Java and Jelly code trying to push the Jenkins web ui in the right direction, but routinely bumping into the walls of inflexible page DOM, incompatible legacy JS libraries, plugin compatibility concerns and life draining skepticism from the community. With the release of a new version of Jenkins, Tom and now Keith and I are eager to cast off some of this baggage so we can start bringing the Jenkins web user experience.

## Subject areas

The Jenkins GUI has a potentially broad reach and many areas for possible and substantial improvement. To help clarify the effort, the following 15 subprojects are a helpful means of dividing possible areas of work:

[https://www.youtube.com/embed/yra\\_HzRzt1k](https://www.youtube.com/embed/yra_HzRzt1k) (6min and again 11min as Tom demos themes which are an upside enabled by semantic pages and required for config pages demoed at 24min)

1. Initial plugin selection and installation  
<https://youtu.be/kzRR8XR8hu4>
2. Flexible LESS driven theming (Tom's forgotten effort)  
This is not as well articulated but touched on in this JUC talk:  
[https://www.youtube.com/embed/yra\\_HzRzt1k](https://www.youtube.com/embed/yra_HzRzt1k) (11 min in)
3. New item categorization and dynamic choice offering  
[https://www.youtube.com/watch?v=A\\_lu-P0OhNQ](https://www.youtube.com/watch?v=A_lu-P0OhNQ)
4. Configuration page overhaul  
[https://www.youtube.com/watch?v=A\\_lu-P0OhNQ](https://www.youtube.com/watch?v=A_lu-P0OhNQ)
5. Plugin Manager and Update Center overhaul  
<https://www.youtube.com/watch?v=9vPUMe3lzfo>  
<https://www.youtube.com/watch?v=kcXKvyX4Oq8>  
<https://www.youtube.com/watch?v=jf5s0sqIYBY>
6. Fully responsive app page template wrapper
7. Workflow results monitoring
8. Improved console out (easier access and better readability)
9. Multi and inline action grid control (with sorting and grouping)
10. Global context build queue tracking
11. Improved post-run build analytics ([video](#))
12. [Terminology cleanup](#)

## References

- [Unknown User \(gusreiber\)'s blog post](#)
- [Jenkins blog post with references to previous discussions and presentations](#)

## Related Issues

- [JENKINS-14538](#) Separate "configure tools" page
  - [Unknown User \(gusreiber\)](#) So far, I have not looked more deeply into the notion of splitting the control of plugin added configuration from stock job configuration (as suggested here, if I am understanding correctly). Even outside of #14538, the idea of somehow segmenting plugin added controls from stock controls has come up in JUC and makes a lot of sense. What I am suggesting for configuration is collapsible regions. As shown here:
    - [https://www.youtube.com/watch?v=qKLi\\_ViPxy8](https://www.youtube.com/watch?v=qKLi_ViPxy8)
    - [https://www.youtube.com/watch?v=A\\_lu-P0OhNQ](https://www.youtube.com/watch?v=A_lu-P0OhNQ)
- [JENKINS-17011](#) More convenient job selection control
  - [Unknown User \(gusreiber\)](#) As part of a CloudBees feature development, we have constructed a new file browsing control that does a sort of type-complete of your folder structure, allowing you to type or pick your path smoothly. This control should be generalized and open-source.
- [JENKINS-17373](#) Unconditionally display tool installation selector
  - [Unknown User \(gusreiber\)](#) This seems Maven job specific, or am I misunderstanding?
- [JENKINS-18435](#) Navigating back, then forward, from a job configuration causes loss of config data
- [JENKINS-19828](#) Kill or rework auto refresh
  - [Unknown User \(gusreiber\)](#) yes, this is awful. I would like to see real time sensitive data separated a bit from configuration. Then real time data could be handled in a more modern way and most importantly not bog down the sort of regular page navigation that is required for configuration (which is also less than ideal).
- [JENKINS-19584](#) Ability to redirect user to configuration screen when erroneous data is saved
  - [Unknown User \(gusreiber\)](#) This issues seems specific to a particular error case that is not adequately handled. The behavior described by the issue reported is the behavior we want in all possible cases.
- [JENKINS-21092](#) Adaptable UI for Mobile Devices

- [Unknown User \(gusreiber\)](#) I believe this needs to happen. For it to happen, many pages will need to have moderate DOM structure changes that will require a refactoring of hudson-behavior.js and will introduce some plugin instability.
- [JENKINS-20190](#) Ability to sort or group /lib/hudson/newFromList:form
- [JENKINS-21486](#) Fail a plugin if its dependencies doesn't exist
- [JENKINS-22926](#) Extension point for augmenting Plugin Manager display
  - [Unknown User \(gusreiber\)](#) I think plugin handling needs to be completely overhauled:
  - <https://www.youtube.com/watch?v=9vPUMe3lzfo>
  - <https://www.youtube.com/watch?v=kcxKvyX4Oq8>
  - <https://www.youtube.com/watch?v=jf5s0sqIYBY>