Hello Mr. Jenkins

Michał Mróz - mmroz122@gmail.com
What is Jenkins?

- Open source test automation server written in Java in 2004 as Hudson at Sun Microsystems.
- Became a fork of Hudson and changed name in 2011 to Jenkins.
- Builds and tests software and monitors status of builds.
- Over 1000+ plugins.
- Integrates with many tools: version control, reporting, testing tools, bug tracking.
Continuous Integration

- CI is a development practice that requires developers to integrate code into a shared repository several times a day. Each check-in is then verified by an automated build, allowing teams to detect problems early.
- By integrating regularly, you can detect errors quickly, and locate them more easily.
Installation

- Available for FreeBSD, OpenBSD, Ubuntu, Debian, Windows and more...
- To install (FreeBSD) `pkg install jenkins`.

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

```
/usr/local/jenkins/secrets/initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password
Installation

Welcome to Jenkins!

Please create new jobs to get started.
Manage Jenkins

- Configuring System – plugin settings, notifications, workspaces, environment variables.
- Users, security settings.
- System logs.
- Nodes.
- Credentials.
- Plugins.
Plugins

- Support for many version control systems: Perforce, Subversion, Git, Mercurial, CVS…
- Bug tracking – JIRA, Redmine…
- Raporting – XML, HTML…
- Notifications – Slack, Gmail, Discord…
- Dashboard plugins to edit views.
Managing plugins

- All plugins are available from Manage Plugins view where you can update and install them.

<table>
<thead>
<tr>
<th>Plugin</th>
<th>Description</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCM</td>
<td>This plugin collects the CCM analysis results of the project modules and visualizes the found warnings.</td>
<td>3.2</td>
</tr>
<tr>
<td>NuGet</td>
<td>NuGet build support plugin.</td>
<td>1.1</td>
</tr>
<tr>
<td>MSBuild</td>
<td>This plugin makes it possible to build a Visual Studio project (.prj) and solution files (.sln).</td>
<td>1.28</td>
</tr>
<tr>
<td>MSTester</td>
<td>This plugin converts MSTester TPX test reports into JUnit/HTML reports so it can be integrated with Hudson’s JUnit features.</td>
<td>0.20</td>
</tr>
<tr>
<td>MSTestRunner</td>
<td>This plugin runs MSTester command line tool to execute unit tests for .NET projects.</td>
<td>1.3.0</td>
</tr>
<tr>
<td>Net</td>
<td>This plugin is a .NET build for building .NET projects.</td>
<td>1.4.3</td>
</tr>
<tr>
<td>PowerShell</td>
<td>This plugin allows Jenkins to invoke Windows PowerShell as build scripts.</td>
<td>1.3</td>
</tr>
</tbody>
</table>
Creating jobs

- Jobs – executing shell locally/remote.
- Pipelines – multiple step build.
- GitHub Organization – build on repository.

Enter an item name

* This field cannot be empty, please enter a valid name

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCMs with any build systems, and this can be even used for something other than software build.

Pipeline
Orchestrates long running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike views, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

GitHub Organization
Scans a GitHub organization (or user account) for all repositories matching some defined markers.

Multibranch Pipeline
Creates a set of Pipeline projects according to defined branches in one SCM repository.

If you want to create a new item from other existing, you can use this option:

Copy from  Type to autocomplete
Pipeline

- Write in Groovy.
- Jenkinsfile.
- Declare stages and steps.
- Parallel steps.
- Running other jobs...

```groovy
pipeline {
    agent any

    stages {
        stage('Build') {
            steps {
                echo 'Building..'
            }
        }
        stage('Test') {
            steps {
                echo 'Testing..
            }
        }
        stage('Deploy') {
            steps {
                echo 'Deploying....'
            }
        }
    }
}
```
Results Blue Ocean

<table>
<thead>
<tr>
<th>Test</th>
<th>Description</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>test_check_routing_not_conf_c765</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_server_add_page_c762</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_secure_add_page_c772</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_account_add_page_c773</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_check_dashboard_c777</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_check_sessions_c778</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_check_users_c779</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_check_servers_c780</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_check_accounts_c781</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_check_listeners_c782</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_check_safes_c783</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_password_policy_c794</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
<tr>
<td>test_password_password_change_c795</td>
<td>- test_check_all_pages</td>
<td>&lt;5s</td>
</tr>
</tbody>
</table>
Questions ?