**CI CD Flow**

**Tomer Baum**

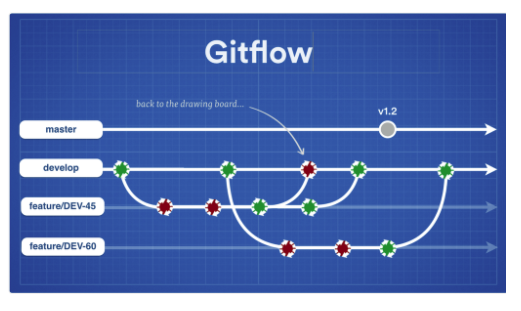
1. Commit to code like C#, Java, Python, PHP, HTML … to feature branch.  
     
   In version control systems, a commit adds the latest changes to [part of] the source code to the repository, making these changes part of the head revision of the repository. Unlike commits in data management, commits in version control systems are kept in the repository indefinitely. Thus, when other users do an update or a checkout from the repository, they will receive the latest committed version.

1. Jenkins Build  
    Maven command plus JUnit tests during the build

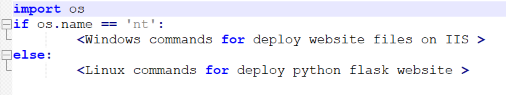
Usually automation QA team or developers write the unit test code and update the pom.xml, so the Maven will know what tests to run   
  
Jenkins comes with a test harness built around JUnit to make test development simpler. This harness provides the following features:

* Starts an embedded servlet container so that the test code can exercise user interaction through HTTP and assert based on the outcome.
* [HtmlUnit](http://htmlunit.sourceforge.net/) with a bit of enhancements allows you to productively test HTTP/UI interaction.
* Prepares and tears down a fresh Jenkins instance for each test case. So each test method will run in a fresh environment, isolated from other tests.
* Test code can also directly access Jenkins object model. This allows tests to assert directly against the internal state of Jenkins, as well as perform some operations directly, as opposed to doing so only through the HTML scraping.
* Declarative annotations to specify the environment in which a test will be run. For example, your test method can request that Jenkins shall be started with a certain HUDSON\_HOME contents.
* Declarative annotations to maintain association between tests and bugs/discussions.

1. If build succeeded there is **merge to master**

  
Instead of a single master branch, this approach uses two branches to track the history of the project. While the master branch contains tags and/or commits that record the project's official release history, a shared integration branch (usually called "develop") gives your team a place to ferret out bugs and incompatible changes.

1. Deploy: with Post receive script - after the push command, the python script or bash script run in the GIT, and it will check first if its Windows or Linux, and only then act accordingly with the web deploy commands.

  
Like many other Version Control Systems, Git has a way to fire off custom scripts when certain important actions occur. There are two groups of these hooks: client-side and server-side. Client-side hooks are triggered by operations such as committing and merging, while server-side hooks run on network operations such as receiving pushed commits. You can use these hooks for all sorts of reasons

Deploy can be done to QA environments and Developers environments, and when release is ready after QA tests and automations team scripts like HP , it will deploy to code Freeze environment and then to Client Production as new Release.