# CI & TDD

## Double Rainbow

# Overview

1. Developers use the Dev branch to make changes, enhancements, and updates.

2. Developer commits got to GitHub Dev branch which is tested via Jenkins

3. Jenkins compiles, deploys, and test the Dev branch

4. The Dev branch build process succeeding triggers the Test branch to update

5. The Test branch is merged with the Dev branch and triggers a build process

6. The Test branch is built and tested via Jenkins

7. On failure the CI pipeline sends emails out to prompt action

8. Upon completion of sprint and testing succeeding the code is promoted to the Stable branch.

# Test Drive Development

1. Requirements become User Story

2. User Story becomes Test Plan task (test team w/ Cucumber )

2a. refine requirement as needed (test/dev/req team)

3. Test Plan becomes unit test and then function/feature (developers w/ Jasmine)

3a. refine requirement as needed (dev/test/req team)

4. Continuous Integration (test and dev team)

4.a Unit Testing succeeds -->

4.b Functional/Usability Testing succeeds -->

4.c Build succeeds and is promoted to Stable

# Development

## Dashboard

On the server running via IIS

Dev - http://irdsdev.vaftl.us:7001 - DevDB

Test - http://irdsdev.vaftl.us:7003 - TestDB

Stable - http://irdsdev.vaftl.us:7005 - StableDB

## Database

Reach\_Dev – Development database built ad-hoc to produce scripts and guides for future use.

Reach\_Test – Test database built using scripts and deployment guides.

Reach – Master database with Stable code.

The Master database will be backed up each sprint. Major changes or enhancements will be stored in scripts for future release guides.

## Testing

Cucumber:

/project/src/

/project/features/steps/

/project/features/LoginFunction.feature (example)

/project/features/steps/login\_function\_steps.rb

# Jenkins

PR\_DEV

1. Dev branch of PR

2. Triggers PR\_DEV\_TEST

PR\_DEV\_TEST

1. Dev Unit Tests

2. Triggers PR\_TEST

PR\_TEST

1. Test branch of PR which is merged with Dev branch upon successful build and test

2. Triggers PR\_TEST\_TEST

PR\_Test\_TEST

1. Test Functional Tests (Cucumber)

2. Results represent coverage of functional requirements

PR\_Stable

1. Merge of Test after succeeding builds and test

2. Completion of Sprint and Demo environment

### Notes:

A “Hot Fix” will be developed against the Stable branch. Only one branch will be created per “Hot Fix” branch. After successful testing the “Hot Fix” branch will be merged with the Stable, Test, and Dev branches to close out the process.

A “Feature Branch” will be built against the Dev Branch. The “Feature Branch” will be created for one feature at a time. The “Feature Branch” will be tested an merged after completion.

# Build Instructions

Install: Node.js (v) + IISNode (v) + IIS redirect addon

Compile code: npm install –g grunt-cli, npm install –g bower, npm install, bower install

Configure: IIS

Create site – point site to the directory where the compiled code.

A web.config file must be present with a redirect to the correct file.js.

Database: SQL Server 2012

CI: Jenkins (v)

Configure: install add-ons for Git, NodeJS

Testing: Cucumber (v)

Configure: gem install cucumber, gem install selenium-webdriver, gem install gherkin, gem install capybara, gem install rspec