All:

There seems to be some confusion on how things work w/ to test items in the development branch, and so I'm writing this email to 1) clarify, and 2) have something to point at if this happens again.

So, here we go:

Process for Testing Pre-Release Candidate Items

1. Developer (dev) writes code and has it peer reviewed
2. Code passes peer review & dev merges code into the '**develop**' branch
3. When code is merged into the '**develop**' branch Jenkins automatically makes a new build in the '**ci-dev-companyx**' area
    1. (Slack channel **#deploy-notify-ia**)
4. When Jenkins is done w/ the '**ci-dev-companyx**' build, it outputs a message on Slack
    1. *Ex: Build v_721 is now live at dev-www.companyx.com*
5. QC then takes this "**v-number**" (i.e. v_721) and deploys that to QAT environment in order to test everything that has been merged into development up to that point
6. If you want to know what was in a '**ci-dev-companyx**' build you can review what was included in a build in Jenkins
    1. I've already taken the time to document this here: https://companyx.url.com
7. You can also tell if a story's code was merged into '**develop**' by looking at the story in JIRA. In the '**Development**' section you'll see text similar to this: **1 pull request MERGED Updated X days ago**. When you see this, the story's code has been merged into '**development**'

But Wait, What are the Exceptions?

There are two exceptions:

1. If we need to test a feature that is NOT merged into the '**develop**' branch, then you sometimes will deploy a '**feature branch**' to QAT
    1. I've already taken the time to document this here: https://companyx.url.com
    2. The FA project is a great example of this...
2. At the end of every sprint we create a '**release candidate build**' which QC places on the UAT environment (or wherever they want frankly) for final regression testing
    1. I've already taken the time to document this here: https://companyx.url.com

OK, so this should hopefully clear up any confusion anyone has, and I'll be happy to resend this email to anyone as they require. :)

Thanks!
Nathan