Reviewing pull requests

An important way you can help move Titanium forward is to review the changes that others have submitted. Proper functional testing helps us merge changes more quickly and with more confidence.

This document describes the process you can follow to test any of the "NPM-installable" projects, including Alloy and the CLI. While you could test PRs for the Titanium SDK itself, that would also involve Building the Titanium SDK From Source which is covered elsewhere in the documentation.

One-time setup:

1. Clone the project repo to a local directory. For example, for Alloy you would use `git clone https://github.com/appcelerator/alloy.git` (you can't do this with your fork of the project)
2. In the resulting directory, open the **.git/config** file in your text editor.
3. Follow the instructions at https://help.github.com/articles/checking-out-pull-requests-locally to modify your git configuration to support fetching pull requests

Checkout a PR

GitHub for Mac (or Windows) users, you'll see a button to check out the PR directly from the GitHub repo page. This will work only if you have write access to the repo. (You can check out https://github.com/blog/1582-checking-out-pull-requests for more details.) Sorry, you're going to have to use the command-line for this:

1. Update your local repo with `git fetch origin` (which will also fetch all the PRs).
2. Check out the specific PR using `git checkout pull/origin/999 -b 999` where 999 is the PR number (the -b 999 creates a new local branch named after the PR)
3. Install from your local branch: `[sudo] npm install -g .` (with the dot at the end). You're ready now to do your testing, etc.

Functional Review testing

Tickets should contain a description of what steps should be followed to review the changes. Alloy tickets, for example, should note a sample app (either attached to the PR or to the ticket). Run that app and confirm it displays the expected behavior.

Optionally, you could build one of your own projects with the PR version. This could uncover issues that might be missed by the simple test case app included on the ticket.

When you're done, add a comment on the ticket (preferred) or pull request. If all works fine, a simple "FR passes" comment will suffice. If there are errors or if the feature doesn't work as expected, please post detailed comments including any error messages

Post FR cleanup

1. Install the release version again. E.g. `[sudo] npm install -g alloy`
2. Switch to your master branch: `git checkout master`
3. Finally, force-delete your local PR branch: `git branch -D 999` (this will destroy any changes you made to the files)

Alloy's jake app runner

Alloy includes the jake (JavaScript make) utility, which you can use this to run any of the PR test apps, as well as the sample/demo apps included in the repo. For example, from your local Alloy repo directory:

```
jake app:run dir=ALOY-1234 // to run the app in test/apps/testing/ALOY-1234
jake app:run dir=basics/simple // to run the test/apps/basics/simple demo app

// additional command-line options are:
platform=android   // to build for android,ios,mobileweb,blackberry
tiversion=3.2.2.GA  // to build using a specific SDK version (must match tiapp.xml)(
tisdk=<path_to_sdk>

// e.g.
jake app:run dir=ALOY-1234 platform=android tiversion=3.2.2.GA
```

The resulting app will be installed to your simulator/emulator under the name **Harness**. You can import the Harness app into Studio to build for other platforms, install to device, etc. The Harness app is wiped with each jake app:run command.