**Façade API development environment set up**

**Below are needed, follow links for installation guide.**

1. Cygwin [https://www.cygwin.com](https://www.cygwin.com/) (command line tool), check python to include it as part of installation.
2. Git <http://git-scm.com/> source control, <https://www.codeschool.com/courses/try-git> and tons of information on google.
3. Maven <https://maven.apache.org/download.cgi>
   1. Extract Maven to c:/maven
   2. Open Cygwin, add environments variable below if you don’t already have it to the file .bash\_profile
   3. Make sure you are in the home folder. Open .bash\_profile and add

export PATH=$PATH:/cygdrive/c/maven/bin

export JAVA\_HOME="/cygdrive/c/Program Files (x86)/Java/jre7“

* 1. Restart Cygwin and check Maven is there by typing in mvn --version

1. Nodejs <https://nodejs.org/>, we use Nodejs to test Façade API
2. Install mocha, chai, supertest. npm install -g mocha, npm install –g chai, npm install –g supertest
3. Sublime <http://www.sublimetext.com/>, for editing javascript etc. (optional if you have other prefer editor)
4. Install Arcanist, for submitting code review into Phabricator <http://tinyurl.com/lugay8w>

**Check out/in from/to Git**

1. Start Cygwin, in your home folder create a new folder /façade-api
2. In façade-api, pull down the latest “git clone http://stash1.rebellion.t-mobile.com/scm/reb/api.git
3. Switch to the branch you want to work on. e.g. master “git checkout master or sprint branch “git checkout sprint-43”
4. Cd into the proxy you want to work on, all proxies are under the folder api/apigee e.g api/apigee/web-customer. To make sure you have an independent development space, deploy the proxy by using your TMO ID as the prefix for both proxy name and base path. (use “-“ instead of “.”) e.g. louis-lim5. Do not check in the prefix part, roll back to what it was before.
   1. To change the proxy name and base path.
      1. Prefix “name” xml element in pom.xml e.g.<name>louis-lim5-web-customer</name>
      2. Prefix “BasePath” element in apiproxy/proxies/default.xml e.g. <BasePath>/louis-lim5-web-customer/v1</BasePath>
   2. Use command below to deploy to “rebelliondev” organization, “dev01” environment.
      1. mvn apigee-enterprise:deploy -Pdev01 -Dusername=louis.lim5@t-mobile.com -Dpassword=xxxx
5. When development completed, don’t forget to get the changes into your Git folder so you can check them in. If this is a big changes, you should deploy and run the test again to make sure your changes work and not breaking anything existing.

**When checked in to GIT, start the commit message with user story number.**

**e.g. git commit –m “us26552-creating api-101-getCustomer”**

**test should be checked in as well, Jenkins will run the same set of test after it deploys.**

**Testing Façade API**

Every proxy will have a corresponding test folder. The naming convention is proxy-name/test e.g. web-customer/**test**

1. If you are adding a new API, add a test for that API. If you are updating or fixing a defect, update the existing test.

E.g http://stash1.rebellion.t-mobile.com/projects/REB/repos/api/browse/apigee/web-customer/test

1. Run all test to make sure all test case for this proxy pass.

e.g. in /web-customer/test

run “env=test mocha default.js”

env=test will set the var env which is used to switch the domain we want to test.

1. Submit your changes for code review <http://tinyurl.com/lugay8w>
2. Once review is approved, check in your changes.