

Building an InteractivatePlus Dev Environment

Matt DesVoigne

December 18, 2008

The following steps create a development environment on a local user account for InteractivatePlus development. Temporarily, InteractivatePlus will be kept in a separate branch in SubVersion. When the product becomes more stable, the InteractivatePlus branch will replace the Interactivate branch in SubVersion. These directions assume command line usage. Jar files may alternatively be made through Eclipse.

- Step 1: Check out build scripts, php, database, and Java code from SubVersion. Place the contents of the checkout in an interactivateplus folder inside the public_html directory. (You may name your checkout directory differently, but if you do, you need to modify the INTERACTIVATEROOT variable in the buildplus.sh script.
 - `cd ~/public_html`
 - `svn checkout https://svn.shodor.org/repos/prod/interactivateplus/trunk interactivateplus`
- Step 2: Give yourself permission to execute the buildplus.sh script. The usage of buildplus.sh is the same as build.sh for “Non-plus” Interactivate.
 - `cd interactivateplus/interactivate`
 - `chmod +x buildplus.sh`
- Step 3: Build the applet you want run under InteractivatePlus. (At the time of this writing, the database, php, and Java code are only set up to run fgame. Much of the remaining work will involve implementing the other activities.
 - `./buildplus.sh fgame`
- Step 4: Attain an InteractivatePlus username and password. Pages to create new accounts, retrieve forgotten passwords with security questions, retrieve temporary passwords for forgotten passwords, etc, still need to be created. For now, see Matt DesVoigne for an account. To start out, I will set up accounts to be username: <your_last_name>, password:<your_last_name>pass. For example, my username is desvoigne and my password is desvoignepass. For effective testing and development, please only use your account unless you have a good reason to use somebody else’s for testing and you have the other party’s consent. After the pages to create accounts/reset passwords are created, you can create your own account with a username and password of your choice.
- Step 5: Access InteractivatePlus through your individual web space:
 - https://newdev.shodor.org/~<your_shodor_username>/interactivateplus/login.html
 - For example, my entry page is
 - <https://newdev.shodor.org/~mdesvoig/interactivateplus/login.html>
- Step 6: After you login with your InteractivatePlus username and password, you can load an activity or view your progress. Since fgame is currently the only activity that records student responses in the database on the server and after your first login you have no progress data, it is natural to start by attempting some questions in fgame. After you have completed some questions, return to the login page (a back link still needs to be implemented so you may need to re-login) and view your progress on the fgame questions you answered. If you see your progress, your development environment is set up.

Continued on next page.

InteractivatePlus File Structure

The InteractivatePlus file structure currently stands as follows:

- interactivateplus directory
 - Files
 - login/php page to login
 - php page to record student attempts in the database
 - php page to view student progress data
 - php include file to establish web user connection
 - checkoutCommand.txt: Contains the checkout command (also listed above) to check out the interactivatePlus project from SubVersion.
 - Directories
 - interactivate – a copy of the “non-plus” interactivate Java code (<https://svn.shodor.org/repos/prod/interactivate/trunk>) along with build scripts to construct jar files for activities.
 - Note: each “activity” directory in the interactivate folder will potentially contain a loadActivity.php5 page responsible for passing student id and activity id to the applet to be used when the student attempts a question and attempt data is to be sent back to the server. (Currently there only a loadActivity.php5 in fgame.) It remains to be determined whether we will keep the loadActivity pages or the applets will retrieve student id and activity id info from the greater Ineractivate Plus learning environment.
 - sql – Scripts to connect to the db_interactivateplus database as either an administrator (adminConnect.sh) or a web user (userConnect.sh). Also does/will contain sql stored procedures, example database scripts for reference, etc.