# Developer Workstation Onboarding

## Tools Used by UFG

* GitHub as Code Repository – Git Command Line, GitCracken, Tortoise Git
* Delphix for Local DB instances (hosted on cloud)
* Jira for Sprint, Tasks, Time tracking, etc.
* Confluence for all documentation, reports, requirements, etc.

## Developer Environment Setup

* Setup your GIT Account (you can choose PwC Mail address as your login) and send a mail to [oasisgithubadministrators@unitedfiregroup.com](mailto:oasisgithubadministrators@unitedfiregroup.com) to get access to UFG Repositories (PC, BC & CM). For more details, please refer to link - <https://ufginsurance.atlassian.net/wiki/spaces/OP/pages/21135519/Setting+Up+Your+GIT+Account>
* Create Personal Access Token and Add GitHub token & username to windows credentials manager. (Refer to Step 3 & 7 from the link for steps and details- <http://wiki.cr.unitedfiregroup.com/display/WDT/Create+a+GitHub+Account> ). This will ensure that you don’t enter your git hub username/password for each push/pull requests from repo.
* You will receive a mail when your account is given access to UFG Repo, after which navigate to C:\OASISPrj\UtilsDocuments in your VM and run the windows power shell script – OasisGitRepoSetup. You will notice before running this script that under C:\Oasis\Workspace, we have the OOTB Base installed for PC, BC & CM with an empty configuration folder. The above script will prompt for your GIT Username/password and clone (bring in the latest changes) from the GIt repository (develop branch) under configuration folder. Verify the same after the script is run
* You may choose to use multiple options to interact with GIT. GIT Command line, GITKracken, Git Tortoise, etc.
* You will also receive a link to login to Delphix (Tool for connecting our local PC, BC, CM instances to SQL Server instances). Link - <http://172.16.0.223/login/index.html> . Once you have successfully logged onto Delphix, you should see your local DBs under Container tab. You will need to manually go to your individual GW Centers and update database-config.xml to use your own DB instance.

PC : <dbcp-connection-pool jdbc-url="jdbc:sqlserver://CR12GWSQLDEV01.CR.UnitedFireGroup.com\MSSQLSERVER:1433;DatabaseName=<YourPCDBName>;User=pcUser;password=pc"/>

BC: <dbcp-connection-pool jdbc-url="jdbc:sqlserver://CR12GWSQLDEV01.CR.UnitedFireGroup.com\MSSQLSERVER:1433;DatabaseName=<YourBCDBName>;User=bcUser;Password=bc"/>

CM: <dbcp-connection-pool jdbc-url="jdbc:sqlserver://CR12GWSQLDEV01.CR.UnitedFireGroup.com\MSSQLSERVER:1433;DatabaseName=<YourCMDBName>;User=abUser;Password=ab"/>

* Delphix gives a lot of functionalities to refresh/take snapshots of data in time. As per client’s expectations, you may receive a training on delphix before you get access. Link to Delphix documentation - <https://ufginsurance.atlassian.net/wiki/spaces/OP/pages/21168207/Using+Delphix>
* The DB provided by Delphix automatically comes with loaded Admin data and Rate books loaded for SBT (CA as of now). Once your local PC, BC, CM is connected, you don’t need to load any other data to get started.
* Install DCEVM -> Refer to documentation under the link : <https://ufginsurance.atlassian.net/wiki/spaces/OP/pages/21332074/Guidewire+Setup> and install DCEVM. Skip other items on this link, as your VM should already have all of other items configured

## Using GIT

Refer to the link for a detailed documentation - <https://ufginsurance.atlassian.net/wiki/spaces/OP/pages/21332070/Using+GIT>

Key Points to note:

* Create a new branch in local for every story or defect from the develop branch
* Commit one or multiple changes (as needed) for that story to the local new branch.
* Once ready to push the changes (Commit is a three step process here), using Tortoise/GitKracken/Git Command line) , push the changes to a new remote branch (you can use the same name as your local branch name)
* Create a pull request on GIT HUB UI (This is called PULL in reference to the develop branch). Add Reviewers – (Neeraj Gautam, Mike Jones, David Shireman) and description of the changes
* Once approved, either you or even reviewer can complete the pull requests and merge the changes to the develop branch and delete the remote story branch (which was created by you while pushing the changes to remote repository)

## Coding Standards

Please refer to the link to download the coding standards and follow the same during development.

<https://ufginsurance.atlassian.net/wiki/spaces/OP/pages/21135515/Coding+Standards>

## Link to UFG Environments (Dev/QA)

<https://ufginsurance.atlassian.net/wiki/spaces/OP/pages/39879036/OASIS+Servers>

## JIRA Link -

## <https://ufginsurance.atlassian.net/secure/Dashboard.jspa>

## Confluence Link -

## <https://ufginsurance.atlassian.net>