**Content:**

* **Focus: In this course, we will focus on GIT COMMANs and prove WHY almost professional Developers use GIT COMMANDs**
* **Agenda: (Duration: 3 hours)**
  + **What is Branch?**
  + **Local and Remote repositories/branches (distributed version control)**
  + **Commands (work together in an example project in GIT)**
    - **Init**
    - **Config**
    - **Status**
    - **Add**
    - **Stash**
    - **Reset**
    - **Commit**
    - **Push**
    - **Checkout**
    - **Pull**
    - **Branch**
    - **Fetch**
    - **Merge (+tools)**
    - **Remote**
  + **Recap**
* **Note: In this course we will work together in a example project with real cases.**

Example outline:

1. Basic concepts of distributed version control
   1. Local repository, remote repositories
   2. Every dev has own local repository
   3. All local repositories can be configured to be each other's remotes
   4. Committing on local (add, commit), syncing your commits with others (fetch, merge, pull, push)
2. Branches in Git
   1. Branching is cheap
   2. Branch is just a pointer to a commit
   3. Switching between branches is easy
   4. Nothing special with master branch --> "master" is just default name for first branch
   5. The team can choose own workflow (lots of short-lived temp branches, or a few long-lived branches)
   6. Commands (branch, checkout)
3. Starting a Git repository
   1. New local repo (git init)
   2. Clone an existing remote repo (git clone)
   3. Version control (contents of .git directory, such as objects, refs, logs)
   4. Keeping repo clean (.gitignore, .gitattributes)
   5. Setup own config (git config, git config --global user.name "Nguyen Van A", credentials)
4. Working on local
   1. Edit files in working directory (use any editor, IDE…)
   2. See status (git status)
   3. Staging files (git add, git add <filename>)
   4. Unstaging files (git reset, git reset HEAD <filename>)
   5. Committing (git commit, git commit -m "Added whitespace")
   6. Branching (git branch <newbranchname>), current branch (HEAD points to current branch)
   7. Switching branches (git checkout <otherbranch>)
   8. Merging branches (git merge) <-- simple cases
5. Working with remotes
   1. See all remotes (git remote, git remote -v)
   2. Cloned repos have remote "origin" --> nothing special with origin, just a default name
   3. Add remotes (git remote add <name> <url>)
   4. Show more info on remote (git remote show origin)
   5. Get all changes from a remote (git fetch, git fetch <remote>)
   6. Tracking; local branch tracks a remote branch (git checkout --track <remotebranch>)
   7. Merge changes from a remote branch (git merge, git pull)
   8. Sharing your code by pushing to remote (git push)
6. Merging, merge conflict, diff, history
   1. Fast-forward merges and automatically resolved merges (git merge)
   2. Merge conflict (git mergetool, also use GUI)
   3. See history (git log)
   4. Compare diffs (git diff HEAD <filename>, git difftool, also use GUI)
   5. Storing changes without commit (git stash)
   6. Rebasing (git rebase); merge vs. rebase

**Trainees : 36 trainees**

Number of Groups: 4 groups

**Day 1 & Day 2: April 23 & April 24, 2015**

|  |  |
| --- | --- |
| **Group 1: 8.30am-10am** | **Group 2: 10.30am-12am** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **ID** | **Name** | **Absent** | **Note** | **Team** | |  | Lý Chí Cường |  |  | HR Tool | |  | Phan Công Úy |  |  | HR Tool | |  | Nguyễn Thị Trinh |  |  | HR Tool | |  | Trần Quang Vinh |  |  | Conexus 1 | |  | Lê Quang Nguyên |  |  | Conexus 1 | |  | Nguyễn Anh Tuấn Hiển |  |  | Crooze | |  | Phạm Phi Phúc |  |  | Conexus 0 | |  | Nguyễn Trung Trực |  |  | Isolution | |  | Trầm Minh Trí |  |  | Isolution | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **ID** | **Name** | **Absent** | **Note** | **Team** | |  | Nguyễn Ngọc Tư |  |  | Conexus 4 | |  | Đinh Duy Linh |  |  | Conexus 4 | |  | Nguyễn Phương Quyên |  |  | Conexus 4 | |  | Trần Văn Quang |  |  | Conexus 4 | |  | Phạm Thái Hoàng |  |  | Conexus 4 | |  | Đỗ Trần Nhựt Khánh |  |  | Conexus 4 | |  | Bùi Hoàng Hải |  |  | Penentrace | |  | Ngô Đức Quốc |  |  | Isolution | |  | Trần Huy Thanh |  |  | Isolution | |

|  |  |
| --- | --- |
| **Group 3: 1pm-2.30pm** | **Group 4 : 3pm-4.30pm** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **ID** | **Name** | **Absent** | **Note** | **Team** | |  | Nguyễn Phạm Minh Quân |  |  | Kikora | |  | Cao Quân Hào |  |  | Kikora | |  | Đinh Bá Nhựt |  |  | Kikora | |  | Nguyễn Nhất Nguyên |  |  | Kikora | |  | Huỳnh Sơn Lâm |  |  | Kikora | |  | Nguyễn Minh Nhật |  |  | Conexus 2 | |  | Trần Lực |  |  | Conexus 2 | |  | Nguyễn Văn Nam |  |  | Conexus 2 | |  | Nguyễn Công Nhịn |  |  | Conexus 2 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **ID** | **Name** | **Absent** | **Note** | **Team** | |  | Đỗ Huỳnh Long |  |  | Penentrace | |  | Lâm Chí Hiếu |  |  | Conexus 2 | |  | Phạm Ngọc Linh |  |  | Conexus 2 | |  | Nguyễn Lê Minh Khôi |  |  | Conexus 2 | |  | Đàm Quang Định |  |  | Conexus 2 | |  | Nguyễn Việt Tùng |  |  | Conexus 2 | |  | Lương Trọng Nghĩa |  |  | Conexus 2 | |  | Phạm Danh Vũ |  |  | Conexus 2 | |  | Đào Ngọc Khải |  |  | Survey Tool | |

**Trainers:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Group 1** | **Group 2** | **Group 3** | **Group 4** |
| Nhat Van  ?TBD | Nhat Van  ?TBD | Ville  Công Phạm  Nhựt Đinh | Ville  Công Phạm |

**Timing:**

|  |  |  |
| --- | --- | --- |
| **Group 1** | **Group 2** | **Group 3** |
|  |  |  |
|  |  |  |
|  |  |  |

**Notes:**

|  |  |  |
| --- | --- | --- |
| **Group 1** | **Group 2** | **Group 3** |
|  |  |  |