<div align="center">

PROJECT SPM - CDs MANAGER
Instructor: Prof. Robert Strandh
Group:
Nguyen An Bien
Vien Thanh Nha
Nguyen Thi Ngoc An
Nguyen Thanh Hung

GIT MANUAL GUIDE
Nguyen An Bien

</div>

1. Project information

account: cdmanagerpuf
email: squallnickybien@gmail.com
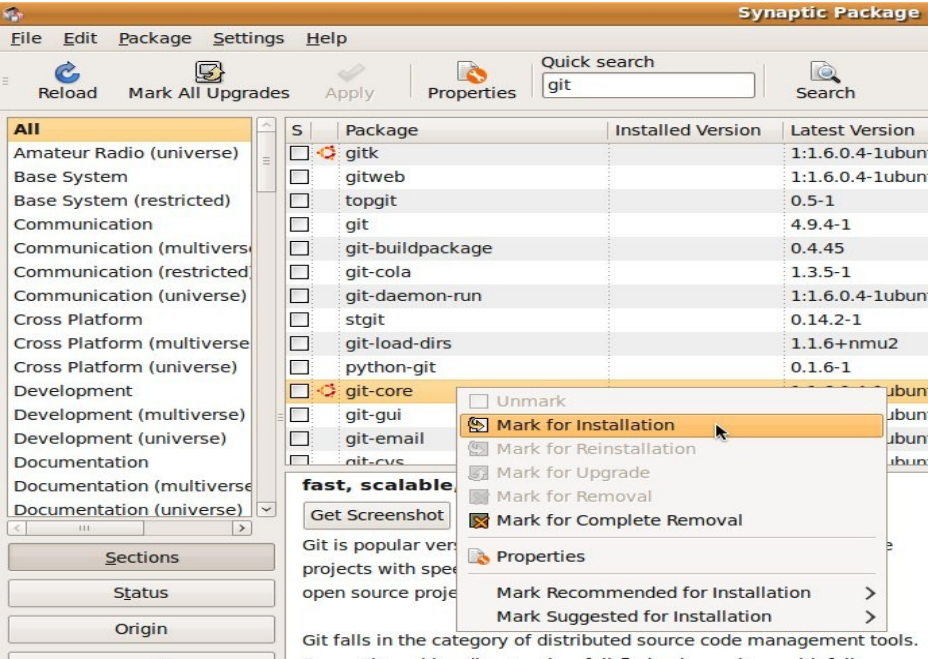password: 123cdmanagerpuf
project name: CDManager
Homepage URL: https://github.com/cdmanagerpuf/CDManager
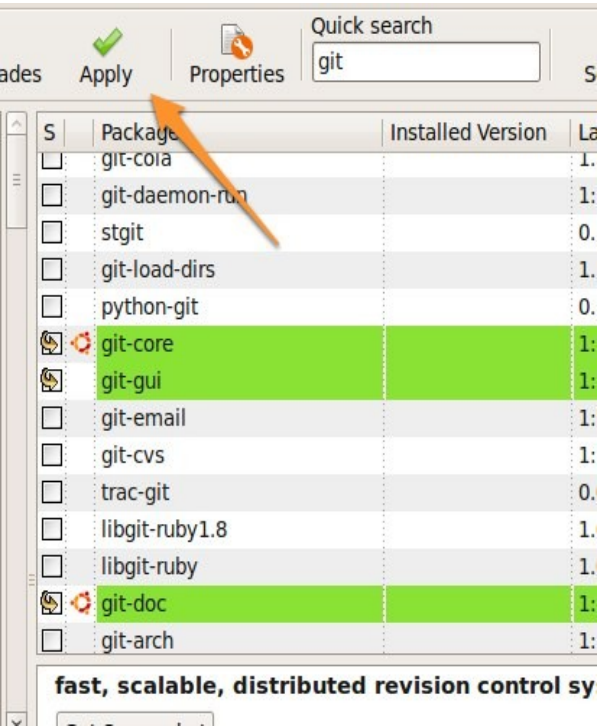
2. Installing git (Linux)

Launch Synaptic:

Select the packages: git-core, git-gui, and git-doc

Click apply to install:

2. Generating SSH keys (Linux)

- Open Terminal

```
$ ssh-keygen -t rsa -C [your email]
Generating public/private rsa key pair.
Enter file in which to save the key (/home/[your home folder]/.ssh/id_rsa):
Enter passphrase (empty for no passphrase): 123cdmanagerpuf
Enter same passphrase again: 123cdmanagerpuf
Your identification has been saved in /home/[your home folder]/.ssh/id_rsa.
Your public key has been saved in /home/[your home folder]/.ssh/id_rsa.pub.
The key fingerprint is:
01:0f:f4:3b:ca:85:d6:17:a1:7d:f0:68:9d:f0:a2:db squallnickybien@gmail.com
```

- Adding the key to your GitHub account

Now launch your browser and open the account page (https://github.com/account). In the "SSH Public Keys" section click "add another public key", then paste your public key into the "key" field. If you leave the title blank the key comment (your email) will be used for the title.

Make sure you use the public key (id_rsa.pub in our example), and do not add any newlines or whitespace inside the key. To ensure you copy the key correctly, you can copy the key directly into the clipboard from the terminal using xclip:

```
$ sudo apt-get install xclip
$ cat ~/.ssh/id_rsa.pub | xclip -sel clip
```

3. Testing things out

Testing if our new key works is simple, we just ssh to github. If this is the first time you're connecting you'll have to save the server's key. You can confirm the fingerprint on the SSH keys tab of the account settings page

```
$ ssh git@github.com
The authenticity of host 'github.com (207.97.227.239)' can't be established.
RSA key fingerprint is 16:27:ac:a5:76:28:2d:36:63:1b:56:4d:eb:df:a6:48.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'github.com,207.97.227.239' (RSA) to the list of known hosts.
ERROR: Hi tekkub! You've successfully authenticated, but GitHub does not provide shell access
Connection to github.com closed.
```

If you do not get the "successfully authenticated" message, check out the troubleshooting guide (http://github.com/guides/addressing-authentication-problems-with-ssh).

4. Next steps: get all the data from server

```
mkdir CDManager
cd CDManager
git clone https://github.com/cdmanagerpuf/CDManager.git
```

Sử dụng GIT:

Khoi tao project: git init
Lay toan bo du lieu tu REMOTE ve: git clone git@github.com:cdmanagerpuf/CDManager.git
----
Thêm file cho GIT theo dõi: git add [file muốn add]
Xóa file không cho GIT theo dõi (xóa khỏi đĩa cứng luôn): git rm

----
commit -a -v -m "nội dung comment" (-v nghia la tu dong luu vet cac thay doi)
commit -a : lam the nay thi do mat cong dung lenh git add
flex
merge
push (chỉ được push lên origin)

Vd: để add thêm 1 file README đưa lên Remote

```
git init
touch README
```

```
git add README
git commit -m 'first commit'
git remote add origin git@github.com:cdmanagerpuf/CDManager.git
git push -u origin master
```

Xem lại history: gitk
Xem remote repository: git remote -v
Lấy dữ liệu từ Remote Repository về:

```
git pull git@github.com:cdmanagerpuf/CDManager.git (pull là lấy về và merge luôn)
git push git@github.com:cdmanagerpuf/CDManager.git[branch trên máy trạm]
```

-------------------------- BRANCH va MERGE ----
Xem vi du rat hay trong sach progit
Tao nhanh: git branch [name]
Chuyen sang nhanh khac: git checkout [branch name]
Phuc hoi file da lo thay doi: git checkout -- [ten file]
Xoa nhanh: git branch -d [branch name]

Merge nhánh hiện tại với nhánh cần: git merge [branch name]
Nó sẽ tổng hợp các chỗ thêm bớt của cả 2 nhánh lại thành 1 phiên bản, tuy nhiên nếu
có những chỗ cùng sửa (vd cùng sửa 1 hàm), thì sẽ bị conflict. Ta mở Notepad ra để sửa
bằng tay các conflict đó (GIT có đánh dấu), sau đó "git add" và commit lại.