# Green Team Documentation

--SAXGOR Security

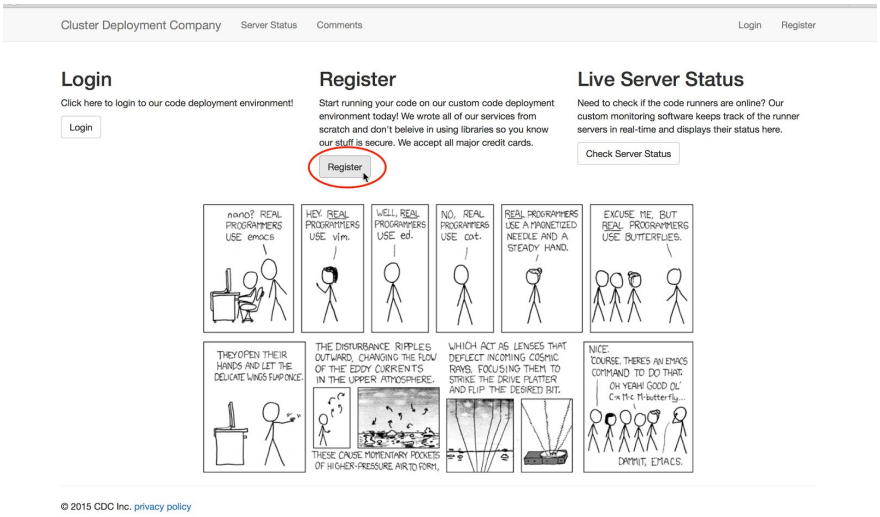
Call us for Technical Support at **232-2506**

# Table of Contents

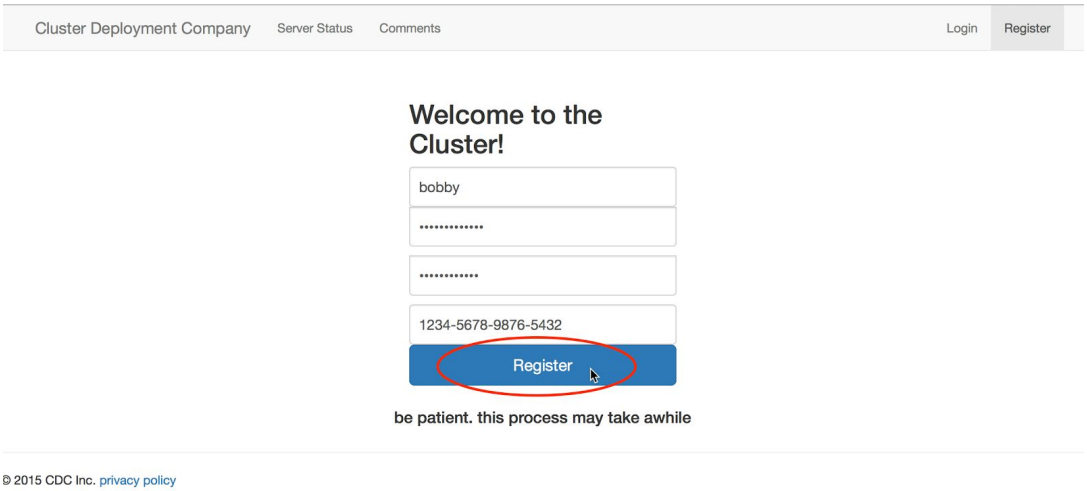
# User Guide

## Creating and Managing Your Account

1. **Creating a new user account**
   - **1.1.** Open your favorite browser and navigate to the main website at www.team6.isucdc.com
   - **1.2.** Click the Register button located in the middle of the page




   - **1.3.** On the registration page fill out the form with the desired account credentials
   - **1.4.** Click Register to complete the registration



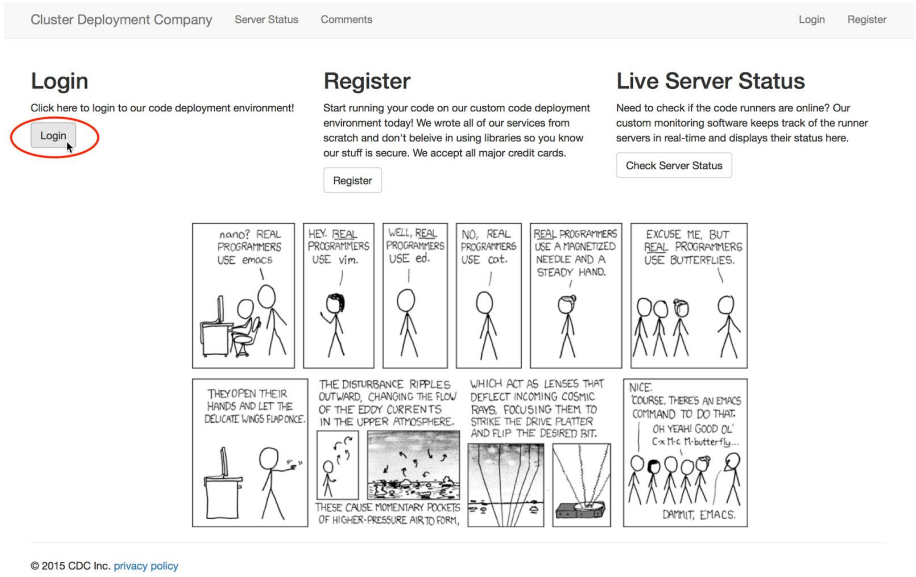

Note: It may take a few minutes before the registration is processed. If after 5 minutes you are still unable to log on to your account please call our Technical Support at **232-2506**

## 2. Logging In

Open your favorite browser and navigate to the main website at [www.team6.isucdc.com](http://www.team6.isucdc.com)

**2.1.** Click the Login button located on the left side of the page

**2.2.**

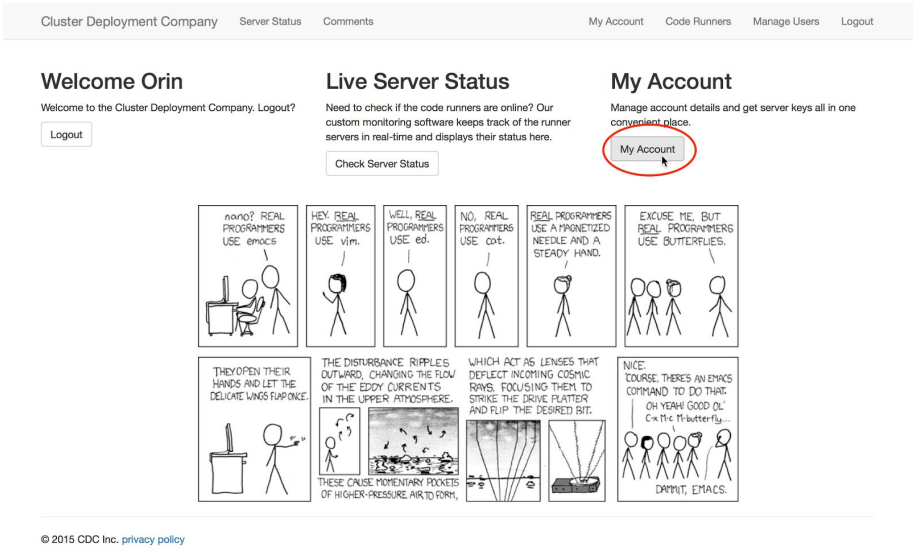



**2.3.** Enter your username and password in their respective boxes

**2.4.** Click Sign in to complete the login and be redirected to the main page

## 3. Changing your password

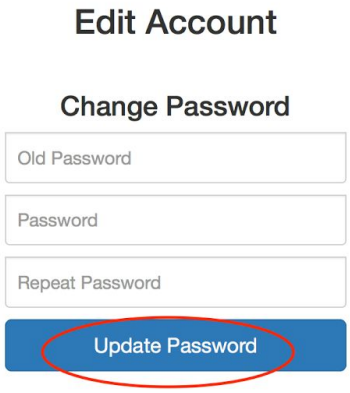
**3.1.** Once logged in to the website locate and click the My Account button at the top right of the page




**3.2.** Look on the right side of the Account Management page for the form marked ‘Change Password’

**3.3.** Fill out the form by entering your current password and the desired password

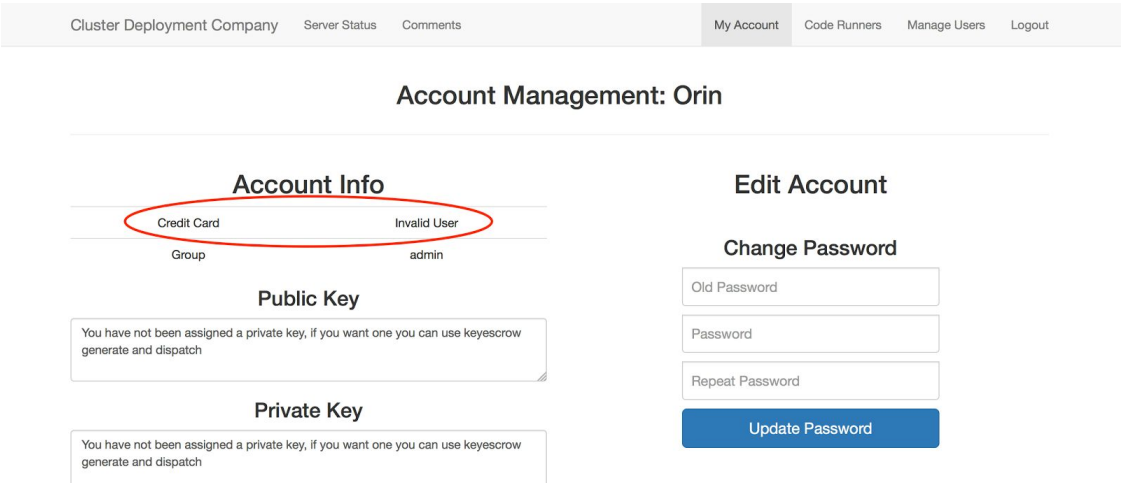
**3.4.** Click Update Password and your password will be updated in our database



## 4. Viewing and updating credit card information

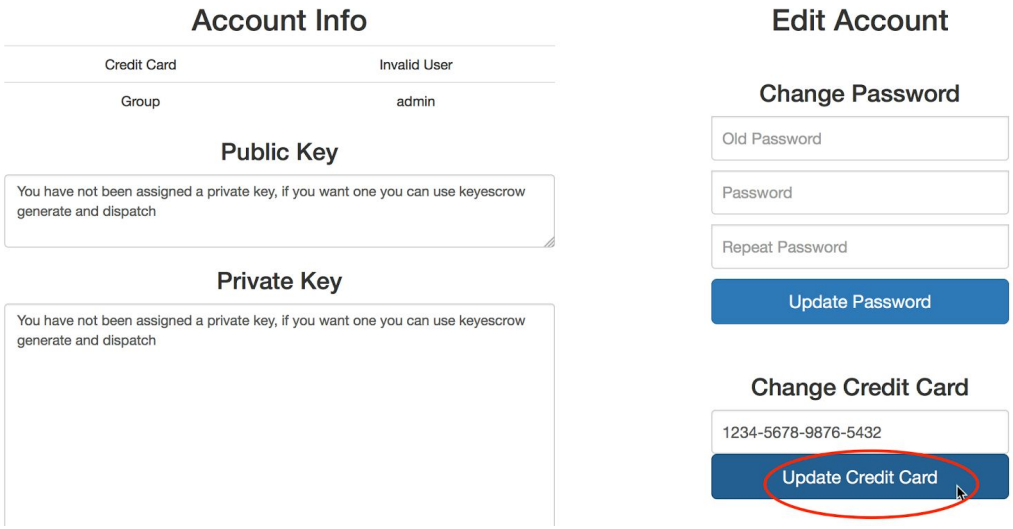
**4.1.** When logged in to the website locate and click the My Account button at the top right of the page

**4.2.** On the left side of the Account Management page you will see your current credit card number




**4.3.** If you look on the right side of the page you will see a form marked 'Change Credit Card'

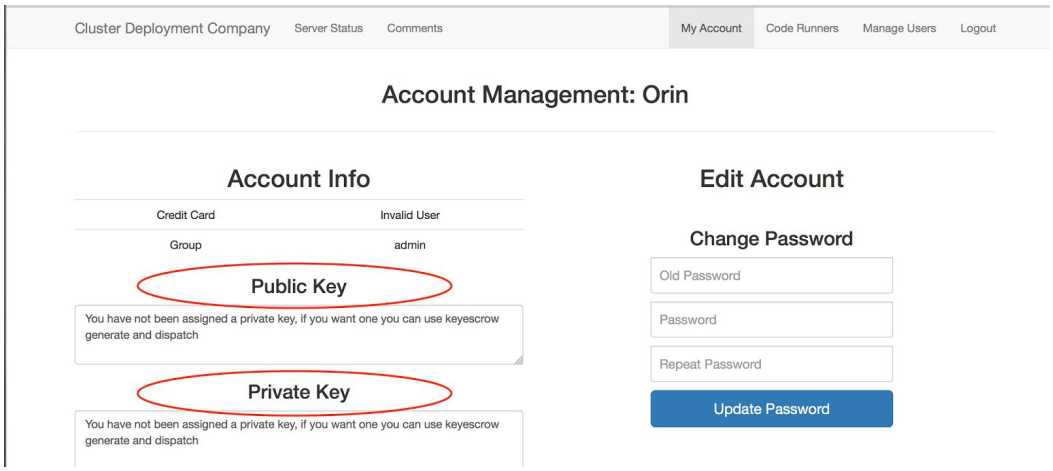
**4.4.** Enter the desired Credit Card number and click the Update Credit Card



## 5. Viewing your public and private keys

**5.1.** When logged in to the website locate and click the My Account button at the top right of the page

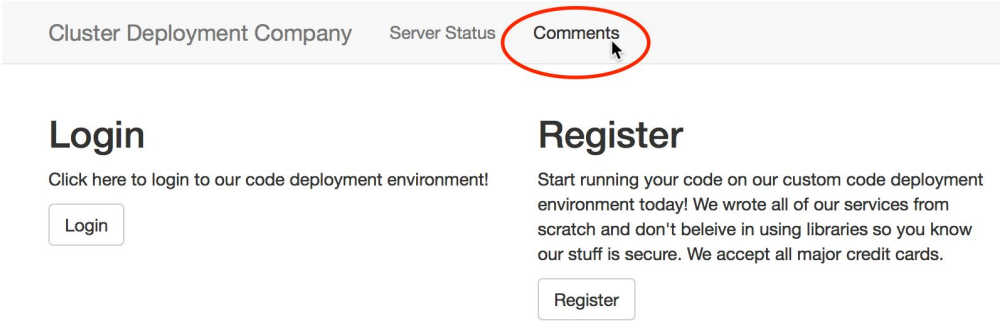
**5.2.** On the Left side of the Account Management page you can will be able to see both your public and private keys



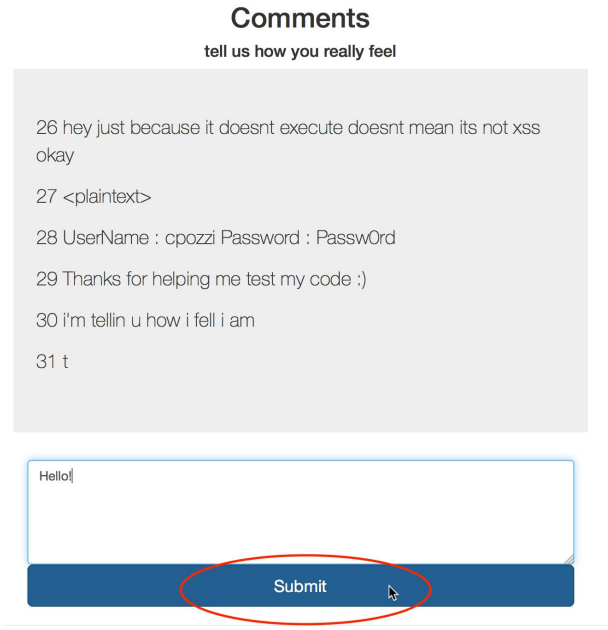

## 6. Leaving a comment

**6.1.** Open your favorite browser and navigate to the main website at [www.team6.isucdc.com](http://www.team6.isucdc.com)

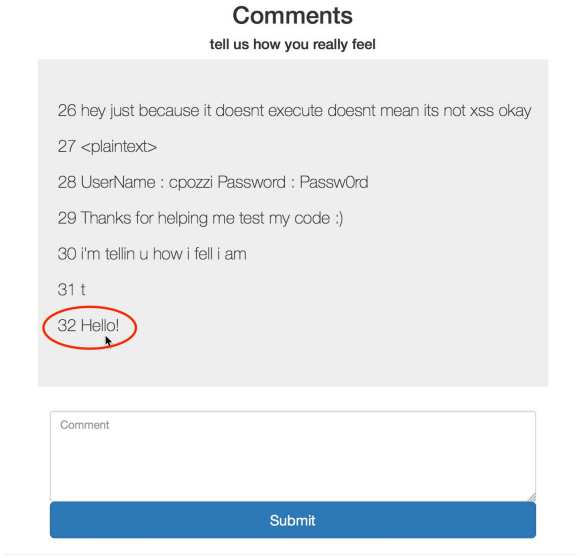
**6.2.** Locate and click the Comments button at the top of the page near our company name



**6.3.** Type your comment in the box near the bottom of the page and click Submit




**6.4.** Your comment will now appear in the gray box below any other recent comments made

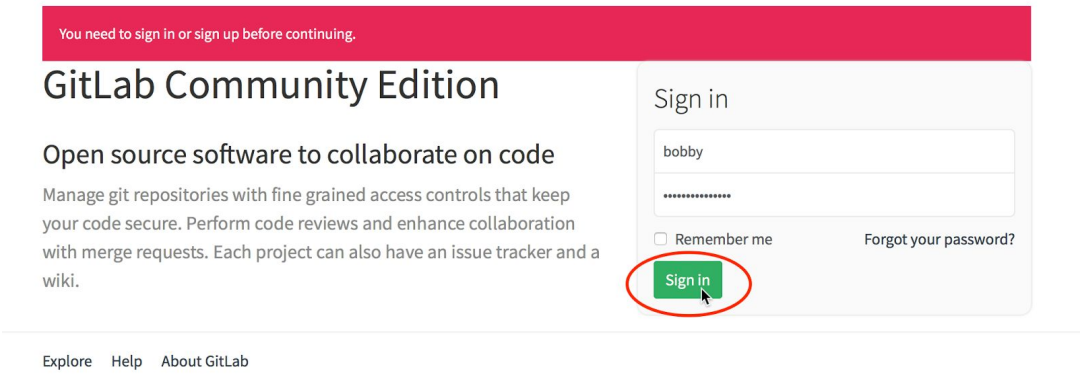


# Developing and Uploading Your Code

## 1. Logging in to Gitlab

**1.1.** Open your favorite browser and navigate to our gitlab site at [git.team6.isucdc.com](git.team6.isucdc.com)

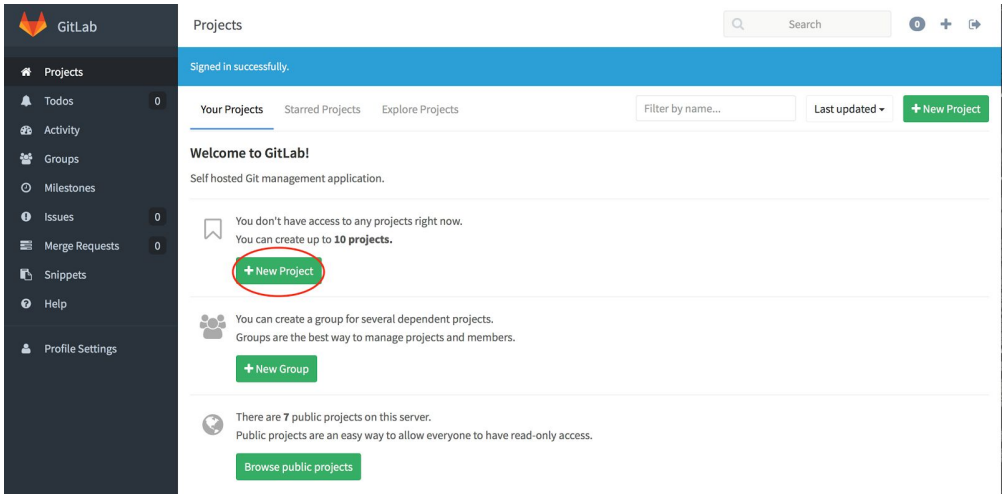
**1.2.** Sign in with your CDC username and password.



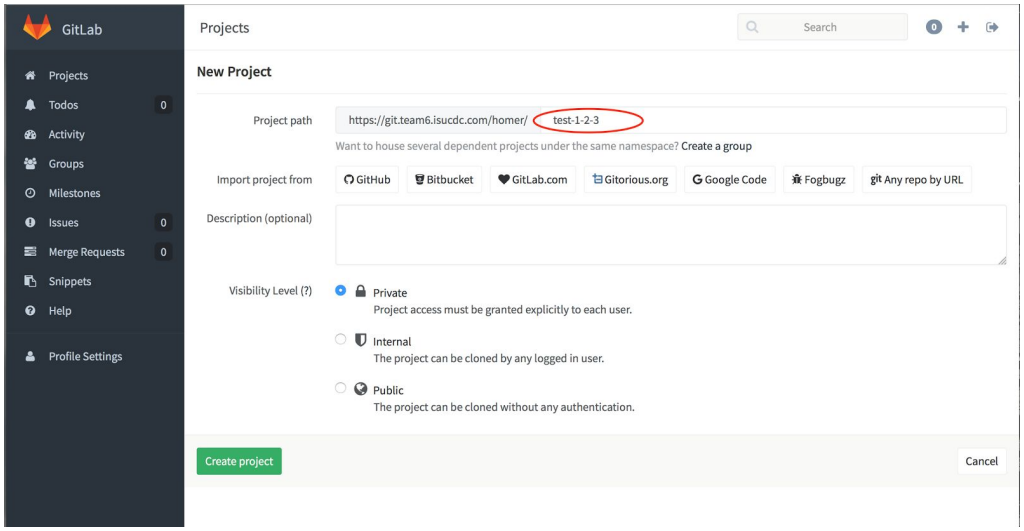

## 2. Creating a new Gitlab Project

**2.1.** Click the **Projects** tab on the left hand pane.

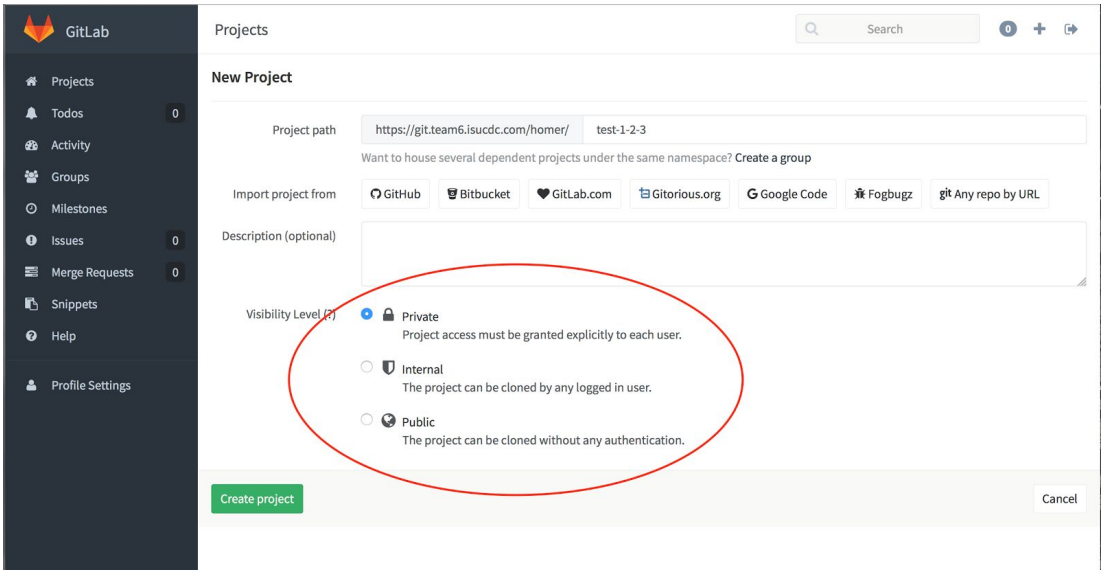
**2.2.** Click **New Project**.



**2.3.** Enter a name for your project in the **Project path**. (Do not include spaces in the name).



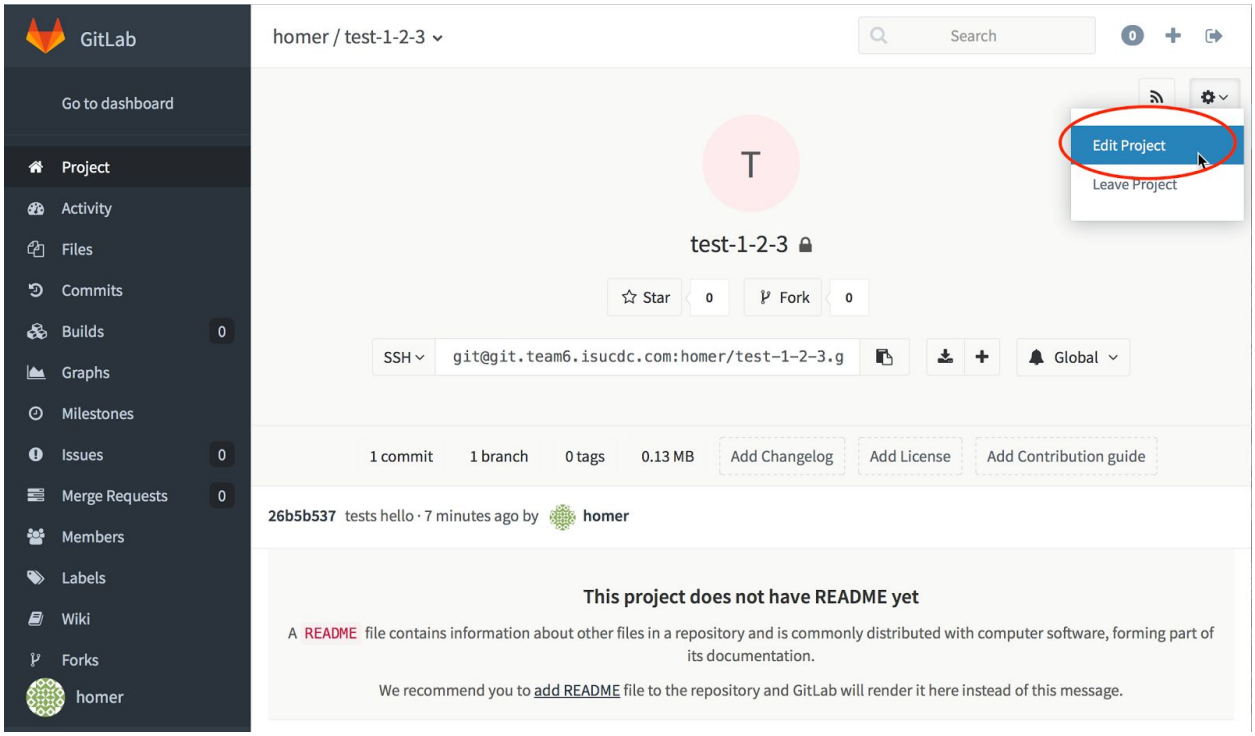

**2.4.** Decide if you want your project to be **Private**, **Internal**, or **Public**. (You can change this later by clicking **Edit** when a project is opened).



## 3. Editing an existing Gitlab Project

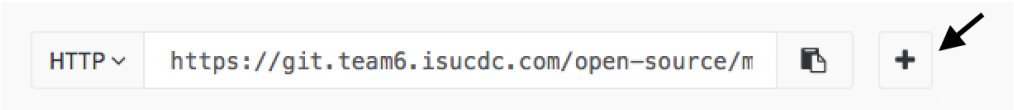
**3.1.** Click the **Projects** tab on the left hand pane.

**3.2.** Click the project you would like to edit.




**3.3.** Add new Files to your project:

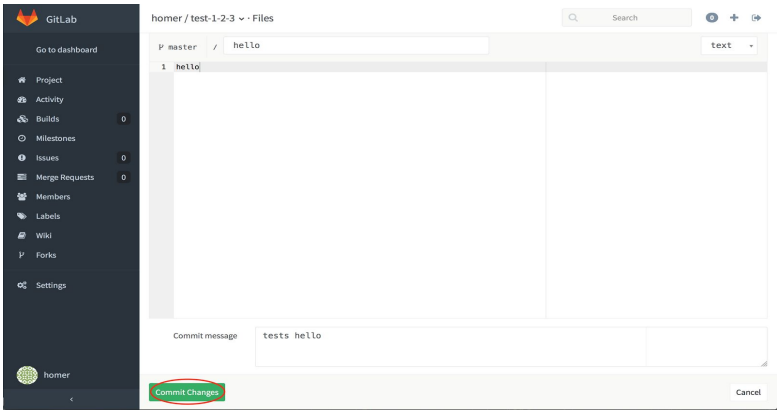
**3.3.1.** Click the + icon in the center of screen (See image). A drop down menu will appear. Click **New file**.




**3.3.2.** The Gitlab text editor will open, allowing you to create your file.

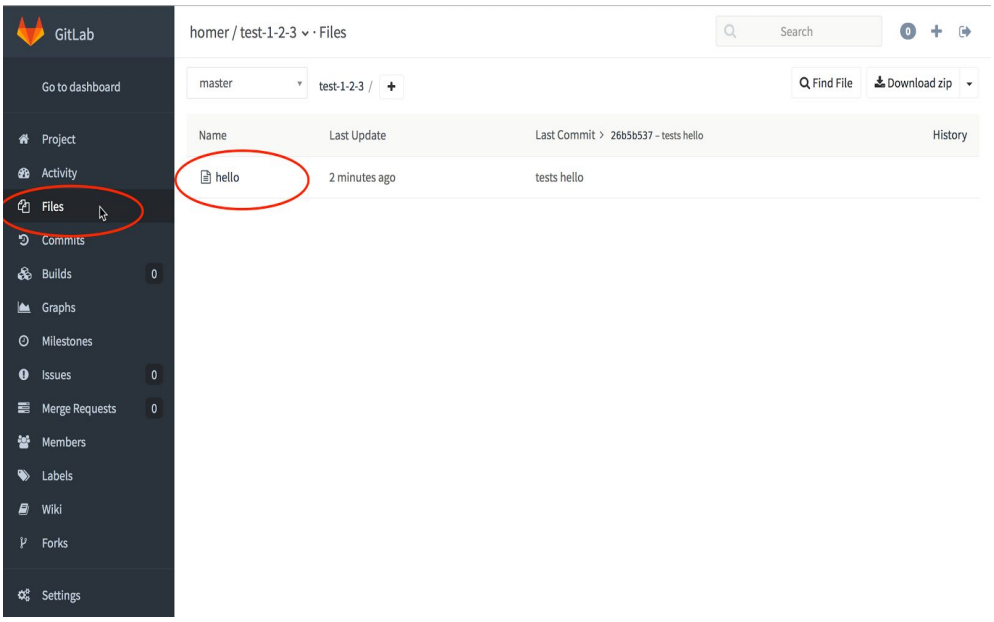
**3.3.3.** When you are finished editing, fill in the **Commit message** to explain what happened to the file. *(Example: "Initializes README file.")*

**3.3.4.** Click **Commit Changes**.



**3.3.5.** See the new file by clicking **Files** on the left hand pane. (You may need to refresh your page).

**3.3.6.** Your new file should be listed under the column **Name**.

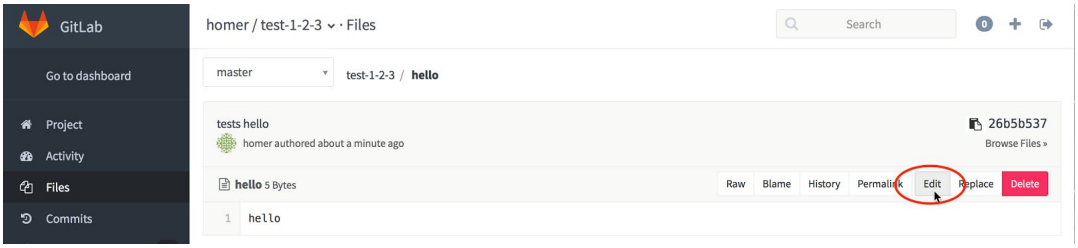



**3.4.** Editing existing files:

**3.4.1.** Click **Files** on the left hand pane.

**3.4.2.** Under the column **Name**, select the file you would like to edit.

**3.4.3.** On the right side of the page, click **Edit**.




**3.4.4.** The Gitlab text editor will open, allowing you to edit your file.

**3.4.5.** When you are finished editing, fill in the **Commit message** to explain what happened to the file. *(Example: "Adds Title to README file.")*

## 4. Developing Code on Shell Server

**4.1.** Using the Shell Server to edit files

**4.1.1.** Make sure to get keys first. Refer to page 14 in the section On the Shell Server

**4.1.2.** Login to the Shell Server using an command line interface. (See instructions at end of document for more information).

```
ssh username@shell.team6.isucdc.com
```

**4.1.3.** Navigate to your home directory.

```
cd ~
```

**4.1.4.** Create a project directory for your code.

```
mkdir my_project
```

**4.1.5.** Navigate into the new project directory.

```
cd my_project
```

**4.1.6.** Develop your code. (The following opens a text editor called Nano. We suggest looking at tutorials online on how to use this text editor.).

```
nano text_file.txt
```

**4.1.7.** Hit control+X to save your code. You will be asked “Save modified buffer?” Hit “y” to save and enter to keep the original file name.

**4.1.8.** You can observe any files you have made with the **list** command.

```
ls
```

**4.2.** Configure your shell project to Gitlab

**4.2.1.** Ensure the keys are updated for Gitlab. (These commands only need to be run once).

```
keyescrow get
keyescrow dispatch
gitlab-update-keys
```

**4.2.2.** Navigate to to your project directory.

```
cd ~/my_project
```

**4.2.3.** Initialize the project to be used with Git.

```
git init
```

**4.2.4.** Copy the SSH project link of one of your Gitlab projects that you want to link to this shell project. (See Image).


**4.2.5.** Paste that link into the following command line statement.

```
git remote add origin paste-here
```

**4.3.** Syncing your shell project with your Gitlab project. (“Pulling a Gitlab project”)

**4.3.1.** Ensure you have already configured you project with Gitlab. (Section 4.2)

**4.3.2.** Ensure you are in your project directory.

```
cd ~/my_project
```

**4.3.3.** “Pull” your Gitlab project files to your local directory.

```
git pull origin master
```

**4.3.4.** Observe that files were downloaded correctly.

```
ls
```

**4.4.** Uploading your shell project to your Gitlab project. (“Pushing a project to Gitlab”).

**4.4.1.** Ensure you have already configured your project with Gitlab and synced your project with Gitlab. (Sections 4.2 and 4.3).

**4.4.2.** After you are finished editing files. Perform the following commands. (Be sure to keep the quotation marks.

```
git add .
git commit -m “Explain what you changed here.”
```

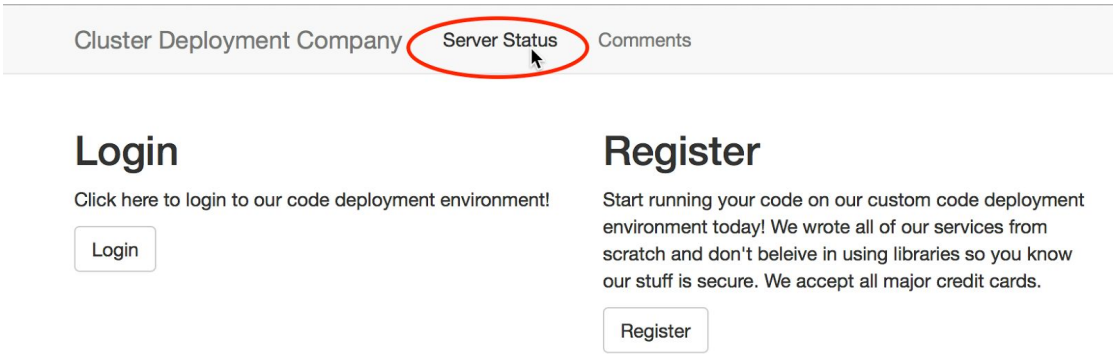
**4.4.3.** Now you can upload your local project to Gitlab.

```
git push origin master
```

# Building and Running Your Code

## On The Website

1. **Viewing the server status of the Runner servers**
   - **1.1.** Open your favorite browser and navigate to the main website at www.team6.isucdc.com
   - **1.2.** Locate and click the Server Status button at the top of the page next to our company name




   - **1.3.** This page will show the server status of Runner 1 and Runner 2, our runner servers. If the server is currently online there will be a list of running processes below the server name. If a server is offline there will be nothing listed underneath the server name.

2. **Using the CodeRunner Web Application**

   Sorry, you can't do this. Use the Shell server like a real programmer.

## On the Shell Server

1. **Accessing the Shell server over SSH**
   - **1.1.** Refer to page 10-11, detailed instructions are located there.

2. **Retrieving keys with keyescrow**
   - **2.1.** Input the command

     **`keyescrow get`**
   - **2.2.** Your public and private keys have now been saved within your home directory

3. **Generating keys with keyescrow**
   - **3.1.** Input the command

     **`keyescrow generate`**
   - **3.2.** The new public and private keys are now saved within your home directory

4. **Setting current user public/private keys to files specified**

**4.1.** For public keys input the command
**keyescrow set -p keypath**
**4.2.** For private keys input the command
**keyescrow set -i keypath**

**5. Distributing keys to be used for authentication on all servers**
**5.1.** Input the command
**keyescrow dispatch**

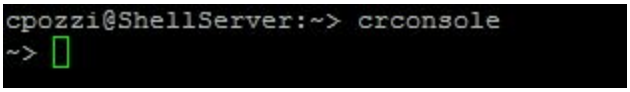
## Using the CodeRunner console

**Note:** Before you use the **`crconsole`** make sure you have retrieved your keys using the **`keyescrow`** command.

**Note:** When using the commands in the instructions below be sure to replace 'project' with the name of your gitlab project and 'runner' with the name of the runner you wish to execute code on. 'runner1' and 'runner2' will be available to you for testing your code.

### *- Starting the CodeRunner console -*

To start the CodeRunner simply type **`crconsole`** in the terminal. If the CodeRunner console starts successfully your terminal will move to the next line containing only **~>**




### *- Deploying your project to a runner -*

To deploy your project from the gitlab runner use the command:

```
deploy :project => :runner
```

### *- Setting stdin for your project (optional) -*

To set an standard input (stdin) for your project you must create a text file containing the desired stdin. To do this we will first exit the console by pressing **Ctrl + Z**.

Now on the command line you can create a text file in your home folder containing your stdin using this command:

```
echo 'stdin goes here' >~/filename
```

After doing this you can confirm the file was created correctly with the command:

```
cat ~/filename
```

Once you have confirmed the contents of the file, use **crconsole** to get back into the CodeRunner console. To use the file we created as the stdin for the project use the command:

```
stdin "~/filename", :project => :runner
```

Finally, confirm the stdin file was uploaded correctly with the command:

```
stdin nil, :project => :runner
```

### *- Building your project -*

**Note:** Be sure to deploy your project to the runner before attempting to build it

To build your project simply use the command:

```
build
```

If you have not run a previous command in the CodeRunner console you may need to specify your project using the long form of the command:

```
build :project => :runner
```

### *- Running your project -*

**Note:** Be sure your project is both deployed and built on the runner before attempting to run it

To run your project simply use the command:

```
run
```

If you have not run a previous command in the CodeRunner console you may need to specify your project using the long form of the command:

```
run :project => :runner
```

### *- Printing the stdout of your project-*

To print the standard output (stdout) after running your project use the command:

```
stdout
```

If you have not run a previous command in the CodeRunner console you may need to specify your project using the long form of the command:

```
stdout :project => :runner
```

### *- Cleaning your Gitlab project -*

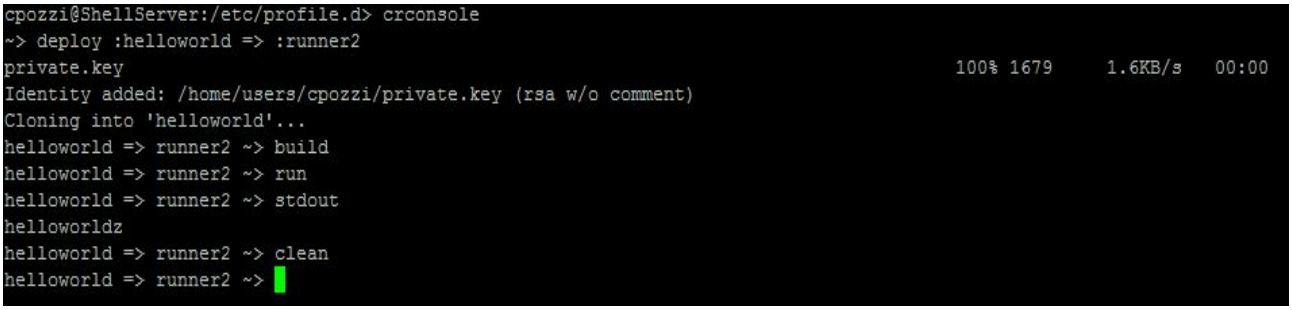
Please clean your project off the runner server when you are finished. To do this use the command:

```
clean
```

If you have not run a previous command in the CodeRunner console you may need to specify your project using the long form of the command:

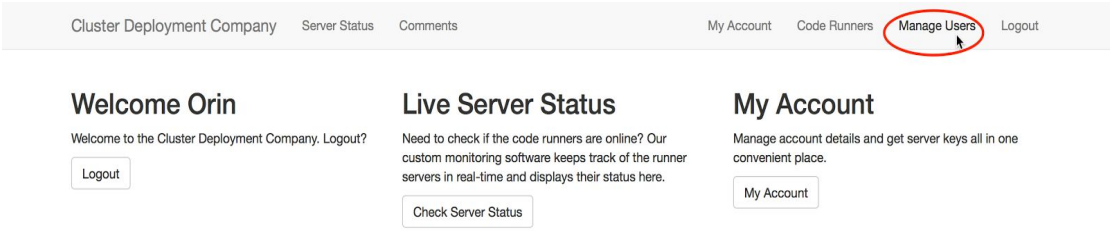
```
clean :project => :runner
```

Here is an example of a full program called ‘helloworld’ being run on the CodeRunner console



# Administration Guide

1. **Viewing a list of user accounts**
   - **1.1.** Login to your Administration account on the main website www.team6.isucdc.com
   - **1.2.** Click the Manage Users button located at the top right of the page




   - **1.3.** Here you can see a list of user accounts currently registered on our servers

2. **Viewing user information**
   - **2.1.** Login to your Administration account on the main website www.team6.isucdc.com
   - **2.2.** Click the Manage Users button located at the top right of the page
   - **2.3.** Click a user's name to be directed to a page where you can view and manage their user information

3. **Changing a user's password**
   - **3.1.** Login to your Administration account on the main website www.team6.isucdc.com
   - **3.2.** Click the Manage Users button located at the top right of the page
   - **3.3.** Click the name of the user's account you wish to update
   - **3.4.** On the right side of the page there is a form labeled 'Change Password'
     - 3.4.1. Refer to section 3. Changing Password in the User Guide

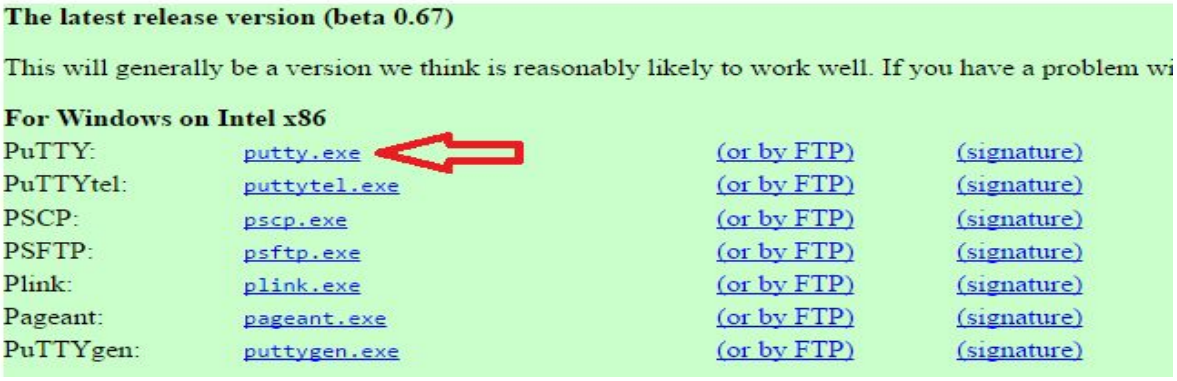
4. **Updating a user's credit card**
   - **4.1.** Login to your account on the main website www.team6.isucdc.com
   - **4.2.** Click the Manage Users button located at the top right of the page
   - **4.3.** Click the name of the user's account you wish to update
   - **4.4.** On the right side of the page there is a form labeled 'Change Credit Card'. Enter the desired new card number and click Update Credit Card and the number will be updated in our databases.
     - 4.4.1. Refer to section 4 of the User Guide

# Accessing servers over SSH

## *Windows*

**Installing Putty:**

To access our servers you will need an ssh client. Putty is an ssh client that is available for free for windows users. It is a standalone executable, which means that all you have to do is download it and it is ready to use. You can find it at:
http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html




**Web Server (www.team6.isucdc.com)**

1. Start PuTTY
2. In the “Host Name (or IP address)” field, enter: www.team6.isucdc.com or 79.5.64.77
3. In the “Port” field, enter “22”
4. Make sure the “SSH” connection type is selected
5. Hit “Enter” or click “Open” at the bottom of the PuTTY screen
6. Login as: <your username>, hit enter.
7. Enter the correct password for that user

**Gitlab Server (git.team6.isucdc.com)**

1. Start PuTTY
2. In the “Host Name (or IP address)” field, enter: git.team6.isucdc.com or 79.5.64.99
3. In the “Port” field, enter “22”
4. Make sure the “SSH” connection type is selected
5. Hit “Enter” or click “Open” at the bottom of the PuTTY screen
6. Login as: <your username>, hit enter.
7. Enter the correct password for that user

**Shell Server (shell.team6.isucdc.com)**

1. Start PuTTY
2. In the “Host Name (or IP address)” field, enter: shell.team6.isucdc.com or 79.5.64.10
3. In the “Port” field, enter “22”
4. Make sure the “SSH” connection type is selected
5. Hit “Enter” or click “Open” at the bottom of the PuTTY screen
6. Login as: <your username>, hit enter.
7. Enter the correct password for that user

**Runner 1**

1. Start PuTTY
2. In the “Host Name (or IP address)” field, enter: 79.5.64.23
3. In the “Port” field, enter “22”
4. Make sure the “SSH” connection type is selected
5. Hit “Enter” or click “Open” at the bottom of the PuTTY screen
6. Login as: <your username>, hit enter.
7. Enter the correct password for that user

**Runner 2**

1. Start PuTTY
2. In the “Host Name (or IP address)” field, enter: 79.5.64.22
3. In the “Port” field, enter “22”
4. Make sure the “SSH” connection type is selected
5. Hit “Enter” or click “Open” at the bottom of the PuTTY screen
6. Login as: <your username>, hit enter.
7. Enter the correct password for that user

**Key Escrow**

1. Start PuTTY
2. In the “Host Name (or IP address)” field, enter: 79.5.64.18
3. In the “Port” field, enter “22”
4. Make sure the “SSH” connection type is selected
5. Hit “Enter” or click “Open” at the bottom of the PuTTY screen
6. Login as: <your username>, hit enter.
7. Enter the correct password for that user

## *Linux/OS X*

If you are a Mac user you already have an ssh client. If you are a linux user you can use the open-ssh client, which is installed in most distros. If you do not have open-ssh installed, you will need to look into getting it from you distro's package manager.

To use your ssh client open a terminal and type "ssh <you username>@<your domain>. For example, a user named "bobby" trying to connect to "awesomedomain.com" would type "ssh bobby@awesomedomain.com" you may also use the IP address, in the example below the IP address is "123.456.7.8".


Note: You may be asked if you wish to continue connecting. If you trust the host, enter "yes". You will then be asked for your login credentials.