

**Link to Redmine:** <http://redmine.shodor.org/>

Redmine is a free open source and web-based project management tool. Using Redmine, you can easily create any number of projects and easily create issues (bugs/features) for your project. Redmine also works best with a bunch of tools (SVN, CVS, GIT, etc.) that allow you to track your project source code modifications.

For the lab, we will be using the Redmine server at Shodor. This will allow to you still create projects but make them private to yourself. If you wish to collaborate with other Redmine members, you will be able to add them as members of your project.

**Redmine Guide:** <https://www.redmine.org/guide>

**Redmine Tutorial:** <https://www.youtube.com/watch?v=sAobJ1kmyqA>

**SVN Cheatsheet:** <http://shodor.org/~mludin/docs/subversion-cheat-sheet-v1.pdf>

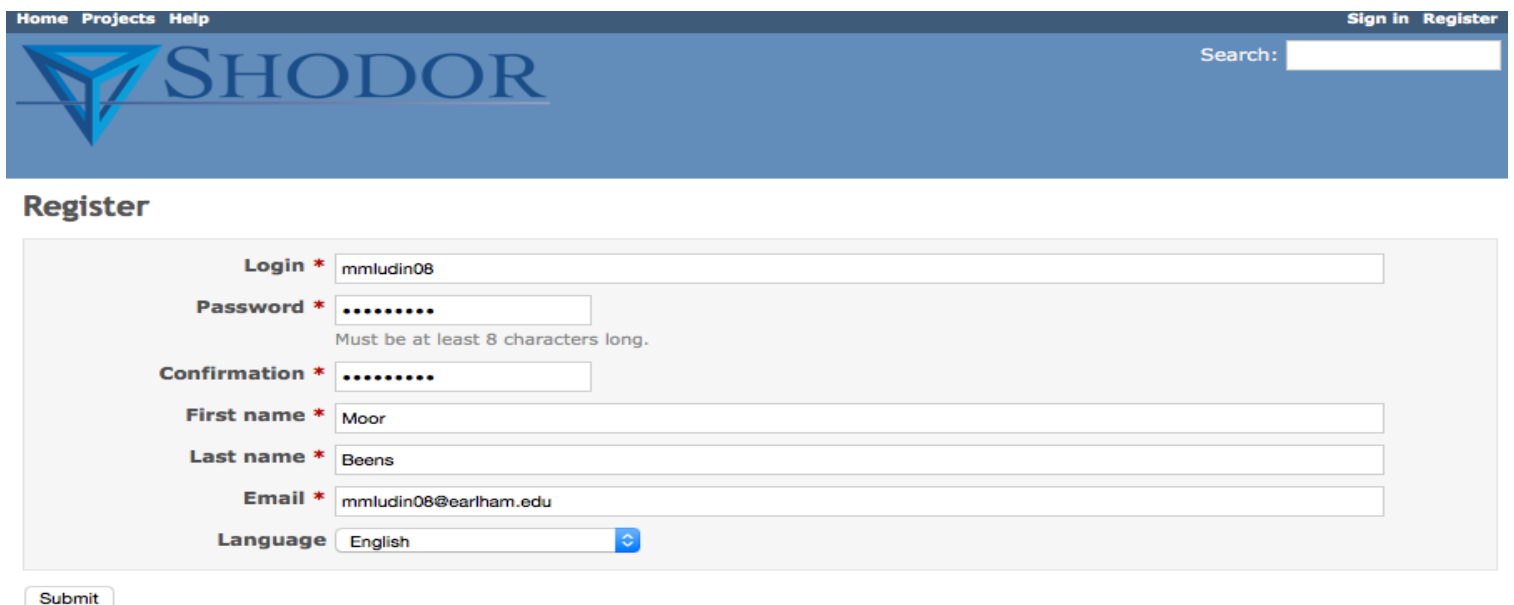
**SVN Book:** <http://svnbook.red-bean.com/en/1.7/svn-book.pdf>

Following are steps for getting started with Redmine:

**Step 1:** Go to following link:

<http://redmine.shodor.org/>

**Step 2:** Top → right corner click on “**Register**”. Which will give you this window:

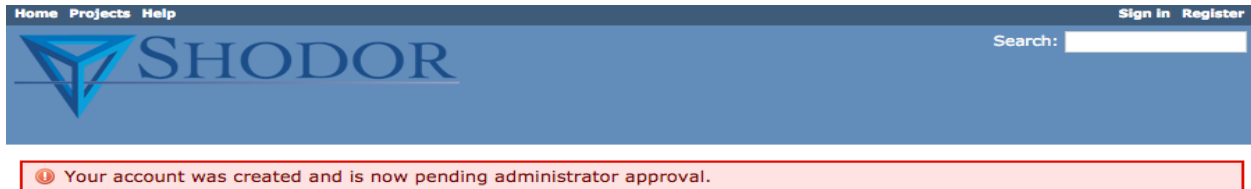


The screenshot shows the 'Register' page of the Redmine application. At the top, there is a navigation bar with 'Home', 'Projects', and 'Help' links on the left, and 'Sign in' and 'Register' links on the right. The Shodor logo is prominently displayed in the center. Below the navigation bar, the page title 'Register' is shown. The main content area contains a registration form with the following fields: 'Login \*' (username) with the value 'mmludin08', 'Password \*' (masked with dots) with a note 'Must be at least 8 characters long.', 'Confirmation \*' (masked with dots), 'First name \*' with the value 'Moor', 'Last name \*' with the value 'Beens', 'Email \*' with the value 'mmludin08@earlham.edu', and 'Language' set to 'English'. A 'Submit' button is located at the bottom left of the form.

Your login (referred to as username) and password do not need to be the same as your Shodor accounts because the Redmine server is not connected to the Shodor authentication servers. If you wish to choose a different login and password, you may.

Once you enter the information → click submit. This will send me an email so that I can approve the account. Please ensure you spell your name correctly so that I know who I am granting access to.

The system will check if the email address has been used previously. If your email hasn't been used for this before, you will see the following page:

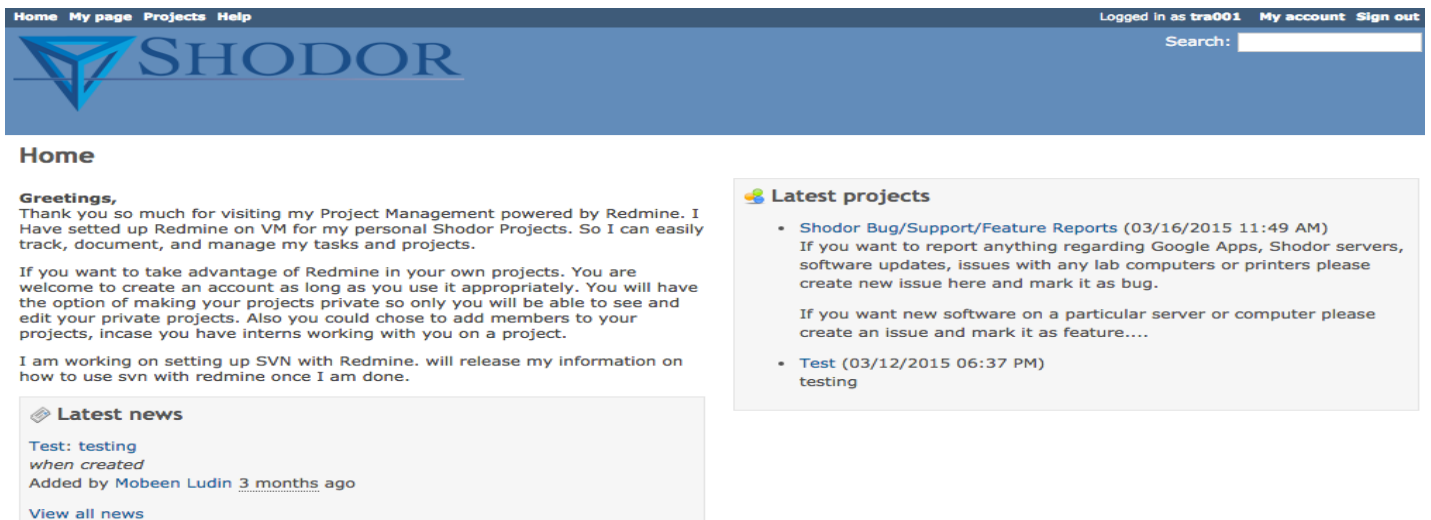


Login:   
Password:   
[Lost password](#)

**Step 3:** Once I approve your request, you should receive an email that your account has been approved. Enter your credentials on the login page and click “**sign in**”.

Login:   
Password:   
[Lost password](#)

**Step 4 - First login:** You should now be able to create/see projects or add issues to current projects. The public projects will be listed on the right side.



If, when you log in, you do not see the page above, click the “**Home**” tab in the top left corner.

To see the issues or tickets that you are in charge of or that you created, click “**My Page**”. Issues/tickets both reported by you and assigned to you by other project members will be listed here. Since you just created your account, these should be empty.

### Step 5: Lets create a project: click **Projects**

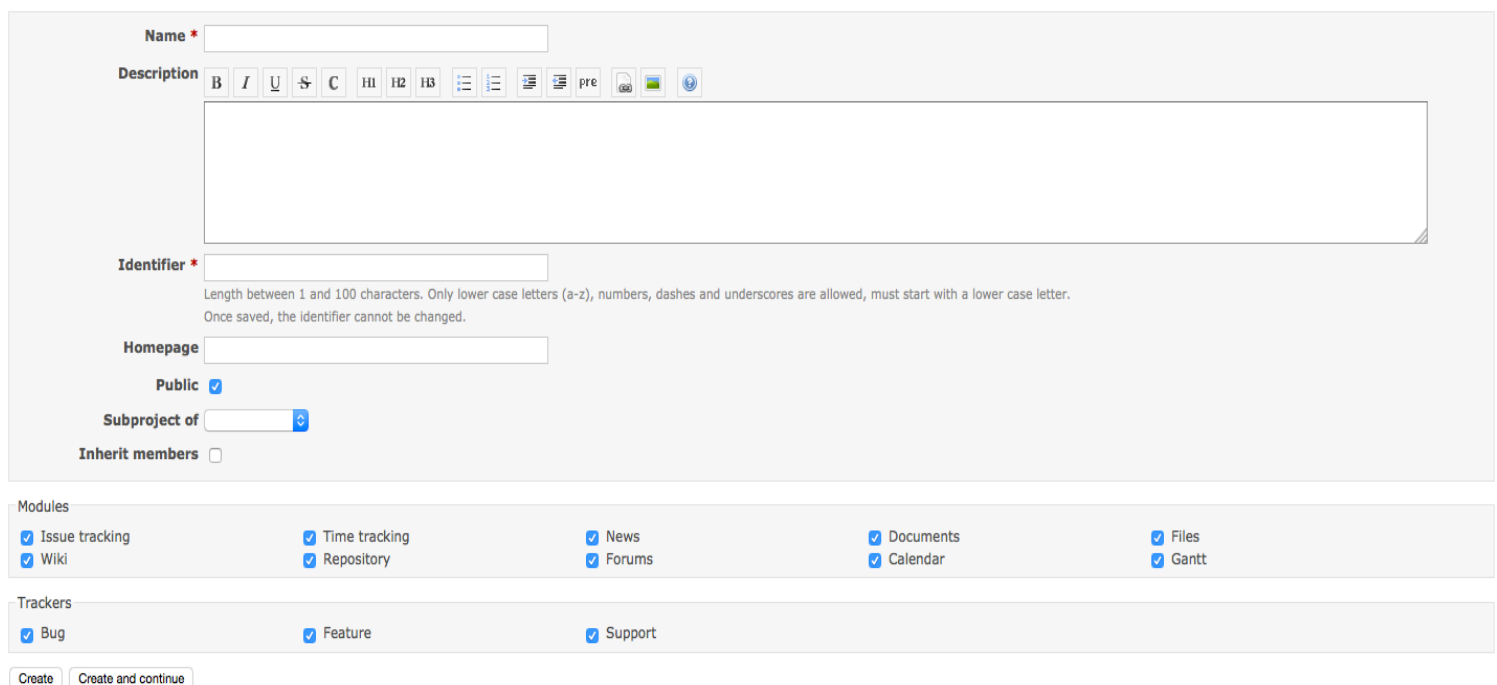
Here you can now see your projects. These include public projects and projects you are member of. To create a new project click on the “**New project**” next to the green circle with “+” sign.



This will open a new window where you would be able to give your project a name, write a description, etc.

The Identifier is one of the most important pieces of information. This is used to create an SVN repository for your project that you can checkout any where.

#### New project



Name \*

Description

Identifier \*

Homepage

Public

Subproject of

Inherit members

Modules

Issue tracking  Time tracking  News  Documents  Files

Wiki  Repository  Forums  Calendar  Gantt

Trackers

Bug  Feature  Support

Create Create and continue

**NOTE:** Please make sure you check the Bug, Feature and Support boxes under Trackers. Also, by default, the project is “**public**” (the box under home page). I suggest you keep your projects private and only add members when needed.

Once you add all the info, click “**Create**”. This should successfully create your project. Then click **Overview** on the left side.

**Step 6 - Play with the project:** Once you click the **Overview** tab on the left side, you should see a window similar to this:

Once your project grows and issues/tickets are created, the **Activity, Issues, Gantt, Calendar, News** and **Files** tabs will become more important.

Things you can do right now:

1. **New Issues:** create new tickets for bugs/feature/support. You should test this out.
2. **Documentation:** where you create formal manual pages and documentation for your project. This includes research articles.
3. **Wiki:** where you can create tutorials and documentation for the software/hardware tools you used for your project in addition to how you used them. If you have written a Wikipedia page before, you should have no problem.
4. **Settings:** where you can change the project overview/description, name, permissions, add members, manage versions, repositories, etc.
  - a. Click: **Settings** → **Repositories** to find out the link for SVN repository.

**NOTE:** you can attach/create any other (Git, CVS, etc) repositories and connect them to your project. The link provided in the **Repository** column will be used to checkout your project on Blue Waters in the next few steps.

**Step 7:** Login to Blue Waters

Training accounts:

```
ssh -Y ludin@bwbay.ncsa.illinois.edu
```

**Step 8:** Go to scratch directory:

```
cd scratch/
```

**Step 8:** Check out your project directory:

```
svn co http://redmine.shodor.org/svn/james-bond --username mmludin08
```

**NOTE:** Let's explain this command line piece by piece:

**svn:** Subversion (SVN) is an open source software used to manage changes in files/directories over time. This is an easy way of creating different versions of your code or document and to track updates/changes.

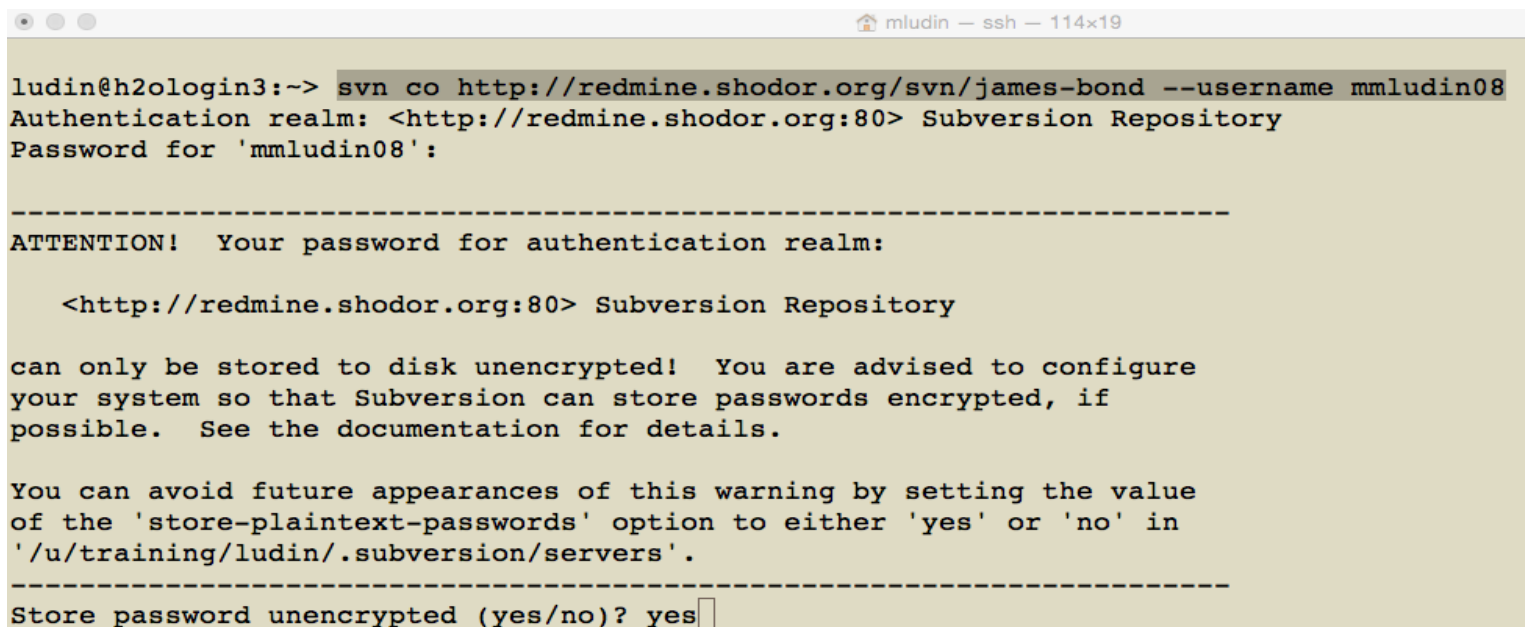
**co:** co stands for checkout. This means you will be making a local copy of a project repository.

**Link:** Replace the link in the example above with the http link specific to you. This can be found in step 6 or login to **Redmine** → **Projects** → **click on your project** → **Settings** → **Repositories**, and copy the link under **Repository** column.

**--username:** Your projects will not be available to anyone else. In order to see the repository, or the checkout of the file, **you must use your Redmine username and password.**

An example of the output from this command is seen below:

~~~~~Note: once you type **“yes”** and enter, you will see this line **“Checked out**



```
mludin -- ssh -- 114x19
ludin@h2ologin3:~> svn co http://redmine.shodor.org/svn/james-bond --username mmludin08
Authentication realm: <http://redmine.shodor.org:80> Subversion Repository
Password for 'mmludin08':

-----
ATTENTION! Your password for authentication realm:

<http://redmine.shodor.org:80> Subversion Repository

can only be stored to disk unencrypted! You are advised to configure
your system so that Subversion can store passwords encrypted, if
possible. See the documentation for details.

You can avoid future appearances of this warning by setting the value
of the 'store-plaintext-passwords' option to either 'yes' or 'no' in
'/u/training/ludin/.subversion/servers'.
-----
Store password unencrypted (yes/no)? yes
```

**revision 0.”**

**Step 9:** Change your directory to the project folder that was just copied:

```
cd james-bond/
```

**Step 10:** Copy all the examples and exercises so far we have used to your project directory. I have most of my example in the `bwi_2015` folder in my home directory. Yours may be different. It is possible you will have to copy your files to your scratch folder.

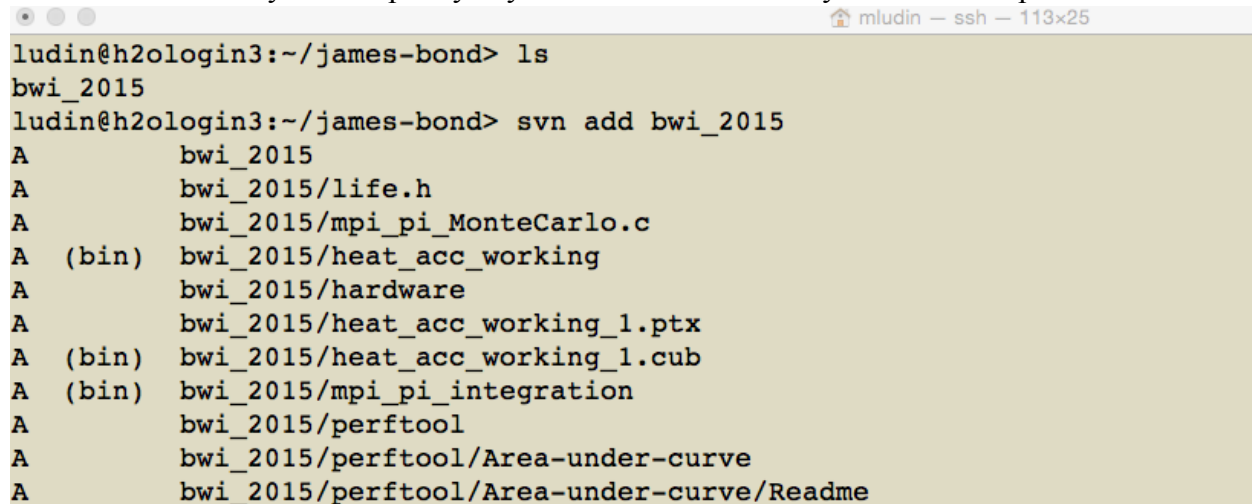
The following command is how I got to my examples:

```
cp -r ~/bwi_2015 .
```

**Step 11:** add your code files and directories to svn.

```
svn add bwi_2015
```

**Note:** after the **add** you can specify any number of file/directory names. Example:



```
mludin - ssh - 113x25
ludin@h2ologin3:~/james-bond> ls
bwi_2015
ludin@h2ologin3:~/james-bond> svn add bwi_2015
A          bwi_2015
A          bwi_2015/life.h
A          bwi_2015/mpi_pi_MonteCarlo.c
A (bin)    bwi_2015/heat_acc_working
A          bwi_2015/hardware
A          bwi_2015/heat_acc_working_1.ptx
A (bin)    bwi_2015/heat_acc_working_1.cub
A (bin)    bwi_2015/mpi_pi_integration
A          bwi_2015/perftool
A          bwi_2015/perftool/Area-under-curve
A          bwi_2015/perftool/Area-under-curve/Readme
```

**Note:** The “**A**” on the far left means that it is adding the files and folders listed on the right to the folder you specified they go into. The **(bin)** means those files are executables.

**Step 11:** now commit your changes/files to svn.

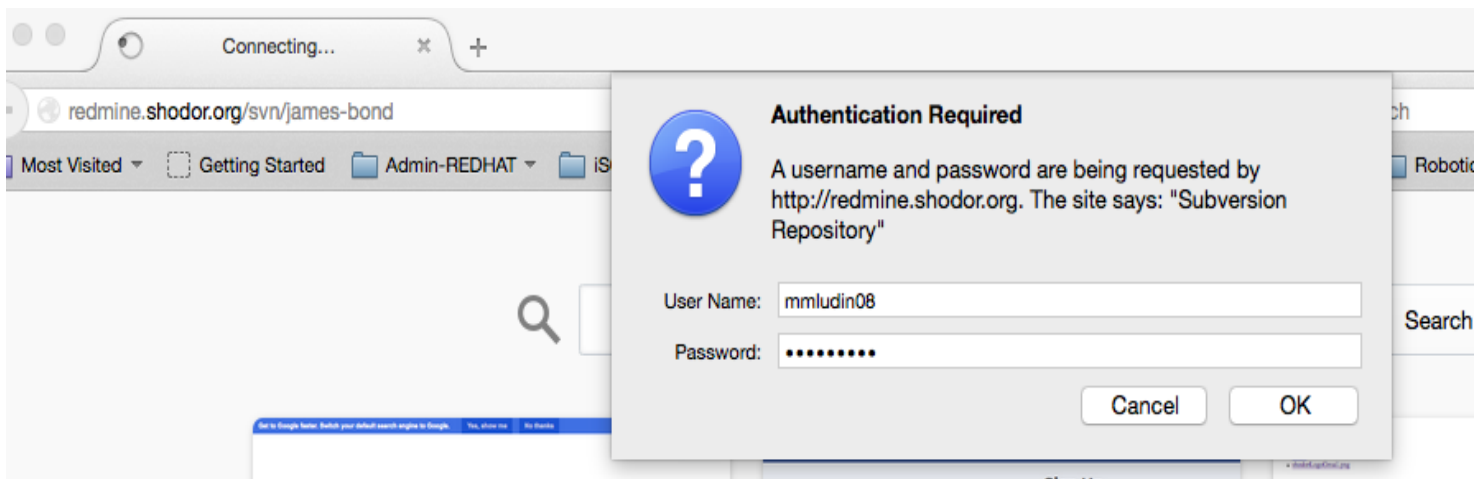
```
svn commit -m “my first svn commit”
```

**Note:** svn commit will now submit your changes to the repository on Redmine server  
The “**-m “my first svn commit”**” is used to add a comment to the version of the file you are committing. This allows you and your other group members to keep track of your changes. Change **my first svn commit** to whatever fits the changes made to your project.

```
Adding (bin) bwi_2015/perftool/vectoradd/addvectors
Adding      bwi_2015/perftool/vectoradd/addvectors.c
Adding      bwi_2015/perftool/vectoradd/addvectors.h
Adding      bwi_2015/perftool/vectoradd/profile_data
Transmitting file data .....
.....
Committed revision 1.
ludin@h2ologin3:~/james-bond> █
```

**REMEMBER:** whenever you do **svn add** or **svn commit**, make sure you are in your project directory. Otherwise **svn add/svn commit** will fail.

**Step 12:** To make sure your changes/files were really committed/submitted to your project repository. Open the project link in a browser and see if it did work.

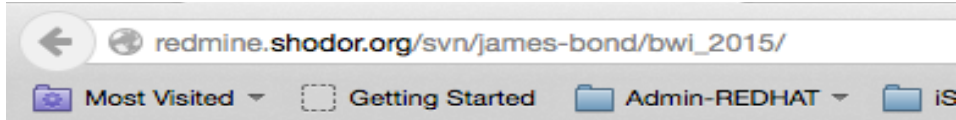


What is shared: the part of the link that is shared between all Redmine users is:

<http://redmine.shodor.org/svn/>

followed by the personal project name. That's why choosing a good **identifier** is important during project creation.

Note: the web-browser will also authenticate you. **You must provide your Redmine username and password.** If you don't, it will fail.



## james-bond - Revision 1: /bwi\_2015

- [..](#)
- [gprof\\_analysisfile.txt](#)
- [hardware/](#)
- [heat\\_acc\\_working](#)
- [heat\\_acc\\_working.c](#)
- [heat\\_acc\\_working\\_1.cub](#)
- [heat\\_acc\\_working\\_1.ptx](#)
- [life-serial](#)
- [life-serial.c](#)
- [life.h](#)
- [mpi/](#)
- [mpi\\_pi\\_MonteCarlo.c](#)

You can click on the directory to browse as well as click on the files. They will be opened in the browser. The executables, though, will not open.

**PLEASE USE REDMINE+SVN, JUST BECAUSE I SET THEM UP!!**

### Additional Resources:

**Useful Unix intro+tutorial (web tutorial):** highly recommended for Unix noobs

<http://www.ee.surrey.ac.uk/Teaching/Unix/unixintro.html>

**Software carpentry (video lectures):** Highly recommended for scientist-programmers

<http://software-carpentry.org/>

Some sections are particularly relevant to this internship program, with links to videos:

- [The Shell](#)
- [Version Control](#) (using Subversion – svn)
- [Testing](#)
- [Python](#)
- [Systems Programming](#) (from Python)
- [Classes and Objects](#) (object oriented programming)
- [Make](#)
- [Matrix programming](#) (linear algebra, NumPy)
- [Regular expressions](#)

**PLEASE USE THESE RESOURCES, JUST BECAUSE I GATHER THEM!!**