

Program Startup Guide

*Whenever you have a problem, **Google** it First.*



Sesame Workshop ©, Sesame Street ® & associated characters, trademarks & design elements are owned & licensed by Sesame Workshop. © 2009 Sesame Workshop. All Rights Reserved.

Lecturer: Huiyu Weng

Presented by TA Group 2015

Regulations

Attendance

Wednesday 18:00 to 20:00 p.m. @ SEIEE Building 4-311

Every one will fill in a seat table. Your seat cannot be changed once selected. We are going to check your attendance by checking the seat table every week. Your attendance may or may not lead to the punishment, which depends on your performance in the Program Test. Full attendance will grant you extra credit if you screw up the Program Test. We do not force students with good programming background to show up every week. But you must hand in your homework before DDLs and perform well enough in the Program Test.

Team

You will team up to review code and discuss issues among your team. Normally a team with no more than 7 students is fine. Each team will have a team leader whose responsibility includes but not limited to disseminating notifications, collecting homework and reporting issues to TA. Team leader can get extra credit for good service to the team.

Dev Environment

C++ IDE

We recommend devc++ as your C++ IDE for short code on Windows systems. Other IDEs like Codeblocks and Visual Studio are also good, but be sure not to use any library not included in standard C++ (<http://www.cplusplus.com/reference/>). If you work on Visual Studio, please create the project with “create empty project” option to avoid including Windows specified libraries.

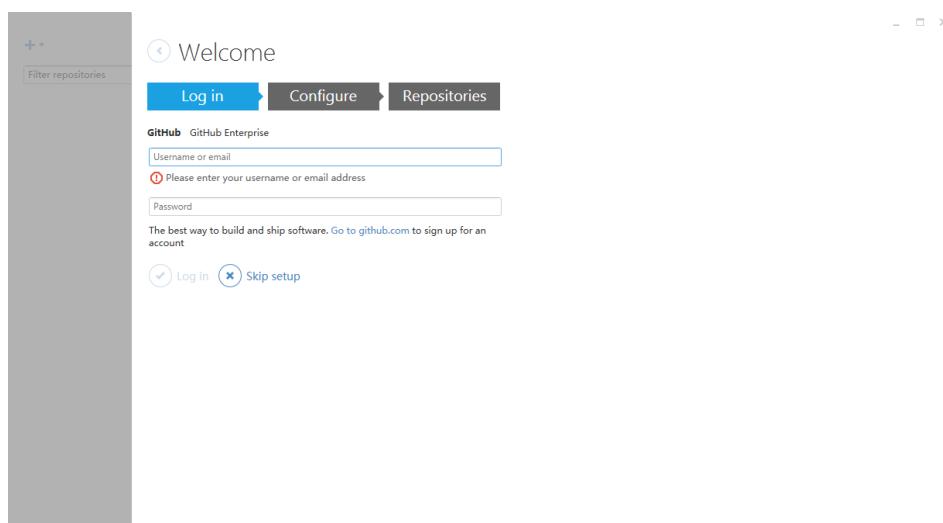
For Mac users, Xcode is fine.

Github

In this course, we will use Github to cross-review your code among team members. If you have no idea about it, please go to wiki <http://en.wikipedia.org/wiki/GitHub>. It is very likely that you still don't know what Github can do after you read this but it's OK. Please follow the instruction to setup your Github account. You can refer to <https://help.github.com/> if you have any problem.

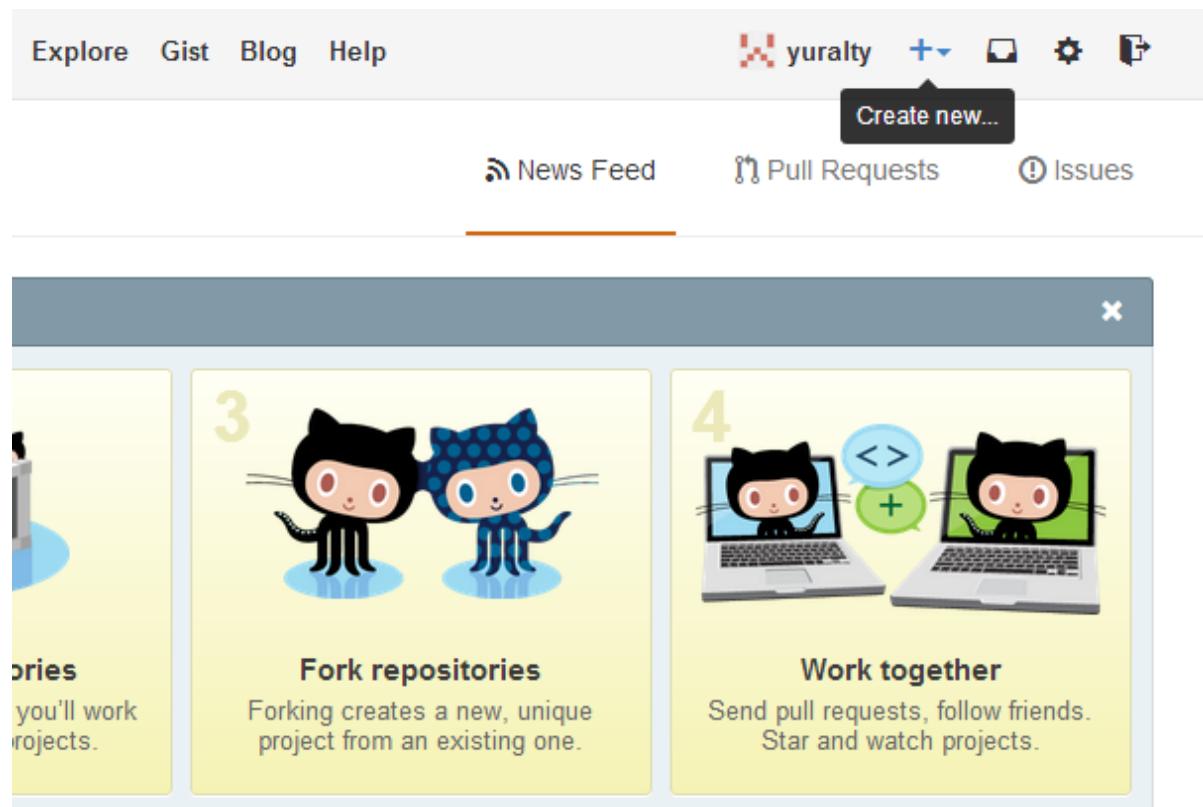
Setup Git

- Download Github client for Windows: <https://windows.github.com/>
- Install it (This will take some time. Go ahead to sign up or read the wiki.)
- Sign up <https://github.com/>
- After installation is complete, login with your account. (*Important: this will generate an SSH key, which is necessary to access your Github from your desktop client. For Linux users, you need to manually generate an SSH key)



Create your first repository

- Create a repo on Github website



The screenshot shows the GitHub homepage with the navigation bar at the top. A prominent 'Create new...' button is highlighted. Below it, there are links for News Feed, Pull Requests, and Issues. A modal window is open, showing two steps: 'Fork repositories' (step 3) and 'Work together' (step 4). Step 3 shows two GitHub octocats, and step 4 shows two laptops with code and a plus sign between them.

Owner: yuralty / **Repository name:** hellogit

Great repository names are short and memorable. Need inspiration? How about [freezing-octo-nemesis](#).

Description (optional): hello world to github

Visibility: **Public**
Anyone can see this repository. You choose who can commit.

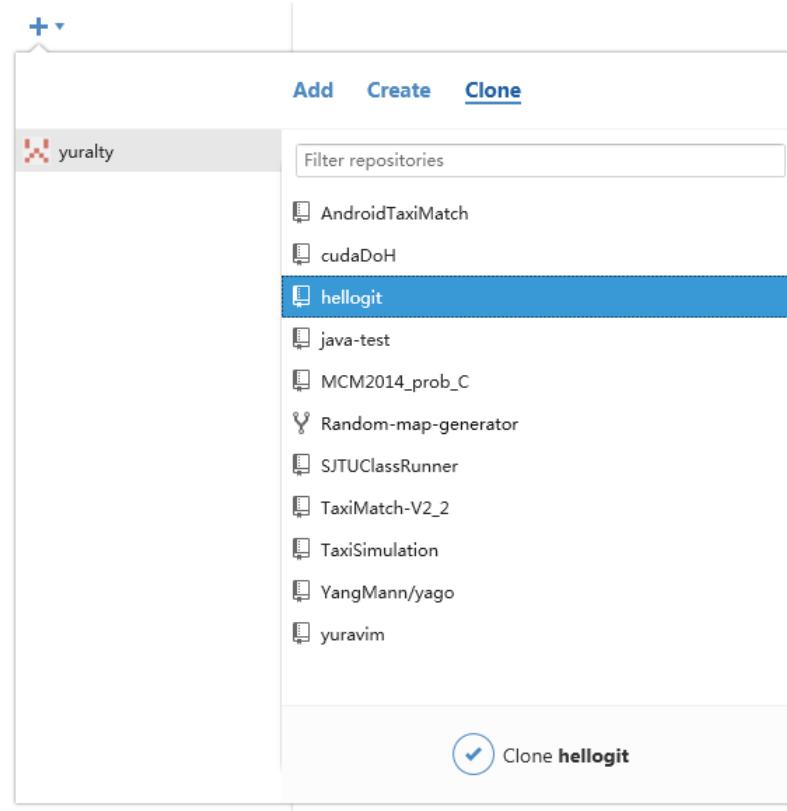
Private
You choose who can see and commit to this repository.

Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** | Add a license: **None** | ⓘ

Create repository

- Clone it to your local repo in your client



Start working on hello world

- Write a hello world .cpp file, compile, and run. Be sure to save the files in your local repo, which by default is C:\Documents\Github\<your repo name>\.

```

File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.8.1 32
(globals)
Project Classes Debug hello.cpp
1 #include <iostream>
2
3 int main() {
4     std::cout << "hello world" << std::endl;
5     return 0;
6 }
7

```

The screenshot shows the Code::Blocks IDE interface. The menu bar includes File, Edit, Search, View, Project, Execute, Tools, AStyle, Window, and Help. The toolbar has various icons for file operations like Open, Save, and Build. The project navigation bar shows 'globals'. The main workspace has tabs for Project, Classes, Debug, and hello.cpp. The code editor displays the following C++ code:

```

1 #include <iostream>
2
3 int main() {
4     std::cout << "hello world" << std::endl;
5     return 0;
6 }
7

```

- In your client, you can see your changes in your local repo. You can now commit your changes to your repo. Usually we do not commit the binary files (.exe file in Windows) to our repo.

The screenshot shows the GitHub desktop application interface. On the left, there's a sidebar with a '+' button, a 'Filter repositories' dropdown set to 'hellogit', and a repository list containing 'hellogit'. The main area is titled 'master' and shows 'Uncommitted changes'. A list box contains 'hello.cpp added' and 'first cpp file'. Below this is a button labeled 'Commit to master' with the note '1 file to be committed'. To the right, under 'Files to commit', 'hello.cpp' is selected. The code editor shows the following C++ code:

```

@@ -0,0 +1,8 @@
#include <iostream>
int main() {
    std::cout << "hello world" << std::endl;
    return 0;
}

```

Below the code editor, another section shows 'hello.exe' with a checkbox next to it. At the bottom, a history pane shows an 'Initial commit' by 'yuralty' 9 minutes ago.

- Commit only make changes to your local repo, to push it to your remote repo, click sync button.

The screenshot shows the GitHub desktop client after performing a sync. The top status bar indicates 'Sync +1'. The main area displays a commit history for the 'ded' branch. The first commit is shown with the message 'sync this branch (ahead by 1 commit, behind by 0 commits)' and a timestamp of '9 minutes ago by yuralty'. Below the commit message is a revert and collapse all button. The commit details show the following changes:

```

@@ -0,0 +1,7 @@
#include <iostream>

int main() {
    std::cout << "hello world" << std::endl;
}

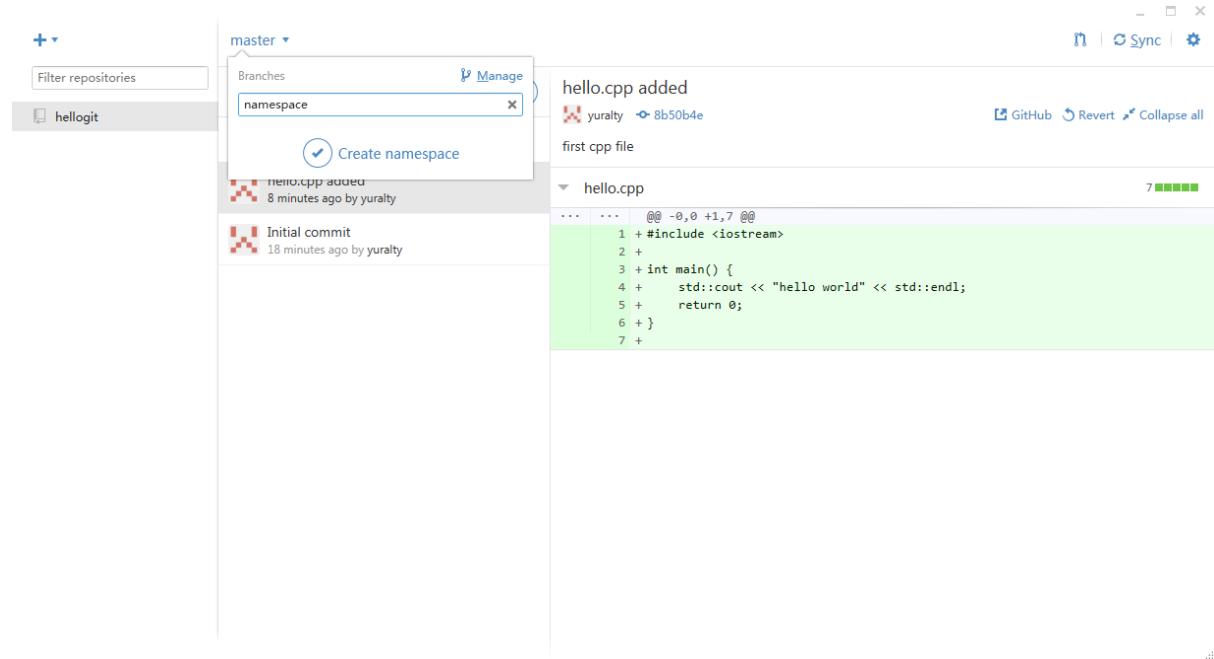
```

A progress bar at the bottom indicates the sync status: 7/7 completed.

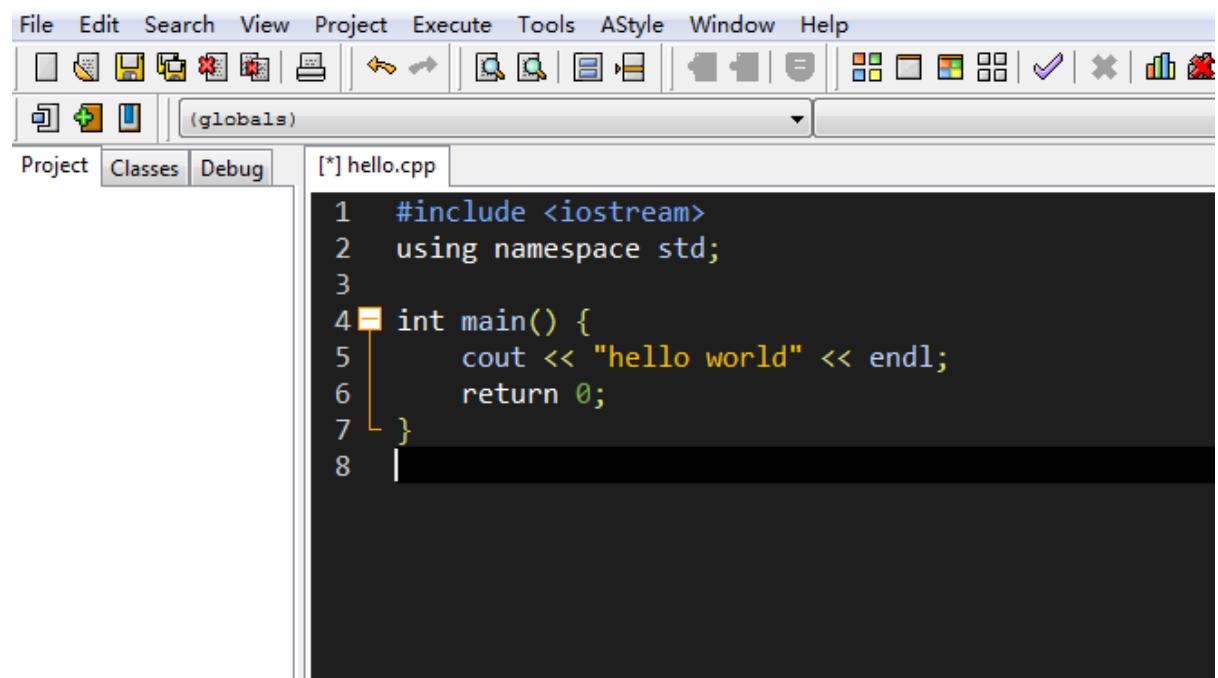
Play around branches

Branch is the way you can make change to your code without worrying about corrupting it. Next you will learn how to create, merge and discard a branch.

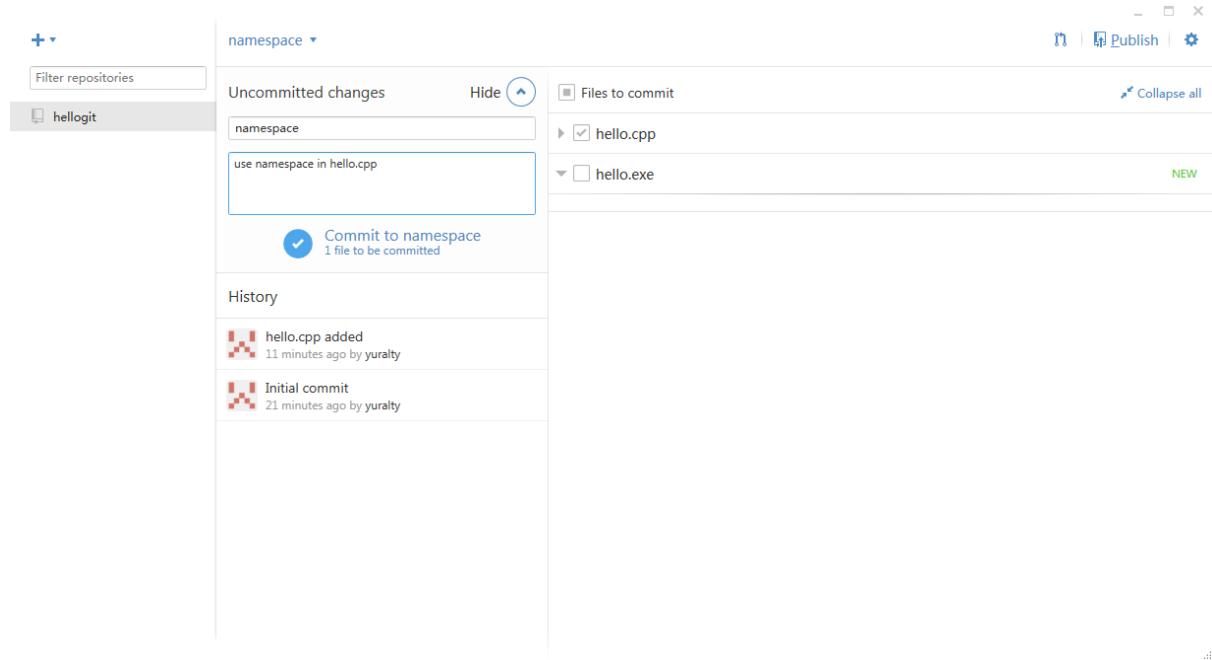
- Create a new branch from your client.



- Make some change in your code.



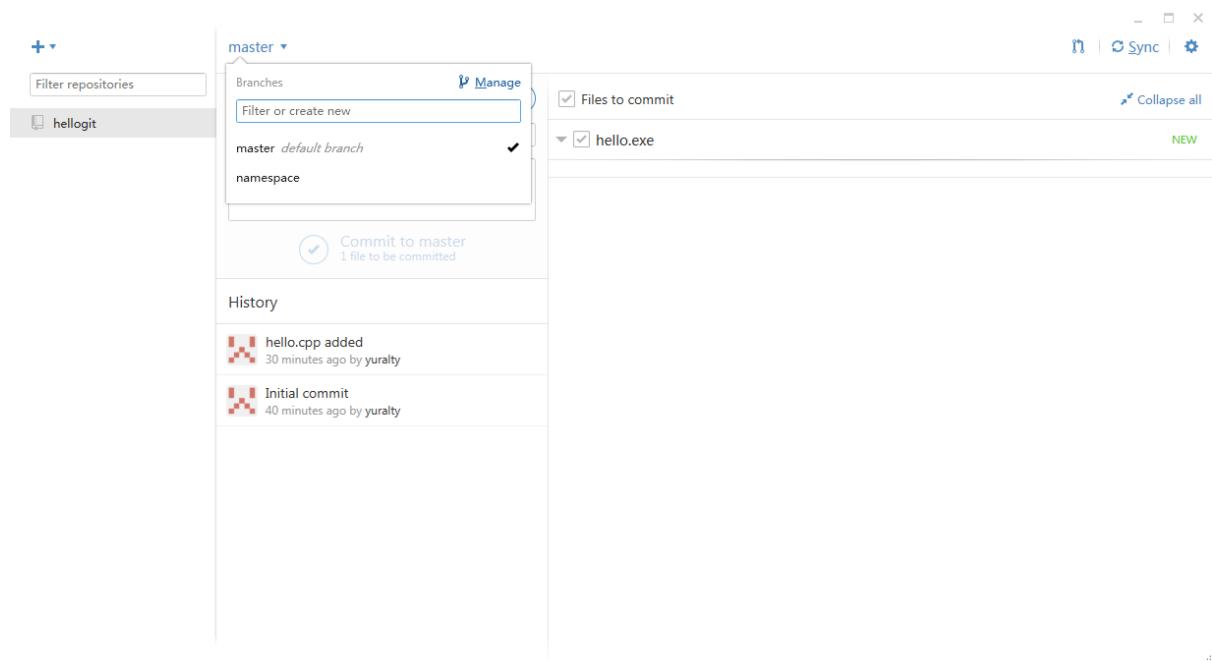
- Commit your changes and publish your branch to the remote repo.



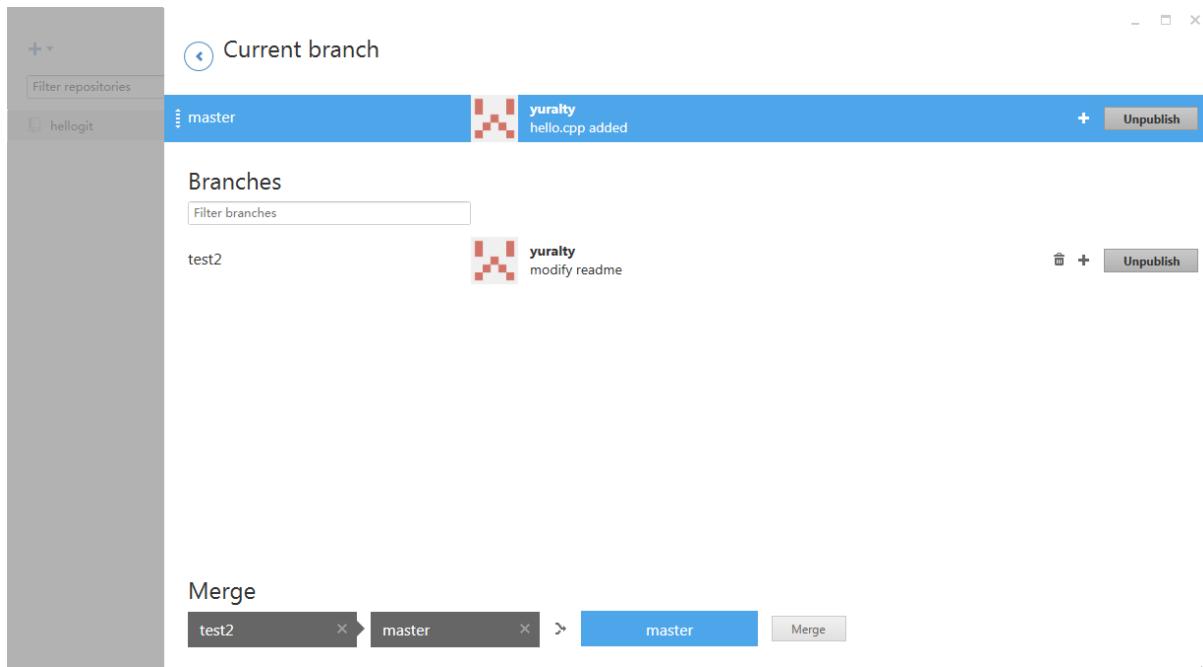
Now you can switch among branches and reopen your file to see the difference.

If you don't want this branch anymore

- Switch to your master branch.



- In “manage” tab, delete your new branch.



If you are satisfied with your changes, you can merge your new branch to your master branch at the same place.

Discuss issues

One feature of Github is to create issues on others' code. We will leverage this feature to do cross-review among team members. Here is an example of how to create, close and reopen issue.

The screenshot shows a GitHub repository page for 'yuralty / hellogit'. At the top, there's a navigation bar with a user icon, the repository name, a 'branch: master' dropdown, and buttons for 'Unwatch', 'Star', 'Fork', and 'Issues'. Below the navigation, a file named 'hello.cpp' is displayed. The file has 8 lines (5 sloc) and 0.09 kb. It contains the following C++ code:

```

1 #include <iostream>
2
3 int main() {
4     std::cout << "hello world" << std::endl;
5     return 0;
6 }
7

```

On the right side, there's a sidebar with various icons for repository management, including 'Issues' which is currently selected.

test issue #2

Edit New issue

Closed yuralty opened this issue 17 minutes ago · 1 comment



yuralty commented 17 minutes ago

Owner



test



yuralty commented 17 minutes ago

Owner



test reply



yuralty closed this 15 minutes ago



Write Preview

Markdown supported



Leave a comment

Attach images by dragging & dropping, [selecting them](#), or pasting from the clipboard.

Reopen issue

Comment

Labels

None yet

Milestone

No milestone

Assignee

No one—assign yourself

Notifications

Unsubscribe

You're receiving notifications because you modified the open/close state.

1 participant



Lock issue

Homework

1. Setup your Dev environment and Github account.
2. Write a hello world, and push it to your Github repo.
3. Select a team leader. The team leader should fill in an info list of your team members and send it to your TA.
4. Every one open a test issue in someone else's hello world repo.