

Session Information

Group Project 1: Fork the Syllabus

Group Project 2: Peer-Review

Group Project 3: GitHub Pages

Glossary

View the Presentation (<http://oudiglearn.github.io/olcinnovate-github/>)

Other Resources

Fork the Presentation (<https://github.com/oudiglearn/olcinnovate-github/tree/master/presentation>)

Suggest a Resource

PresentersAdam Croom, Director of Digital Learning, University of Oklahoma, @acroom (<https://twitter.com/acroom>)John Stewart, Asst Director of Digital Learning, University of Oklahoma, @jstew511 (<https://twitter.com/jstew511>)**Session Title**

FORK U! A GITHUB APPROACH TO LEARNING AND COLLABORATION

Abstract

You've heard the Git terminology ("fork," "repo," "pull") but you haven't dared to venture into the land of Github, the online network with 9 million users, more than 21.1 million repositories, and the largest host of source code in the world. While Github is a platform dominated by open source developers, it's worth understanding the use cases within higher education for courses, peer review, OER textbooks, syllabi, instructional design, and more.

We want to examine how this idea of individual ownership of one's work with a centralized hub, termed "federated" by Ward Cunningham, can alter student experience with texts and created a shared environment for faculty.

Since Fall 2014, the University of Oklahoma has been supporting learning through a Domain's of One's Own project, OU Create, which affords students with domains and open source powered web space. This has led to the creation of more than 2,000 webspaces built by students, faculty, and staff powered by tools like Wordpress and Omeka. As this primarily residential-based research institution has tip toed into the open web, the project has provided an opportunity for the OU Digital Learning team to explore platforms that allow for sharing of user-generated content amongst users (and beyond) with Github.

This workshop will provide a basic literacy of Github as well as a hands-on component where participants will work within Github to build out their first repository. These group-based activities will lay a foundation for participants to reflect on the affordances of open teaching and scholarship.

Group Project 1: Fork the Syllabus

Objective

For this project, we want you to get to experience what it's like to create a branch off of a central repository. You'll do this by editing information on a syllabus template.

Directions

Step 1: If you haven't yet, create a Github account and verify your email address.

Step 2: Go to <https://github.com/oudiglearn/syllabus> (<https://github.com/oudiglearn/syllabus>) and click on "Fork." This will create your own branch of the repository and name it `github.com/YOURUSERNAME/syllabus`.

syllabus template for OLC Innovate <http://oudiglearn.github.io/syllabus> — Edit

307 commits 3 branches 0 releases 6 contributors

Branch: gh-pages ▾ New pull request New file Upload files Find file HTTPS https://github.com/oudiglearn/syllabus

Step 3: Find and click on the file labeled `README.md`. Once you have opened `README.md` click the Edit icon.

Fork me on GitHub
(<https://github.com/oudiglearn>)

Branch: `gh-pages` / [syllabus](#) / [README.md](#)

adamcroom Update README.md 8d48d98 a day ago

5 contributors 

218 lines (97 sloc) | 8.44 KB

[Raw](#) [Blame](#) [History](#)  

Step 4: You'll find that the Syllabus is written in Markdown. Feel free to utilize this Markdown Cheatsheet (<https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet>) to decipher the codes for headers, lists, bold, etc.

Step 5: Begin to customize the syllabus to either a specific course that you've taught or a fictional course (or possibly appropriate one of those fancy massive open-licensed courses).

Step 6: Once you have made a significant amount of edits, COMMIT your changes.

Step 7: Once you've completed your changes, view them live on your repositories webpage. The address will be `YOURUSERNAME.github.io/syllabus`.

Bonus Materials

Do you feel like there we could be our syllabus template? Contribute to the centralized syllabus by starting a PULL REQUEST!

Step 1: Go back to the original repository for the syllabus: <https://github.com/oudiglearn/syllabus> (<https://github.com/oudiglearn/syllabus>). Make the changes to the `README.md` file that you think would help make a better template, and then instead of committing your changes, select Create a new branch for this commit and start a pull request and then click Propose file change.

Commit changes

Update `README.md`

I think units should be labeled "Units / Departments"

Commit directly to the `gh-pages` branch.

Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

 adamcroom-patch-1

[Propose file change](#) [Cancel](#)

Step 2: On the next page, you are able to had labels, milestones, or assign the pull request to a member of the original repository.

Open a pull request

The change you just made was written to a new branch named `adamcroom-patch-1`. Create a pull request below to propose these changes.

Group Project 2: Peer-Review

Objective

In this group project, you are going to use GitHub to provide peer review and editing for an article manuscript. The article manuscript is currently in the review phase for publication in *Digital Pedagogy in the Humanities: Concepts, Models, and Experiments* (<https://digitalpedagogy.commons.mla.org/>). This project describes itself as: >a curated collection of reusable and remixable pedagogical artifacts for humanities scholars in development by the Modern Language Association.

Kim Middleton's article "Remix" is an apt choice for this demonstration, both in terms of its content and its current phase of production. Still in draft form, Kim's article is awaiting peer review. We will put the article into peer review and provide line-by-line feedback. Feel free to provide commentary, substantive suggestions for the text, or copy-editing notes. The point of the demonstration is to write a few lines in markdown and submit your commentary through GitHub.

As you are working through this demonstration, please keep the instructions open in either a second window or next to you as a printout.

Instructions

Step 1: If you haven't yet, please create a GitHub account.

Step 2: In a new tab, follow this link (https://github.com/oudiglearn/Group2-peer_review/blob/master/keywords/remix.md) to open the article at Group2-peer_review/keywords/remix.md .

Step 3: Scroll through the file and look to see how the article is laid out. Notice that the meta-data for the article is presented at the top of the screen for this draft form and that the current status of the article is presented in the first section.

The screenshot shows a GitHub repository page for 'oudiglearn / Group-2--peer-review'. The repository has 2 stars, 0 forks, and 40 contributors. The file 'keywords/remix.md' is displayed, showing 184 lines (123 sloc) and 23.8 KB. The page includes a table of remix metadata and a 'REMX (Draft)' section featuring Kim Middleton's information. A red box highlights the 'Publication Status' section.

title	subtitle	chapter	URL	author	editor	publisher	tj								
Digital Pedagogy in the Humanities	Concepts, Models, and Experiments	Remix	keywords/remix.md	<table border="1"> <tr><th>family</th><th>given</th></tr> <tr><td>Middleton</td><td>Kim</td></tr> </table>	family	given	Middleton	Kim	<table border="1"> <tr><th>family</th><th>given</th></tr> <tr><td>Davis</td><td>Rebecca Frost</td></tr> </table>	family	given	Davis	Rebecca Frost	Modern Language Association	b
family	given														
Middleton	Kim														
family	given														
Davis	Rebecca Frost														

REMX (Draft)

Kim Middleton

Mount Saint Mary's University, Los Angeles | website?

Publication Status:

- unreviewed draft
- draft version undergoing editorial review
- draft version undergoing peer-to-peer review
- published

The article itself is divided into two sections: a “Curatorial Statement” about remixing and a “Curated Artifact” section which describes several examples of remixing.

Step 4: Click on the `edit` icon (a pencil in the gray menu bar) at the top of the screen

The screenshot shows a GitHub forked repository for 'oudiglearn / Group-2--peer-review'. The file 'remix.md' is open, displaying a table of bibliographic information. The 'Edit' button in the top right corner of the code editor is highlighted with a red box.

title	subtitle	chapter	URL	author	editor	publisher	tj								
Digital Pedagogy in the Humanities	Concepts, Models, and Experiments	Remix	keywords/remix.md	<table border="1"> <tr><th>family</th><th>given</th></tr> <tr><td>Middleton</td><td>Kim</td></tr> </table>	family	given	Middleton	Kim	<table border="1"> <tr><th>family</th><th>given</th></tr> <tr><td>Davis</td><td>Rebecca Frost</td></tr> </table>	family	given	Davis	Rebecca Frost	Modern Language Association	b
family	given														
Middleton	Kim														
family	given														
Davis	Rebecca Frost														

REMIX (Draft)

Kim Middleton

Mount Saint Mary's University, Los Angeles | website?

Publication Status:

- unreviewed draft
- draft version undergoing editorial review
- draft version undergoing peer-to-peer review
- published

Because you don't have administrative privileges for this file, when you clicked on the edit button, GitHub created a new "forked" copy of the repository in your account's repositories. When you get done with your edits, you will save your copy of the repository and suggest that we adopt those changes through a "pull-request."

Step 5: In the section entitled Publication Status, change the publication status to undergoing review by changing which option is **bold**. To bold something you use two asterisks * before and after the word. For a full cheat sheet of markdown coding, please follow this link **Markdown Cheat Sheet** (<https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet>)

Step 6: Add a couple of notes in the file suggesting changes. To do this add a paragraph wherever you want to make a suggestion using the format Yourname: your suggested revisions

Step 7: When you get done making your edits, scroll to the bottom of the screen to see the "Propose File Change" section. Add some information about your changes in the optional extended description text box. Then click the green Propose File Change button.

approaches to modify and reassemble materials to suit the tastes of their tribes, therein extending the definition of authorship for community engagement. Twentieth century technologies, in the meantime, evolved from analog to digital modes, making the foundational practices of remix--cut, paste, layer, juxtapose--available to extraordinary numbers of amateurs and artists. For all of the simplicity of those functions, however, the sudden accessibility of remix practices also encoded an array of complex implications belied by the "pop" face of remix. As Julie Levin Russo and Francesca Coppa explain in their analysis of fan video: "Technological innovations and Internet platforms support a developing ecology of [video] remix forms with unprecedented reach, richness, and cultural influence. At the same time, the value and legitimacy of this popular production is hotly contested on the basis of artistic merit, traditional literacies, and intellectual property" (1.2).

34

35

Simultaneously pervasive and controversial, remix emerged as a fundamental vernacular practice that requires defense. Law professor and activist Lawrence Lessig situates the right to remix as "a critical expression of creative freedom that . . . no free society should restrict" (56). He defends this bold claim by placing remix at the center of a culture in which citizens have the ability not only to consume cultural artifacts made by professionals, but to contribute to the culture by creating and re-creating what is around them. Like Lessig, Media Studies scholar Henry Jenkins identifies this phenomenon as "participatory culture" wherein individuals and affinity groups "archive, annotate, appropriate, and recirculate media content in powerful new ways." For Jenkins, these extant participatory practices

Propose file change

Update remix.md

Add an optional extended description...

Propose file change **Cancel**

Step 8: After you have clicked Propose File Change a new page will load showing a differential comparison between the old state of the "Remix" article and your modified version of the article. The older version of the file will be presented in red and the newer in green. After you have reviewed your changes, click on the green button labeled "Create Pull Request." This will submit the changed version of the "Remix" article back to our repository.

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks.

base fork: oudiglearn/Group-2-peer-review · base: master · head fork: gandalf511temp/Group-2-p... · compare: patch-1

Able to merge. These branches can be automatically merged.

Create pull request Discuss and review the changes in this comparison with others.

-> 1 commit 1 file changed 0 commit comments 1 contributor

Commits on Apr 15, 2016

gandalf511temp · Update remix.md ... e5f8c86

Showing 1 changed file with 1 addition and 1 deletion.

Unified Split

2	keywords/remix.md	@@ -180,4 +180,4 @@ Ridolfo, Jim and Danielle Nicole DeVoss. "Composing for Recomposition: Rhetori	View
180	180	Russo, Julia Levin and Francesca Coppa. "Fan/Remix Video (A Remix)." *Transformative Works and Cultures.* vol. 9 (2012). n. pag.	
181	181	Web. 22 Feb. 2016.	
182	182		
183	183	-Scott, Suzanne. "Teaching Transformativity/Transformative Teaching: Fair Use and the Video Essay." *Cinema Journal Teaching Dossier*, Vol 1:2 (Spring/Summer 2013). Web. 22 Feb. 2016.	
		+Scott, Suzanne. "Teaching Transformativity/Transformative Teaching: Fair Use and the Video Essay." *Cinema Journal Teaching Dossier*, Vol 1:2 (Spring/Summer 2013). Web. 22 Feb. 2016.	

No commit comments for this range

Group Project 3: Github Pages

One of the affordances of Github Pages is that it can host static HTML webpages. Here are a couple different ways to quickly get a website up and running.

Fork a website

Seriously. That's it. Fork another website. So let's fork this one called Clean Blog (<https://github.com/BlackrockDigital/startbootstrap-clean-blog-jekyll>).



Man must explore, and this is exploration at its greatest

Problems look mighty small from 150 miles up

Posted by Start Bootstrap on September 24, 2014

Jekyll

This template uses a CMS called Jekyll, which generates static HTML pages. This means that to blog we won't have to write actual HTML, instead we will write in a Github flavor of Markdown. If necessary, feel free to use this Markdown Cheatsheet (<https://github.com/adam-p/markdown-here/wiki/Markdown-Here-Cheatsheet>).

Is this thing already live?!

Yes!

Step 1: To find your web address, click on Settings.

Pulse Graphs **Settings**

Unwatch ▾ 1 Star

1 branch 0 releases 6

Step 2: Now scroll down to the Github Pages section to see where your site has been published.

GitHub Pages

✓ Your site is published at <http://adamcroom.github.io/startbootstrap-clean-blog-jekyll>.

Update your site
To update your site, push your HTML or **Jekyll** updates to your `gh-pages` branch. Read the [Pages help article](#) for more information.

Overwrite site
Replace your existing site by using our automatic page generator. Author your content in our Markdown editor, select a theme, then publish.

[Launch automatic page generator](#)

This is because Github automatically publishes repositories that have a special branch called “`gh-pages`.” If you go back to your repo home page, you’ll notice `gh-pages` on the branches dropdown.

adamcroom / startbootstrap-clean-blog-jekyll
forked from [BlackrockDigital/startbootstrap-clean-blog-jekyll](#)

[Code](#) [Pull requests 0](#) [Wiki](#) [Pulse](#) [Graphs](#)

A Jekyll version of the Clean Blog theme by Start Bootstrap [html](#)

Branch: **gh-pa...** ▾ [New pull request](#) [New file](#) [Upload files](#)

Switch branches/tags ×

Find or create a branch...
[Branches](#) [Tags](#)

gh-pages

l:gh-pages.
2015-06-10-hello-world.n
57 from midchidan/gh-p
ge author is not set

How can I write a post?

Step 1: Click on the folder in your repository labeled “`_posts`”.

A Jekyll version of the Clean Blog theme by Start Bootstrap <http://startbootstrap.com/template-overviews/clean-blog/> — Edit

This branch is 3 commits ahead of BlackrockDigital:gh-pages.

adamcroom Rename newpost.markdown to 2015-06-10-hello-world.markdown Latest commit 6e50a82 14 days ago

_includes Merge pull request #57 from midchidan/gh-pages 3 months ago

_layouts Default site title if page author is not set a year ago

_posts Rename newpost.markdown to 2015-06-10-hello-world.markdown 14 days ago

css Merge pull request #69 from f41c0r/gh-pages 3 months ago

fonts Bootstrap 3.3.2 Update a year ago

img Initial commit 2 years ago

..

Step 2: You'll find a set of posts already in there labeled in a format of YYYY-MM-DD-title-goes-here.markdown. Let's go ahead and edit one of them to see how this works. Click on the file label "2015-06-10-hello-world.markdown" and then click the pencil icon to edit the file.

This branch is 3 commits ahead of BlackrockDigital:gh-pages.

adamcroom Rename newpost.markdown to 2015-06-10-hello-world.markdown Latest commit 6e50a82 14 days ago

..

2014-06-10-dinosaurs.markdown added some alt tags a year ago

2014-07-01-dreams.markdown added some alt tags a year ago

2014-07-08-failure-is-not-an-option.mark... added some alt tags a year ago

2014-08-24-science-has-not-yet-mastere... added some alt tags a year ago

2014-09-18-i-believe.markdown added some alt tags a year ago

2014-09-24-man-must-explore.markdown add some alt tags a year ago

2015-06-10-hello-world.markdown Rename newpost.markdown to 2015-06-10-hello-world.markdown 14 days ago

Branch: gh-pa... / **startbootstrap-clean-blog-jekyll** / _posts / 2015-06-10-hello-world.markdown

adamcroom Rename newpost.markdown to 2015-06-10-hello-world.markdown 6e50a82 14 days ago

1 contributor

11 lines (9 sloc) | 250 Bytes

Raw **Blame** **History**

layout	title	subtitle	date	author	header-img
post	Hello world	because they lacked opposable thumbs and the brainpower to build a space program.	2015-06-10 05:00:00 -0700	Start Bootstrap	img/post-bg-01.jpg

Hello?

What you see here is some formatting data at the top that looks like this:

```
---
layout: post
title: "Hello world"
subtitle: "because they lacked opposable thumbs and the brainpower to build a space program."
date: 2015-06-10 12:00:00
author: "Start Bootstrap"
header-img: "img/post-bg-01.jpg"
---
```

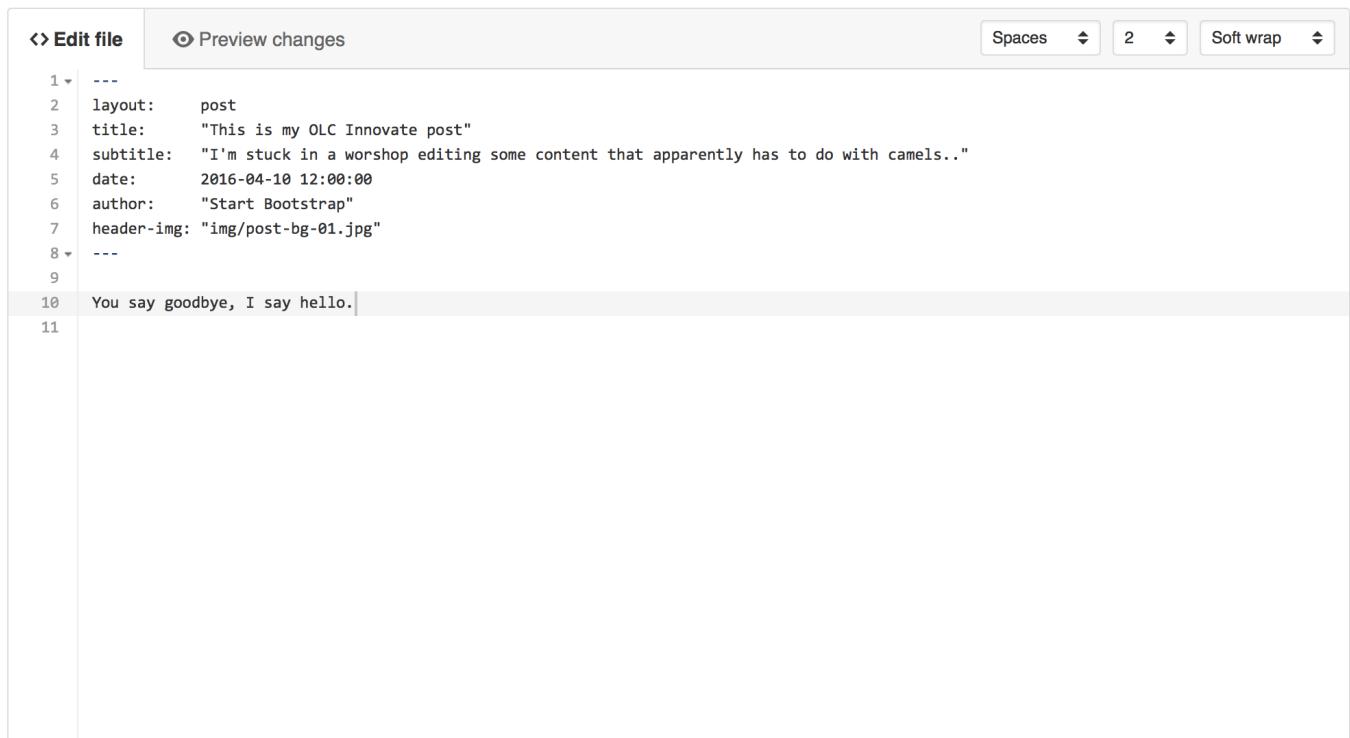
This is called YAML (yet another markup language). This theme has HTML templates that know to associate specific fields styling. For instance, it knows this is a post because the layout associated with it is "post." It also knows the title, subtitle, data, author, and header image.

Step 3: Play around with changing the title and subtitle as well as the date.

*Warning: do not change the layout or it will no longer know that it's a post!

Step 4: Similarly, add content your blog post by replacing "Hello?" located directly below the YAML content.

Step 5: Once you are done, you'll want to Commit your change.



```

<> Edit file << Preview changes
Spaces 2 Soft wrap
1 ---  

2 layout: post  

3 title: "This is my OLC Innovate post"  

4 subtitle: "I'm stuck in a workshop editing some content that apparently has to do with camels.."  

5 date: 2016-04-10 12:00:00  

6 author: "Start Bootstrap"  

7 header-img: "img/post-bg-01.jpg"  

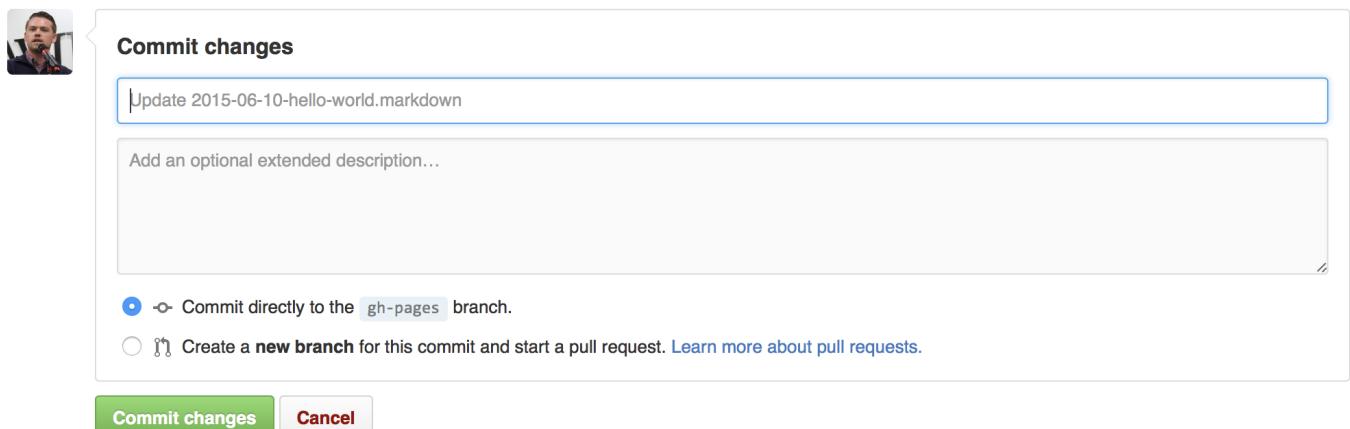
8 ---  

9  

10 You say goodbye, I say hello.  

11

```



Commit changes

Update 2015-06-10-hello-world.markdown

Add an optional extended description...

Commit directly to the `gh-pages` branch.
 Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes **Cancel**

Step 6: (Optional) Now go back to Settings recover that link and you'll see the changes take place (you may need to refresh the page).



This is my OLC Innovate post

I'm stuck in a workshop editing some content that apparently has to do with camels..

Posted by Start Bootstrap on April 10, 2016

You can, of course, create new blog posts too.

Step 1: Go back into your _posts folder and create a new file. Make sure you stick with the file format YYYY-MM-DD-title.markdown .

Step 2: Add the YAML at the top of the post:

```
---
```

```
layout: post
title:
subtitle:
date: YYYY-MM-DD 12:00:00
author:
header-img:
---
```

Hosting HTML

You may just be looking to host your site, and Github can simply host HTML as well. Here are some directions on how to do that.

Step 1: Create a new repository, or go to an existing one.

Step 2: In the repository overview, click the branch drop-down on the left-hand side. Type in gh-pages and press enter.