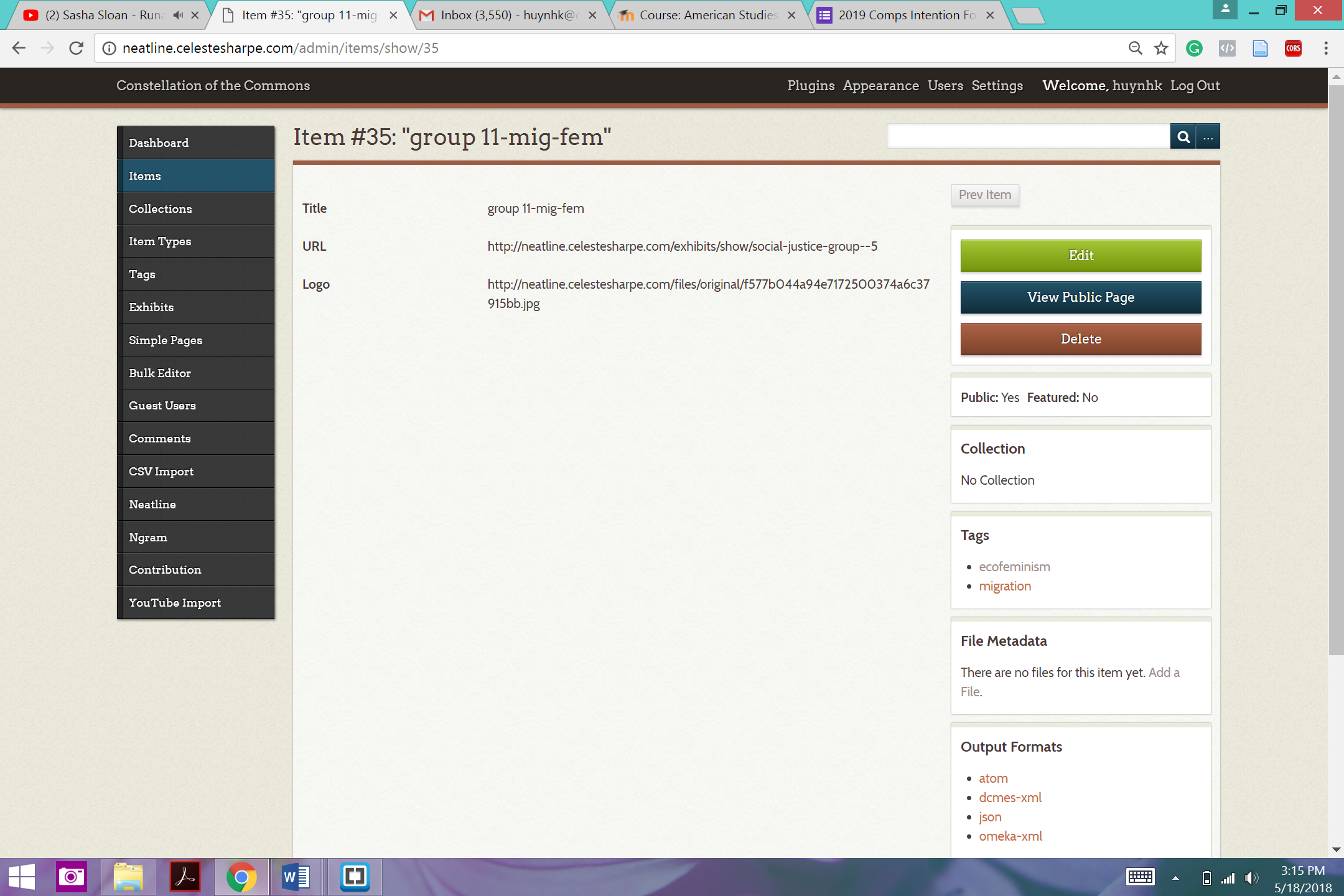
**Documentation for Constellation of the Commons Proto Site:**

**Using Omeka API and D3.js for Data-driven Visualizations on Omeka Sites**

**High-level:** We are organizing our data for themes and organizations on Omeka Site Manager in such a way that the Site Manager input-to-visualization workflow involving (Omeka Site Manager input 🡪 Omeka API 🡪 D3.js 🡪 homepage data-driven visualization of themes and orgs ) is a functional as well as an optimized automation.

**Omeka Site Manager and Omeka API**

Data should be organized within Omeka Site Manager as follows to be accessed efficiently by Omeka API:

* Each theme is represented by a tag, so add tags with theme names
* Each organization is represented on the Omeka site (‘front-end’) as a Collection. On the back-end, it is an Item of Organization Item type for D3 purposes. The Item for an org. will be tagged with all relevant theme tags, contain a link field that stores the corresponding Collection page URL, and contain a logo image link field that stores the corresponding logo image URL. And, the Item’s Title will be the org’s name. Note: the logo image must have a transparent background. Example of an Organization Item on Site Manager framework below.

**In the HTML-CSS-Javascript file (source code):**

**Using Omeka API to retrieve data to be used by D3, lines 67-107:**

2 API requests:

* Get theme names (represented by Omeka tags) by *get tags* query
* Get org names (represented by Omeka Organization Item) and corresponding logo URL, collection URL, and related tags per org by *get items / ?item\_type=18* query

Omeka API-retrieved data should be used to populate a growing database to be accessed correctly by D3.

Database comprised of:

* All node dicts are stored in a list named nodes.
* All link dicts are stored in a list named links.

High-level steps of script to put API data into database:

1. GET all tags, which represent themes, and put into list of themes
2. Create dict for each theme and add to nodes
3. GET all items of Organization Item Type and create dict for each org and add to nodes
   1. . For each org, add link between each of its themes (tags) and the org by id. Also, add themes to list as value for “themes” key.

Expected output:

* Each node is represented by a dictionary of key-value pairs
  + Theme nodes look like: [{“id”: *id #,* "name": "*theme name*", "level": "0", "themes": ["*theme name*"]}
  + Org nodes look like: {“id”: *id #,* "name": "*organization name*", "level": "1", "logo": "*logo URL*", "url": "*collection URL*", "themes": ["*related theme name*", ["*secondary related theme name if applicable”, etc.*]}
* Each link between nodes are represented by a dictionary storing key-value pairs for source node id and target node id.
  + Ex:. {"source": *element id #*, "target": *element id #,* “value”: *1*}

**Example of API-returned data in D3-ready format:**

var nodes = [{"id": 0, "name": "ecofeminism", "level": "0", "themes": "ecofeminism"},

{"id": 1, "name": "ecological education", "level": "0", "themes": "ecological education"},

{"id": 2, "name": "urban planning", "level": "0", "themes": "urban planning"},

{"id": 3, "name": "human rights", "level": "0", "themes": "human rights"},

{"id": 4, "name": "new media", "level": "0", "themes": "new media"},

{"id": 5, "name": "migration", "level": "0", "themes": "migration"},

{"id": 6, "name": "group 1-edu", "level": "1", "logo": "https://github.com/favicon.ico", "url": "http://neatline.celestesharpe.com/exhibits/show/social-justice-group--1","themes": ["ecological education"]},

{"id": 7, "name": "group 2-urb", "level": "1", "logo": "http://diylogodesigns.com/blog/wp-content/uploads/2016/06/Nasa-Logo-Transparent-Background-download.png", "url": "http://nasa.gov", "themes": ["urban planning"]},

{"id": 8, "name": "group 3-fem", "level": "1", "logo": "https://github.com/favicon.ico", "url": "http://neatline.celestesharpe.com/exhibits/show/social-justice-group--1", "themes": ["ecofeminism"]},

{"id": 9, "name": "group 4-med-fem", "level": "1", "logo": "http://www.masonbruce.com/wp-content/uploads/2015/03/android-logo-transparent-background.png", "url": "http://neatline.celestesharpe.com/exhibits/show/social-justice-group--1", "themes": ["new media", "ecofeminism"]},

{"id": 10, "name": "group 5-hum-urb", "level": "1", "logo": "http://neatline.celestesharpe.com/files/original/f577b044a94e7172500374a6c37915bb.jpg", "url": "http://neatline.celestesharpe.com/exhibits/show/social-justice-group--1", "themes": ["human rights", "urban planning"]},

{"id": 11, "name": "group 6-med", "level": "1", "logo": "http://assets.stickpng.com/thumbs/580b57fcd9996e24bc43c537.png", "url": "http://neatline.celestesharpe.com/exhibits/show/social-justice-group--1", "themes": ["new media"]},

{"id": 12, "name": "group 7-urb", "level": "1", "logo": "https://arcticportal.org/images/Logos/Arctic%20Portal/AP.png", "url": "http://neatline.celestesharpe.com/exhibits/show/social-justice-group--1", "themes": ["urban planning"]},

{"id": 13, "name": "group 8-edu-hum", "level": "1", "logo": "https://www.printful.com/static/images/layout/printful-logo.png", "url": "http://neatline.celestesharpe.com/exhibits/show/social-justice-group--1", "themes": ["ecological education", "human rights"]},

{"id": 14, "name": "group 9-edu", "level": "1", "logo": "http://www.pngpix.com/wp-content/uploads/2016/07/PNGPIX-COM-Pepsi-Logo-PNG-Transparent-500x667.png", "url": "http://neatline.celestesharpe.com/exhibits/show/social-justice-group--1", "themes": ["ecological education"]},

{"id": 15, "name": "group 10-fem-edu", "level": "1", "logo": "http://www.pngpix.com/wp-content/uploads/2016/07/PNGPIX-COM-Pepsi-Logo-PNG-Transparent-500x667.png", "url": "http://neatline.celestesharpe.com/exhibits/show/social-justice-group--1", "themes": "ecofeminism", "ecological education"]},

{"id": 16, "name": "group 11-mig-fem", "level": "1", "logo": "http://neatline.celestesharpe.com/files/original/f577b044a94e7172500374a6c37915bb.jpg", "url": "http://neatline.celestesharpe.com/exhibits/show/social-justice-group--1", "themes": "migration", "ecofeminism"]}]

var links = [{"source": 6, "target": 1, "value": 1},

{"source": 7, "target": 2, "value": 1},

{"source": 8, "target": 0, "value": 1},

{"source": 9, "target": 4, "value": 1},

{"source": 9, "target": 0, "value": 1},

{"source": 10, "target": 3, "value": 1},

{"source": 10, "target": 2, "value": 1},

{"source": 11, "target": 4, "value": 1},

{"source": 12, "target": 2, "value": 1},

{"source": 13, "target": 1, "value": 1},

{"source": 13, "target": 3, "value": 1},

{"source": 14, "target": 1, "value": 1},

{"source": 15, "target": 0, "value": 1},

{"source": 15, "target": 1, "value": 1},

{"source": 16, "target": 5, "value": 1},

{"source": 16, "target": 0, "value": 1}

]

**Displaying the data once its in the correct format, lines 34-54, lines 110-end:**

1. Initialize svg container and physics model/force relations for visualization.
2. Links are displayed as lines between source and target nodes.
3. Each theme node is displayed as circle objects colored by its theme color and captioned by its theme name; also classed as a theme in CSS.

* Color dependent on function circleColour(d): Node color is dependent on the *themes* key for a given node d. Currently, ecological education maps to green, human rights to blue, urban planning to purple, ecofeminism to red, new media to yellow, and migration to orange. Color mapping subject to change.
* Text caption dependent on the *name* key for a given node d.

1. Each org node is displayed as its logo image that holds a hyperlink to its Collection URL (wil redirect upon clicking); also classed as an org in CSS.

* Logo image is dependent on the *logo* key for a given node d.
* Hyperlinking is dependent on the *URL* key for a given node d.

1. Allow for user input (dragging nodes) to auto-update locations of nodes in visualization.

**Benefits of implementing Omeka API & D3.js-powered visualization over Neatline/other static visualizations**

What D3.js accomplishes through a physics/gravity-based graph model:

* Displays org. nodes as connected to and gravitating toward relevant theme nodes via lines
* Displays org logos for org nodes and theme name labels on circles for theme nodes, with each node colored by the color of its dominant theme
* Allows for easy linking between Omeka collections and exhibits URLs and corresponding node displays (so if a user double-clicks on a theme node circle labelled Ecofeminism, user should be redirected to Omeka ecofeminism exhibit).
* Allows for the process of drawing and sizing nodes in relation to one another to be automated/live-updated when new data is added to Omeka site, versus manual redrawing, resizing, and reuploading by a human on Omeka Neatline plugin every time a new piece of data is added 🡪 degree of scalability and streamlining

**Setting the source code as the Omeka Site homepage**

The source code file should be uploaded as the raw code for an Omeka *Simple Pag*e set as the Omeka site’s homepage.