Magnet Documentation

Release 1.0

Jon Robison

March 25, 2015

Contents:

Introduction

The magnet website is a project dedicated for effectively taking care of navigation, presentation, and hosting of magnet materials. It exists to provide a simple place through which a person may read through the documents, meeting magnet standards for websites.

1.1 Features

- A security model for restricting access

- Side navigation for two click access to any document

- Easily embedded pdfs, transparent to user or document maintainer

- A safe, backed up back end hosting materials through version control

1.2 Installation

If the magnet website is not already set up, there are a few ways to make it.

1.2.1 Openshift

1. Sign up for an account on openshift.com

2. From the openshift.com dashboard, create a new django 1.6 project

3. From 'settings', click 'add a new key...'

4. Copy/paste your ssh-keygen'd public key to provided space and click save

5. Note 'source code' URL from your application control panel

6. In root magnet project, git set-upstream that url and push

1.2.2 Roll your own

1. Use a django supported service like amazon ec2 vm or host in house

1.3 Support

Contact Jon Robison for questions or concerns about this product

Add Machine

Whether starting from scratch or allowing a new computer to upload documents to the website, follow these instructions.

2.1 Access Control

Magnet website includes security features to disallow the public from uploading new documents. This is done via SSH and git through the use of public/private keys. In order to upload new documents, a user will have to generate their own keys, then upload them to the git host (e.g. openshift) or get a currently existing user with access to upload their key for them.

2.2 Install Dependent Software

Before configuration, a user must install software.

2.2.1 Windows

1. Install mysgit and enable all command line tools

2.2.2 Unix-based

1. Install openssh and git from your package provider

2. e.g. if apt/debian: `apt-get install openssh git`

3. e.g. if yum/redhat: `yum install openssh git`

4. e.g. if mac: install hombrew, then `brew install openssh git`

2.3 Generate Keys

1. From windows, right click on desktop, click *Open git bash here*. From unix based system, simply open terminal

2. Type `ssh-keygen -t rsa` and press enter

3. Follow on screen instructions

2.4 Upload Keys

1. From your user's home directory, go in to the .ssh folder

2. Open the file .pub that you just made (e.g. *id_rsa.pub*) with vim or notepad

3. Copy the entire contents of the file

2.4.1 Openshift

1. Log in to your openshift account

2. From 'settings' click 'Add a new key...'

3. Copy entire contents of public key to text box and save

2.4.2 Roll your own server

1. Place your public key contents in the authorized_keys section on your server

Working with Documents

Most of the magnet document coordinators time will be spent working with documents. The magnet website workflow at a high level should look like:

1. Create and update documents

2. Place them all in static directory

3. Add, commit, and push documents via git

This results in the documents being uploaded and references properly. See below for more explanation on this process.

3.1 Create and Update Documents

Links to sources of evidence are created using hyperlinks. They should also pop up in a new tab, so the magnet reviewer may close the tab and resume where they left off.

1. Create sources of evidence and place in the "data" folder

2. Create documents as desired, suggest using latex or microsoft word

3. When linking sources of evidence, highlight the item and right click

4. Click Add/Edit hyperlink

5. Use the root url of your site, add "/static/", then add the path to your source of evidence e.g. www.example.com/static/Evidence/sourceEvidence1.pdf

To make links appear in a new tab, from the edit hyperlink window:

1. Click "Target window"

2. In the scroll box on the left, scroll all the way down to '_top' and select

3. Click 'Ok' to update the hyperlink

Note: Microsoft Word does not always output target window '_top' properly, but the requirement indicating the back button should return the user to the previous position in the document is satisfied because of a special pdf viewer, so this is no longer required.

3.2 Document Folder Structure

Documents must be placed in the correct directory structure and named correctly to show up. The top level folders must be ordered by the same order as navigation.txt, located in the static folder (<project path>/data). There should be no spaces, but otherwise capitalized how you want them to display in the navigation bar. The first line shows the top item, second line second item, etc.

Within each top level folder, a folder for the item name should be in all caps. Inside this folder, a pdf with the same name should be placed. Optionally, you may include sources of evidence, which show up as a list in the top, but it is suggested to avoid this and use the alternative static mechanism given above.

Examples:

- NK4EO location: <project path>/data/NewKnowledge/NK4EO/nk4eo.pdf
- NK4EO source of evidence: <project path>/data/sources/nk4eoSource1.pdf
- NK4EO link (in pdf): http://www.example.com/static/sources/nk4eoSource1.pdf

3.3 Upload to Website

After the documents are updated they must first be placed in the correct directory structure, described above. After they are all placed correctly, commit and push with git. You may either use mysgit in windows are terminal in windows/unix based system.

3.3.1 Mysgit

1. Right click on static folder and open mysgit ui
2. Click 'Stage Changed'
3. Add meaningful commit message, e.g. "-Adding NK 2-4, updating OO3"
4. Click 'Commit' button
5. Click 'Push' button

3.3.2 Terminal

1. From windows, right click on static directory, and click 'Git bash here'. From unix terminal, navigate to project's static directory
2. Type *git add .*
3. Type *git commit -m "message"*, where message is a meaningful message for what you are doing, e.g. "-Adding NK 2-4, updating OO3"
4. Type *git push origin* to send all the updated docs to server. This will restart the server and take some time. Should end with 'success' message(s).

3.4 Common Suggestions/Pitfalls

- If working from multiple computers, when anyone updates, all other computers must *git pull* before pushing
- Documents can go in subfolders within static for convenience, e.g. <project path>/data/sources

- Only run commands in Upload to Website when all documents, banners, sources, and text documents are updated to save time. It will pick up all changes.

- Use all lowercase names for pdfs. If inconsistent casing, duplicates may be introduced and upon testing the website confusing older documents may show up.

- When referencing, the full URL should never contain consecutive slashes '/'

- Stay away from using spaces in filenames e.g. "Source of Evidence.pdf", rather rename e.g. "SourceOfEvidence.pdf". If you see %20 and errors occurring, this is likely why.

Site Customization

4.1 Theme

The magnet website is flexible in choosing any bootstrap compatible theme. Many free themes are housed on bootswatch. Navigating to bootswatch, the user can see many themes scrolling down. Upon finding a theme of interest, at the top navigation bar note the Themes pull down menu, which allows the user to pick a theme for a more in depth look.

If a new theme is chosen, after selecting the theme from the "Themes" drop down menu, select the "Download" menu and download "bootstrap.min.css" and save it to the magnet folders `data/cssjs` folder.

4.2 Templates

To add or edit the stock text of top level pages (e.g. Transformational Leadership), navigate to data/top and note any `.html` files there. If a html file exists with a name corresponding to a major magnet section, upon clicking that section in the website navigation, the html page will be retrieved and displayed. If you want to add or edit the appearance of the page upon clicking, one must add or edit the corresponding .html file.

An example file has been place in the project named `ExemplaryProfessionalPractice.html`. These files are plain html, so any questions can be satisfied by w3schools or similar queries. However, some common operations are below.

If regular paragraphs should be added, or the text changed, then after pasting the text in, each paragraph should start with `<p>` and end with `</p>`. For a five paragraph page, there should be five sets of p elements.

If an image should be added, e.g. a banner at the top, then an `img` element should be used. SVGs are preferred here, since images are responsive in nature and would thus maintain clarity at any screen resolution, but alternatively `.png` images work (as do `.jpg`). Place the image in the data/top folder, then reference in an `img` element like:

```
<img src="/static/top/exampleBanner.svg" class="img-responsive">
```

4.3 Main Banner Image

By default, the index page banner is included as 'bannerIndex.png'. If you wish to update this, simply copy your cropped image over 'bannerIndex.png' and run **'Upload to Website'_** commands after everything is updated. The image must be a .png.

Note: The about page also may have a banner, named "bannerAbout.png" in the same folder

---

4.4 Glossary

The glossary is defined in the `glossary.txt` document in the static folder. The definitions should be pipe delimited '|', and any number of definitions may be given. To update glossary, edit `glossary.txt` in your favorite editor (e.g. vim, notepad) and append or edit definitions, one per line, as desired.

Authorization

By default, there is one admin account and one read only account. This is so the website administrator may manage who is allowed to view magnet documents, while a shared read only user can be shared to magnet reviewers.

5.1 Defaults

The default username is 'magnet' and the default password is `AEwgEVfJM7spUsgt` The default administrator is 'administrator' and the default password is `vemaq9JDe2qKPQk`

5.2 Changing a Password

1. Go to the website as normal, but add /admin to the end of the url
2. Log in current administrator credentials
3. Click 'Users'
4. Click on the user whose password you wish to modify
5. In the password section, bottom right, click 'this form'
6. If you can't find 'this form', add 'password' to url and press enter
7. Type in new password as desired and click ok to save user

5.3 Adding a User

1. Go to the website as normal, but add /admin to the end of the url
2. Log in current administrator credentials
3. Beside 'Users' click 'Add'
4. Enter desired Username / Password credentials
5. Click 'Save' in bottom right

5.4 Promoting User to Admin

1. Go to the website as normal, but add /admin to the end of the url

2. Log in current administrator credentials

3. Click 'Users'

4. Click on the user you wish to modify

5. Scroll down to 'User permissions'

6. Click and drag to select all available user permissions

7. Click right right arrow (which moves all selected to chosen permissions)

8. Click 'Save' in bottom right

Create USB stick

To create a USB stick or autorun-compliant cd/dvd, please take not of the 'usb' directory and its contents. This is windows only, though non windows users can navigate to the magnet/data directory to view documents offline. Any removable media should be organized as such:

1. Copy and paste the files in the 'usb' folder to the root of the drive

2. Download and extract/install "portable python" to a `pp` directory in the root of the usb drive

3. Place the built magnet documents in a folder named "magnet" in the root of the drive

4. Copy all the files inside the "data" folder to "wsgi/static"

When this is done, the removable media should now be ready. To test, insert the media and run the `autorun.bat` executable. A console window ending with something like "waiting for requests" should remain, at which point a browser tab should appear bringing the user to the site.

Indices and tables

- *genindex*
- *modindex*
- *search*