
FOSSRIT Runbook

Release 1.0.0

Justin W. Flory

Feb 20, 2020

1	Responsibilities and contacts	3
1.1	FOSS@RIT	3
1.2	FOSS@MAGIC	4
2	Ansible	5
2.1	Contact information	5
2.2	Background	5
2.3	Overview	5
2.4	git repo	6
2.5	Directory setup	6
2.6	Additional resources	7
3	Chat bridges	9
3.1	Matterbridge	9
3.2	TeleIRC	10
4	DigitalOcean	11
4.1	Team	11
4.2	Droplets	12
5	Discourse (fossrit.community)	15
5.1	Contact information	15
5.2	Hosts	15
5.3	Deploying	15
5.4	Configuration	16
5.5	Upgrading	16
5.6	Backups	16
6	IRC (Freenode)	19
6.1	Project registration	19
6.2	Cloaks	19
7	Matrix	21
7.1	How we use Matrix	21
7.2	RIT community	21
7.3	Known rooms	22

8	Telegram	23
8.1	Roles	23
8.2	List of known groups	26
9	Website (fossrit.github.io)	27
9.1	About	27
9.2	Create a development environment	28
9.3	How to update FOSS Hours time/place	29
9.4	How to add content	29
10	Indices and tables	33

This is the main page for the FOSSRIT Runbook. Other pages are available for you to navigate and explore below. For more information, see the [repo on GitHub](#).

Responsibilities and contacts

This page documents administrative responsibility and contact information for the FOSS@RIT community. Generally, there are two separate entities: **FOSS@RIT** and **FOSS@MAGIC**.

1.1 FOSS@RIT

FOSS@RIT usually means the wider free/open source community at RIT. This includes organizations and groups such as the following:

- Free/Open Source Software initiative at RIT MAGIC Center
- RIT Linux Users Group
- Computer Science House
- Sustainability at RIT

However, the FOSS@RIT identity is not exclusive. It is not “owned” by any single group or organization. Generally, anyone who works in free/open communities at RIT, not limited to software, is included in this identity.

1.1.1 Responsibilities

Responsibilities associated with the FOSS@RIT community are typically on a volunteer basis and informal. This list is not exhaustive and will change over time, as older folks move on and new folks get involved.

- Moderating *IRC channels* on Freenode
- Moderating Matrix/Riot chat rooms
- Moderating *Telegram groups*
- Moderating *fossrit.community Discourse* discussions
- Maintaining open source projects on FOSSRIT GitHub organization

1.1.2 Community contacts

The table below documents who the best person to contact is for different parts of the FOSS@RIT community:

1.2 FOSS@MAGIC

FOSS@MAGIC is the Free/Open Source Software initiative at the RIT Center for Media, Arts, Games, Interaction, and Creativity (MAGIC). More details about FOSS@MAGIC are available on the [official website](#). Anything associated with FOSS@MAGIC is directly connected to RIT and the MAGIC Center.

1.2.1 Responsibilities

Responsibilities associated with FOSS@MAGIC are typically related to accounting or event organization. This list is not exhaustive, but should remain fairly consistent as most of these responsibilities are associated to paid faculty and staff of RIT:

- Accounting
 - Domain names (e.g. `fossrit.community`, `teleirc.com`)
 - Infrastructure services (e.g. DigitalOcean)
- Event organizing
 - Booking rooms/space for meetings (e.g. FOSS Hours, TeleIRC developer meetings)
 - Ordering food/beverages

1.2.2 MAGIC contacts

The best points of contact for FOSS@MAGIC are [Prof. Stephen Jacobs](#) and [Brenda Schlageter](#). Jacobs is the formal head of the FOSS@MAGIC initiative, while Schlageter is the Operations Manager of the MAGIC Center. The best way to get in touch with FOSS@MAGIC staff is via email.

Send an email to the below address to get in contact (obfuscated to reduce spam from web-scrapers bots):

```
foss [at] rit [dot] edu
```


This page documents the Ansible configuration management repository at [FOSSRIT/infrastructure](#). Any changes to services or applications managed on FOSS@MAGIC infrastructure are managed in this git repo.

2.1 Contact information

- **Owner:** FOSS@MAGIC Tech Team
- **Contact:** [#rit-foss-admin](#) / [#rit-foss-admin:matrix.org](#) / [telegram.me/fossrit_infra](#)
- **Persons:** [Justin W. Flory](#), [Tim Zabel](#)
- **Location:** Toronto, Canada (TOR1)
- **Purpose:** Configuration management for FOSS@MAGIC infrastructure.

2.2 Background

The FOSS@MAGIC Tech Team uses Ansible for system change management. Previously, no configuration management tool was used for our infrastructure. Some git commits go back to 2013 for a few scripts used back in the day, but the migration to Ansible started in September 2018.

2.3 Overview

Ansible playbooks are run locally by a system administrator. A system administrator can run Ansible playbooks once a user and SSH key is set up for them on the host. New SSH keys and users are defined in the `sshd` role (see [Roles](#)).

Playbooks are idempotent (or should be). Meaning you should be able to re-run the same playbook over and over and it should get to a state where 0 items are changing.

2.4 git repo

There is one repository associated with Ansible, [FOSSRIT/infrastructure](#). This is a public repository. Private data is stored via encryption with [Ansible Vault](#). The encryption passphrase is shared on a need-to-know basis with infrastructure maintainers.

It is recommended to configure `git` to use `pull --rebase` by default. Do this with the following command in your local clone of the infrastructure git repo:

```
git config --bool pull.rebase true
```

This prevents unneeded merges which can occur if someone else pushes changes to the repository while you are working on your own local changes.

2.5 Directory setup

2.5.1 Inventory

The `inventory/` directory tells Ansible what hosts are available and any groups they belong to. All files in this directory are concatenated together, so you can split out groups/hosts into separate files for readability. They are in `ini` file format.

Additionally, under the inventory directory are `host_vars/` and `group_vars/` subdirectories. These are files named for the host or group and containing variables to set for that host or group. You should strive to set inventory variables in the highest level possible, and precedence is in:

- Global
- Group
- Host order.

2.5.2 Vars

This directory contains global variables as well as OS specific variables. Note that in order to use OS-specific variables, you must have `gather_facts` as `True` or Ansible will not be able to determine the OS.

2.5.3 Roles

Roles are a collection of tasks/files/templates that can be used on any host or group of hosts that all share that role. In other words, roles should be used except in cases where configuration only applies to a single host. Roles can be reused between hosts and groups and are more portable/flexible than tasks or specific plays.

2.5.4 Scripts

The `scripts/` directory includes miscellaneous utility scripts for sysadmins.

2.5.5 Playbooks

There is a `playbooks/` directory where YAML playbooks are stored. The top level contains utility playbooks for `sysadmins`. These playbooks perform one-off functions or gather information. Under this directory are `hosts/` and `groups/` playbooks. These playbooks are for specific hosts and groups of hosts, from provision to fully configured. You should only use a host playbook in cases where there will never be more than one of that thing.

2.6 Additional resources

- [Upstream docs](#)
- [Examples:](#)
 - github.com/ansible/ansible-examples
 - gist.github.com/marktheunissen/2979474
- [Jinja docs](#)

This page documents the bridge services maintained by the [FOSS@RIT Tech Team](#). Chat bridges are used to “bridge” different chat rooms together (e.g. a Telegram group to an IRC channel).

3.1 Matterbridge

[Matterbridge](#) is an open source chat bridge that supports various chat platforms. Currently, FOSS@RIT uses it to connect the [RITlug Slack](#) to IRC channels on Freenode (see [IRC \(Freenode\)](#)).

3.1.1 Maintenance

The FOSS@RIT Matterbridge instance is deployed on [RITlug](#) infrastructure hosted in RIT’s Institute Hall data center. Configuration and deployment details are managed in [FOSSRIT/infrastructure](#) as an [Ansible Role](#). The upstream Ansible Role is maintained at [jwflory/ansible-role-matterbridge](#).

3.1.2 Upgrading

Upgrading Matterbridge requires changing two variables in the Ansible Role:

1. Version number (`matterbridge_config.version`)
2. Binary checksum (`matterbridge_config.binary_checksum`)

Both the version and binary checksum are found on Matterbridge’s [releases page](#). You can find the binary checksum in the `checksums.txt` file, attached as an artifact to Matterbridge releases. We use the hash for the `linux-64bit` compiled binary.

See the following pull requests for examples:

- [FOSSRIT/infrastructure#59](#)
- [FOSSRIT/infrastructure#66](#)

- [FOSSRIT/infrastructure#76](#)

3.2 TeleIRC

TeleIRC is an open source chat bridge for connecting Telegram groups to IRC channels. It is maintained by the RIT Linux Users Group. Currently, FOSS@RIT uses it to connect various Telegram groups to IRC channels on Freenode (see [Telegram](#), [IRC \(Freenode\)](#)).

3.2.1 Maintenance

All FOSS@RIT TeleIRC instances are deployed on [RITlug](#) infrastructure hosted in RIT's Institute Hall data center. Configuration and deployment details are managed in [FOSSRIT/infrastructure](#) as an [Ansible Role](#). The upstream Ansible Role is maintained at [jwflory/ansible-role-teleirc](#).

3.2.2 Upgrading

There are two types of upgrades: TeleIRC and the Ansible Role.

TeleIRC

Only one variable in the Ansible Role needs to be changed in most cases: `default_version`. Change this variable to the targeted version and run the Ansible playbook to deploy the update.

See the following pull requests for examples:

- [FOSSRIT/infrastructure#55](#)
- [FOSSRIT/infrastructure#63](#)

Ansible Role

Occasionally the upstream Ansible Role will change. When this happens, the Ansible Role needs to be upgraded in the downstream repository, i.e. [FOSSRIT/infrastructure](#). The upgrade can be done from the root directory of the repo with the following commands:

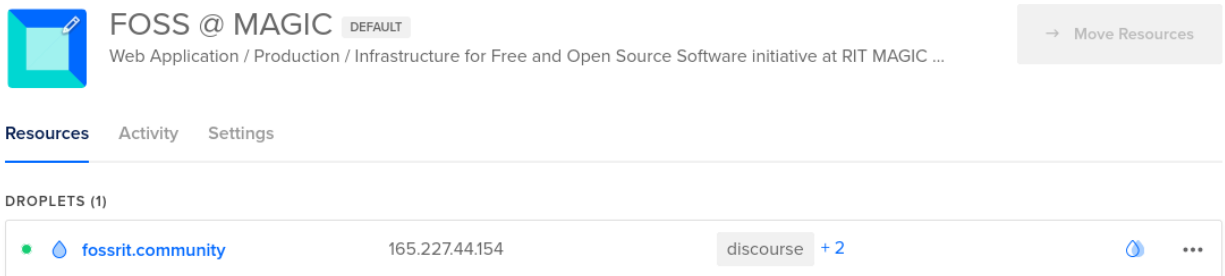
```
ansible-galaxy role install --force jwflory.teleirc
git checkout HEAD roles/jwflory.teleirc/vars/{main,vault}.yml
```

An example of an upgrade is found in [FOSSRIT/infrastructure@75c0bbc](#).

This page documents the DigitalOcean team and infrastructure used for FOSS@MAGIC.

4.1 Team

There is a *FOSS @ RIT MAGIC* team registered in DigitalOcean's admin interface. The DigitalOcean team is a shared group of accounts with direct access to the DigitalOcean Droplets:



Screenshot

of default DigitalOcean project from admin interface, shared among members of the FOSS @ RIT MAGIC team




4.1.1 Access

Currently, three people have access to the shared *FOSS @ RIT MAGIC* DigitalOcean team:

- Brenda Schlageter
- Justin W. Flory
- Stephen Jacobs

3 Members

Invite Members

Name	Role	Status	Sign-in method	
 Justin W. Flory (you)	Owner	Joined	Email + 2FA	...
 Brenda Schlageter	Owner	Joined	Email	...
 Stephen Jacobs	Member	Pending	—	...

Screenshot

of FOSS @ RIT MAGIC team members from DigitalOcean admin interface

4.2 Droplets

Droplets are the billable “unit” in DigitalOcean. They represent a specific service or server. FOSS@MAGIC covers the expenses for the Droplets on a monthly basis. This is set up with automatic billing to the RIT MAGIC Center via an RIT-managed credit card.

4.2.1 Services

Currently, only one service is hosted as a DigitalOcean Droplet:

- [Discourse](#)

The Discourse Droplet is configured with automatic snapshot backups, taken once a week:

Backups

Backups are currently enabled. They are scheduled to start weekly Friday 3 PM to Saturday 2 PM (EST).

Backups for this Droplet cost \$2.00 per month.

[Disable Backups](#)

Droplet backups



fossrit.community 2020-01-25

Backup created 1 month ago

[More](#) ▾



fossrit.community 2020-02-01

Backup created 19 days ago

[More](#) ▾



fossrit.community 2020-02-08

Backup created 12 days ago

[More](#) ▾



fossrit.community 2020-02-15

Backup created 5 days ago

[More](#) ▾

Screenshot

of Discourse Droplet backup configuration and history of recent backups in DigitalOcean admin interface

Discourse (fossrit.community)

This page documents the Discourse service used for our community forums at fossrit.community. Discourse is where anyone may share topics for discussion with the wider FOSS@RIT community.

5.1 Contact information

- **Owner:** FOSS@MAGIC Tech Team
- **Contact:** [#rit-foss-admin](#) / [#rit-foss-admin:matrix.org](#) / [telegram.me/fossrit_infra](https://t.me/fossrit_infra)
- **Persons:** [Justin W. Flory](#)
- **Location:** Toronto, Canada (TOR1)
- **Servers:** `fossrit.community`
- **Purpose:** Discussion forums for FOSS@RIT community.

5.2 Hosts

The current deployment is made up of a single DigitalOcean droplet, `fossrit.community`. This host runs:

- Docker
- [Discourse Docker image](#)

This host relies on:

- SendGrid API key (for outgoing mail, eventually incoming)

5.3 Deploying

Our Discourse site was deployed exactly as described in upstream's documentation:

- [Install guide](#)
- [Email setup guide](#)

5.4 Configuration

We do not use config files to maintain Discourse. The config settings are dynamic and stored inside the image container. If a rollback is needed, see *Backups*.

5.5 Upgrading

Excerpt from meta.discourse.org.

If you self-host Discourse, you occasionally need to run a manual update via the command line to get the latest security releases newest libraries. These updates are not picked up in admin/upgrade, which is why you'll occasionally need to do this additional step.

- Discourse itself should be updated about twice a month, by clicking the “Update to Latest Version” button in your admin dashboard (admin/upgrade). We do beta releases roughly once every week.
- Every two months we recommend SSH'ing into your web server and doing

```
cd /var/discourse
git pull
./launcher rebuild app
```

- As for Ubuntu updates, make sure you have automatic security updates enabled for your Ubuntu! The command is:

```
dpkg-reconfigure -plow unattended-upgrades
```

However, that just covers critical security updates. Every so often you should get all the OS updates like so:

```
apt-get update
apt-get dist-upgrade
```

To summarize:

- update Discourse twice a month via web updater
- update the container every two months
- update the OS every six months

You could double these numbers and still be fairly safe, e.g. update Discourse once a month, container every 4 months, OS once every 12 months, and so on.

But you really, really want automatic security updates enabled in Ubuntu, as listed above – all our Digital Ocean “one-time installs” already have automatic security updates set up.

5.6 Backups

Two types of backups are made of the Discourse site:

5.6.1 Site-level backups

The Discourse site takes a weekly backup of site content, usually between 00:00–01:00 on Thursdays. These backups are available from the admin panel:

`fossrit.community/admin/backups`

These backups can be used for rollbacks, triggered by admin users.

5.6.2 Droplet-level snapshots

Snapshots are currently enabled via DigitalOcean (\$2/month). They are scheduled to start weekly Friday 3 PM to Saturday 2 PM (EST). Snapshots can be accessed via the DigitalOcean droplet settings.

This page documents the [Internet Relay Chat \(IRC\)](#) infrastructure for the FOSS@RIT community. Currently, this only includes our presence on the [Freenode](#) IRC network.

6.1 Project registration

TODO...

6.2 Cloaks

Cloaks are a type of “host mask” that replaces your IP address in IRC clients with a custom string of text. The IRC protocol exposes your IP address or hostname by default when you join a channel. When you authenticate with IRC’s NickServ, a cloak replaces your IP address/hostname with the custom string.

Note that Freenode does not support the underscores (`_`) in cloaks. Other non-alphanumeric characters may also not be supported and will be omitted from the cloak.

6.2.1 Request a cloak

In order to receive a cloak, you must have completed the following steps:

1. Register your nick with NickServ
2. Set an email address with NickServ
3. Created an alternate nick and linked it to primary nick with NickServ

If you have not completed these steps, Freenode will not allow us to create your cloak. For help doing these things, check out the [Nickname Registration help page](#) on Freenode’s website.

Once you have completed the above steps, you can request a new cloak. Request a cloak by [opening a new issue](#) on the [FOSSRIT/tasks](#) repository.

6.2.2 Available cloaks

Members of the RIT community may choose from the following cloaks:

General

- @rit/student/<your IRC nick>
- @rit/professor/<your IRC nick>
- @rit/faculty/<your IRC nick>
- @rit/alumnus/<your IRC nick>
- @rit/alumna/<your IRC nick>
- @rit/alum/<your IRC nick>

RIT FOSS

Cloaks for the [RIT FOSS community](#).

- @rit/foss/student/<your IRC nick>
- @rit/foss/faculty/<your IRC nick>
- @rit/foss/alumnus/<your IRC nick>
- @rit/foss/alumna/<your IRC nick>
- @rit/foss/alum/<your IRC nick>

RITlug

Cloaks for the [RIT Linux Users Group](#).

- @rit/ritlug/member/<your IRC nick>
- @rit/ritlug/alumnus/<your IRC nick>
- @rit/ritlug/alumna/<your IRC nick>
- @rit/ritlug/alum/<your IRC nick>

6.2.3 Granted cloaks

The table below is a list of community members who were issued IRC cloaks.

This page documents the use of the [Matrix.org network](#) and [Riot chat client](#) in the FOSS@RIT community. Matrix is a decentralized chat network while Riot is one (of many) possible clients for participating in the chat network.

7.1 How we use Matrix

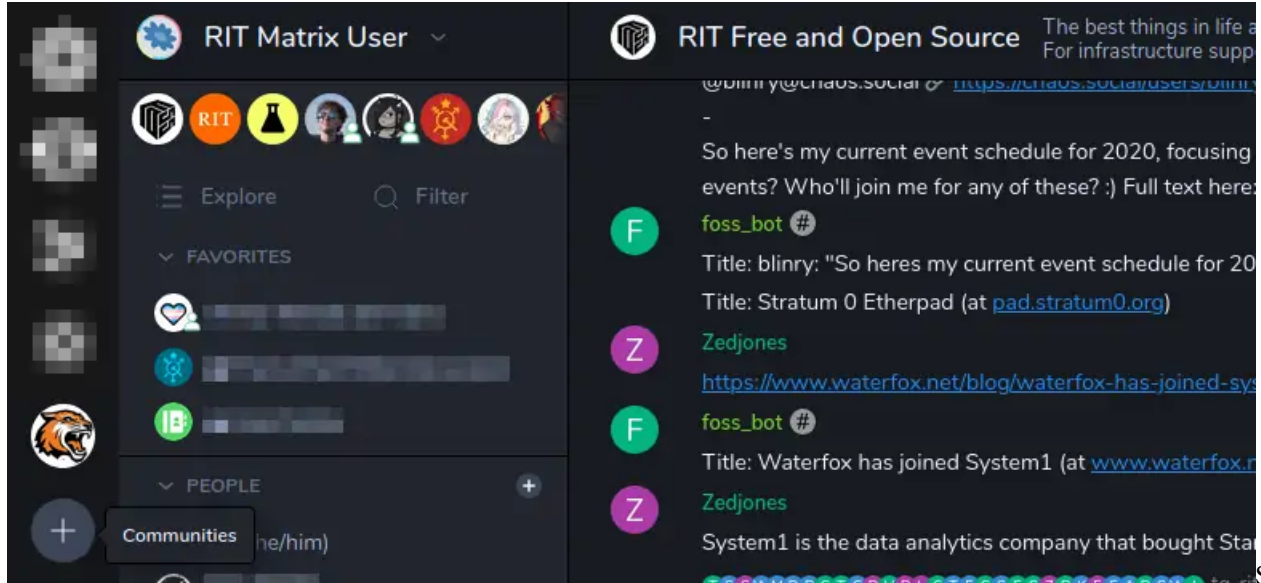
Matrix is convenient for us because it offers compatibility with existing IRC networks, such as Freenode. Since the FOSS@RIT community was established on Freenode in 2010, we have a historical presence on IRC (see [IRC \(Freenode\)](#)). Matrix/Riot provides a modern user interface in a chat client and integrates nicely into existing IRC channels. Many FOSS@RIT community members use the Matrix network, but can communicate seamlessly with IRC channels.

For help using Matrix/Riot, refer to the following links:

- [IRC for the 21st Century: Introducing Riot](#)
- [How do I register/identify to NickServ?](#)

7.2 RIT community

In 2017, [Riot.im announced communities](#), a feature that groups rooms and users together (similar to a Slack team or a Discord server). (Note eventually this will be replaced by [Groups-as-Rooms](#).) There is a RIT community on the [matrix.org homeserver](#): `+rit:matrix.org` ([view on Riot](#)).



Screenshot

of Communities button in Riot.im web client

The RIT community is owned by @jflory7:matrix.org. At the time of writing, it is not possible to add other admins to communities.

7.3 Known rooms

The following is a list of all known Matrix rooms related to RIT or the FOSS@RIT community.

This page documents the Telegram chat groups in the wider FOSS@RIT community.

8.1 Roles

There are different types of roles recommended for FOSS@RIT Telegram groups. The *Creator/Owner* role is defined by Telegram. The *Admin* and *Moderator* roles are custom roles commonly used in the FOSS@RIT community.

8.1.1 Creator/Owner

The original creator of a group retains full permissions as Owner. This includes full discretion to change group settings (including the public URL), add new admins, moderate chat, and more. This role is created by Telegram itself and will always exist. The group owner role persists even if the owner leaves and returns to the group later.

Group ownership is transferrable. If a group creator is no longer involved in the FOSS@RIT community, it is a good idea to transfer ownership to someone actively involved with FOSS@MAGIC.

At time of writing, the following group settings are only changeable by the Owner:

1. Public username / group URL
2. Visibility of chat history for new members (*visible* or *hidden*)

8.1.2 Admins

Admins are users with full permissions to change all settings of a group. Admins are appointed by the Owner or other Admins. They should have all permissions to fully manage the group:

1. Change group info
2. Delete messages
3. Ban users

4. Invite users via link
5. Pin messages
6. Add new admins
7. No custom title

What can this admin do?

- Change group info
- Delete messages
- Ban users
- Invite users via link
- Pin messages
- Add new admins

This admin will be able to add new admins with the same (or more limited) permissions.

Transfer group ownership

Custom title

admin

Desktop when adding a new admin to a group

Screenshot from Telegram

8.1.3 Moderators

Moderators are users with permissions to moderate the group chat, but not modify group information and settings. Moderators are appointed by the Owner or other Admins. They should have the following permissions:

1. Delete messages
2. Ban users
3. Invite users via link
4. Custom title: **Moderator**

What can this admin do?

- Change group info
- Delete messages
- Ban users
- Invite users via link
- Pin messages
- Add new admins

This admin will not be able to add new admins.

Custom title

Moderator

A custom title that will be shown to all members instead of 'admin'.

gram Desktop when adding a new moderator to a group

Screenshot from Tele-

8.2 List of known groups

The following is a list of known Telegram groups associated with FOSS@RIT:

Website (fossrit.github.io)

This page documents the official FOSS@MAGIC website hosted on GitHub Pages. The website source code is found in [FOSSRIT/fossrit.github.io](https://github.com/FOSSRIT/fossrit.github.io).

9.1 About

- The site is built using [Jekyll](#), [Bootstrap](#) (via CDN), [Font Awesome](#) (via CDN), & [GitHub Pages](#)
- [PDF.js](#) is used for embedding PDFs in pages (via CDN)
- “Pretty links” is turned on in the Jekyll configuration (see `permalink: pretty`)
- `github-pages` Gem used to simplify dependencies
- The live site is updated by committing changes to the `master` branch

The site uses an undocumented feature of Jekyll where categories are defined using `category/_posts/` instead of `_posts/category/`.

9.1.1 Git work flow

This describes how to make any type of contribution to the FOSS@MAGIC website:

1. Fork repository to your GitHub account (go to [FOSSRIT/fossrit.github.io](https://github.com/FOSSRIT/fossrit.github.io) and click “fork”)
2. Clone repository locally to your workstation
3. Create/add files
4. Git commit
5. Git push to your fork
6. Open pull request against `master` branch of [FOSSRIT/fossrit.github.io](https://github.com/FOSSRIT/fossrit.github.io) repository
7. When pull request is merged, site updates in a couple of minutes

9.1.2 Jekyll and Front Matter

Jekyll is what builds the site. It renders Markdown and handles templating. *Jekyll requires that all post pages start with a valid date in the syntax YYYY-MM-DD-<title>.md.*

Jekyll uses something called [Front Matter](#) to know more information about the page. Front Matter defines what the page looks like and metadata such as the title, author, and event details (where applicable).

Frontmatter should be at the top of a post and should be surrounded by triple-dashes (---). *All times are of the timezone America_New_York.*

Additionally, Jekyll uses the string `<!--more-->` to define where the summary of pages end. This is used when showing post previews on the site. Optionally, the `excerpt` frontmatter can be defined instead to override this (if using an HTML page instead of Markdown in special cases, the `excerpt` frontmatter *should be used* instead of `<!--more-->`). Please use this functionality to keep lists of posts readable.

Example post

```
---
title: Example thing
layout: post
---
Hello FOSS folks!

This week...

<!--more-->

Other things happening.

- Signature
```

9.2 Create a development environment

This website uses [containers](#) for a development environment. A container runtime like Docker or Podman works on Linux, macOS, and Windows. Containers closely mimic how the production version of the site is deployed. This means when you test your changes locally with containers, they will deploy in production successfully. This is also how maintainers test new pull requests.

9.2.1 Pre-requisites

Install one of the following container runtimes:

Docker

- Linux
 - Fedora
 - Ubuntu
- macOS

Podman

- [Podman installation instructions](#)

9.2.2 Build and serve Jekyll locally

To build the site and serve it as a website from your workstation, run the following script:

```
./build.sh
```

This script downloads the Jekyll container from Docker Hub, builds the website, and serves it on port 4000. Make sure other services are not running on port 4000 when running the script. Once the site finishes building, access the local version of the website at this URL:

```
http://localhost:4000
```

PRO-TIP: Running the script takes a while to install dependencies. However, the container continues to rebuild the website so long as the script is still running. To take advantage of this, run the `build.sh` script when you first start developing, and leave it running until you are done.

9.3 How to update FOSS Hours time/place

Edit `_config.yml` and update the settings `meeting-day`, `meeting-place`, and `meeting-time`. These settings are used across the website automatically. The meeting time and place are referenced in the `_config.yml` so they are updated in one place.

Example:

```
# --- meeting schedule settings ---  
  
# They appear exactly as typed. Follow the format already here when updating.  
meeting-day: Thursday  
meeting-place: MSS/71-3190 (conference room)  
meeting-time: 5:00PM - 7:00PM
```

9.4 How to add content

This section describes how to add new content to the FOSS@MAGIC website.

Add a new `.md` file to one of the `_posts/` folders for a category. Store assets (a.k.a. pictures, PDFs, or other downloadable files) in a folder within that category (e.g. `projects/assets/`). Include the date in asset file names so it is clear what references it.

Content categories include the following:

- [Announcements](#)
- [Events](#)
- [Projects](#)

9.4.1 How to create announcements

Announcements spread the word about what is happening in the community. Posts in this category are added to an RSS Feed (at fossrit.github.io/feeds/latest.xml).

- Create a new Markdown file in `announcements/_posts/` named with the convention `YYYY-MM-DD-<title>.md`

Example announcement

See `2019-12-17-spring-2020-volunteers.md`:

```
---
author: Justin W. Flory
title: "Call for Volunteers: Spring 2020"
layout: post
---

Hi FOSS folks!
We are looking for volunteers from the RIT community!
If you are interested in volunteering with FOSS@MAGIC in the Spring 2020 semester...
```

9.4.2 How to create events

Events are written before or after an event.

Events often use an image gallery functionality. Typically, photos are added to the gallery after an event. Use the `images` Front Matter with an array of photos. The `images` parameter takes relative links to the RITlug site or absolute links for images hosted on other sites. See the example event below for how to do this.

- Create a Markdown file in `events/_posts/` named with the convention `YYYY-MM-DD-<title>.md`

Example event

See `2019-11-05-election-night-hackathon.md`:

```
---
layout: event
title: Election Night Hackathon 2019
authors:
- Justin W. Flory
- D. Joe Anderson
excerpt: Help transcribe suffragist papers or hack on other project during Election_
↪Night.
images:
- /events/assets/2019/11/election1.jpg
- /events/assets/2019/11/election2.jpg
- /events/assets/2019/11/election3.jpg
- /events/assets/2019/11/election4.jpg
---

Students and faculty collaborated and worked on projects during the 2019 US midterm_
↪elections.

{% include content-blocks/gallery.html %}
```

(continues on next page)

(continued from previous page)

```
> FOSS@MAGIC proudly presents... the 9th Annual Election Night Hackathon!
```

9.4.3 How to add projects

Projects are about things the FOSS@RIT community work on. They are not added to an RSS Feed.

- Create a Markdown file in `projects/_posts/` named with the convention `YYYY-MM-DD-<title>.md`
 - The date is required because of a Jekyll requirement to list projects
 - Should be updated at least once a year if project is on-going

Like events, projects often use the image gallery.

Some projects use custom HTML (in-line or entirely) to have a better show effect. If using custom HTML, remember to set the `excerpt` Front Matter. If doing the page entirely in custom HTML, the `layout` should be set to `default`, which includes the FOSSRIT navbar & footer, but not the page background.

The `author/authors` frontmatter should be defined. Additionally, the `permalink` property is commonly used to give a more special URL.

Example project

See [2018-01-01-teleirc.md](#):

```
---
layout: project
title: TeleIRC
permalink: /projects/teleirc
authors:
- Mark Repka (@repkam09, original author)
- Justin W. Flory (@jwfl, current maintainer)
- Seth Hendrick (@xforever1313)
- Nate Levesque (@thenaterhood)
- and others
excerpt: TeleIRC bridges Telegram groups and IRC channels.
---

{% include content-blocks/gallery.html %}

NodeJS implementation of Telegram <=> IRC bridge...
```


CHAPTER 10

Indices and tables

- `genindex`
- `modindex`
- `search`