

User Manual



Git OSS-um

By: Gary Baker, Van Steinbrenner, Stephen White

Project Sponsor: Dr Igor Steinmacher,
Assistant professor, SICCS, Northern Arizona University

Faculty Mentor: Ana Paula Chaves Steinmacher

Table of Contents

1. Introduction.....	2
2. Installation.....	2
3. Configuration and Daily Operations.....	3
a. New User Creating an Account and Logging In.....	4
b. Request for a Repository to be Mined.....	4
c. Search and Filter Mined Repositories.....	5
d. Compare 2-3 Repositories Side-by-Side.....	6
e. View Details Page with Graphics for a Specific Repository.....	6
f. Access Admin Portal.....	7
g. Accept, Decline, or Blacklist a Mining Request.....	7
h. Delete a Repository from Mined Repos List.....	9
i. Manage Blacklisted Repos.....	12
j. Delete Users.....	12
k. Edit Users.....	14
l. Managing Periodic tasks.....	15
4. Maintenance.....	24
5. Troubleshooting.....	25
6. Conclusion.....	27

Introduction

We are pleased that you have chosen Git-OSS-um for your business needs.

Git-OSS-um is a powerful system for getting newcomers to open source started on choosing a good first repository in which to make a contribution. Some of the key highlights of this product include:

- Mining data for specific repositories directly from GitHub.
- Displaying data in easy to read data and graphics for users to interpret.
- Filtering through mined repositories.
- Allowing administrators to manage repositories mined as well as user accounts.

The purpose of this user manual is to help you successfully use, administer, and maintain the Git-OSS-um product in your actual business context going forward. Our aim is to make sure that you are able to benefit from our product for many years to come.

Installation

Since this software is a web-based application, this section can be very brief. For web-based solutions like this, all you need to do to make use of the product is:

1. Open an internet browser of your choosing.
2. Enter the URL 'gitossum.com' into the search bar.
3. Click on the 'Sign up' button in the top right of the screen.
4. Create an account, navigate to your email and click the link provided to confirm your account.

5. Now go back to 'gitossum.com' and log in to the account you just made.
6. Freely navigate the application.

If you need to install the product in order to make changes to or view the source code, please refer to the final documentation for this product where this process is completely laid out for you and any other users that need to make changes or additions to the codebase.

Configuration and Daily Operation

As the owner of this product, you already have a superuser account. This means that you have access to ALL features of the application and will not need to do any account configuration for yourself. Also, there are no more actions that need to be taken to get this product deployed. It is currently on the Digital Ocean server in which you provided and is ready for daily use and distribution to other users in the form of just the URL 'gitossum.com'.

There are many use cases associated with this product and these will be defined in the following sections as well as step-by-step instructions on how to accomplish the goal of each use case. The main use cases that we feel should be outlined in this manual are as follows:

Regular Users: A regular user is anyone who creates an account and does not have any administrative or superuser privileges. The regular user can only use the public facing portion of the application and no admin portal access will be available to them.

1. New User Creating an Account and Logging In

- a. Click on the Sign Up link in the top right of the screen.
- b. Fill in the text fields with the information that you would like to use for your account and press the Sign Up button below.
- c. Navigate to the email in which you provided in your account creation, there should be an email from gitossum asking you to confirm your email address. Click the link provided in said email to confirm your email.
- d. Navigate back to gitossum.com and click the Login button in the top left of the screen.
- e. Enter the username and password you set in your account creation and click on the Login button below.

2. Request for a Repository to be Mined

- a. Go to the Nav bar on any page of the application and click on the 'Mining Request Form' tab, this will redirect the user to the mining request form.
- b. Enter the name of the repository from GitHub that you would like to have more information about and press the Submit Request button below.
- c. The request must be in the format of 'Repo-Owner-Name/Repo-Name', the repository must exist, and the repository must have at least 1 pull

request within the project.

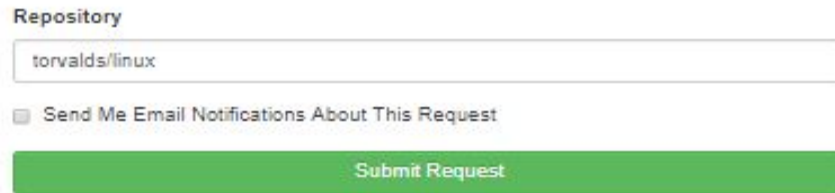
The image shows a web form for submitting a mining request. It features a text input field with the value 'torvalds/linux' and a label 'Repository' above it. Below the input field is a checkbox labeled 'Send Me Email Notifications About This Request'. At the bottom of the form is a green button with the text 'Submit Request'.

Figure 1: Mining Request Form

- d. You may click the checkbox below the input field if you would like to be notified via email when your request has been accepted or denied as well as when your repository has finished mining and is ready to view on the application if the request was accepted.
- 3. Search and Filter Mined Repositories**
- a. Go to the Nav bar on any page of the application and click on the 'Mined Repos' tab.
 - b. To search through the list of mined repositories, click on the search bar on the right side of the page and type any characters within the repository name or owner name that you are looking and press the enter key or filter button. Results will be displayed in the list area to the left.
 - c. To filter, either click on 1 or more languages, the 'Has Wiki' checkbox, or enter a min, max, or both for number of pull requests. Once all desired filters are added, press the filter button below. Results will be displayed to the left in the list area.

4. Compare 2-3 Repositories Side-by-Side

- a. Go to the Nav bar on any page of the application and click on the 'Mined Repos' tab.
- b. Click the check box to the left of the repository pictures of 2 or 3 desired repositories and click the 'Compare Repos' button below.
- c. The application will redirect you to a page where the repositories you chose to compare will be displayed, all in one table with some highlight data for each being shown.
- d. Above the table are each repositories picture, name, a link to their details page within the Git-OSS-um application, and the description of the project. The names are also links to their respective GitHub project landing pages.

5. View Details Page with Graphics for a Specific Repository

- a. Go to the Nav bar on any page of the application and click on the 'Mined Repos' tab.
- b. Click on the name of the desired repository, this will take you to the details page generated by Git-OSS-um where you can view all the statistics and graphics we have mined and generated for that repository.

Admin Users: An admin user is anyone whose account either has admin or superuser privileges. Admin users have access to not only the public facing portion of the application, but also the admin portal which is where all the managerial and administrative tasks are taken care of.

1. Access Admin Portal

- a. From any page on public facing portion of application, click on 'login' at the top right of page.
- b. If your account has admin or superuser access, the 'Admin Portal' tab will be displayed in the Nav bar which is accessible from any page on the public facing portion of the application. Clicking this tab will take you to the admin portal homepage.

2. Accept, Decline, or Blacklist a Mining Request

- a. From the homepage of the Admin portal, click on 'Mining Requests' near the bottom of the page under 'USER_APP'. This takes you to a page with

a list of all mining requests that require action.

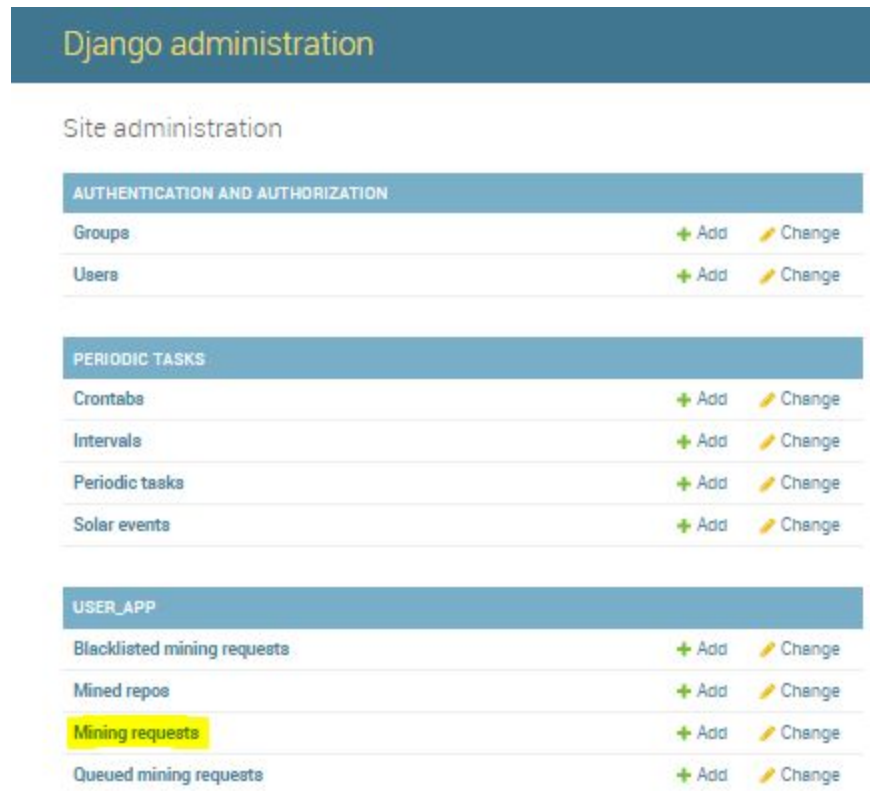


Figure 2: Admin Portal - Mining Requests

- b. To accept, decline, or blacklist one or multiple requests, click the checkbox next to any requested repositories, then click on the drop down to the top of the list of requests. The drop down contains 3 options, choose one of the 3 options and click the 'Go' button to the right and that

action will be performed on all checked requests.

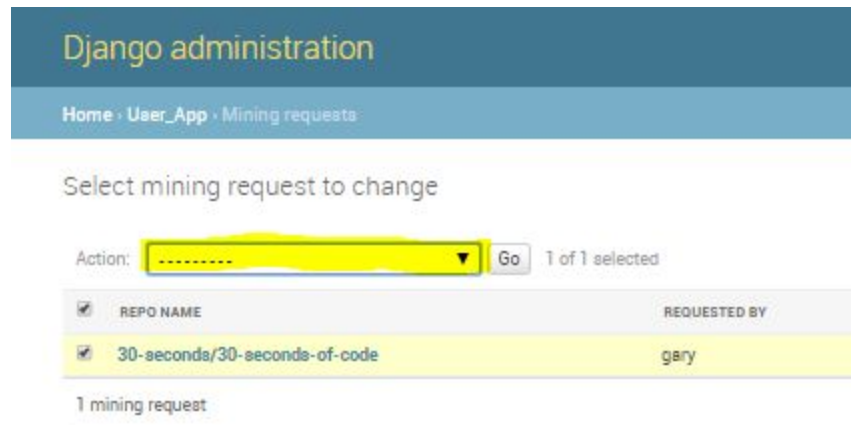


Figure 3: Mining Request Section of the Admin Portal

- c. NOTE: All accepted requests are then moved to the 'Queued Mining Requests' list which can be accessed from the Admin homepage under 'USER_APP'.
 - d. NOTE: All blacklisted requests are then moved to the 'Blacklisted Mining Requests' list which can be accessed from the Admin homepage under 'USER_APP'.
- 3. Delete a Repository from Mined Repos List**
- a. From the Admin homepage, under 'USER_APP', click the 'Mined Repos' link.

Django administration

Site administration

AUTHENTICATION AND AUTHORIZATION	
Groups	+ Add ✎ Change
Users	+ Add ✎ Change

PERIODIC TASKS	
Crontabs	+ Add ✎ Change
Intervals	+ Add ✎ Change
Periodic tasks	+ Add ✎ Change
Solar events	+ Add ✎ Change

USER_APP	
Blacklisted mining requests	+ Add ✎ Change
Mined repos	+ Add ✎ Change
Mining requests	+ Add ✎ Change
Queued mining requests	+ Add ✎ Change

Figure 4: Admin Portal - Mined Repos

- b. To delete a repository from the list of Mined Repos, click one or multiple checkboxes to the left of the repository name, then click the drop down at the top of the list, click the 'Delete selected repos from database' option

and click the 'Go' button to the right.

Django administration

Home > User_App > Mined repos

Select mined repo to change

Action: Go 1 of 17 selected

<input type="checkbox"/>	REPO NAME	REQUESTED BY	COMPLETED TIMESTAMP
<input type="checkbox"/>	elixir-lang/elixir	igor	May 7, 2019, 12:37 a.m.
<input checked="" type="checkbox"/>	torvalds/linux	gary	May 7, 2019, noon
<input type="checkbox"/>	google/gumbo-parser	stephen	May 7, 2019, noon
<input type="checkbox"/>	microsoft/calculator	stephen	May 7, 2019, noon
<input type="checkbox"/>	jabref/jabref	stephen	May 7, 2019, 12:02 p.m.
<input type="checkbox"/>	qiime2/qiime2	stephen	May 7, 2019, 12:03 p.m.
<input type="checkbox"/>	blockchain/blockchain-wallet-v4-frontend	stephen	May 7, 2019, 12:03 p.m.
<input type="checkbox"/>	tidyverse/dplyr	stephen	May 7, 2019, 12:04 p.m.
<input type="checkbox"/>	trekhleb/javascript-algorithms	stephen	May 7, 2019, 12:04 p.m.
<input type="checkbox"/>	tuvtran/project-based-learning	stephen	May 7, 2019, 12:04 p.m.
<input type="checkbox"/>	valloric/youcompleteeme	stephen	May 7, 2019, 12:05 p.m.
<input type="checkbox"/>	kriadmin/30-seconds-of-python-code	stephen	May 7, 2019, 12:05 p.m.
<input type="checkbox"/>	django/django	stephen	May 7, 2019, 12:13 p.m.
<input type="checkbox"/>	rails/rails	stephen	May 7, 2019, 12:29 p.m.
<input type="checkbox"/>	inorichi/tachiyomi	LILDEE1R	May 7, 2019, 12:30 p.m.
<input type="checkbox"/>	tencent/omi	anachaves	May 7, 2019, 12:31 p.m.
<input type="checkbox"/>	nau-oss/cs499-group02	gare-bake	May 7, 2019, 12:31 p.m.

17 mined repos

Figure 5: Mined Repos in the Admin Portal

- c. NOTE: The deleting of a repository from the database cannot be undone after clicking the confirmation that pops up after clicking the 'Go' button. The only way a repo can be brought back to the database is to submit a request for it again and accept it.

4. Manage Blacklisted Repos

- a. From the Admin homepage, under 'USER_APP', click the 'Blacklisted Repos' link.
- b. To delete a repository from the list of Blacklisted Repos, click one or multiple checkboxes to the left of the repository name, then click the drop down at the top of the list, click the 'Delete selected repos from database' option and click the 'Go' button to the right.
- c. Deleting a repository from this list will allow users to request that it be mined again. While a repository is on this list, the users will be notified when trying to make a request for that repository has been blacklisted and their request will not be submitted.

5. Delete Users

- a. From the Admin homepage, under 'Authentication and Authorization', click the 'Users' link.

Django administration

Site administration

AUTHENTICATION AND AUTHORIZATION	
Groups	+ Add Change
Users	+ Add Change

PERIODIC TASKS	
Crontabs	+ Add Change
Intervals	+ Add Change
Periodic tasks	+ Add Change
Solar events	+ Add Change

USER_APP	
Blacklisted mining requests	+ Add Change
Mined repos	+ Add Change
Mining requests	+ Add Change
Queued mining requests	+ Add Change

Figure 6: Admin Portal - Users

- b. From this list page, you can delete users by clicking the checkbox to the left of the username, then clicking the drop down menu and selecting the 'Delete selected users' option. Then click the 'Go' button to the right to delete.

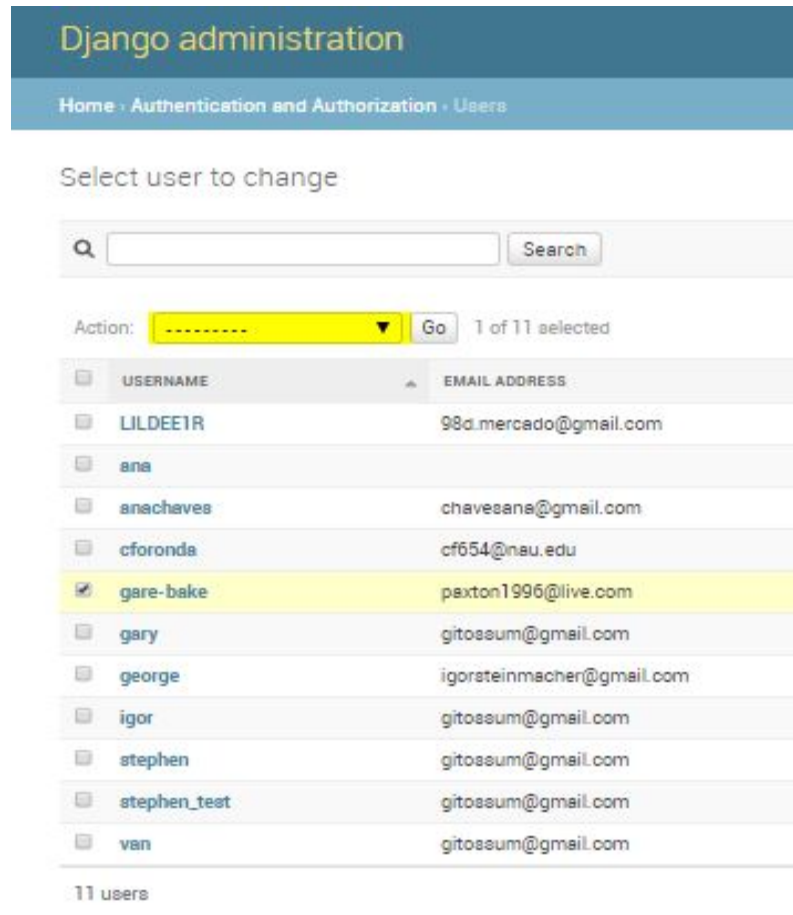


Figure 7: Users within the Admin Portal

6. Edit Users

- a. From the Admin homepage, under 'Authentication and Authorization', click the 'Users' link.
- b. From this list, click on a username of a user you would like to edit. This will bring you to a page with all of the user's information as it relates to the Git-OSS-um application.
- c. Within the user page, you can update their username, personal info, and permissions.

- d. There are 3 checkboxes (Active, Staff Status, Superuser) directly under the 'Permissions' heading that are quick permissions adjustments.
 - i. **Active** just denotes whether or not a user is active, this can be unchecked instead of completely deleting a user if an admin feels it is necessary.
 - ii. **Staff Status** gives the user permission to log in to the admin portal and do ONLY the things manually set in the User Permissions menu below. All permissions must be added manually but individual permissions may be selected rather than just giving permission to everything.
 - iii. **Superuser** status gives a user the permission to access every feature of the application both on the public facing side as well as the admin side.

7. Managing Periodic tasks

- a. **Crontabs**: Crontabs are another form of interval that can be used to schedule periodic tasks using month, day of month, day of week, hour, and minute.

- i. Crontabs can be managed by navigating to the 'Crontabs' link under the 'Periodic Tasks' heading on the Admin homepage.

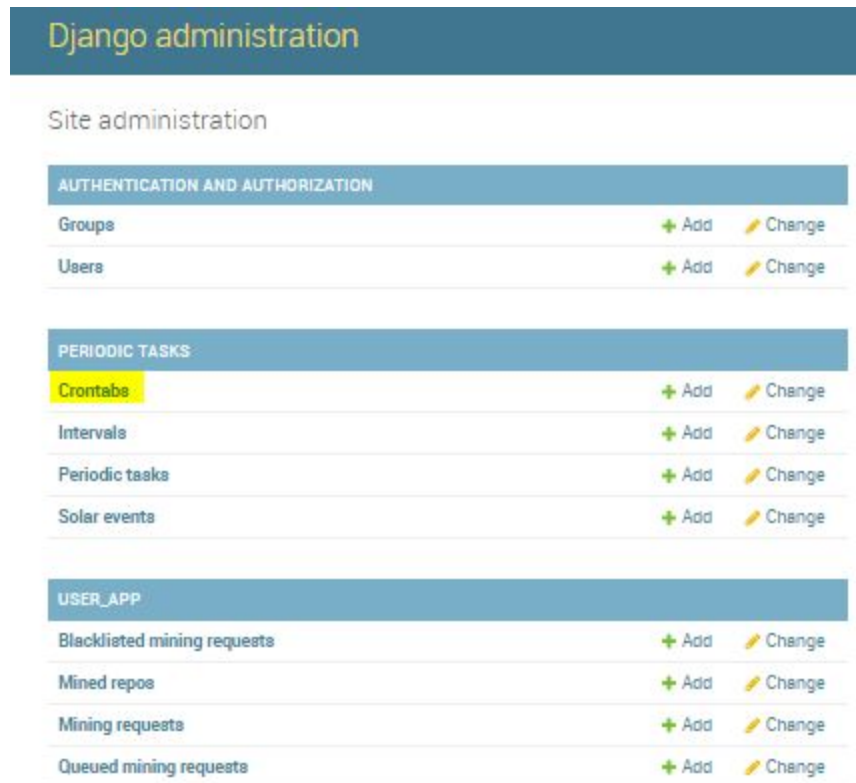


Figure 8: Admin Portal - Crontabs

- ii. Crontabs can be deleted by clicking on the check box to the left of the Crontab name, then clicking the checkbox above the list and selecting 'Delete selected Crontabs' and clicking the 'Go' button.
- iii. Crontabs can be edited by clicking on the name of the Crontab.
- iv. New Crontabs can be created by clicking the 'ADD CRONTAB' button near the top right of the page. This will open a form in which you enter the details of the Crontabs you would like to create.

When the details are entered, click the 'Save' button at the bottom right of the form.

b. **Intervals:** These are the different recurrence intervals that can be used when scheduling periodic tasks using days, hours, minutes, seconds, and microseconds.

i. Intervals can be managed by navigating to the 'Intervals' link under the 'Periodic Tasks' heading on the Admin homepage.

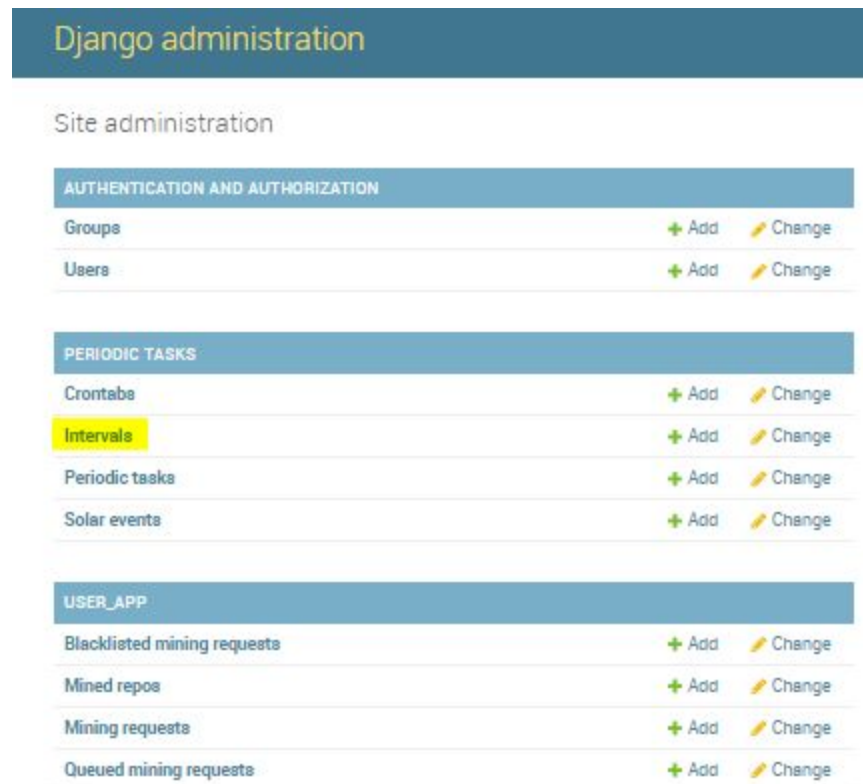


Figure 9: Admin Portal - Intervals

ii. Intervals can be deleted by clicking on the check box to the left of the interval name, then clicking the checkbox above the list and selecting 'Delete selected intervals' and clicking the 'Go' button.

- iii. Intervals can be edited by clicking on the name of the interval.
 - iv. New intervals can be created by clicking the 'ADD INTERVAL' button near the top right of the page. This will open a form in which you enter the details of the interval you would like to create. When the details are entered, click the 'Save' button at the bottom right of the form.
- c. **Periodic Tasks:** Periodic tasks are scheduled tasks that happen when specified by the admin using a Crontab, Interval, or Solar Event. These tasks can be used with the following specific functions created within the software:

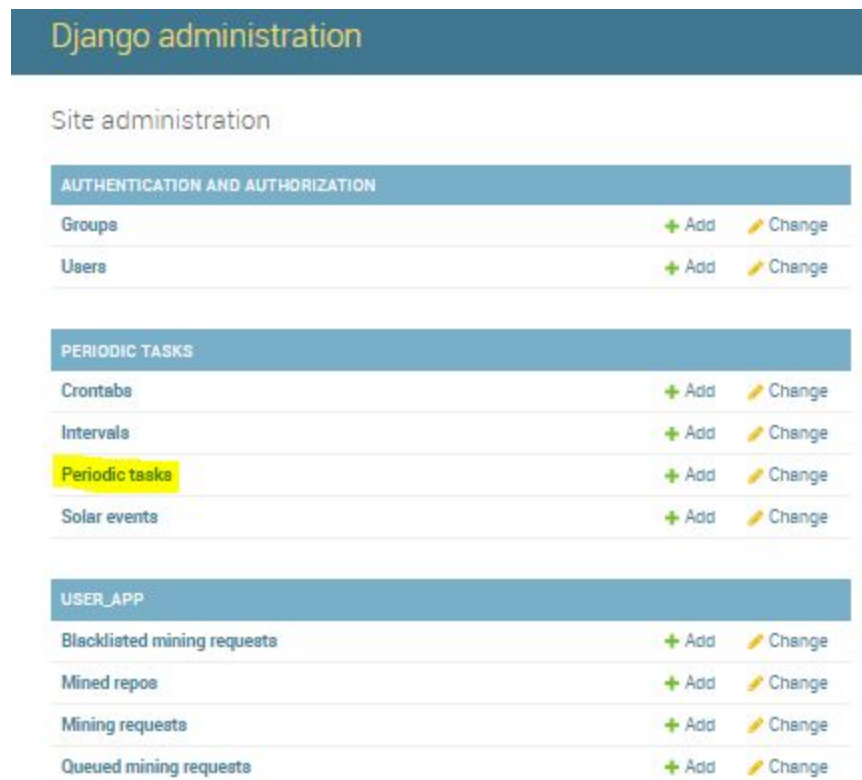


Figure 10: Admin Portal - Periodic Tasks

- i. **Visualize Repos:** The image below displays what the admin would see within the “visualize all repos” periodic task. By default this task is set to check for new repositories that need to be visualized every 30 seconds. This value can be changed under the “interval” or “crontab” subsection within the schedule section, and the changes will automatically be updated for the celery scheduler.

Name:
Useful description

Task (registered):

Task (custom):

Enabled

Schedule



Interval:  

Figure 11: Visualize Repos Periodic Task

- ii. **Update all repos:** This periodic task is used to make sure all of the repositories that the system mines will be updated in a timely manner to make sure all information that is presented to the user is indeed up to date. By default, this task is made to use the crontab

(0 */12 * * *), which means on the 0th minute every 12 hours, every day, every month, every year. By that logic, all of the repositories will be updated twice daily, once at midnight, and again at noon. The image below shows what you should see within the admin panel. If you would like to change the interval, just make changes to the “crontab” subsection with the “schedule” section.

Name: Update all repos
Useful description

Task (registered):

Task (custom): tasks.update_all_repos

Enabled

Schedule

Interval: ----- ✎ +

Crontab: 0 */12 * * * (m/h/d/dM/MY) ✎ +
Use one of interval/crontab

Figure 12: Update all Repos Periodic Task

- iii. **UpdateSpecificRepos:** This task is slightly different than the previous two. In order to update a specific repository, you would

first go to the periodic tasks section of the Django Admin Portal, and click on “Add Periodic Task +” which is placed within the top right corner.

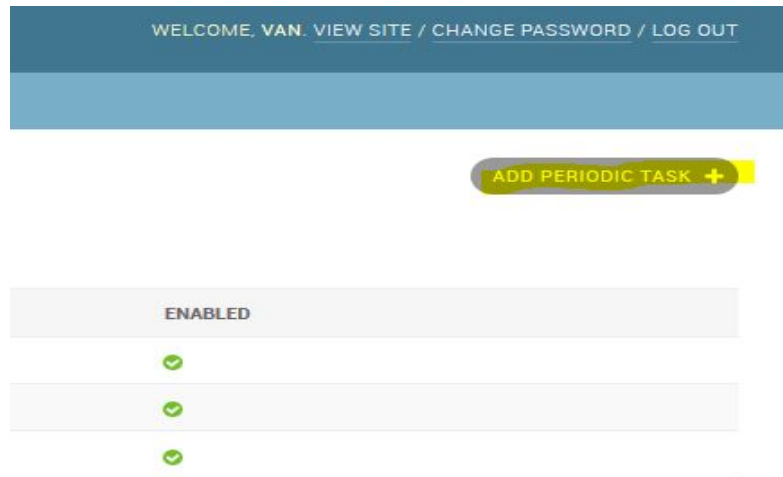


Figure 13: Add a Periodic Task

- iv. You will then fill out some general information about the repository that you would like to update such as the name of this periodic task, the specific celery task which will be used (this will always be “tasks.update_specific_repo”), and the interval/crontab you would like to use. This is depicted in Figure 14.

Add periodic task

Name:
Useful description

Task (registered):

Task (custom):

Enabled

Schedule

Interval: ✎ +

Crontab: ✎ +
Use one of interval/crontab

Solar: ✎ +
Use a solar schedule

Figure 14: Information to be Filled for Adding Periodic Task

- v. The most important step here however, is to then click on the “show” button under the arguments tab, and to add the name of repository as a string, as depicted in Figure 15.

Arguments (Hide)

Arguments:

JSON encoded positional arguments

Figure 15: Name of Repo Under Arguments

- vi. Click “Save” and now this repository will be added to the celery beat scheduler!

- d. **Solar Events:** Solar events are similar to intervals. These allow you to create a recurring interval for a periodic task to use and they are based on the solar events.
 - i. Solar Events can be managed by navigating to the ‘Solar Events’ link under the ‘Periodic Tasks’ heading on the Admin homepage.

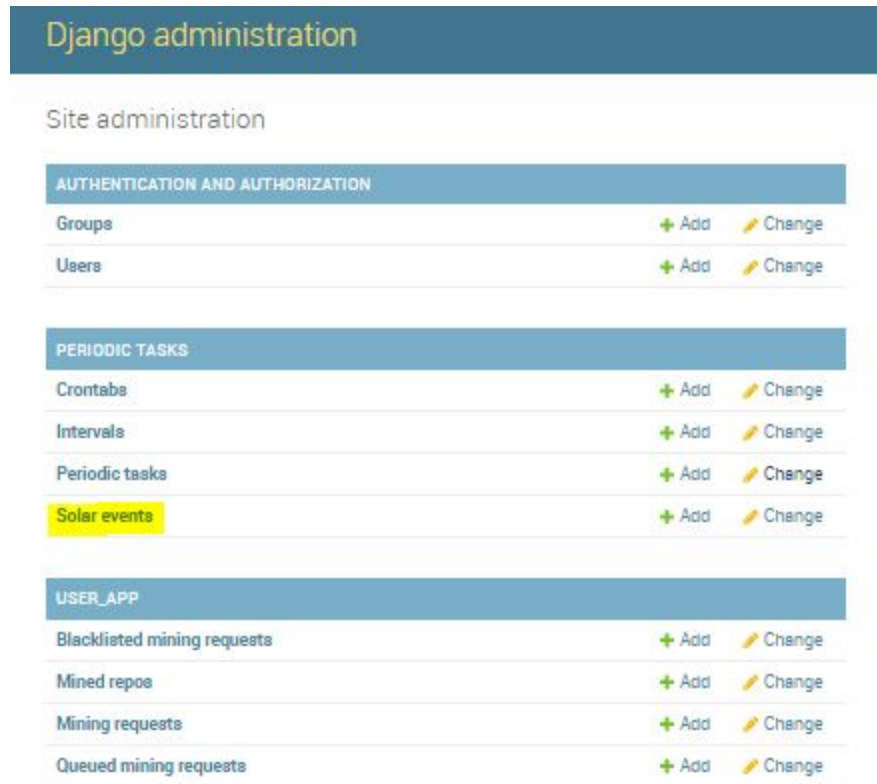


Figure 16: Admin Portal - Solar Events

- ii. Solar Events can be deleted by clicking on the check box to the left of the event name, then clicking the checkbox above the list and selecting ‘Delete selected solar events’ and clicking the ‘Go’ button.
- iii. Solar Events can be edited by clicking on the name of the event.

- iv. New solar events can be created by clicking the 'ADD SOLAR EVENT' button near the top right of the page. This will open a form in which you enter the details of the event you would like to create. When the details are entered, click the 'Save' button at the bottom right of the form.

Maintenance

When using your application, it is necessary to take care of some simple things to keep it running smoothly. In this case, there are just two things that should be done on an occasional basis (about once a month):

1. Clean out the logfile for Celery tasks.
 - a. Within the digital ocean server, navigate to `/var/celery/logs`. Here you will see two log files that celery writes to for your own convenience. Every once in a while it would be good practice to clear these out so the server doesn't take up unnecessary space with files full of text. To do this, simply download the log files in their current state, save those copies to a more permanent location, and delete the ones on the server.
2. Restart the Digital Ocean server.
 - a. Simply type the command "sudo shutdown -r now" within the command line to reboot the server. The server will automatically disconnect you, and will be back online shortly thereafter. This should be done every once in a while to clear out garbage processes that may be laying around.

Troubleshooting

Throughout your use of the product, things may go wrong or not work exactly as expected. The following are some of the issues that we think may happen from time to time and some proposed fixes to those potential issues.

1. Repository Not Mining

a. Solution 1

- i. Access database via command line and delete this repository from database.
- ii. Re-request the repository, accept the request, check if it mines within a reasonable amount of time.

b. Solution 2

- i. Restart the Digital Ocean server using the process listed in the Maintenance section above.

c. DISCLAIMER

- i. Keep in mind that very large repositories with tens of thousands of pull requests could potentially take many hours to mine so if a repository you are trying to mine is large, make sure you give it a reasonable amount of time before you start troubleshooting.

2. The Repo I Requested Exists, but Git-OSS-Um Says it Doesn't

a. Solution 1

- i. Contact the administrator so they may fix this issue. This issue arises when the regular expression the developers have written to

encapsulate GitHub repository names did not account for a particular character pattern. A simple update to the code base can easily fix this issue.

3. A Repo Shows up on the Mined Repos Tab, But Says 404 Repo Not Found When I Click On It

a. Solution 1

- i. This indicates that something went wrong with the “visualize repo” task. Contact the administrator and have them check to see if the repository does exist within the database.

b. Solution 2

- i. If the repository does exist within the database and still shows “404 Repo Not Found”, the administrator may need to remove this repository and mine it again.

Conclusion

We would like to take this last moment to thank you for the opportunity to work on such a fun and exciting project and for providing such a great Capstone experience for our team. We truly enjoyed working with you on this product and have learned a great deal in the process. We hope that this tool will be useful for you and many others from now on and maybe someday it could be adopted by GitHub and advertised as a true tool for newcomers to open source.

While we are moving on to our professional careers, we would be happy to answer short questions in the coming months to help you while mastering your new product and making full use out of it.

With best wishes from your Git-OSS-um development team,

- Gary Baker - gpb27@nau.edu
- Van Steinbrenner - vws2@nau.edu
- Stephen White - smw444@nau.edu