| TLE Filtering and Notam Plotting |  |  |  |
| --- | --- | --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  | **PROJECT** |
|  |  | TLE Filtering and Notam Plotting |
|  |  |
|  |  | **WORK PACKAGE NUMBER** |
|  |  | WP10001 |
|  |  |
|  |  | **Document No.** |
|  |  | ASL\_GEN\_000 |
|  |  |
|  |  | **REVISION** |
|  |  | 2.0 |
|  |  |
|  |  | **EDITED DATE** |
|  |  | 2 February 2022 |
|  |  |
|  |  | **STATUS** |
|  |  | Release |
|  |  |
| **PREPARED BY Worapong Veraburenon** |  |  | **CLASSIFICATION** |
|  |  |  | Non-Classified |
|  |  |
| **REVIEWED BY Sittiporn Channumsin** |  |  | **RESEARCH LABORATORY** |
|  |  |  |  |
|  |  |
| **APPROVED BY Keerati Puttasuwan** |  |  |
|  |  |  |
|  |  |
| เอกสารฉบับนี้เป็นสมบัติของสทอภ. ห้ามนำไปใช้นอกเหนือจากวัตถุประสงค์ที่ได้กำหนดไว้ |  |  | **สนง.พัฒนาเทคโนโลยีอวกาศและภูมิสารสนเทศ** 88 หมู่ 9 ต.ทุ่งสุขลา อ.ศรีราชา จ. ชลบุรี 20230 ประเทศไทย โทรศัพท์: 033-046-300 |
|  |  |
|  |  |
|  |  |

**สารบัญ**

**หน้า**

[1. Reentry TLE Filtering 1](#_heading=h.26in1rg)

[1.1.1. Webpage 1](#_heading=h.lnxbz9)

[1.1.2. MySQL data update 2](#_heading=h.1ksv4uv)

[1.2.1. Setting up refresh time on webpage and celery server 3](#_heading=h.z337ya)

[1.2.2. Function “SGP4\_120km” 4](#_heading=h.4i7ojhp)

[1.2.3. Function “Spacetrack\_to\_TLE” 4](#_heading=h.2xcytpi)

[1.2.4. Function “upload\_to\_mysql” 5](#_heading=h.2bn6wsx)

[2. Plotting Notam 5](#_heading=h.qsh70q)

[2.3.1. Function “import\_launchsite” 8](#_heading=h.3o7alnk)

[2.3.2. Function “notam\_degree\_convert” 8](#_heading=h.23ckvvd)

[2.3.3. Function “distance\_2points” 8](#_heading=h.ihv636)

[2.3.4. Function “centroid” 9](#_heading=h.32hioqz)

[3. Conclusion 9](#_heading=h.1hmsyys)

**สารบัญรูป**

**หน้า**

[Figure 1: Reentry TLE filtering webpage 1](#_heading=h.35nkun2)

[Figure 2: MySQL TLE reentry data 2](#_heading=h.44sinio)

[Figure 3: Error message occur when didn't open redis server 3](#_heading=h.2jxsxqh)

[Figure 4: Webpage's refresh time script 3](#_heading=h.3j2qqm3)

[Figure 5: Celery server's refresh time code 4](#_heading=h.1y810tw)

[Figure 6: Changing directory of the txt. file 5](#_heading=h.1ci93xb)

[Figure 7: space-track.org username an8d password in txt. File 5](#_heading=h.3whwml4)

[Figure 8: Example of NOTAM data 6](#_heading=h.3as4poj)

[Figure 9: NOTAM plotting webpage 6](#_heading=h.1pxezwc)

[Figure 10: Form insided the webpage 7](#_heading=h.49x2ik5)

[Figure 11: Example on how to use the form 7](#_heading=h.2p2csry)

[Figure 12: Output of the example 8](#_heading=h.147n2zr)

**บันทึกการแก้ไข**

| วันที่แก้ไข | เลขที่หน้า | รายละเอียดการแก้ไข |
| --- | --- | --- |
| 2 ก.พ. 2565 | - | สร้างเอกสารครั้งแรก |
|  |  |  |

**การส่งออก**

| จำนวน | ชื่อ | หน่วยงาน |
| --- | --- | --- |
|  | - |  |
|  |  |  |

1. **Reentry TLE Filtering**

In the past few years, the incoming space objected that will impact with the earth are forecasted news from other space agency. However, the incoming broadcast is very late, making the time to prepare or predict, which had to be done manually, the incoming objects less. This program will predict and filtering the space objects surrounding around the earth in the form of TLE (Two Line Element) acquiring from space-track.org.

* 1. Overview
     1. **Webpage**

Text

Description automatically generated

*Figure 1: Reentry TLE filtering webpage*

This page will show the list of the incoming space objects in the form of TLE, the impact time, coordinate, and speed, sorted by the impact time. The page will be refreshed depending on the setting (by default, 5 min.) but it can refresh manually by refreshing the page. The time shows in the page is the time of the last refresh.

The space objects data are from space-track.org. The program will filter the objects which have perigee less than 200 km and predict by using SGP4 prediction. The TLE shown in this page can be copy directly for future use.

The time column in the table shows the time of the impact predicted by SGP4 prediction. The prediction is done iteratively, by using SGP4 prediction of the current TLE data and predict the coordinate and speed of the object every one hour from the current epoch time until the distance, which is calculated from the where x, y and z is the coordinate, is less than the altitude of 120 km. The form of the time shown in the page is [ year, month, day, hour, minute, second, id].

The coordinate column in the table is in the form of true equator mean equinox in kilometer. The speed shown in the table is in the form of kilometer/second. The coordinate and time then can be used for the future use.

* + 1. **MySQL data update**

Graphical user interface, text, application

Description automatically generated

*Figure 2: MySQL TLE reentry data*

This program can also import and update the data calculated automatically without opening the webpage. To do this, the celery function will be used. First, open two terminal in the coding program, one is for celery beat and other is for celery worker. Then, create virtual environment for both terminals. Next, type in “celery -A django\_project worker -l info --pool=solo” for celery worker and “celery -A django\_project beat -l info” for celery beat. After submitting the commands, the error will show as shown in the picture below. To solve this error, open the redis-server and the program will start importing and updating the TLE data, epoch time, impact time, coordinate, and speed to the MySQL.

Graphical user interface, text, application

Description automatically generated

*Figure 3: Error message occur when didn't open redis server*

While opening the auto importing server, it is not necessary to open the webpage simultaneously. However, it should be aware that the refreshing time of the webpage and server shouldn’t be synchronizing. For example, the refresh time of the webpage is 9.00 am. And refresh every 30 min., the server shouldn’t refresh synchronously with webpage refresh time. This will cause the error in importing data from space-track.org, which takes time to import the data (approximately 2 min).

* 1. Method
     1. **Setting up refresh time on webpage and celery server**

To change the refresh time on the webpage, it can be done by going to /pages/template/index.html as shown in the picture below. In the line 14, “setTimeout(‘location.reload()’, 300000)”, the 300000 mean 300 second or 5 min. This means that to refresh to every 1 min, the number on that line should be 60000. However, it’s not recommended to change the refresh time to less than 3 min due to the error in importing data from space-track.orgGraphical user interface, text, application, email

Description automatically generated

*Figure 4: Webpage's refresh time script*

To change the refresh time on celery server, similar to webpage, go to /Django\_project/celery.py as shown in picture below. On line 14, “'schedule': 300.0”, same as webpage, 300.0 mean 300 second or 5 min. This mean 60.0 equal to 1 min. However, same as webpage, it’s not recommended to change the refresh time less than 3 min due to space-track.org importing.

Moreover, requesting the data from the space-track.org too frequently can lead to the error too. Space-track.org limits the requested data for each user, requesting too many data can lead to the bad credential to that user and the website will prevent the user from requesting more data. So, it is recommended to setting the refresh time of the server to every 1 hour, which is the time some of the data is updated and not effect with the credential of the space-track website.

Graphical user interface, text, application

Description automatically generated

*Figure 5: Celery server's refresh time code*

* + 1. **Function “SGP4\_120km”**

This function needs the TLE line 1 and line 2. It will calculate the epoch date and time from TLE and will run the calculation iteratively using SGP4 prediction until the altitude of the object is 120 km. The function then returns the epoch date/time, date and time, coordinate, and speed when the object reaches 120 km altitude and the error code. Error 0 means no error. Error 1-6 means error in SGP4 prediction and Error 7 mean the iteration is diverging.

* + 1. **Function “Spacetrack\_to\_TLE”**

This function will import the TLE data from the space-track.org. The code below will query the TLE data from the space-track database, where NORAD\_CAT\_ID/> 40000/ means NORAD ID more than 40000, perigee/<200/ means perigee less than 200 km, EPOCH/>now-10 mean epoch date and time from present to last 10 days and orderby/EPOCH%20desc mean data are sorted by epoch date. 

To login to spacetrack-org through python, the path of the text file that consist of username and password need to be assigned, as show below. The context of text file should be in the form as shown in picture below. The data then will return the TLE line 1 and 2 that is quried by the database command.



*Figure 6: Changing directory of the txt. file*

Text

Description automatically generated

*Figure 7: space-track.org username and password in txt. File*

* + 1. **Function “upload\_to\_mysql”**

This function need NORAD id, TLE line1 and 2, epoch date/time, impact date/time, coordinate, and speed. These data will be upload to MySQL database by this function. The function will be run when running the celery server. The process of the function will be printed in the celery worker terminal. New NORAD id object will be added. The existing data will be skipped, and the updated data will be update in MySQL

1. **Plotting Notam**

To reduce the weight and have better fuel efficiency, the multi- stage rocket is required to send the satellite to the orbit. Especially nowadays, the number of the rocket launch is gradually increasing. One of the disadvantages of the multi-stages rocket is the lack of controlling the separated part, which can cause a damage or casualty near the crash site. For Thailand, NOTAM zones are declared by Aeronautical Radio of Thailand, which can also be used to clarify the separation phase of the rocket and the crash site. However, the NOTAM data is needed to be converted into a coordinate. This program will help plotting the NOTAM zone and flight path of that rocket to predict the hazardous area and potential damage from the impact.

Text

Description automatically generated

*Figure 8: Example of NOTAM data*

* 1. Overview

Map

Description automatically generated

*Figure 9: NOTAM plotting webpage*

When opening the website, the map will show, along with the form on the upper left conner. The map can be zoom in and out by using the + and – symbol on the upper left corner or using the scroll wheel. The map can also pan by hold and drag left click. The rocket symbol on the map is a launch site location, which can be clicked to show the name of the launch site. The launch site data are imported from discosweb.esoc.esa.int and uploaded to MySQL, which can be add or edit the data of the launch site.



*Figure 10: Form insided the webpage*

On the form, there are one drop box, one text box and two button. On the drop box, the launch site or the launch location of the can be selected here. On the text box, the NOTAM zone can be inserted or pasted here. If user wants to add more NOTAM zone, click on the “Add More Fields” button to generate additional textbox. When finish applying the form, click submit to draw the NOTAM polygons and the flight path from the selected launch site to the furthest polygon’s centroid on the map.

* 1. Example

Text

Description automatically generated

From the picture above, the rocket is launching from the Sriharikota Space center with three NOTAM zone. So, select the Sriharikota Space center from the drop-down box. Click the “Add More Field” button to create more text box for more NOTAM zone. Then, copy each line, which represent each NOTAM zone to each text box. The NOTAM data can paste non sequential order. The look of the form will be as shown below. Click submit to show the result as shown below.

A picture containing text

Description automatically generated

*Figure 11: Example on how to use the form*

Map

Description automatically generated

*Figure 12: Output of the example*

* 1. Method
     1. **Function “import\_launchsite”**

This function will import the launch site data from the MySQL database, including name, azimuth, latitude, longitude, and altitude of the launch site. These data can be edited or added by open MySQL, host = “172.27.188.71”, user = “arlapp” ,password = “fdsus2019”, and go to internship/launchsite.

* + 1. **Function “notam\_degree\_convert”**

This function will convert the polygon in the form of NOTAM to form of latitude and longitude. The input data can be copied and pasted from the NOTAM data and the function will return the data in the form of list containing the latitude and longitude of each coordinate of the polygon.

For example, the input data of this function is 'N123000E0824000 N131500E0825000 N124500E0841000 N120000E0840000'. The data then return '[(12.5, 82.66666666666667), (13.25, 82.83333333333333), (12.75, 84.16666666666667), (12.0, 84.0)]', where each element in this list is latitude and longitude of each coordinate of the polygon.

* + 1. **Function “distance\_2points”**

This function will calculate the distance between two input coordinates. The input coordinate should be in the form of latitude and longitude. For example, the input of this function is '((3, 8),(12, 15))', where (3,8) is one of the coordinate and (12,15) is another coordinate. The output of this function will be ‘11.4017’. This output then be used to find the furthest polygon from the launch site to draw the flight path.

* + 1. **Function “centroid”**

This function will calculate the coordinate of the centroid of the input polygon. The input coordinate can be in the form of the output of the function “notam\_degree\_convert”. For example, the input data is '[(12.5, 82.66666666666667), (13.25, 82.83333333333333), (12.75, 84.16666666666667), (12.0, 84.0)]'. The output of the function will be ‘[12.625, 83.41666666666667]’, where the first element is latitude, and second element is longitude of the centroid of the input set of coordinates.

1. **Conclusion**

To be conclude, these projects, reentry TLE filtering and NOTAM plotting, are part of Space Situation Awareness project. The TLE filtering import the data from space-track.org and predict the reentry date/time, coordinate, and speed by using SGP4 propagation. The program then able to auto import the predicted data to MySQL database by using the celery function. The NOTAM plotting is useful in showing the hazardous area causing by the impact of the separated part from the rocket launch. The function built in the program can also be used in another related program or future project.

For future project, The TLE filtering then can be used to predict the reentry process by using C++ program. This prediction can predict the impact location and connect to the map database to create the warning system and show the hazardous area when the incoming object is going to impact. It will be very useful which can reduce the workload and much more accurate prediction and result

For the NOTAM plotting, the project will continue importing the NOTAM data from the web service automatically and calculate the launch site and flight path of these rocket from the imported NOTAM data automatically by calculate the most reasonable launch site from the NOTAM polygons’ centroid trajectory and distance.