Swapnal Shahil

Mail: swapnalsahil@gmail.com
Portfolio: https://swapnalshahil.github.io/
Github: https://github.com/swapnalshahil
Linkedin: https://www.linkedin.com/in/swapnalshahil/

SCoRe Lab - Open MF

Google Summer of Code 2021

About Me

I am currently in the 2nd year of my Bachelor's Degree from the Indian Institute of Technology Guwahati (IIT G). I have been tinkering with the codes since the first year of my college. I love working on open-source projects and learn every time new things from projects and the community. I have good knowledge of Python, React, Js, ES7, HTML, CSS with my interest in the web, and I have been contributing to this organization since December and will continue further in all my possible way.

Analytics API and UI Development

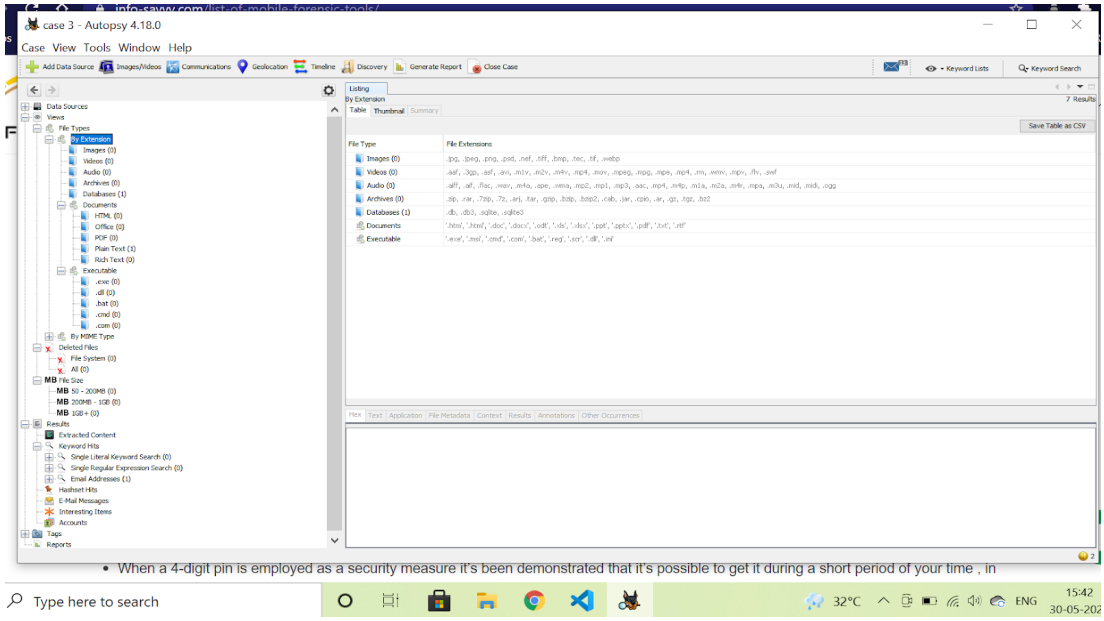
Research:

Available Analytics tool:

1. Sleuth Kit(+Autopsy):
    - It's an open-source digital forensics toolkit that can be used for in-depth analysis of various file systems.

Features:

- List allocated and deleted ASCII and Unicode file names.
- Display file names and metadata structure
- Lookup file hashes in a hash database, such as the NIST NSRL, Hash Keeper, and custom databases that have been created with the 'md5sum' tool.
- Organize files according to their type.
- Multimedia - Extract EXIF from pictures and watch videos
- Give the final report in various modules like HTML, text, excel report.
- Also, it opens one case at a time and works on it.

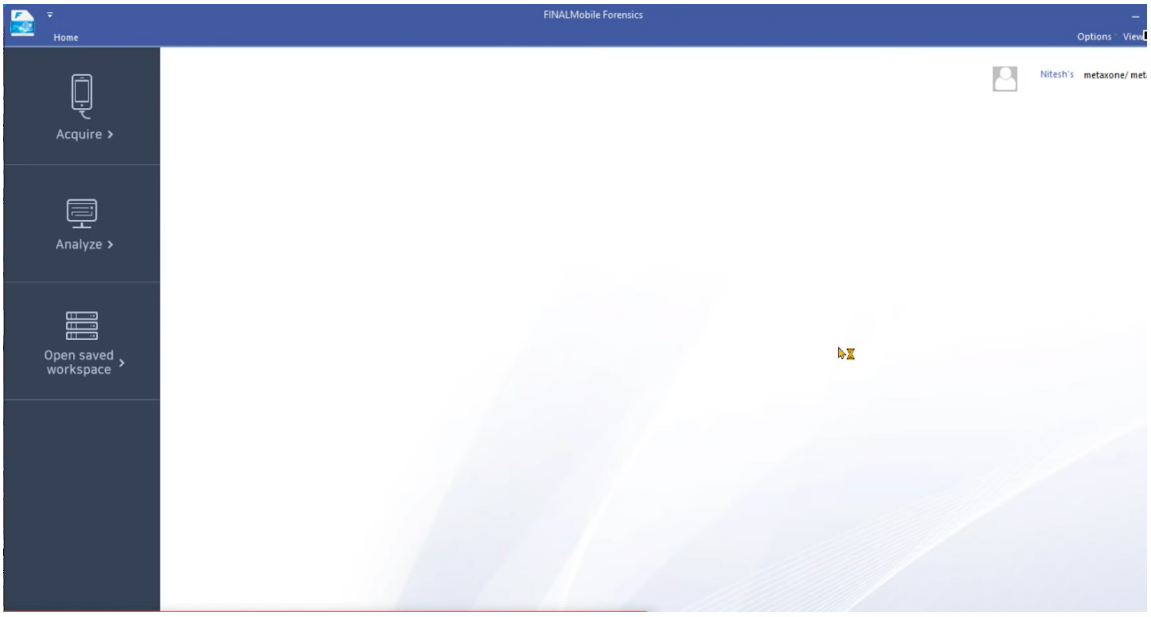


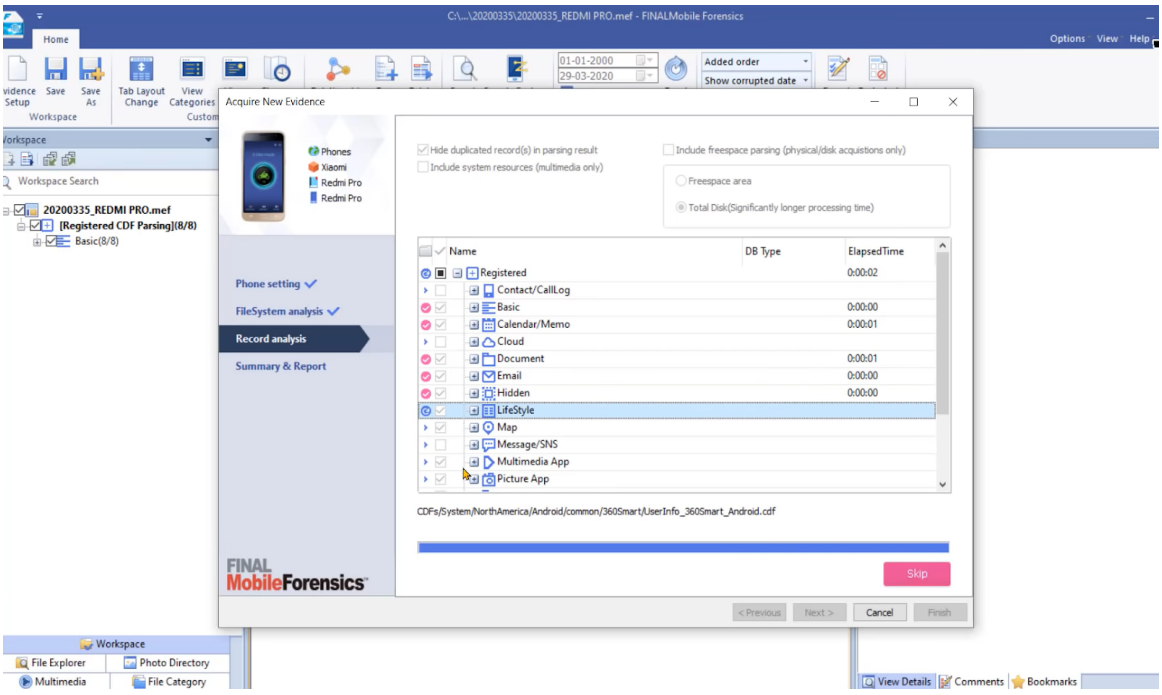
2.FINALMobile Forensics

- It simply captures data and analyzes it from mobile devices through logical and physical acquisitions.
- Not an open-source tool
- Subscription needed.

Features:

- Program is able to read its own generated phone images(MEFs)
- After data is loaded it auto-generates filename on basis of the model number of mobile devices.
- Results are generated in the HTML,pdf,excel.
- Display the file and all formats of files.



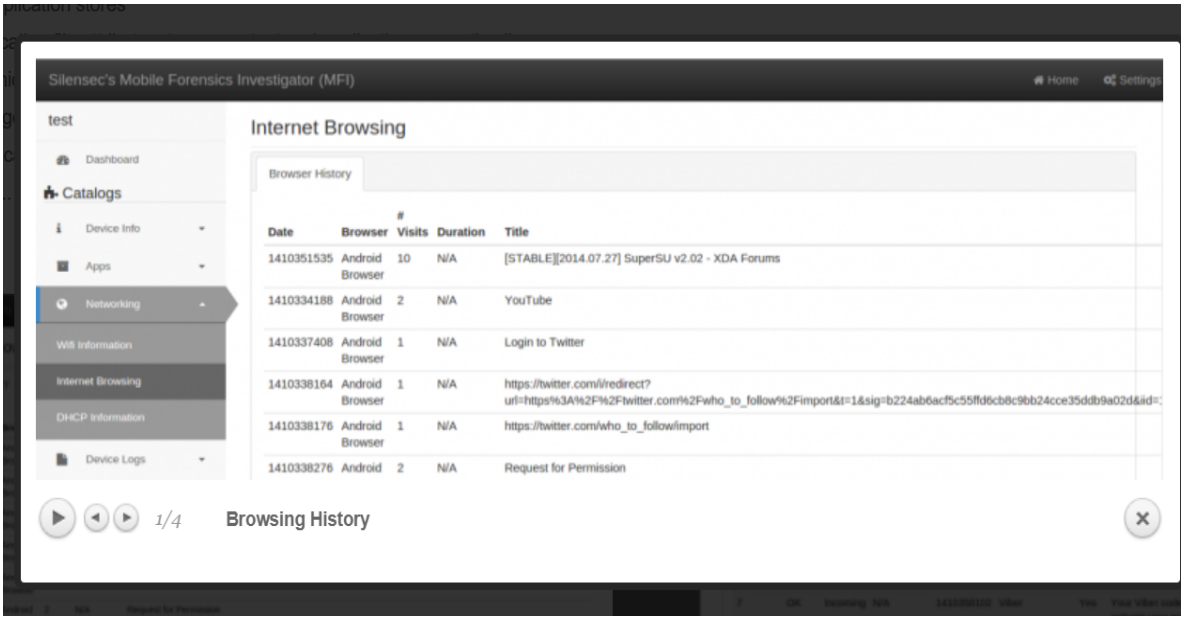


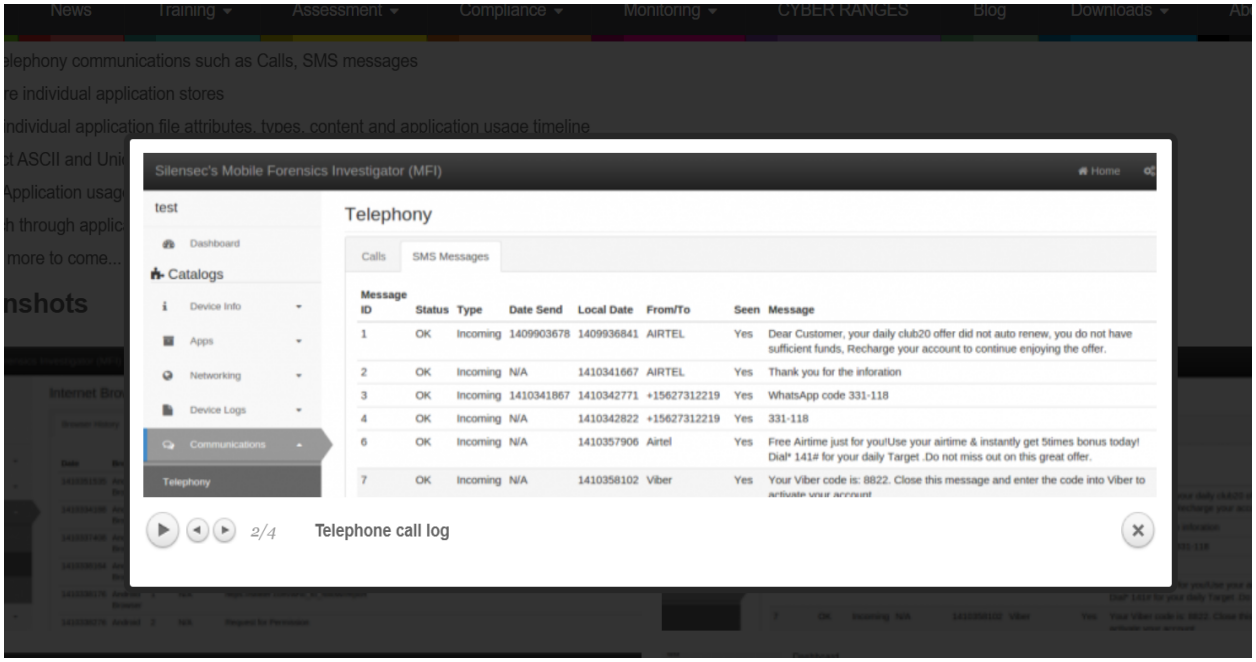
3.Nyuki Forensic Investigator (NFI)

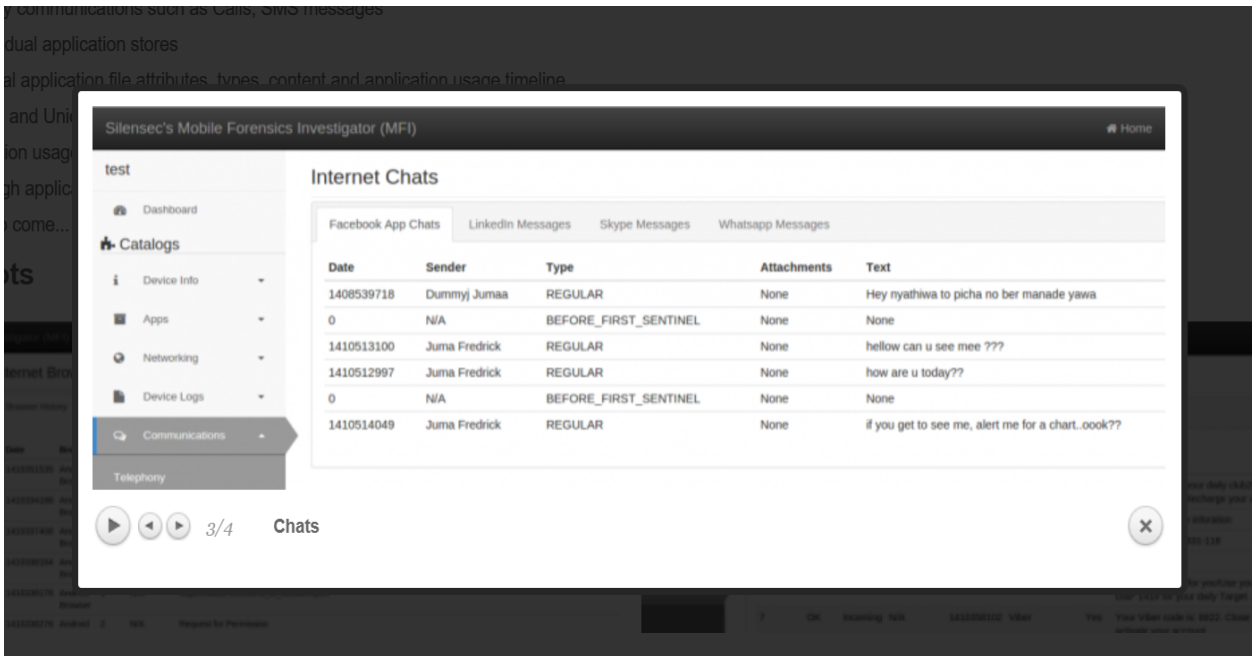
- It’s an open-source application that provides analysis for mobile device artifacts.

Features:

- Explore the device contacts, Facebook friends, LinkedIn connections, Whatsapp contacts
- Read through Facebook, LinkedIn, Skype chats
- List Telephony communications such as Calls, SMS messages
- Information related to Bluetooth devices, wifi networks, IP address leases.
- Extract ASCII and Unicode strings from Application files
- Files in a systematic way.




Silensec's Mobile Forensics Investigator (MFI)
Home
test
Dashboard
Catalogs
Device Info
Apps
Networking
Device Logs
Communications
Telephony
Telephony
Calls
SMS Messages
Message ID
Status
Type
Date Send
Local Date
From/To
Seen
Message
1 OK Incoming 1409903678 1409936841 AIRTEL Yes Dear Customer, your daily club20 offer did not auto renew, you do not have sufficient funds, Recharge your account to continue enjoying the offer.
2 OK Incoming N/A 1410341667 AIRTEL Yes Thank you for the inforation
3 OK Incoming 1410341867 1410342771 +15627312219 Yes WhatsApp code 331-118
4 OK Incoming N/A 1410342822 +15627312219 Yes 331-118
6 OK Incoming N/A 1410357906 Airtel Yes Free Airtime just for you!Use your airtime & instantly get 5times bonus today! Dial* 141# for your daily Target .Do not miss out on this great offer.
7 OK Incoming N/A 1410358102 Viber Yes Your Viber code is: 8822. Close this message and enter the code into Viber to
2/4
Telephone call log


Silensec's Mobile Forensics Investigator (MFI)
Home
test
Dashboard
Catalogs
Device Info
Apps
Networking
Device Logs
Communications
Telephony
Internet Chats
Facebook App Chats
LinkedIn Messages
Skype Messages
Whatsapp Messages
Date
Sender
Type
Attachments
Text
1408539718 Dummyj Jumaa REGULAR None Hey nyathiwa to picha no ber manade yawa
0 N/A BEFORE_FIRST_SENTINEL None None
1410513100 Juma Fredrick REGULAR None hellow can u see mee ???
1410512997 Juma Fredrick REGULAR None how are u today??
0 N/A BEFORE_FIRST_SENTINEL None None
1410514049 Juma Fredrick REGULAR None if you get to see me, alert me for a chart..oook??
3/4
Chats

1. Correlation between different data present in the database:-
    - One-to-one(1:1): Ex- Account No. and Account holder name. Each account can have only one account holder.
    - One-to-many(1:N): Ex- Like a customer can order different things from the store.
    - Many-to-many(N:N): Ex - In a company N employees can work on M projects.
2. Analytics:-

    A. Graphical: It basically shows data in a graphical way which saves lots of time.

    Types:
    - Centrality analysis: Estimates how important a node is for the connectivity of the network.
    - Connectivity analysis: Determine how strongly or weakly connected two nodes are.
    - Path analysis: Examines the relationships between nodes. Mostly used in shortest distance problems.

    B. String-based:

    Most of the digital investigations rely on textual evidence, this is obviously due to the fact that most of the stored digital data is linguistic. Ex- logged conversation, a lot of important text-based evidence can be gathered while dumping strings from images (smartphone memory dumps) and can include emails, instant messaging, address books, browsing history, and so on. Most of the currently available digital forensic tools rely on matching and indexing algorithms to search textual evidence at a physical level so that they search every byte to locate specific text strings.

C. Pattern Based:

Pattern Analytics can be defined as a discipline of Big Data that enables business leaders to understand how different variables of the business interact and are linked with each other. Variables can be of any kind and within any data source, structured as well as unstructured. Machine Learning(ML) is mainly used to handle such big data.

3. Data can be presented in one of the three ways:

a) Text presentation: Text can be used to provide interpretation or emphasize certain data. If quantitative information to be conveyed consists of one or two numbers, it is more appropriate to use written language than tables or graphs.

b) Table presentation: Tables convey information that has been converted into words or numbers in rows and columns. Easy to understand. Tables are the most appropriate for presenting individual information and can present both quantitative and qualitative information.

c) Graph presentation: Graphs are effective for presenting large amounts of data, they can be used in place of tables to present small sets of data. A graph format that best presents information must be chosen so that readers and reviewers can easily understand the information.

Project Goals

1. Generate dummy dataset from a rooted device.
2. Extraction of the text messages.
3. Extraction of browser history.
4. Extraction of Bluetooth files.
5. Extracting data (EXIF) from images(like location, date, and time)
6. Extraction of location from the device.
7. Keyword search - Implementation of keyword search to get all the cases that have the searched keyword.
8. Implementing keyword search from cases.
9. Maximum occurred common word/data between two cases.
10. Returning all the common words between two cases.
11. Filtering Cases by Date or range of date.
12. Creating APIs for the above.
13. Tag association and tag-based search like searching cases that are tagged with theft to get all the theft-related cases.
14. Customized keyword search <kw1> + <kw2> + … + <kwN>
15. Analytics based on location.
16. Implementing analytics on the front end.

17. Presentation of observed analytics - graph, library, an API will be developed which will send the database data to the graph generating library to create required graphs and that will show in our OpenMF frontend.

18. Presentation of observed analytics between two cases using graphs and tables.

19. Update the project wiki.

20. Will write blogs and explain the project on Linkedin and Medium. Also, I have an idea to make video tutorials explaining the project and work, which further publish on Youtube and Linkedin to gain the crowd on the Website.

Timeline

Note: Each cell contains a time period of 1 week.

Week No.	Date	Work
0)	**17 May - 7 June**	Community Bonding (take feedback from mentors, Modify timeline and project details)
Coding officially begins!		
1)	**7 June - 13 June**	• Generate dummy dataset from a rooted device. • Extraction of the text messages. • Creating API which will show the extracted text messages. • Implementation from the command line so that we can directly extract SMS.
2)	**14 June - 20 June**	• Extraction of browser history • Extraction of Bluetooth files/data. • Creating APIs which will show the extracted files for the above. • Implementation from the command line to extract browser history

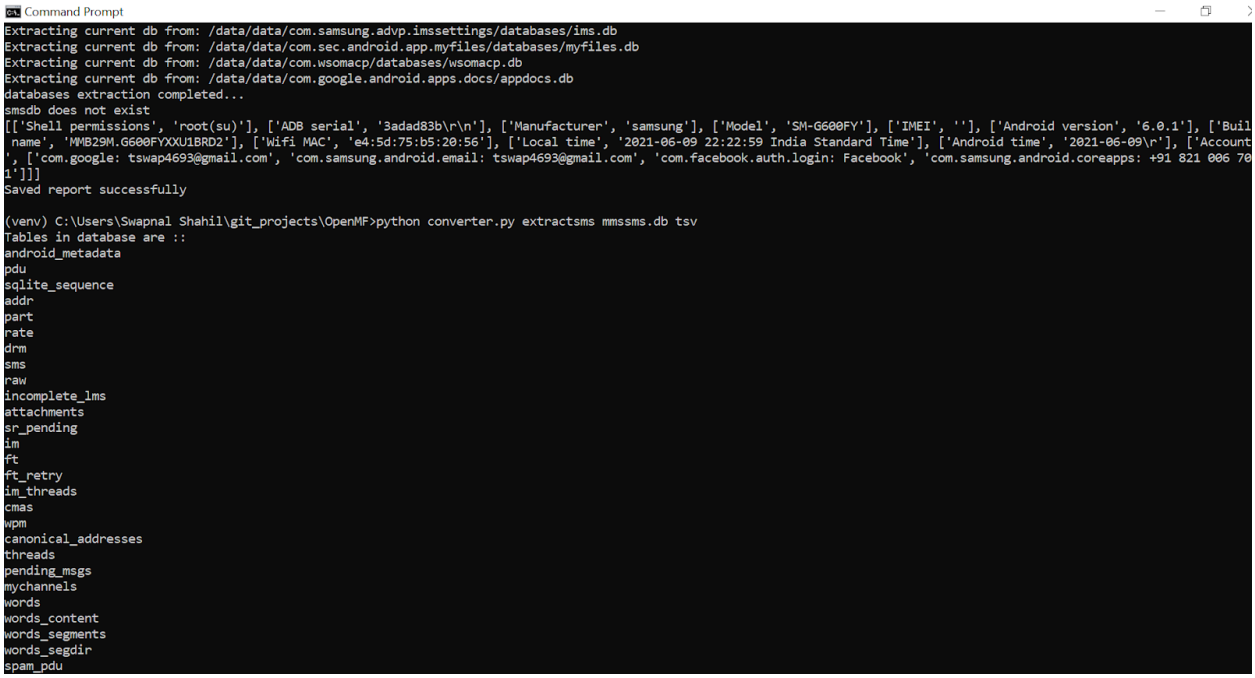
		and Bluetooth data directly.
3)	**21 June - 27 June**	• Extracting data (EXIF) from images(like location, date, and time) • Extraction of location from the device • Creating APIs which will show the extracted files for the above. • Implementation from the command line to extract EXIF and location data directly.
4)	**28 June - 4 July**	• Keyword search - Implementation of keyword search to get all the cases that have the searched keyword. • Implementing keyword search from cases. • Creating APIs for the above.
5)	**5 July - 11 July**	• Maximum occurred common word/data between two cases. • Returning all the common words between two cases. • Creating APIs for the above.
Evaluations (July 12- July 16)		
6)	**12 July - 18 July**	• Taking feedback from mentors and working on bugs(if any). • Leftover work(if any) • Filtering Cases by Date or range of date. • Creating API for that. • Implementing frontend from week 4th,5th, and 6th in the analytics section.
7)	**19 July- 25 July**	• Tag association and tag-based search like searching cases that are tagged with theft to get all the theft-related cases. • Customized keyword search <kw1> + <kw2> + … + <kwN> • Analytics based on location • Frontend implementation for this week.
8)	**26 July - 1 Aug**	• Presentation of observed analytics - graph, library, an API will be developed which will send the database data to the graph generating library to create required graphs and that will show in our OpenMF frontend.
9)	**2 Aug - 8 Aug**	• Presentation of observed analytics between two cases using graphs and tables.
10)	**9 Aug - 15 Aug**	• Leftover work (if any) • Writing Wiki.

		• Writing Medium Blog for documenting all progress and work done.
Final Evaluation week starts (Aug 16- Aug 23)		
Final results of Google Summer of Code 2021 announced (August 31)		

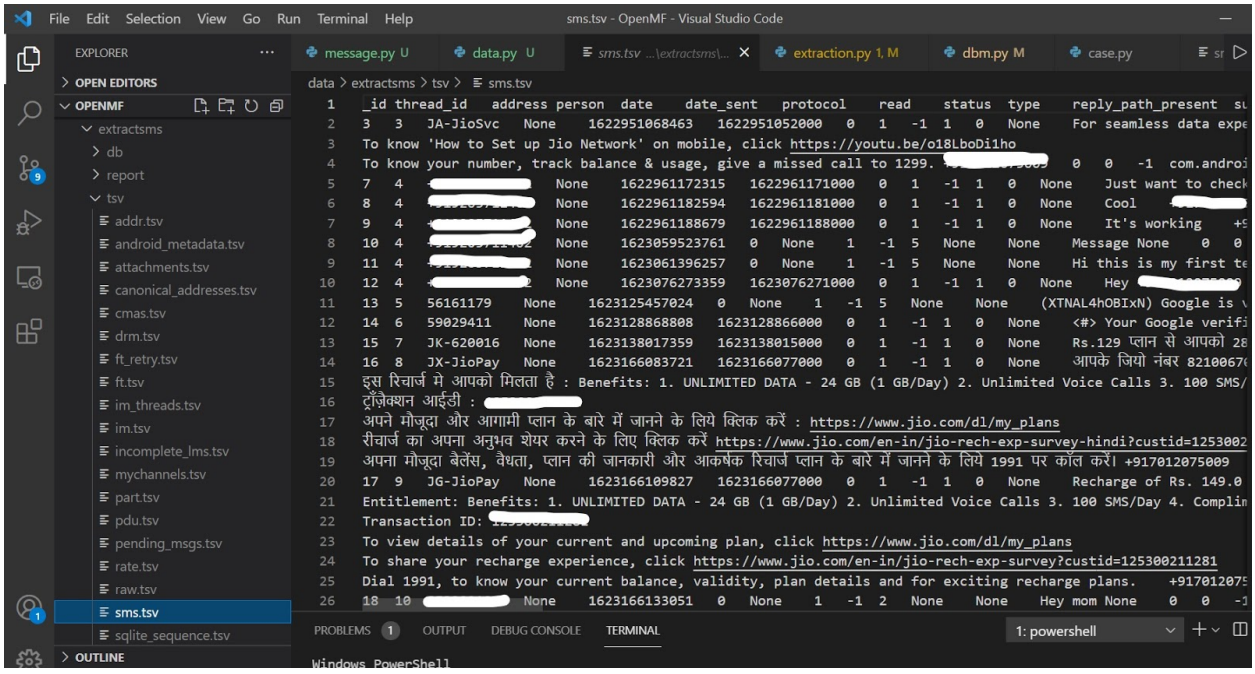
Implementation:

(7 June - 13 June)

Setting up the project and generated some dummy data which will be important in the next work period. Also, working on the extraction part to extract text messages (SMS) from android devices and Implementation from the Postman and CMD so that SMS can be directly extracted through dedicated command. To make this useful on the frontend and analysis, An API is also created that will provide a data path for SMS of the desired case stored in the database.



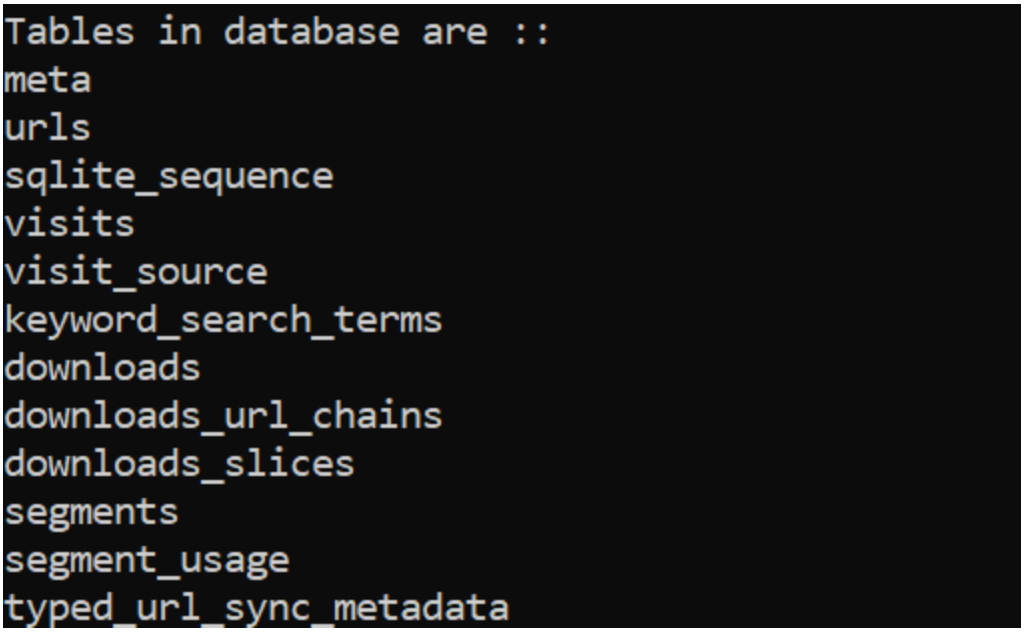
(Extraction of text messages using CMD)



(Extracted text messages(SMS))

(14 June - 20 June)

Work on the extraction part of the browser and write scripts for the same. For this, research all the tables from the database and try to gather all the necessary information which will be useful for mobile forensic. Implementation from POSTMAN and CMD to directly extract browser data through dedicated command and write an API that will provide the data path to history.tsv, which will contain all the necessary information of the desired case stored in a database.



Tables from the chrome history database

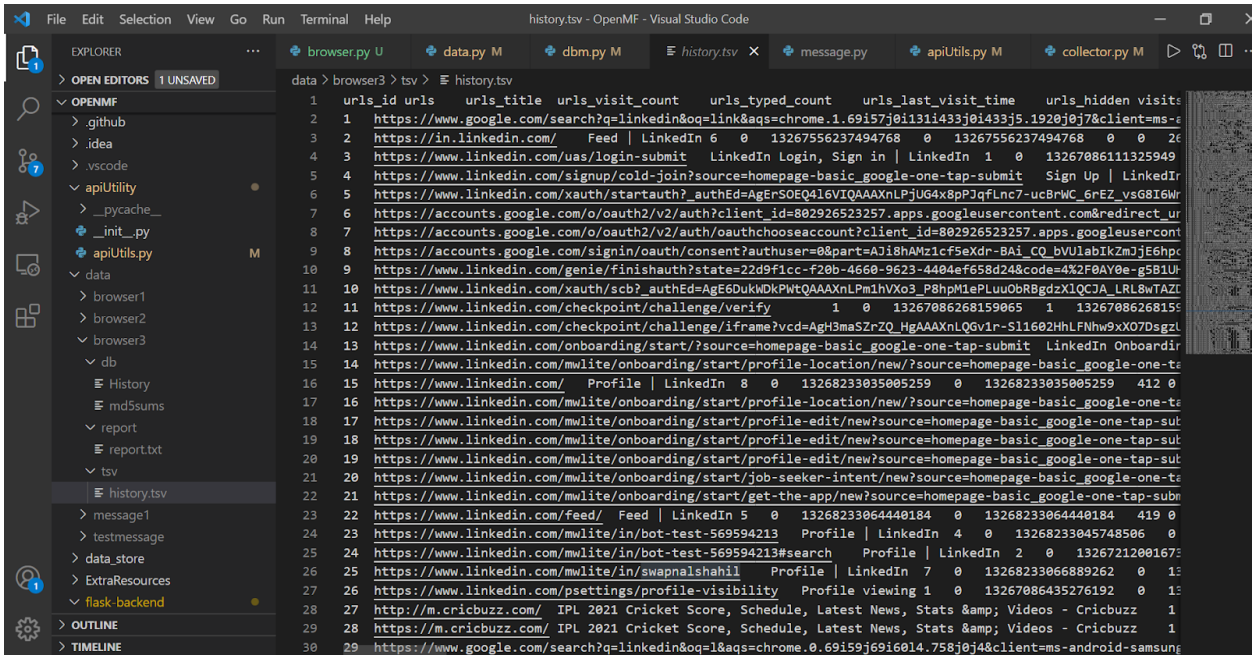
```
(venv) C:\Users\Swapnal Shahil\Documents\OpenMF>python collector.py --option browser --session_name browser4
* daemon not running; starting now at tcp:5037
* daemon started successfully
 Shell permissions: root(su)

General Device Information

 ADB serial: 3adad83b

 Device model: samsung SM-G600FY
 IMEI:
 Android version: 6.0.1
 Build number: MMB29M.G600FYXXU1BRD2
 Wi-fi MAC: e4:5d:75:b5:20:56
 Local time: 2021-06-17 11:39:12 India Standard Time
 Android time: 2021-06-17
 Sync'ed Accounts.
com.google: tswap4693@gmail.com
com.samsung.android.email: tswap4693@gmail.com
com.samsung.android.coreapps: [redacted]
{'-h', '--help'}
Starting data extraction plan for given options
Running data extraction for selected options :
Options to parse are :  ['browser']
Session name :  browser4
Extracting all common databases ...
Extracting current db from: /data/data/com.google.android.apps.messaging/databases/bugle_db
Extracting current db from: /data/data/com.google.android.dialer/databases/dialer.db
Extracting current db from: /data/data/com.android.providers.contacts/databases/calllog.db
```

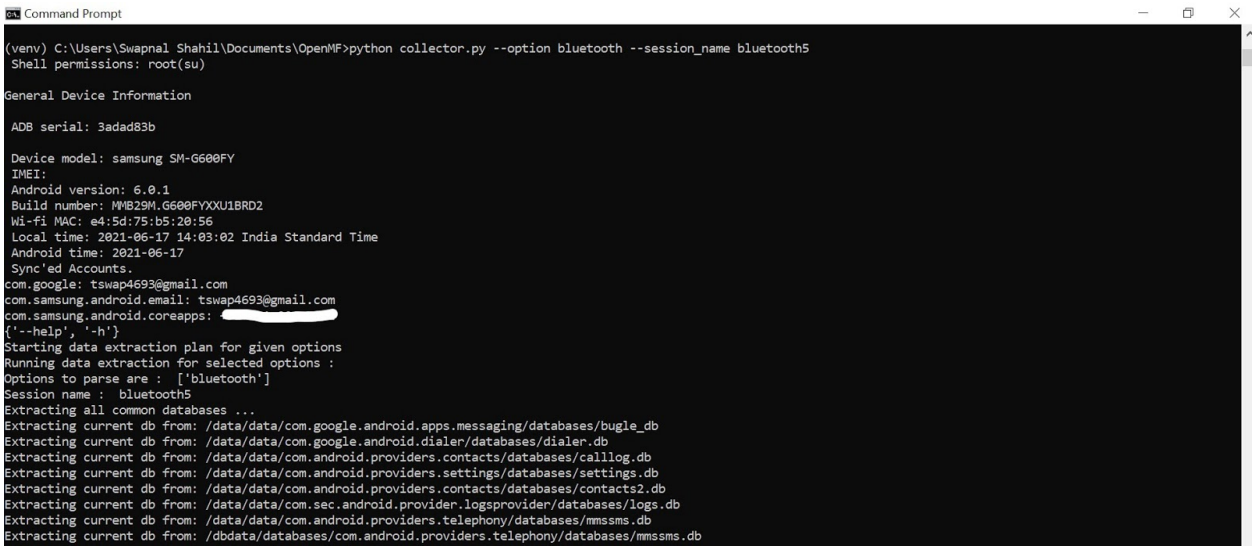
Extraction of browser data using CMD



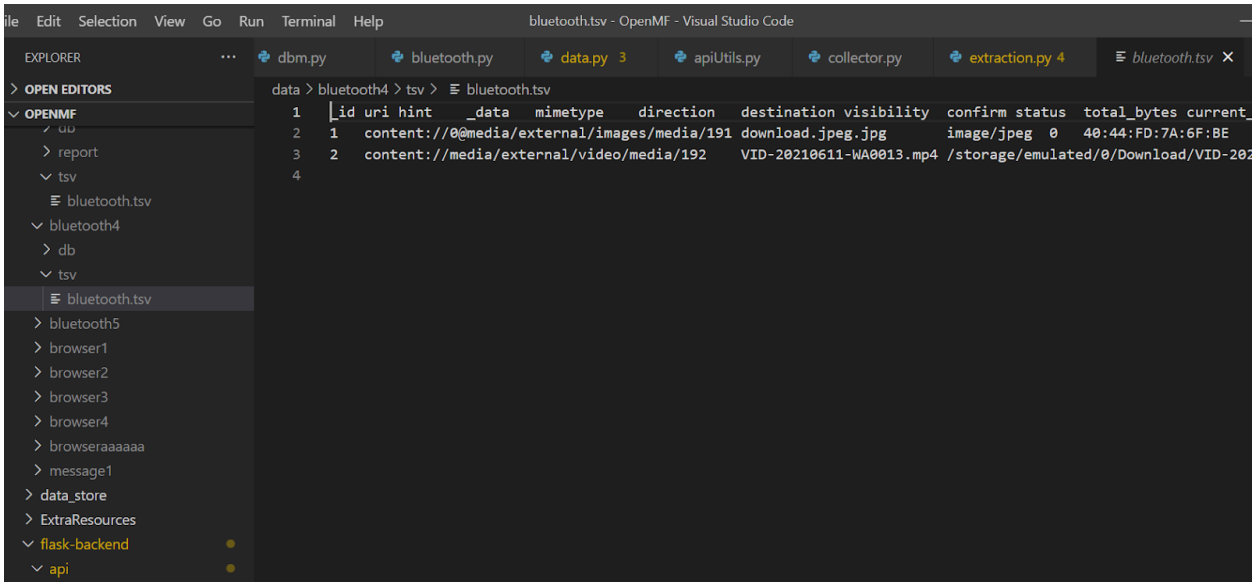
Extracted data in history.tsv

Later work on the extraction of Bluetooth data from an android device and implement this to extract from Postman and CMD, same as the browser part. From this, we will get all the

information like which data is transferred from the current android device to whatever connected device, data path, time when data is shared, direction and destination, total bytes, and device name.



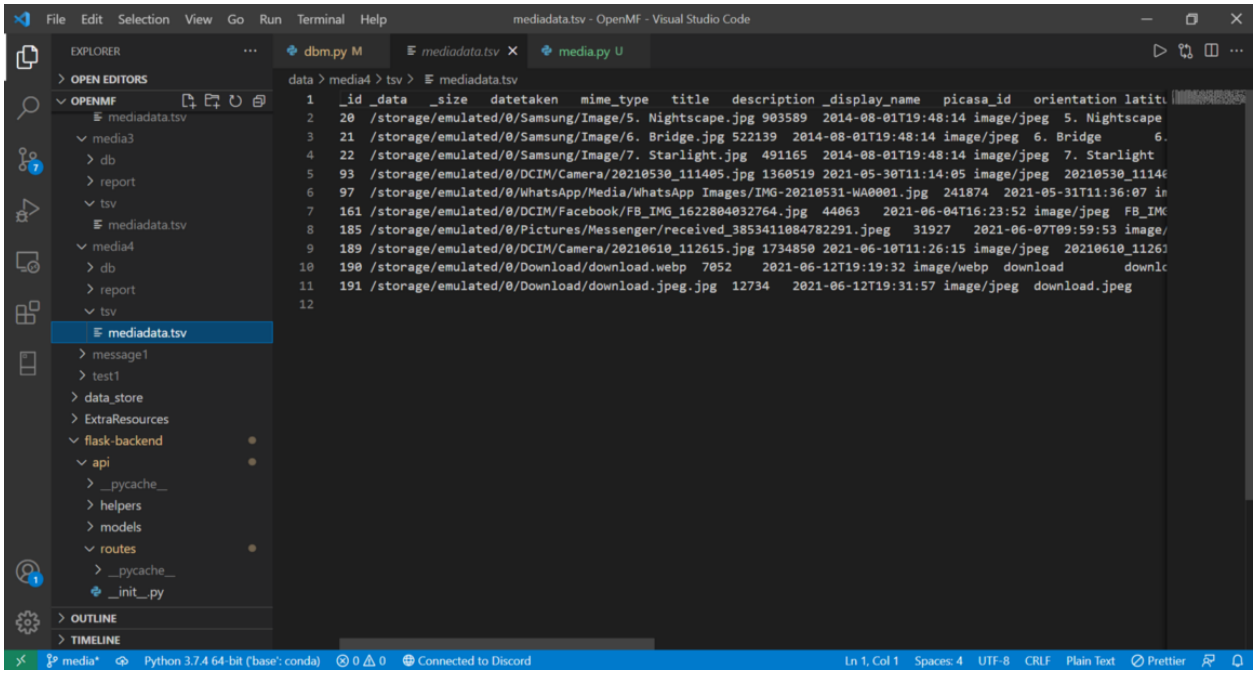
Extraction of Bluetooth data using CMD



Extracted data in bluetooth.tsv(above)

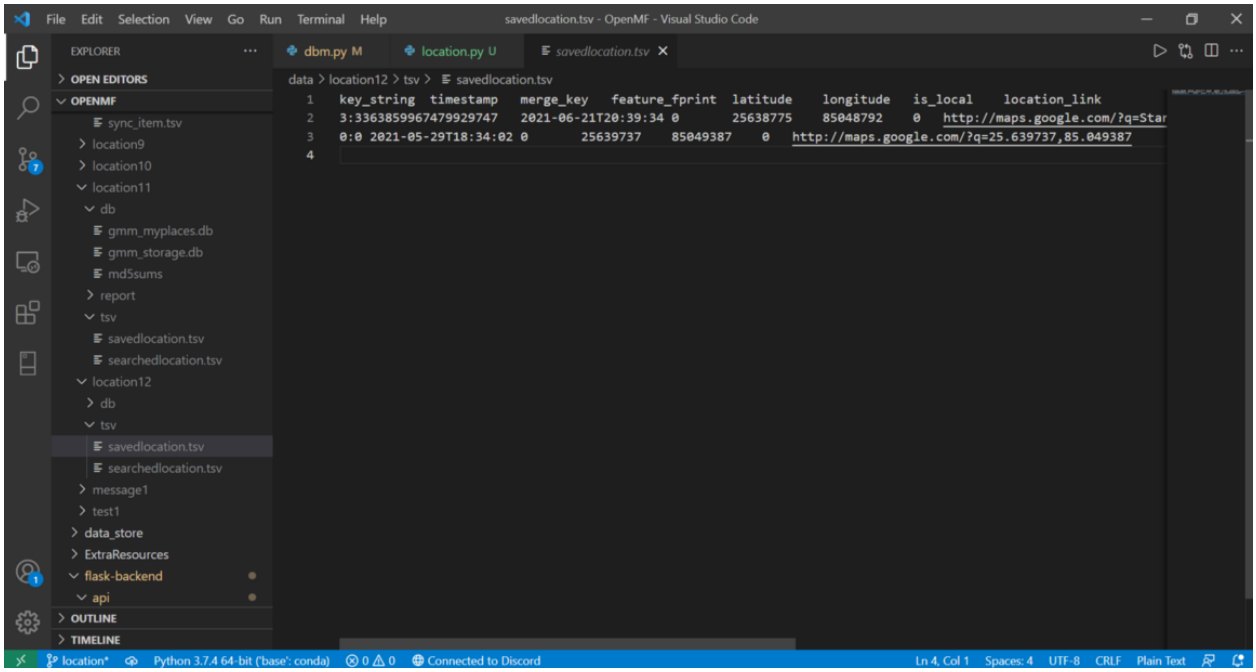
(21 June - 27 June)

Work on the extraction of EXIF data from images and location data from an android device. EXIF stands for Exchangeable image file format and refers to the basic metadata that is generated and stored by a camera or device whenever you take a photo. In simple words whenever we click a photo some data get embedded within our image and include pieces of information like date when it was taken, date if modified with any software, software id, a location where the pic was clicked, time, description, bucket id, bucket name, album, tags, width, height and many more.



Data from Images (above)

Later work on the location part to get all the saved locations and searched locations. Create APIs for the above and implement from CMD and POSTMAN both.

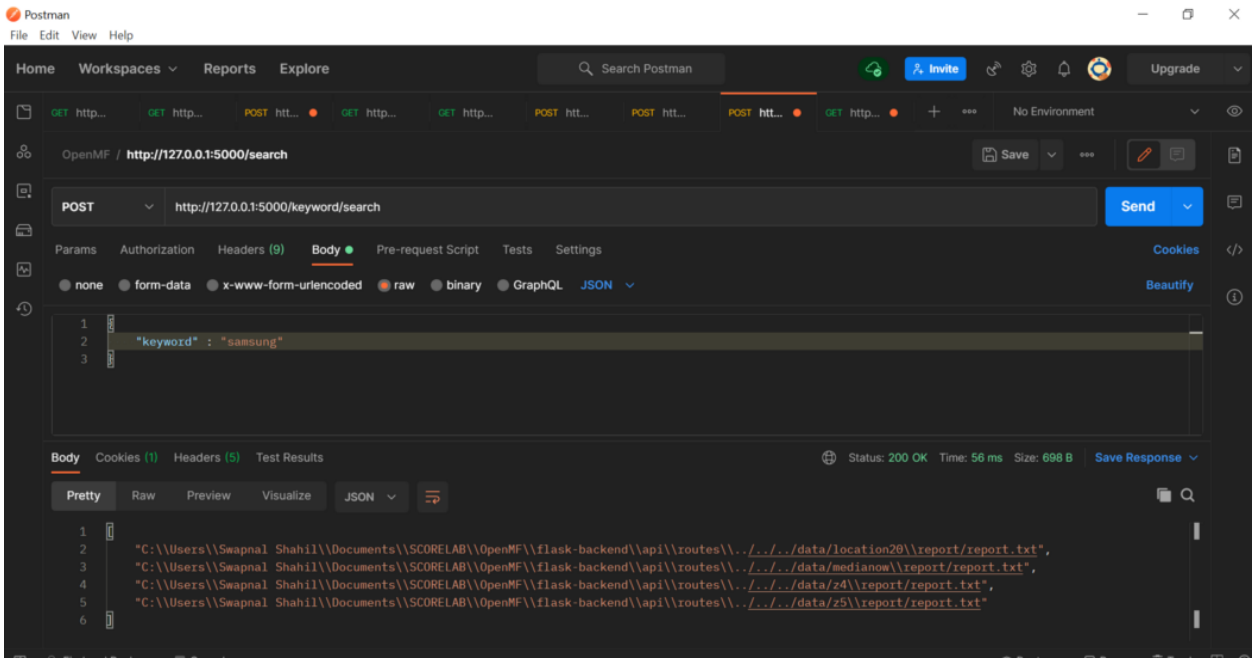


Saved location Data (above)

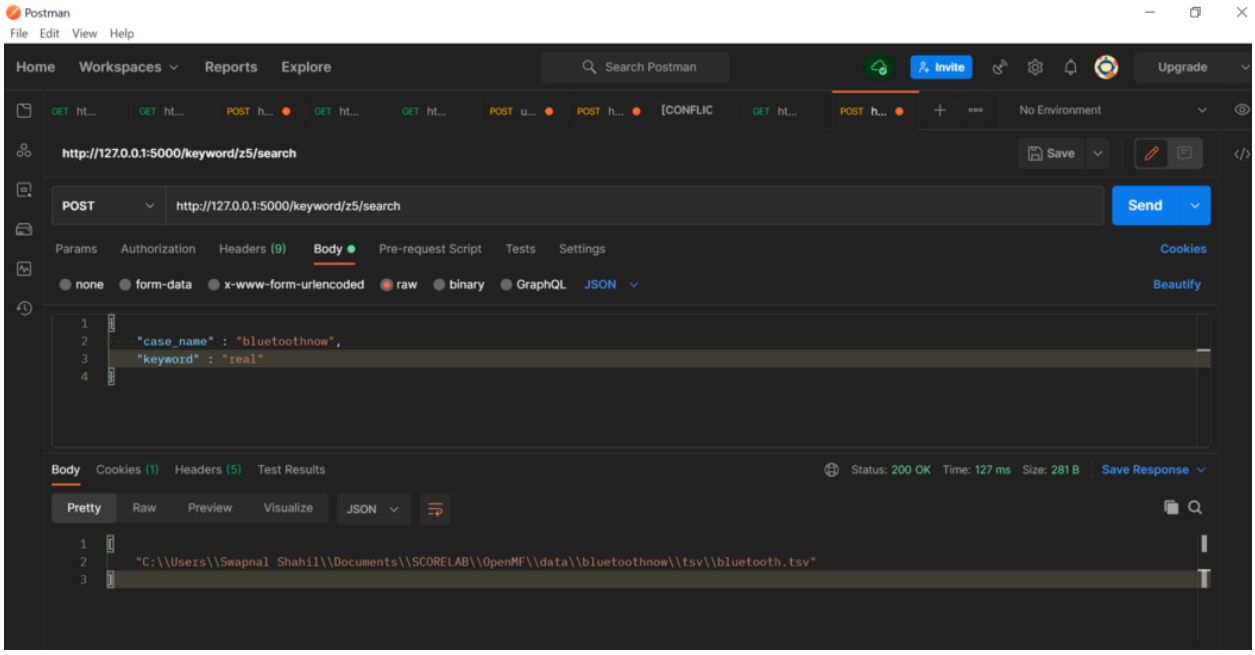
(28 June - 4 July)

Work on keyword search to get all the cases that have the searched keyword. Basically main purpose for this was to find a word from the database if the admin wants to search while

analyzing the mobile forensic case. For this, make two APIs that have some similar kind of function but have different use in their own place. One API search the keyword from the whole database so that analyst can relate one case with other cases and another API works within the case to get the files which have that keyword.



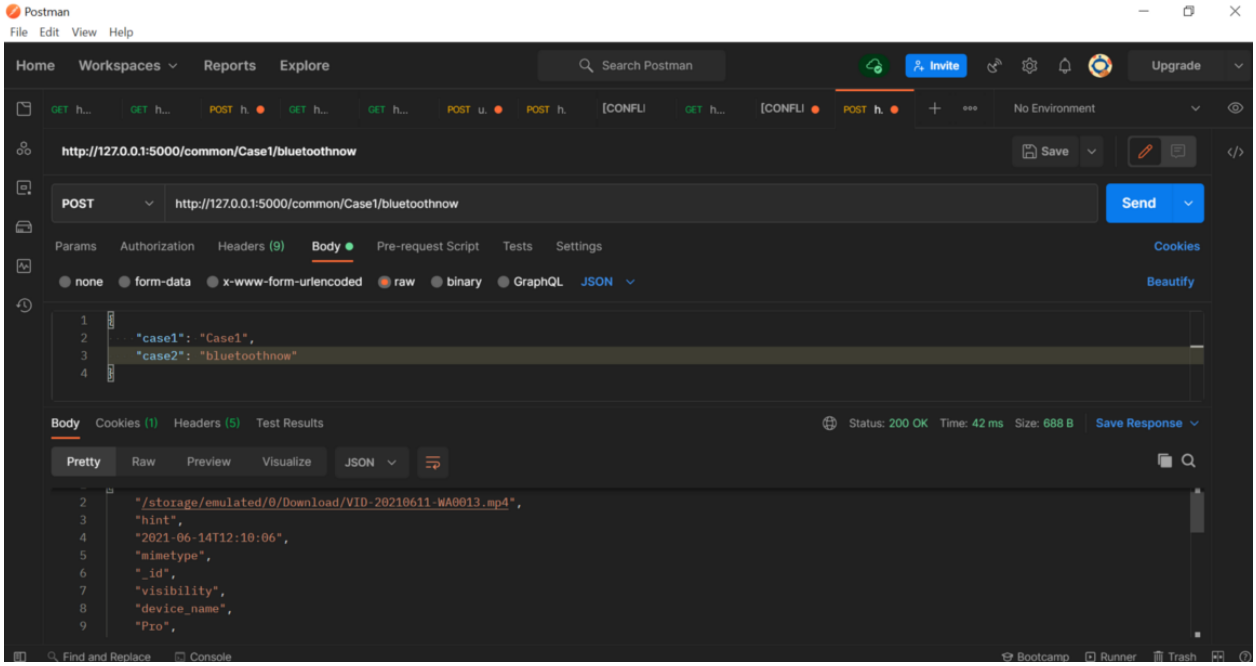
Keyword search from the whole database (above)



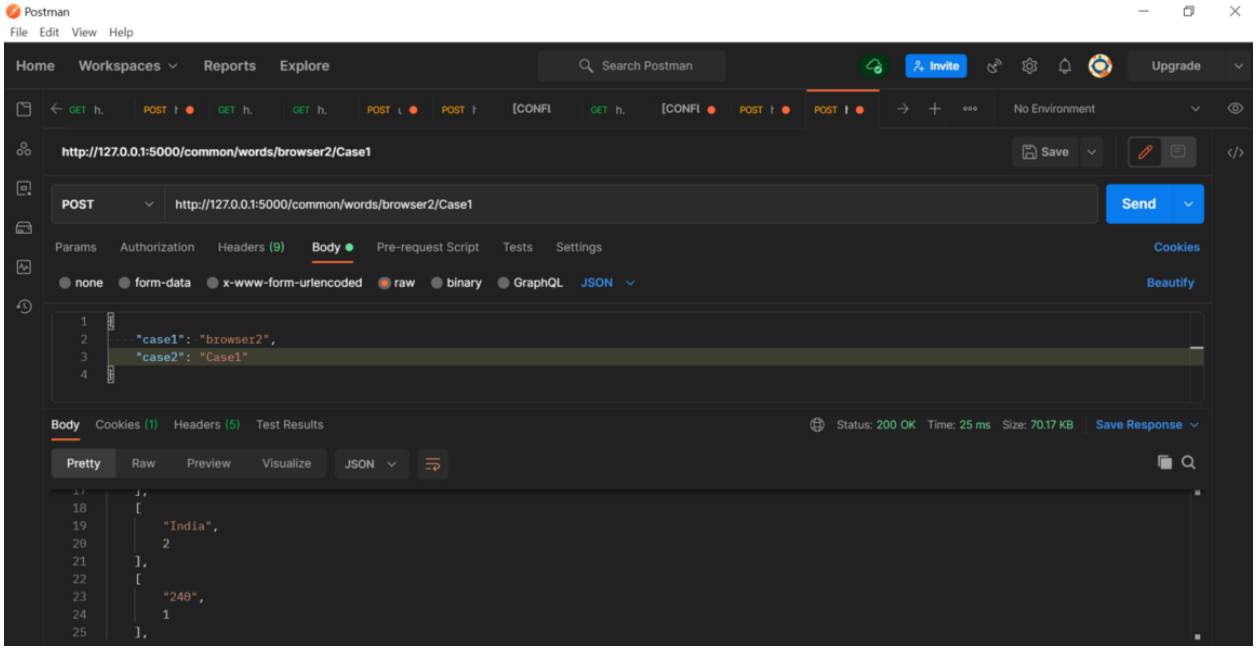
Keyword search from a particular case (above)

(5 July - 11 July)

Work on the analytics part of this project and create APIs respectively to find common words between the two cases and the maximum word common between the cases with their frequencies.



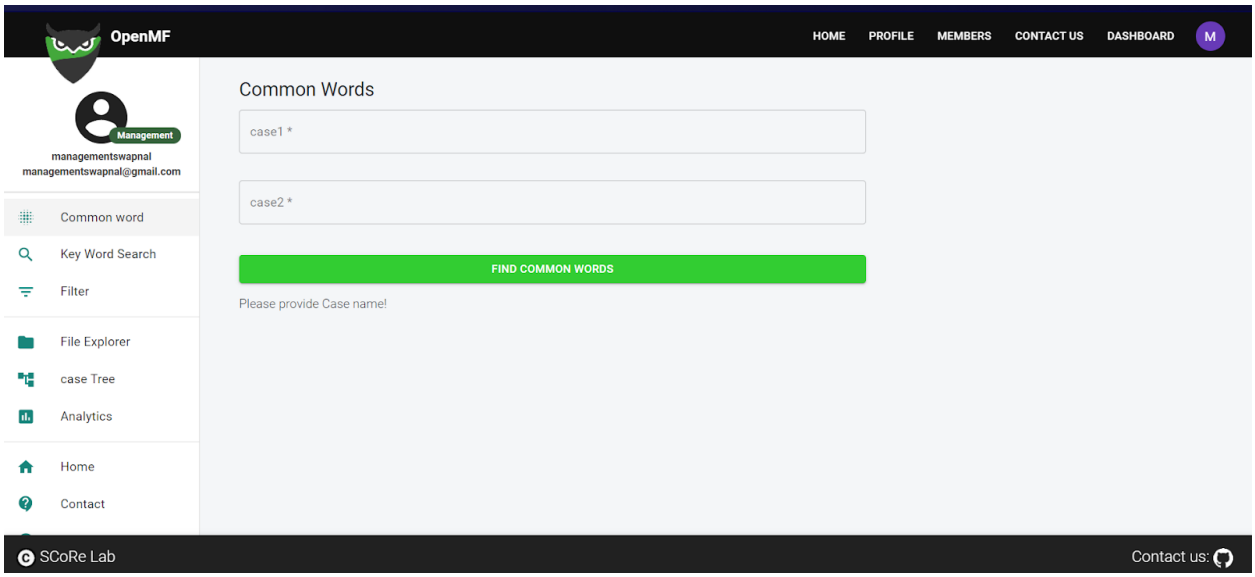
Common words between two cases.(above)



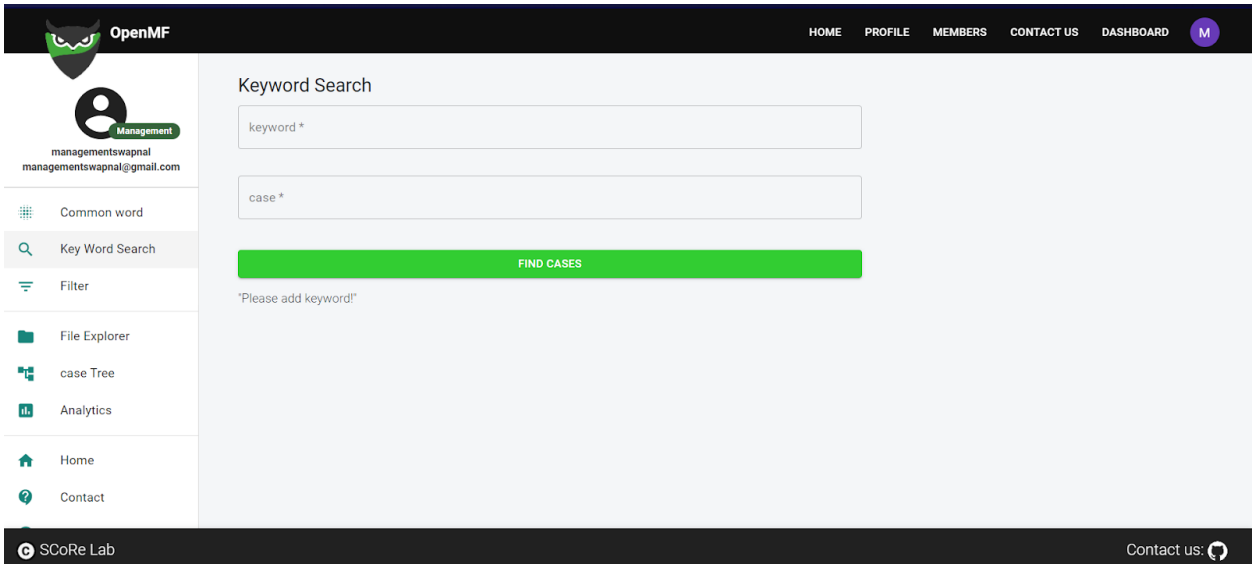
Maximum common word with frequency.

(12 July - 18 July)

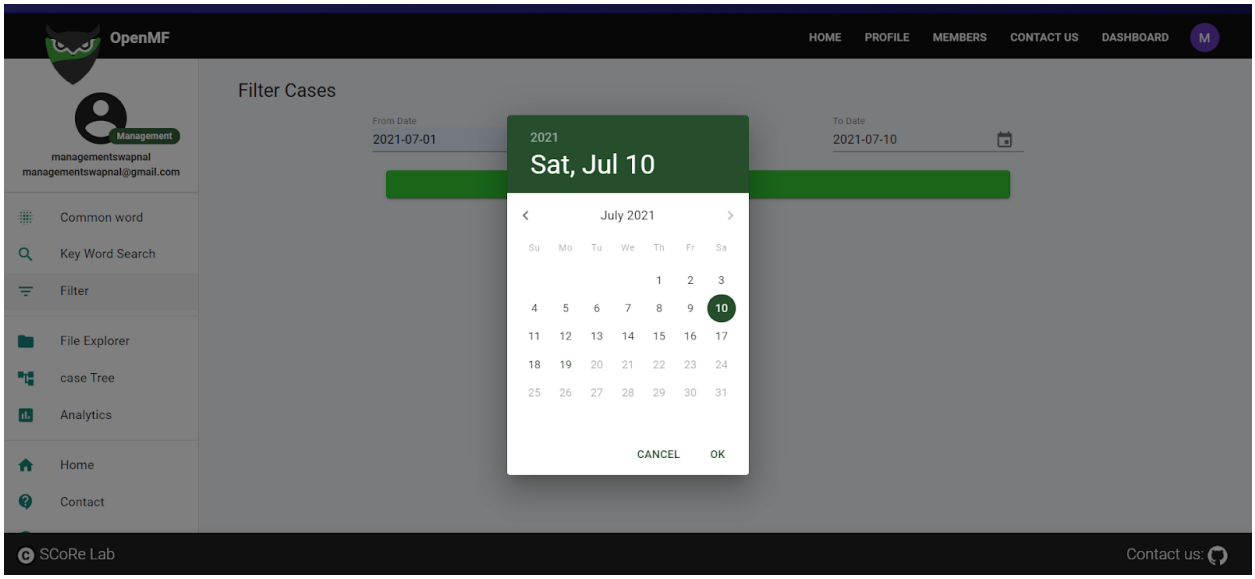
Work on the analytics part to filter the cases by creating API and implementing third, fourth, fifth, and sixth-week work on the OpenMF front end. Use Redux for state management



Common Word Page (above)



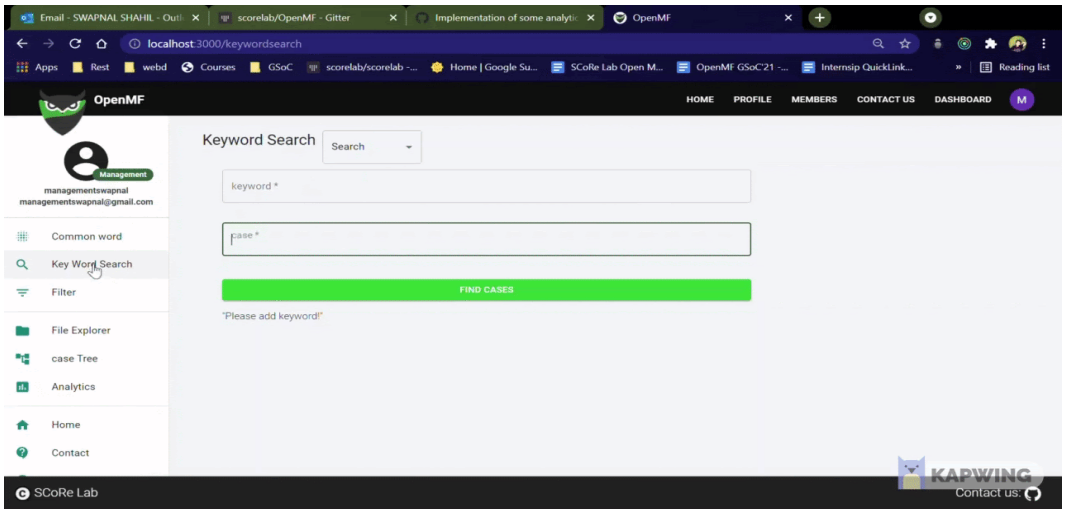
Keyword Search Page (above)

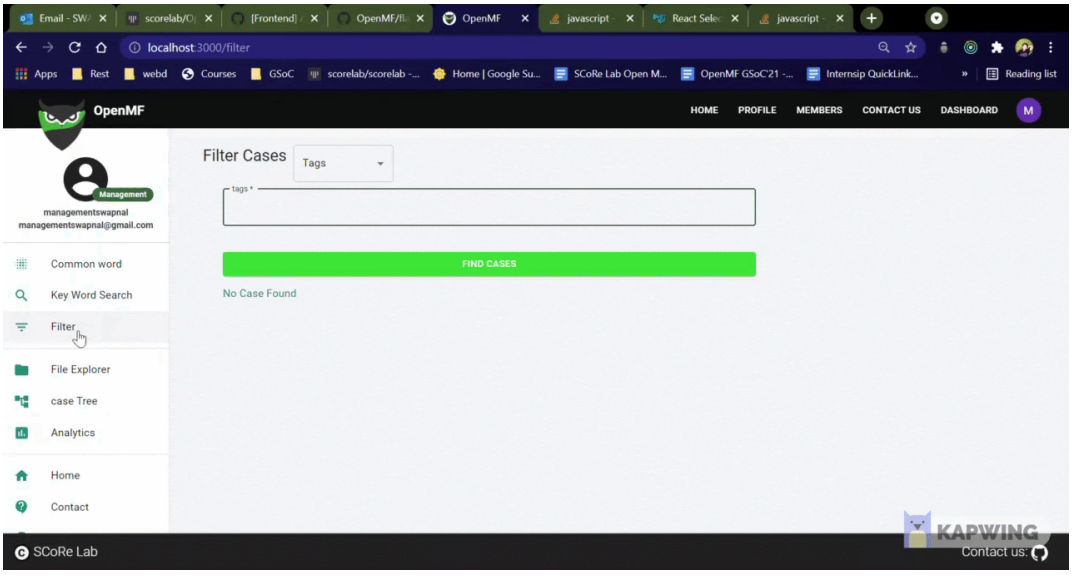


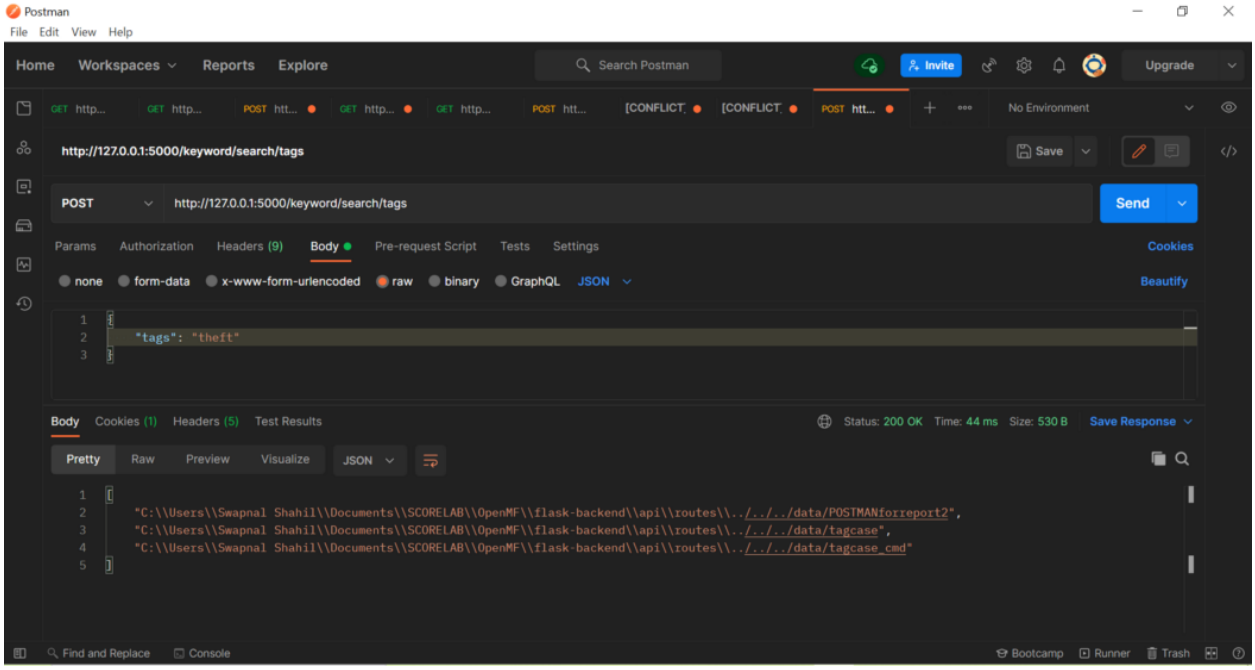
Filter Cases (above)

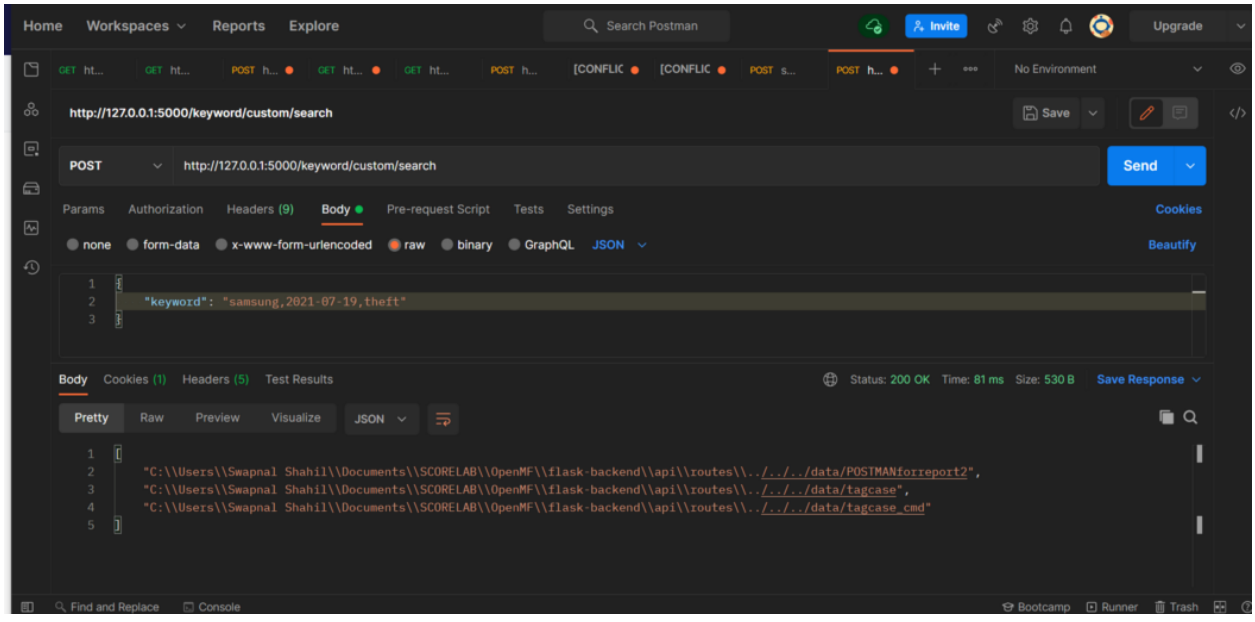
(19 July - 25 July)

Worke on “Tag association” and tag-based search like searching cases that are tagged with theft to get all the theft-related cases and on customized keyword search.



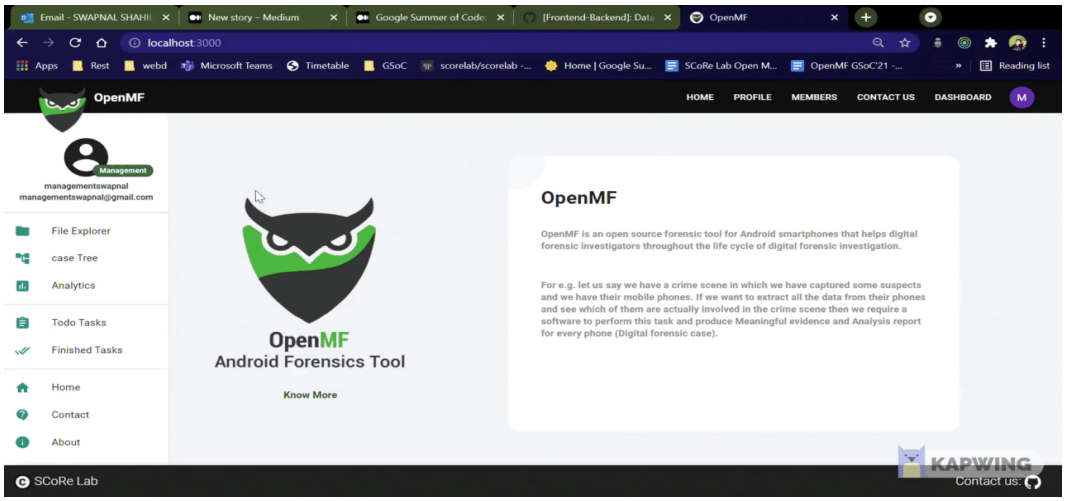


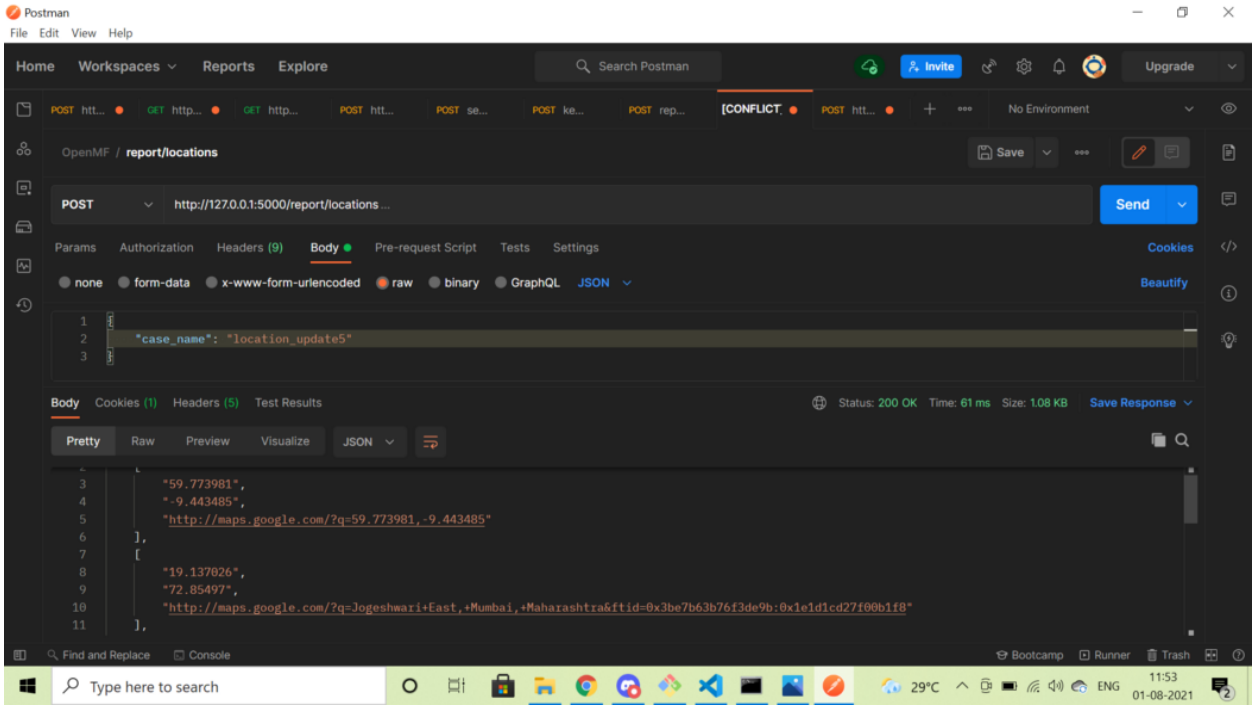

Postman
File Edit View Help
Home Workspaces Reports Explore
Search Postman
Invite
Upgrade
No Environment
http://127.0.0.1:5000/keyword/search/tags
Save
POST
http://127.0.0.1:5000/keyword/search/tags
Send
Params Authorization Headers (9) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"tags": "theft"
}
Body Cookies (1) Headers (5) Test Results
Status: 200 OK Time: 44 ms Size: 530 B
Save Response
Pretty Raw Preview Visualize JSON
[
"C:\\Users\\Swapnal Shahil\\Documents\\SCORELAB\\OpenMF\\flask-backend\\api\\routes\\../../../data/POSTMANforreport2",
"C:\\Users\\Swapnal Shahil\\Documents\\SCORELAB\\OpenMF\\flask-backend\\api\\routes\\../../../data/tagcase",
"C:\\Users\\Swapnal Shahil\\Documents\\SCORELAB\\OpenMF\\flask-backend\\api\\routes\\../../../data/tagcase_cmd"
]
Find and Replace
Console
Bootcamp
Runner
Trash

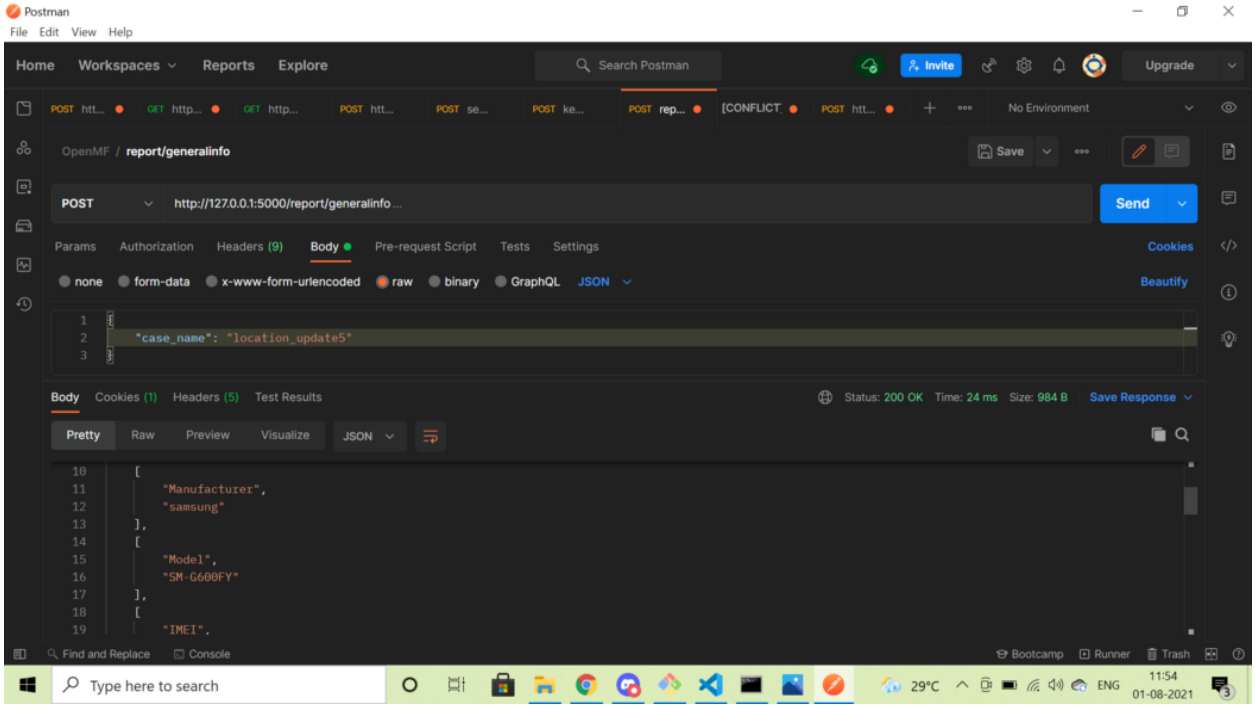

Home Workspaces Reports Explore
Search Postman
Invite
Upgrade
No Environment
http://127.0.0.1:5000/keyword/custom/search
Save
POST
http://127.0.0.1:5000/keyword/custom/search
Send
Params Authorization Headers (9) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"keyword": "samsung,2021-07-19,theft"
}
Body Cookies (1) Headers (5) Test Results
Status: 200 OK Time: 81 ms Size: 530 B
Save Response
Pretty Raw Preview Visualize JSON
[
"C:\\Users\\Swapnal Shahil\\Documents\\SCORELAB\\OpenMF\\flask-backend\\api\\routes\\../../../data/POSTMANforreport2",
"C:\\Users\\Swapnal Shahil\\Documents\\SCORELAB\\OpenMF\\flask-backend\\api\\routes\\../../../data/tagcase",
"C:\\Users\\Swapnal Shahil\\Documents\\SCORELAB\\OpenMF\\flask-backend\\api\\routes\\../../../data/tagcase_cmd"
]
Find and Replace
Console
Bootcamp
Runner
Trash

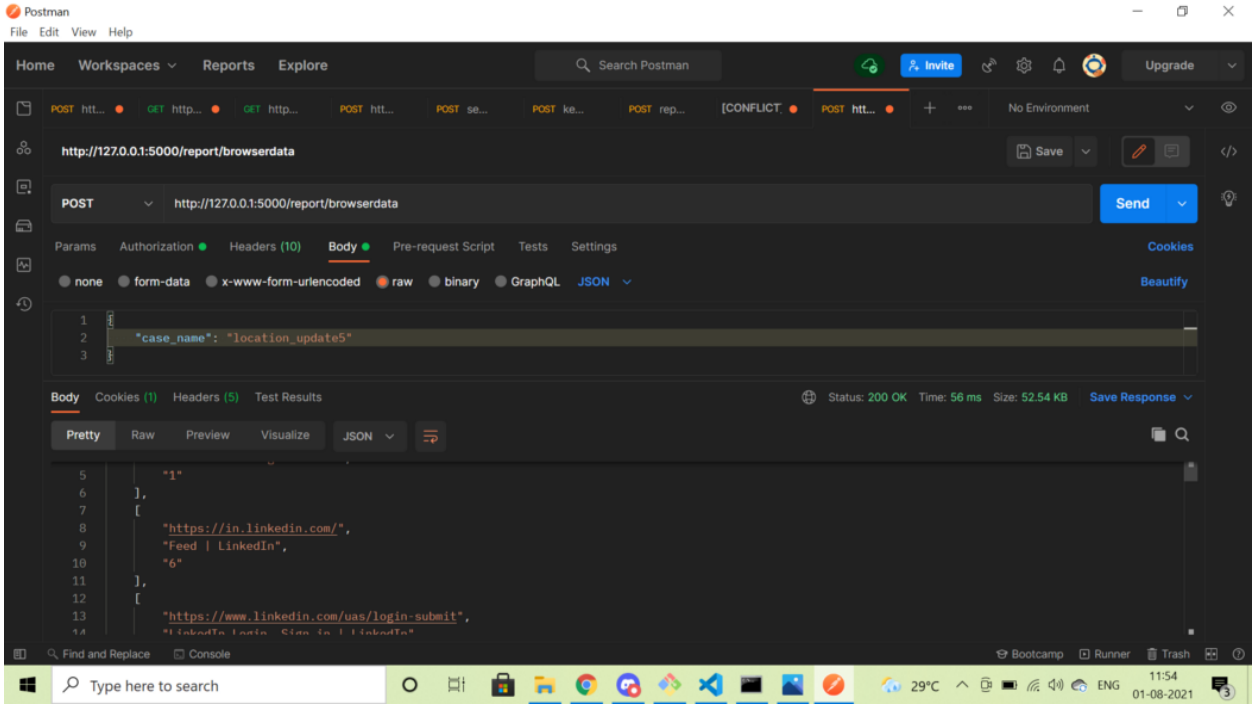
(26 July - 1 Aug)

Worke on Data Visualization in the OpenMF frontend and try to present observed analytics through graphs, tables, maps of each case in the report section of OpenMF Analytics.






Postman
File Edit View Help
Home Workspaces Reports Explore
Search Postman
Invite
Upgrade
No Environment
OpenMF / report/generalinfo
Save
POST
http://127.0.0.1:5000/report/generalinfo
Send
Params Authorization Headers (9) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"case_name": "location_update5"
}
Body Cookies (1) Headers (5) Test Results
Status: 200 OK Time: 24 ms Size: 984 B
Save Response
Pretty Raw Preview Visualize JSON
[
"Manufacturer",
"samsung"
],
[
"Model",
"SM-G600FY"
],
[
"IMEI",
Find and Replace Console
Bootcamp Runner Trash
Type here to search
29°C
ENG
11:54
01-08-2021


Postman
File Edit View Help
Home Workspaces Reports Explore
Search Postman
Invite
Upgrade
No Environment
http://127.0.0.1:5000/report/browserdata
Save
POST
http://127.0.0.1:5000/report/browserdata
Send
Params Authorization Headers (10) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"case_name": "location_update5"
}
Body Cookies (1) Headers (5) Test Results
Status: 200 OK Time: 56 ms Size: 52.54 KB
Save Response
Pretty Raw Preview Visualize JSON
"1"
],
[
"https://in.linkedin.com/",
"Feed | LinkedIn",
"6"
],
[
"https://www.linkedin.com/uas/login-submit",
Find and Replace Console
Bootcamp Runner Trash
Type here to search
29°C
ENG
11:54
01-08-2021

SCoRe Lab Contributions:

I have been contributing to this organization since last year and will continue as per need.

- [Merged] https://github.com/scorelab/OpenMF/pull/218
  “Data Visualization of Cases”
- [Merged] https://github.com/scorelab/OpenMF/pull/210
  “Implementation of some analytics API on front end OpenMF”
- [Merged] https://github.com/scorelab/OpenMF/pull/207
  “Create Filter API to get Cases by Date or range of date.”
- [Merged] https://github.com/scorelab/OpenMF/pull/200
  “List all common word/data between two cases”
- [Merged] https://github.com/scorelab/OpenMF/pull/195
  “Keyword Search”
- [Merged] https://github.com/scorelab/OpenMF/pull/192
  “Extraction of images data”
- [Merged] https://github.com/scorelab/OpenMF/pull/182
  “Extract Bluetooth data”
- [Merged] https://github.com/scorelab/OpenMF/pull/181
  “Extraction of browser history”
- [Merged] https://github.com/scorelab/OpenMF/pull/175
  “Extraction of text messages(SMS)”
- [Merged] https://github.com/scorelab/OpenMF/pull/47
  "Frontend documentation #47"

- [Merged]https://github.com/scorelab/OpenMF/pull/44
  "fixed typo in README.md #44"
- [Open]https://github.com/scorelab/OpenMF/pull/157
  "[Frontend] enabled lazy loading #157"
- [Open]https://github.com/scorelab/OpenMF/pull/117
  "[frontend] Know more button fixed #117"
- [Issue]https://github.com/scorelab/OpenMF/issues/46
  “Documentation needs update! #46”
- [Issue]https://github.com/scorelab/OpenMF/issues/116
  “Know More Button not working in login page #116”
- [Issue]https://github.com/scorelab/OpenMF/issues/174
  “Extraction of text messages”
- [Issue] https://github.com/scorelab/OpenMF/issues/177
  “Extraction of Bluetooth data”
- [Issue] https://github.com/scorelab/OpenMF/issues/178
  “Extraction of browser history”
- [Issue] https://github.com/scorelab/OpenMF/issues/187
  “Extraction of EXIF data from images (like location, date, time, etc.)”
- [Issue] https://github.com/scorelab/OpenMF/issues/186
  “Extraction of location data from the android device”
- [Issue] https://github.com/scorelab/OpenMF/issues/170
  “Extract all data from rooted Android device not working”
- [Issue] https://github.com/scorelab/OpenMF/issues/193
  “Keyword search (for getting cases)”
- [Issue] https://github.com/scorelab/OpenMF/issues/194
  “Keyword search from cases”

- [Issue] https://github.com/scorelab/OpenMF/issues/198
  "Maximum common word/data between two cases."
- [Issue] https://github.com/scorelab/OpenMF/issues/199
  "List all common word/data between two cases."
- [Issue] https://github.com/scorelab/OpenMF/issues/201
  "Add comments in the worked file"
- [Issue] https://github.com/scorelab/OpenMF/issues/205
  "Create Filter API to get Cases by Date or range of date."
- [Issue] https://github.com/scorelab/OpenMF/issues/206
  "[Frontend]: Implementation on analytics section in management."
- [Issue] https://github.com/scorelab/OpenMF/issues/208
  "Bug in API for getting list of cases"
- [Issue] https://github.com/scorelab/OpenMF/issues/211
  "Tag association"
- [Issue] https://github.com/scorelab/OpenMF/issues/212
  "Customized Keyword search"
- [Issue] https://github.com/scorelab/OpenMF/issues/214
  "Report.txt did not get store when extracting from POSTMAN"
- [Issue] https://github.com/scorelab/OpenMF/issues/217
  "Data visualization"

Personal Information:

- Name: Swapnal Shahil
- Email: swapnalsahil@gmail.com
- GitHub: swapnalshahil
- LinkedIn: swapnalshahil

- Twitter: eulersgamma
- Duo: swapnalsahil@gmail.com
- Gitter nickname: swapnalshahil
- First language: Hindi, proficient in English
- Time zone: Indian Standard Time (GMT +5:30)
- Contact: +91 9205711402
- University: Indian Institute of Technology, Guwahati
- Year of study: 2nd year B. Tech.
- Major: Chemical Science and Technology (Batch of 2023)

Reference:

- /scorelab/OpenMF/

Questions

1. **Are you a SCoRe contributor/ Have you contributed to SCoRe before?**
   Yes, I have contributed to SCoRe Lab Organization since December 2020. Yes, My contributions are in **OpenMF** and **Fact-Bounty** projects
2. **How can we reach you (eg: email) if we have questions about your application?**
   Email: swapnalsahil@gmail.com
   Gitter: swapnalshahil
   Mob: 9205711402
3. **What is your GitHub username(s):**
   swapnalshahil

Project Specific Questions

4. **Which SCoRe GSoC project are you applying for (please submit separate applications for each project):**

5. What do you plan to accomplish over this summer for this project?

- Generate dummy dataset from a rooted device.
- Extraction of the text messages.
- Extraction of browser history.
- Extraction of Bluetooth files.
- Extracting data (EXIF) from images(like location, date, and time)
- Extraction of location from the device.
- Keyword search - Implementation of keyword search to get all the cases that have the searched keyword.
- Implementing keyword search from cases.
- Maximum occurred common word/data between two cases.
- Returning all the common words between two cases.
- Filtering Cases by Date or range of date.
- Creating APIs for the above.
- Tag association and tag-based search like searching cases that are tagged with theft to get all the theft-related cases.
- Customized keyword search <kw1> + <kw2> + … + <kwN>

- Analytics based on location.
- Implementing analytics on the front end.
- Presentation of observed analytics - graph, library, an API will be developed which will send the database data to the graph generating library to create required graphs and that will show in our OpenMF frontend.
- Presentation of observed analytics between two cases using graphs and tables.
- Writing Wiki and blogs.

6. **If you have your own project to propose, please describe it here:**
   Not now but maybe in the future.
7. **List down any plans you have during this summer.**
   I don't have any other plan for this summer, so I can give my whole day for this project.
8. **Education:**
   - **University:** Indian Institute of Technology Guwahati
   - Department: Chemical Science and Technology
   - Current year of study: 2nd year of B.Tech
   - Programming courses:
     - Object-Oriented Programming
     - Data Structures and Algorithm
     - Interactivity with JavaScript
     - ReactJs
     - Python and C++ Programming
     - Git and Github
9. **Do you have work experience in programming?**
   Yes, I did some projects of mine till now but contributed to many organizations on regular basis. One of my projects (project-corona)

shows the current number of COVID19 { active cases, death count, recovered count } using bar graph of all countries.

10. **Do you have previous open source experience? Briefly describe what you have done.**
    From my 1st year of my B. Tech I am involved with open-source. Also, I participated in Hactoberfest and able to create more than 10 valuable PRs. In my free time, I contribute to reactjs.org (Facebook React Community).
11. **Tell one interesting fact about yourself.**
    I am a very social person and I love photography.