FINANCIAL COMPLAINTS REDRESS SYSTEM

The Deciders

May 2023

INDEX

Introduction

The Financial Complaints Redress System is a data-intensive application designed to analyze and resolve financial complaints against various companies. Inspired by the recent events of multiple firms facing financial complaints, we built a system capable of handling a large volume of complaints, tracking their status, and providing insights to the stakeholders. The system has two main functionalities - operation and analytics. The operation primarily focuses on receiving and tracking complaints, while analytics assists in examining the data in real time, revealing significant trends, and aiding in strategic decision-making.

Data Source

Source link: Consumer Complaint Database.
Mockaroo:- https://www.mockaroo.com/ (For names and addresses)
Python Packages for random number generators.

Application Design

Developing tools include: Python, MySQL Workbench, PyQt Designer.
It requires additional packages to install:-

1. Pandas
2. Numpy
3. Matplotlib
4. Mplcusror

The application consists of two main modules: Operation Module and Analytical Module.

Operation Module

Customers can register with this part of the system, log in, submit complaints against different financial institutions, and then track the status of their complaints by using this part of the system. If a complaint is filed against a company, the customer may include information such as the type of product, the issue, and the company against which the complaint is filed. In order to track the complaint, the system validates these details and creates a unique complaint ID.

Analytical Module

As part of this module, we will analyze the data that has been collected from the operation module and the government websites. In the dashboard you can find information on the performance of different companies, the number of complaints totaled and resolved, as well as various trends determined by the time, place, and type of issue. There is also forecasting, and comparative analysis performed between companies as part of the system. The administrative feature also enables the data flow from operational tables to be retrieved in real-time for the purpose of making urgent decisions based on real-time data.

Database Design

During the design process, a picture of normalized tables has been kept in mind, as well as the integrity constraints associated with all the tables. Creating a product with this system was effective due to its simplicity and effectiveness. Normalization of the database ensured that data was stored efficiently, and redundancies were eliminated. This resulted in a lightweight design that was easy to maintain and update. This allowed for a more efficient and reliable product.

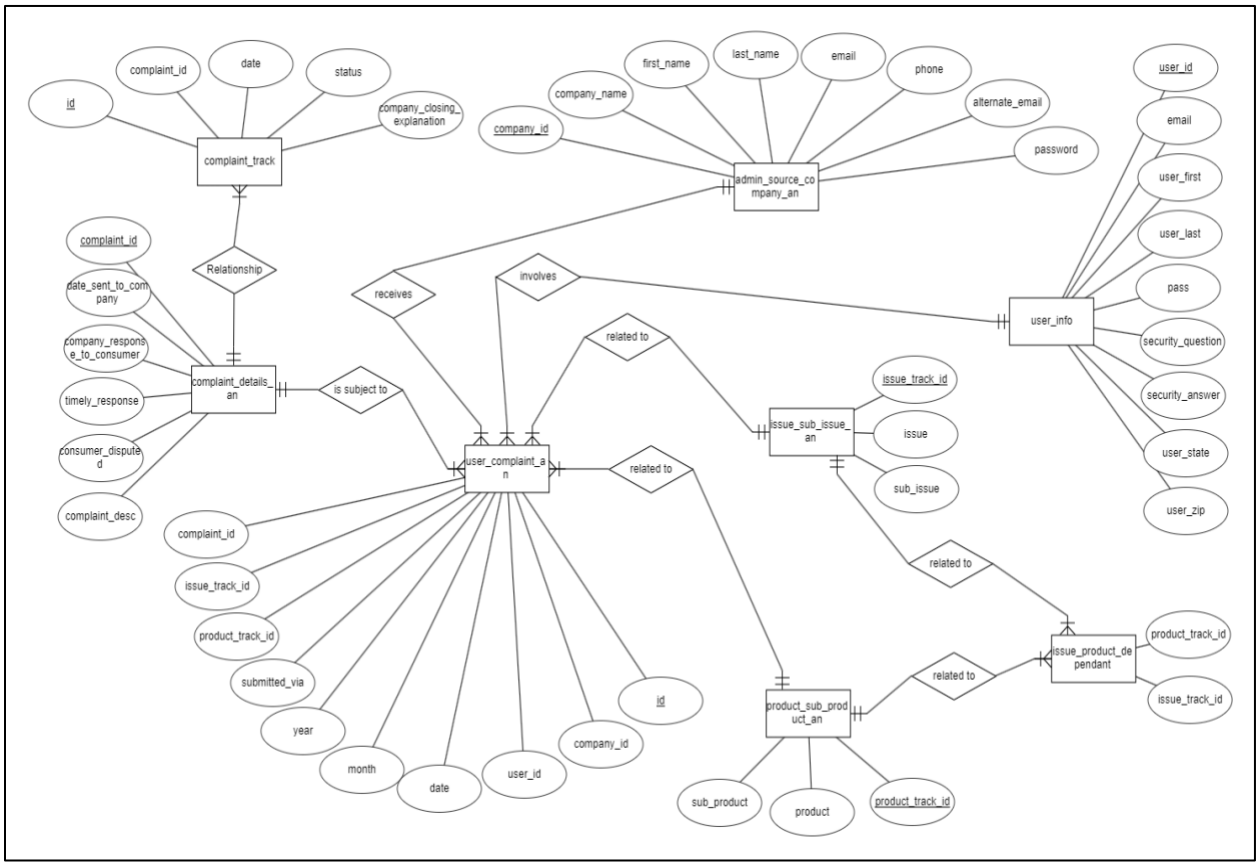


Fig 1. ER Diagram

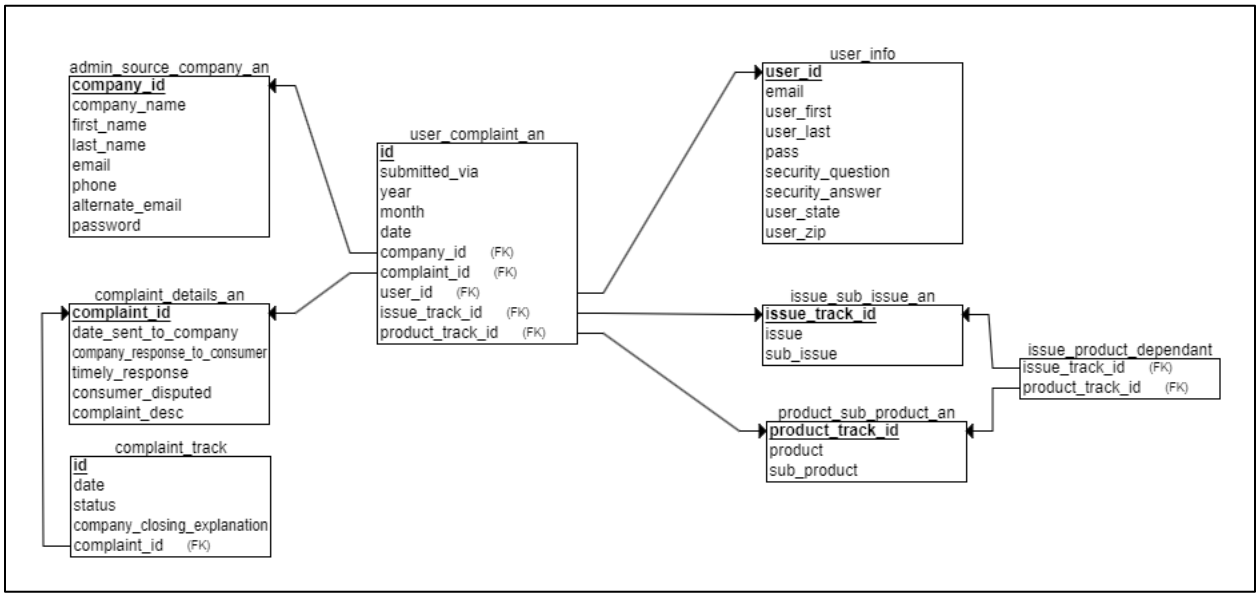


Fig 2. Relational Schema

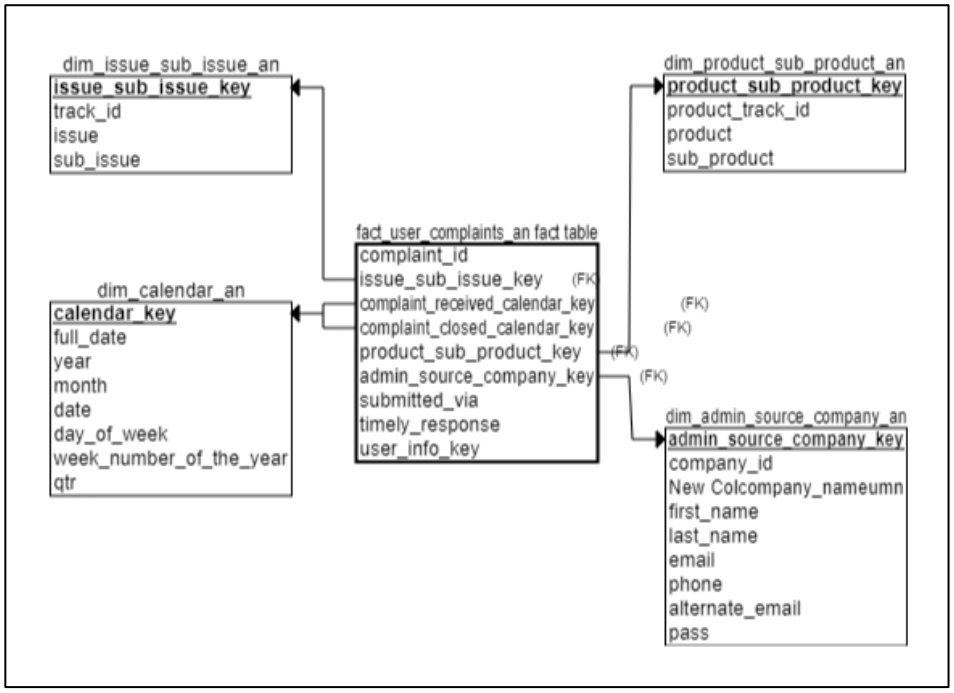


Fig 3. Star Schema

Working of the Operational module

The operational aspect revolves around 2 primary roles: the ***User*** and the ***Company Representative***. Each role has distinct features. For instance, the user can log in or create a new account and submit a complaint against the company, while the company representative's role involves addressing the complaint raised by users. Within our database, users can lodge complaints against companies. To initiate a complaint, certain factors must be taken into account, such as the specific product and sub-product involved, as well as the main issue and corresponding sub-issue. There exists a relationship between the product and the issue, which enables the complaint process.

Once all necessary information is provided, users are directed to a review page. A unique complaint ID is generated and assigned to the user, which serves as a crucial reference for tracking the progress of the complaint.

For company representatives, they have the ability to monitor and manage complaints. They can track the status of each complaint and have the option to shift its status to different stages, such as "working," "in progress," or "closed." This allows them to effectively address and resolve the raised complaints.

Specifications and Usability of Operational Module

1. Login page

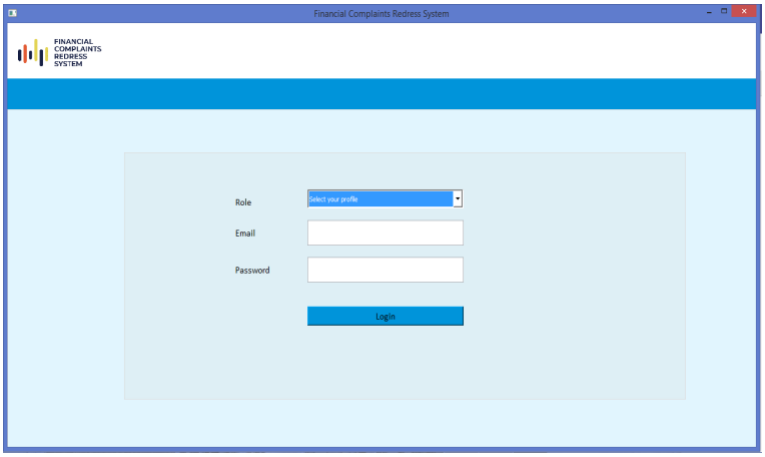


Fig 4. Login Page

- The application has login page from where we can login with the username and password. We can create a user account if you are new user.
- There are two roles:
- **User** - User can raise complaint and track complaint.
- **Company Representative** - can see the complaints raised under his company and change status of complaints.

2. Creating a New Account

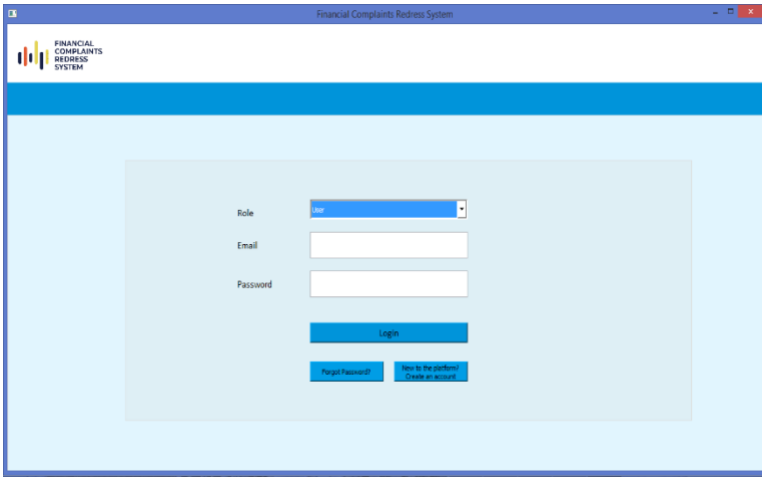


Fig 5. User Selection

- After selecting the user role, three options appear on the portal. Out of which one of them is '*Create an account*'.
- This option allows the user to establish a personal account.

3.Validation on confirming passwords

Fig 6. Password Validation

- During the account creation, user is required to provide mandatory details including full name, user id, email id, city, and zip code. If any field is left empty, the user is prompted to fill those details.
- Password validation ensures that both password and confirm password matches.
- User is also asked to select a security question and provide an answer for added security.
- Duplicate email id and user id is detected to avoid account duplication.
- Once the user clicks "submit," their account is successfully created, and they can set their password.

4. Validation if the user already exists

Fig 7. Password Validation

- When any user tries to create an account with already used email id or user id, the account will not be created
- User will also get a prompt on the portal indicating the user id or email id is used

5. Successful creation of account

Fig 8. Successful Account Creation

- When all user details are validated, the account will be created successfully
- User will also get a prompt on the portal indicating the same

6. Forgot Password

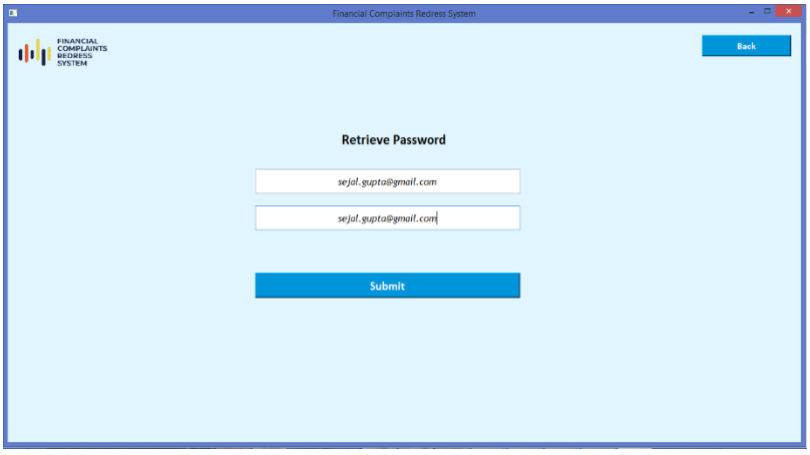


Fig 9. Forgot the password

- After selecting the user role, three options appear on the portal. Out of which one of them is '*Forgot Password'*
- This option allows the user to retrieve the forgotten password
- The user is required to enter both user id and email id linked to his/her account

7. *Resetting the password after the validation*

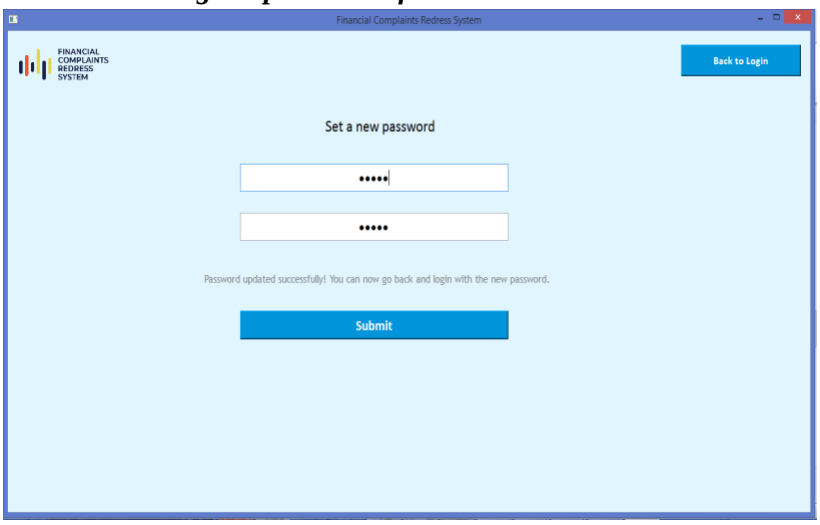


Fig 10. Successful resetting of password

- Once the user enters correct answer to the security question, the user will then be asked to provide the new password
- Password validation ensures both the password and confirm password are the same
- User will be prompted if both passwords does not match
- Once both passwords are validated, user will be promoted about the successfully resetting of the password

8. *Successfully logged in as a user*

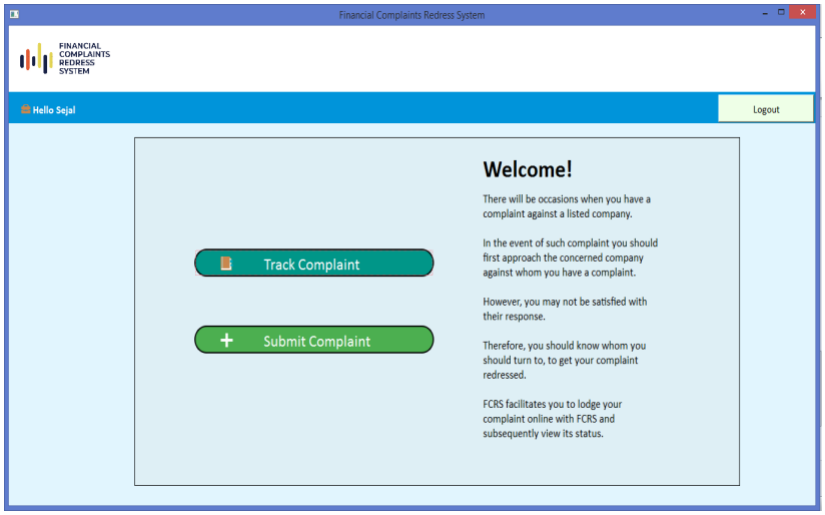


Fig 11. Successfully logged in as a user

- The user can perform two operations:
- Raise a complaint by clicking on **'submit complaint'** button.
- Track the status of complaints by clicking on '**Track Complaint'** button.

9. ***Submit Complaint***: There are five steps to raise a complaint. All are mandatory. We cannot go to the next step without filling in all the details. There would be a message box alert raised if we do so.

Fig 12. Step 1

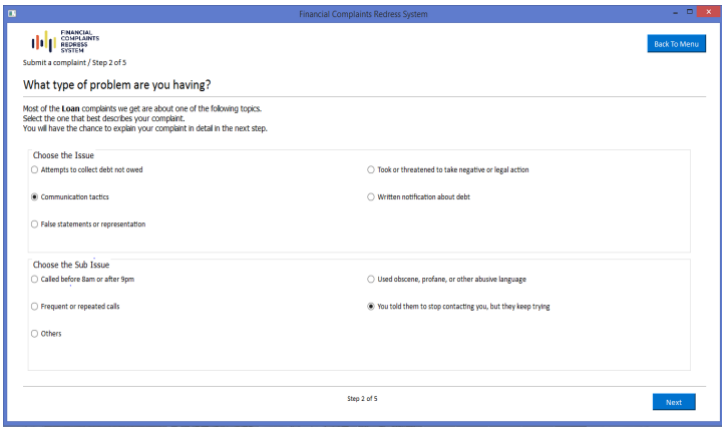


Fig 13. Step 2

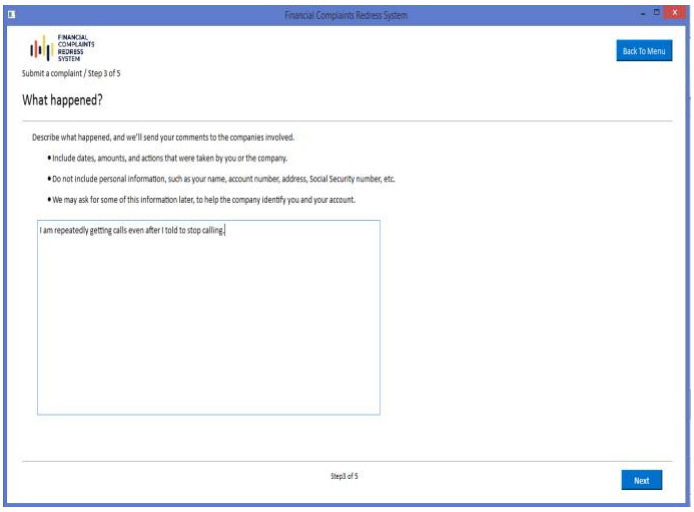


Fig 14. Step 3

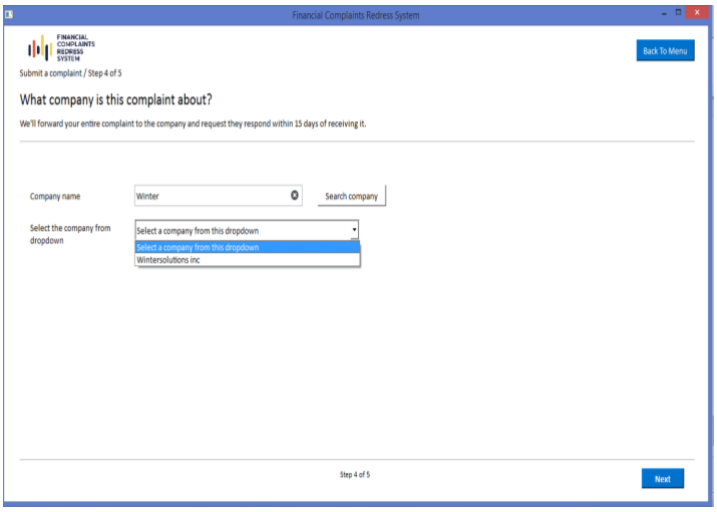


Fig 15. Step 4

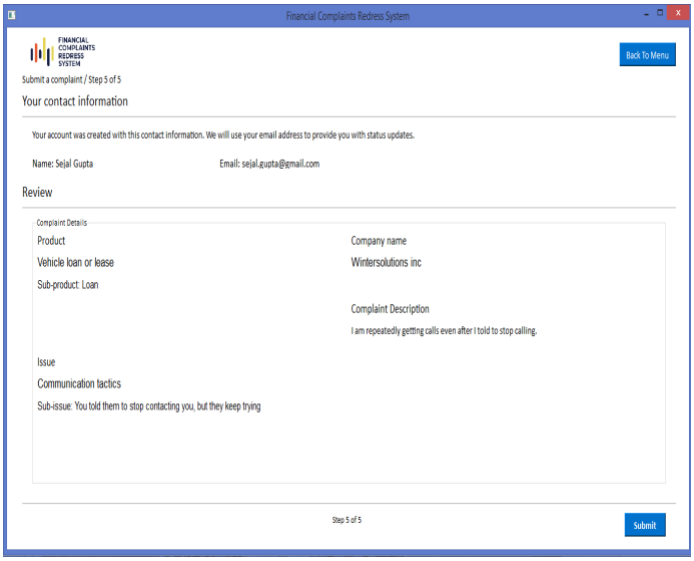


Fig 15. Step 5

- After selecting of all the relevant factors regarding the complaints from Step 1- 4. We get to review our complaint details at the 5th Step.
- Once we confirm the details we can submit, and the complaint will be raised.
- Sample input: Wintersolutions inc

10. *Complaint Progress Tracking for existing users*

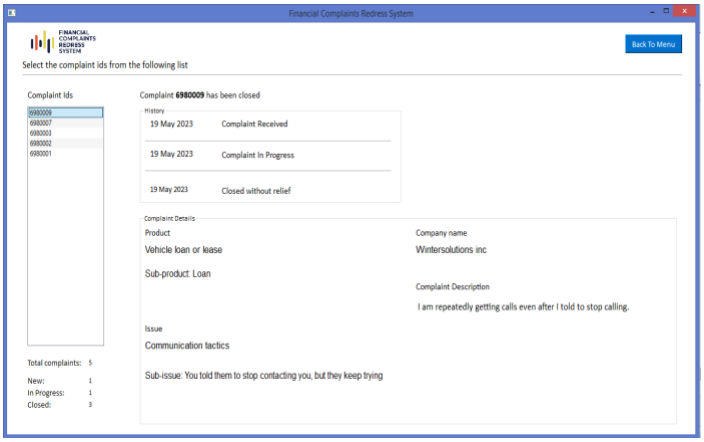


Fig 16. Track User Complaint

- After user logs in and track the complaint, the user can see all the complaints that are raised and what is the status of the complaints.
- This assures that the complaints maintenance and builds the trust on the companies

11. Login as Company Representative

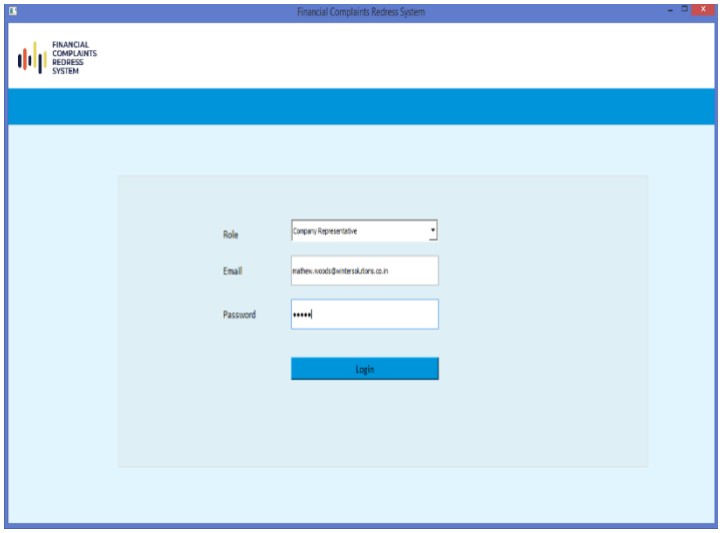


Fig 17. Company Representative login

- As complaints can be raised against the companies our product offers companies to look at the complaints take actions and based on type of the complaint.
- If the complaint is solved within a time frame of 1-15 days, we mark it as a timely response or else delayed.

Company Representative role:

Email: mathew.woods@wintersolutions.co.in

Password: Comp@1234

12. Listing of complaints along with status

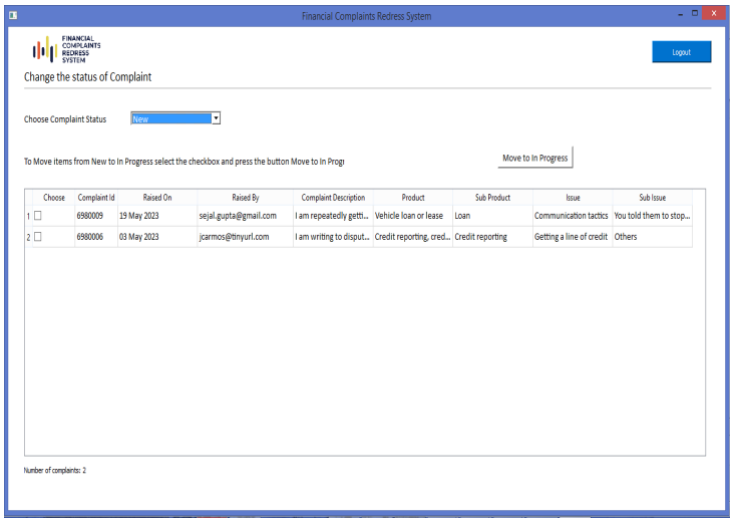


Fig 18. Newly Added Complaints

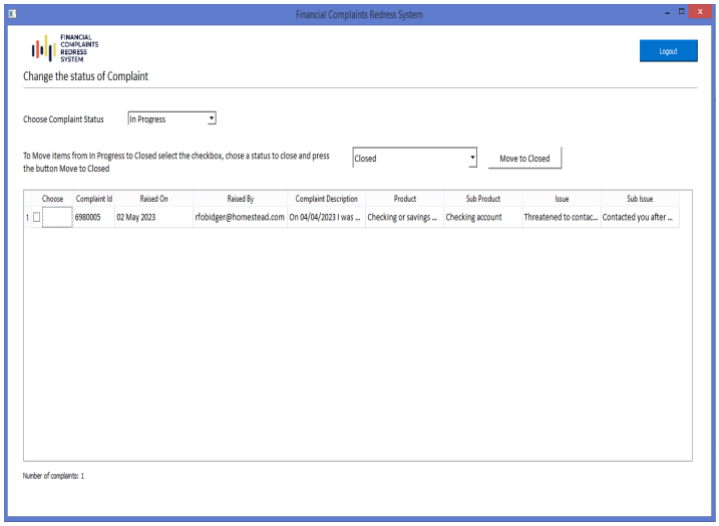


Fig 19. In Progress Complaints

- The above example is demonstrated with complaint-id ‘6980009’.
- When new complaints arrive, the company end the company representative has the option of moving to In-Progress.
- In a similar way, the In-Progress listing of the complaints can be moved to the closed once the company has done working issue and the complaint is resolved.
- When moving to the Closed Complaints tab, the company can choose the type of complaint from the drop-down as shown below, depending on what type of issue or product the user has raised the complaint about.

Operational UI login details:

User role: Email: sejal.gupta@gmail.com

Password: User@1234

Company Logins	Email	Passowrd
Wells Fargo & Company	Gladys.Mahurin@wells.co.in	2H58YQJ
Capital One Financial Corporation	Jacqueline.Jones@capital.co.in	pbo9dnE
Jpmorgan Chase & Co.	Dana.Cattaneo@jpmorgan.co.in	t5GfGtz
Bank Of America, National Association Citibank, N.A.	Joseph. Hockman@bank.co.in	d5ksTce

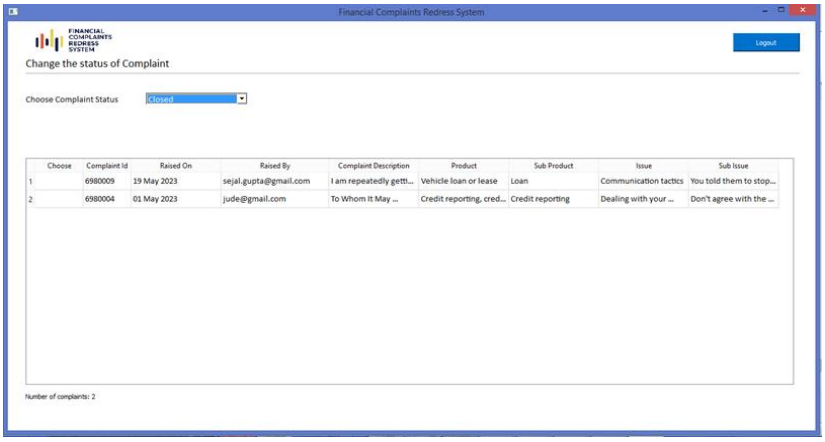


Fig 20. Closed Complaints Tab

The company can close complaints with following options: -

Closed
Closed with relief
Closed without relief
Closed with monetary relief
Closed with non-monetary relief
Closed with explanation

Summary For Operational Module

Registering as a customer allows the user to raise complaints against a company based on their experience with the product or issue. By registering a complaint, the company can be made aware of the issue and take corrective action. This improves the customer experience and ensures customer satisfaction. Complaints can also be used to identify and rectify any systemic issues with a product or service. As soon as the issue is raised, they can track their complaints from the same dashboard. In terms of company perspective, the companies are now able to recognize the complaints in a format, such as new, in-progress, or closed, which avoids confusion and increases the efficiency of the process.

The product handles most customer services and company forums with ease of understanding from both user and company perspectives. This provides companies with a clear picture of how their customer service is performing and allows them to make necessary changes to improve their customer service. Additionally, this system allows companies to respond to customer complaints quickly and efficiently.

Working of Analytical Module

The product focuses on any customer care service and is based on real-time scenarios of the banking portal. The idea behind this product was to create a system that allows different levels of users to focus on different aspects of the business. We have considered four levels of users when designing our current product.

1. The officials of CFPB (Consumer Finance Protection Bureau): - A report of the complaints and the key performance indicators (KPIs) set by the company, along with the percentage of whether those KPIs have been achieved, will be of interest for the government officials. “Complaints dashboard”

2. Complaints to the department head or product manager: - When government officials report that the KPIs are not met, the department head can perform further monitoring. In the "Summary Report," the HOD can check for products and categories where major issues are coming from and address them. If there is a specific issue like "Debt Collection", the HOD can obtain information from this "Summary Dashboard". When there is any kind of important analysis to be made, the HOD of the department has access to the operational dashboard and the analytical dashboard with a few quick validations.

3. The managers or government representatives (GR):- Once a manager or GR receives a product name or list of companies requiring attention, they can search for them (maximum three) and compare their behavior in detail to determine the exact cause of the issue and then report it to the company. Thus, we can solve this problem by providing a

solution to companies and governments to avoid unfortunate situations at banks and assure the public that the government is trustworthy.

4.The employee: - Users are mainly entry-level. Using this user, you can keep track of changes in company behavior and product issues every day. By using the "Comparative dashboards", users can compare any two categories against each other. Companies can be compared with products, sub-products, issues, or sub-issues. Also, at times we can compare four elements of two categories. Thus, it allows a total of 10 combinations, and each combination can be searched with four elements. The dashboard aims to drill down to the core of each category to assist with problem-solving. Furthermore, we can ROLL UP the results on a monthly, quarterly, and weekly basis depending on the severity of the issue.

Specifications and Usability of Analytical Module

1. Product Detail Description

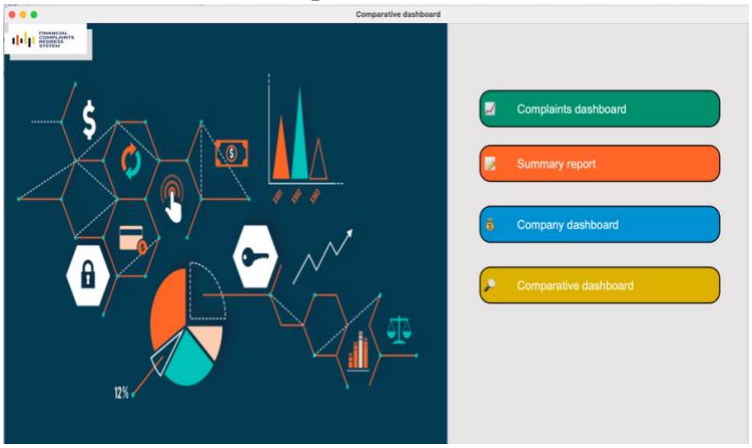


Fig 21. Analytical Module Intro

- This page is the initial page of the project which allows user to select the dashboards based on the information required

2. Complaints dashboard

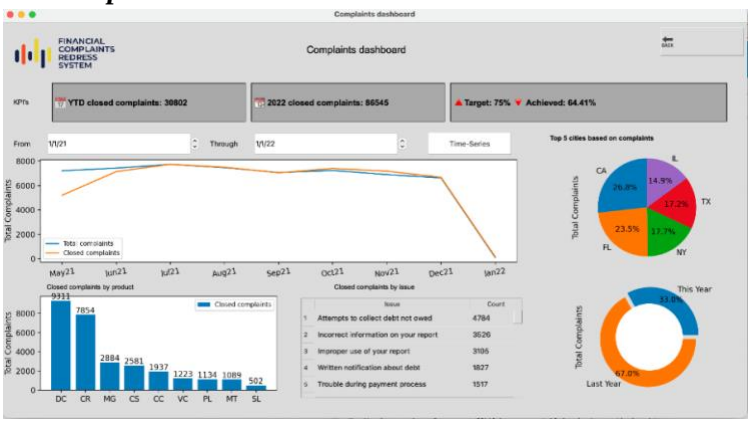
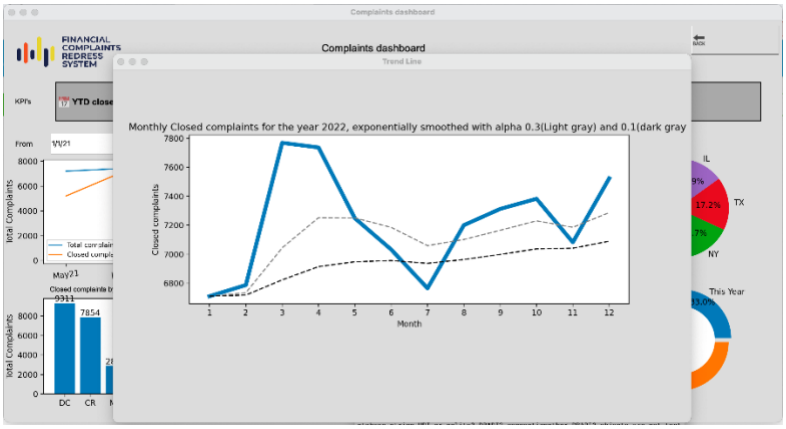


Fig 22. Complaints dashboard

- **"Complaints dashboard": -** This is Complaints Dashboard page where executive views this page. It displays the overall data in organized and graphical ways.
- **Functionalities**: - Executive can select a date range from and through to see the closed complaints by Month-Year. To display the complaints data in Month-Year we need to use the concept of **DRILL-UP**.
- A company's KPIs are used to measure its performance. YTD stands for year to date. 75% indicates company has set target to close 75% of tickets.
- Closed complaints are displayed by short names. Hover over the title to see the detailed names of products.

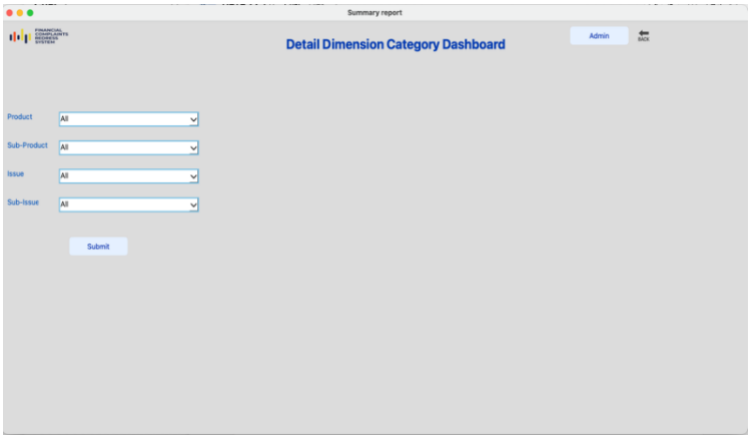
3. Time series



- **"Time-Series": -** When a executive clicks on Time-Series he/she can view a monthly closed complaints for last year.
- **Functionalities**: - Exponentially smoothed with alpha 0.3 and 0.1.
- The predicted graph using alpha 0.3 is close to the actual value. Hence, we can use alpha 0.3 to predict the future value of closed complaints.
- Use of **Moving Average and Exponential Smoothening concepts**.

Fig 23. Time series forecasting

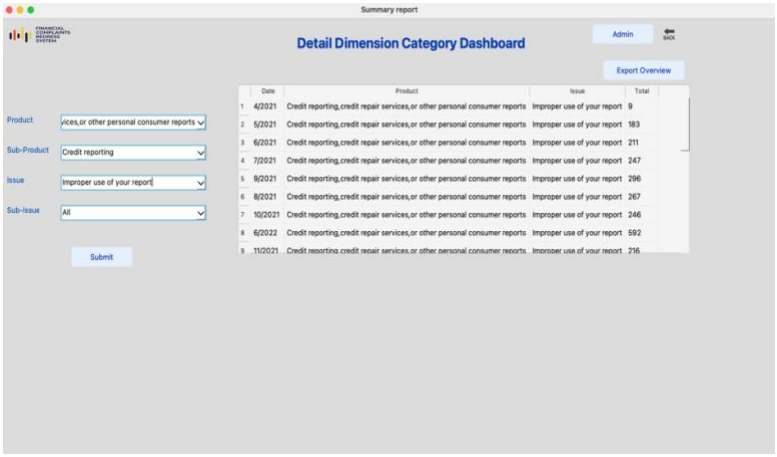
4. Summary Report



- **"Summary Report":** - This page allows user to select the product, sub-product, issue, sub-issue. User can select "All" if user is not sure about the selection.
- **Functionalities**: - The database is used to get the dependent data in the Combo-box. Based on selection of Product the rest of combo box changes and based on Issue, Sub- issue changes.
- We can not only select values from dropdown but also it allows us to type the values in the combo box.
- **OLAP** operation are demonstrated in this page.

Fig 24. Summary Report

5. Output of overview



- After selection of the values, we click on the submit button and results are displayed in tabular format.
- **Functionalities**: - On-clicking header the values are stored based on selected columns.
- This table shows the date Product, Issue and Total number of complaints.
- To know the details of the complaints we can click on the cell of the table and the detail table is displayed below.
- If the data is required to share, the page allows to export the table once clicked on export button.
- The **data filtering, grouping, extraction process** from database is demonstrated on this page.
- If there is no data for selection the results will not be displayed.

Fig 25. Overview report

Sample Input: -
Product: - Credit reporting, credit repair services, or other personal consumer reports
Sub-Product: - Credit reporting
Issue: - Incorrect information on your report
Sub-Issue: - Information belongs to someone else

6. The output of a detailed overview

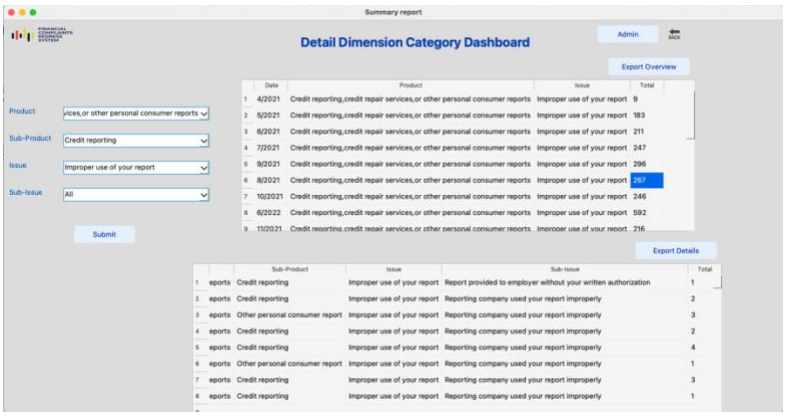


Fig 26. Overview report

- After clicking on the cell of the "Overview table" the details splitting on the total number of into Date (MM/YY), Company, Sub-Product, Issue, Sub-Issue, total number of complaints.
- **Functionalities**: - This tables shows the concepts of aggregation in the database.

7. *Admin Procedure*

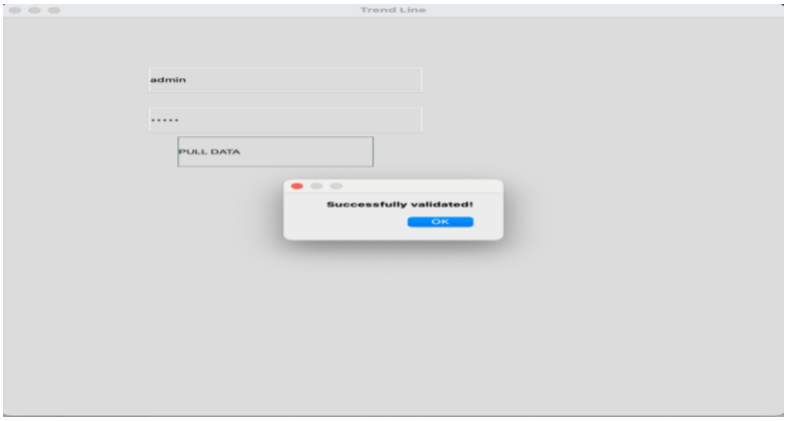


Fig 27. Admin procedure of pulling data

- There is "Admin" button on the top: - This button redirects to the page where it validated admin user and passwords and allows us to pull data from operational data base to analytical database.
- **Functionalities**: - This Pull commands can be used in case of critical situation like we need to restart the server, or some important analysis needs to be performed. We can take a backup of the operational server.

6. *Company Dashboard*

Fig 28. Company Dashboard

- On landing on this page at the first we can see the top 5 companies' data having maximum number of complaints till date.
- There are six charts: -

i. Number of timely response "Red" (16+ days) represents the delay and "Green" show Timely (from 1-15 days).
ii. Table: -This shows the short names used for companies and average time company takes to response to complaints.
ii. Number of complaints received in last 3 weeks (21days) from the top 5 companies
v. Number of complaints received in last 1 month (30 days) from the top 5 companies
v. Number of complaints received in last 3 months (90 days) from the top 5 companies
i. Number of complaints received in last 1 year (365 days) from the top 5 companies

Functionalities: -

- This page shows the properties of drill-down operations based on the hierarchical data (i.e., date).
- It allows you to select the dates from in FROM and TO text edit with the calendars.
- We can also view focus graphs on selecting the options like 3W (3 Weeks), 3M (3 months), 1M (1 month), 1Y (1 year), and All. This allows us to show the drill-down operations in the database systems.
- Company Selection: - User can select multiple companies (3 max) for comparison by clicking on the "+" button on the UI and remove after clicking on "- "on the UI

- Reset button: - This button will reset everything and refresh the page.

Fig 28. After selecting the Date Range

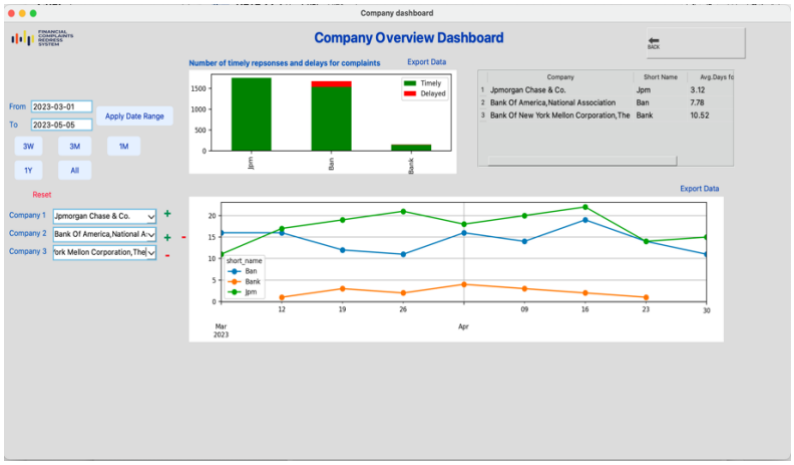


Fig 29. After selecting 3 companies for comparison

Sample input: -
FROM: - 2022-06-01 TO: - 2023-05-06
Company: - "All"
OR
Company 1: - Wells Fargo & Company
Company 2: - Capital One Financial Corporation
Company 3: - Jpmorgan Chase & Co.

7. *Comparative Analysis Dashboard*

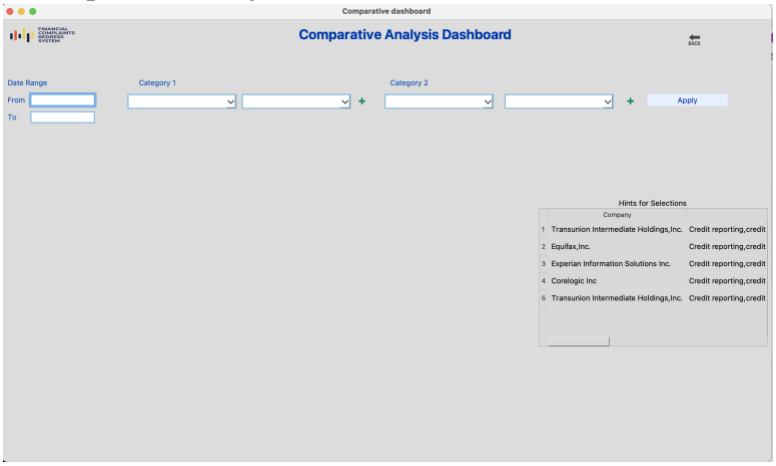


Fig 29. After selecting the Date Range

- "The Comparative Analysis Dashboard": - Allows user to compare between two categories for different time range. There is hint table for selection for completely newly user. Same can be used for sample input.
- **Functionalities: -** It performs the ROLLUP, DRILL DOWN operations on the based on the user selection.
- It shows the aggregations group by on the database system

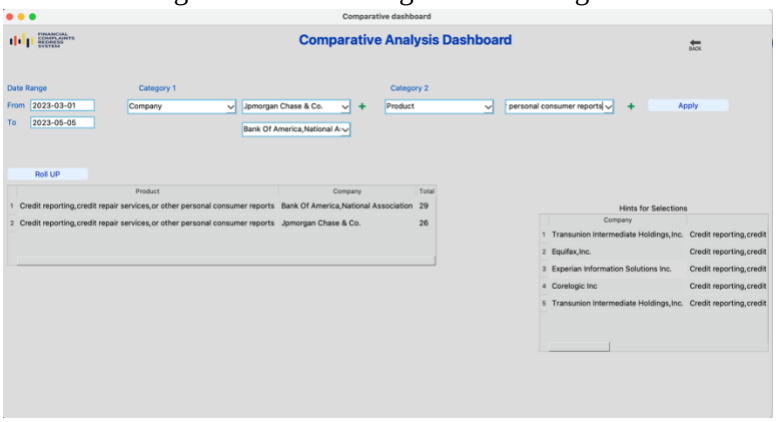


Fig 30. Enabling Roll UP

- Based on the selection the of categories. The results are shown in the tabular format.
- Then user can opt for "Roll UP".
- Once the user opt for roll up the database queries are performed and ROLL UP table in displayed.

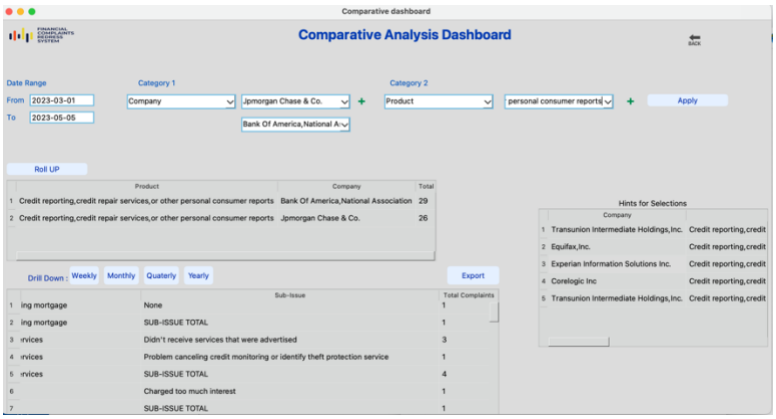


Fig 31. Enabling Drill Down

- Initially, the roll up operational table will have roll up based on the Product, Sub-Product, Issue, Sub-Issue for selected time range for number of complaints.
- After the loading of initial table, the user gets the option to select the “Drill Down” Based on weekly, monthly, quarterly, and yearly.

Fig 32. Drill down based on Quarterly

- In this case, quarterly option is selected. Therefore, another dataset column is added to the existing table that shows drill down operations.

Sample input: -
FROM: - 2022-06-01 TO: - 2023-05-06
Category 1: - Company: - 1) Ford Motor Credit Co.
2) Transunion Intermediate Holdings, Inc.
Category 2: - Product: - 1) Debt collection
2) Credit reporting, credit repair services, or other personal consumer reports

Summary For Analytical Module

The product aims to serve most user types, including executives, mid-levels, and entry-level. The tool assists companies in making informed decisions and identifies the areas they should focus on. It provides users with actionable insights and helps them to optimize their business processes. The tool also helps to minimize risks by providing a comprehensive overview of the company's performance.

There are two main features of the product: first, it can be used as a top-down flow where problems are raised at the top level, and then as a bottom-up flow where entry-level users can notice an issue, download a report, and let the manager know.

Customer complaints and customer care services can be handled with this tool, and it's user-friendly.

Technical Aspects

The launching of the application can be done from the Python terminal with selecting the folder of the project saved and executing the following commands: -

1. For operational: - python main_final.py
2. For analytical: - python Main.py

Files: -

1. **Common Files**
   which_ini.py → Is the supporting files that help us to navigate from server to local connection with changing the name of ini file
   DATA225utils → It is the utility file that have common function that required all over the project

2. **Operational**: -
   - main_final.py → Is the main file that is used to execute that holds all execution of all the project. We need to run this file for launching the application
   - server_connection.ini → Connection to the database on the server
   - server_connection_wh.ini → Connection to the warehouse on the server.

3. **Analytical**: -
   - Main.py → Is the main file that is used to execute that holds all execution of all the project. We need to run this file for launching the application
   - admin_login_page.py → This helps the admin to pull data from the operational database to the analytical database
   - complaints_dashboard.py → This page runs the complaints dashboard.
   - time_series.py → This page is for time series forecasting on complaints_dashborad.py
   - server_connection.ini → Connection to the database on the server
   - server_connection_warehouse.ini → Connection to the warehouse on the server.

4. **ETL**:-
   Loading data in database in warehouse.
   - Creating a Admin Source Company dimension.ipynb
   - Creating a calendar dimension.ipynb
   - Creating a Issue dimension.ipynb
   - Creating a Product dimension.ipynb
   - Creating Fact Table.ipynb

Note: - Application may time take to load as it has larger datasets.

Operational Queries

1. **Created view all_complaints to view complaints in company representative page**:

```
CREATE VIEW all_complaints AS
SELECT u.complaint_id,
DATE_FORMAT(STR_TO_DATE(c.date_sent
_to_company, '%Y-%m-%d'),'%d %M %Y') AS
date_string,
us.email, c.complaint_desc, p.product,
p.sub_product, i.issue, i.sub_issue,
c.company_response_to_consumer,
u.company_id, us.user_id
FROM user_complaint_an u
INNER JOIN complaint_details_an c ON
u.complaint_id=c.complaint_id
LEFT JOIN product_sub_product_an p ON
u.product_track_id=p.product_track_id
LEFT JOIN issue_sub_issue_an i ON
u.issue_track_id=i.issue_track_id
LEFT JOIN user_info us ON
u.user_id=us.user_id
LEFT JOIN admin_source_company_an comp
ON u.company_id=comp.company_id;
```

2. **Created the procedure create_new_complaint for submitting a complaint:**

```
CREATE PROCEDURE
create_new_complaint(IN complaint_desc
VARCHAR(500),
                     IN product_track_id
VARCHAR(45),
                     IN issue_track_id
VARCHAR(45),
                       IN user_id
VARCHAR(45),
                       IN company_id
VARCHAR(45),
                       OUT new_complaint_id
VARCHAR(20))
BEGIN
        DECLARE EXIT HANDLER FOR
SQLEXCEPTION
  BEGIN
    ROLLBACK;
    RESIGNAL;
```

```
    END;
    START TRANSACTION;

        SELECT @complaint_id :=
MAX(complaint_id)+1 FROM
complaint_details_an INTO new_complaint_id;

        INSERT INTO
complaint_details_an(complaint_id,
complaint_desc, date_sent_to_company,
company_response_to_consumer,timely_respon
se,consumer_disputed)
        VALUES (@complaint_id,
complaint_desc,
DATE_FORMAT(CURDATE(),
'%Y-%m-%d'), "New", "No", "None");

        INSERT INTO
user_complaint_an(complaint_id,
product_track_id, issue_track_id, user_id,
company_id,
        submitted_via, year, month, date)
        VALUES (@complaint_id,
product_track_id, issue_track_id, user_id,
   company_id, "Web", YEAR(CURDATE()),
MONTH(CURDATE()), DAY(CURDATE()));

    INSERT INTO
complaint_track(complaint_id, date, status,
company_closing_explanation)
        VALUES (@complaint_id,
CURDATE(), "New", "None");

COMMIT;
END;
```

Analytical Queries

1. **Pulling the data from operational to analytical database:** This stored procedure pulls the data from operational to analytical using the concept of **Slowly changing dimension 0**

```
CREATE PROCEDURE
update_dim_and_fact_complaints()
 BEGIN
  IF (SELECT count(*) as cnt FROM
dim_calendar_an cal RIGHT JOIN
thedeciders_db.user_complaint_an u. on u.month =
cal.month and u.year = cal.year and u.date =
cal.date. WHERE cal.month is null) > 0
THEN
   INSERT INTO
dim_calendar_an(full_date,year,month,date,day_of
_week,week_number_of_the_year,qtr)
            SELECT date_format(concat(u.year,'-
',u.month,'-',u.date),'%Y-%m-%d') as
full_date,u.year,u.month,u.date,
dayofweek(date_format(concat(u.year,'-',u.month,'-
',u.date),'%Y-%m-%d')) as
day_of_week,weekofyear(date_format(concat(u.ye
ar,'-',u.month,'-',u.date),'%Y-%m-%d')) as
weeknum   ,quarter(date_format(concat(u.year,'-
',u.month,'-',u.date),'%Y-%m-%d')) as quarter
            FROM
thedeciders_db.user_complaint_an u
            LEFT JOIN dim_calendar_an cal
on u.month = cal.month and u.year = cal.year and
u.date = cal.date WHERE
 cal.month is null;
            end if;

            IF (
            SELECT count(*) as cnt FROM
fact_user_complaints_an fact
            RIGHT JOIN
thedeciders_db.user_complaint_an u
            on u.complaint_id = fact.complaint_id
            where fact.complaint_id is null) > 0
            THEN
            INSERT INTO
fact_user_complaints_an(complaint_id,submitted_
via, timely_response,
complaint_received_calendar_key,issue_sub_issue
_key,product_sub_product_key,admin_source_co
mpany_key ,user_info_key,complaint_closed_cale
ndar_key)
             SELECT
u.complaint_id,u.submitted_via,'Yes' as
timely,cal.calendar_key
            ,iss.issue_sub_issue_key
            ,p.product_sub_product_key
            ,comp.admin_source_company_key,u
ser.user_info_key,cal.calendar_key
             FROM fact_user_complaints_an fact.
RIGHT JOIN thedeciders_db.user_complaint_an u
 on u.complaint_id = fact.complaint_id
  INNER JOIN dim_product_sub_product_an p
   on p.product_track_id = u.product_track_id
```

```
  INNER JOIN dim_admin_source_company_an
comp  ON comp.company_id = u.company_id
INNER JOIN dim_user_info user
on user.user_id = u.user_id
  INNER JOIN dim_issue_sub_issue_an iss
                on iss.track_id = u.issue_track_id
   INNER JOIN dim_calendar_an cal ON u.month
= cal.month and u.year = cal.year and u.date =
cal.date
                where fact.complaint_id is null;
                end if;

                SELECT count(*) as cnt
                 FROM fact_user_complaints_an fact
                 RIGHT JOIN
thedeciders_db.user_complaint_an u
                on u.complaint_id = fact.complaint_id
                INNER JOIN
dim_product_sub_product_an p
                on p.product_track_id =
u.product_track_id
                INNER JOIN
dim_admin_source_company_an comp
                on comp.company_id = u.company_id
                INNER JOIN dim_user_info user
                on user.user_id = u.user_id
                INNER JOIN
dim_issue_sub_issue_an iss
                on iss.track_id = u.issue_track_id
                INNER JOIN dim_calendar_an cal
                                                on
u.month = cal.month and u.year = cal.year and
u.date = cal.date
                where fact.complaint_id is null;

            END$$
```

2. **Created a view for analytical queries: -**

```
CREATE VIEW
 `table_click_details` AS
SELECT `complaint_id`, `company_name`,
concat(`cal`.`month`,'/',`cal`.`year`) AS
`month_year`,
`timely_response, `product, `sub_product`,
`issue`,`sub_issue`,
from ((((`fact_user_complaints_an` `ft` join
`dim_admin_source_company_an` `adm`
on((`ft`.`admin_source_company_key` =
`adm`.`admin_source_company_key`)))
join `dim_issue_sub_issue_an` `isi`
on((`ft`.`issue_sub_issue_key` =
`isi`.`issue_sub_issue_key`)))
join `dim_product_sub_product_an` `psp`
on((`ft`.`product_sub_product_key` =
`psp`.`product_sub_product_key`)))
join `dim_calendar_an` `cal`
on((`ft`.`complaint_received_calendar_key` =
`cal`.`calendar_key`)))
```

3. **Retrieving the data based on product selection:**

For the “Summary Report” this is a dynamic query that selection of products and issues from db.

```
        SELECT
         CONCAT(month, '/', year) AS
month_year,product,issue,COUNT(complaint_id)
AS total  FROM
         fact_user_complaints_an ft
         JOIN dim_issue_sub_issue_an isi ON
ft.issue_sub_issue_key = isi.issue_sub_issue_key
         JOIN dim_product_sub_product_an psp
ON ft.product_sub_product_key =
psp.product_sub_product_key
         JOIN dim_calendar_an cal ON
ft.complaint_received_calendar_key
=cal.calendar_key
 WHERE CONCAT(month, '/', year) IS NOT
NULL
         AND product = '{}'
         AND sub_product = '{}'
         AND issue = '{}'
         AND sub_issue = '{}'
```

4. **On the table click the retrieving data from DB:**

```
SELECT month_year, company_name, product,
sub_product, issue, sub_issue,
COUNT(complaint_id) as total
      FROM table_click_details
      WHERE month_year COLLATE
utf8mb4_general_ci = "{}"
      AND product COLLATE
utf8mb4_general_ci = "{}"
      AND issue COLLATE utf8mb4_general_ci
= "{}"
      GROUP BY month_year, company_name,
product, sub_product, issue, sub_issue
```

5. **Retrieving data for the “Company Overview”: -**

For default the “where” statement was removed and LIMIT =5 was set

```
SELECT company_name,
COUNT(complaint_id) AS `total_complaints`,
COUNT((CASE WHEN (`timely_response` =
'Yes') THEN 1 END))
AS`number_timely_responsed`,
 COUNT((CASE WHEN (`timely_response` =
'No') THEN 1 END)) AS `number_delayed`,
ROUND(AVG(days_to_close),2) as
`average_time`
FROM table_click_details_1
WHERE company_name IS NOT NULL
AND company_name ='{}' OR  company_name
='{}'  OR  company_name ="{}"
GROUP BY company_name ORDER BY
total_complaints DESC;
```

6. **Selection of drill down based on weekly, monthly, yearly, etc. in Company Overview: -**

```
SELECT t1.company_name, CONCAT('Week ',
WEEK(t2.received_date)) AS week_number,
DATE_FORMAT(MIN(t2.received_date), '%b %d')
AS week_start, COUNT(t2.complaint_id) AS total
            FROM (
               SELECT company_name,
COUNT(complaint_id) AS total_complaints
  FROM thedeciders_wh.table_click_details_1
 WHERE company_name IS NOT NULL
AND company_name ='{}'
OR  company_name ='{}'
   OR  company_name ="{}"
  GROUP BY company_name
  ORDER BY total_complaints DESC ) t1
JOIN thedeciders_wh.table_click_details_1 t2 ON
t1.company_name = t2.company_name WHERE
t2.received_date >= (SELECT MAX(received_date) -
INTERVAL 21 DAY FROM
thedeciders_wh.table_click_details_1)
            GROUP BY t1.company_name,
week_number;
```

7. **When the date range was selected the between query was used: -**

```
SELECT t1.company_name, received_date,
COUNT(t2.complaint_id) AS total
              FROM (
            SELECT company_name,
COUNT(complaint_id) AS total_complaints
            FROM
thedeciders_wh.table_click_details_1
                 WHERE company_name IS NOT
NULL AND company_name ='{}'
OR  company_name ='{}'
OR  company_name ="{}"
GROUP BY company_name
ORDER BY total_complaints DESC  ) t1
       JOIN thedeciders_wh.table_click_details_1 t2
ON t1.company_name = t2.company_name
       WHERE t2.received_date BETWEEN '{}'
AND '{}'
       GROUP BY t1.company_name,
received_data
             AS sub_issue,
                    COUNT(complaint_id) AS
TOTAL_COMPLAINTS
       FROM table_click_details_1
       WHERE received_date IS NOT NULL
       """+text+"""
       AND received_date between
"""+str(date1)+""" and """+str(date2)+"""
       GROUP BY  company_name, product,
sub_product, issue,sub_issue
       WITH ROLLUP
```

8. **ROLL UP query for Comparative dashboard.**

```
SELECT
      IF (GROUPING(company_name),
'Company TOTAL',     company_name)
            AS company_name,
      IF (GROUPING(product),    'PRODUCT
TOTAL',   product)
            AS product,
      IF (GROUPING(sub_product),     'SUB-
PRODUCT TOTAL',    sub_product)
            AS sub_product,
      IF (GROUPING(issue),       'ISSUE TOTAL',
issue)
            AS issue,
        IF (GROUPING(sub_issue),       'SUB-
ISSUE TOTAL',      sub_issue)
```

9. **Drill down with roll up for Comparative analysis:-**

```
SELECT IF (GROUPING(company_name),
'***COMPANY TOTAL', company_name) AS
company_name,
```

```
        IF (GROUPING(product), '***PRODUCT
    TOTAL', product) AS product,
        IF (GROUPING(sub_product), '***SUB-
    PRODUCT TOTAL', sub_product) AS
    sub_product,
        IF (GROUPING(issue), '***ISSUE TOTAL',
    issue) AS issue,
        IF (GROUPING(sub_issue), '***SUB-ISSUE
    TOTAL', sub_issue) AS sub_issue,
        IF (GROUPING("""+by+"""),
    """+as_label+""","""+by+""") AS """+by+""",
            COUNT(complaint_id) AS
TOTAL_COMPLAINTS
        FROM drill_down
        WHERE received_date IS NOT NULL
"""+text+"""
AND received_date between """+str(date1)+"""
and """+str(date2)+"""
GROUP BY company_name, product,
sub_product, issue, sub_issue, """+by+"""
        WITH ROLLUP;
```

10. Drill up for displaying Month-Year in Complaints dashboard page:

```
SELECT V.monyear,V.year,V.month, V.closed_cnt, ope.open_cnt

    FROM

    (

    SELECT date_format(Cal.full_date, '%b%y') As monyear,Cal.month,Cal.year,

    Count(complaint_id) As closed_cnt FROM fact_user_complaints_an F

    INNER JOIN dim_calendar_an Cal

    on Cal.calendar_key = F.complaint_closed_calendar_key

    WHERE Cal.full_date BETWEEN '{}' and '{}'

    GROUP BY date_format(Cal.full_date, '%b%y'),Cal.month,Cal.year

    ) AS V

    INNER JOIN

    (

    SELECT date_format(C.full_date, '%b%y') As monyear,C.month,C.year,Count(complaint_id) As open_cnt FROM
fact_user_complaints_an F

    INNER JOIN dim_calendar_an C

    ON C.calendar_key = F.complaint_received_calendar_key

    WHERE C.full_date BETWEEN '{}' and '{}'

    GROUP BY date_format(C.full_date, '%b%y'),C.month,C.year

    ) ope ON ope.monyear = V.monyear

    order by V.year,V.month
```

Appendix

Field name	Description	Data type	Notes
Date received	The date the CFPB received the complaint	date & time	
Product	The type of product the consumer identified in the complaint	plain text	This field is a categorical variable.
Sub-product	The type of sub-product the consumer identified in the complaint	plain text	This field is a categorical variable. Not all Products have Sub-products.
Issue	The issue the consumer identified in the complaint	plain text	This field is a categorical variable. Possible values are dependent on Product.
Sub-issue	The sub-issue the consumer identified in the complaint	plain text	This field is a categorical variable. Possible values are dependent on product and issue. Not all Issues have corresponding Sub-issues.
Consumer complaint narrative	Consumer complaint narrative is the consumer-submitted description of "what happened" from the complaint. Consumers must opt-in to share their narrative. We will not publish the narrative unless the consumer consents, and consumers can opt-out at any time. The CFPB takes reasonable steps to scrub personal information from each complaint that could be used to identify the consumer.	plain text	Consumers' descriptions of what happened are included if consumers consent to publishing the description and after we take steps to remove personal information.
Company public response	The company's optional, public-facing response to a consumer's complaint. Companies can choose to select a response from a pre-set list of options that will be posted on the public database. For example, "Company believes the complaint is the result of an isolated error."	plain text	Companies' public-facing responses to complaints are included if companies choose to publish one. Companies may select a public response from a set list of options as soon as they respond to the complaint, but no later than 180 days after the complaint was sent to the company for response.
Company	The complaint is about this company	plain text	This field is a categorical variable.
State	The state of the mailing address provided by the consumer	plain text	This field is a categorical variable.
ZIP code	The mailing ZIP code provided by the consumer	plain text	The mailing ZIP code provided by the consumer. The 5-digit United States Postal Service ZIP code will be published where provided unless the consumer lived in a ZIP code aligned to a United States Census Bu20reau ZIP Code Tabulation Area (ZCTA) with fewer than 20,000 people and consented to publication of their complaint narrative. In those cases, where the Census ZCTA had fewer than 20,000 people, the 3-digit ZIP code will be published if the 3-digit ZCTA has more than 20,000 people. Otherwise, no ZIP code will be published.
Tags	Consumer complaints can be searched and sorted more easily. A complaint involving 62 or older consumers is called, ‘Older American.’ Complaints involving service members, spouses, or dependents of service members are called, ‘Servicemember.’ It includes active-duty, reserve, and National Guard servicemen, as well as veterans and retirees.	plain text	

Consumer consent provided?	Identifies whether the consumer opted in to publish their complaint narrative. We do not publish the narrative unless the consumer consents and consumers can opt-out at any time.	plain text	• This field shows whether a consumer provided consent to publish their complaint narrative, as listed below: • Consent provided: Consumers opted in to share their complaint narrative. • Data populates in this field 60 days after the complaint was sent to the company for response or after the company provides an optional company public response – whichever comes first, and after steps have been taken to scrub personal information from the complaint narrative. • Consent not provided: Consumer did not opt-in to publish their complaint narrative. Data populates in this field 60 days after the complaint was sent to the company for response or after the company provides an optional company public response – whichever comes first. • Consent withdrawn: Consumer opted in to publish their complaint narrative and later withdrew their consent. • N/A: Consumers did not have the option to publish their consumer complaint narrative or the complaint was received before March 19, 2015. Data populates in this field immediately. • Other: Complaint does not meet criteria for narrative publication. • Blanks appear until at least 60 days after the complaint is sent to the company for response or until the company provides an optional company public response – whichever comes first.
Submitted via	How the complaint was submitted to the CFPB	plain text	This field is a categorical variable.
Date sent to company	The date the CFPB sent the complaint to the company	date & time	
Company response to consumer	This is how the company responded. For example, "Closed with explanation."	plain text	This field is a categorical variable.
Timely response?	Whether the company gave a timely response	plain text	yes/no
Consumer disputed?	Whether the consumer disputed the company's response	plain text	Yes/ No N/A: The Bureau discontinued the consumer dispute option on April 24, 2017.
Complaint ID	The unique identification number for a complaint	number	