

# **CHAPTER 1**

## **INTRODUCTION**

### **1.1 Domain Overview**

This project collected data from the largest hacking community and did not consider other hacking communities. Future studies will therefore need to generalize our findings by investigating a wider range of hacking communities. Second, this study has focused on the CaaS and crimeware available in the cybercrime underground, but much in-depth analysis remains to be done on the configurations of cybercrime networks. In this project we use Machine Learning as domain as A Data Analytics approach to analyze the Cybercrime and its Community. We use Data-Analytics Domain with the use of Machine Learning Naïve bayes algorithm approach for predicting a large set of data. We use Python as Coding language with a Django web frameworks for Domain Overview.

### **1.2 Project Overview**

As the threat posed by massive cyberattacks (e.g., ransomware and distributed denial of service attacks (DDoS)) and cybercrimes has grown, individuals, organizations, and governments have struggled to find ways to defend against them. In 2017, ransomware known as WannaCry was responsible for nearly 45,000 attacks in almost 100 countries. The explosive impact of cybercrime has put governments under pressure to increase their cybersecurity budgets. United States President Barack Obama proposed spending over \$19 billion on cybersecurity as part of his fiscal year 2017 budget, an increase of more than 35% since 2016. The cybercrime underground has thus emerged as a new type of organization that both operates black markets and enables cybercrime conspiracies to flourish. In contrast, cybercrime networks are lateral, diffuse, fluid, and evolving the threat posed by the rise of highly professional network-based cybercrime business models, such as Crimeware-as-a-Service (CaaS), remains mostly invisible to governments, organizations, and individuals. Despite the rapid escalation of cyber threats, there has still been little research into the foundations of the subject or methodologies that could serve to guide Information Systems researchers and practitioners who deal with cyber security. In addition, little is known about Crime-as-a-Service (CaaS), a criminal business model that underpins the cybercrime underground.

### **1.3 Existing System**

Cybercrime has undergone a revolutionary change, going from being product-oriented to service-oriented because the fact it operates in the virtual world, with different spatial and temporal constraints, differentiates it from other crime taking place in the physical world. As part of this change, the cybercrime underground has emerged as a secret cybercrime marketplace because emerging technological changes have provided organized cybercriminal groups with unprecedented opportunities for exploitation. The cybercrime underground has a highly professional business model that supports its own underground economy. This business model, known as CaaS, is “a business model used in the underground market where illegal services are provided to help underground buyers conduct cybercrimes, such as attacks, infections, and money laundering in an automated manner,”. Thus, CaaS is referred to as a do-it-for-me service, unlike crimeware which is a do-it-yourself product. Because CaaS is designed for novices, its customers do not need to run a hacking server or have high-level hacking skills. Consequently, the CaaS business model can involve the following roles: writing a hacking program, performing an attack, commissioning an attack, providing an attack server (infrastructure), and laundering the proceeds.

Sood and Enbody have suggested that crimeware marketplaces have three key elements, namely actors (e.g., coders, operators, or buyers), value chains, and modes of operation (e.g., CaaS, pay-per-install, crimeware toolkits, brokerage, or supplying data). Periodic monitoring and analysis of the content of cybercrime marketplaces could help predict future cyber threats.

### **1.4 Drawbacks of Existing System**

- A previous study proposed a data mining framework for crime, dividing crimes harmful to the general public into eight categories:
  1. Traffic violations.
  2. Sex Crime.
  3. Theft.
  4. Fraud.
  5. Gang/Drug Offenses.
  6. Arson.
  7. Violent crime.
  8. Cyber-crime.
- Although this previous study explained how data mining techniques could be applied to crime analysis, it did not consider the specific features of cyber-crime.

- The “absence of capable guardians against crime” is due to organizations failing to take preventive measures against cybercrime.
- Periodic monitoring and analysis of the content of cybercrime marketplaces could help predict future cyber threats.
- Crimeware marketplaces have three key elements, namely actors (e.g., coders, operators, or buyers), value chains, and modes of operation (e.g., CaaS, pay-per-install, crimeware toolkits, brokerage, or supplying data).
- Consequently, the CaaS business model can involve the following roles:
  - i. writing a hacking program
  - ii. performing an attack
  - iii. commissioning an attack
  - iv. providing an attack server (infrastructure) and laundering the proceeds.

## 1.5 Problem Statement

These proposed sets of definitions for different types of CaaS the most common account hacking methods are phishing and brute force attacks. With an emphasis on selling this as a service, we define an account hacking service as a service that offers to gain unauthorized access to a target’s account by obtaining account information (e.g., username and password) or extra security information (e.g., security questions and answers).

Industry	Company
<b>Technology</b> (e.g., software, automobiles)	3M, Adobe, BMW, CISCO, EA, Exino Inc., GE, GlobalScape, HP, IDC Research Inc., Intel Corp., KDDI Japan, LG, Microsoft, Oracle, Panasonic, Panda Security, Philips, Samsung, Scania, Simba, Softbank Korea, Sony, Sybase, Sycore Business Solutions Corp., SynLan Technologies, Western Digital, Yamaha
<b>Content</b> (e.g., social network services, Internet, news)	ABC, AOL, Baidu, Bang Bros, CBS, Craigslist, Facebook, Google, IMDB, Instagram, Justin.tv, Last.FM, LinkedIn, LiveJournal, MSN, NBC, SoundCloud, Twitter, Warner Bros, Yahoo, YouTube, Zynga
<b>Finance</b> (e.g., banking, investing, and payments)	AlertPay, American Express, AMP, Personal Banking, Bank of America, Blackrock, Canadian Bank, Clickbank, Digital River, Goldman Sachs, iBank, Indian Bank, IP Capital, Kidd, Liberty Reserve, Moneybookers, PayPal, PlaySpan, Polish Bank, State Bank of India, Tradestation, Western Union
<b>E-commerce</b> (e.g., products and services)	Amazon, Best Buy, Dope, eBay, GameStop, GoDaddy, Groupon, Netflix, Nike, Staples, Uber, Walmart
<b>Tele Comm.</b> (e.g., smartphones and service providers)	Apple, AT&T, HTC, KT Freetel, MetroPCS, Nokia, Sprint, Swisscom, T-Mobile, Verizon Wireless
<b>Others</b>	Airsoft Gun, Ajanta Pharmaceuticals, ARMA International, FedEx, Green Leaf Technology, UPS, USG Corp.

**Figure 1.5 Problem Faced by Companies due to Cyber crime**

## 1.6 Project Scope & Motivation

To achieve this goal, we propose (1) a data analysis framework for analyzing the cybercrime underground, (2) CaaS and crime ware definitions, and (3) an associated classification model. In addition, we (4) develop an example application to demonstrate how the proposed framework and classification model could be implemented in practice.

We then use this application to investigate the cybercrime underground economy by analyzing a large dataset obtained from the online hacking community. By taking a design science research approach, this study contributes to the design artifacts, foundations, and methodologies in this area. Moreover, it provides useful practical insights to practitioners by suggesting guidelines as to how governments and organizations in all industries can prepare for attacks by the cybercrime underground.

## 1.7 Proposed System

The goal of our data analysis framework is to conduct a big-picture investigation of the cybercrime underground by covering all phases of data analysis from the beginning to the end. This framework comprises four steps: (1) defining goals; (2) identifying sources; (3) selecting analytical methods; and (4) implementing an application. Because this study emphasizes the importance of RAT for analyzing the cybercrime underground, the proposed RAT-based definitions are critical to this framework: **Steps 1–2** all contain the RAT elements.

**A.Step 1: Defining Goals** The first step is to identify the conceptual scope of the analysis. Specifically, this step identifies the analysis context, namely the objectives and goals. To gain an in-depth understanding of the current CaaS research, we investigated the cybercrime underground, which operates as a closed community. Thus, the goal of the proposed framework is to “investigate the cybercrime underground economy.”

**B.Step 2: Identifying Sources** the second step is to identify the data sources, based on the goals defined by Step 1. This step should consider what data is needed and where it can be obtained. Since the goal of this study is to investigate the cybercrime underground, we consider data on the cybercrime underground community. We therefore collected such data from the community itself and obtained a malware database from a leading global cyber security research firm. Because cybercriminals often change their IP addresses and use anti-crawling scripts to conceal their communications, we used a self-developed crawler that can resolve captcha’s and anti-crawling scripts to gather the necessary data. We

collected a total of 2,672,091 posts selling CaaS or crimeware, made between August 2008 and October 2017, from a large hacking community site ([www.hackforums.net](http://www.hackforums.net)) with over 578,000 members and more than 40 million posts.

We also collected 16,172 user profiles of sellers and potential buyers, based on their communication histories, as well as prices and questions and answers about the transactions. The black market uses traditional forum threads (e.g., bulletin boards) instead of typical e-commerce platforms (e.g., eBay, and Amazon). For example, sellers create threads in marketplace forums to sell items, and potential buyers comment on these threads. One of the most significant challenges was therefore converting this unstructured data into structured data.

## **1.8 Advantages of Proposed System**

- Compelling and relevant content will grab the attention of potential customers and increase brand visibility.
- You can respond instantly to industry developments and be seen as ‘thought leader’ or expert in your field.
- This can improve how your business is seen by your audience.
- Positive feedback is public and can be persuasive to other potential customers.
- Negative feedback highlights areas where you can improve.
- Data mining is a technique used to mine out patterns of useful data from large data sets.
- The goal of the proposed framework is to “investigate the cybercrime underground economy.” This Crimeware marketplaces have three key elements, namely
  - 1.actors (e.g., coders, operators, or buyers),
  - 2.value chains, and
  - 3.modes of operation (e.g., CaaS, pay-per-install, crimeware toolkits, brokerage, or supplying data).

## **1.9 Objective of the Project**

This project collected data from the largest hacking community and did not consider other hacking communities. Future studies will therefore need to generalize our findings by investigating a wider range of hacking communities. Second, this study has focused on the CaaS and crimeware available in the cybercrime underground, but much in-depth analysis remains to be done on the configurations of cybercrime networks.

To achieve this goal, we propose (1) a data analysis framework for analyzing the cybercrime underground, (2) CaaS and crime ware definitions, and (3) an associated classification model. In addition, we (4) develop an example application to demonstrate how the proposed framework and classification model could be implemented in practice. We then use this application to investigate the cybercrime underground economy by analyzing a large dataset obtained from the online hacking community. By taking a design science research approach, this study contributes to the design artifacts, foundations, and methodologies in this area. Moreover, it provides useful practical insights to practitioners by suggesting guidelines as to how governments and organizations in all industries can prepare for attacks by the cybercrime underground.

## **1.10 Organization of Report**

This chapter summarizes introduction of project which is elaborately described in later section of report, so the second chapter provides a detailed survey about this projected system which constitutes various paper related to how the performance of Python/Django can be improved. In the third chapter the requirements, constraints listed by the user for designing this application is described and following chapter illustrates design of the application which comprises of sequence diagram, architecture, and so on. Fifth chapter gives insight on implementation part and then testing techniques and test cases used for verifying this application is depicted in chapter six. Analysis of report containing snapshots is present in seventh chapter and finally conclusion and future enhancement is specified in the end along with a references or bibliography.

## **CHAPTER 2**

### **LITERATURE SYRVEY**

#### **2.1 Literature Review**

Cybercrime has undergone a revolutionary change, going from being product-oriented to service-oriented because the fact it operates in the virtual world, with different spatial and temporal constraints, differentiates it from other crime taking place in the physical world. As part of this change, the cybercrime underground has emerged as a secret cybercrime marketplace because emerging technological changes have provided organized cybercriminal groups with unprecedented opportunities for exploitation. The cybercrime underground has a highly professional business model that supports its own underground economy. This business model, known as CaaS, is “a business model used in the underground market where illegal services are provided to help underground buyers conduct cybercrimes, such as attacks, infections, and money laundering in an automated manner,”.

Thus, CaaS is referred to as a do-it-for-me service, unlike crimeware which is a do-it-yourself product. Because CaaS is designed for novices, its customers do not need to run a hacking server or have high-level hacking skills. Consequently, the CaaS business model can involve the following roles: writing a hacking program, performing an attack, commissioning an attack, providing an attack server (infrastructure), and laundering the proceeds.

According to this theory, three elements are necessary for crimes to be committed:

- (1) a likely offender,
- (2) a suitable target, and
- (3) the absence of capable guardians against crime.

In a cybercrime context, the “likely offenders” are motivated sellers and potential buyers in the underground market, and the “suitable targets” are the targeted vulnerable organizations. The “absence of capable guardians against crime” is due to organizations failing to take preventive measures against cybercrime. Two types of product or service are available in the cybercrime underground. The first can be either CaaS or crimeware that are related to attack strategy, for example, phishing, brute force, or DDoS attacks, or can be used for spamming or creating botnets, exploits, ransomware, rootkits, or Trojans. Attack

strategies often exploit system vulnerabilities such as application loopholes. In addition, social engineering attacks exploit human vulnerabilities. The most well-known example of such an attack is the use of a “secret question” for password recovery: attackers check into the user’s background to guess the secret question and hence steal the account. Examples include encryption and virtual private network (VPN) services, crypters, and proxies. From the perspective of RAT, the likely offenders are attackers motivated to attack organizations or products that constitute a suitable target. If such targets are attacked, however, both the targets and those who supply their cybersecurity products become aware of the vulnerabilities that made the attack possible, leading them to apply security updates to their software. These updates can be seen as capable guardians against crime, and the preventive measures taken can be identified by looking through each program’s version history. However, this is not the end of the matter, because the attackers will then develop and sell new versions of their hacking tools to combat the guardians, thus re-establishing the third RAT condition, the absence of capable guardians against crime. Such events can also be identified by the version numbers of the hacking tools sold on the black market: since it is an online marketplace, attackers must give detailed explanations to retain their customers’ confidence.

Sl no	Paper Title	Publication name & Year of Publication	Findings of Proposed System (Implemented)	Implementation, Algorithm and methodology used	Remarks (Not yet implemented)
1	Cybercrime underground economy	IEEE Transactions (2018)	Data analytics approach to the system	Naïve Bayes Algorithm MAP functions and methodology	Under Implementation
2	Cybercrime at a Scale	IEEE community (2017)	Deployment of HTTP based botnet commands	HTTP commands and control panels	Deployment Stage
3	Toward a better Measure of business proximity	MIS Quart, Conf. (2016)	Fact sheet Cybersecurity national Action plan	Modeling for Industry Intelligence and facts.	Implemented
4	Bypassing Antivirus	Proc.,18 <sup>th</sup> Panhellenic Conf.2014	Detection with Encryption	Antivirus bypassing and detection of Encryption	Conference



5	VPN	IEEE potentials (2001)	VPN and its activities	VPN and proxy network models	Implemented
6	CAAS-crimeware	IEEE (2013).	A survey of commoditized crimeware	Crimeware-as-a-service and Crimeware underground market	Conference
7	Cyberterrorism Today	Emerald group publications (2014).	Understand Cyberterrorism	Cyberterrorism and its access towards business	Publication
8	Hacking vulnerability	relevant result (July 6, 2011). BBC news (Feb. 24,2016).	Hacking vulnerability monitoring system.	Hacking vulnerability disclosed and the earlier signal from the underground	Published
9	Myths about malware	www.welivesecurity.com online platform	Malware and its Myths	Myths about malware and exploits	Uploaded
10	Trends in cybercrime	Justic,vol. 20,pp.,2005	The dark side of the Internet	Understanding the dark side of Internet.	Published

**Table 2.1 Literature Review**

## 2.2 Conclusion of Review

The data analysis step of the proposed framework involves four steps. Here, we report the data analysis results: CaaS and crimeware classification and market trends, cybercrime market dynamics, and potential hacking targets.

### **CaaS and Crimeware Classification and Market Trends**

Here, we evaluate the accuracy of the proposed classifications. Specifically, we analyze the CaaS and crimeware trends between 2008 and October 2017 based on these classifications. the most common classes overall were botnets (17%) and exploits (17%). The most popular classes in 2017 were botnets (33%), VPN services (20%), exploits (13%), and brute force attack services (7%). To validate our classification model, we used a confusion matrix, a common method of calculating classifier output accuracy. The training and testing datasets comprised 300 and 700 items, respectively. This gave an accuracy of 82.6% with a 95% confidence interval of (70.74%, 81.24%) for identifying the risks posed by CaaS- and crimeware related messages.

## CHAPTER 3

# SOFTWARE REQUIREMENTS SPECIFICATION

### 3.1 Introduction

A **Software Requirements Specification (SRS)** is a document that describes the nature of a project, software or application. In simple words, SRS document is a manual of a project provided it is prepared before you kick-start a project/application. This document is also known by the names SRS report, software document. A software document is primarily prepared for a project, software or any kind of application. There are a set of guidelines to be followed while preparing the software requirement specification document. This includes the purpose, scope, functional and nonfunctional requirements, software and hardware requirements of the project. In addition to this, it also contains the information about environmental conditions required, safety and security requirements, software quality attributes of the project etc.

#### 3.1.1 Purpose of the requirements document:

A requirements document is a [document](#) containing all the requirements to a certain product. It is written to allow people to understand *what* a product should do. Purpose and [scope](#), from both a technical and business perspective. Product overview and use cases Requirements, including functional requirements (e.g. what a product should do) usability requirements technical requirements (e.g. security, network, platform, integration, client) environmental requirements support requirements interaction requirements (e.g. how the product should work with other systems) Assumptions Constraints Dependencies.

#### 3.1.2 Project Perspective:

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

## **3.2 Functional Requirements:**

Functional requirements are used for expressing the behavior of a project, purpose and role each component. This is represented as using inputs, outputs and its behavior based on the specified input. While implementing design phase of system, functional requirements are considered and behavior of the whole project is realized using the use cases. The use case depicts the behavior, relationship between the components or modules of the project and the use case explanation for this project is illustrated in system design chapter.

- Graphical User Interface with the User & Admin Interface.
- Upload/Review Module.
- Chat Modules.
- Download/View Module.
- Feedback Module.
- Data Analysis via Bar/Pie/Spline Chart.

## **3.3 Non-Functional Requirements:**

### **3.3.1 Practicality**

Project practicality is where you use only what is viewed as least needed to meet your goals. A communication system like this needs hundreds of thousands of users to survive and thrive. Therefore, it should be designed to support large numbers of users, e.g., a substantial percent of a town or a campus.

### **3.3.2 Efficiency**

Efficiency measures how well and productively a manager uses his resources to achieve goals. Project management places heavy focus on how to acquire the right project team to perform project tasks and to close project successfully within the agreed constraints.

### **3.3.3 Cost**

Cost management is concerned with the process of planning and controlling the budget of a project or business. It includes activities such as planning, estimating, budgeting, financing, funding, managing, and controlling costs so that the project can be completed

within the approved budget. Cost management covers the full life cycle of a project from the initial planning phase towards measuring the actual cost performance and project completion.

### **3.3.4 Flexibility**

The need for flexibility is to deal with changed circumstances. Flexibility is used to scale back activities needing less effort while diverting resources to areas with unexpected problems.

### **3.3.5 Modularity**

Modularity refers to the concept of making multiple [modules](#) first and then linking and combining them to form a complete system. Modularity enables re-usability and minimizes duplication. In addition to re-usability, modularity also makes it easier to fix problems as bugs can be traced to specific system modules, thus limiting the scope of detailed error searching.

### **3.3.6 Extensibility**

Extensibility is a [software engineering](#) and [systems design](#) principle where the implementation takes future growth into consideration. The term extensibility can also be seen as a systemic measure of the ability to extend a [system](#) and the level of effort required to implement the extension. Extensions can be through the addition of new functionality or through modification of existing functionality.

### **3.3.7 Reliability**

Reliability refers to the probability and or the likelihood that a given product will perform in the way and or manner it was intended to perform in the efforts that have been deemed required of that given product within or under a specific period of time required.

### **3.3.8 Maintainability**

It is the probability that a system or system element can be repaired in a defined environment within a specified period of time. Increased maintainability implies shorter repair times.

### **3.3.9 Portability**

It is a measure of how easily an application can be transferred from one computer environment to another. A computer software application is considered portable to a new environment if the effort required to adapt it to the new environment is within reasonable limits. The phrase "to port" means to modify software and make it adaptable to work on a different computer system.

## **3.4 System Requirements:**

### **3.4.1 Hardware Requirements**

- Processor : Intel i3 7<sup>th</sup> Gen 2.4 GHz
- Hard Disk : Min 40 GB
- RAM : 4GB or above
- External : Mouse,Keyboard

### **3.4.2 Software Requirements**

- Operating system : Windows 10
- Front End : HTML5, CSS3, JavaScript, Bootstraps
- Coding Language : Python
- Database : MySQL/Sqlite3
- Tool : Pycharm IDE
- Framework : Django web frameworks
- Browsers : Chrome, Mozilla firefox

### **3.4.2 Software Requirements Specification (SRS) Table**

In this CHAPTER we specifically give a brief about Requirement Traceability Matrix which is mainly to check whether the requirements are satisfied or not.

<b>Serial No.</b>	<b>Requirements ID</b>	<b>Requirement In Brief</b>	<b>Requirement Description</b>
1	RID1	<b>Administrator</b>	admin is an business entiry, person to maintain the entire system.
2	RID2	<b>Users</b>	Users is the one who visits the application and can access the basic information.
3	RID3	<b>System Maintainers (Registered User)</b>	System Maintainer is the service receiver
4	RID4	<b>Home Page</b>	When we run the application we get the application home page.
5	RID5	<b>Login Module</b>	User of the system gets login using the credentials (id an password).
6	RID6	<b>Dataset</b>	In this module we collect opinion dataset from the reputed websites
7	RID7	<b>Data Mining</b>	Process of analysing the data from different perspective and summarize it into useful information.
8	RID8	<b>Machine Learning</b>	Used to develop softwares, microsoft technology more developer friendly.
9	RID9	<b>Django, Pycharm</b>	Used to develop browser based application.
10	RID10	<b>Python</b>	More Suitable programming langauage for real time projects.
11	RID11	<b>MYSQL Server</b>	Microsoft technology used to store the data (opinions data)
12	RID12	<b>Browsers</b>	IE, Google Chrome, Firefox

**Table 3.4.2 SRS Table**

## CHAPTER 4

### SYSTEM DESIGN

#### 4.1 Introduction to System Design

**System design** is a substantially important phase in project building and creation stages of whole development cycle. The system design is a mechanism for depicting and representing the overall architecture of the system, interfaces between the different components, the methods and parameters defined for each module and data for a system according to requirements specified by the user.

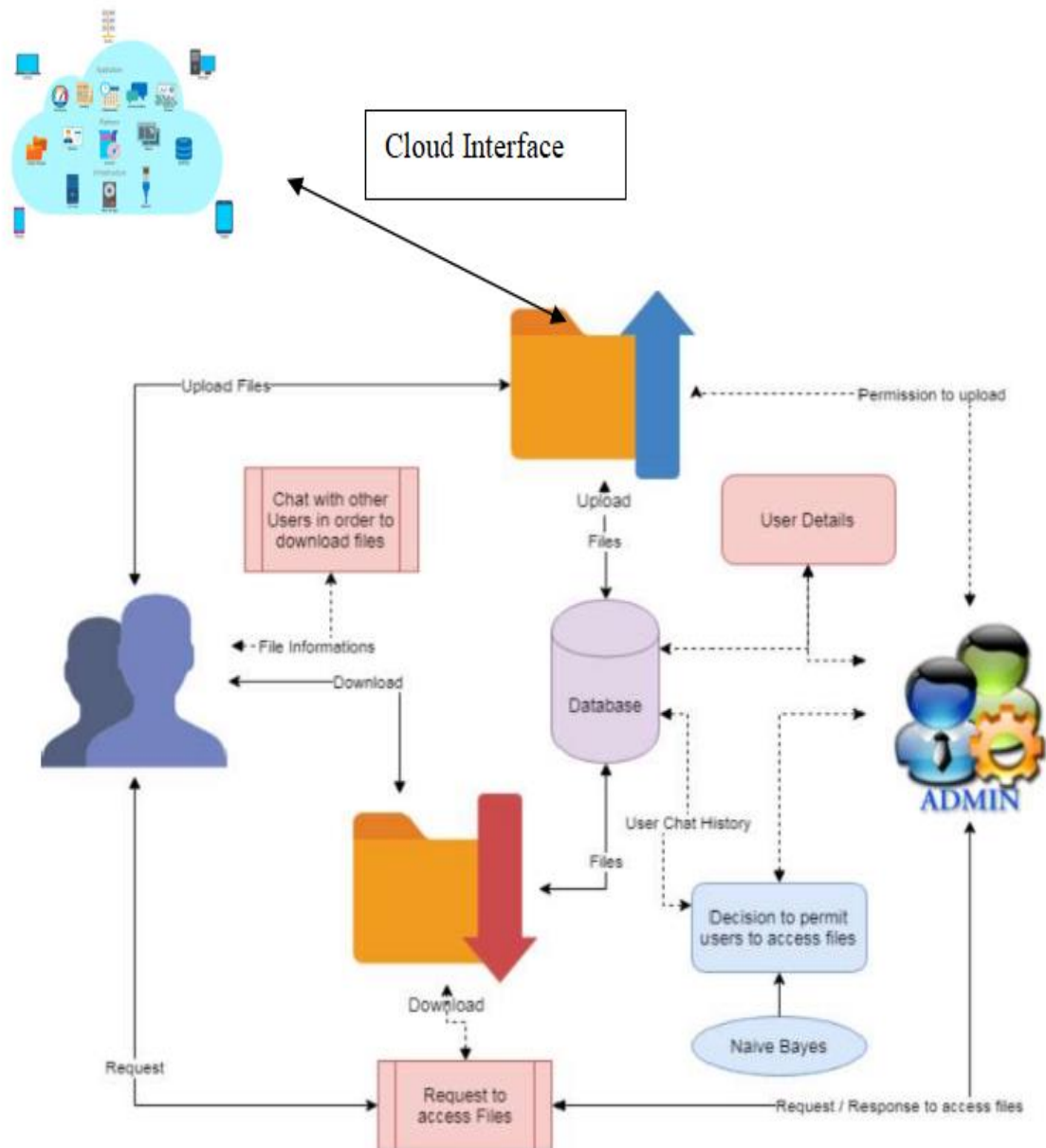
In order to design a system, first step is to collect the system requirements, functional and non functional requirements , constraints from the user. Second step is designing the system in an abstract manner, this step provides outline of all major components that is required for designing system architecture. Third step is detecting and addressing bottlenecks generated in the abstract or high-level design due to violation of some constraints specified by the user. Next operation is designing system in more elaborate and detailed manner and this step constitutes specifying the methods, parameters, interfaces to application components.

#### 4.2 High Level Diagram

High level design reveals an abstract layout of entire application where abstract HLD pictorially depicts primitive constituents of system to be developed. The architecture of the system, the diagrams depicting flow of relationship, flow of data are all considered as the high level designs and these designs are written using non-technical terms with slight additional technical terms.

##### 4.2.1 System Architecture

System Architecture of application projects a blueprint of entire system in pictorial illustration. In this project the architecture three major components: Upload ,Download and User Chat as shown in the figure 4.2.1



**Figure 4.2.1 Architecture of proposed model.**

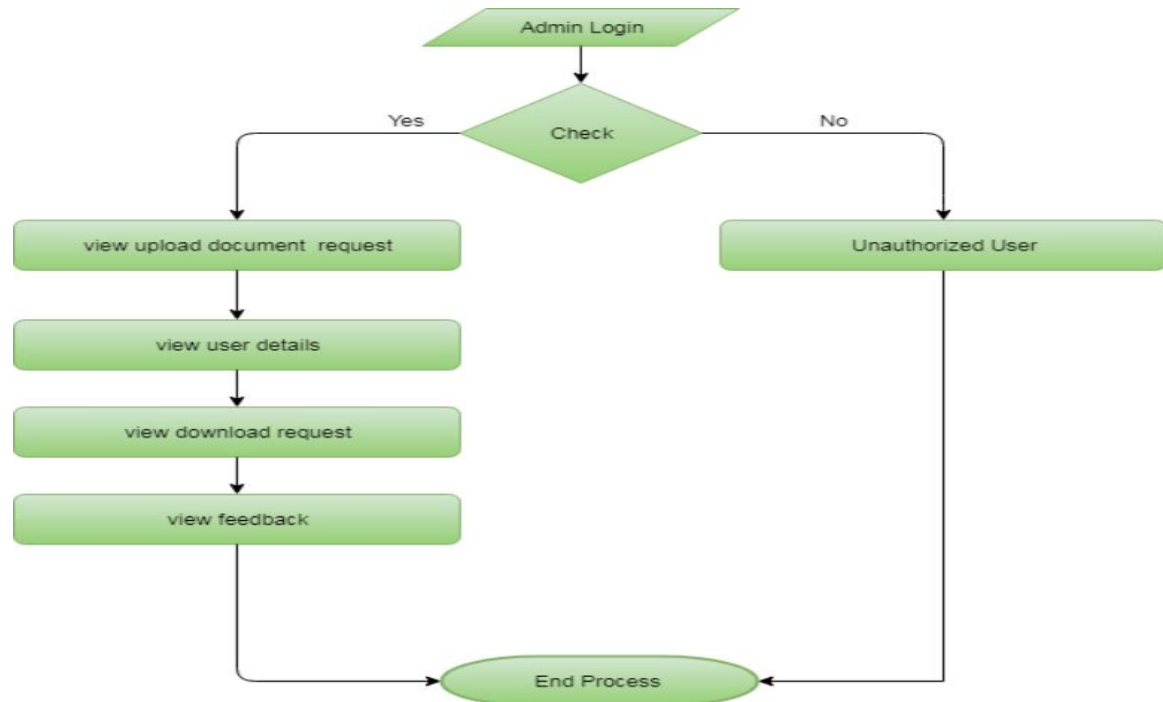
- Step 1:User upload: Users upload their data through any media to the database or cloud server.
- Step 2:Admin process: Admin response for the request and process data.
- Step 3:Decisions to provide access: Using algorithm for decision making and provide access to requested customers.
- Step 4:Download access files: Tracking of files and download the requested files.



### 5.2.2 Data Flow Diagram

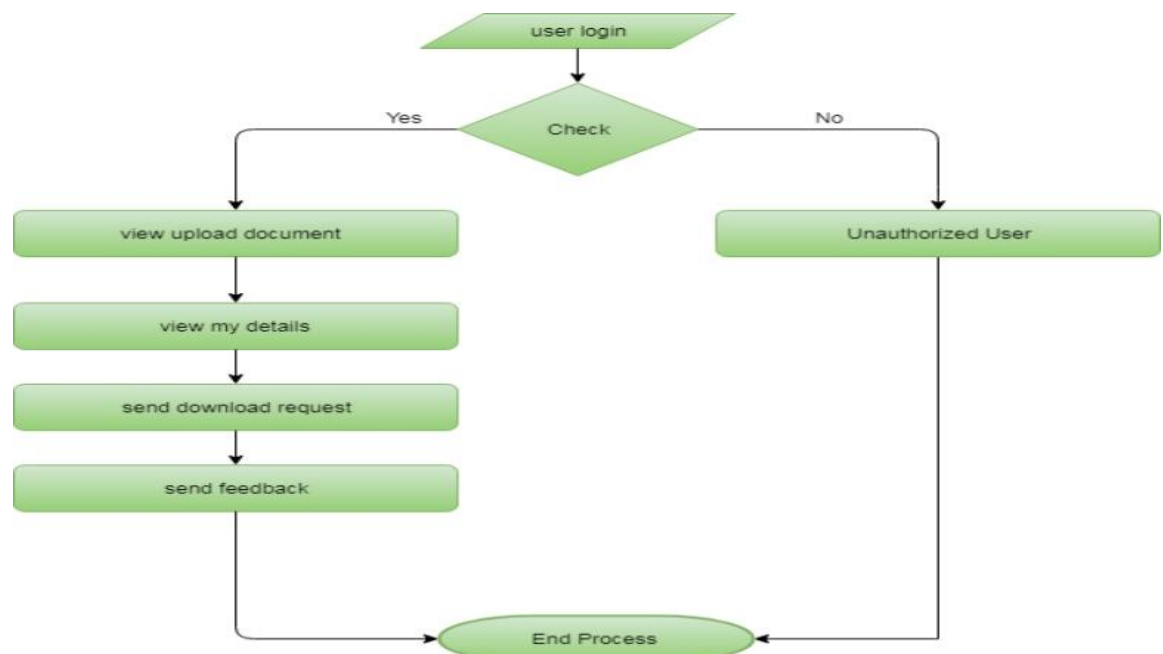
Data-flow diagram is a way of representing a flow of a data of a process or a system and information about the outputs and inputs of each entity and the process itself.

**Business Entity/Admin :**



**Figure 4.2.2(i):Data Flow diagram for admin module**

**User:**



**Figure 4.2.2(i):Data Flow Diagram for user**

### 4.3 Low Level Design

Low level design is used describing major components of system in detail and elaborate manner so this technique is also called detailed-design. In this technique the diagram is constructed by iteratively refining the given details, requirements and constraints and also depicts modules, their parameters, methods and relationship among them.

#### 4.3.1 Flow Chart

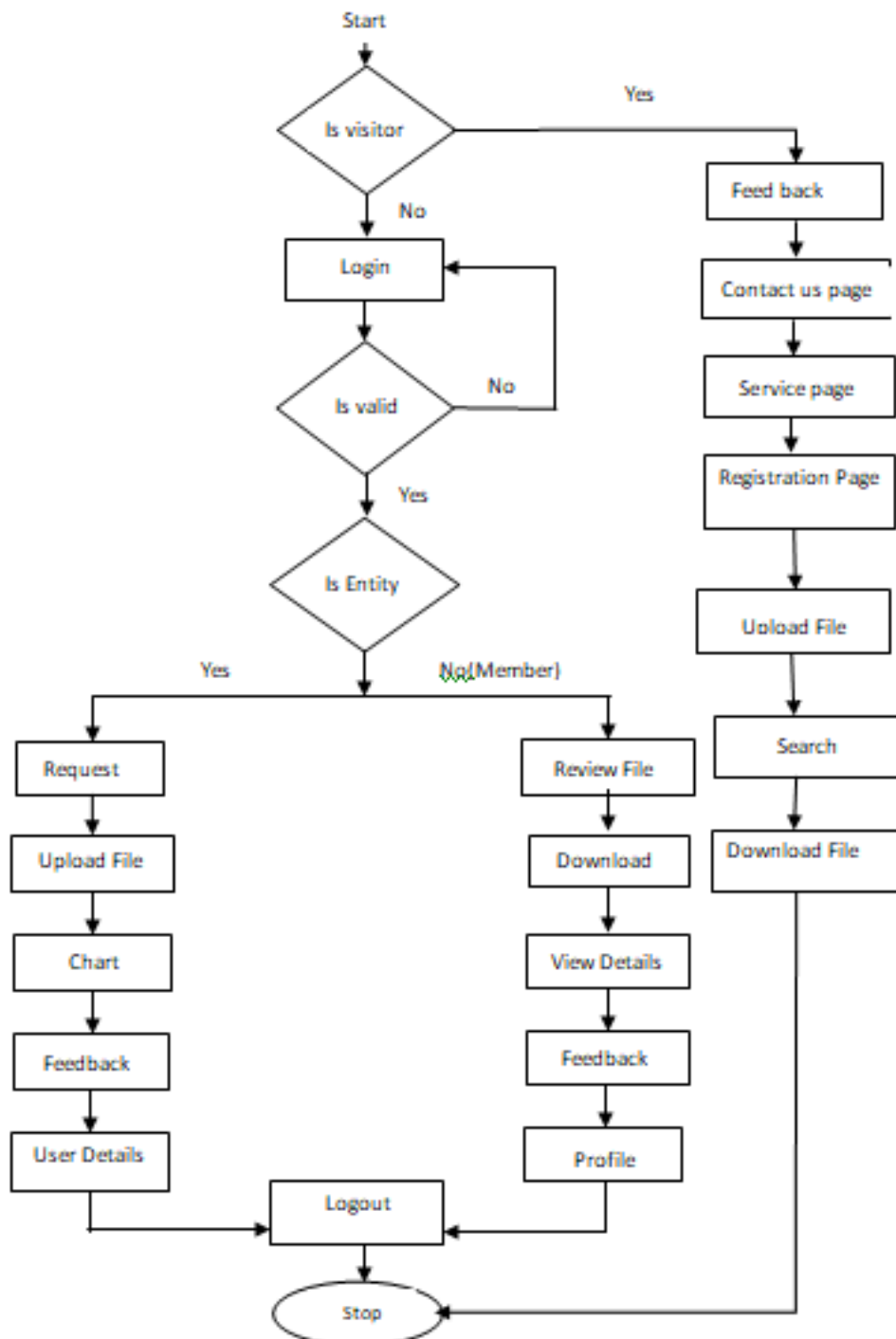


Figure 4.3.1 Flow Chart

### 4.3.2 Sequence Diagram

#### a) User

When Admin sign up successfully he/she can view, add and delete eco products category and also he/she can view ,add and delete any eco products, as well as view and respond to customer orders and can also view customer feedback for future upgradation and at last admin also have an option to update his/her password.

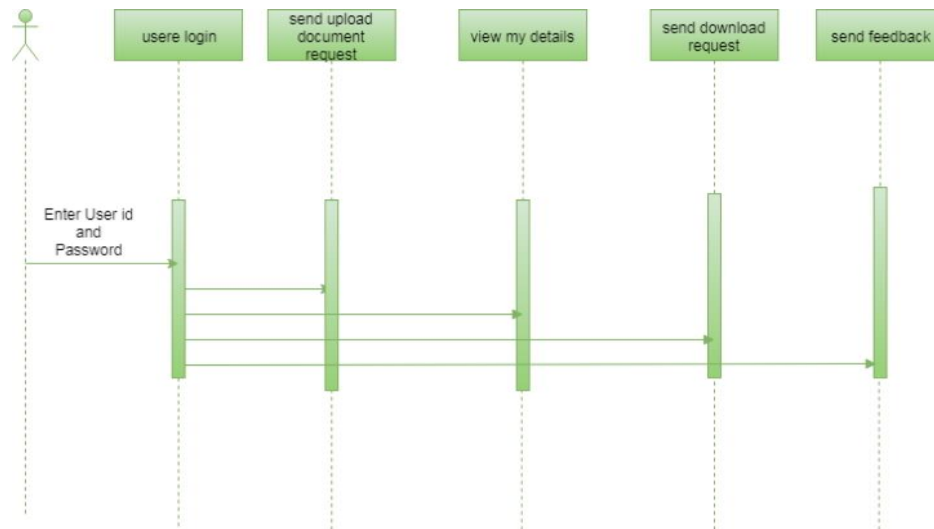


Figure 4.2.2(i): Sequence Diagram For User

#### b) Admin:

When a user gets signed up, he/she can view or delete review order and can add to place a order , also can view order history and can add his/her feedback and at last he/she can also update his/her profile

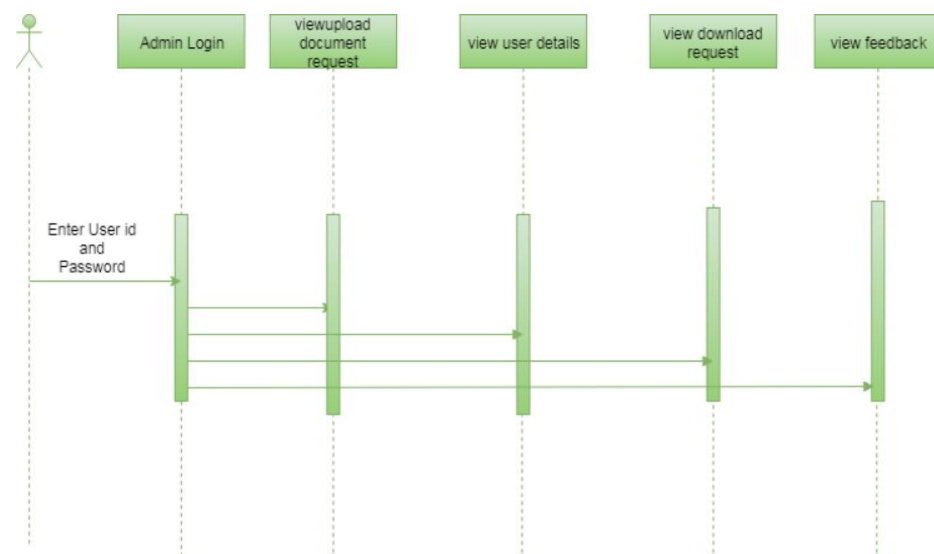


Figure 4.2.2(ii): Sequence Diagram For Admin

### 4.3.3 Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.

#### Business Entity/User:

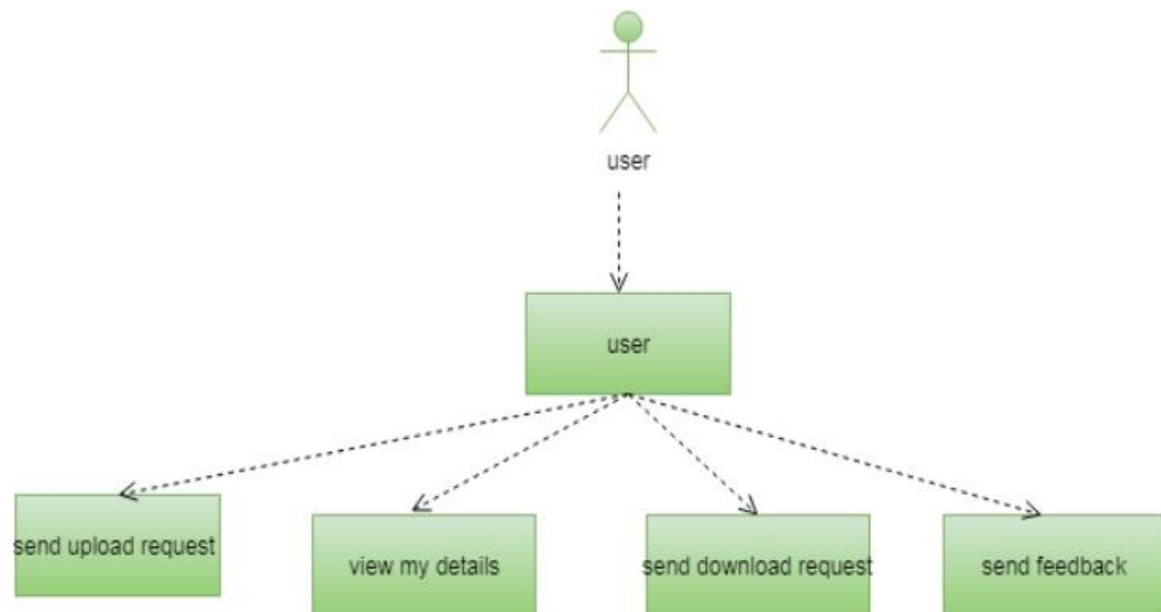


Figure 4.3.3(i) Use Case Diagram for Entity/User

#### Admin:

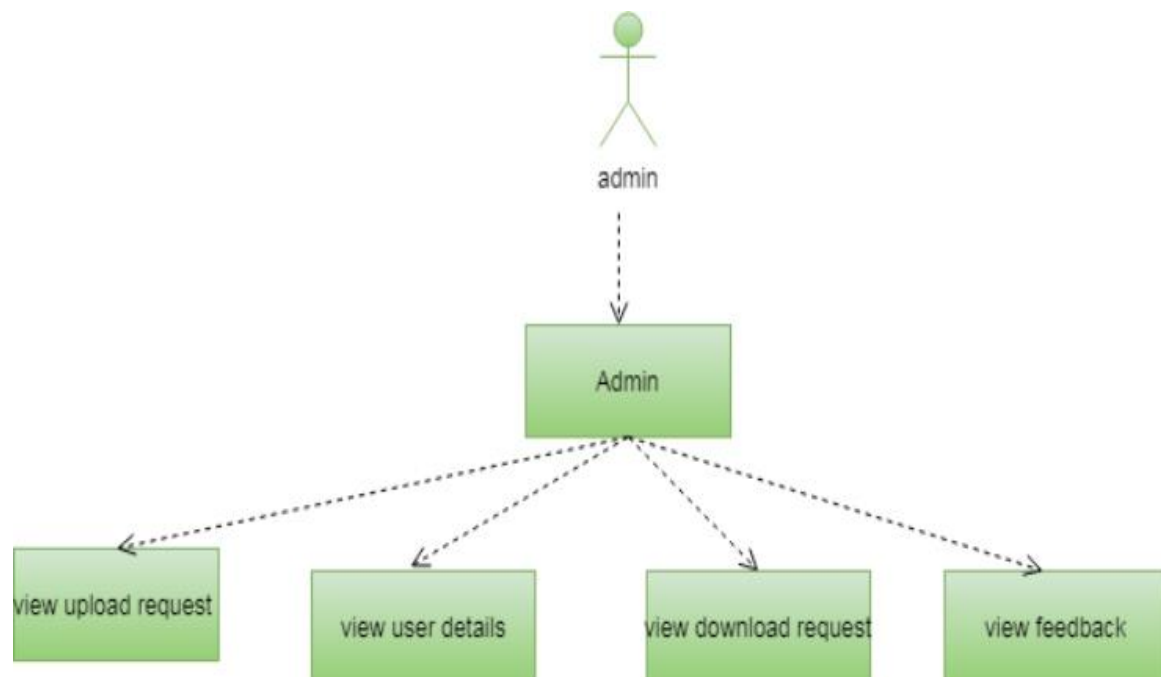


Figure 4.3.3(ii) Use Case Diagram for Entity/Admin

#### 4.3.4 Activity Diagram

Activity diagram is a kind of behavioral representation which describes how workflows in overall system and is used to describe actions, interactions and activities of system in step-by-step manner. It can also be called as a type of flowchart. Figure 5.5 showcases activity diagram which describes how the workflows between all the modules i.e job tracker, mapping process and reducing job.

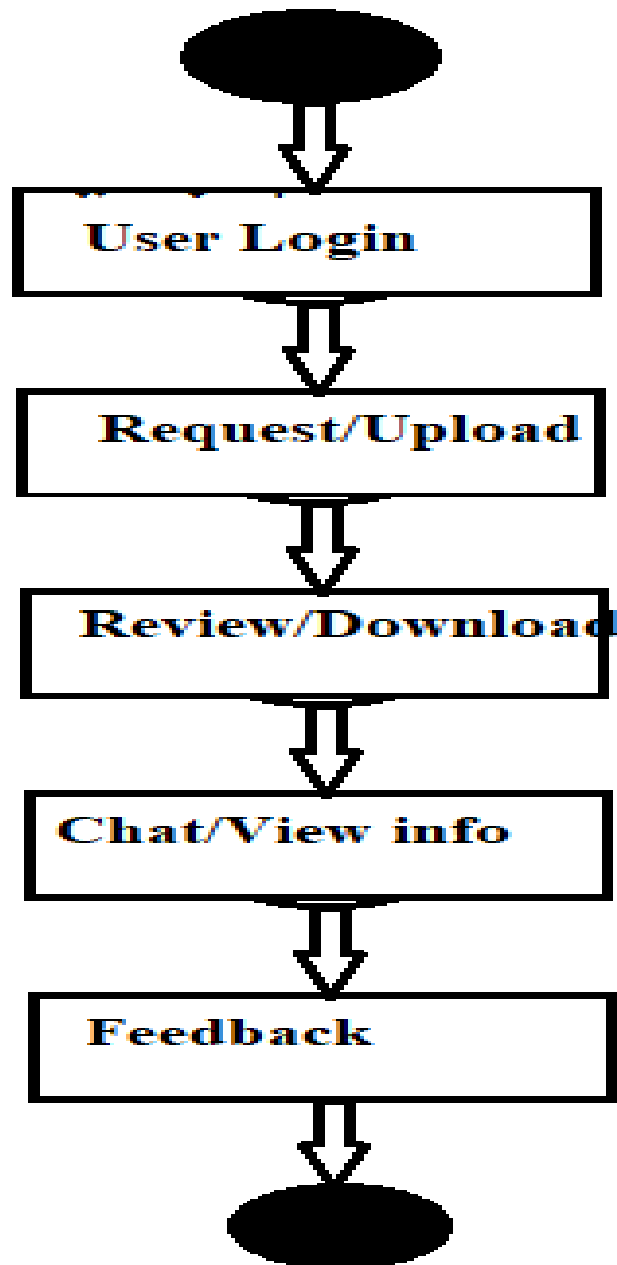


Figure 4.3.4 Activity Diagram

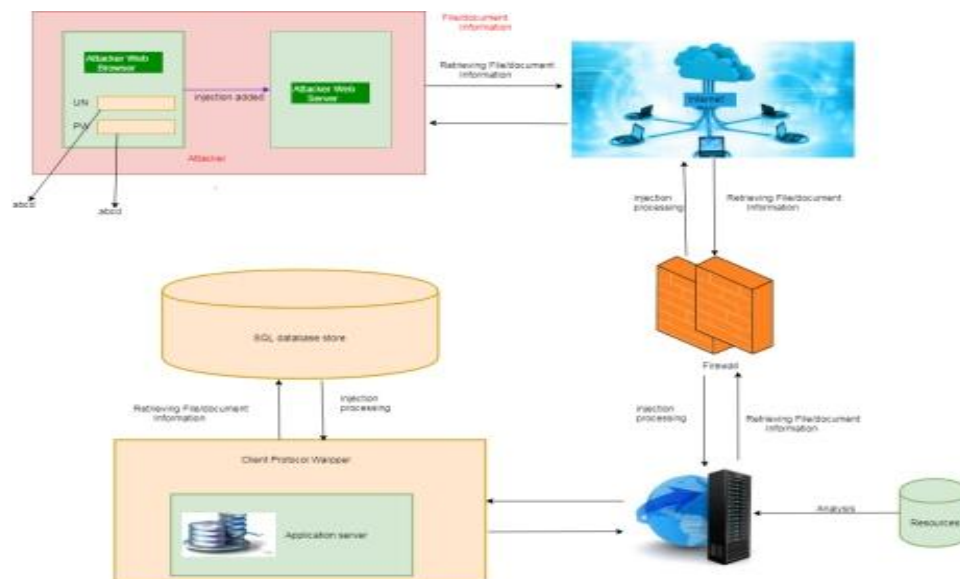
## CHAPTER 5

### SYSTEM DEVELOPMENT

#### 5.1 Introduction to System Development

This project collected data from the largest hacking community and did not consider other hacking communities. Future studies will therefore need to generalize our findings by investigating a wider range of hacking communities. Second, this study has focused on the CaaS and crimeware available in the cybercrime underground, but much in-depth analysis remains to be done on the configurations of cybercrime networks.

Implementation is one of the critical stages of the project; it is nothing but a change of working system from the theoretical system design.



**Figure 5.1 Implementing System Development**

- The attacker injects payload in website database by submitting a vulnerable form with some malicious JavaScript.
- The victim browser page loads with the payload that has been injected as a part of the HTML body.
- The script will send the victims cookie to the attacker's server. The attacker will now extract the victim's cookie when the request arrives to the server.
- Here Java Script allows the attacker to use malicious scripts to hack important details.

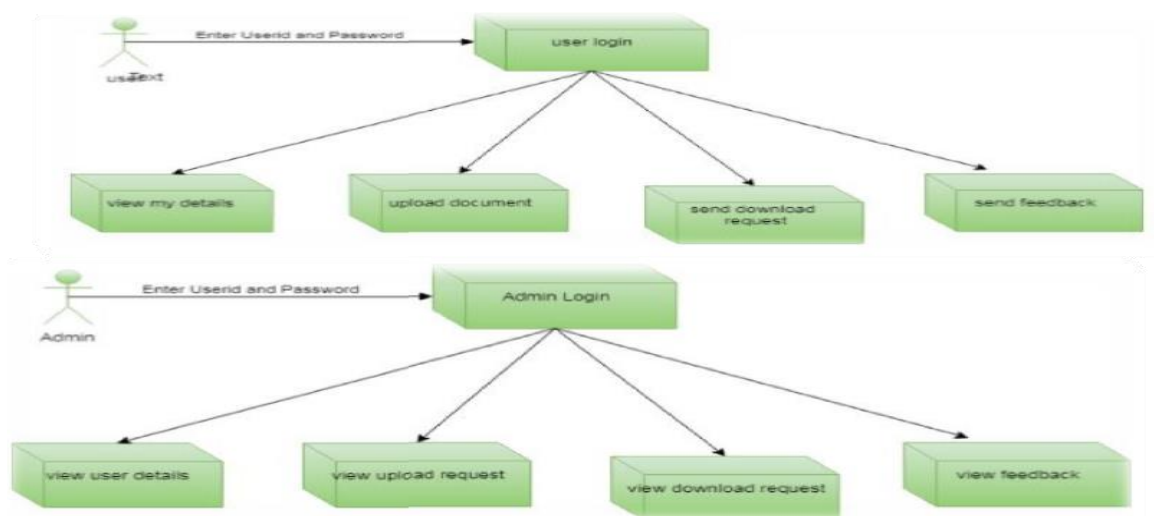
- The victim will not be directly targeted by the attacker. Instead he delivers malicious java scripts while the user visits the website. Here when the attacker injects the malicious java scripts via login page.
- He will be directly allowed to access the database of our website where he can get all important information he needed.
- Finally the user can use the victim's stolen cookie.

## 5.2 Modules and Methodology

- In contrast, the goal of our data analysis framework is to conduct a big-picture investigation of the cybercrime underground by covering all phases of data analysis from the beginning to the end .
- This framework comprises four steps:  
  
(1) defining goals; (2) identifying sources; (3) selecting analytical methods; and (4) implementing an application.
- The data analysis step of the proposed framework involves four steps. Here, we report the data analysis results: CaaS and crimeware classification and market trends, cybercrime market dynamics, and potential hacking targets.

A. CaaS and Crimeware Classification and Market Trends .

B. Cybercrime Market Dynamics



**Figure 5.2.1: System Design Modules.**

## 5.2.2 Modules



**Figure 5.2.2: Modules for Implementation**

Modules that are Implementing this system include the following,

### 1.Upload Files

Users are allowed to upload the files with the tags given. Once the file is uploaded, then it is sent to approval from admin to publish or make view to other users. These uploaded files can be in any form document, audio or video but not allowed to upload the executable (.exe) files.

### 2.Conversation Monitoring

Users are allowed to communicate among the other users. This could be monitor by the admin. The malicious conversion likes to threaten the data. In order to protect the cybercrime and prevents from forming cybercrime community. This can be achieved by the help of classification algorithm named naïve Bayes classification.

### 3.Download Files

The files can be downloading by requesting for the file and once admin approved the files then can be downloadable. The decision to approve files can be taken from the conversation between users. Admin takes the action on download files and approvable status of users. The users are allowed further actions based on the users.

### 5.Graphical Representations

The analyses of proposed systems are calculated based on the approvals and disapprovals. This can be measured with the help of graphical notations such as pie chart, bar chart and line chart. The data can be given in a dynamical data.



## CHAPTER 6

### IMPLEMENTATION

#### 6.1 Introduction to System Implementation

Implementation stage of an application creation is actualization of ideas, design and requirement specification into source code. The primary objective of implementation part of building a project is production of source codes with good style and comments when necessary, by applying a proper and best coding technique which is suitable with the help of proper documents. Program codes are created in accordance to the structured coding techniques, which adheres to control flow, so that execution sequence follows the order in which codes are scripted. This makes the code unambiguous and more readable, which eases understanding, modifying, debugging, testing, and documentation of the programs.

#### 6.2 Language Used for Implementation

##### 1.PYTHON

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java. It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many operating systems. Python, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library. Current Python version using for project is python3.x

##### 2.DJANGO

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to

reinvent the wheel. It's free and open source. Django's primary goal is to ease the creation of complex, database-driven websites. Django emphasizes [reusability](#) and "pluggability" of components, rapid development, and the principle of [don't repeat yourself](#). Python is used throughout, even for settings files and data models.

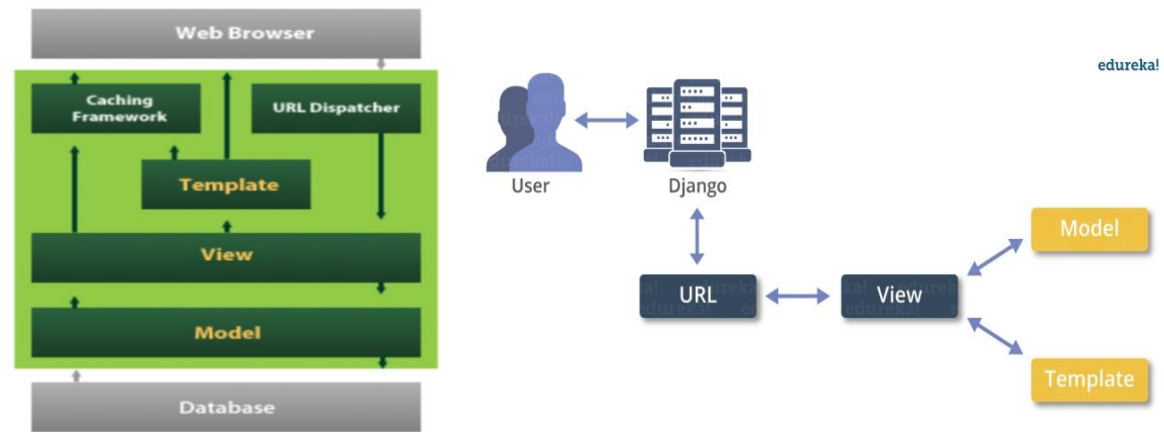


Figure 6.2.2 Django web framework

Django also provides an optional administrative [create, read, update and delete](#) interface that is generated dynamically through [introspection](#) and configured via admin models. Current version used is Django2.2.5

### 6.3 Algorithm: Naive Bayes Classifier

Naive Bayes is a classification algorithm for binary (two-class) and multi-class classification problems. The technique is easiest to understand when described using binary or categorical input values. It is called naive Bayes or idiot Bayes because the calculation of the probabilities for each hypothesis is simplified to make their calculation tractable. Rather than attempting to calculate the values of each attribute value  $P(d_1, d_2, d_3|h)$ , they are assumed to be conditionally independent given the target value and calculated as  $P(d_1|h) * P(d_2|h)$  and so on. This is a very strong assumption that is most unlikely in real data, i.e. that the attributes do not interact. Nevertheless, the approach performs surprisingly well on data where this assumption does not hold.

Make Predictions with a Naive Bayes Model. Given a naive Bayes model, you can make predictions for new data using Bayes theorem.

$$\text{MAP}(h) = \max(P(d|h) * P(h))$$

Using our example above, if we had a new instance with the weather of sunny, we can calculate:

$$\text{go-out} = P(\text{weather=sunny} \mid \text{class=go-out}) * P(\text{class=go-out}) \quad \text{stay-home} = P(\text{weather=sunny} \mid \text{class=stay-home}) * P(\text{class=stay-home})$$

We can choose the class that has the largest calculated value. We can turn these values into probabilities by normalizing them as follows:

$$P(\text{go-out} \mid \text{weather=sunny}) = \text{go-out} / (\text{go-out} + \text{stay-home}) \quad P(\text{stay-home} \mid \text{weather=sunny}) = \text{stay-home} / (\text{go-out} + \text{stay-home})$$

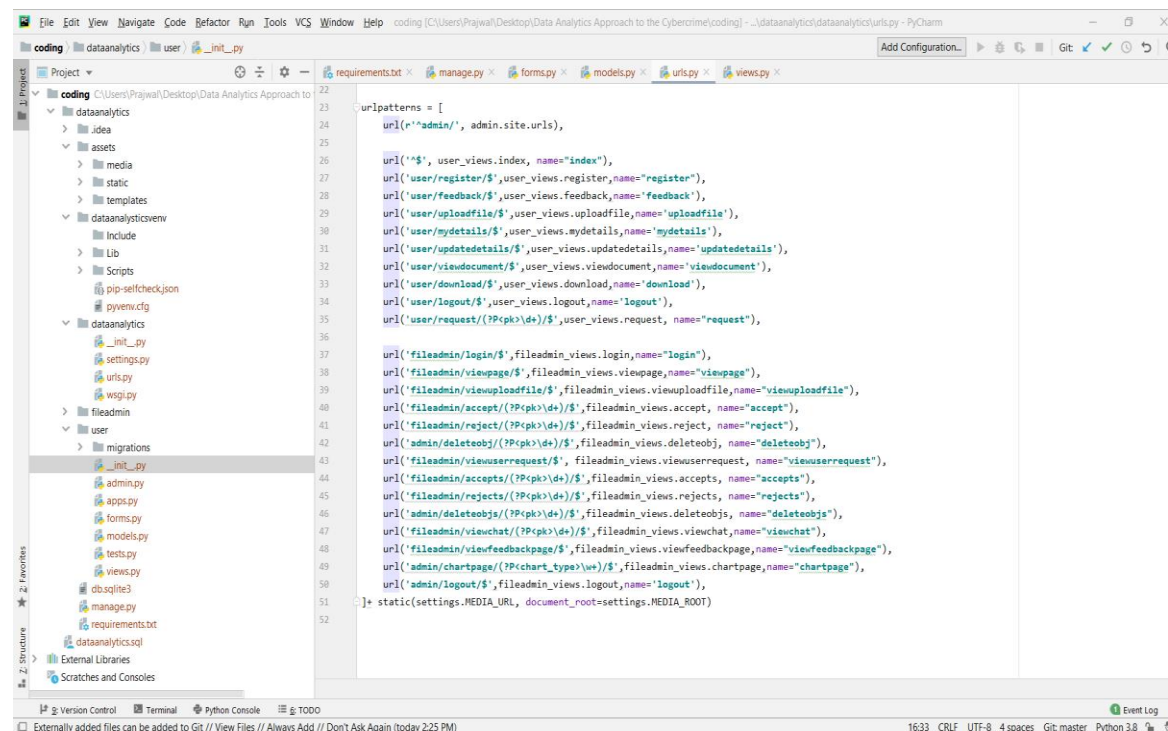
If we had more input variables we could extend the above example. For example, pretend we have a “car” attribute with the values “working” and “broken”.

We can multiply this probability into the equation. For example below is the calculation for the “go-out” class label with the addition of the car input variable set to “working”:

$$\text{go-out} = P(\text{weather=sunny} \mid \text{class=go-out}) * P(\text{car=working} \mid \text{class=go-out}) * P(\text{class=go-out})$$

## 6.4 Snapshot of Code Snippet

### 6.4.1 urls.py



```

22 urlpatterns = [
23     url(r'^admin/', admin.site.urls),
24
25     url(r'^$', user_views.index, name="index"),
26     url('user/register/$', user_views.register, name="register"),
27     url('user/feedback/$', user_views.feedback, name="feedback"),
28     url('user/uploadfile/$', user_views.uploadfile, name="uploadfile"),
29     url('user/mydetails/$', user_views.mydetails, name="mydetails"),
30     url('user/updatedetails/$', user_views.updatedetails, name="updatedetails"),
31     url('user/viewdocument/$', user_views.viewdocument, name="viewdocument"),
32     url('user/download/$', user_views.download, name="download"),
33     url('user/logout/$', user_views.logout, name="logout"),
34     url('user/request/(?P<pk>[0-9]+)/$', user_views.request, name="request"),
35
36     url('fileadmin/login/$', fileadmin_views.login, name="login"),
37     url('fileadmin/viewpage/$', fileadmin_views.viewpage, name="viewpage"),
38     url('fileadmin/viewuploadfile/$', fileadmin_views.viewuploadfile, name="viewuploadfile"),
39     url('fileadmin/accept/(?P<pk>[0-9]+)/$', fileadmin_views.accept, name="accept"),
40     url('fileadmin/reject/(?P<pk>[0-9]+)/$', fileadmin_views.reject, name="reject"),
41     url('admin/deleteobj/(?P<pk>[0-9]+)/$', fileadmin_views.deleteobj, name="deleteobj"),
42     url('fileadmin/viewuserrequest/$', fileadmin_views.viewuserrequest, name="viewuserrequest"),
43     url('fileadmin/accepts/(?P<pk>[0-9]+)/$', fileadmin_views.accepts, name="accepts"),
44     url('fileadmin/rejects/(?P<pk>[0-9]+)/$', fileadmin_views.rejects, name="rejects"),
45     url('admin/deleteobj/(?P<pk>[0-9]+)/$', fileadmin_views.deleteobj, name="deleteobj"),
46     url('fileadmin/viewchat/(?P<pk>[0-9]+)/$', fileadmin_views.viewchat, name="viewchat"),
47     url('fileadmin/viewfeedbackpage/$', fileadmin_views.viewfeedbackpage, name="viewfeedbackpage"),
48     url('admin/chartpage/(?P<chart_type>[0-9]+)/$', fileadmin_views.chartpage, name="chartpage"),
49     url('admin/logout/$', fileadmin_views.logout, name="logout"),
50 ]
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2
```

## 6.4.2 models.py

```

1  from django.db import models
2
3  class User(models.Model):
4      first_name = models.CharField(max_length=100)
5      last_name = models.CharField(max_length=100)
6      dob = models.CharField(max_length=100)
7      age = models.CharField(max_length=100)
8      user_id = models.CharField(max_length=100)
9      password = models.CharField(max_length=100)
10     mobile_number = models.CharField(max_length=100)
11     email_id = models.CharField(max_length=100)
12     gender = models.CharField(max_length=100)
13
14     class UploadModel(models.Model):
15         user_id = models.ForeignKey(User, on_delete=models.CASCADE)
16         name = models.CharField(max_length=100)
17         topic = models.CharField(max_length=100)
18         document = models.FileField()
19         date = models.CharField(max_length=100)
20         request = models.CharField(max_length=100, default='Pending')
21
22     class ChatModel(models.Model):
23         user_id = models.ForeignKey(User, on_delete=models.CASCADE)
24         sender_id = models.CharField(max_length=100)
25         chat = models.CharField(max_length=300)
26
27     class RequestModel(models.Model):
28         access_token = models.CharField(max_length=100)
29         access_token = models.CharField(max_length=100)
30         request = models.CharField(max_length=200, default='pending')
31
32     class FeedbackModel(models.Model):
33         name = models.CharField(max_length=300)
34         mobile_number = models.CharField(max_length=300)
35         email_id = models.CharField(max_length=300)
36         feedback = models.CharField(max_length=300)
37
38     class ChatModel(models.Model):
39         user_id = models.ForeignKey(User, on_delete=models.CASCADE)
40         sender_id = models.CharField(max_length=100)
41         chat = models.CharField(max_length=300)

```

Figure 6.5.2 models.py

## 6.4.3 views.py

```

1  from django.shortcuts import render
2  from django.contrib.auth import login, authenticate
3  from django.contrib.auth.forms import UserCreationForm
4  from django.contrib.auth import login, authenticate
5  from django.contrib.auth import login, authenticate
6  from django.contrib.auth import login, authenticate
7  from django.contrib.auth import login, authenticate
8  from django.contrib.auth import login, authenticate
9  from django.contrib.auth import login, authenticate
10     from django.contrib.auth import login, authenticate
11     from django.contrib.auth import login, authenticate
12     from django.contrib.auth import login, authenticate
13     from django.contrib.auth import login, authenticate
14     from django.contrib.auth import login, authenticate
15     from django.contrib.auth import login, authenticate
16     from django.contrib.auth import login, authenticate
17     from django.contrib.auth import login, authenticate
18     from django.contrib.auth import login, authenticate
19     from django.contrib.auth import login, authenticate
20     from django.contrib.auth import login, authenticate
21     from django.contrib.auth import login, authenticate
22     from django.contrib.auth import login, authenticate
23     from django.contrib.auth import login, authenticate
24     from django.contrib.auth import login, authenticate
25     from django.contrib.auth import login, authenticate
26     from django.contrib.auth import login, authenticate
27     from django.contrib.auth import login, authenticate
28     from django.contrib.auth import login, authenticate
29     from django.contrib.auth import login, authenticate
30     from django.contrib.auth import login, authenticate
31     from django.contrib.auth import login, authenticate
32     from django.contrib.auth import login, authenticate
33     from django.contrib.auth import login, authenticate
34     from django.contrib.auth import login, authenticate
35     from django.contrib.auth import login, authenticate
36     from django.contrib.auth import login, authenticate
37     from django.contrib.auth import login, authenticate
38     from django.contrib.auth import login, authenticate
39     from django.contrib.auth import login, authenticate
40     from django.contrib.auth import login, authenticate
41     from django.contrib.auth import login, authenticate
42     from django.contrib.auth import login, authenticate
43     from django.contrib.auth import login, authenticate
44     from django.contrib.auth import login, authenticate
45     from django.contrib.auth import login, authenticate
46     from django.contrib.auth import login, authenticate
47     from django.contrib.auth import login, authenticate
48     from django.contrib.auth import login, authenticate
49     from django.contrib.auth import login, authenticate
50     from django.contrib.auth import login, authenticate
51     from django.contrib.auth import login, authenticate
52     from django.contrib.auth import login, authenticate
53     from django.contrib.auth import login, authenticate
54     from django.contrib.auth import login, authenticate
55     from django.contrib.auth import login, authenticate
56     from django.contrib.auth import login, authenticate
57     from django.contrib.auth import login, authenticate
58     from django.contrib.auth import login, authenticate
59     from django.contrib.auth import login, authenticate
60     from django.contrib.auth import login, authenticate
61     from django.contrib.auth import login, authenticate
62     from django.contrib.auth import login, authenticate
63     from django.contrib.auth import login, authenticate
64     from django.contrib.auth import login, authenticate
65     from django.contrib.auth import login, authenticate
66     from django.contrib.auth import login, authenticate
67     from django.contrib.auth import login, authenticate
68     from django.contrib.auth import login, authenticate
69     from django.contrib.auth import login, authenticate
70     from django.contrib.auth import login, authenticate
71     from django.contrib.auth import login, authenticate
72     from django.contrib.auth import login, authenticate
73     from django.contrib.auth import login, authenticate
74     from django.contrib.auth import login, authenticate
75     from django.contrib.auth import login, authenticate
76     from django.contrib.auth import login, authenticate
77     from django.contrib.auth import login, authenticate
78     from django.contrib.auth import login, authenticate
79     from django.contrib.auth import login, authenticate
80     from django.contrib.auth import login, authenticate
81     from django.contrib.auth import login, authenticate
82     from django.contrib.auth import login, authenticate
83     from django.contrib.auth import login, authenticate
84     from django.contrib.auth import login, authenticate
85     from django.contrib.auth import login, authenticate
86     from django.contrib.auth import login, authenticate
87     from django.contrib.auth import login, authenticate
88     from django.contrib.auth import login, authenticate
89     from django.contrib.auth import login, authenticate
90     from django.contrib.auth import login, authenticate
91     from django.contrib.auth import login, authenticate
92     from django.contrib.auth import login, authenticate
93     from django.contrib.auth import login, authenticate
94     from django.contrib.auth import login, authenticate
95     from django.contrib.auth import login, authenticate
96     from django.contrib.auth import login, authenticate
97     from django.contrib.auth import login, authenticate
98     from django.contrib.auth import login, authenticate
99     from django.contrib.auth import login, authenticate
100    from django.contrib.auth import login, authenticate

```

Figure 6.5.3 views.py

## **CHAPTER 7**

### **TESTING**

#### **7.1 Introduction to Testing**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

#### **7.2 TYPES OF TESTS**

##### **7.2.1 Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

##### **7.2.2 Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successful unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

##### **7.2.3 Functional Test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:

- Valid Input : identified classes of valid input must be accepted.
- Invalid Input : identified classes of invalid input must be rejected.
- Functions : identified functions must be exercised.
- Output : identified classes of application outputs must be exercised.
- Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

### **i. System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

### **ii. White Box Testing**

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

### **iii. Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

## **7.2 Test Cases Table**

Test case ID	Description	Test steps	Expected value	Actual value	OK/Error
1.	Verify login page	Input username and password	Login page	Invalid data	error
2	Verify login page	Input username and password	Login page	Login page	ok
3.	Verify registration page	Registration	User profile	Registration failed	error
6.	Verify registration page	Registration	User profile	User profile	ok
5.	Query is to be posted	Posting query	Query is posted	Unable to post the query	error
6.	Query is to be posted	Posting query	Query is posted	Successfully posted	ok
7.	Files to be upload	Upload file	File to upload	File name is entered	ok
8.	Files to be upload	Upload file	File to upload	Unknown file	error
9.	Files to be requested	Enter request file	File to be send	File name is Request	ok
10.	Files to be reuested	Enter request file	File to be send	File not found	error
11.	Chat Module	Enter text	Negative Text is entered	Possitive Text is entered	error
12.	Chat Module	Enter text	Possitive Text is entered	Possitive Text is entered	ok

**Table 7.2 Test cases**



## CHAPTER 8

### RESULT ANALYSIS

#### 8.1 Snapshot with Description

##### 8.1.1 Login Page/Home Page :



Figure 8.1.1 Login Page

- Home Page for User or Admin Login to the System

##### 8.1.2 New User Registration:

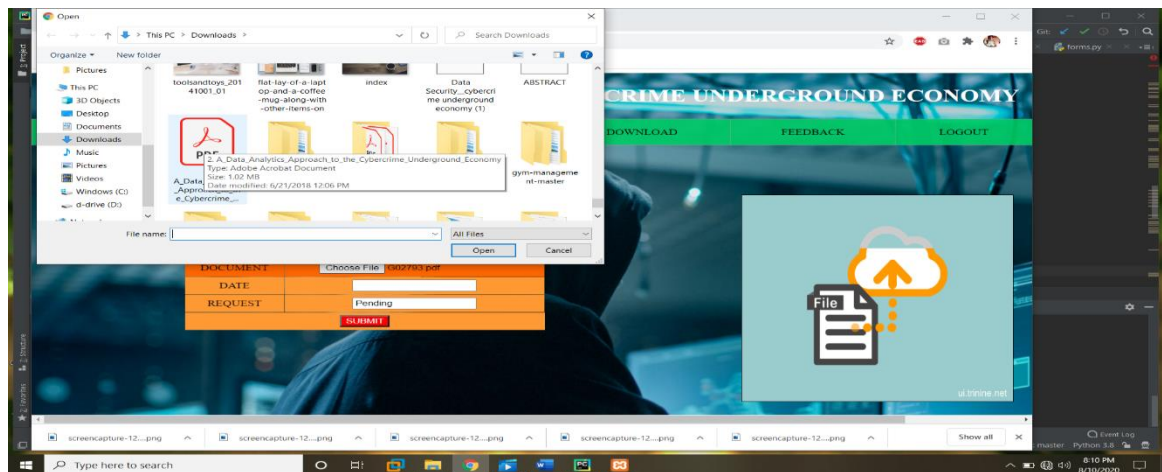


Figure 8.1.2 Registration Page

- Registration Page for New Users by entering their basic information.



### 8.1.3 Upload File Page:



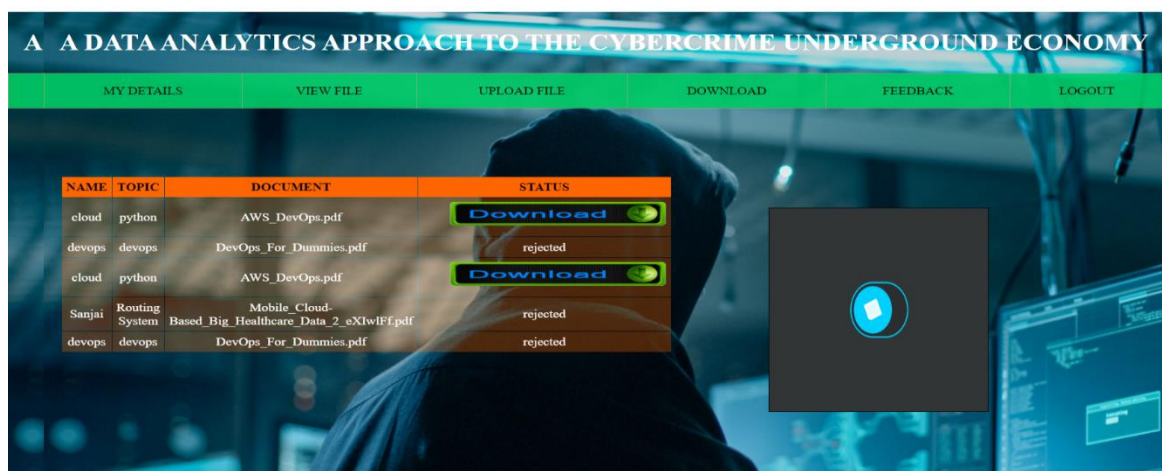
**Figure 8.1.3 Upload File Page**

### 8.1.4 View File Uploaded Page:



**Figure 8.1.4 View File Uploaded Page**

### 8.1.5 Download Page :



**Figure 8.1.5 Download Page**

### 8.1.6 Admin View Uploaded Page :

A DATA ANALYTICS APPROACH TO THE CYBERCRIME UNDERGROUND ECONOMY						
USER DETAILS		VIEW UPLOAD FILE		USER REQUEST		LOGOUT
FIRST NAME	LAST NAME	DATE OF BIRTH	AGE	USER ID	MOBILE NUMBER	EMAIL ID
Sabari	Nathan	09-05-1997	21	Sabari	9789672189	sabarinathan@1350@gmail.com
Sanjai	Kumar	04-11-1996	22	Sanjai	7853974352	sanjai@gmail.com
Santhosh	Kumar	05-01-1997	21	Santhosh	9674322516	santhosh23@gmail.com
Suresh	Babu	06-11-1997	21	Suresh	8694326754	suresh@gmail.com
Somesh	Varan	08-05-1995	23	Somesh	9543742376	sabari43@gmail.com
Karthik	Raja	06-02-1990	28	Karthik	945173275	karthik@gmail.com
Soundhar	Rajan	05-03-1996	22	Soundhar	9145725427	soundhar@gmail.com
Vignesh	Varan	08-04-1992	26	Vignesh	7459287125	vignesh54@gmail.com
Gokul	Raj	05-02-1994	24	Gokul	9441853851	gokul32@gmail.com
Mani	Kandan	08-01-1989	29	Mani	9521677421	mani765@gmail.com
user	user	00/00/0000	21	1	9999999999	user@gmail.com
user	user	00/00/0000	00	123	9999999999	user@gmail.com

**Figure 8.1.6 Admin View Uploaded Page**

### 8.1.7 Admin View Request Page :

A DATA ANALYTICS APPROACH TO THE CYBERCRIME UNDERGROUND ECONOMY						
USER DETAILS		VIEW UPLOAD FILE		USER REQUEST		LOGOUT
NAME	TOPIC	DOCUMENT	STATUS	CHAT	REQUEST STATUS	DELETE
Sabari	Keyset	m1_et4eDAN.txt	accepted	chat	Accept Reject	delete
Sanjai	Python	Registration_ID_FXgbU9T.txt	accepted	chat	Accept Reject	delete
Sabari	Keyset	m1_et4eDAN.txt	accepted	chat	Accept Reject	delete
Santhosh	Java	SIVA_xvIMTG7	accepted	chat	Accept Reject	delete
Gokul	Python	Prediction_Based_Mobiel_Data_Offloading_1.pdf	accepted	chat	Accept Reject	delete
Karthik	Security	Coded_Caching_Clusters.pdf	accepted	chat	Accept Reject	delete
Sanjai	Routing System	Mobile_Cloud-Based_Big_Healthcare_Data_2_eXIwIFF.pdf	accepted	chat	Accept Reject	delete
Mani	Java	Coded_Caching_Clusters_oYBb9Xm.pdf	accepted	chat	Accept Reject	delete
Santhosh	Java	SIVA_xvIMTG7	pending	chat	Accept Reject	delete
Gokul	Hacking	SS_QR_CODE_2.pdf	pending	chat	Accept Reject	delete
Sanjai	Python	Registration_ID_FXgbU9T.txt	accepted	chat	Accept Reject	delete

**Figure 8.1.7 Admin view Request Page**

### 8.1.8 Feedback Page :

A DATA ANALYTICS APPROACH TO THE CYBERCRIME UNDERGROUND ECONOMY			
USER DETAILS		VIEW UPLOAD FILE	
NAME	MOBILE NUMBER	EMAIL ID	FEEDBACK
Santhosh	9789672189	santhosh43@gmail.com	Thank you,it's very useful for me
Santhosh	9789672189	santhosh43@gmail.com	Thank you,it's very useful for me
Sanjai Kumar	7493674328	talktonoo@gmail.com	it's very well
Suresh Babu	9451734529	smartsuresh@gmail.com	Improve the data base system
Sabari Nathan	9523645741	sabari345@gmail.com	It's good effort
aravana Kumar	9451754336	saravanan12@gmail.com	It's very well
user	9999999999	user@gmail.com	Sir uploading my file

**Figure 8.1.8 Feedback Page**



### 8.1.9 Positive Conversation Page :

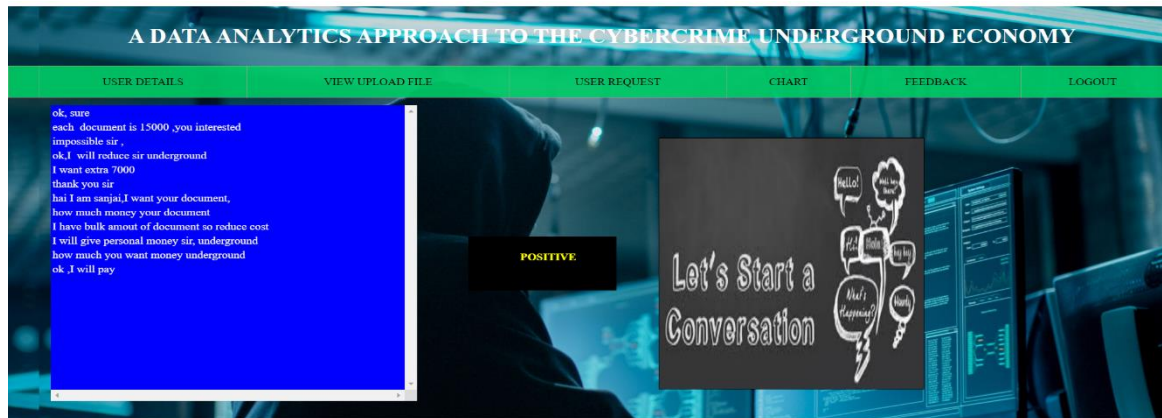


Figure 8.1.9 Positive Conversation Page

### 8.1.10 Negative Conversation Page :

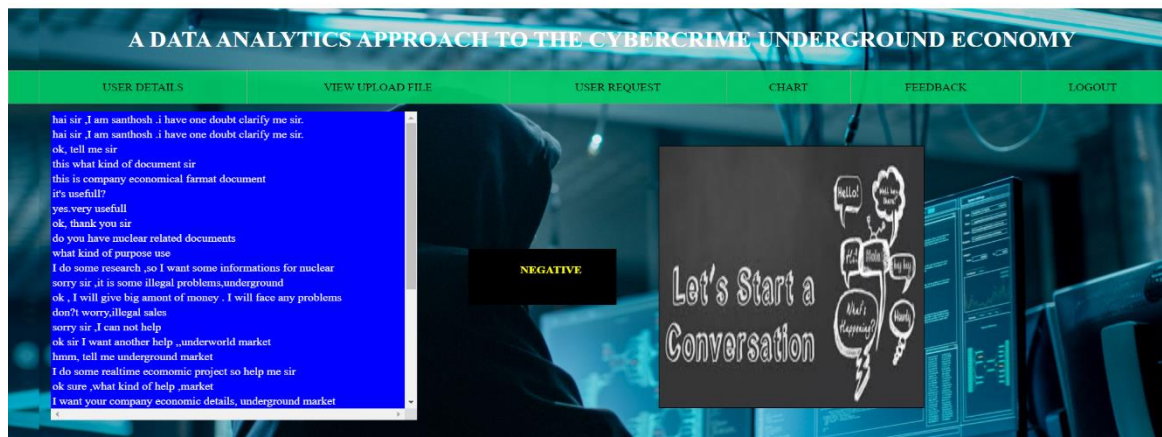


Figure 8.1.10 Negative Conversation Page

### 8.1.11 CSV file in Excel conversation :

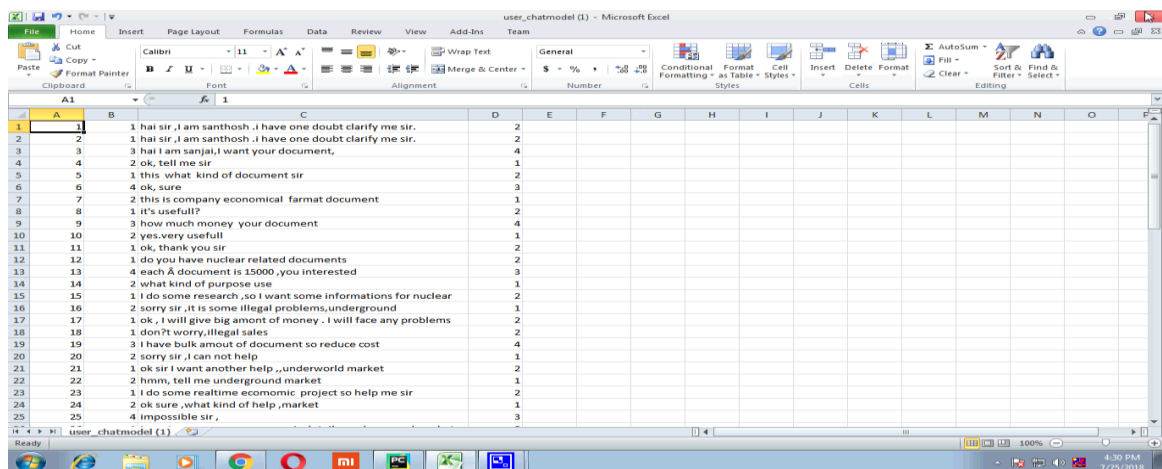
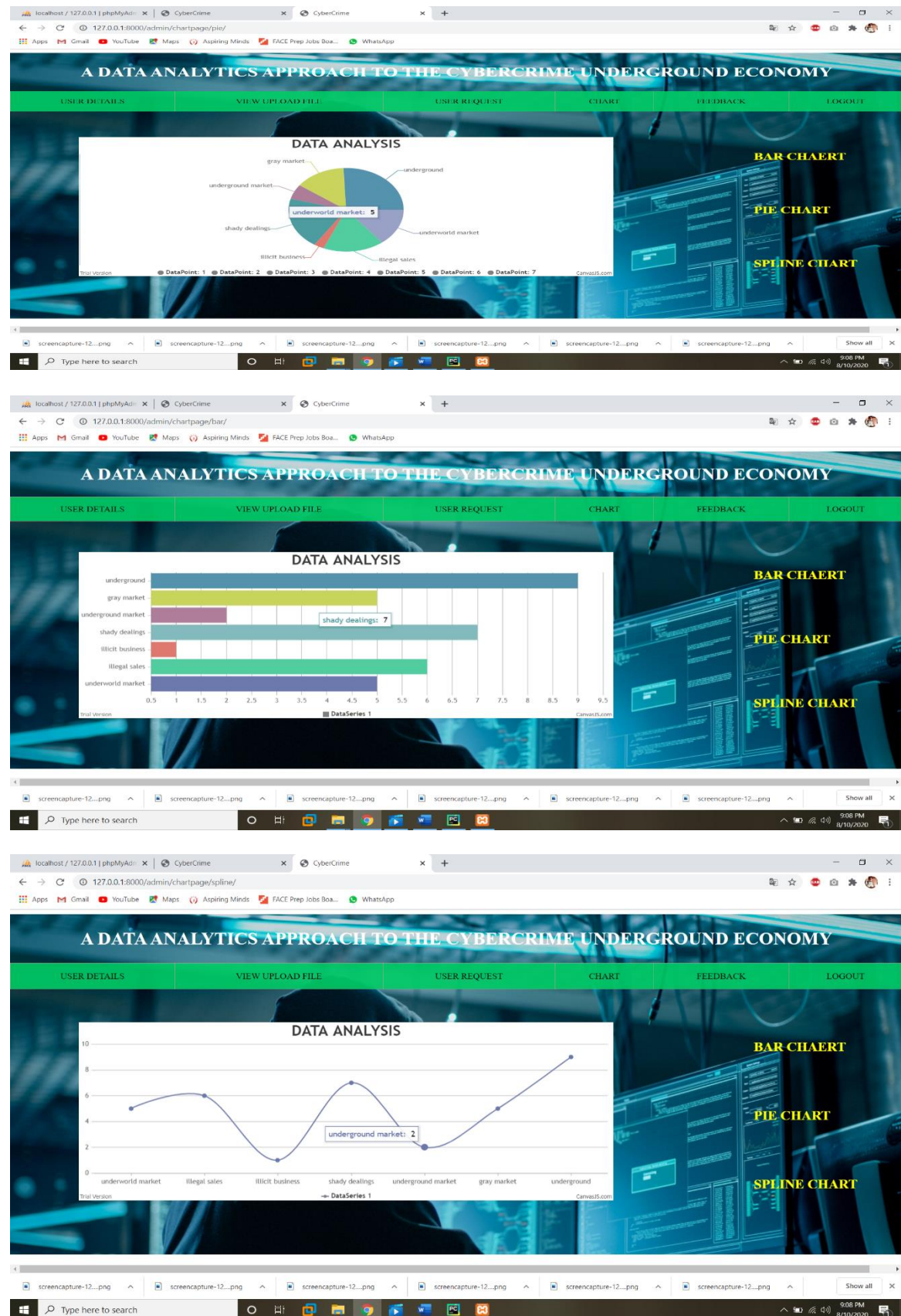


Figure 8.1.11 CSV File in Excel

### 8.1.12 Data Analysis Graph:



**Figure 8.1.12 Data Analysis Graph Chart (Bar,Pie,Spline chart)**

## **CONCLUSION**

This study contributes to the DSR literature in a broader IS context in several ways. Because it takes a DSR approach, it contributes to the design artifacts, foundations, and methodologies in this area. First, by creating example front-end applications, we have demonstrated how our design artifacts (the proposed framework and classification model) can be implemented in practice. We have therefore proposed two artifacts: a data analysis framework and a classification model. We have also conducted an ex-ante evaluation of our classification model's accuracy and an ex-post evaluation of its implementation using example applications. In line with the initiation perspective of DSR, these four example applications demonstrate the range of potential practical applications available to future researchers and practitioners. Unlike previous studies that have presented general discussions of a broad range of cybercrime; our study has focused primarily on CaaS and crime ware from an RAT perspective. We have also proposed sets of definitions for different types of CaaS (phishing, brute force attack, DDoS attack, and spamming, crypting, and VPN services) and crime ware (drive-by download, botnets, exploits, ransomware, rootkits, Trojans, crypters , and proxies) based on definitions taken from both the academic and business practice literature.

Looking at the CaaS and crimeware trends, our results show that the prevalence of botnets (attack-related crimeware) and VPNs has increased in 2017. This indicates that attackers consider both the preventive measures taken by organizations and their vulnerabilities. The most common potential target organizations are technology companies (28%), followed by content (22%), finance (20%), e-commerce (12%), and telecommunication (10%) companies. This indicates that a wide variety of companies in a range of industries are becoming potential targets for attackers, having become more vulnerable due to their greater reliance on technology.

## FUTURE WORK

Although our study has made several significant findings, it nevertheless has several limitations that will need to be addressed in future studies. These will be able to add more analysis and significant further insights.

**First**, we only collected data from the largest hacking community and did not consider other hacking communities. Future studies will therefore need to generalize our findings by investigating a wider range of hacking communities.

**Second**, this study has focused on the CaaS and crimeware available in the cybercrime underground, but much in-depth analysis remains to be done on the configurations of cybercrime networks. Future research could cluster keywords and threats by industry to provide a deeper understanding of the potential vulnerabilities, and it could attempt to discover the network effects involved or the leaders of the cybercrime underground. For example, they should be aware that there are cybercrime underground markets where hacking tools are sold.

**Third**, this study calls for researchers, companies, antivirus vendors, and governments to collaborate in the fight against cybercrime using civil resources. Rather than acting alone, these groups should unite to maximize their efficiency and effectiveness.

**Finally**, this study also has important implications for society. Over the last few years, the world has been facing cyber terrorism and cyberwar threats from nation-sponsored attackers. Pollitt defined cyber terrorism as “the premeditated, politically motivated attack against information, computer systems, computer programs and data which results in violence against non-combatant targets by subnational groups or clandestine agents.”

Unlike most cybercrime, which is primarily motivated by monetary gain, cyber terrorists are politically motivated. As a result, governments should, for example, strengthen their ability to protect their citizens in online virtual environments by enhancing their immediate responses to threats such as cyber espionage and cyber terrorism.

## REFERENCES

### BIBLIOGRAPHY

- [1] **“A Data-Analytics Approach to Cyber Crime Underground Economy”** Jungkook An and Hee-Woong Kim Graduate School of Information IEEE Transactions on Information Forensics and Security ( Year : 2018 ).
- [2] K. Sood, S. Zeadally, and R. Bansal. **“Cybercrime at a Scale: A Practical Study of Deployments of HTTP-Based Botnet Command and Control Panels,”** IEEE Commun. Mag., vol. 55, no. 7, pp. 22– 28.2017.
- [3] Z. Shi, G. M. Lee, and A. B. Whinston, **“Toward a Better Measure of Business Proximity: Topic Modeling for Industry Intelligence,”** MIS Quart., vol. 40, no. 4, pp. 1035–1056, 2016. **“FACT SHEET: Cybersecurity National Action Plan,”** ed: The White House, 2016.
- [4] V. G. Tasiopoulos and S. K. Katsikas, **“Bypassing Antivirus Detection with Encryption,”** in Proc., 18th Panhellenic Conf. on Informatics - PCI '14, New York, New York, USA, 2014, pp. 1–2: ACM Press.
- [5] G. Giacomello, **“Close to the Edge: Cyberterrorism Today,”** in Understanding Terrorism, Emerald Group Publishing Limited, 2014, pp. 217–236.
- [6] A. K. Sood and R. J. Enbody, **“Crimeware-as-a-service—A survey of commoditized crimeware in the underground market,”** Int. J. Crit. Infr. Prot., vol. 6, no. 1, pp. 28–38, 2013.
- [7] A. Majchrzak and S. L. Jarvenpaa, **“Safe Contexts for Interorganizational Collaborations Among Homeland Security Professionals,”** J. Manag. Inf. Syst., vol. 27, no. 2, pp. 55–86, 2010.
- [8] S. W. Brenner, **“Organized Cybercrime? How Cyberspace May Affect the Structure of Criminal Relationships,”** N. C. J. Law & Technol., vol. 4, no. 1, pp. 1-50, 2002.
- [9] R. Venkateswaran, **“Virtual private networks,”** IEEE Potentials, vol. 20, no. 1, pp. 11–15, 2001.
- [10] M.M. Pollitt, **“Cyberterrorism — Fact or Fancy?,”** Comput. Fraud Security, vol. 1998, no. 2, pp. 8–10, 1998.