Chapter -1

Introduction to company

1.1 About Sabudh Foundation

Sabudh Foundation is formed by the leading data scientists in the industry in association with the Punjab government with the objective to bring together data and young data scientists to work on focused, collaborative projects for social benefit. The aim is to enable the youth to use powerful AI technologies for the greater good of society by working on real-world problems in partnership with non-profit organizations and government agencies, to tackle data-intensive high impact problems in education, healthcare, public policy, agriculture etc.

The name of the company, i.e. Sabudh, aims to use the knowledge for the collective good and to aim the potential for the continuous improvement of the society as well as to promote real world problem solving skills.

Data science can be used across a number of industries in order to be beneficial for the society.For example in agriculture, there are now Agrobots and drones being used to gauge the health of the harvest that can help farmers improve their crop yield and reduce costs. With the help of advanced technologies, we're able to save 90% of the spraying costs. These technologies can help states like Punjab which has always been the food basket of India to rehabilitate food security while improving crop health.

It provides a very healthy learning environment and the open discussions make it even more interesting to learn and explore new topics and implement them innovatively. The continuous learning is an integral part of the organisation and a lot of emphasis is given on that.

[Chapter -2]

Introduction to problem

2.1 Overview

TheThe Internet has opened up a new world of possibilities for human kind especially for the younger generation who are well equipped with tools and skills to connect with, explore, and discover the world around them.Although there are enormous benefits to be derived from the Internet, a number of concerns and Internet threats have arisen from uncensored and largely unregulated cyberspace. Parents are facing new parenting challenge involving the use of the Internet which earlier generations of parents did not have to confront. Although there are various techniques and technologies which can be adopted by parents to guide, protect, and supervise their children's use of the Internet, many parents feel ill equipped to do so. This system intends to find out how parents guide and supervise children's use of the Internet and the factors which lead to their satisfaction in a job well done. The major targets are families with children between the ages of 6 and 17, where parents are responsible for supervising both computer and Internet usage. A household survey with a representative sample of 2,579 families based on the sampling frame maintained by the Government was conducted in late 2009. Findings suggest that better-educated parents, the adoption of an authoritative parenting style, more active involvement in children's online activities, and discussion of the online experience are factors associated with positive parental influence on children's behavior.

The automatic monitoring of cyberbullying on social networking sites has potential for signalling harmful messages, preventing many threats involving physical assault or violence, the misuse of pictures or videos of a pornographic,sexual or embarrassing nature, cases demonstrating signs of suicidal ideation by victims, hate speech (e.g. racism, homophobia), commands to commit suicide and hate pages and fake profiles, sexualised cyberbullying, defamation and personal denigration and all such events are directed towards vulnerable people (e.g. young children) and providing timely responses.There are a large number of online discussion fora on the internet today which are meant for users to express, discuss and exchange their views and opinions on various topics. For example, news portals, blogs, social media channels such as Youtube typically allow users to express their views through comments. In such fora, it has been often observed that child conversations often derail and become inappropriate such as hurling abuses, passing rude and discourteous comments on individuals or certain groups/communities. In this context, we define any textual message or conversation or search of any objectionable words as inappropriate. If your child posts or view harmful or negative content online, it may not only harm other children but your child also, it can affect their online reputation, which can have negative implications for them and their family.

2.2 Existing System

If you are a parent of a teen, you know how tech-savvy they are. They know how a smartphone or computer works even if they just picked it up. As they always connected to the Internet, they know all trends of social media, the latest technologies and much more. Unfortunately, the advancement in technology has exposed children to serious risk of abuse online while leaving parents and guardians helpless to intervene.There are a lot of limitations as it becomes difficult to monitor activities, as the existing systems are not opened source and are too costly.

2.3 User Requirement Analysis

There are many threats involving physical assault or violence, the misuse of pictures or videos of a pornographic,sexual or embarrassing nature, cases demonstrating signs of suicidal ideation by victims, hate speech (e.g. racism, homophobia), commands to commit suicide and hate pages and fake profiles, sexualised cyberbullying, defamation and personal denigration and all such events are directed towards vulnerable people (e.g. young children).The parents were concerned whether it might be less important to detect certain types of cyberbullying. In addition to indicating priorities for detection, many parents stated that other cases should not be neglected and that all cases should eventually be considered and assessed. They argued that each case of cyberbullying has the potential to cause damage and that their severity and impact are highly dependent upon the vulnerability of individual victims.So, under which such systems should be implemented, including effective follow-up strategies, protecting the adolescent's privacy and safeguarding their self-reliance.

2.3.1 User Specification

The application needs any operating system and an active internet connection. All the complexities have been taken care of in the backend and the user only has to install the script with settings set to auto startup and an active internet connection with any online email account. The user need not to have a high end technical knowledge, just any operating system would be enough along with pyhton installed

2.3.2 Non - Functional Requirements

- Emails should be sent with a latency of no greater than 5 minutes(user defined) from such an activity.
- It should have a stable internet connection.
- It should have the appropriate hardware to support the smooth running.
- The system should be secure.

2.3.3 Functional Requirements

- A system must send an email when a certain condition is met.
- The system needs software to run which is open source application. Ex: Jupyter Notebook.

- It needs various libraries such as pandas, Beautiful Soup to be installed
- It needs to have python installed.

2.3.4 System Requirements

- Python installed with desired libraries.
- RAM: 4 GB
- Processor: Core i3(Min.)
- Internet Connectivity.
- Backdoor Script installed in remote computer.

2.3.5 Software Development Life Cycle

As in any other engineering discipline, software engineering also has some structured models for software development. This document will provide you with a generic overview about different software development methodologies adopted by contemporary software firms. SDLC is a process used by the software industry to design, develop and test high quality software. The SDLC aims to produce high-quality software that meets or exceeds customer expectations, reaches completion within time and cost estimates. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process. There are various software development life cycle models defined and designed which are followed during the software development process. These models are also referred as Software Development Process Models". Each process model follows a Series of steps unique to its type to ensure success in the process of software development.

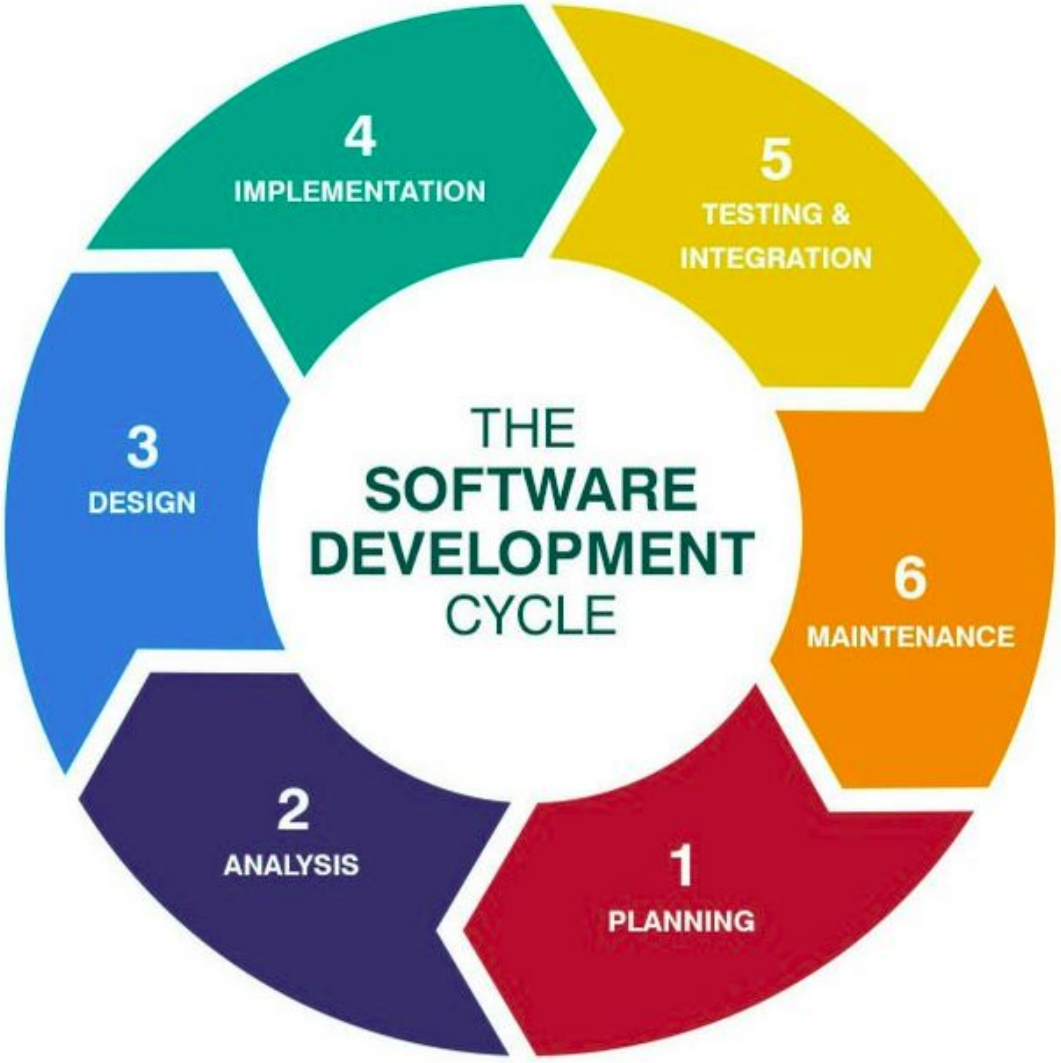

Figure 2.1: Software Development Life Cycle

We chose Iterative Waterfall model as our systems development life cycle (SDLC), also referred to as the application development life-cycle. In the Iterative model, iterative process starts with a simple implementation of a small set of the software requirements and iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed.

An iterative life cycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which is then reviewed to identify further requirements. This process is then repeated, producing a new version of the software at the end of each iteration of the model.

Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented. After each iteration, design modifications are made and new functional capabilities are added. The basic idea behind this

method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).

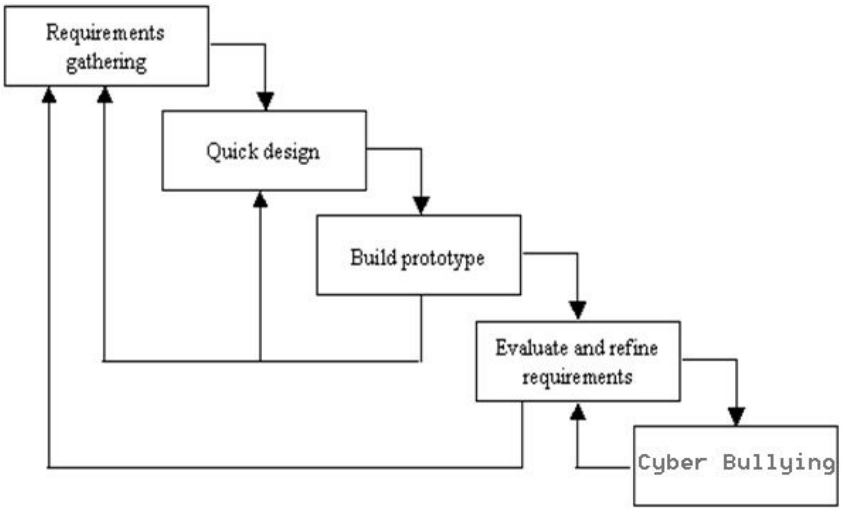

Figure 2.2: Iterative Waterfall Model

2.4 Feasibility Study

Feasibility is the determination of whether a project is capable of implementation or not. The process followed in making this determination is called feasibility study. Feasibility Study concentrates on the following areas:

- Technical Feasibility
- Operational Feasibility
- Economic Feasibility

2.4.1 Technical Feasibility

For taking screenshots pyautogui library is used, for image to text extraction Tesseract OCR library (version 4.0 neural based)library is used and all are present in Python 3.0 and are open sourced.Hence project is technical feasible.

2.4.2 Operational Feasibility

Normally parents have to monitor their child manually and child can do various things behind their back but with the implementation of this project they can monitor them everytime.

2.4.3 Economic Feasibility

Libraries used in this project all are open sourced ,so the project is economical feasible

.

2.5 Objectives of Project

- To prevent the child from bullying , profane words etc.
- Discreetly sends data on all Internet browsing and other web-related activity to a secure account, making it an efficient way for the parents
- To prevent illegal work and alerting the police,view all captured data, which highlights only items identified as Risky Behavior.
- To provide a backdoor to the police in case of illegal activity is done.

Chapter -3

Product Design

It is function-oriented design, the system is comprised of many smaller sub-systems known as functions. These functions are capable of performing significant task in the system. The system is considered as top view of all functions.

It inherits some properties of structured design where divide and conquer methodology is used.

This design mechanism divides the whole system into smaller functions, which provides means of abstraction by concealing the information and their operation.. These functional modules can share information among themselves by means of information passing and using information available globally.

Another characteristic of functions is that when a program calls a function, the function changes the state of the program, which sometimes is not acceptable by other modules. Function oriented design works well where the system state does not matter and program/functions work on input rather than on a state.

• The whole system is seen as how data flows in the system by means of data flow diagram.

• DFD depicts how functions changes data and state of entire system.

• The entire system is logically broken down into smaller units known as functions on the basis of their operation in the system.

• Each function is then described at large.

3.1 Product perspective

The current system is not so efficient as the parents have to monitor their child manually and children always find some hack to bypass the system and even police finds it harder to find the cyber criminal as they don't have efficient systems. Most of the cybercrimes are done by the teenagers and they always get past the system and laws as the criminal leaves no traces.

So we just needed an efficient system to monitor the child without hurting their privacy but only when it is needed and an efficient way to caught the cyber criminal by the police and some ways to track the culprit.

3.2 Product functions

The script is needed to be set on the startup of the pc with administrative priviliges so it can't be turned off by the user.

- A script which is installed at the target PC.
- A script which is needed to be install at CyberPolice's PC.
- Online dictionary built for the parents.
- Online dictionary for police and Police IP Address.

3.3 User characteristics

User needs software to run which is open source application. Ex: Jupyter Notebook.

- User must have Internet connection in the pc.
- User needs to defined their particular emails to get the desired screenshots.
- User must have the basic knowledge of the operating system is very user friendly so there is no need to have very vast knowledge.

3.4 Constraints

- **Time Constraint**

  The project needed to be completed in the given deadline and it got completed on time.

- **Scope Constraint**

  This project is desired by many people now a days as child's activity is needed to be monitored by parents/institutions/cyber-cafes/cyber-police and this project is performing well on it.

- **Cost Constraint**

  This project uses only open sourced libraries thus it is under the budget.

3.5 Use Case Model/Flow Chart

3.5.1 Use Case Model

A use-case model is a model of how different types of users interact with the system to solve a problem. As such, it describes the goals of the users, the interactions between the users and the system, and the required behavior of the system in satisfying these goals. A use-case model consists of a number of model elements. The most important model elements are: use cases, actors and the relationships between them. A use-case diagram is used to graphically depict a subset of the model to simplify communications. There will typically be several use-case diagrams associated with a given model, each showing a subset of the model elements relevant for a particular purpose. The same model element may be shown on several use-case diagrams, but each instance must be consistent. The use-case model may contain packages that are used to structure the model to simplify analysis, communications, navigation, development, maintenance and planning. Much of the use-case model is in fact textual, with the text captured in the Use-Case Specifications that are associated with each use-case model element. These specifications describe the flow of events of the use case. The use-case model serves as a unifying thread throughout system development.

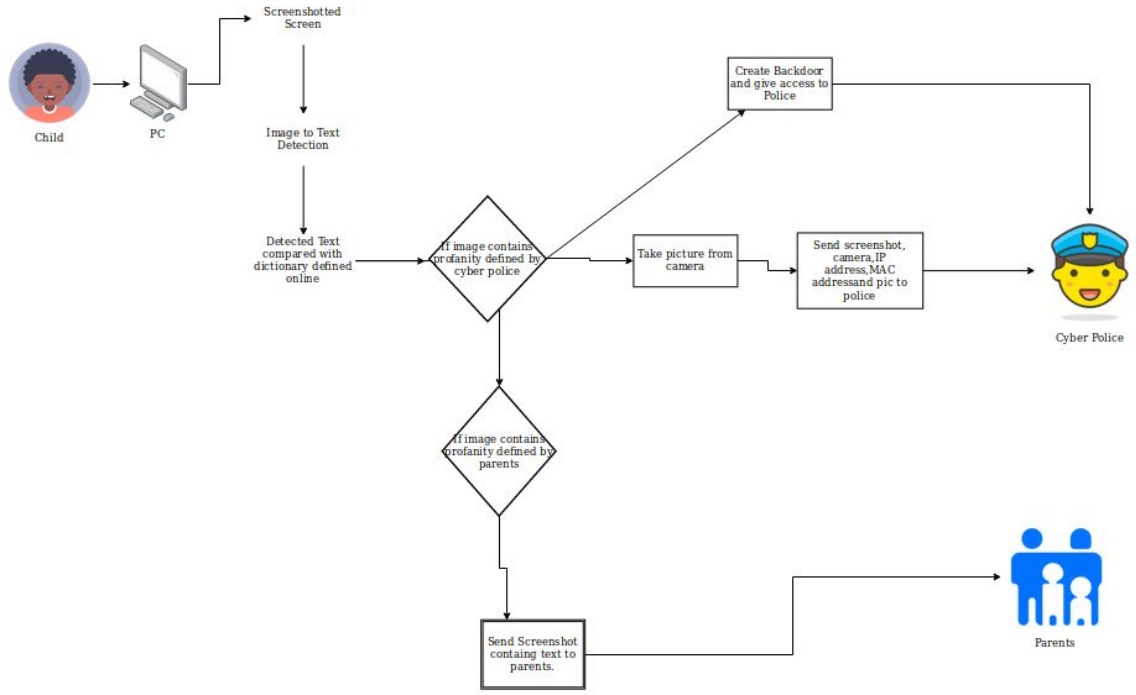

Figure 3.1: Use Case Model

3.5.2 Flowchart

An architecture diagram represents the framework of a system .It is very important to visualize the acts as a blueprint of the entire system.Architecture diagram gives the overall representation of the system. It must be self descriptive and accurate.

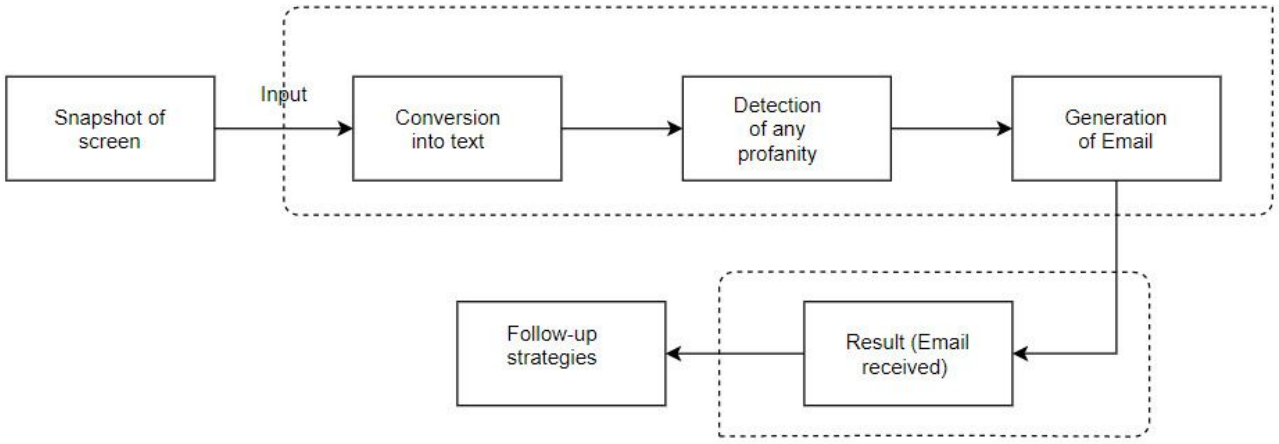

Figure 3.2: Flow Chart

3.6 Assumptions and Dependencies

It is assumed that the user has a basic knowledge of operating a pc which can either be a windows or linux.Dependencies arise when a particular instruction is linked to another instruction and cannot be implemented until the execution of the previous instruction is completed. The dependency of the whole project is on internet connectivity. Until and unless the internet connection is not provided then the mail would not be sent and would fail to alert the parents

3.7 Specific Requirements

3.7.1 Hardware Requirements

- RAM: 4 GB
- Processor: Core i3(Min.)
- Internet Connectivity.
- Backdoor Script installed in remote computer.

3.7.2 Software Requirements

- Any operating system
- Python installed with desired libraries.
- Backdoor Script installed in remote computer.

Chapter-4

Development and Implementation

4.1 Introduction to Languages

4.1.1 Python

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms. The Python interpreter is easily extended with new functions and data types implemented in C or C++ (or other languages callable from C). Python is also suitable as an extension language for customizable applications.

Python is dynamic enough to be utilized for the most simplistic programs to the most scaled, agile, and complex data operations. Python may be the language at the forefront of the tech industry's migration toward Artificial Intelligence, Machine Learning, Deep Learning, Automation, and Neural Networks. Python is also a Data Science giant that is becoming increasingly popularized in FinTech alongside languages like R. Python developers' commitment to optimizing libraries, frameworks, & packages speaks for the power and potentiality of Python's future!

4.1.2 Beautiful Soup

Beautiful Soup is a Python library for getting data out of HTML, XML, and other markup languages.It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers hours or days of work.Say you've found some webpages that display data relevant to your research, such as date or address information, but that do not provide any way of downloading the data directly. Beautiful Soup helps you pull particular content from a webpage, remove the HTML markup, and save the information.

4.1.3 Tesser OCR

Tesseract is an **OCR engine** with support for unicode and the ability to recognize more than 100 languages out of the box. It can be trained to recognize other languages.

A simple, Pillow-friendly, wrapper around the tesseract-ocr API for Optical Character Recognition (OCR).Tesserocr integrates directly with Tesseract's C++ API using Cython which allows for a simple Pythonic and easy-to-read source code. It enables real concurrent execution when used with Python's threading module by releasing the GIL while processing an image in tesseract.Tesserocr is designed to be

Pillow-friendly but can also be used with image files instead.You may need to manually compile tesseract for a more recent version. Note that you may need to update your LD_LIBRARY_PATH environment variable to point to the right library versions in case you have multiple tesseract/leptonica installations.Cython (>=0.23) is required for building and optionally Pillow to support PIL.Image objects.Tesseract is used for text detection on mobile devices, in video, and in Gmail image spam detection.

4.1.4 PIL(Python Imaging Library)

Python Imaging Library (abbreviated as PIL) (in newer versions known as Pillow) is a free library for the python programming language that adds support for opening, manipulating, and saving many different image file formats. It is available for Windows, Mac OS X and Linux. The latest version of PIL is 1.1.7, was released in September 2009 and supports Python 1.5.2–2.7, with python 3 support to be released later.

Development appears to be discontinued with the last commit to the PIL repository coming in 201. Consequently, a successor project called Pillow has forked the PIL repository and added Python 3 support. This fork has been adopted as a replacement for the original PIL in Linux distributions.

The Python Imaging Library adds image processing capabilities to your Python interpreter.

This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities.

The core image library is designed for fast access to data stored in a few basic pixel formats. It should provide a solid foundation for a general image processing tool.

Let’s look at a few possible uses of this library.

4.2 Any other Supporting Languages or tools

4.2.1 Jupyter notebook platform

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.It supports

- **Python 3:** from the version 3.5 up to the version 3.8

In addition, in the Professional edition, one can develop Django, Flask and Pyramid applications. Also, it fully supports HTML (including HTML5), CSS, JavaScript, and XML: these languages are bundled in the

IDE via plugins and are switched on for you by default. Support for other languages and frameworks can also be added via plugins.

4.2.2 Backdoor

A backdoor is a means to access a computer system or encrypted data that bypasses the system's customary security mechanisms.A developer may create a backdoor so that an application or operating system can be accessed for troubleshooting or other purposes. However, attackers often use backdoors that they detect or install themselves as part of an exploit. In some cases, a worm or virus is designed to take advantage of a backdoor created by an earlier attack.Whether installed as an administrative tool, a means of attack or as a mechanism allowing the government to access encrypted data, a backdoor is a security risk because there are always threat actors looking for any vulnerability to exploit.In her 2000 article, "Who gets your trust?" security consultant Carole Fennelly used an analogy to illustrate the situation: "Think of approaching a building with an elaborate security system that does bio scans, background checks, the works. Someone who doesn't have time to go through all that might just rig up a back exit so they can step out for a smoke -- and then hope no one finds out about it.

4.3 Implementation of problem

Input image is the screenshot of the screen that is being used by the child, the text extracted by the tesserocr are stored into an array, array gets assigned to dataframe.In each row of the dataframe df, every image extracted text are stored. This dataframe is compared with dataframes name bad and police_bad.The screenshots of the screen matching text with both dataframes are sent to the respective accounts,The extracted text from the snapshot is stored in the dataframe.

The dataframe name being df is shown below:

```
In [23]: df
```

Out[23]:

	comment	bad	police_bad
0	[eoe, cma, file, edit, view, insert, cell, kernel, «, widgets, help, trusted, python, ek, br, run, mc, dp, coe, in, def, main, global, bl, lengcdf, location, screenshats, clean, dict, mailit, time, steep, ope, ee, any, a, firefox, in, door, agnetooggnaitcom, jagnetoocegmail, con, home, jagnetodesktopmeta, id, childpasswordgetpassgetpassenter, password, for, i, in, range, words, ce, cr, ld, tlthreading, thread, targetmain, tlstart, try, if, doors, threading, thread, targetbackdoor, tstart, except, print, enter, password, eere]	0	0
1	[file, edit, view, insert, cell—, kernel, «, widgets, help, trusted, python, ek, br, run, mc, dp, coe, in, in, “def, main, global, bl, lengcdf, location, screenshats, clean, dict, mailit, time, steep, shutilrmtreelocation, doorg, agnetooggnaitcom, jagnetoocegmail, con, home, jagnetodesktopmeta, id, childpasswordgetpassgetpassenter, password, for, i, in, range, words, tlthreading, thread, targetmain, tlstart, try, if, doors, threading, thread, targetbackdoor, tstart, except, print, enter, password, bre, rees, ce, cr, ore, er, ea, saat, qal, images, ©, shopping, news, g, maps, settings, tools, w, enwikipediaorg, wiki, », weed, weed, wikipedia, a, weed, is, a, plant, considered, undesirable, in, a, particular, situation, “a, plant, in, the, ...]	2	10
2	[file, edit, view, insert, cell—, kernel, «, widgets, help, trusted, python, ek, br, run, mc, dp, coe, in, in, “def, main, global, bl, lengcdf, location, screenshats, clean, dict, mailit, time, steep, shutilrmtreelocation, doorg, agnetooggnaitcom, jagnetoocegmail, con, home, jagnetodesktopmeta, id, childpasswordgetpassgetpassenter, password, for, i, in, range, words, tlthreading, thread, targetmain, tlstart, try, if, doors, threading, thread, targetbackdoor, tstart, except, print, enter, password, mozilla, firefox, private, browsing, et, a, ©, mujaheedin, —, s, s, firefox, search, the, web, ce, cr, ld, eere]	0	1
3	[file, edit, view, insert, cell, kernel, «, widgets, help, trusted, python, oe, ©, bt, ¥, wru, mc, op, core, a, in, def, main, global, bl, lengcdf, location, screenshots, clean, dict, maitit, time, steep, shutilrmtree, location, in, door, emailidchildjagnetoognailcon”, enailidpolicejagnetoocegmailcom, honejagnetodesktopmeta, idchildpasswordgetpassgetpassenter, password, for, iin, range, words, tthreading, thread, targetmain, tlstart, try, if, door, tthreading, thread, targetbackdoor, tstart, except, print, enter, password, in, de, le, ee, ec, cr, eg, er, mujaheedin, qal, gimages, news, videos, f, books, i, more, settings, tools, about, results, seconds, showing, results, for, mujahideen, search, instead, for, mujaheedin, mujahideen, arabic, la, mujahidin, is, the, plural, form, of, ...]	2	2
4	[ec, ee, gt, cr, tare, pree, eer, ee, es]	0	0
5	[er, and, conn, pree, google, search, mozilla, firefox, private, browsing, ”, wwwmarketwatchcom, investing, stock, fb, fb, stock, price, facebook, inc, cl, a, stock, quote, us, nasdaq, fb, complete, facebook, inc, cl, a, stock, news, by, marketwatch, view, realtime, stock, prices, and, stock, quotes, for, a, full, financial, overview, wwwcnbccom, quotes, », symbolfb, ¥, fb, facebook, stock, price, quote, and, news, cnbc, get, facebook, fb, nasdaq, realtime, stock, quotes, news, and, financial, information, from, cnbc, financeyahoocom, quote, fb, ¥, facebook, inc, fb, stock, price, quote, history, news, find, the, latest, facebook, inc, fb, stock, quote, history, news, and, other, vital, information, ...]	0	0

Figure 4.1: Dataframe

4.4 Test cases

4.4.1 Testing

This explains Testing for EduCollab application. Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Testing presents an interesting of a system using various test data. Preparation of the test data plays a vital role in the system testing. After preparation the test data, the system under study is tested those test data. Errors were found and corrected by using the following testing steps and corrections are recorded for future references. Thus, a series of testing is performed on the system before it is ready for implementation**.**

4.4.1.1 Unit Testing

The primary goal of unit testing is to take the smallest piece of testable software in the application, isolate it from the remainder of the code, and determine whether it behaves exactly as you expect. Each unit is tested separately before integrating them into modules to test the interfaces between modules. Unit testing has proven its value in that a large percentage of defects are identified during its use. Unit testing is a software verification and validation method where the programmer gains confidence that individual units of source code are fit for use. A unit is the smallest testable part of an application. In procedural

programming a unit may be an individual program, function, procedure, etc., while in object-oriented programming the smallest unit is a class, which may belong to a base/super class, abstract class or derived/child class. . Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended. Its implementation can vary from being very manual (pencil and paper) to being formalized as part of build automation.

4.4.1.2 Integration Testing

Integration testing, also known as integration and testing(I&T), is a software development process which program units are combined and tested as groups in multiple ways. In this context, a unit is defined as the smallest testable part of an application. Integration testing can expose problems with the interfaces among program components before trouble occurs in real world program execution Integration testing is a component of Extreme Programming (XP), a pragmatic method of software development that takes a meticulous pragmatic method of software development that takes a meticulous. There are two major ways of carrying out an integration test, called the bottom-up method and the top-down method. Bottom-up integration testing begins with unit testing, followed by tests of progressively higher-level combinations of units called modules or builds. In top-down integration testing, the highest-level modules are tested first and progressively lower-level modules are tested after that. In a comprehensive software development environment, bottom-up testing is usually done first, followed by top-down testing.

4.4.2 Test Plan

A test plan can be defined as a document describing the scope, approach, resources, and schedule of intended testing activities. It identifies test items, the features to be tested, the testing tasks, who will do each task, and any risks requiring contingency planning. In software testing, a test plan gives detailed testing information regarding an upcoming testing effort, including- Scope of testing Schedule Test Deliverables, Release Criteria, Risks and Contingencies. It is also described as a detail of how the testing will proceed, who will do the testing, what will be tested, in how much time the test will take place, and to what quality level the test will be performed. Test plan is a document which includes, introduction, assumptions, list of test cases, list of features to be tested, approach, deliverables, resources, risks and scheduling. A test plan is a systematic approach to testing a system such as a software. The plan typically contains a detailed understanding of what the eventual workflow will be. A record of the test planning process detailing the degree of tester independence, the test environment, the test case design techniques and test measurement techniques to be used, and the rationale for their choice. One term associated with testing is Failure which is a manifestation of an error (or defect or bug). But, the mere presence of an error may not necessarily lead to a failure.

Chapter-5

Conclusion and Future Scope

5.1 Conclusion

The proposed system that is an application called CyberBullying, which has been developed using Python scripts for the backend, successfully fulfills the purpose of monitoring child activities by their parents

This script helps you become a responsible digital parent by protecting your kids against all online and offline dangers:

- online predators
- cyber bullying
- suicide prevention
- sexting
- inappropriate behavior
- drug & alcohol use

Apart from being on the lookout for people predating on your child, being a cyber parent also involves controlling how long your child spends on the screen.It discreetly sends data on all Internet browsing and other web-related activity to a secure account. Parents or Police can see exactly what children or criminals are doing and who they're talking to on the Internet

5.2 Future Scope

The proposed project is continuously being developed and there are changes being made with the addition of new modules almost everyday. The profanity detection is working in the backend and it would also be trained for recognization of the punjabi text too. Compression algorithm would also be implemented to compress the capture screenshots resulting in more efficiency of the system.

References

[1] Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. Every picture tells a story: Generating sentences from images. In Proceedings of the 11th European Conference on Computer Vision: Part IV, ECCV'10, pages 15–29, Berlin, Heidelberg, 2010. Springer-Verlag.

[2] Polina Kuznetsova, Vicente Ordonez, Alexander C. Berg, Tamara L. Berg, and Yejin Choi. Collective generation of natural image descriptions. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1, ACL '12, pages 359–368, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[3] Siming Li, Girish Kulkarni, Tamara L. Berg, Alexander C. Berg, and Yejin Choi. Composing simple image descriptions using TesserOcr. In Proceedings of the Fifteenth

Conference on Computational Image to Text, CoNLL '11, pages 220–228, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[4] Xinlei Chen and C. Lawrence Zitnick. Learning a recurrent visual representation for Snapshot generation. CoRR, abs/1411.5654, 2014.

[5] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, and Alan L. Yuille. Cyber Parenting with automatic monitoring of cyber bullying.

Appendix

Screenshots of the developed project :

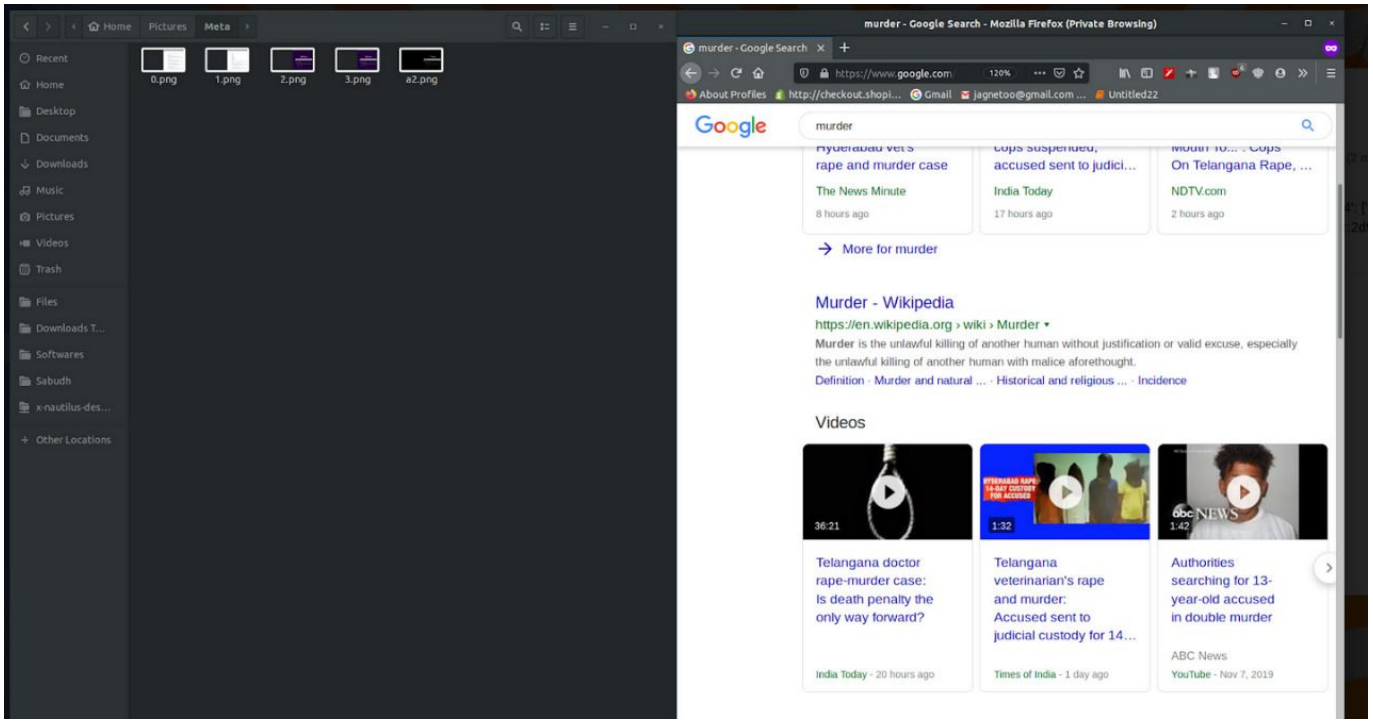

Figure 5.1: Child Search

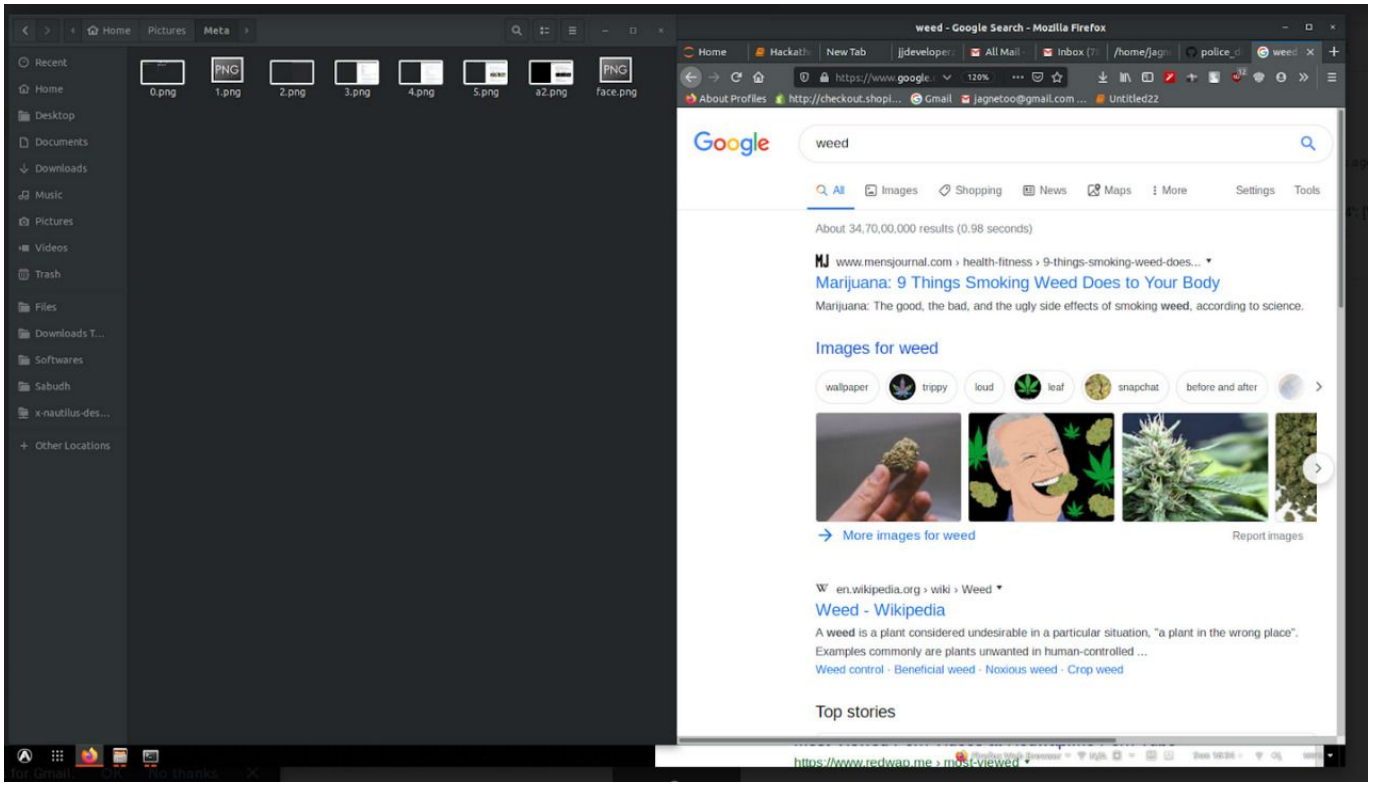

Figure 5.2: Criminal Search

```
In [23]: df
```

Out[23]:

	comment	bad	police_bad
0	[eoe, cma, file, edit, view, insert, cell, kernel, «, widgets, help, trusted, python, ek, br, run, mc, dp, coe, in, def, main, global, bl, lengcdf, location, screenshats, clean, dict, maillt, time, steep, ope, ee, any, a, firefox, in, door, agnetooggnaitcom, jagnetoocegmail, con, home, jagnetodesktopmeta, id, childpasswordgetpassgetpassenter, password, for, i, in, range, words, ce, cr, ld, tlthreading, thread, targetmain, tlstart, try, if, doors, threading, thread, targetbackdoor, tstart, except, print, enter, password, eere]	0	0
1	[file, edit, view, insert, cell—, kernel, «, widgets, help, trusted, python, ek, br, run, mc, dp, coe, in, in, “def, main, global, bl, lengcdf, location, screenshats, clean, dict, maillt, time, steep, shutilrmtreelocation, doorg, agnetooggnaitcom, jagnetoocegmail, con, home, jagnetodesktopmeta, id, childpasswordgetpassgetpassenter, password, for, i, in, range, words, tlthreading, thread, targetmain, tlstart, try, if, doors, threading, thread, targetbackdoor, tstart, except, print, enter, password, bre, rees, ce, cr, ore, er, ea, saat, qal, images, ©, shopping, news, g, maps, settings, tools, w, enwikipediaorg, wiki, », weed, weed, wikipedia, a, weed, is, a, plant, considered, undesirable, in, a, particular, situation, “a, plant, in, the, ...]	2	10
2	[file, edit, view, insert, cell—, kernel, «, widgets, help, trusted, python, ek, br, run, mc, dp, coe, in, in, “def, main, global, bl, lengcdf, location, screenshats, clean, dict, maillt, time, steep, shutilrmtreelocation, doorg, agnetooggnaitcom, jagnetoocegmail, con, home, jagnetodesktopmeta, id, childpasswordgetpassgetpassenter, password, for, i, in, range, words, tlthreading, thread, targetmain, tlstart, try, if, doors, threading, thread, targetbackdoor, tstart, except, print, enter, password, mozilla, firefox, private, browsing, et, a, ©, mujaheedin, —, s, s, firefox, search, the, web, ce, cr, ld, eere]	0	1
3	[file, edit, view, insert, cell, kernel, «, widgets, help, trusted, python, oe, ©, bt, ¥, wru, mc, op, core, a, in, def, main, global, bl, lengcdf, location, screenshots, clean, dict, maitit, time, steep, shutilrmtree, location, in, door, emailidchildjagnetoognailcon”, enailidpolicejagnetoocegmailcom, honejagnetodesktopmeta, idchildpasswordgetpassgetpassenter, password, for, iin, range, words, tthreading, thread, targetmain, tlstart, try, if, door, tthreading, thread, targetbackdoor, tstart, except, print, enter, password, in, de, le, ee, ec, cr, eg, er, mujaheedin, qal, gimages, news, videos, f, books, i, more, settings, tools, about, results, seconds, showing, results, for, mujahideen, search, instead, for, mujaheedin, mujahideen, arabic, la, mujahidin, is, the, plural, form, of, ...]	2	2
4	[ec, ee, gt, cr, tare, pree, eer, ee, es]	0	0
5	[er, and, conn, pree, google, search, mozilla, firefox, private, browsing, ”, wwwmarketwatchcom, investing, stock, fb, fb, stock, price, facebook, inc, cl, a, stock, quote, us, nasdaq, fb, complete, facebook, inc, cl, a, stock, news, by, marketwatch, view, realtime, stock, prices, and, stock, quotes, for, a, full, financial, overview, wwwcnbccom, quotes, », symbolfb, ¥, fb, facebook, stock, price, quote, and, news, cnbc, get, facebook, fb, nasdaq, realtime, stock, quotes, news, and, financial, information, from, cnbc, financeyahoocom, quote, fb, ¥, facebook, inc, fb, stock, price, quote, history, news, find, the, latest, facebook, inc, fb, stock, quote, history, news, and, other, vital, information, ...]	0	0

Figure 5.3: Dataframe

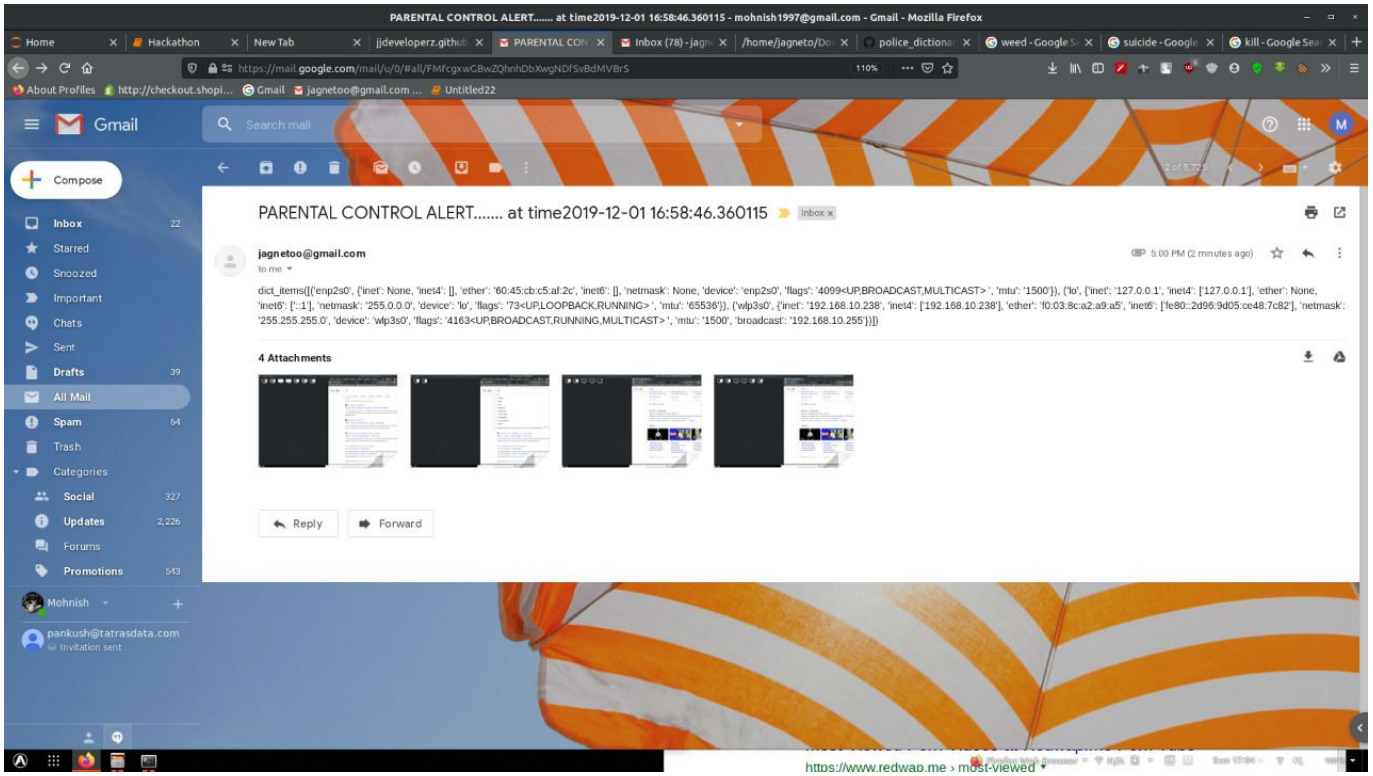

Figure 5.4: Parent’s Mail

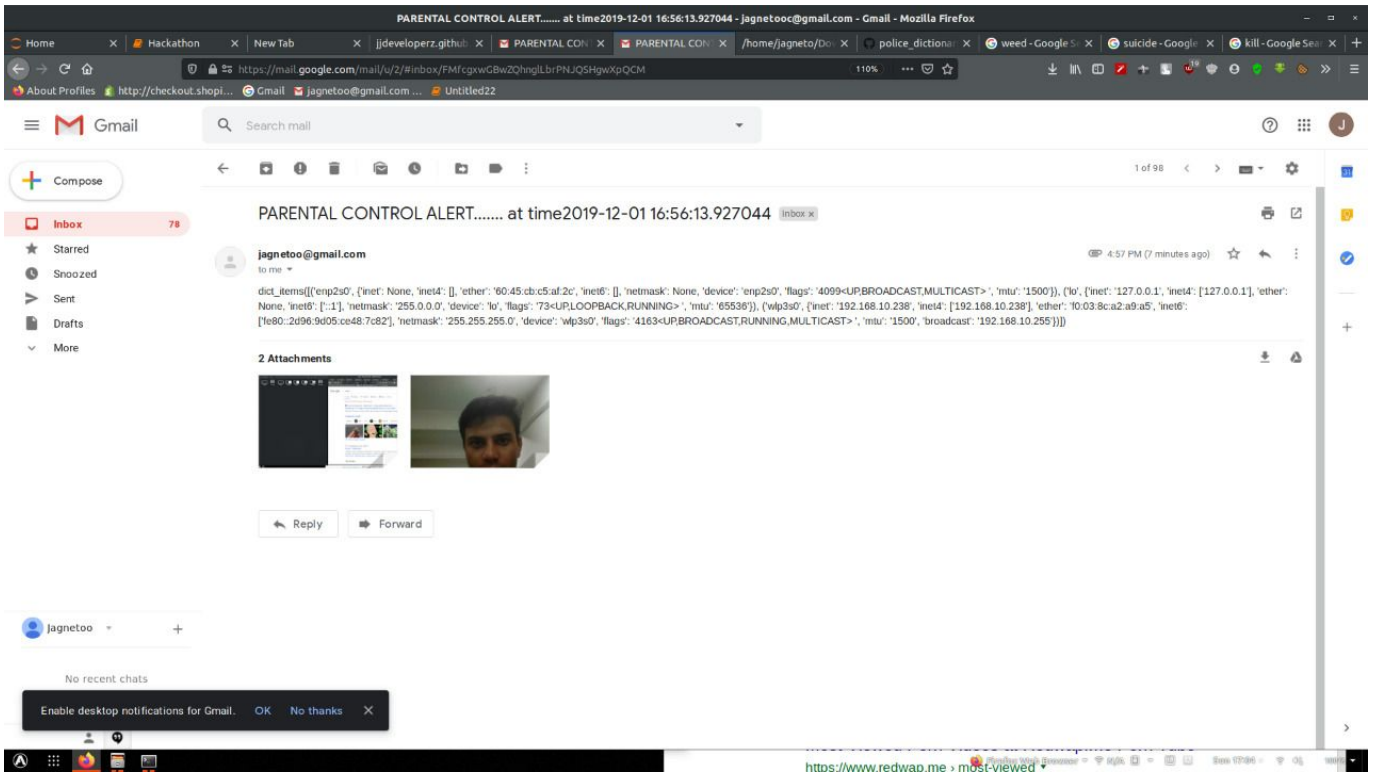

Figure 5.5: Police Mail

```
        data = client.recv(1024).decode("utf-8")
        if data == "exit":
            print("Terminating connection", addr[0])
            break
        print(data)
client.close()
sock.close()
```

```
Listening on port 2222
jagneto@jagnetox:/home/jagneto/Desktop$ ifconfig
enp2s0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 60:45:cb:c5:af:2c  txqueuelen 1000  (Ethernet)
        RX packets 90642  bytes 90183744 (90.1 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 45937  bytes 4968173 (4.9 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 816572  bytes 268738807 (268.7 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 816572  bytes 268738807 (268.7 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlp3s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.10.238  netmask 255.255.255.0  broadcast 192.168.10.255
        inet6 fe80::2d96:9d05:ce48:7c82  prefixlen 64  scopeid 0x20<link>
        ether f0:03:8c:a2:a9:a5  txqueuelen 1000  (Ethernet)


       RX packets 3108931  bytes 2779050012 (2.7 GB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1460237  bytes 513484263 (513.4 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Figure 5.6: Backdoor

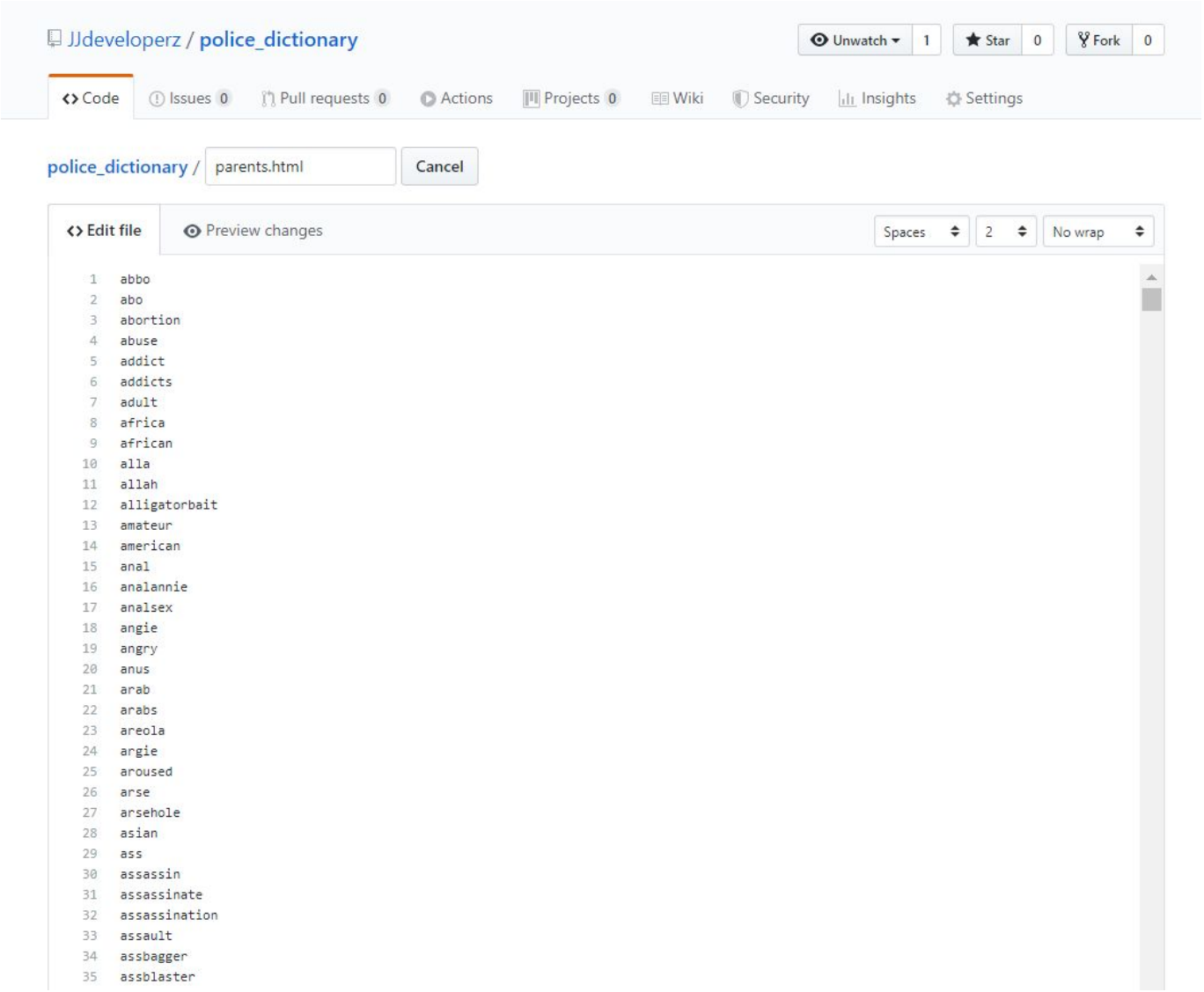

Figure 5.7: Parent's Bad-words Dictionary

18 lines (18 sloc) | 121 Bytes

```
1  Chitta
2  weed
3  bhang
4  nasha
5  weapons
6  asla
7  ak-47
8  abuse
9  terror
10 terrorist
11 terrorism
12 isis
13 isi
14 pakistan
15 drugs
16 hashish
17 hafeem
18 riots
```

Figure 5.8: Police's Bad-words dictionary

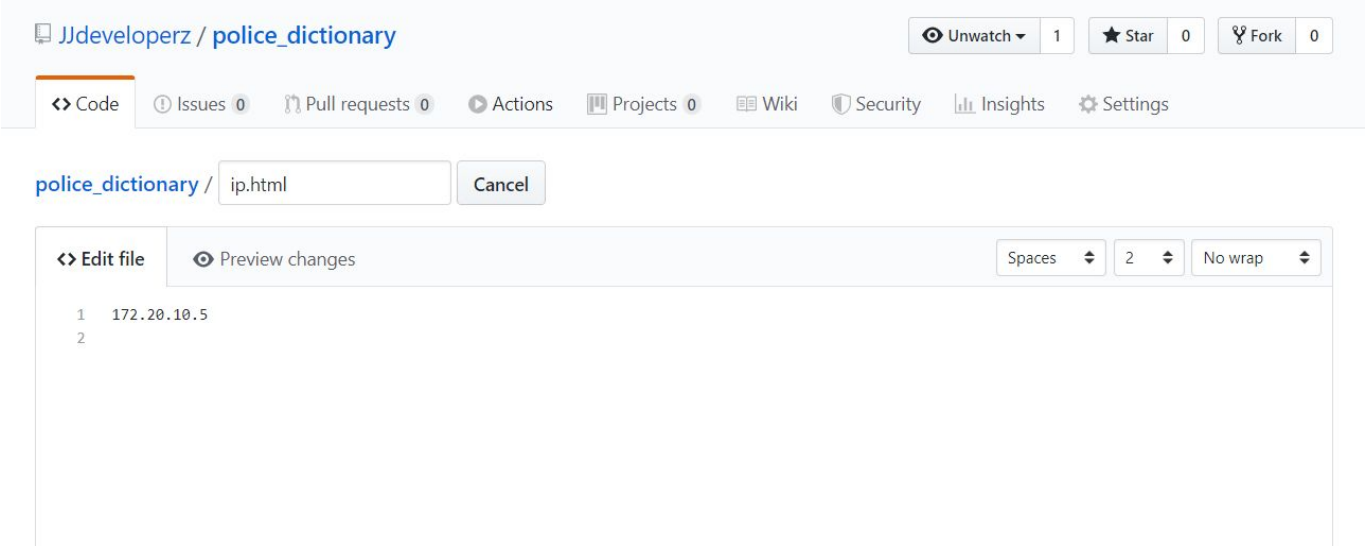

Figure 5.9: Police's IP address