Digital Locker

Submitted By

**ADARSH KUMAR SINGH (1603331009)**

**AMRISH YADAV (1603331022)**

**DHEERAJ SHARMA (1603331053)**

Under the guidance of

**Prof. N.K. YADAV**

**Prof. SANDEEP BHATIA**

**(**Project Coordinator**)**

**Department of Electronics & Communication Engineering**

**RAJ KUMAR GOEL INSTITUTE OF TECHNOLOGY**

**5th, K.M. Stone, Delhi-Meerut Road, Ghaziabad Uttar Pradesh - 201003**

**April, 2020**

**VISION**

To continually develop excellent professional capable of providing sustainable solutions to challenging problems in their fields and prove responsible global citizens.

**MISSION**

We wish to serve the nation by becoming a reputed deemed university for providing value-based professional education.

**VISION OF THE DEPARTMENT**

To develop the Department into a full-fledged Center of learning in various fields of Electronics & Communication Engineering keeping in view the latest developments in the world.

**MISSION OF THE DEPARTMENT**

To educate the students in Contemporary Technologies in Electronics & Communication Engineering to meet the Industrial and Societal needs.

**Program Educational Objective (PEOs)**

|  |  |  |
| --- | --- | --- |
| **PEO 1** | **Knowledge Acquisition** | Graduates having strong background in Mathematics, Science and Engineering & Technology and able to employ necessary techniques, hardware and software tools for modern engineering application. |
| **PEO 2** | **Competency for Employment** | Graduates will be involved in professional societies, formal educational opportunities; Industry oriented training programs that enhanced their ability to be productive in their place of employment. |
| **PEO 3** | **Innovative Skills** | Graduates will demonstrate high level of creativity, critical thinking, responsibility, team work and leadership in their careers. |
| **PEO 4** | **Social Contribution** | Graduates understand professionals, ethical and social responsibilities. |
| **PEO 5** | **Entrepreneurship** | Graduates will use their technical and presentation skills to be successful entrepreneur to enhance the self-employability. |

**POs Electronics and Communication Engineering**

**1. Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

**2. Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

**3.** **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

**4. Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions. 5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations**.**

**6. The engineer and society:** Apply to reason informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice**.**

**7. Environment and Sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

**8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

**9. Individual and teamwork:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings**.**

**10. Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions**.**

**11. Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

**12. Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change**.**

**PSOs Electronics and Communication Engineering**

**1.** The ability to understand and apply fundamental knowledge of various fields of Electronics and Communication Engineering in the analysis and design of different types of electronics and communication systems.

**2.** Ability in using electronic modern tools (both software and hardware) for the design and analysis to solve complex electronic problems.

**3**. Ability to adapt to the changing work environment, to develop good interpersonal skills as an individual and leader in a team in appreciation of professional ethics and societal responsibilities.

**MAPPING OF POs & PSOs WITH PROJECT**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** | **PSO3** |
| **Digital Locker** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**JUSTIFICATION OF MAPPING PROJECT WITH POs & PSOs**

|  |  |
| --- | --- |
| **PO** | **JUSTIFICATION** |
| **1** | The project requires the overall knowledge of all disciplines including mathematics, electronics, analytical approach in conjunction with computer programming knowledge. |
| **2** | Various research papers were referred to and their end results were exploited to accomplish this project. |
| **3** | The project is aimed at improving security and privacy of data. Thereby we have tried to accomplish the aim we started working for. |
| **4** | We have made use of OpenCV library which is known to yield 98% accurate results. The algorithm used is highly accurate and solves the requirement of our project. |
| **5** | We used python programming language, imported various libraries and modules. We set with an aim to tackle complex problem and successfully did so using programming skills. |
| **6** | Our primary focus is on user’s data security and privacy. We have accomplished and demonstrated the same. |
| **7** | The project focuses and safety of user’s data as well as privacy. Is totally sustainable and in line with environmental norms. 100% nature friendly. |
| **8** | The project focuses on public’s data protection. All the sources used are open sources and no violation is done to accomplish the project. |
| **9** | We were two members working as a team. Both of us worked efficiently as a team as well as individually. |
| **10** | We could communicate effectively with our users, could understand their concern and revert back with solution. Project was done under seamless sharing of information from peer group. |
| **11** | The project was managed very well with both of the team members being on schedule, working and understanding problems and coming up with solutions on academia, economic as well as time factors. |
| **12** | This project aims at making data security and privacy of its user base invincible. Even though we have used open source code, we are striving to train our own face recognition model to enhance living standard of the society. |

|  |  |
| --- | --- |
| **PSO** | **JUSTIFICATION** |
| **1** | The project uses end result of electronics engineering, that is, computer hardware to program and execute python code. It endeavors to enhance cyber security and uses engineering knowledge to design a secure data storage solution. |
| **2** | The project uses modern software for programming in conjunction with latest computing devices to accomplish privacy and security of user’s data. |
| **3** | The project is based on biometric authentication using facial features, with encryption standard of highest order. This technology is going to be the next big step for humanity. It also gave us an opportunity to perform among peer groups with display of interpersonal skills. |

**TABLE OF CONTENTS**

Page

[CERTIFICATE i](#_Toc40564141)

[DECLARATION ii](#_Toc40564142)

[ACKNOWLEDGEMENT iii](#_Toc40564143)

[ABSTRACT iv](#_Toc40564144)

[LIST OF FIGURES v](#_Toc40564145)

[LIST OF ABBREVIATIONS vi](#_Toc40564146)

[Chapter 1: INTRODUCTION 1](#_Toc40564147)

[1.1 Introduction to the problem 1](#_Toc40564148)

[1.2 What is biometric authentication? 4](#_Toc40564149)

[1.3 Choosing Facial Authentication Over Facial Recognition 7](#_Toc40564150)

[1.4 Putting Encryption to work 9](#_Toc40564151)

[Chapter 2: FACING THE CHALLENGE 11](#_Toc40564152)

[2.1 Web Development 11](#_Toc40564153)

[2.1.1 HTML 11](#_Toc40564154)

[2.1.2 CSS 15](#_Toc40564155)

[2.1.3 JavaScript 17](#_Toc40564156)

[Chapter 3: THE OUTCOME 23](#_Toc40564157)

[APPENDIX 24](#_Toc40564158)

[REFERENCES 25](#_Toc40564159)

# CERTIFICATE

This is to certify that Project Report titled “Digital Locker” which is submitted by Adarsh Kumar Singh, Amrish Yadav & Dheeraj Sharma in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Electronics and Communication Engineering of Dr. APJ Abdul Kalam Technical University, is a record of the candidate own work carried out by him under my/our supervision. The matter embodied in this thesis is original and has not been submitted for the award of any other degree.

Date: Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Guide: Prof N.K. YADAV

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Coordinator: ProfSANDEEP BHATIA

# DECLARATION

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Date:

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: ADARSH KUMAR SINGH

Roll No: 1603331009

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: AMRISH YADAV

Roll No: 1603331022

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: DHEERAJ SHARMA

Roll No: 1603331053

# ACKNOWLEDGEMENT

We would like to thank our friend Kushagra Khare for his valuable insights and direction during the tenure of the project. We are indebted to our parents and teachers for always being there. We also are thankful to Professor Sandeep Bhatia - Department of Electronics and Communication & Professor N.K. Yadav - Department of Computer Science Engineering, for believing in us and empowering us with their constant support.

Date:

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: ADARSH KUMAR SINGH

Roll No: 1603331009

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: AMRISH YADAV

Roll No: 1603331022

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: DHEERAJ SHARMA

Roll No: 1603331053

# ABSTRACT

In this project we are bringing together authenticating users to access a repository of their data stored on a server using face of the user. We often struggle at recalling passwords, or keying the password in different circumstances when we are trying to log into various online platforms.

To overcome this, we have made an effort to build a system that allows authentication of its users by using face data. However, there is a challenge with this method regarding privacy. We are putting together measures such as encrypting the face data and strict policy of not disclosing, re-using or deploying this data in any way.

Face recognition has been an interesting and important research fields in the recent years. The reasons come from the need of automatic recognitions and surveillance systems, the interest in human visual system on face recognition, and the design of human-computer interface, etc. These researches involve knowledge and researchers from disciplines such as neuroscience, psychology, computer vision, pattern recognition, image processing, machine learning, etcetera.

Facial authentication also is helpful in process of determining whether someone or something is, in fact, who or what it declares itself to be. Possibility of this technology going mainstream is very likely. As a matter of fact, it is already getting adopted by technology giants like Apple, Google, Amazon.

We were able to accomplish the goal we started this journey for. In this report we discuss the approach we have exploited for making authentication work using biometric, that is, face. This approach is relatively new and we strive to reinforce privacy of users. We are using programing languages such as Python, Web Development languages and so on.

# LIST OF FIGURES

[Figure 1. Facial pattern/ feature extraction. 7](#_Toc40564100)

[Figure 2. AES – A symmetric encoding algorithm. 9](#_Toc40564101)

[Figure 3. RSA – An asymmetric encoding algorithm. 10](#_Toc40564102)

[Figure 4. Basic structure of HTML code 12](#_Toc40564103)

[Figure 5. Linking CSS to a HTML page 16](#_Toc40564104)

[Figure 6. CSS code block 16](#_Toc40564105)

[Figure 7. Behind the webpage 17](#_Toc40564106)

# LIST OF ABBREVIATIONS

# Chapter 1: INTRODUCTION

## 1.1 Introduction to the problem

In this project we are bringing together authenticating users to access a repository of their data stored on a server using face of the user. We often struggle at recalling passwords, or keying the password in different circumstances when we are trying to log into various online platforms.

**A tale of passwords**

A method of authentication that requires username and password combination is vulnerable to phishing or spoofing, interception and other cyber threats. Passwords are a weak form of protection for many reasons. One major reason is that passwords depend on the weakest link in the computer and network security chain; namely, the human user. Most users think that security procedures are either a joke, the cloak-and-dagger stuff of system and network administrators, and/or due to paranoia. As a result, they do not pay sufficient attention to wisely choosing passwords nor protecting them.

There are several ways in which an intruder can attack password-protected systems. The most common form of attack is password guessing. People often choose their own name, username, telephone number, or some variant as their password; next, they choose the name of family members or friends, pets, special interests, or some variant. And how does an attacker find this information? In many cases, it's easy. The Finger utility, a known security weakness waiting to be exploited, displays the status of all currently active users complete with username, one item of information that an attacker cannot do without. Finger listings1 also display the users' real name; the PLAN.TXT and PROJECT.TXT files often supply additional personal information with which an intruder can launch a password guessing attack, as well as information about the last login. Many individual's WWW pages supply even more personal information.

The normal approach against password-guessing attacks is a feature called blacklisting, which limits the number of consecutive unsuccessful login attempts. In a typical implementation, a login attempt counter is set to zero after a successful login and incremented after any unsuccessful login attempt. If the counter ever reaches the blacklist threshold (usually between 3 and 7), account login is disabled even if the correct password is supplied.

Intruders can use blacklisting as the basis for another form of attack. Even if they cannot break into a system, attackers can effectively deny service to users with a blacklist attack, where the attacker can effectively disable many (or all) users by purposely blacklisting them. To prevent system-level accounts from being blacklisted by an attacker, most operating systems allow logins to system accounts from the operator's console regardless of the account's blacklist status.

A second possible attack is to steal a system's password file, an amazingly simple thing to do if the file is not assigned the correct access protection. While passwords are almost always stored in some encrypted or hash form in the file, they are still susceptible to attack via a dictionary attack, where a large number of words are encrypted using the operating systems' encryption scheme in an effort to find a match in the password file. Some studies suggest that there is a 99 percent chance of successfully cracking at least one password in a file containing as few as 16 passwords. With today's high-speed processors available on the desktop at modest cost, nearly anyone with a spell checker can launch a dictionary attack.

Along these lines, it is important to note that the length of a password is not the major factor in determining how good it is. Most users today still choose passwords containing only lowercase letters, most often forming a word or string of words. These types of passwords are the most susceptible to a dictionary attack.

Another form of attack is called login spoofing, and can be particularly successful in public terminal rooms at educational institutions. In this scenario, the attacker runs a program that displays what appears to be a legitimate login message. When another user attempts to login, the program makes the usual queries for the username and password, writes the information to a file, displays an "Invalid login" message, and then logs the attacker out. The legitimate user, thinking that they must have made a typographical error, tries again to login and succeeds. This attack works often and, if lucky, the attacker finds a user who has a high level of system privilege.

A fourth attack is to actually monitor the traffic between the user and computer. If this attack is used, the attacker may be able to find usernames and passwords in plain text. In a local network, this form of attack requires that the intruder gain physical access to the communications lines or wiring closet; on the Internet, an intruder may just need to monitor the packets used for Telnet, the WWW, or other passworded accounts.

After obtaining legitimate usernames and passwords, the attacker can engage in a replay attack, where the attacker resends the valid authentication information to a target system to gain entry. Any system that uses constant identification and/or authentication information is susceptible to such as attack.

The list goes on. It must be obvious from the above discussion that passwords are a weak form of protection. The challenges in cyber security are innumerable. We attempt to overcome this weakness by making an effort to build a system that allows *authentication* of its users by using *biometric* data.

## 1.2 What is biometric authentication?

Biometric authentication refers to security processes that verify a user’s identity through unique biological traits such as retinas, irises, voices, facial characteristics, and fingerprints. Biometric authentication systems store this biometric data in order to verify a user’s identity when that user accesses their account. Because this data is unique to individual users, biometric authentication is generally more secure than traditional forms of multi-factor authentication.

Thanks to the dark web and account takeover fraud, authenticating the identity of users presents an ever-evolving challenge.

On one side of the authentication, challenge are users who demand speed and convenience and do not want to have to remember numerous passwords or make their way through a complex login or verification process every time they access an app or site. But, on the other hand, security requirements are quickly evolving to demand a rigorous approach to authentication.

Traditional methods of authentication such as username and password, knowledge-based authentication and SMS-based two-factor authentication have fallen out of favor due to a variety of security vulnerabilities ranging from account takeover to phishing to social engineering. Consequently, IT departments are exploring more robust authentication systems that mitigate the potential for theft and fraud.

**Authentication defined:**

Authentication can be defined as “the real-time corroboration (with an implied or notional confidence or level of trust) of a person’s claim to an identity previously established to enable their access to an electronic or digital asset.” Put simply, authentication is the process of determining whether someone or something is, in fact, who or what it declares itself to be.

After the user has been vetted via a remote identity proofing methodology, that same user usually does not need to go through the process again. Instead, the user can now use credentials (i.e., username and password) that were set up during the account opening to access the account or perform certain actions. The verification of those credentials is what we call authentication.

**Benefits of Facial Authentication:**

Unfortunately, traditional authentication methods, such as SMS-based two-factor authentication and knowledge-based authentication, are no longer considered best practices because of reliability and security concerns such as phishing attacks and man-in-the-browser exploits.

That’s why modern online companies are also increasingly looking for facial authentication solutions that more reliably establish the online identity of new users (during the account setup process) and existing users for authentication events — especially high-risk transactions such as wire transfers or password resets.

There are a number of tangible advantages of facial authentication over traditional authentication methods, which are briefly discussed below:

**Better Identity Assurance**

By requiring a valid government-issued ID and matching it to a selfie (with embedded 3D liveness detection), enterprises can have a much higher level of assurance that the person is who they claim to be.

**Easier for Users**

Thanks to the popularity of Apple’s Face ID, facial authentication has become more familiar and accepted. When combined with certified anti-spoofing capabilities, biometric-based authentication improves the user experience and reduces friction because of its ease of use, speed and omnichannel support (across mobile devices and webcams).

**Stronger Fraud Detection**

How well can the solution detect malware, bots and bad actors? Companies need to be able to confidently assess risk and make more informed application-processing decisions while minimizing fraud loss. Companies also need to better detect identity theft, account takeover and synthetic fraud — some of the largest threats facing online organizations today.

**Need for Speed**

While most users are willing to endure a bit of friction when creating an online account, they’re increasingly demanding authentication solutions that are fast, simple and reliable. Modern facial authentication can now quickly compare a new selfie to a previously captured selfie in seconds with a very high level of identity assurance.

**Cross-Platform Portability**

Since 3D face maps can be created on almost any device with a camera, true cross-platform biometric authentication is now possible — users can enrol using a laptop webcam and authenticate later from a smartphone or tablet. This means it’s now possible to use face authentication for everything from unlocking a car door to performing a secure password reset to accessing your bank account.

## 1.3 Choosing Facial Authentication Over Facial Recognition

For consumers and businesses alike, facial authentication is a win-win. Unlike facial recognition systems which are often performed without the user’s consent, facial authentication is permission-based and provides high levels of security to a user while letting them seamlessly access their own accounts or devices. The elegance of facial authentication is that the user does not need to be subjected to the entire identity proofing process — they just need to take a new selfie when then log into their favorite app.

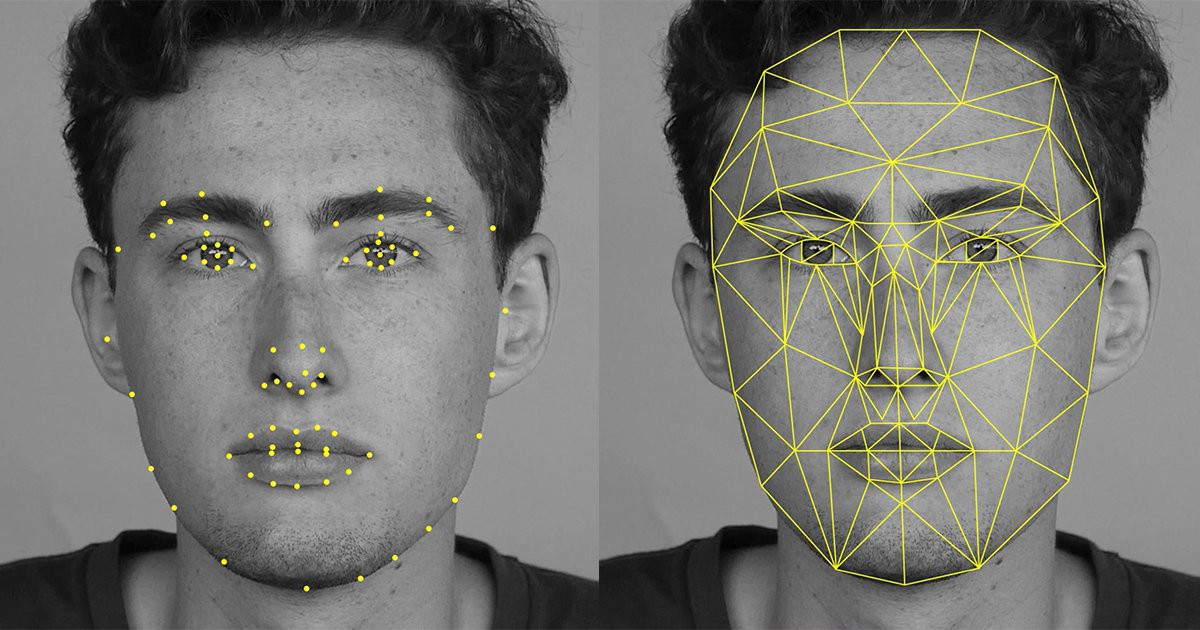


Figure . Facial pattern/ feature extraction.

For businesses, facial authentication solutions help ensure that only legitimate users are creating and accessing their online accounts — not some fraudster who has stolen the credentials or ID documents of identity theft victims off the dark web. Another key benefit of facial authentication is that businesses can leverage the same biometric data (i.e., a certified 3D face map) captured during enrolment and repurpose that same biometric for future authentication events. This means they can use the same solution for identity proofing and for ongoing user authentication.

Of course, no technology is entirely without risk. Facial recognition is highly data-intense, which can make processing and storage an obstacle. Despite enormous advances, recognizing faces from multiple camera angles or with obstructions (such as hats) is still not perfect. Plus, there have been controversies related to privacy issues, particularly in retail and government settings. This is why facial recognition should not be used for identity proofing, but instead leveraged with multifactor methods (i.e., facial authentication) to strengthen user access.

Biometric technologies including facial recognition and facial authentication represent the new wave of identity and authentication solutions. With their inherent and varied benefits, it’s important to understand the distinction between these technologies and how they can be deployed to solve your unique business challenges.

In addition to deploying facial recognition to facilitate facial authentication, we are also using *encryption* to safeguard the user’s data stored on the server.

## 1.4 Putting Encryption to work

The basic idea of encryption is to convert data into a form in which the original meaning is masked, and only those who are properly authorized can decipher or read it. This is done by scrambling the information using mathematical functions based on a number called a key. An inverse process, using the same or a different key, is used to unscramble (or decrypt) the information. If the same key is used for both encryption and decryption, the process is said to be symmetric. If different keys are used the process is defined as asymmetric.

Two of the most widely used encryption algorithms today are AES and RSA. Both are highly effective and secure, but they are typically used in different ways. Let’s take a look at how they differ.

**AES Encryption**

AES (Advanced Encryption Standard) has become the encryption algorithm of choice for governments, financial institutions, and security-conscious enterprises around the world. The U.S. National Security Agency (NSC) uses it to protect the country’s “top secret” information.

The AES algorithm successively applies a series of mathematical transformations to each 128-bit block of data. Because the computational requirements of this approach are low, AES can be used with consumer computing devices such as laptops and smartphones, as well as for quickly encrypting large amounts of data.

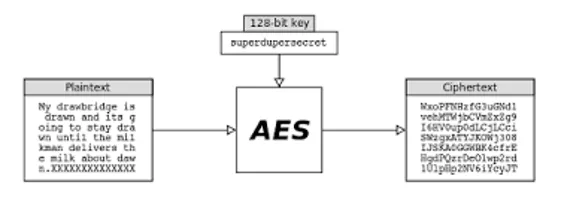


Figure . AES – A symmetric encoding algorithm.

For example, the IBM z14 mainframe series uses AES to enable pervasive encryption in which all the data in the entire system, whether at rest or in transit, is encrypted.

AES is a symmetric algorithm which uses the same 128, 192, or 256-bit key for both encryption and decryption (the security of an AES system increases exponentially with key length). With even a 128-bit key, the task of cracking AES by checking each of the 128 possible key values (a “brute force” attack) is so computationally intensive that even the fastest supercomputer would require, on average, more than 100 trillion years to do it. In fact, AES has never been cracked, and based on current technological trends, is expected to remain secure for years to come.

**RSA Encryption**

RSA is named after the MIT scientists - (Rivest, Shamir, and Adleman) - who first described it in 1977. It is an asymmetric algorithm that uses a publicly known key for encryption, but requires a different key, known only to the intended recipient, for decryption. In this system, appropriately called public key cryptography (PKC), the public key is the product of multiplying two huge prime numbers together. Only that product, 1024, 2048, or 4096 bits in length, is made public. But RSA decryption requires knowledge of the two prime factors of that product. Because there is no known method of calculating the prime factors of such large numbers, only the creator of the public key can also generate the private key required for decryption.

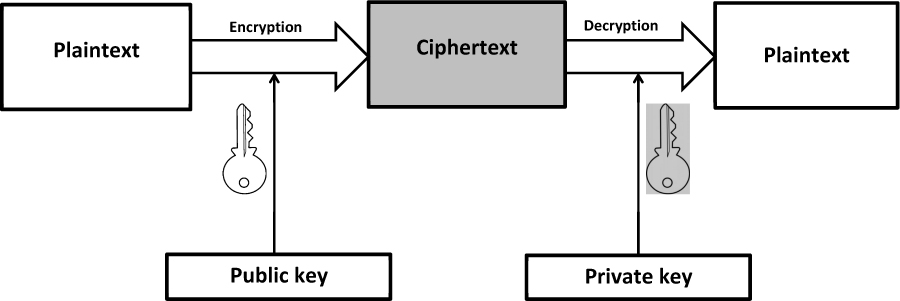


Figure . RSA – An asymmetric encoding algorithm.

RSA is more computationally intensive than AES, and much slower. It’s normally used to encrypt only small amounts of data. In our project we are demonstrating encryption using AES algorithm. It also enables faster encryption, making it ideal for software applications, firmware and hardware that require either low latency or high throughout, such as firewalls and routers.

# Chapter 2: METHODOLOGY: FACING THE CHALLENGE

Now, we discuss the methodology and programing that we are using to proceed towards creating the solution, as discussed in the previous chapter. We divide the process of building the into three parts:

* Web Development – HTML, CSS, JavaScript
* Server & Face Recognition – Python

## 2.1 Web Development

As we commenced with intention to implement the idea, we learnt that the optimal way to go about it is by learning each coding language/ framework one at a time. First, we focused our attention on learning the Web Development aspect.

In order to learn Web Development, we started with basics. HTML to begin with, followed by CSS. HTML is very easy to understand and required no prerequisite knowledge of coding. Let us understand the basics about HTML.

### 2.1.1 HTML

It stands for HyperText Markup Language. It is the most basic building block of the Web. It defines the meaning and structure of web content. "Hypertext" refers to links that connect web pages to one another, either within a single website or between websites. Links are a fundamental aspect of the Web. By uploading content to the Internet and linking it to pages created by other people, you become an active participant in the World Wide Web.

HTML uses "markup" to annotate text, images, and other content for display in a Web browser. HTML markup includes special "elements" such as <head>, <title>, <body>, <header>, <footer>, <article>, <section>, <p>, <div>, <span>, <img>, <aside>, <audio>, <canvas>, <datalist>, <details>, <embed>, <nav>, <output>, <progress>, <video>, <ul>, <ol>, <li> and many others.

An HTML element is set off from other text in a document by "tags", which consist of the element name surrounded by "<" and ">". The name of an element inside a tag is case insensitive. That is, it can be written in uppercase, lowercase, or a mixture. For example, the <title> tag can be written as <Title>, <TITLE>, or in any other way.

**Anatomy of an HTML document**

To wrap up the basics of individual HTML elements, let us now look at how individual elements are combined to form an entire HTML page. The below code serves as an example (which essentially contains the basic structure for a HTML document):



Figure . Basic structure of HTML code

**The following description makes it easier to understand:**

**<!DOCTYPE html>** — the doctype. It is required preamble. In the mists of time, when HTML was young (around 1991/92), doctypes were meant to act as links to a set of rules that the HTML page had to follow to be considered good HTML, which could mean automatic error checking and other useful things. However, these days, they don't do much, and are basically just needed to make sure your document behaves correctly. That's all you need to know for now.

**<html></html>** — the <html> element. This element wraps all the content on the entire page and is sometimes known as the root element.

**<head></head>** — the <head> element. This element acts as a container for all the stuff you want to include on the HTML page that isn't the content you are showing to your page's viewers. This includes things like keywords and a page description that you want to appear in search results, CSS to style our content, character set declarations and more.

**<meta charset="utf-8">** — This element sets the character set your document should use to UTF-8 which includes most characters from the vast majority of written languages. Essentially, it can now handle any textual content you might put on it. There is no reason not to set this and it can help avoid some problems later on.

**<title></title>** — the <title> element. This sets the title of your page, which is the title that appears in the browser tab the page is loaded in. It is also used to describe the page when you bookmark/ favorite it.

**<body></body>** — the <body> element. This contains all the content that you want to show to web users when they visit your page, whether that's text, images, videos, games, playable audio tracks or whatever else.

Additionally, we deal with marking up text as *heading*, *paragraph element*, inserting *image element* and inserting a *link element*. A brief discussion about these is given below:

**Paragraph Element**

Written as below, paragraph elements are for containing paragraphs of text.

<p>Any text goes between this tag</p>

**Heading Element**

Heading elements allow you to specify that certain parts of your content are headings — or subheadings. In the same way that a book has the main title, chapter titles and subtitles, an HTML document can too.

<h1>My main title</h1>

<h2>My top level heading</h2>

<h3>My subheading</h3>

<h4>My sub-subheading</h4>

HTML contains 6 heading levels, <h1>–<h6>, although you'll commonly only use 3 to 4 at most

**Image element**

Let's turn our attention to the <img> element:

<img src="images/image-file-name.png" alt="My test image">

The above code embeds an image into our page in the position it appears. It does this via the src (source) attribute, which contains the path to our image file.

This much understanding of HTML is good enough for us to commence with the next step, that is, *CSS*.

### 2.1.2 CSS

CSS is a language for specifying how web documents are presented to users — how they are styled, laid out, etc.

Presenting a document to a user means converting it into a form usable by your audience. Browsers, like Firefox, Chrome, or Edge, are designed to present documents visually, for example, on a computer screen, projector or printer.

CSS can be used for very basic document text styling — for example changing the color and size of headings and links. It can be used to create layout — for example turning a single column of text into a layout with a main content area and a sidebar for related information. It can even be used for effects such as animation.

**Adding CSS to our document**

The very first thing we need to do is to tell the HTML document that we have some CSS rules we want it to use. There are three different ways to apply CSS to an HTML document that you'll commonly come across, however, for now, we will look at the most usual and useful way of doing so — linking CSS from the head of your document.

We create a file in the same folder as our HTML document and save it as styles.css. The css extension shows that this is a CSS file.

To link styles.css to our one of the multiple web pages we are rendering during the project, say: usersite.html add the following line somewhere inside the <head> of the HTML document:



Figure . Linking CSS to a HTML page

**The basics to get us started:**

CSS is a rule-based language — you define rules specifying groups of styles that should be applied to particular elements or groups of elements on your web page. For example, "I want the main heading on my page to be shown as large red text."

To style a portion of webpage we need to grab or select that portion. To select that portion, *selectors* are used. There are three commonly used types of selectors in CSS. Element selector, Class selector and ID selector.

The following code shows a very simple CSS rule using *element* selector that would achieve the styling described above:



Figure . CSS code block

CSS can be used to make webpages look vibrant and live with animation, colors, structured in visually pleasing manner. However, our primary focus in this project is on adding the necessary functionalities rather than its aesthetic aspect.

Apart from HTML and CSS, for a webpage to behave as intended we make use of *JavaScript*.

### 2.1.3 JavaScript

JavaScript is a scripting or programming language that allows you to implement complex features on web pages — every time a web page does more than just sit there and display static information for you to look at — displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc. — you can bet that JavaScript is probably involved. It is the third layer of the layer cake of standard web technologies, two of which (HTML and CSS) we have covered earlier.

**Basics of JavaScript**

Let's briefly recap the story of what happens when you load a web page in a browser (first talked about in our How CSS works article). When you load a web page in your browser, you are running your code (the HTML, CSS, and JavaScript) inside an execution environment (the browser tab). This is like a factory that takes in raw materials (the code) and outputs a product (the web page).

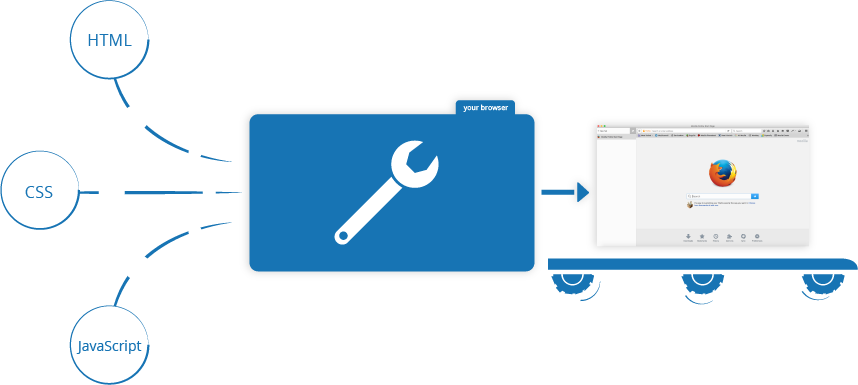


Figure . Behind the webpage

A very common use of JavaScript is to dynamically modify HTML and CSS to update a user interface, via the Document Object Model API (as mentioned above). Note that the code in your web documents is generally loaded and executed in the order it appears on the page.

If the JavaScript loads and tries to run before the HTML and CSS it is affecting has been loaded, errors can occur. Therefore, the aforementioned contingency has been avoided in our project.

**Adding JavaScript to our webpage:**

JavaScript is applied to a HTML page in a similar manner to CSS. Whereas CSS uses <link> elements to apply external stylesheets and <style> elements to apply internal stylesheets to HTML, JavaScript only needs one friend in the world of HTML — the <script> element.

In our project we have used both, internal as well as external JavaScript. However, using external script makes the webpage code more legible.

**Why JavaScript?**

JavaScript is mainly used for web-based applications and web browsers. But JavaScript is also used beyond the Web in software, servers and embedded hardware controls.It allows users to interact with web pages. There are almost no limits to the things you can do with JavaScript on a web page – below mentioned we mention just a few examples:

* Show or hide more information with the click of a button
* Change the color of a button when the mouse hovers over it
* Slide through a carousel of images on the homepage
* Zooming in or zooming out on an image
* Displaying a timer or count-down on a website
* Playing audio and video in a web page
* Displaying animations
* Using a drop-down hamburger menu

We have used JavaScript to dynamically populate dropdown menu on user’s webpage. Also, the webcam functionality was made possible by a JavaScript webcam.js library. The upload and download features are also its contribution.

### 2.2 Server & Face Recognition

Python has a number of web frameworks that can be used to create web apps and APIs. The most well-known is Django, a framework that has a set project structure and which includes many built-in tools. This can save time and effort for experienced programmers, but can be overwhelming. Flask applications tend to be written on a blank canvas, so to speak, and so are more suited to a contained application such as our project.

**.**

**.**

**.**

**.**

**.**

**.**

**.**

**.**

**Libraries/ Modules:**

A Python library is a reusable chunk of code that you may want to include in your programs/ projects. Compared to languages like C++ or C, a Python library does not pertain to any specific context in Python. Here, a ‘library’ loosely describes a collection of core modules.

A module is a Python file containing Python statements and definitions. For example, a file evenodd.py is a module, and we call it ‘evenodd’. We put similar code together in one module. This helps us modularize our code, and make it much easier to deal with. And not only that, a module grants us reusability. With a module, we don’t need to write the same code again for a new project that we take up.

A package, in essence, is like a directory holding sub-packages and modules. While we can create our own packages, we can also use one from the Python Package Index (PyPI) to use for our projects.

We have attempted to describe various packages and libraries used in our project as below:

* **face\_recognition:** Recognize and manipulate faces from Python or from the command line withthe world’s simplest face recognition library. Built using dlib’s state-of-the-art face recognition built with deep learning. The model has an accuracy of 99.38% on the Labeled Faces in the Wild benchmark. This also provides a simple face\_recognition command line tool that lets you do face recognition on a folder of images from the command line!
* **cv2:** OpenCV (Open source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage then Itseez (which was later acquired by Intel). The library is cross-platform and free for use under the open-source BSD license.
* **os:** The OS module in Python provides a way of using operating system dependent functionality. The functions that the OS module provides allows you to interface with the underlying operating system that Python is running on – be that Windows, Mac or Linux.
* **sys:** The sys module provides information about constants, functions and methods of the Python interpreter. dir(system) gives a summary of the available constants, functions and methods. Another possibility is the help() function. Using help(sys) provides valuable detail information.
* **subprocess:** The subprocess module present in Python (both 2. x and 3. x) is used to run new applications or programs through Python code by creating new processes. It also helps to obtain the input/output/error pipes as well as the exit codes of various commands.
* **Random:** Python offers random module that can generate random numbers. These are pseudo-random number as the sequence of number generated depends on the seed. If the seeding value is same, the sequence will be the same. For example, if you use 2 as the seeding value, you will always see the following sequence.
* **AES:** (Advanced Encryption Standard) is a symmetric block cipher standardized by NIST. It has fixed data block size of 16 bytes. Its keys can be 128, 192, or 256 bits long.
* **signal:** Python signal handlers are always executed in the main Python thread, even if the signal was received in another thread. This means that signals can't be used as a means of inter-thread communication. You can use the synchronization primitives from the threading module instead.
* **os.path:** The os.path module is always the path module suitable for the operating system Python is running on, and therefore usable for local paths. However, you can also import and use the individual modules if you want to manipulate a path that is always in one of the different formats.
* **listdir:** This method in python is used to get the list of all files and directories in the specified directory. If we don't specify any directory, then list of files and directories in the current working directory will be returned.
* **time:** Python has defined a module, “time” which allows us to handle various operations regarding time, its conversions and representations, which find its use in various applications in life. The beginning of time is started measuring from 1 January, 12:00 am, 1970 and this very time is termed as “epoch” in Python.
* **string:** The string module contains a number of useful constants and classes, as well as some deprecated legacy functions that are also available as methods on strings. In addition, Python’s built-in string classes support the sequence type methods described in the Sequence Types — str, unicode, list, tuple, bytearray, buffer, xrange section, and also the string-specific methods described in the String Methods section. To output formatted strings, use template strings or the % operator described in the String Formatting Operations section. Also, see the re module for string functions based on regular expressions.
* **win32api:** Python extensions for Microsoft Windows Provides access to much of the Win32 API, the ability to create and use COM objects, and the Pythonwin environment.
* **win32con:** it is a package that includes extensions for accessing the Win32 API, Windows COM bindings (e.g. modules like "win32api", "win32com", etc.) and the PythonWin IDE.

# Chapter 3: THE OUTCOME

# APPENDIX

# REFERENCES