



STORING AND DISPLAYING CERTIFICATES OF STUDENTS BY INSTITUTIONS USING DJANGO WITH PYTHON

¹Manichanthan, ²Harshavardana, ³Manideep, ⁴Bijaya Kumar

¹B.Tech. Student, ²B.Tech. Student, ³B.Tech. Student, ⁴Assistant Professor

^{1,2,3,4}Department of Computer Science and Engineering

^{1,2,3,4}Vardhaman College of Engineering, Shamshabad, Telangana

Abstract: Competing with others is vital in a busy, competitive society, and most students pick a different path than their peers because of this. They are displaying talents they do not possess by utilizing false credentials. Some pupils lose sight of their achievements over time if they don't save them for the future. Purely based on student achievements and subject-matter expertise, certain organizations are unable to evaluate the efficacy of the institutions. The goal of this thesis is to develop a website, that helps in preventing students from submitting fake certificates, stores student certificates from any institution for a longer period, and combines all the achievements of students from any institution in one location for easy retrieval and evaluation. The python-based Django framework is being used to build this website, and an SQLite database is being utilized to hold the certificates. The Front-End design was created using HTML, CSS, and Bootstrap templates. This paper depicted the project's architecture as well as the project requirements.

Index Terms - Credentials, Organization, Institution, achievements, Certification, subject matter.

I. INTRODUCTION

Today, achieving the desired return for your hard work may be difficult for a variety of reasons. One explanation is the use of forged credentials to demonstrate a person's labour or talents. It is illegal, nevertheless many continued to make use of it. To stop this, the benefits given to candidates with fake certificates must be terminated.

Many students excel in academic and extracurricular activities when they are young. However, as they become older, they often lose their way. They will eventually lose sight of their accomplishments along the process of growing due to a variety of factors, including concentrating on their careers, not seeing any progress in their achievement activities, and so on. Even though they wish to remember in the future about their prior accomplishments, they are unable.

Many corporate entities now choose to hire staff directly from colleges and other institutions. but to do that, they need to know which university ranks highest in terms of student involvement in academic activities both within and outside the institution, How many pupils participated actively in the work? Which institution has the best subject-matter competence among students? In this paper, we are creating a website that addresses all the above-mentioned issues.

1. Any institution may only be granted one account on this website, and that account will be managed by a designated individual (chosen by the institution's staff members), who alone has the authority to upload certificates and make modifications to certificates (Even the student does not have the authorization to upload their certificate).
2. Since this website can store certificates, they will continue to exist even decades from now. Any student may see their certificates at any time by just signing into their account, where all of their certificates will be shown.
3. Once the institution has been approved, the administrator will give them a special ID. With the help of this ID, any private or public organization can login as a visitor and view all the certifications that the institution's students have earned, which they can then validate.

For this website, Django with python is used to build dynamic web pages. Django makes it simple and well-liked technology to use by internally converting the python code into the SQL code for the database and offering a free SQLite database. The certificates of the students are kept in an SQLite database. Python was used to create the website's Back-End while bootstrap templates and Hyper Text Markup Language were used for Front-End design. Python's easy readability and reusability of the code make it a popular language to use.

II. LITERATURE SURVEY

Some of the paper publications and websites that helped to complete the project's objectives are listed below.

[1] Paras Nath Barwal and Pallavi Yadav (2014)

This study explains how to create responsive websites with HTML and CSS. Making responsive websites is important these days because they allow for panning, scrolling, and automatic resizing. Users can access the website from any device, and the website will automatically rearrange itself to make it more convenient for them. It is useful for Showing all content to users without losing any data.

[2] Mukesh, Shyam, Nitin and Adama (2020)

The study illustrates how students can use the website to advance their academic careers. This website is used as a platform for students to view their academic syllabus, to study last year's exam papers, user can also trade their used books, and receive notes for their preparation. But they cannot store the achievements of students.

[3] W3 Schools

This is an open and free website that explains how to write simple codes in languages such as HTML, CSS, and Python for Front-End and Back-End development. This website also contains sample code with outputs to assist users in developing their websites.

[4] Sitepoint (django photo sharing app)

This website assists developers who want to create a photo-sharing app in Python using Django technology. Everything from installation to final output is described on the website on how to build a Django project for a photo-sharing app. We are sharing our certificate photos through our website.

[5] Pooja kumari and Deepa Mangala (2015)

This study describes the frauds that occur in the corporate world as well as the methods and techniques used to stop these frauds in the literature. According to this paper, using a single fraud detection technique will not effectively reduce fraud in corporate. And in the case of our website, with the help of the institution's authorized person, we are somehow reducing the use of fake Certificates.

[6] Qing, Sajid, Ahmed and Yan (2019)

This paper discusses the use of Blockchain to prevent fraud in educational certificates. This project is built with blockchain technology, which helps to reduce data management costs. It allows users to directly verify the validity of the certificate using Blockchain without requiring permission from the organization that initially delivered the certificate.

After reviewing various papers, it is discovered that some project outcomes were designed, but they are not all in one place. It was previously not possible to retrieve various certificates and analyze the expertise of numerous students in one location. This project aims to address all of the disadvantages mentioned above, as well as to achieve all of the objectives in one location.

III. IMPLEMENTATION

3.1 ARCHITECTURAL DIAGRAM

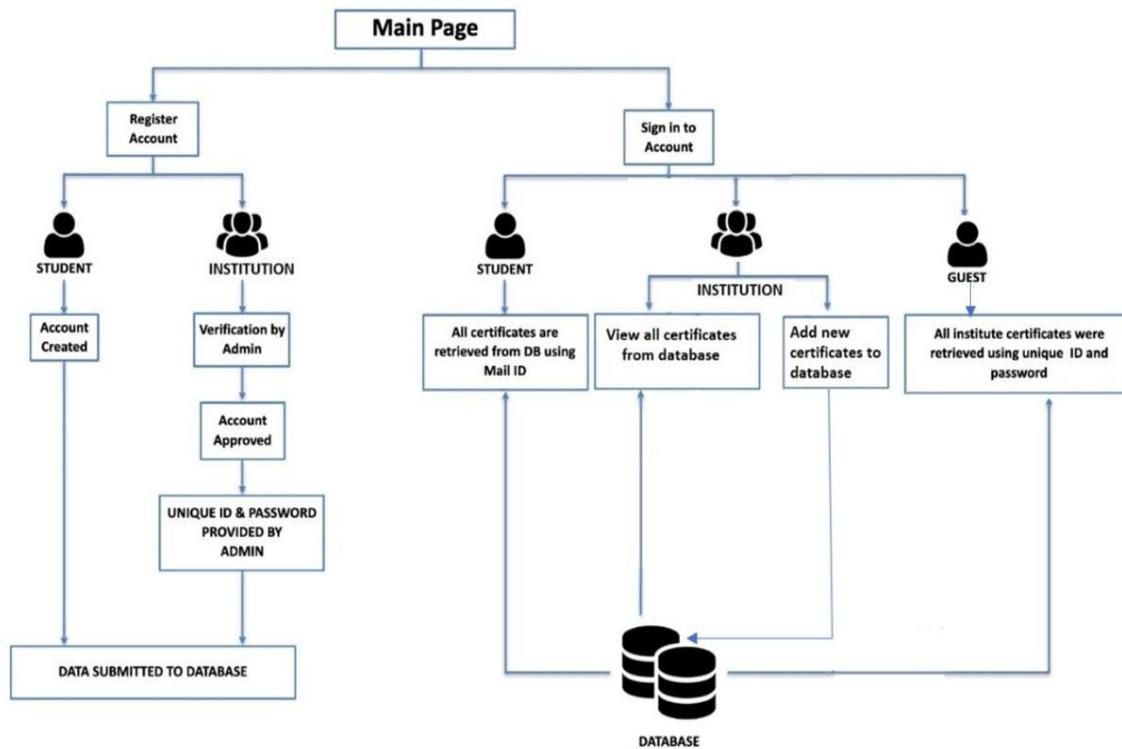


fig 1: website architecture

3.2 TECHNOLOGIES USED

DJANGO:-

Django was created between 2003 and 2005 by a team responsible for creating newspaper websites. Django is widely known as a server-side framework. When you run the server, it considers your system as a host. Django develops web applications using the MVT (Model View Template) design architecture. When you create an app with the basic skeleton, the Django administration site is automatically created. Django aims to assist you in developing software that is complete, versatile, secure, scalable, maintainable and portable.

PYTHON:-

Python is an Object-Oriented and Procedure-Oriented Interpreter language, which means it can be used to focus on data objects as well as functions (reusable code). It is the most popular programming language among developers because it is free and portable, and it also supports GUI. It is widely used because it is simple to understand and read. Python is a dynamically typed language, which means that defining the type of variable is not required while writing the code.

SQLITE:-

SQLite is a free and open-source database that aids in the interaction with relational databases. SQLite is a portable database resource. It could be extended in whatever programming language is used to access the database. When you launch the Django server for the first time, a new file called "db.sqlite3" will be created automatically. Django's default configuration makes use of an SQLite database. When a user gives their input, it acts as a database, storing all of the information. In this instance, the database file contains all the certificates and their associated information.

CSS:-

CSS is an acronym that stands for Cascading Style Sheets. It is employed in the styling of HTML-created web pages. For instance: You can add different colors, change the font style, font size, font family and even animations can also be added, among other things. The website will look beautiful when HTML and CSS are used together.

BOOTSTRAP:-

Bootstrap templates are predefined website designs that include all of the HTML and CSS work. These templates are used to create neat and clean web pages as quickly as possible. It is also possible to make customized changes to that template. When you download and use these templates, you get pre-defined web pages with HTML and CSS.

3.3 METHODOLOGY

The following are the key steps that we are taking while developing this project:

- 1) Setting up the platform for the project (In our case: Visual Studio Code)
- 2) Install Python, then add this to your Windows path.
- 3) Using the command below, install Django.
`pip install Django`
- 4) Go to the destination location, where you want to create the project by using the cd commands in the terminal.
- 5) Use the following commands to create a new project.
`django-admin startproject YOURPROJECTNAME`
`cd YOURPROJECTNAME`
- 6) Use this command in the terminal to create an app for the project.
`django-admin startapp APPNAME`
- 7) Copy the template folder and paste it into the newly created project folder (If you have any Front-End templates to add to this project)
- 8) Create functions in views.py to process data for each request sent.
- 9) The URL requests in urls.py are routed to the functions in views.py
- 10) Perform Migrations on the project, to save all the changes done to the database.
- 11) To start the server, enter this command in the terminal.
`python manage.py runserver`

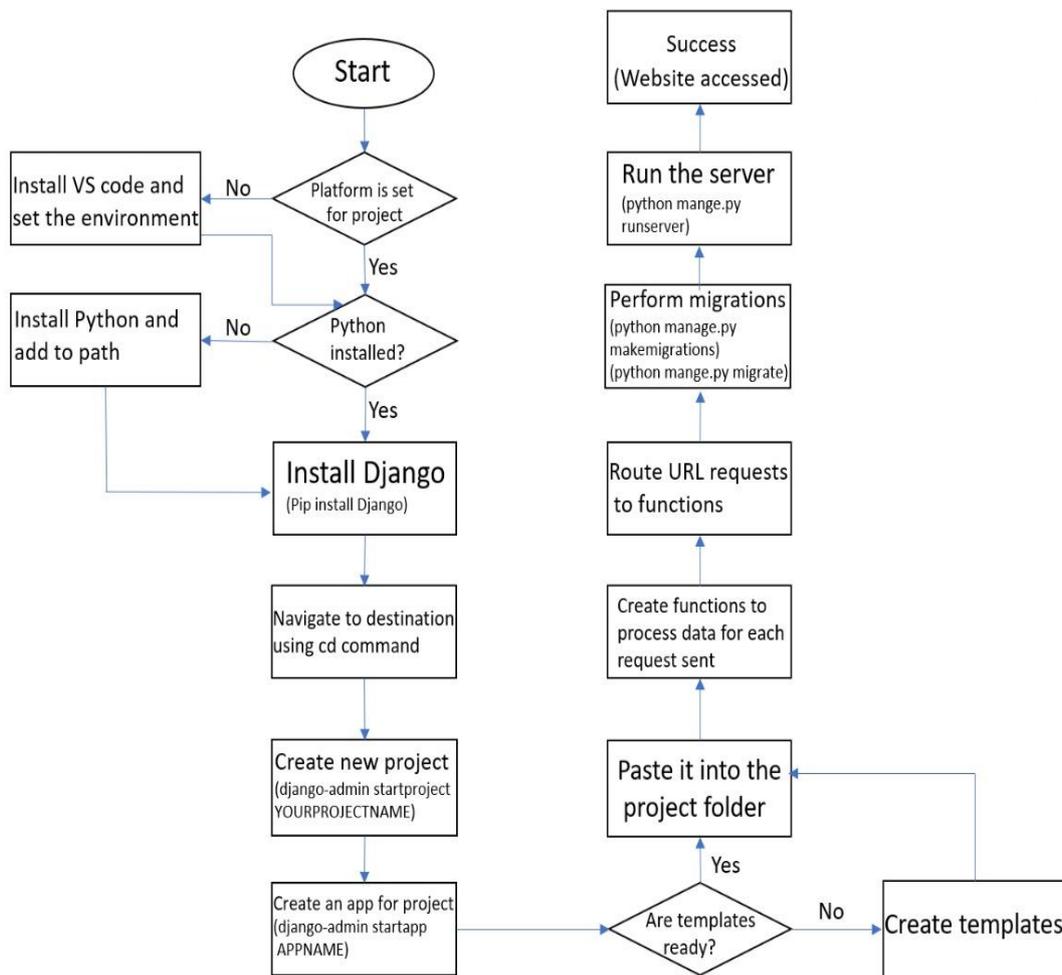


fig 2: procedure of the project

3.4 PROJECT REQUIREMENTS

The following features are required to complete this website.

1. Student Registration :

To create an account on this website, students must first register by entering their full name, email address, and creating a password. When all of these details have been gathered, the student account will be approved.

2. Institution Registration :

The process for creating an institution account differs slightly from that for students. The account handler from the institution is required to provide the following information such as the name of the institution, city, country, email address, password creation, and so on. The account will only be accepted when the admin has personally confirmed that the information provided is accurate (only one account is possible for one institution). The admin will then supply the institution with a unique ID and password.

3. Institution Login :

All of the certificates that the institution has submitted will be accessible anytime they enter into their account. These certifications are available for them to examine and modify. They can also upload new certificates. When uploading new certificates, they must supply basic information such as certificate name, student name, student mail, date, and certificate picture. The student mail ID in this instance is utilized to identify their certificates and enable accessing it through their account. The date they have provided will be utilized to filter out those certificates.

4. Student Login :

The student's email address was used to store all the certificates along with some additional information. As a result, if a specific student wants to get his or her certificates, they must log into their account. All certificates supplied to that student with their mail ID (regardless of the institution) will then be displayed there. To protect their privacy, we provide them the option to conceal or reveal their certificates. Even if the institution uploads a student's certificate, the student has the option to make it visible or hide it from the guest's view. Apart from this operation, the student has no authority to perform any other operations on the certificates.

5. Guest Login :

After their account is approved, each institution is given a unique ID and password, with which an individual can login as a visitor and view all of the achievements of that institute's students. Additionally, visitors can also filter the certificates according to their preferences (by student name, student mail, and range of dates). This allows them to easily analyze the students' engagement in the academic outcomes of that institution.

6. DATA Handling:

Connecting SQLite with Django handles the data from the database file. To connect SQLite and Django, you must follow a couple of predefined steps.

a. Defining Models.

We must create models to store information. To define those models, we have to construct a class named after that model which is inherited from models. Model attributes are declared as class variables.

b. Making Migrations.

Migrations enable database schemas to evolve in response to changing requirements. Migrations are useful for applying schema changes to the environment.

To migrate, use the commands listed below.

```
python manage.py makemigrations
python manage.py migrate
```

Once the connection has been successfully established, queries can be passed to perform operations on the data.

IV. RESULTS AND OUTPUTS

A few images illustrating the website's primary features are provided below. The project's goals are described through these images, which show that they were successfully carried out. creating an account for the institution, a student account, adding certificates to the website, Viewing certificates through an institution login, viewing certificates as a visitor using a special ID and password, students retrieving their certificates from various universities, and website images that serve as descriptions.

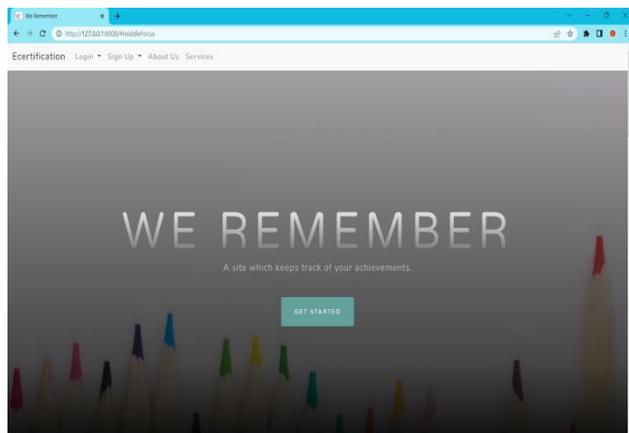


fig 3: main page

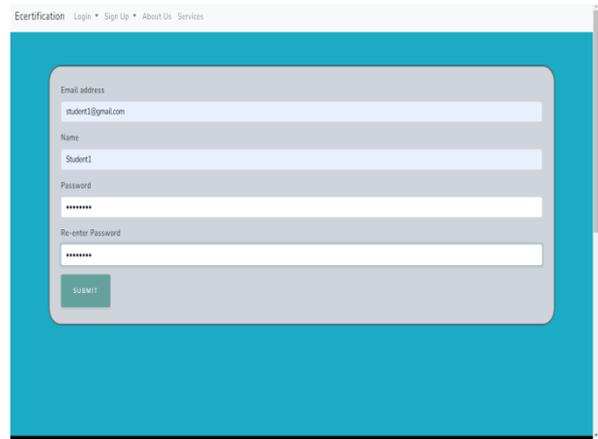


fig 4: student sign up



fig 5: institution sign-up

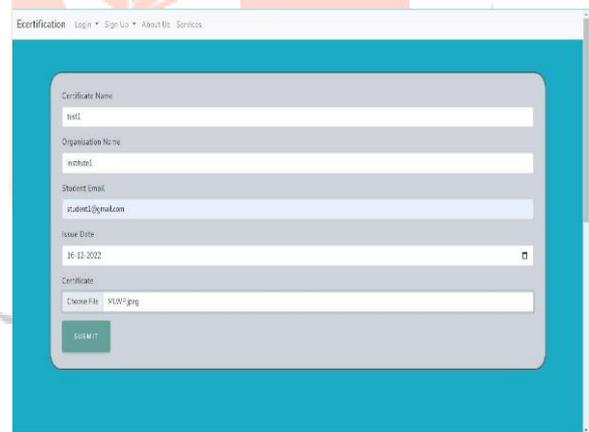


fig 6: adding certificates

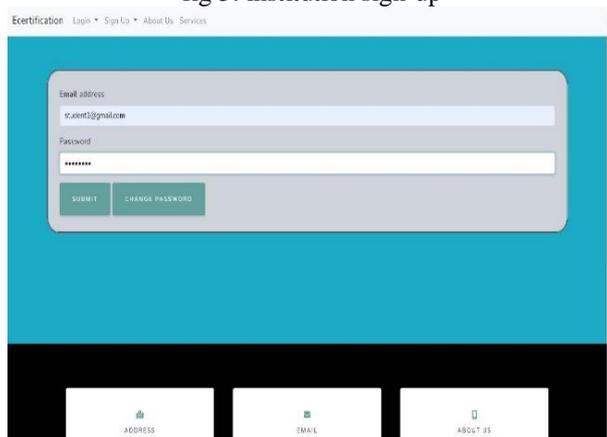


fig 7: student login



fig 8: certificates in the student account

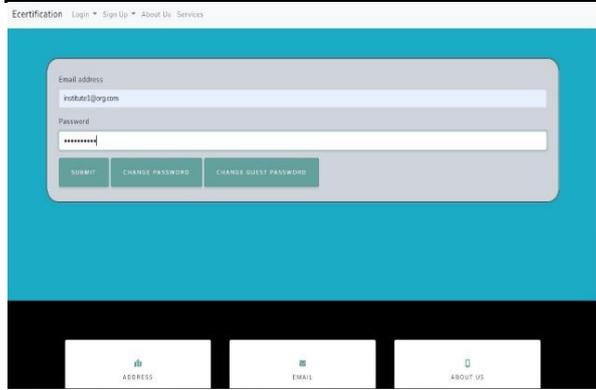


fig 9: institution login

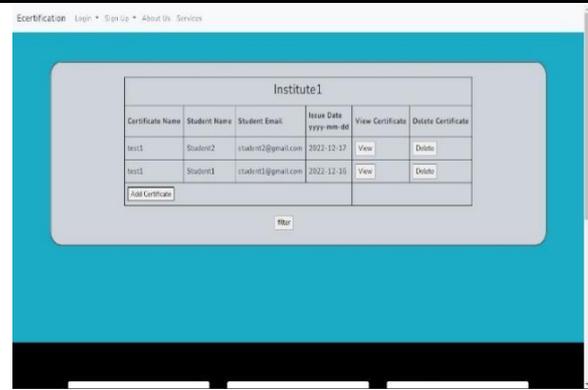


fig 10: certificates in institution account

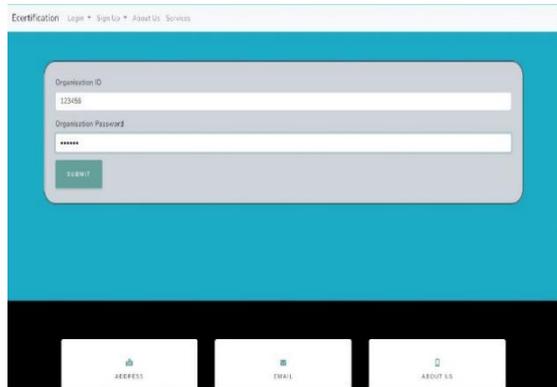


fig 11: guest login

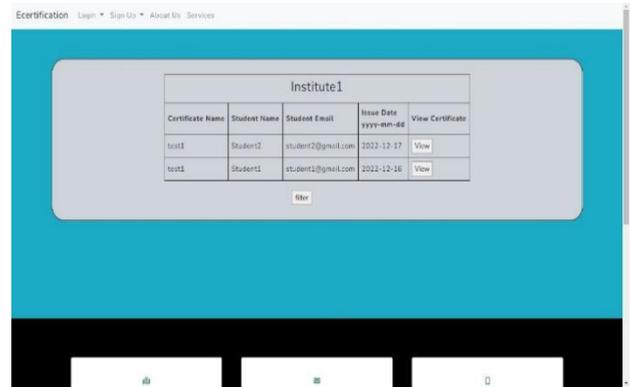


fig 12: certificates in the guest login

One of the main advantages of utilizing Django with Python is that there is no need to develop an admin page; it immediately provides an admin interface, which we can work on directly. It reads model metadata to give a rapid, model-centric interface. To process data, the backend automatically translates Python code into SQL queries.

V. CONCLUSION AND FUTURE SCOPE

This website was built with Django using Python and offers a free SQLite database. This database can be used to store certificates uploaded by any institution, and those certificates can be kept for a longer period and used for future reference. Django comes with some built-in features for dealing with user-submitted data and making it secure for an application. These features filter out potentially dangerous characters in requests, ensuring that the data is safe for the application to use.

Every student achievement from an institution can be gathered in one place, making it easier for organizations to assess students subject matter expertise.

In the future, we can also grant information (institute credentials) access to the registered authorized organizations in that area, meaning that only those organizations will be able to learn the institution credentials without personally asking them. The security provided to sensitive data in the database can be improved by using "Security Libraries for Django", "Clickjacking Protection" and other similar tools.

The scope of this website's use may expand and assist employees in private or government organizations with their certificates.

REFERENCES

- [1] Paras Nath Barwal and Pallavi Yadav, "Designing Responsive Websites using HTML and CSS", International Journal of Scientific and Technology Research, vol 3, issue 11, Nov 2014.
- [2] Mukesh, Shyam, Nitin and Adamyia, "A Django-based educational Resource sharing website: Sheric", Journal of Scientific Research, 64, pg 138-152, doi: 10.37398/JSR.2020.640134, 2020.
- [3] <https://www.w3schools.com/>
- [4] <https://www.sitepoint.com/django-photo-sharing-app/>
- [5] Pooja and Deepa, "Corporate fraud prevention and detection: Revisiting the literature", Journal of commerce and accounting research, vol 4, Issue 1, Jan 2015.
- [6] Qing, Sajid, Ahmed and Yan, "Frauds Avoidance of Digital Educational Certificates using blockchain", International journal of scientific and research publications, doi: 10.29322/IJSRP.9.11.2019.p9564, 2019.
- [7] Toyotaro, Suzumura, Michiaki and tatsubori, "HTML templates that fly a template engine approach to automated offloading from server to client", International World Wide Web Conference, pg 951-960, doi: 10.1145/1526709.1526837, Jan 2009.
- [8] Felix Freiling, Sebastian Nemetz and Sven Schmitt, "A standardized corpus for SQLite database forensics", Digital Investigation, vol 24, pg S121-S130, 1742-2876, doi: 10.1016/j.diin.2018.01.015, 2018.
- [9] J.Tu, L.Chen, j.Chou, H. Ding and B. Xu, "A method of Optimizing Django Based on Greedy strategy", 10th Web information system and application conference, pp 176-179, doi: 10.1109/WISA.2013.41, 2013.
- [10] Vivek Thoutam, "A study on python web application framework", Journal of Electronics, computer networking and applied mathematics(JECNAM), ISSN:2799-1156, 1(01), pg 48-55, doi: 10.55529/jecnam.11.48.55, 2021.