

A PROJECT REPORT
ON
“BLOG WEB APPLICATION”
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF
DIPLOMA IN
INFORMATION TECHNOLOGY



SUBMITTED TO
MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI
SUBMITTED BY

Name of Student(s) (Full Name)

Enrollment No.

1. Vijay Tiwari

1805220250

2. Anurag Pandey

1805220251

3. Viraj Gavas

1805220273

4. Akash Adarkar

1905220033

GUIDED BY
Mr. Chintamani Chavan



THAKUR POLYTECHNIC

JUNE 2021



THAKUR POLYTECHNIC

CERTIFICATE

This is to Certify that the project report entitled “Blog Web Application ”Was successfully completed by Student of sixth semester Diploma in (**Information Technology**).

- | | |
|------------------|------------|
| 1. Vijay Tiwari | 1805220250 |
| 2. Anurag | 1805220251 |
| 3. Viraj Gavas | 1805220273 |
| 4. Akash Adarkar | 1905220033 |

in partial fulfillment of the requirements for the award of the Diploma in **Information Technology** and submitted to the Department of **Information Technology** of **Thakur Polytechnic** work carried out during a period for the academic year **2020-21** as per curriculum .

Mr. Chintamani Chavan

Guide

Mrs. Suwarna Thakre

HOD

Dr. Shankarayya M. Ganechari

External Examiner

Principal

ACKNOWLEDGEMENT

With deepest gratitude, we would like to thank each and every person who has helped us in this project.

We would also like to acknowledge and express our gratitude to the following people for their magnificent support and contribution in our journey to create this project.

We pay homage to our parents and our subject teachers whose teachings lit a burning desire within us.

We have walked in the shadow of their knowledge and would like to pay special honour to Respected Mr. Chintamani Chauhan Sir , Respected Dr. Ganesh Ganechari Sir, our College Principal Sir, Ms. Suwarna Thakre Madam, our Head of Department and also mainly our all teaching and non-teaching staff members of I.F. Department.

Abstract

“In Our project of Blog Web App we built a web application in which in one person with knowledge of any specific topic can write an article and post it share it with the people worldwide so that they can use it research purposes , entertainment , education ,business promotions and many others fields .In the Project of Blog Web app We have used Django as a framework of python and other front -end technologies for building this site .This web site has almost all the features that compete with the modern day site and we made responsive so that we can also use it on mobile screens.It also has password change / reset functionality through email (SMTP) if the users forget their passwords “

Table of Contents

Chapter No.	Chapter Name	Page No.
1	Introduction	1
1.1	Introduction	2
2	Literature Survey	3
2.1	Abstract	4
2.2	Literature Review	5
3	Scope of the Project	9
3.1	Scope	10
4	Methodology	12
4.1	Methodology	13
5	Details of designs , working and process	15
5.1	System Modules	16
5.2	Data Flow Diagram	19
5.3	Use Case Diagram	24
5.4	Technologies Used	28
6	Results and Applications	36
7	Conclusions and Future Scope	51
7.1	Conclusion	52
7.2	Future Scope	53
References and Bibliography		55

Chapter-1

Introduction

Introduction

Blogging has become such a mania that a new blog is being created every second of every minute of every hour of every day. A blog is your best bet for a voice among the online crowd. Blogs were usually the work of a single individual, occasionally of a small group, and often covered a single subject. More recently, "multi-author blogs" (MABs) have developed, with posts written by large numbers of authors and professionally edited. MABs from newspapers, other media outlets, universities, think tanks, advocacy groups, and similar institutions account for an increasing quantity of blog traffic. The rise of Twitter and other "microblogging" systems helps integrate MABs and single-author blogs into societal new streams. Blog can also be used as a verb, meaning to maintain or add content to a blog. A novel is a long, fictional narrative which describes intimate human experiences. WEBLOG is a combination of both Blog as well as Novels. Blog contain the Information of various things related to Technology, Education, News, International, Business, Sports, Entertainment and ongoing college activities. The main aim of this project is to provide data to users in only one site. Users can gather the information from one site as well as give their feedback and create their own blog. Users can post their views and thoughts and analyze themselves. Besides all such core functionalities,

the application also includes features like FAQ, request, feedback etc. so as to provide a satisfactory user experience.

Chapter 2

Literature

Survey

Abstract

“In Our project of Blog Web App we built a web application in which in one person with knowledge of any specific topic can write an article and post it share it with the people worldwide so that they can use it research purposes , entertainment , education ,business promotions and many others fields .In the Project of Blog Web app

We have used Django as a framework of python and other front -end technologies for building this site .This web site has almost all the features that compete with the modern day site and we made responsive so that we can also use it on mobile screens.It also has password change / reset functionality through email (SMTP) if the users forget their passwords “

Literature Review

In this section we presented reviews of all related research papers , previously built projects , project reports , sources of learning technologies etc.

1. Google Bloggers :

Google Bloggers is a blog web application by Google which lets users with google account can post articles and do earn from the articles they post this application was the inspirations for us to built a blog application

2. Social Networking for Web-Based Communities:

From Social Networking for Web-Based Communities(2013) by Tomayess B.T Issa we learnt about the web application as social networking site

3. Social Networking:

From Social Networking (2019) by Matthew N.O Sadiku,Adedamola A.Omotoso,Sarhan M Musa took the plan of building a social media site for sharing content to others

4. The Django Based Educational Resource sharing Website:

From The django Based Educational Resource sharing Website(2020) by Adamya Shayam gave us an methodology for uploading media content on website to make our site more functional for the user

5. Web Application Development:

From Web Application Development by Ashish Arora we learnt about

the basic idea for web development for web application and its advantages and disadvantages of web application .

6. Issues and Challenges of using Web Blogs as a Medium for Research Communication:

From Issues and Challenges of using Web Blogs as a Medium for Research Communication by Zuwati Hasim we gathered information about how users use the blogs for research and resource collection and it gave us an idea for search queries of blogs to show relevant information /results

7. Tumblr:

Tumblr is a blog web application which lets users with tumblr account can post articles and do earn from the articles they post this application was the inspirations for us to built a blog application

8. Medium:

Medium is a blog web application which lets users with a medium account can post articles and earn from the articles they post. This application was the inspiration for us to build a blog application. It is an easy-to-use blogging platform with limited social networking features.

9. Ghost:

Ghost is a blog web application with Simplified user interface which lets users with ghost account can post articles and do earn from the articles they post this application was the inspirations for us to built a blog application

10.Django Crash Course:

From Django Crash Course by Daniel Roy Greenfeld and Audrey Roy

Greenfield we learnt Django framework about different aspects of django framework about views , URLs ,models ,templates , admin etc.

11.Bootstrap:

From Bootstrap by Maher Flosa we learnt about the reference file for Bootstrap how to use Bootstrap classes

12.SQLite Database:

From SQLite Database by Waseem Wsm we learnt about the reference file for Sqlite database which came by default with django framework to store our data

13.Javascript Reference:

From Javascript Reference b y Majo Flosa we used it as our reference for javascript in front end used in the front of the project

14.Optimizing Front-end Applications and Javascript:

From Optimizing Front-End Applications and Javascript by V . Myalapalli we used it as our reference for to learn javascript and imply as much as the functionality for project

15.The Definitive Guide to Django Web Development Done Right:

The Definitive Guide to Django Web Development Done Right by Obak Kando, Adrian Holovaty used it as our reference for the django framework in a more comprehensive way .

16.Photo Blog Web Application using Twitter Bootstrap:

From Photo Blog Web Application using Twitter Bootstrap(2015) by Daniel J LeBlanc gave us an methodology for uploading media content on website with the use of Bootstrap

17.Django 2 by Example:

From Django2 by Example by Packt Publication we took the features like comments and tags for our posts/articles. These are additional features which made our site more appealing and flexible and als to compete with modern-day sites

18.Django 3 for Beginners:

From Django3 for Beginners by Wiiliam Vincent we learnt about basic working and terminologies , different approaches of writing django code for blog application we learnt about the admin pages , templates etc.

19.Blog in web application: a software engineering perspective :

From Blog in web application: a software engineering perspective by Karan Gupta, Anita Goel gave us the importance of blogging in the field of education , entertainment , business etc.

20.A python Book:Beginning Python,advanced Python and Python Exercises:

A python Book:Beginning Python,advanced Python and Python Exercises by Kedar Jain we learnt about the basic of python programing language, about OOP concepts , different paradigms ,

21.Learn to program with python3:

From Learn to program with python3 by Apress we learnt about the basic of python programing language, about OOP concepts , different paradigms

22.Webblog:

From Webblog by Kritika Chauhan we looked in the report for the blog web application .

23.Research Methodologies for Beginners:

From Research Methodologies for Beginners by Kitsakorn Locharoenrat by which we learnt about the research writing for the project

Chapter 3.

Scope of the Project

Scope

It may help collecting perfect management in details in a very short time, the collection will be obvious ,simple and sensible .it will help a person to know the management of passed year perfectly and widely .it also helps in current all works related to Online blogging System .it will be also reduced the cost of collecting the management & collection procedure will go on smoothly.

Our project aims at Business process automation.e we have tried to computerize various process of Blog Web Application

- In computer system the person has to fill the various forms & numbers of copies of forms
- In the computer system it is necessary to create the manifest but we can directly print it which saves our time.
- To assist the staff in capturing the effort spent on their respective working areas
- To utilize the resources in an efficient manner by in increasing their productivity through automation
- the system generates types of information that can be for various purposes

- to satisfy the user requirement
- be easy to understand by the user and operator
- be easy to operate
- have a good user interface
- be expandable
- Delivered on schedule within the budget.
- To use the personal images in greeting our loved ones.To use the music of our choice to greet in the language we want.
- To provide a personalized mp4 video to the customers.
- To satisfy the customers and provide them with the ordered video before time

Chapter 4.

Methodology

Methodology

The methodology we followed for this project was agile methodology

AGILE methodology is a practice that promotes **continuous iteration** of development and testing throughout the software development life cycle of the project. In the Agile model, both development and testing activities are concurrent, unlike the Waterfall model.



The **Agile software development** methodology is one of the simplest and effective processes to turn a vision for a business need into software solutions. Agile is a term used to describe software development approaches that employ continual planning, learning, improvement, team collaboration, evolutionary development, and early delivery. It encourages flexible responses to change.

Agile methodology helped us to build the web application in small chunks and

without messing up the others components and features of the project.

We used this methodology because our aim at first was to implement basic necessary elements for the blog web app then we would work on additional features which were already discussed but later we would come up with more ideas or modifications with design or templates for front end we change or alter this without causing an error in the currently working other components of the application.

And for our future deployment we had to keep in mind about services which we will be using for databases , DNS , servers etc.

So to implement these extras services which easily done using agile methodology. It would be difficult to implement these features if we used the traditional methodology

So we followed this approach ,initially we built the integral part which was to create articles ,post articles and user authentication i.e. login ,logout functionality

Then we moved to database , admin controls , templates to beautify the site and after that we added features like comments section and profile editing etc.

Advantages of agile methodology

1. Persistent software delivery
2. Increased stakeholder satisfaction
3. Inspect and Adapt
4. Welcome to changes at any stages
5. Design is important
6. Daily Interaction

Chapter 5.

Details of designs, working and processes

System Modules

The modules involved in this project are:

1. User
2. Admin

1. User

In this module :

- User can sign up
- User can login
- User can upload multiple images
- User can choose images from database
- User can add information
- User can add comments
- User can search for any blog article

2. Admin

In this module :

- Admin can block user id
- Admin can resume user id
- Admin can see all pages

- Admin can maintain all records of users
- Admin can maintain all site
- Admin can access and process all requests
- Admin can delete/update/selected users
- Admin can provide all information related to any topic

Change account Information :

Update Account information like Name ,Email-address and their details

Search Reference Case:-

System will provide an interface for Administrator & Subordinate searching comprehensively the reference information from the blogs of application on the basis of various parameters, like Title of blogs e.g. health

Inputs Requirements of the System:

- User Information
- Login information
- Comment information
- Search Query information
- Recent Post Information

- Registered Information
- Topic Information

Outputs Requirements of the System

- Blog Information
- Assigned information of the user
- Search information

Maintenance : the system allows following Maintenance process

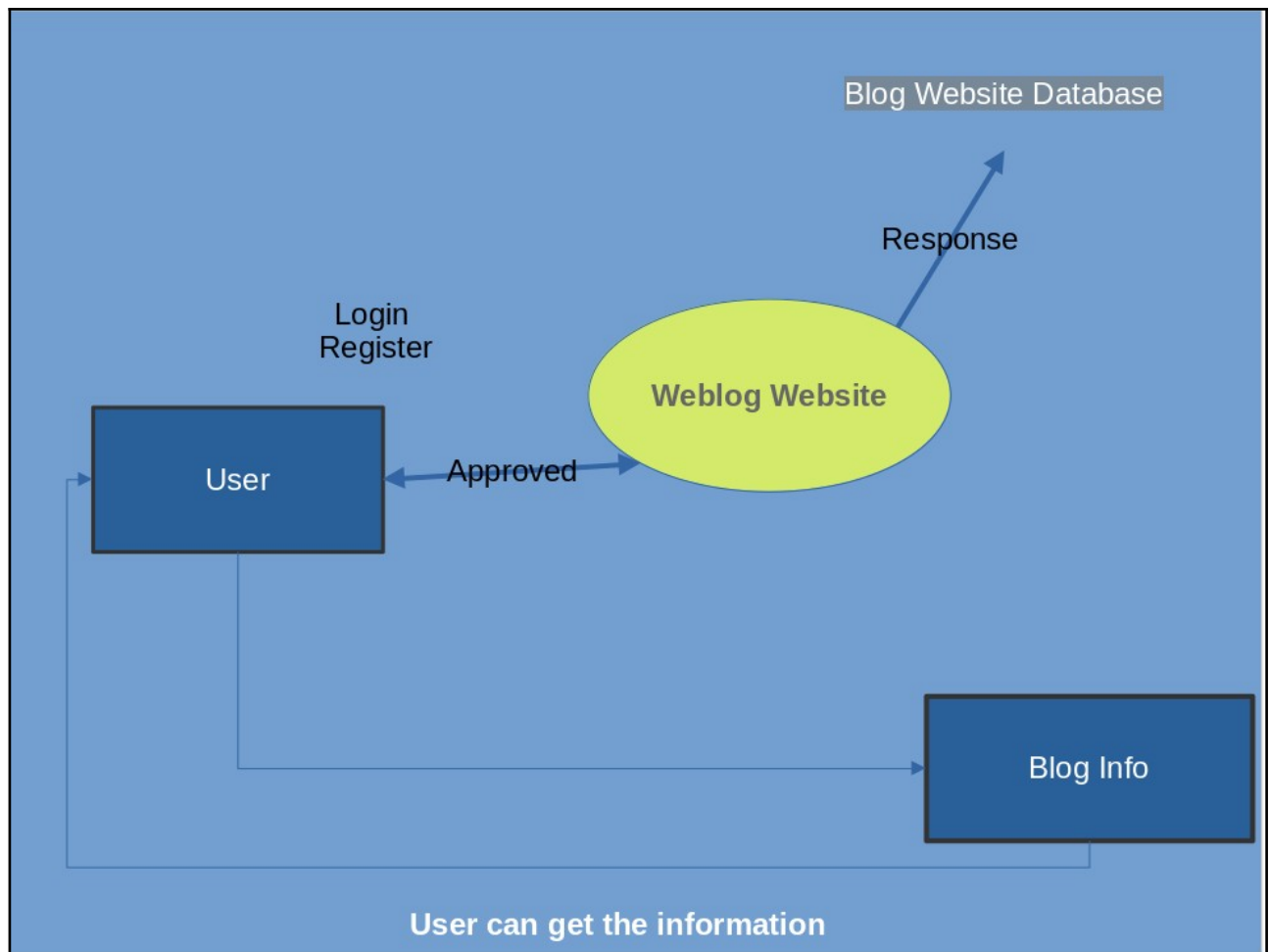
- Manage Blogs
- Manage Blogs Comments
- Manage Recent Posts
- Manage users
- Manage Images

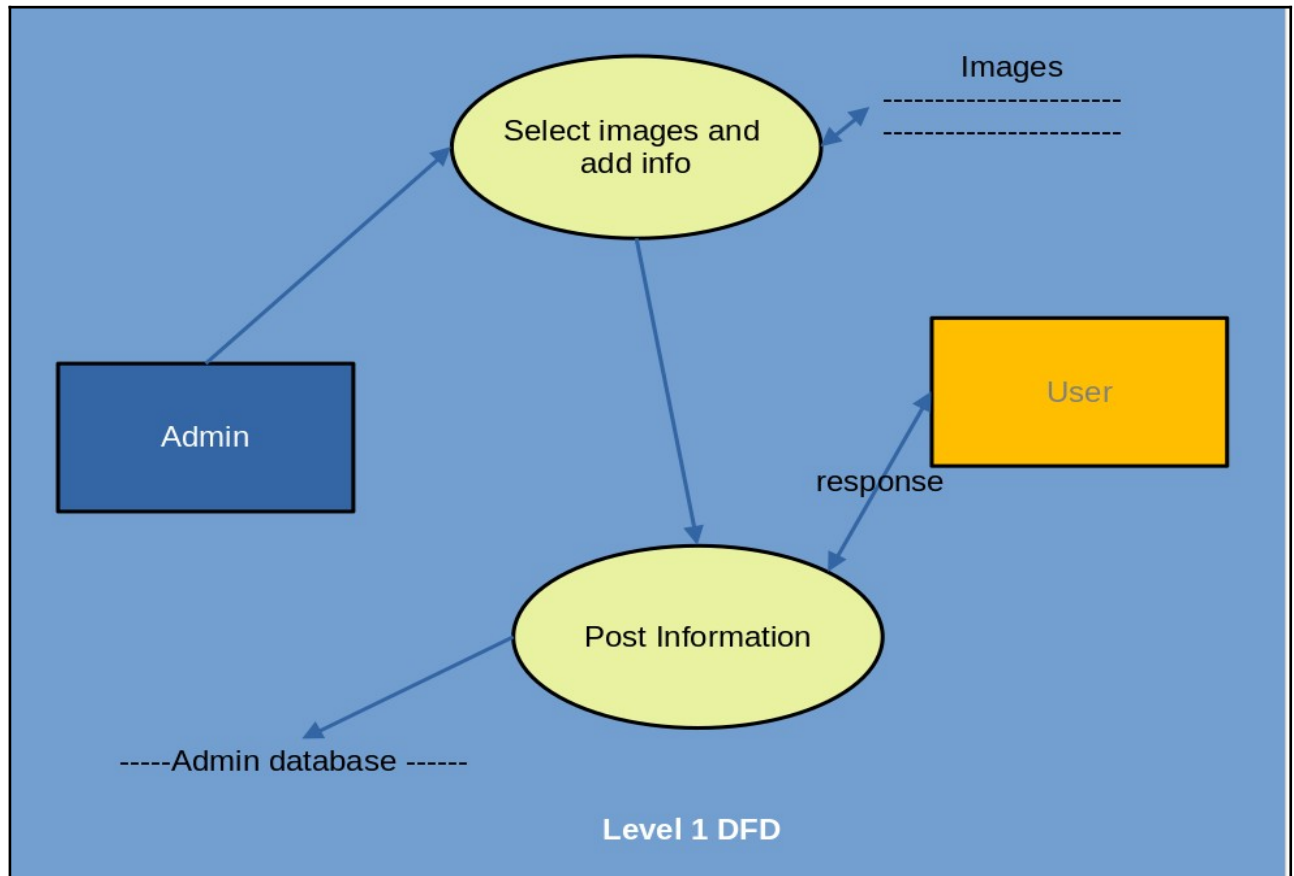
Data Flow Diagram (DFD)

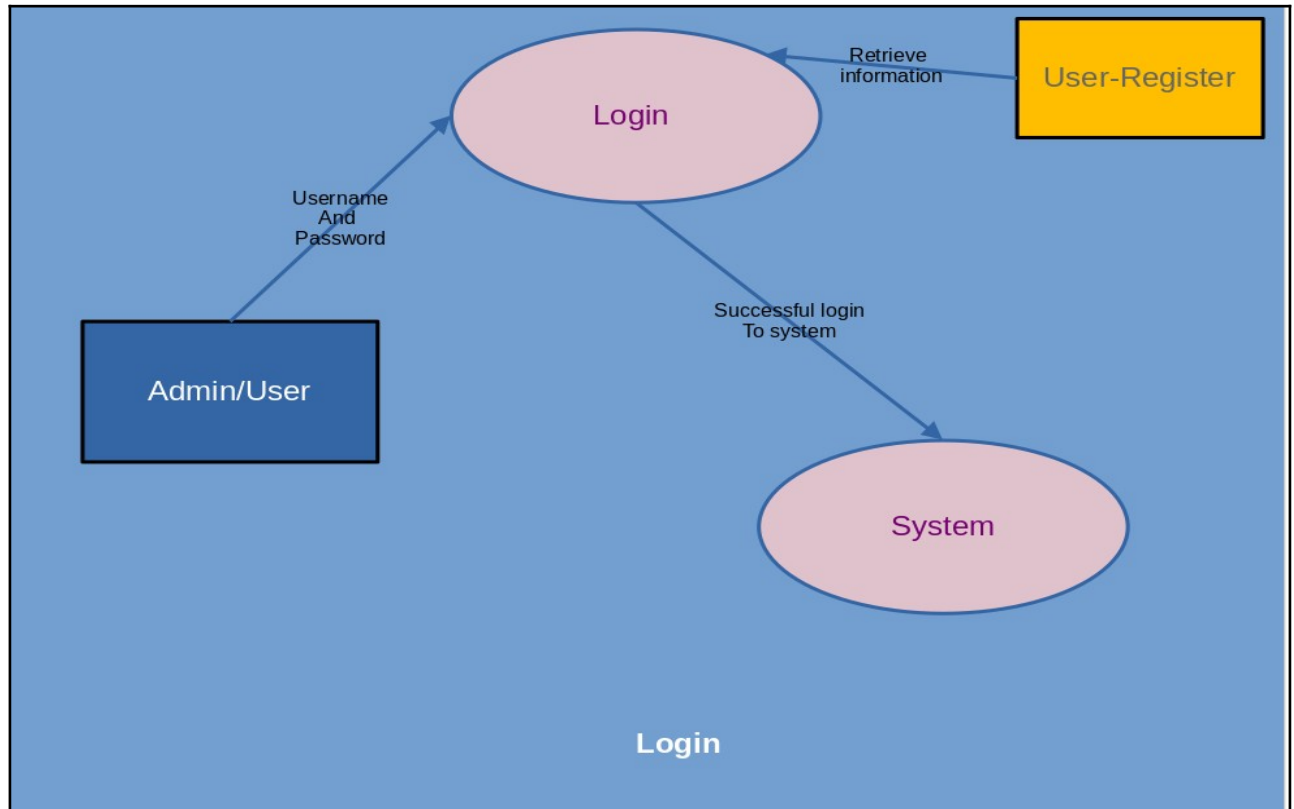
Data Flow Diagram model the flow of the data into through and out of the an information system

- ❖ Show the process that change or transform data
- ❖ Show the movement of data between process
- ❖ Represent a system as a network of processes which transform data flowing between them

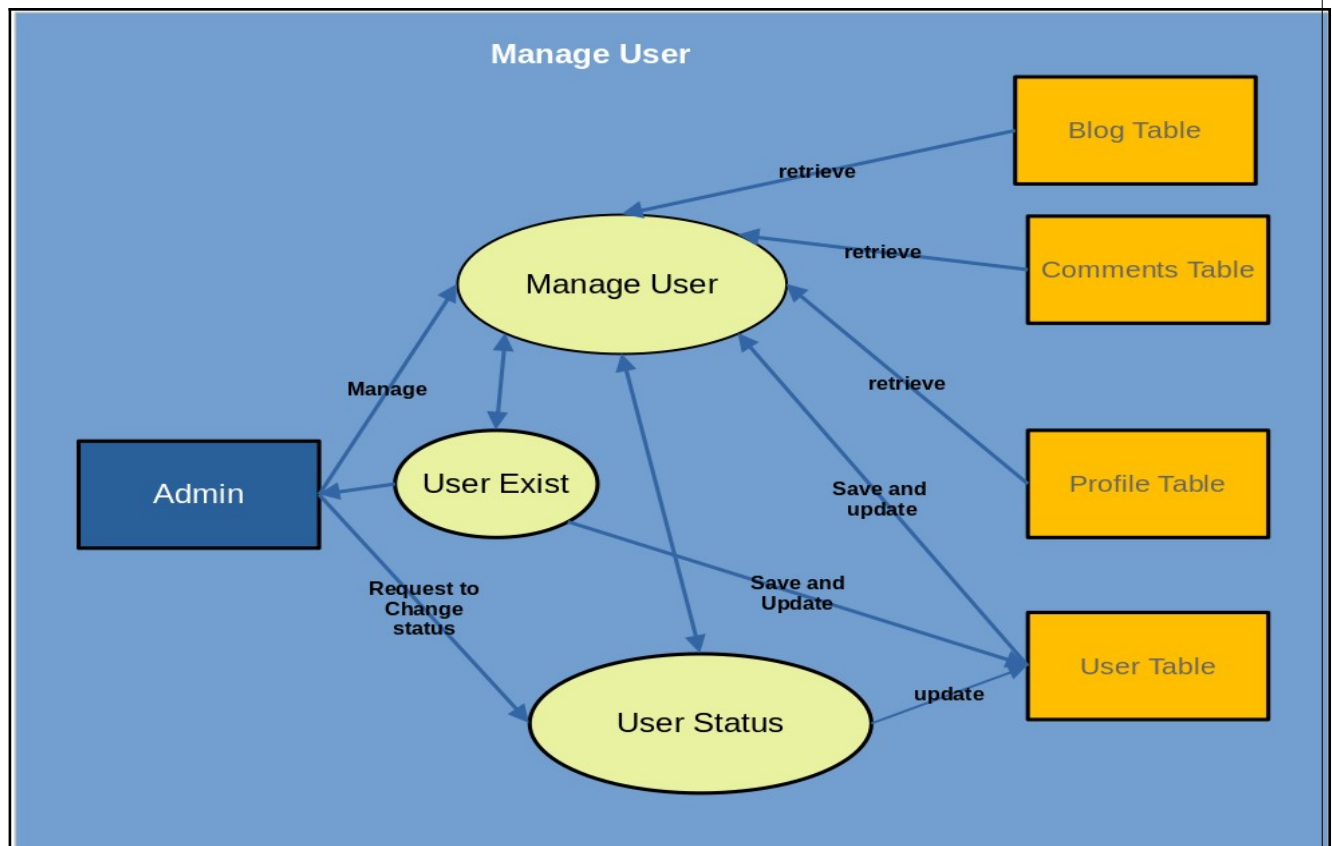
Context Data Flow Diagram







LOGIN



MANAGE USER

System Requirements

The only link to an external system is the link to the SQLite Database to verify the membership of a User and to store all the data provided by the customer. The Admin believes that a member is much more likely to be an effective reviewer and has imposed a membership requirement for a Reviewer. The Database fields of interest to the Admin are member's name, membership (ID) number, contact numbers and e-mail addresses. The details of new members signing up for the events can be added in the list of the previous members and the amendments can be made to the

list accordingly. The requests of each customer can be inserted in the database so that the admin can process it for further use.

Clients, Users, User

These are the ones who visit this particular website and thus have full access to the site and all the functionalities prevailing in it. The user can choose media and upload multiple files. Their session expires as the user selects a song for the background view. The user can select any festival or event according to his requirement.

Functional and Data Requirements

The requirements gathering process is intensified and focused specifically on software. To understand the nature of the program(s) to be built, the software engineer (“analyst”) must understand the information domain for the software, as well as required function, behavior, performance, and interfacing. Requirements for both the system and the software are documented and reviewed with the customer.

Non-Functional Requirements

The software developed here assumes the use of Django for connection between the Front End and the database. The speed of the User’s connection will depend on how fast they approach the site.

The Admin will run the users requests and will have access to the database.

- Reliability- The system is reliable i.e. it's well trusted.
- Security- The system is well secured, i.e. admin and user both have their unique ID and password to login into the system.
- Maintainability- Our project is well maintained with all required features involved.
- Portability- The system is portable i.e. it can work on any other system of the admin. The only requirement is the Internet connection.

Use Case Diagram

Use cases are used during the analysis phase of a project to identify system functionality.

They separate the system into actors and use cases.

Actors represent roles that are played by users of the system.

Users may be humans, other computers, or even other software systems.

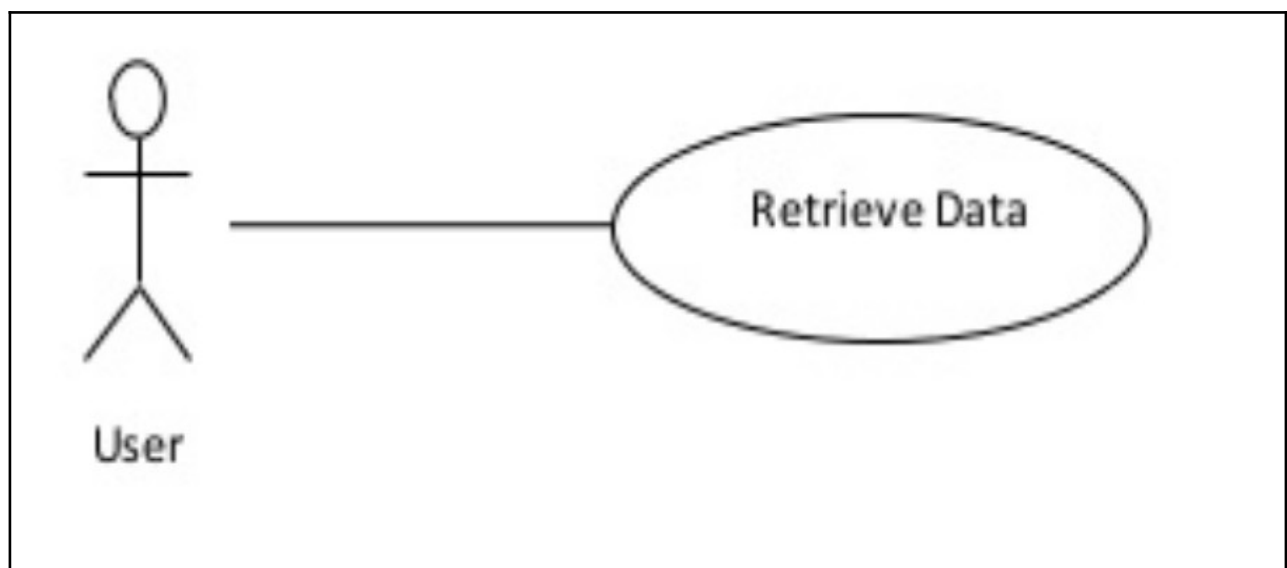
User Use Case

Use Case: Upload Data

BriefDescription:

The user approaches the site and retrieves data of his choice.

Diagram:

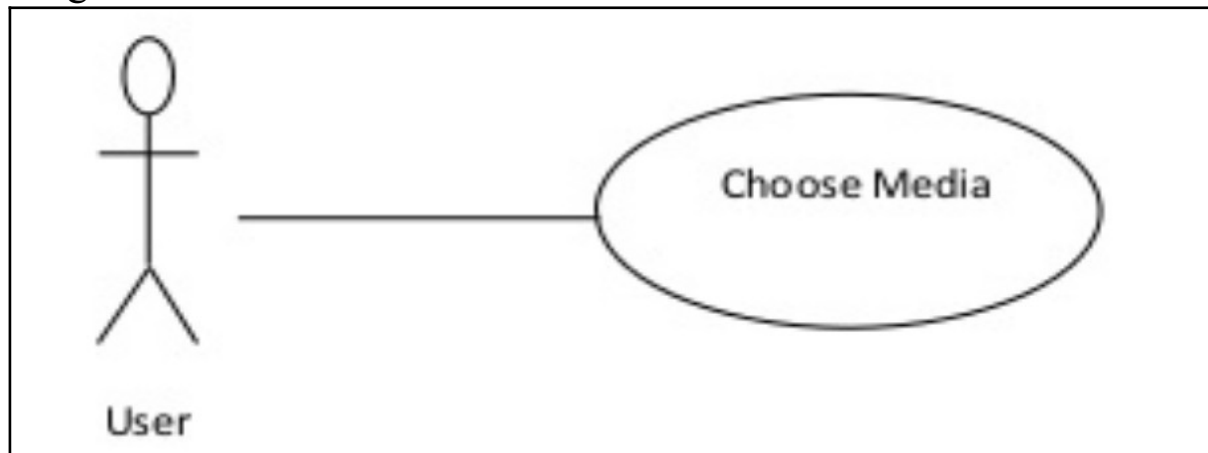


Use Case: Choose Media

BriefDescription:

The customer submits his chosen data to the database for the implementation purpose.

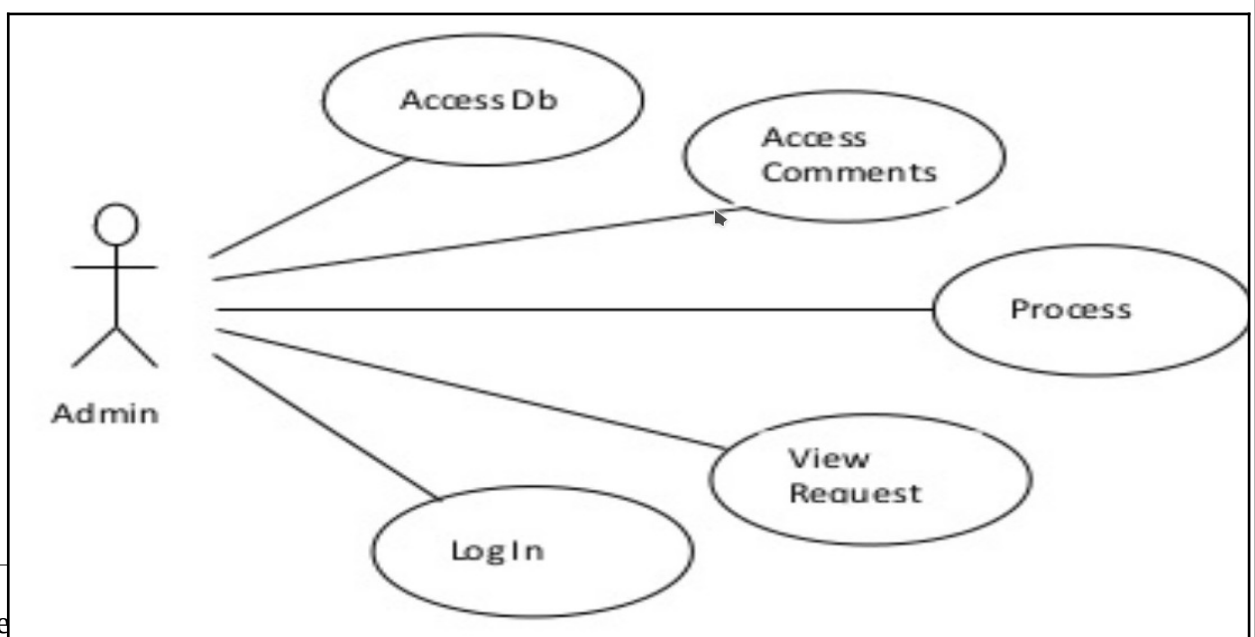
Diagram:



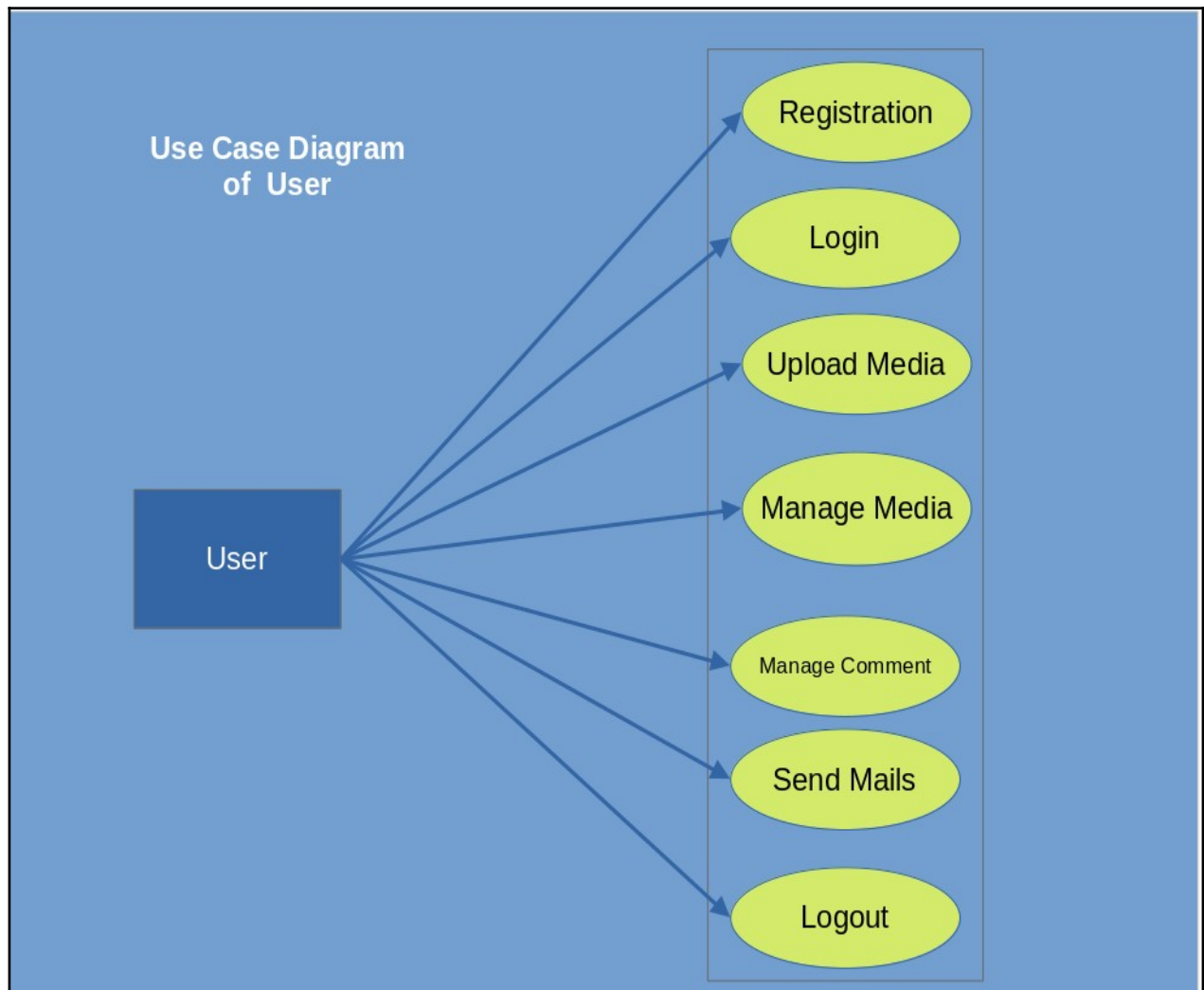
Admin Use Case:

The admin has the following set of use cases:

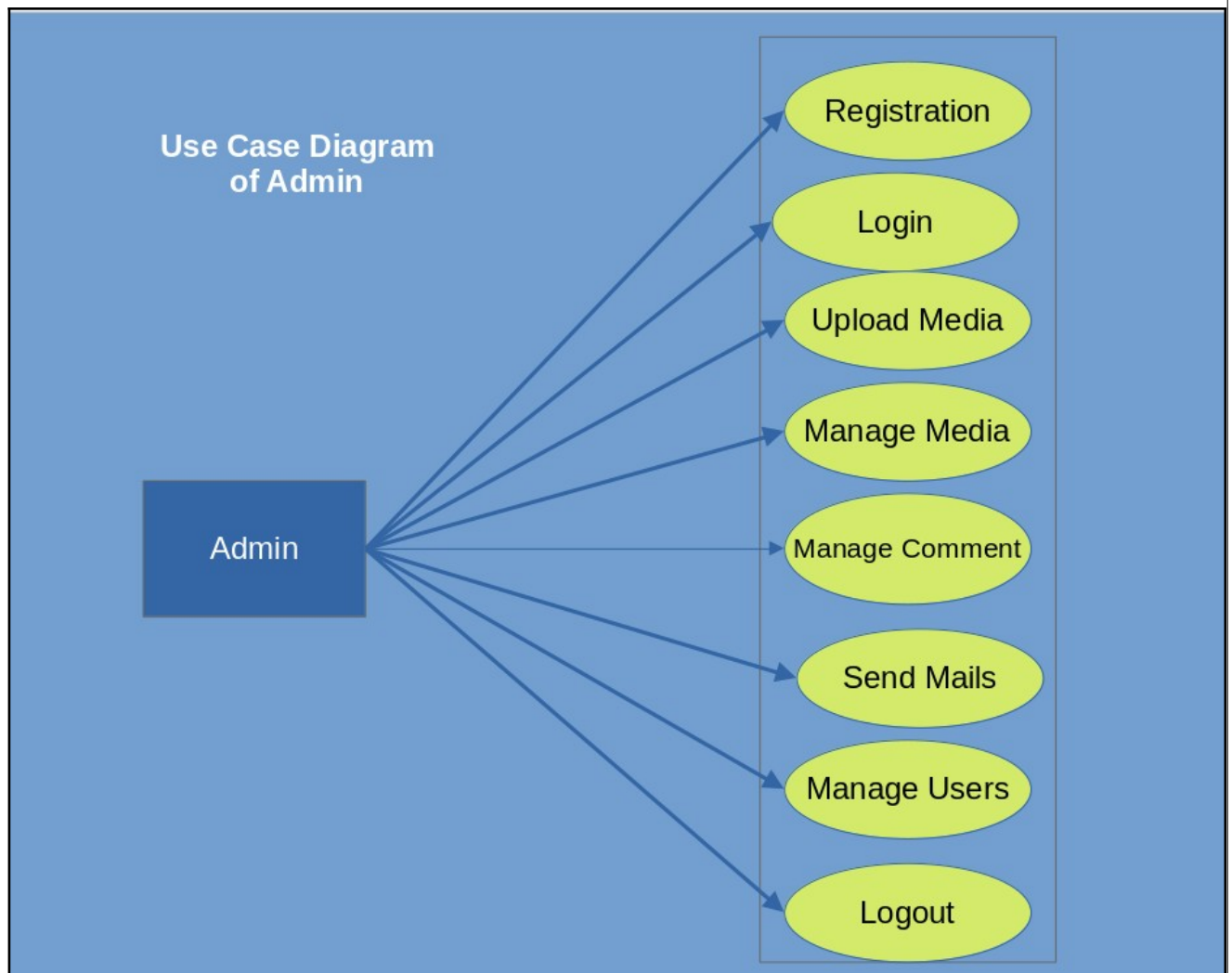
- Log In
- View Requests
- Process Requests
- Access Database



Seme



Use Case Diagram of User



Use Case Diagram of Admin

Technologies used

Django

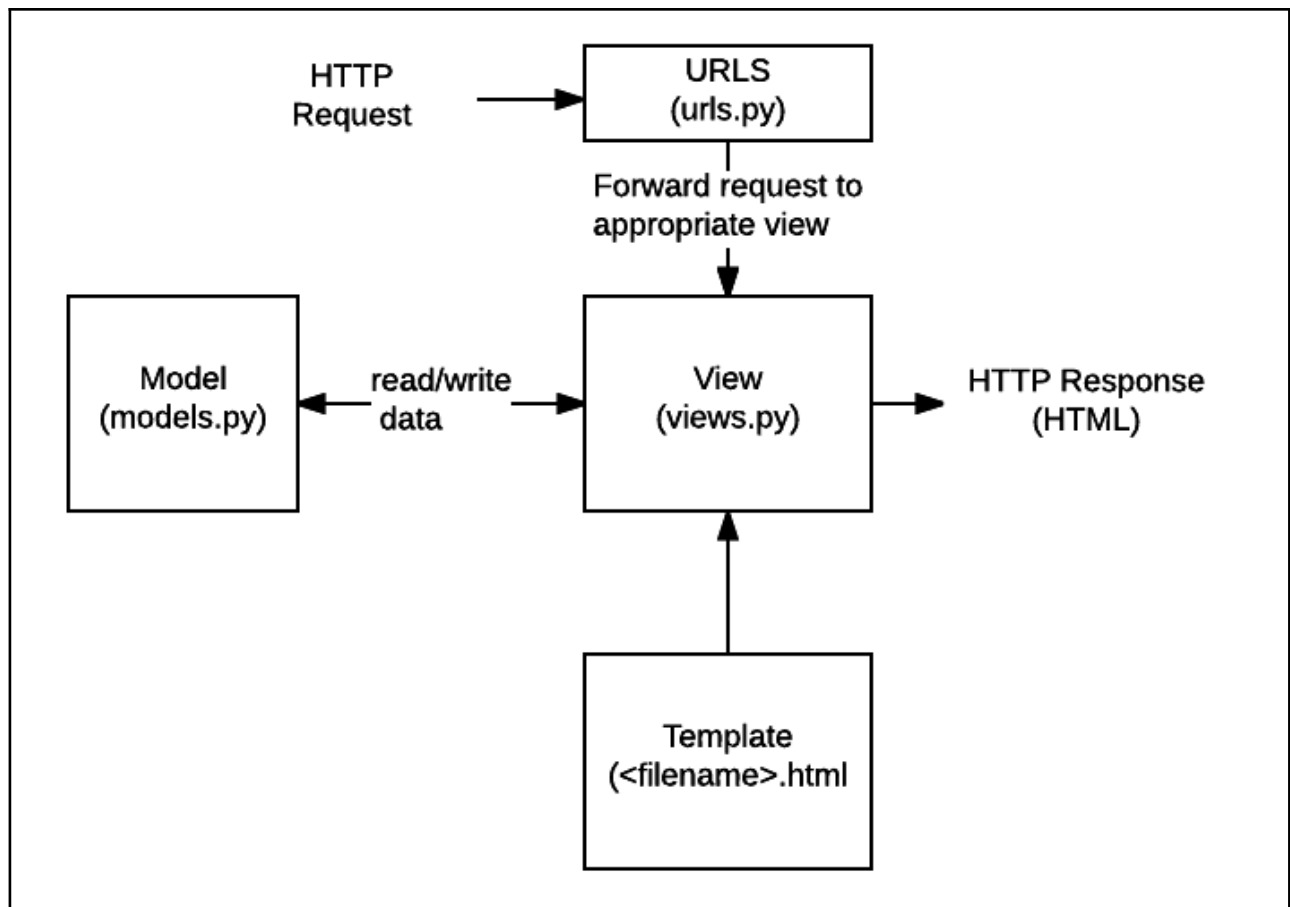
Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support.

What is the scope of Django ?

There isn't any readily-available and definitive measurement of popularity of server-side frameworks (although you can estimate popularity using mechanisms like counting the number of GitHub projects and StackOverflow questions for each platform). A better question is whether Django is "popular enough" to avoid the problems of unpopular platforms. Is it continuing to evolve? Can you get help if you need it? Is there an opportunity for you to get paid work if you learn Django?

Based on the number of high profile sites that use Django, the number of people contributing to the codebase, and the number of people providing both free and paid support, then yes, Django is a popular framework!

High-profile sites that use Django include: Disqus, Instagram, Knight Foundation, MacArthur Foundation, Mozilla, National Geographic, Open Knowledge Foundation, Pinterest, and OpenStack



Working of Django framework

Advantages of Django :

Complete

Django follows the "Batteries included" philosophy and provides almost everything developers might want to do "out of the box". Because everything you need is part of the one "product", it all works seamlessly together, follows consistent design principles, and has extensive and up-to-date documentation.

Versatile

Django can be (and has been) used to build almost any type of website

— from content management systems and wikis, through to social networks and news sites. It can work with any client-side framework, and can deliver content in almost any format (including HTML, RSS feeds, JSON, XML, etc). The site you are currently reading is built with Django!

Internally, while it provides choices for almost any functionality you might want (e.g. several popular databases, templating engines, etc.), it can also be extended to use other components if needed.

Secure

Django helps developers avoid many common security mistakes by providing a framework that has been engineered to "do the right things" to protect the website automatically. For example, Django provides a secure way to manage user accounts and passwords, avoiding common mistakes like putting session information in cookies where it is vulnerable (instead cookies just contain a key, and the actual data is stored in the database) or directly storing passwords rather than a password hash.

A password hash is a fixed-length value created by sending the password through a cryptographic hash function. Django can check if an entered password is correct by running it through the hash function and comparing the output to the stored hash value. However due to the "one-way" nature of the function, even if a stored hash value is compromised it is hard for an attacker to work out the original password.

Django enables protection against many vulnerabilities by default, including SQL injection, cross-site scripting, cross-site request forgery

and clickjacking (see Website security for more details of such attacks).

Scalable

Django uses a component-based “shared-nothing” architecture (each part of the architecture is independent of the others, and can hence be replaced or changed if needed). Having a clear separation between the different parts means that it can scale for increased traffic by adding hardware at any level: caching servers, database servers, or application servers. Some of the busiest sites have successfully scaled Django to meet their demands (e.g. Instagram and Disqus, to name just two).

Maintainable

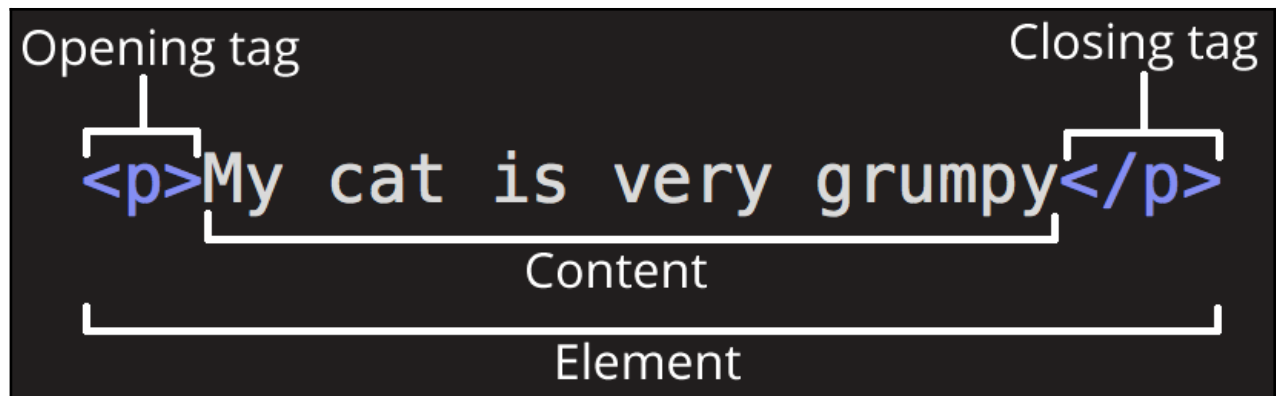
Django code is written using design principles and patterns that encourage the creation of maintainable and reusable code. In particular, it makes use of the Don't Repeat Yourself (DRY) principle so there is no unnecessary duplication, reducing the amount of code. Django also promotes the grouping of related functionality into reusable "applications" and, at a lower level, groups related code into modules (along the lines of the Model View Controller (MVC) pattern).

Portable

Django is written in Python, which runs on many platforms. That means that you are not tied to any particular server platform, and can run your applications on many flavours of Linux, Windows, and Mac OS X. Furthermore, Django is well-supported by many web hosting providers, who often provide specific infrastructure and documentation for hosting Django sites.

HTML

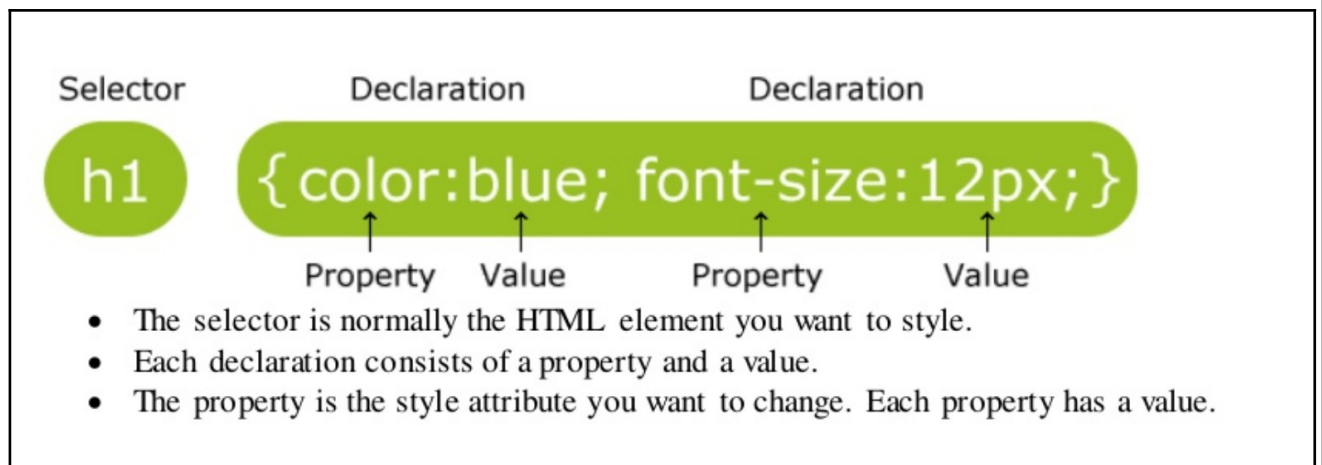
HTML is a *markup language* that defines the structure of your content. HTML consists of a series of elements, which you use to enclose, or wrap, different parts of the content to make it appear a certain way, or act a certain way. The enclosing tags can make a word or image hyperlink to somewhere else, can italicize words, can make the font bigger or smaller, and so on. For example, take the following line of content:



Anatomy of an HTML element

CSS

CSS (Cascading Style Sheets) allows you to create great-looking web pages, but how does it work under the hood? This article explains what CSS is, with a simple syntax example, and also covers some key terms about the language.



Bootstrap

Bootstrap is a free and open source front end development framework for the creation of websites and web apps. The Bootstrap framework is built on HTML, CSS, and JavaScript (JS) to facilitate the development of responsive, mobile-first sites and apps.

Responsive design makes it possible for a web page or app to detect the visitor's screen size and orientation and automatically adapt the display

accordingly; the mobile first approach assumes that smartphones, tablets and task-specific Mobile apps are employees' primary tools for getting work done and addresses the requirements of those technologies in design.

Bootstrap includes user interface components, layouts and JS tools along with the framework for implementation. The software is available precompiled or as source code.

Mark Otto and Jacob Thornton developed Bootstrap at Twitter as a means of improving the consistency of tools used on the site and reducing maintenance. The software was formerly known as Twitter Blueprint and is sometimes referred to as Twitter Bootstrap.

In computers, the word bootstrap means to boot: to load a program into a computer using a much smaller initial program to load in the desired program (which is usually an operating system).

In the physical world, a bootstrap is a small strap or loop at the back of a leather boot that enables you to pull the entire boot on and in general usage, bootstrapping is the leveraging of a small initial effort into something larger and more significant. There is also a common expression, "pulling yourself up by your own bootstraps," meaning to leverage yourself to success from a small beginning.

Software Requirements

- ❖ Python

- python interpreter 3.9.*

- ❖ Web Server

- Django Default Server

- ❖ Web Scripting Language

- ❖ Framework

- Django 2.2.*

- ❖ Database:

- Sqlite

- ❖ Editors:

- Vscode/Vim

- ❖ Supported Operating Systems

- Windows

- 2000/XP/2003/Vista/ Windows 7/ Windows 8/ XP64 / Vista64 / Windows 10

- Linux

- Arch Based distributions (Manjaro) /Ubuntu Based Distributions(Mint)

Hardware Requirements

❖ Minimum Hardware Configurations

- Processor: 800MHz Intel Pentium III or equivalent
- Memory: 512 MB
- Disk space: 750 MB of free disk space
- Screen resolution is 1024x768 pixels.

❖ Recommended Hardware Configurations

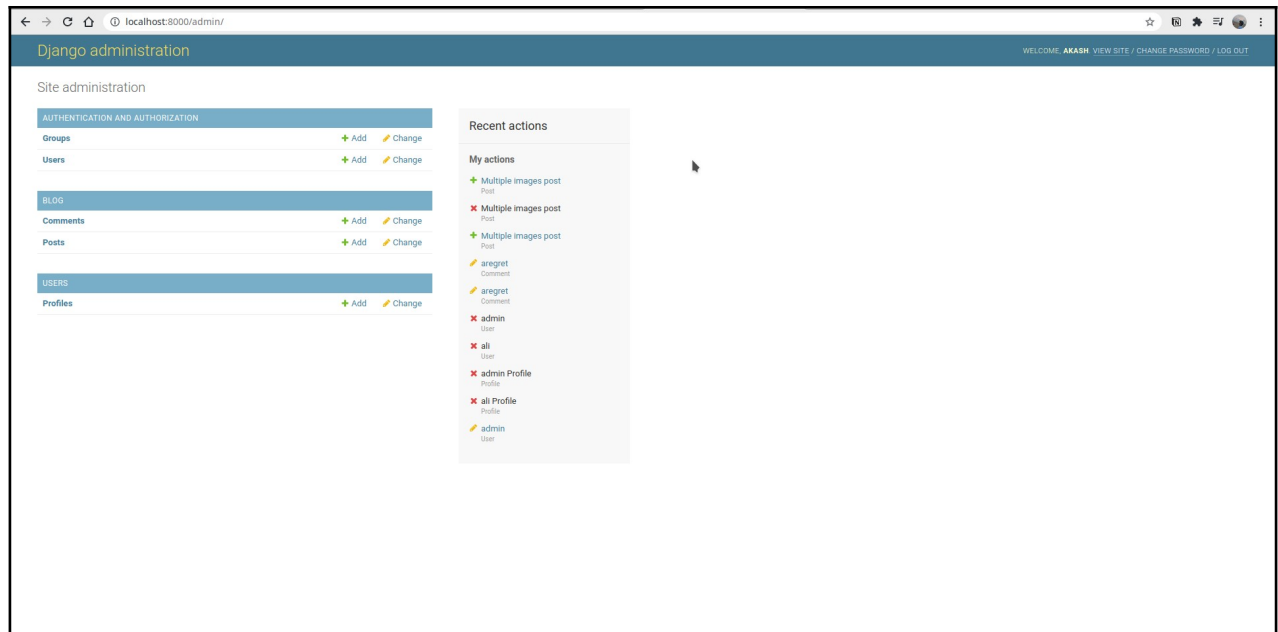
- Processor: Intel Core i5 or equivalent
- Memory: 2 GB (32-bit), 4 GB (64-bit)
- Disk space: 1.5 GB of free disk space
- Screen resolution is 1024x768 pixels

Chapter 6

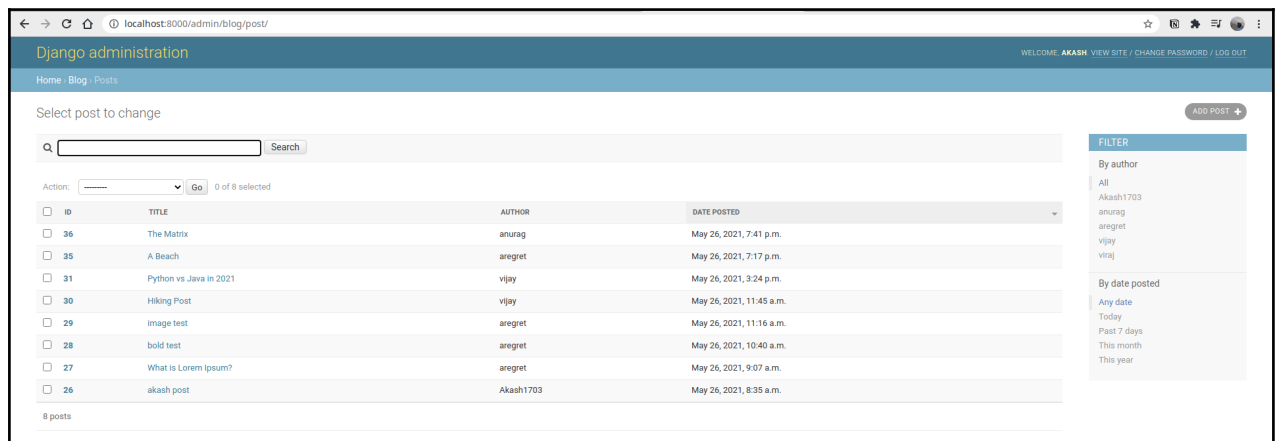
Results and

Applications

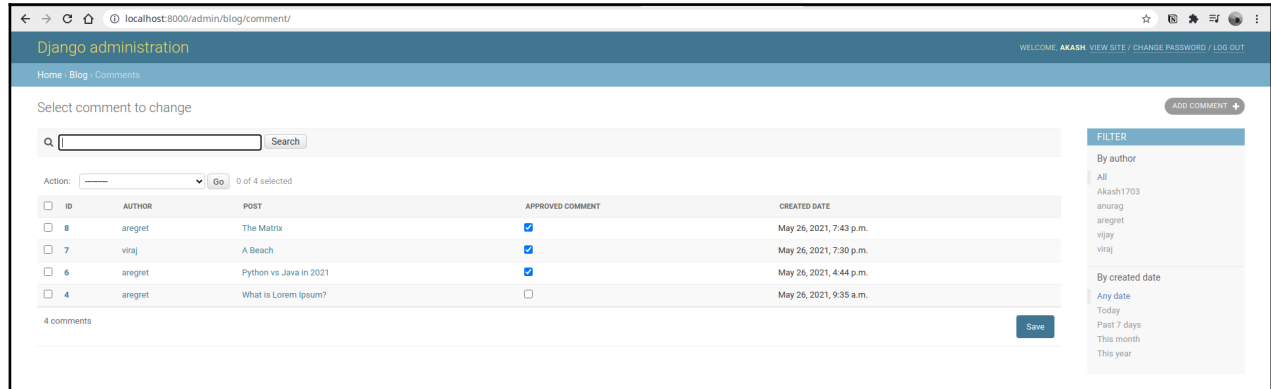
Outputs



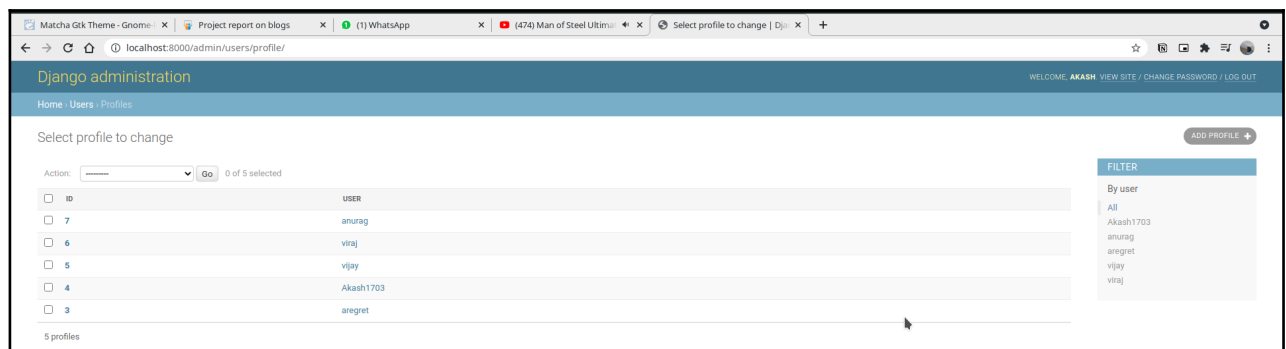
ADMIN PAGE



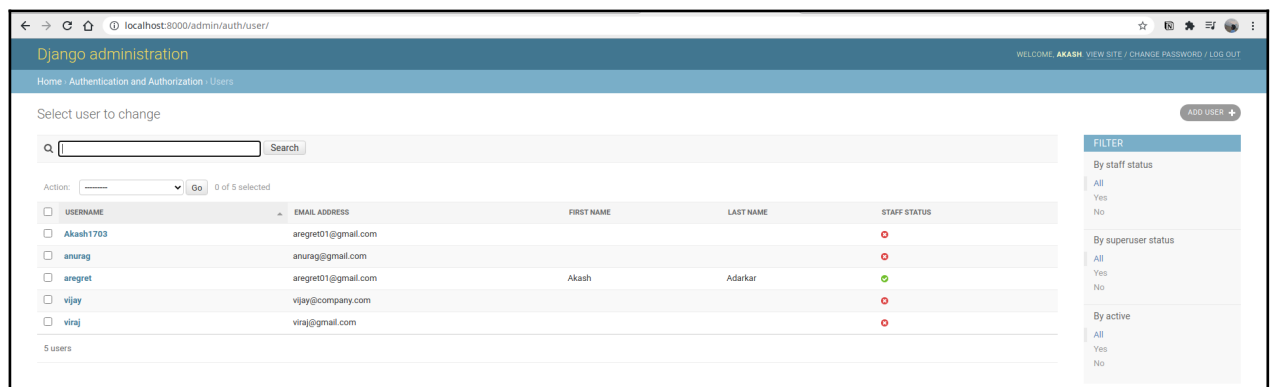
Post Table



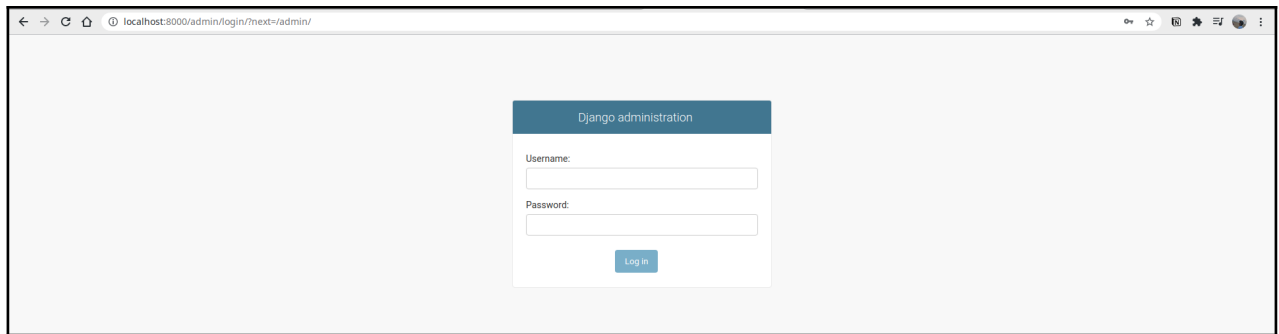
Comment Table



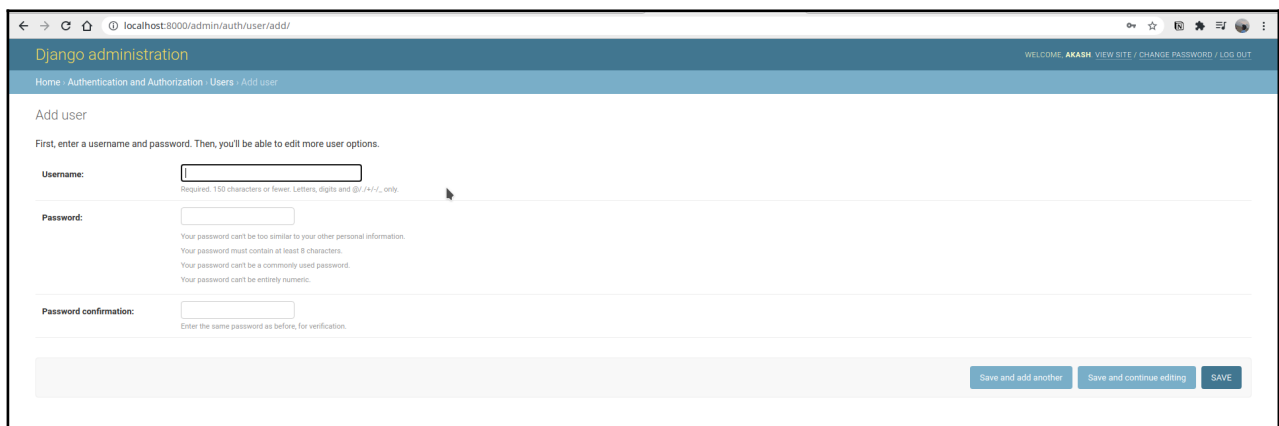
Profile Table



User Table



Admin Login



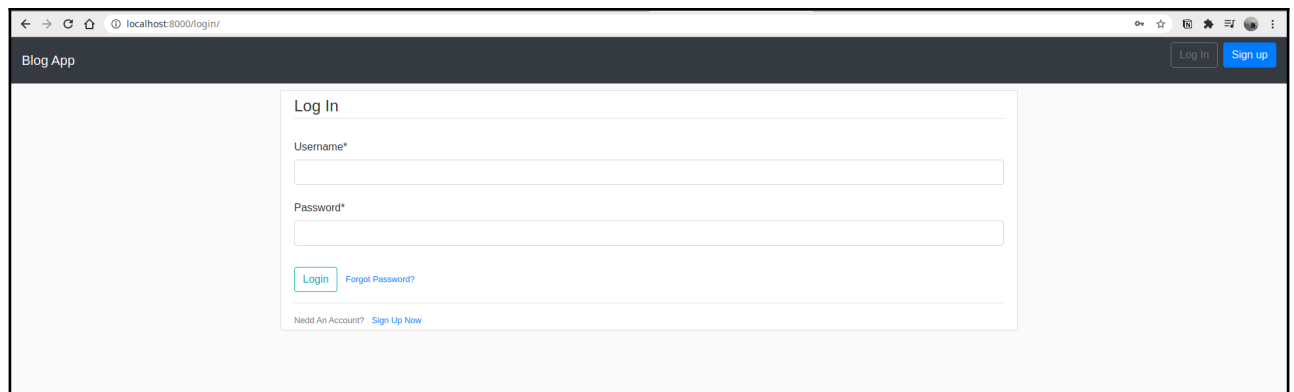
Add User from Admin Page

The screenshot shows the 'Add post' form in the Django administration interface. The form includes fields for 'Author' (a dropdown menu), 'Title' (a text input), and 'Content' (a rich text editor with various formatting options). Below the content field is a 'Like' button. The 'Date posted' section has 'Date' and 'Time' dropdowns, both set to 'Today' and '14:07:45' respectively. A 'Header image' field with a 'Choose file' button is also present. At the bottom, there is a 'COMMENTS' section with a table header: 'AUTHOR', 'TEXT', 'CREATED DATE', 'APPROVED COMMENT', and 'DELETE?'. The table currently has one row with a dropdown for 'AUTHOR', a text input for 'TEXT', and a 'Date' dropdown set to 'Today' and a 'Time' dropdown set to '14:07:45'. The 'APPROVED COMMENT' column has a checked checkbox, and the 'DELETE?' column has a link icon.

Add Posts from Admin Page

The screenshot shows the 'Add comment' form in the Django administration interface. The form includes fields for 'Post' (a dropdown menu), 'Author' (a dropdown menu), and 'Text' (a text input). The 'Created date' section has 'Date' and 'Time' dropdowns, both set to 'Today' and '14:07:24' respectively. A checkbox labeled 'Approved comment' is checked. At the bottom right, there are three buttons: 'Save and add another', 'Save and continue editing', and 'SAVE'.

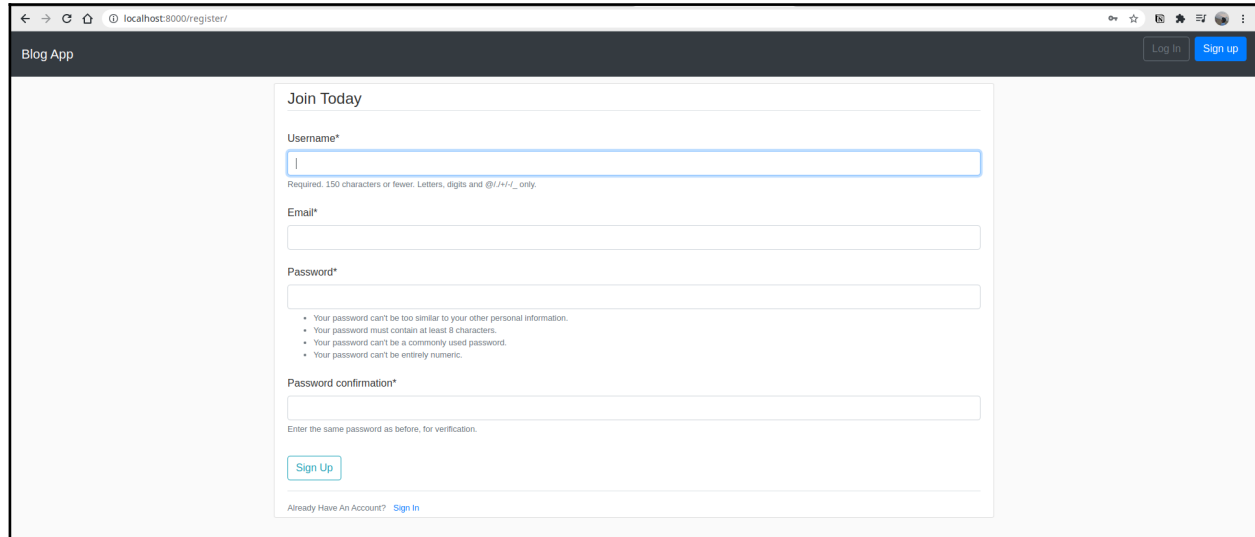
Add Comments from Admin Page



The screenshot displays a web browser window with the address bar showing 'localhost:8000/login/'. The page title is 'Blog App'. In the top right corner, there are two buttons: 'Log in' and 'Sign up'. The main content area features a 'Log In' form with the following elements:

- Log In** title
- Username*** label above a text input field.
- Password*** label above a password input field.
- A **Login** button and a [Forgot Password?](#) link below the password field.
- A link [Need An Account? Sign Up Now](#) at the bottom of the form.

Login Page

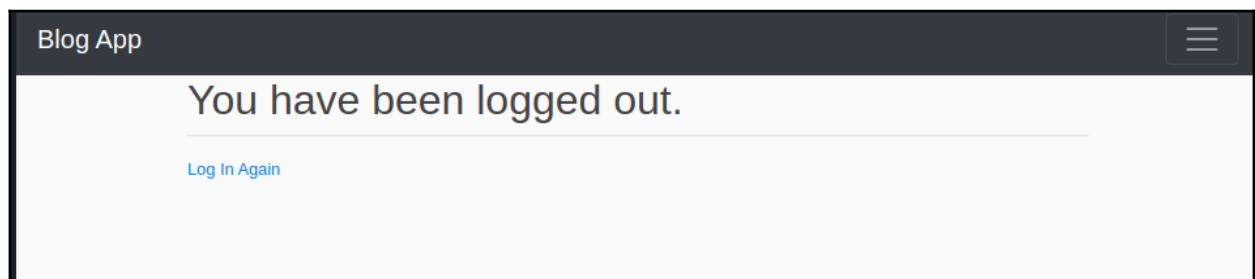


The screenshot shows a web browser window with the address bar displaying 'localhost:8000/register/'. The page title is 'Blog App'. In the top right corner, there are 'Log In' and 'Sign up' buttons. The main content area is titled 'Join Today' and contains a registration form with the following fields and instructions:

- Username***: A text input field with a placeholder character 'i'. Below it, a note states: 'Required: 150 characters or fewer. Letters, digits and @/./+/-/_ only.'
- Email***: A text input field.
- Password***: A text input field. Below it, a list of password requirements is shown:
 - Your password can't be too similar to your other personal information.
 - Your password must contain at least 8 characters.
 - Your password can't be a commonly used password.
 - Your password can't be entirely numeric.
- Password confirmation***: A text input field. Below it, a note states: 'Enter the same password as before, for verification.'

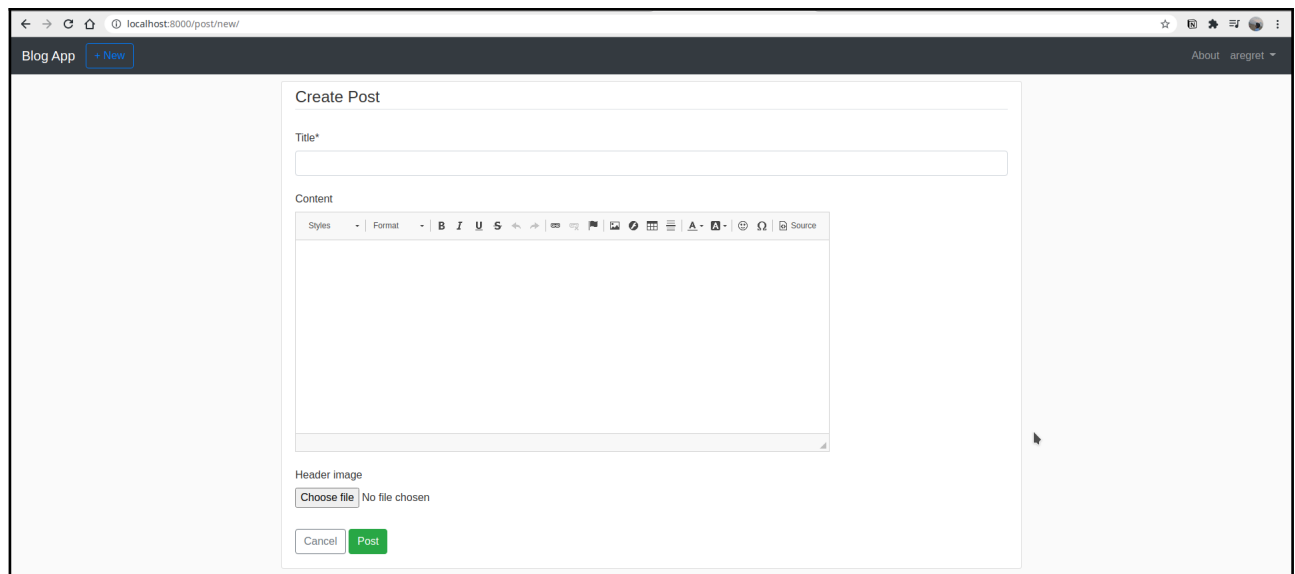
At the bottom of the form is a 'Sign Up' button. Below the button, a link reads: 'Already Have An Account? [Sign In](#)'.

Register Page



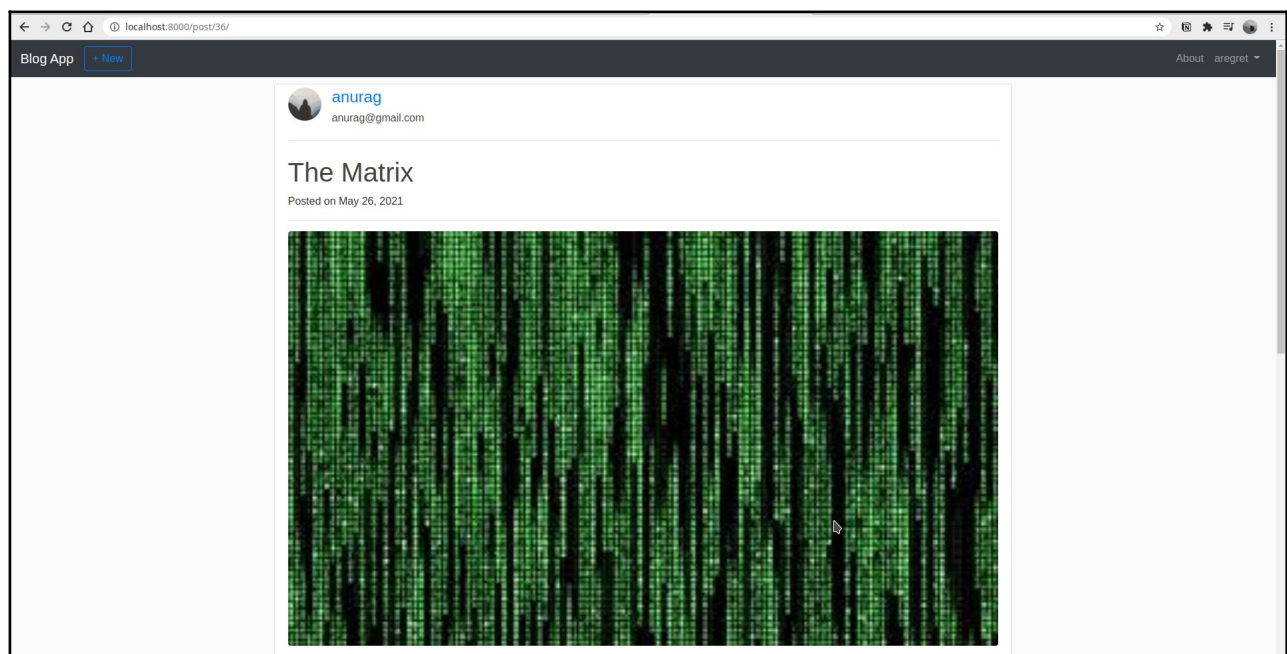
The screenshot shows a web browser window with the address bar displaying 'localhost:8000/logout/'. The page title is 'Blog App'. In the top right corner, there is a hamburger menu icon. The main content area displays the message 'You have been logged out.' in a large, bold font. Below this message is a link that reads 'Log In Again'.

Logout Page

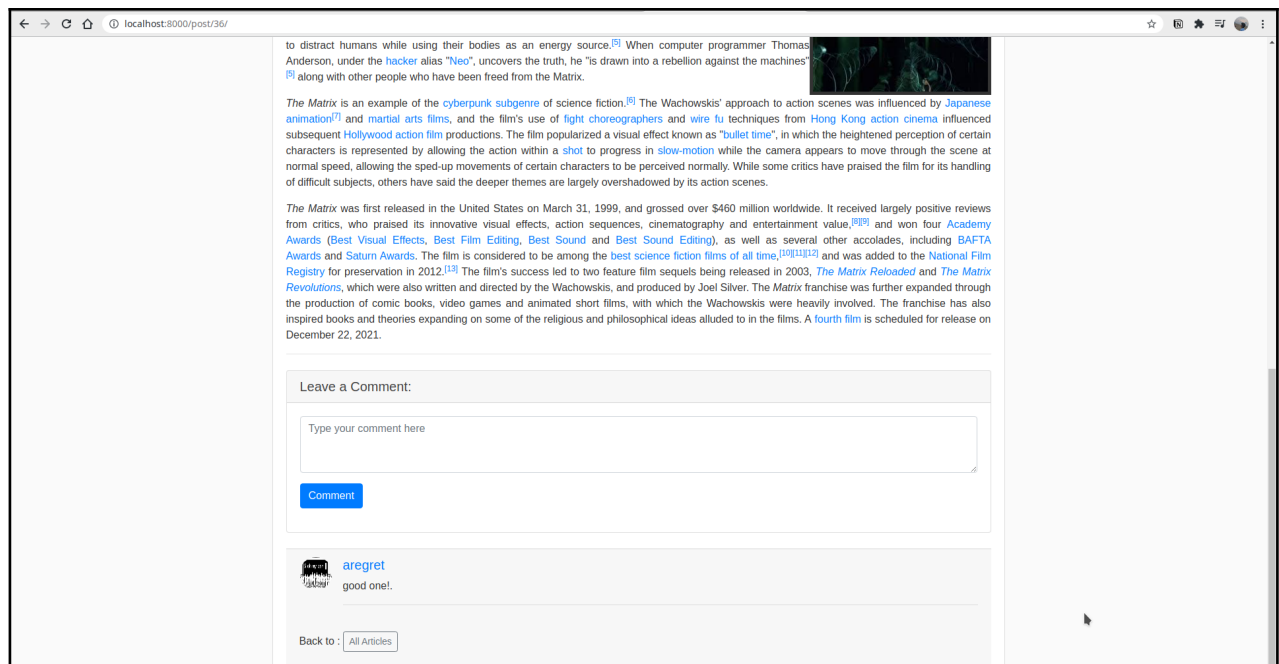


The screenshot shows a web browser at the URL `localhost:8000/post/new/`. The page has a dark header with 'Blog App' and a '+ New' button. The main content area is titled 'Create Post' and contains a 'Title*' input field, a 'Content' area with a rich text editor (including bold, italic, underline, and link buttons), and a 'Header image' section with a 'Choose file' button and the text 'No file chosen'. At the bottom are 'Cancel' and 'Post' buttons.

Create Post Page



Detailed Post Page(1)




Detailed Post Page(2)

Leave a Comment:

Type your comment here

Comment


Comments

 **aregret**
good one!.

Comments Section

Blog App [+ New](#)

About
aregret ▾

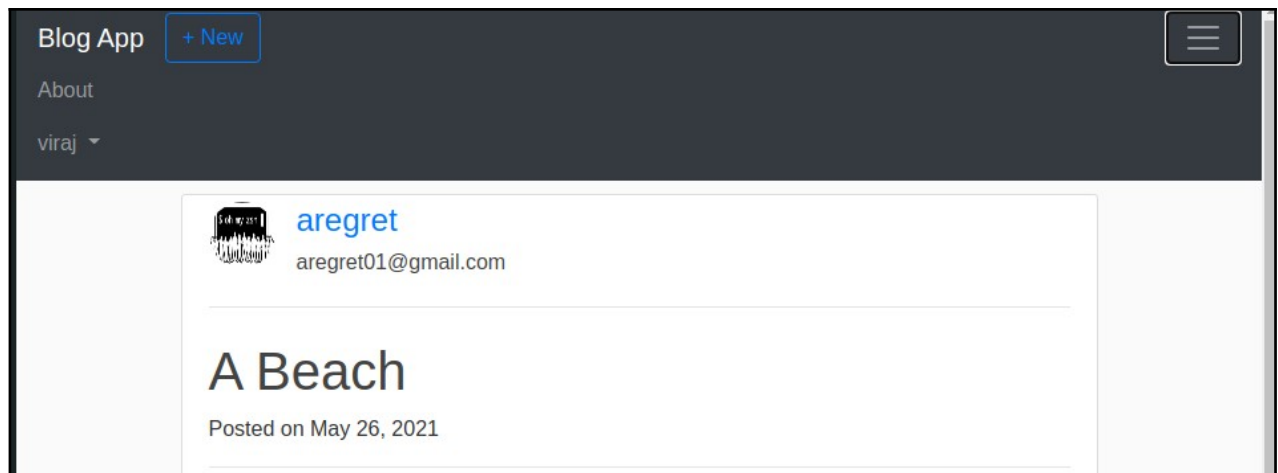
 **aregret**
aregret01@gmail.com

A Beach

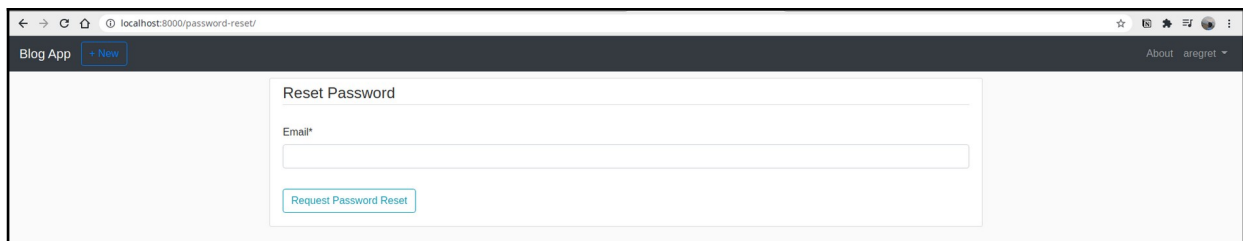
Posted on May 26, 2021

Delete Update

Authorized user/Author can update/delete post

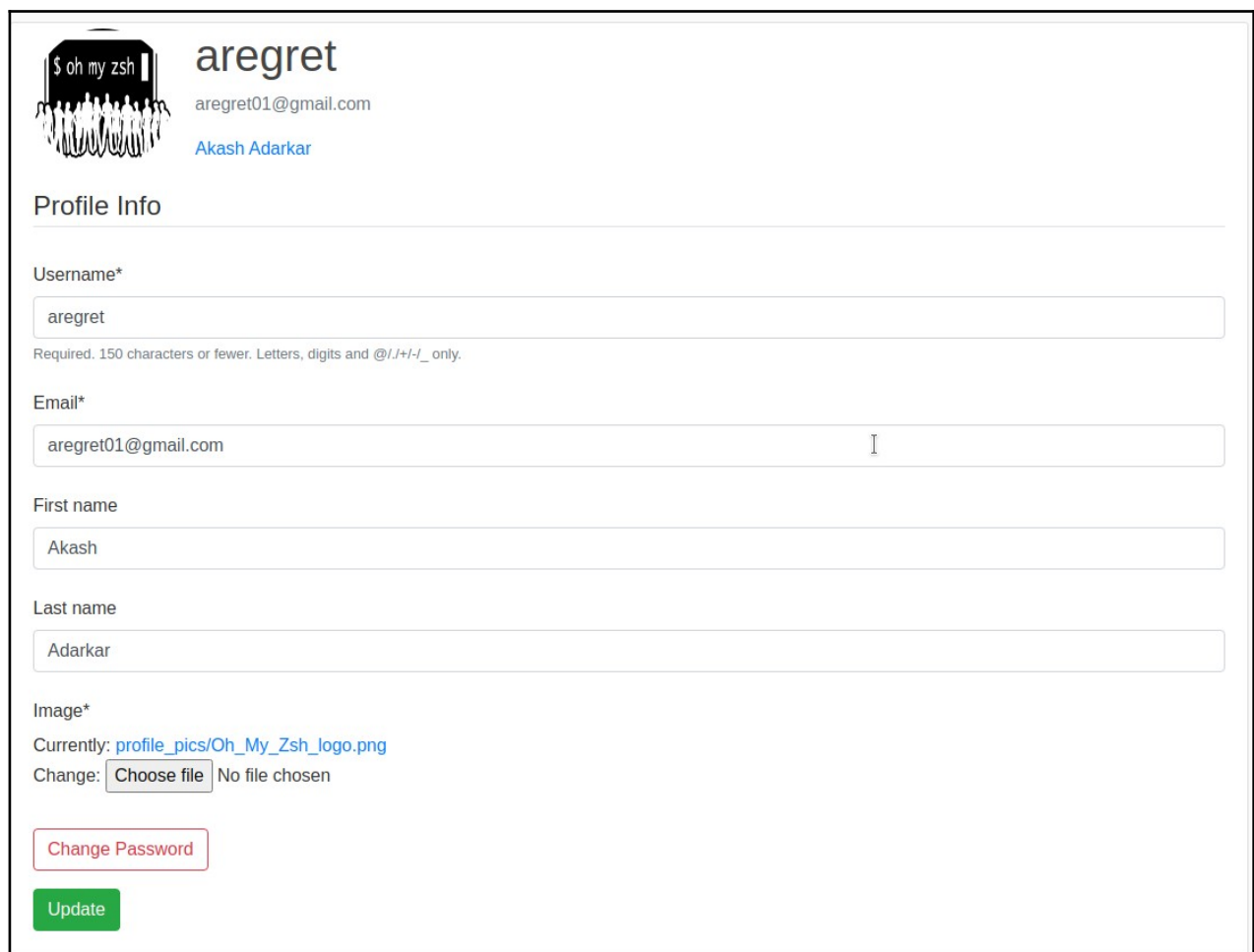


viewer can't update/delete post



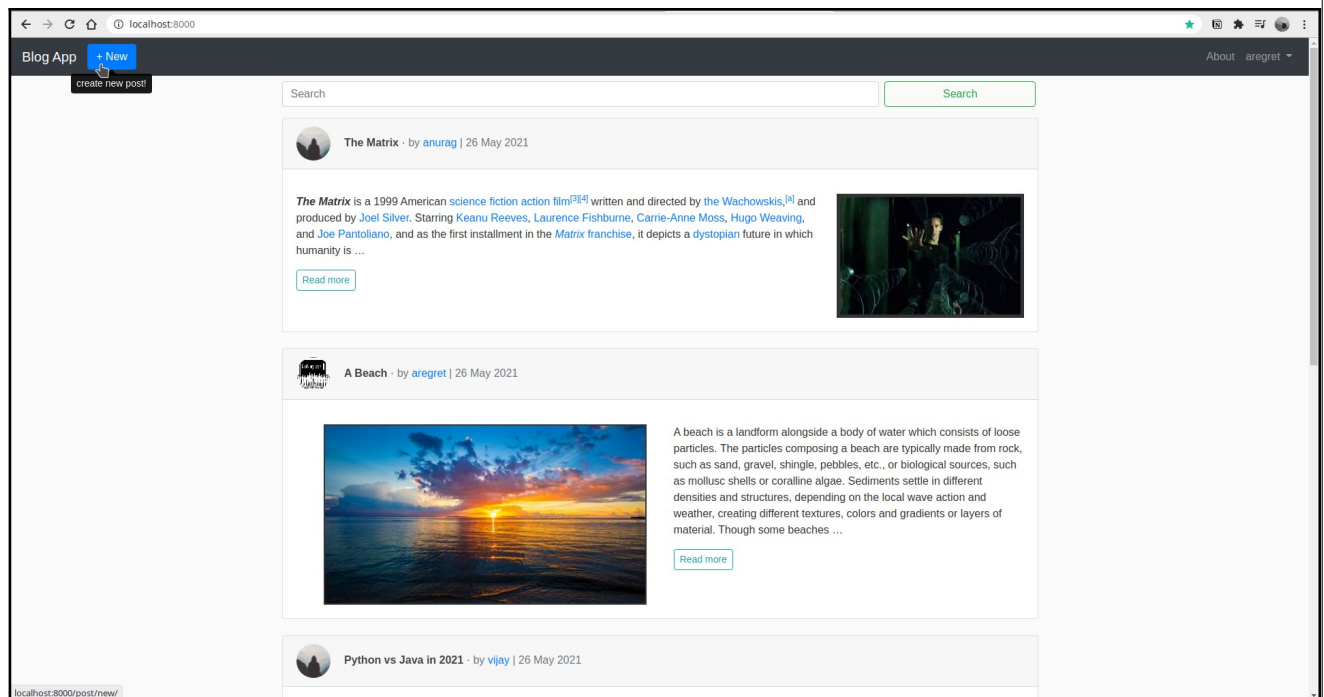
A screenshot of a web browser showing a password reset page. The browser's address bar displays 'localhost:8000/password-reset/'. The page has a dark header with 'Blog App' and a '+ New' button on the left, and 'About | aregret' on the right. The main content area is titled 'Reset Password' and contains an 'Email*' input field and a 'Request Password Reset' button.

Password Reset Page

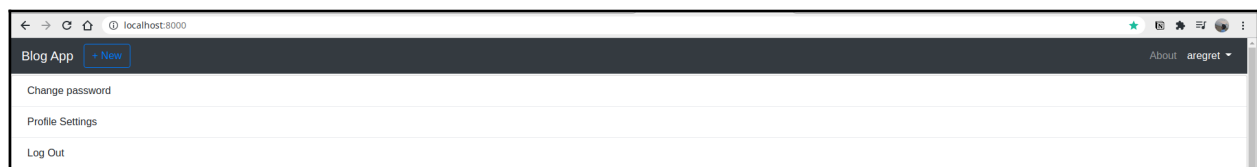


A screenshot of a user profile update page for 'aregret'. The profile header shows a logo with the text '\$ oh my zsh', the username 'aregret', email 'aregret01@gmail.com', and name 'Akash Adarkar'. Below this is a 'Profile Info' section with form fields for 'Username*' (containing 'aregret'), 'Email*' (containing 'aregret01@gmail.com'), 'First name' (containing 'Akash'), and 'Last name' (containing 'Adarkar'). A note below the username field states: 'Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.' The 'Image*' section shows the current profile picture as 'profile_pics/Oh_My_Zsh_logo.png' and includes a 'Choose file' button. At the bottom are two buttons: 'Change Password' and 'Update'.

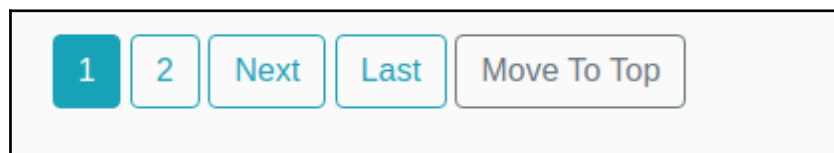
Profile Update Page



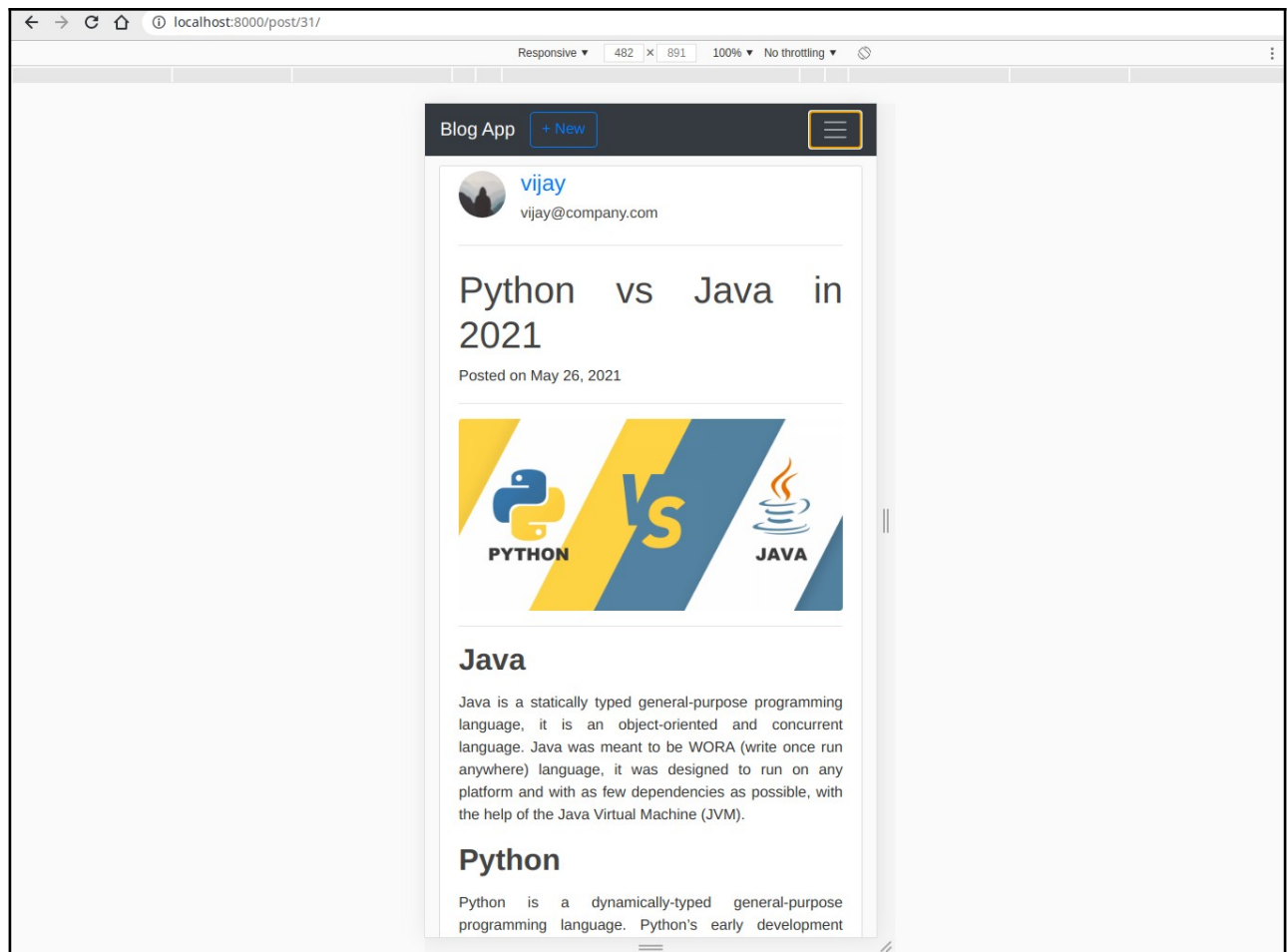
Post List Page



Navigation Bar



Pagination



Responsive Site

Password Reset Process:



Navigation Bar

Reset Password

Email*

Request Password Reset



Enter the appropriate email address:

An email has been sent with instructions to reset your password

Message of successful delivery of instructions

```
-----
Content-Type: text/plain; charset="utf-8"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Subject: Password reset on localhost:8000
From: webmaster@localhost
To: aregret01@gmail.com
Date: Wed, 02 Jun 2021 14:01:06 -0000
Message-ID: <162264246604.15594.17933332490975455517@aregret01>
```

You're receiving this email because you requested a password reset for your user account at localhost:8000.

Please go to the following page and choose a new password:

<http://localhost:8000/password-reset-confirm/NA/5r5-6e1cb466818e3cd20edb/>

Your username, in case you've forgotten: Akash1703

Thanks for using our site!

The localhost:8000 team

Copy the link .



<http://localhost:8000/password-reset-confirm/NA/5r5-6e1cb466818e3cd20edb/>

paste the link in the browser address bar

Reset Password

New password*

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

New password confirmation*

Reset Password



enter new password

Your password has been set successfully

[Sign In Here](#)

message displaying successful reset of the password.

Chapter 7.

Conclusions and

Future scope

Conclusion

Our project is only humble venture to satisfy the needs to manage their project work .Several user friendly coding have also adopted .this package shall prove to be a powerful package is satisfying all the requirements of the school .The objective of software planning is to provide a framework that enables the user to make reasonable estimates made within limited time frame at beginning of the software project should be updated regularly project progresses

At the end it is concluded that we have made effort on following points

A description of the background and context of the project and its relation to work already done in the area.

Made a statement of the aims and objectives of the project.

The description of Purpose ,Scope and applicability we define the problem on which we are working in the project

We describe the requirement Specifications of the system and the actions that can be done on these things

We understand the problem domain and produce a model of the system, which describes operations that can be performed on the system.

We included features and operations in details including screen layouts

We designed user interface and security issue related to system

Finally the system is implemented and tested according to test cases

Future Scope

In a nutshell ,it can be summarized that the future scope of the project circles around maintaining information regarding

- ❖ add more functionality to profile page of user
- ❖ add more models for fields in user and post
- ❖ deploy the website online so that people can access it
- ❖ make more appealing templates for the users
- ❖ add like functionality for posts
- ❖ add page to show the list of post and update and deletion options on that page
- ❖ add themes for user comfort.
- ❖ add tags
- ❖ add more media content functionality i.e videos , gifs etc
- ❖ add replies to comments
- ❖ add chat options so that users and talk to the author directly

The above mentioned points are the enhancements which can be done to increase the applicability and usage of this project .Here we can maintain the records of Blogs and Comments Also ,as it can be seen that now-a-days the players are versatile i.e so there is a scope for introducing a method to

maintain the Online Blog app Enhancements can be done to maintain all the blogs, Comments,New category ,New Blog Technology Blog.

We have left all the options open so that if there is any other future requirement in the system by the user for the enhancement of the system then it is possible to implement them.In the last we would like thanks all the persons involved in the development of the system directly or indirectly .We hope that the project will serve its purpose for which it is develop there by underlying success of process.

References and Bibliography

1. Ashish Arora Web Application Development
https://www.academia.edu/34579745/Web_Application_Development
2. Zuwati Hasim Issues and Challenges of using Web Blogs as a Medium for Research Communication
https://www.researchgate.net/publication/273493095_Issues_and_challenges_of_using_web_blogs_as_a_medium_for_research_communication
3. Karan Gupta, Anita Goel (2014) Blog in web application: a software engineering perspective <https://www.inderscience.com/info/inarticle.php?artid=64789>
4. Kedar Jain (2019) A python Book:Beginning Python,advanced Python and Python Exercises
https://www.academia.edu/41829928/A_Python_Book_Beginning_Python_Advanced_Python_and_Python_Exercises?sm=b
5. Daniel J LeBlanc (2015) Photo Blog Web Application using Twitter Bootstrap
https://www.academia.edu/12833372/Photo_Blog_Web_Application_Using_Twitter_Bootstrap_and_JQuery
6. Adamya Shyam (2020) A Django based Educational Resource sharing website
https://www.academia.edu/43266462/A_Django_Based_Educational_Resource_Sharing_Website_Shreic
7. Obak Kando Adrian Holovaty The Definitive Guide to Django: Web Development Done Right

- https://www.academia.edu/44025451/The_Definitive_Guide_to_Django_Web_Development_Done_Right_Adrian_Holovaty_Jacob_K_Moss
8. V. Myalapalli (2015) Optimizing Front -end Applications and Javascript
https://www.academia.edu/13009088/Optimizing_Front_End_Applications_and_JavaScript
 9. Tomayess B.T Issa(2013 Social Networking for Web -Based Communities
https://www.researchgate.net/publication/262401330_Social_networking_for_web-based_communities
 10. Matthew N.O Sadiku, Adedamola A.Omotoso, Sarhan M Musa (2019)
Social Networking
https://www.researchgate.net/publication/333695257_Social_Networking