

I offer myself at Thy Lotus Feet.

# A C C U J O U R N A L

A Project / Dissertation as a Course requirement for BSc. (Hons)  
Computer Science

Sumiran Rai

204218



SRI SATHYA SAI INSTITUTE OF HIGHER LEARNING  
(Deemed to be university)

Department of Mathematics and Computer Science  
Muddenahalli Campus

April 2023

# Department Letter Head

## CERTIFICATE

This is to certify that this Project / Dissertation titled ACCUJOURNAL submitted by Sumiran Rai, 204218, Department of Mathematics and Computer Science, Muddenahalli Campus is a bonafide record of the original work under my/our supervision as a Course requirement for the Degree of BSc. (Hons.) Computer Science.

-----

Dr. K. Vengata Krishnan  
Project / Dissertation Supervisor

Place - Muddenahalli  
Date \_ 27 April, 2023

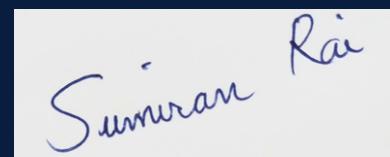
Countersigned by

-----

P. Sunil Kumar  
Head of the Department

# DECLARATION

The Project / Dissertation titled ACCUJOURNAL was carried out by me under the supervision of Dr. K. Vengata Krishnan, Department of Mathematics and Computer Science, Muddenahalli Campus as a Course requirement for the Degree of BSc. (Hons.) Computer Science and has not formed the basis for the award of any degree, diploma or any other such title by this or any other University.

A white rectangular box containing a handwritten signature in blue ink. The signature reads "Sumiran Rai" with a small dot above the "i".

Sumiran Rai

Place - Muddenahalli

204218

Date - 27 April, 2023

III BSc.



Sri Sathya Sai Institute of Higher Learning

(Deemed to be University)

Muddenahalli Campus

## A PROJECT REPORT ON

# ACCUJOURNAL

*(Search engine with a database for indexed journals)*

---

Mentor :

Dr. K. Vengata Krishnan

Students :

Sumiran Rai (204218)

&

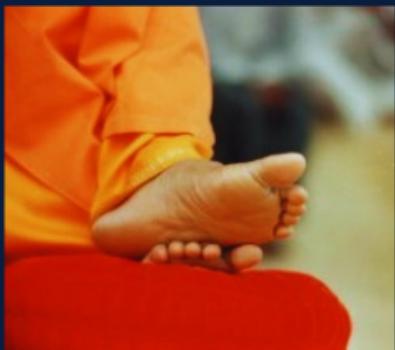
B. Chiruhas (204201)



S A I R A M

# ACKNOWLEDGEMENT

---



My heartfelt gratitude to Bhagawan Sri Sathya Sai Baba.

I would like to pay our gratitude to Dr. K Vengata Krishnan Sir for his relentless support, ideas, feedback, and acting as a mentor throughout this project.

I extend our thanks to Director Sri B Venkatramana Sir and Assistant Professor Sri P Sunil Kumar Sir for providing a systematic schedule for this project with adequate resources.

I owe my thanks to Brother B.Chiruhas, who have helped me throughout this porject.



**Sri Sathya Sai Institute of Higher Learning**  
**(Deemed to be University)**

**Muddenahalli Campus**

---

**C E R T I F I C A T E**

---

This is to certify that this Project titled "ACCUJOURNAL" submitted by Sumiran Rai (Regd. No. - 204218) , Department of Mathematics and Computer Science, Muddenahalli Campus is a Bonafide record of the work done under my supervision as a Course requirement for the Degree of Bachelor of Science (Hons) in Computer Science.

---

**SRI VENGATA KRISHNAN**  
**PROJECT SUPERVISOR**



# Sri Sathya Sai Institute of Higher Learning

## (Deemed to be University)

### Muddenahalli Campus

#### DECLARATION

---

I, Sumiran Rai hereby declare that the project titled "ACCUJOURNAL" is a Bonafide record, carried out by me under the supervision of Sri K. Vengata Krishnan, Department of Mathematics & Computer Science, Muddenahalli Campus, as a course requirement for the Degree of Bachelor of Science (Hons) in Computer Science.

Place : Muddenahalli  
Date : 27th April, 2023

# Table of contents

---

- i. **Introduction**
- ii. **Why ACCUJOURNAL?**
- iii. **Use Cases**
- iv. **Technologies**
- v. **Database Design**
- vi. **Functionality**
- vii. **Coding Modules**
- viii. **Screen Flow**
- ix. **Test Cases**
- x. **Installation**
- xi. **References and Sources**

---

# INTRODUCTION

---

## ABSTRACT

Rapid growth of fake and predatory journals has seriously threatened the academic community. Our aim is to build a search engine that will not only result in accurate and authenticated journals but also detect fake or cloned journals, benefiting the academic community tremendously.

Therefore a search engine is designed, in a way that, it provides the results of all indexed and authenticated journals stored in the database.

---

# INTRODUCTION

---

## AIM

Our aim is to develop a trustworthy database and search engine, for all the indexed and authenticated journals for academic and research purposes, such that students as well as teachers can use this and get the information with ease. We want to prevent publications in predatory/ dubious / sub-standard journals, which reflect adversely and tarnish the image of academia.

# Why ACCOUJOURNAL

---

1. By indexing a large number of scholarly articles from multiple journals, a search engine can provide more accurate and comprehensive search results for users looking for specific information.
2. Researchers and other professionals can use the search engine to quickly and easily locate relevant articles from a wide range of academic journals, allowing them to stay up-to-date with the latest research in their fields.
3. By indexing articles from multiple journals, the search engine can help to increase the visibility of research that may otherwise be difficult to find or access.
4. Rather than searching through multiple journals individually, users can save time and effort by using the search engine to quickly locate relevant articles.

Overall, developing a search engine that utilizes a database of indexed journals can be a powerful tool for researchers, librarians, and others who rely on scholarly literature as a key source of information.

---

# USE CASES

---

- Researchers in various fields of study can use the search engine to quickly locate relevant scholarly articles and stay up-to-date with the latest research in their fields.
  - Students, researchers, and professionals can use the search engine to conduct literature reviews and gather information on specific topics.
  - Publishers and editors can use the search engine to find potential authors and contributors, review existing literature on a topic, and stay up-to-date with industry trends.
  - Libraries can use the search engine to help patrons locate scholarly articles and other research materials, and to keep their collections up-to-date with the latest publications.
  - Patent attorneys and professionals can use the search engine to conduct patent searches and identify relevant prior art.
-

# TECHNOLOGIES

---

## Web Framework

Django - Django is a popular open-source web framework for building web applications using Python. Django is a back-end server side web framework. Django is free, open source and written in Python. Django makes it easier to build web pages using Python.

Venv - It is suggested to have a dedicated virtual environment for each Django project, and one way to manage a virtual environment is `venv`, which is included in Python.

Views - In Django, views are Python functions that handle HTTP requests and return HTTP responses. Views are responsible for implementing the logic of an application, querying the database if necessary and returning an appropriate response.

Urls - URLs map requests to views, and templates are used to render the response. URLs map incoming requests to specific views based on the URL pattern.

Models and Forms - Models define the structure of the data and the relationships between them, while forms handle user input and validation. Together, these components form the backbone of a Django application.

Together, views, URLs, templates, models, and forms work together to create a robust and scalable web application in Django.

## Django packages

**Djongo** - Djongo is a Django database connector for MongoDB that allows developers to use the Django with MongoDB. Djongo aims to provide a seamless experience for Django developers who want to use MongoDB as their database backend. Djongo is a useful tool for developers who want to take advantage of MongoDB's flexibility and scalability while using Django's familiar development patterns and tools.

**Whitenoise** - When building web applications, we want to add some static files like images or css files. Django does not have a built-in solution for serving static files, at least not in production when DEBUG in settings has to be False. We have to use a third-party solution to accomplish this. Whitenoise is a Python library, built for serving static files.

# TECHNOLOGIES

---

## Programming Language

**Python** - Python is a high-level, interpreted programming language that is known for its simplicity, readability, and versatility. It has a large and active community that provides a vast collection of libraries and frameworks, making it suitable for a wide range of applications, including web development, data analysis, machine learning, and artificial intelligence.

## Python Packages Used

**Python Google search**- The Python Google search package is a third-party library that allows Python developers to perform Google searches programmatically, without the need for manual search queries. It provides a simple and easy-to-use interface for accessing Google search results, making it a useful tool for data analysis, research, and other applications.

**Python Google search**- The Python Google search package is a third-party library that allows Python developers to perform Google searches programmatically, without the need for manual search queries. It provides a simple and easy-to-use interface for accessing Google search results, making it a useful tool for data analysis, research, and other applications.

# TECHNOLOGIES

---

**Beautiful Soup** - Beautiful Soup is a Python package used for web scraping purposes to extract the data from HTML and XML files. It provides a convenient way to parse, navigate, and search HTML and XML documents, making it easy to extract the desired information from a web page.

**PyMongo** - PyMongo is a Python package that provides tools for interacting with MongoDB. It allows Python developers to easily connect to a MongoDB database, insert and retrieve data, and perform complex queries using Python code.

**WhiteNoise** - WhiteNoise is a Python package that simplifies the serving of static files in web applications. It allows developers to easily serve static files directly from their application, without the need for a separate server or file storage system.

**Pandas** - Pandas is a Python package used for data manipulation and analysis. It provides powerful tools for data cleaning, preparation, and exploration, as well as data visualization and statistical analysis.

# TECHNOLOGIES

---

**Argon2** - Argon2 hashing package is a third-party library that provides a simple and secure way to use the Argon2 algorithm in Python applications. It offers several configuration options, such as memory and time costs, and is widely considered to be one of the strongest password-hashing algorithms available.

**pretty\_html\_table** - This is a third-party Python package that provides a simple and flexible way to generate HTML tables from data stored in Python data structures such as lists, tuples, and dictionaries. It offers a variety of options for customizing the appearance of tables, including column and row formatting, sorting, pagination.

# SOFTWARE DESIGN

---

## **Key Components of the web application:**

- i. **Search engine:** This component would allow users to search for their desired journals and one can even know if some journal was fake.
  - ii. **About Us:** This page contains the brief description about the website.
  - iii. **Contact:** One can either contact, or send feedback or raise complaints or even ask for queries from this page.
  - iv. **FAQs:** This component stores the FAQs (or) Frequently Asked Questions by the users.
  - v. **Cookies Consent:** This is used to ensure that users' data is not misused while they browse the internet.
  - vi. **Sorted Journals :** Provides data in a pretty table format which are sorted manually.
-

# DATABASE DESIGN

---

- **journals** - This table contains the data about the title, ISSN, publisher, ID, publisher address, category of journal and the eISSN
- **query\_data** - This stores the query inputs entered by the user.
- **contact\_messages** and **contact\_complaints** - This holds the data about Feedback and the Complaints sent by the users regarding any queries or about the updation.
- **url\_data** - The input URLs entered by the users are stored here, in the database.
- **cloned/fake\_journals** - This stores the ID, title, language, publisher, ISSN and the URL data of journals.

# DATABASE DESIGN

- query\_data

The screenshot shows a MongoDB interface with the 'query\_data' collection selected. The left sidebar lists collections: contact\_messages, demo\_contact\_messages, django\_admin\_log, django\_content\_type, django\_migrations, django\_session, journals, login\_data, query\_data, signup\_data, url\_data, admin, config, and local. The right panel displays three documents from the 'query\_data' collection:

- Document 1: \_id: ObjectId('644793631fe80e95ffbf3a0f'), id: 1, title: "calculus", publisher: "", created\_at: 2023-04-25T08:46:27.675+00:00
- Document 2: \_id: ObjectId('644793881fe80e95ffbf3a11'), id: 2, title: "calculus", publisher: "springer", created\_at: 2023-04-25T08:47:04.924+00:00
- Document 3: \_id: ObjectId('64479e4d819da15be2028095'), id: 3, title: "aabru", publisher: "", created\_at: 2023-04-25T09:33:01.324+00:00

- cloned/fake\_journals

The screenshot shows a MongoDB interface with the 'cloned/fake\_journals' collection selected. The left sidebar lists collections: Accujournal, \_\_schema\_\_, auth\_group, auth\_group\_permissions, auth\_permission, auth\_user, auth\_user\_groups, auth\_user\_user\_permissions, cloned/fake\_journals, contact\_complaint, contact\_messages, demo\_contact\_messages, and django\_content\_type. The right panel displays two documents from the 'cloned/fake\_journals' collection:

- Document 1: \_id: ObjectId('64478eda208acfb74df6090a'), title: "Aabru Journal", language: Array, publisher: "Baljeet Singh Raina", isbn: "2456-253X", url: "https://www.aabru.org"
- Document 2: \_id: ObjectId('64478eda208acfb74df6090b'), title: "Adhigam - International Journal of Hindi Research", language: Array, publisher: "State Institute of Education", isbn: "2394-773X", url: "https://hindijournals.org/index.php/adhigam/index"

- url\_data

The screenshot shows a MongoDB interface with the 'url\_data' collection selected. The left sidebar lists collections: contact\_messages, demo\_contact\_messages, django\_admin\_log, django\_content\_type, django\_migrations, django\_session, journals, login\_data, query\_data, signup\_data, url\_data, admin, and config. The right panel displays three documents from the 'url\_data' collection:

- Document 1: \_id: ObjectId('644793ae1fe80e95ffbf3a12'), id: 1, url: "aabru.com", created\_at: 2023-04-25T08:47:42.201+00:00
- Document 2: \_id: ObjectId('6447a2a4c0c91444ec219fbb'), id: 2, url: "http://www.samvadpath.com/index.php", created\_at: 2023-04-25T09:51:32.280+00:00
- Document 3: \_id: ObjectId('6447a825b7cf831c70cf2771'), id: 3, url: "results", created\_at: 2023-04-25T10:15:01.094+00:00

# FUNCTIONALITY

---

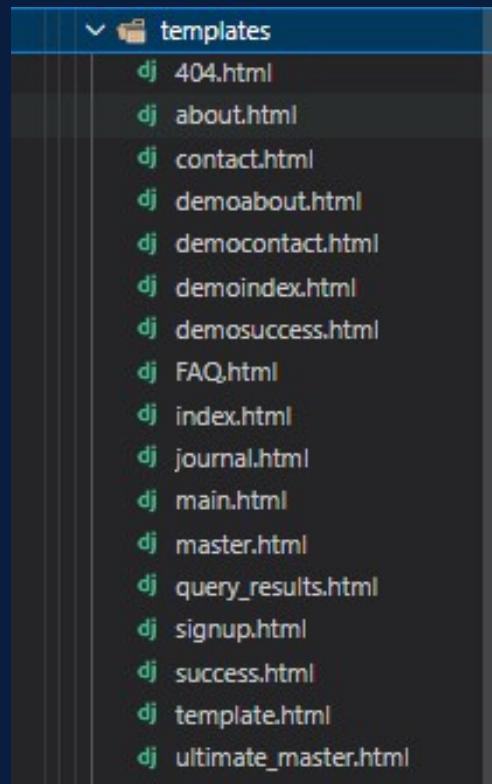
- Modification (or) Updation of the Journals Database and URL Database.
- Maintenance and Monitoring the users' passwords, feedback, complaints and their queries properly.
- Maintain the FAQs systematically so that users can go through them when needed .
- User can go through the journals information subject wise too.
- One can even access the journals just by entering either the title/ISSN , or publisher or the index(in case, one doesnt know all of them)



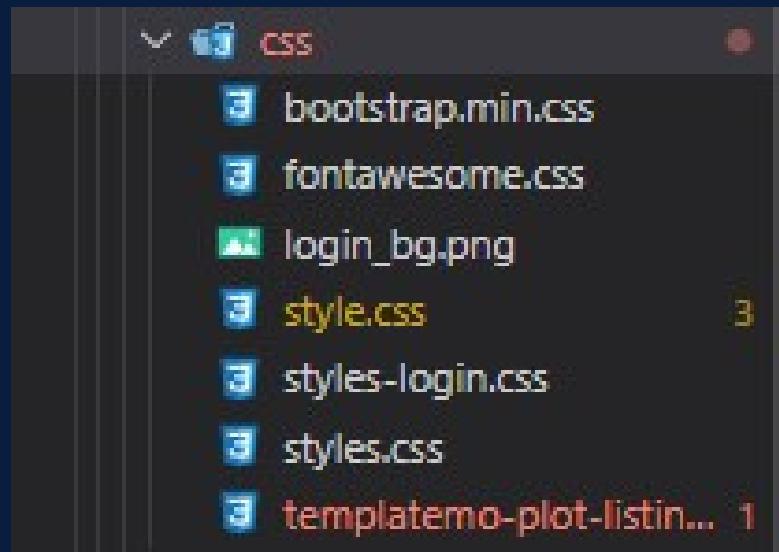
# CODING MODULES

---

## 1. HTML template list

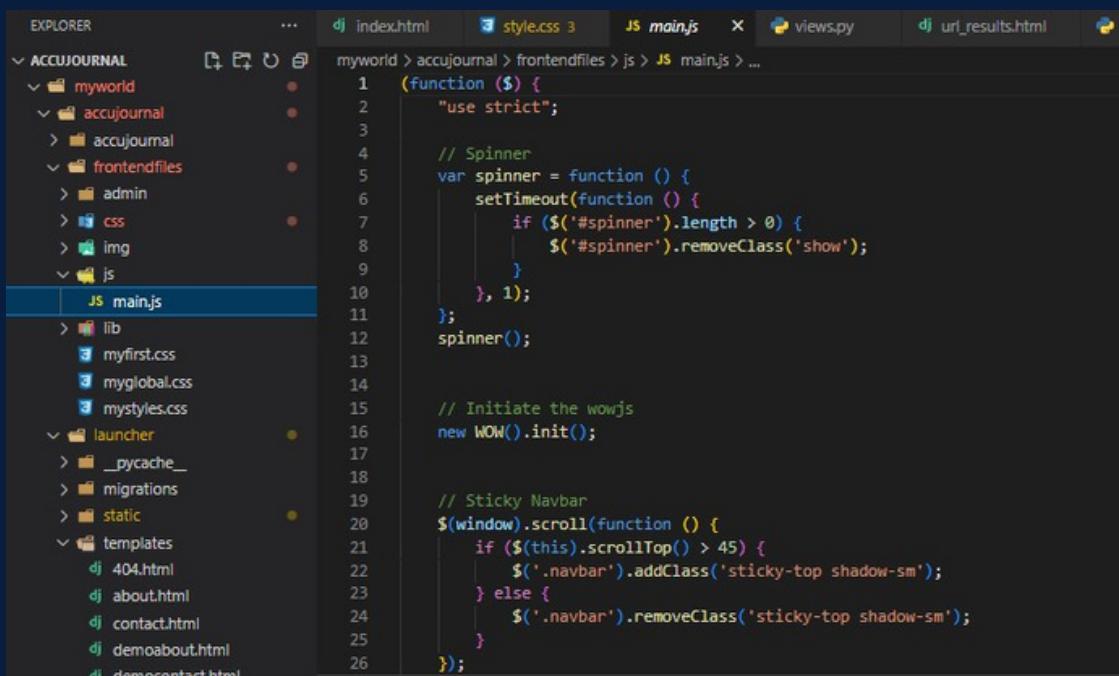


## 2. CSS list



# CODING MODULES

## 3. JavaScript



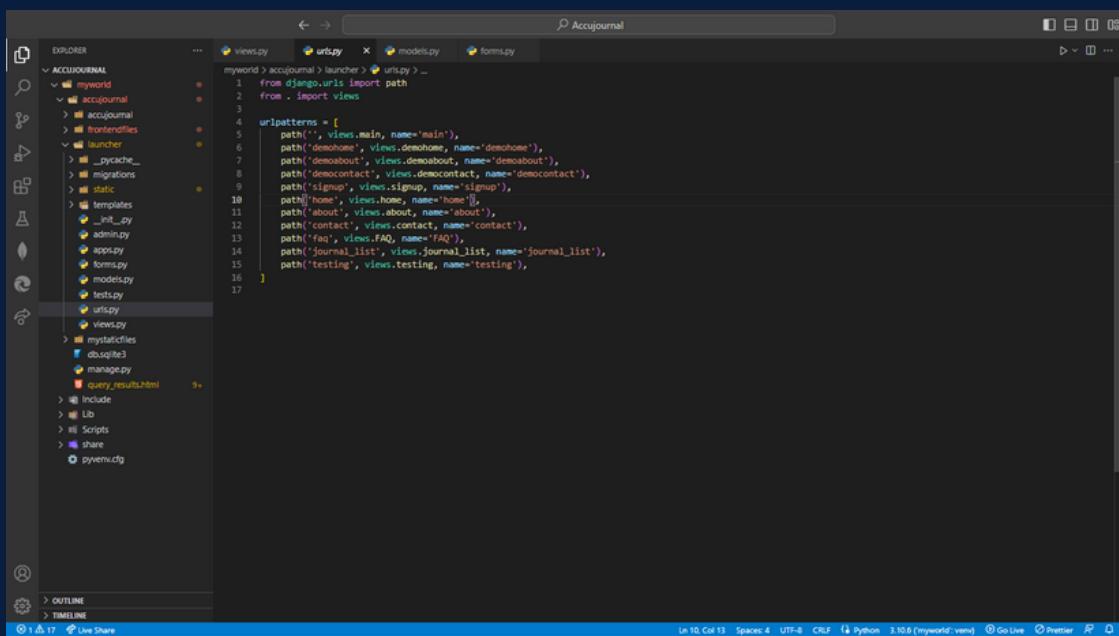
```
function ($) {
    "use strict";

    // Spinner
    var spinner = function () {
        setTimeout(function () {
            if ($('#spinner').length > 0) {
                $('#spinner').removeClass('show');
            }
        }, 1);
    };
    spinner();

    // Initiate the wowjs
    new WOW().init();

    // Sticky Navbar
    $(window).scroll(function () {
        if (this.scrollTop() > 45) {
            $('.navbar').addClass('sticky-top shadow-sm');
        } else {
            $('.navbar').removeClass('sticky-top shadow-sm');
        }
    });
}
```

## 4. URLs



```
from django.urls import path
from . import views

urlpatterns = [
    path('', views.main, name='main'),
    path('demohome', views.demohome, name='demohome'),
    path('demoabout', views.demoabout, name='demoabout'),
    path('democontact', views.democontact, name='democontact'),
    path('signup', views.signup, name='signup'),
    path('home', views.home, name='home'),
    path('about', views.about, name='about'),
    path('contact', views.contact, name='contact'),
    path('FAQ', views.FAQ, name='FAQ'),
    path('journal_list', views.journal_list, name='journal_list'),
    path('testing', views.testing, name='testing'),
]
```

# CODING MODULES

## 5. Views

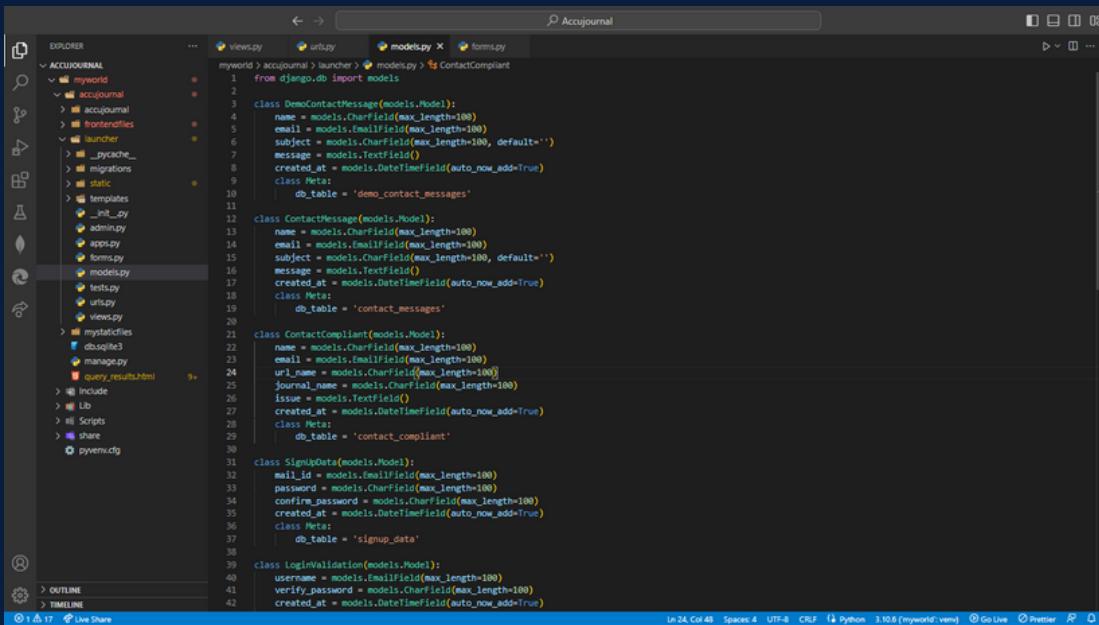
The screenshot shows a Python development environment with the following details:

- File Explorer:** Shows the project structure under "myworld". The "views.py" file is the active tab, containing code for a "Accjournal" application.
- Code Editor:** The "views.py" file contains Python code for a web application. It includes imports for "QueryForm", "UrlForm", and "MongoClient". The code handles POST requests for "query" and "url". For "query", it processes a "title" and "publisher" from the request, creates a query, and returns results as an HTML template. For "url", it performs a search on a MongoDB database and returns the results as an HTML template.
- Terminal:** The terminal shows the command "python manage.py runserver" being run.
- Status Bar:** The status bar at the bottom indicates "In 130 Col 90 Spaces: 4 UTF-8 CR LF & Python 3.10.8 (myworld:venv)".

## 6. Forms

# CODING MODULES

## 7. Models



The screenshot shows a code editor interface with the file `models.py` open. The code defines two Django models: `ContactCompliant` and `ContactMessage`. The `ContactCompliant` model has fields for name, email, subject, message, and created\_at. The `ContactMessage` model has fields for name, email, subject, message, and created\_at. Both models inherit from `models.Model` and have a `Meta` class specifying the database table names.

```
from django import models

class DemoContactMessage(models.Model):
    name = models.CharField(max_length=100)
    email = models.EmailField(max_length=100)
    subject = models.CharField(max_length=100, default='')
    message = models.TextField()
    created_at = models.DateTimeField(auto_now_add=True)
    class Meta:
        db_table = 'demo_contact_messages'

class ContactMessage(models.Model):
    name = models.CharField(max_length=100)
    email = models.EmailField(max_length=100)
    subject = models.CharField(max_length=100, default='')
    message = models.TextField()
    created_at = models.DateTimeField(auto_now_add=True)
    class Meta:
        db_table = 'contact_messages'

class ContactCompliant(models.Model):
    name = models.CharField(max_length=100)
    email = models.EmailField(max_length=100)
    subject = models.CharField(max_length=100)
    journal_name = models.CharField(max_length=100)
    issue = models.TextField()
    created_at = models.DateTimeField(auto_now_add=True)
    class Meta:
        db_table = 'contact_compliant'

class SignupData(models.Model):
    mail_id = models.EmailField(max_length=100)
    password = models.CharField(max_length=100)
    confirm_password = models.CharField(max_length=100)
    created_at = models.DateTimeField(auto_now_add=True)
    class Meta:
        db_table = 'signup_data'

class LoginValidation(models.Model):
    username = models.EmailField(max_length=100)
    verify_password = models.CharField(max_length=100)
    created_at = models.DateTimeField(auto_now_add=True)
```

## 8. Cookies

```
<!-- Cookies-->
<div class="wrapper">
  <header>
    <h2>Cookies Consent</h2>
  </header>

  <div class="data">
    <p>This website use cookies to help you have a superior and more relevant browsing experience on the website. </p>
  </div>

  <div class="buttons">
    <button class="button" id="acceptBtn">Accept</button>
    <button class="button" id="declineBtn">Decline</button>
  </div>
</div>

<script>const cookieBox = document.querySelector(".wrapper"),
      buttons = document.querySelectorAll(".button");

const executeCodes = () => {
  if (document.cookie.includes("cookies")) return;
  cookieBox.classList.add("show");

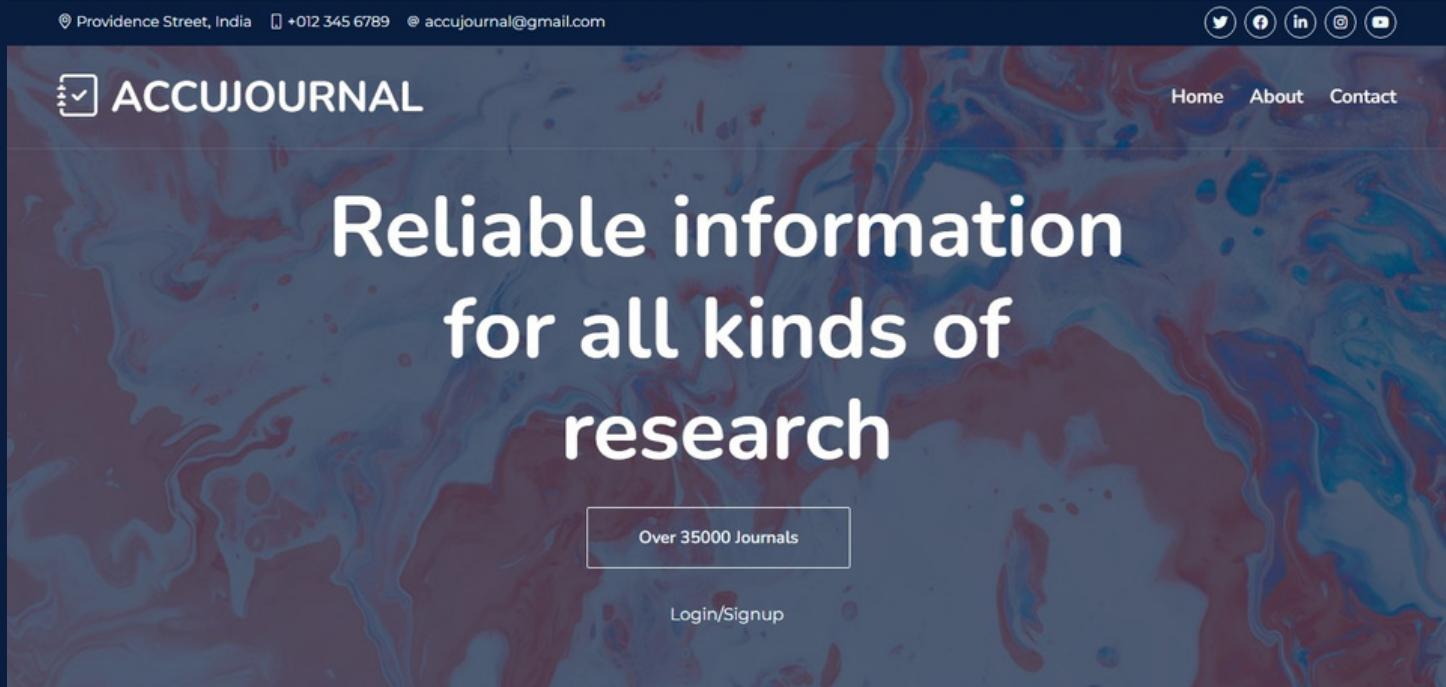
  buttons.forEach((button) => {
    button.addEventListener("click", () => {
      cookieBox.classList.remove("show");

      if (button.id == "acceptBtn") {
        }
      });
    });
}

window.addEventListener("load", executeCodes);
</script>
<!-- Cookies-->
```

# SCREENFLOW

## 1. Home page :



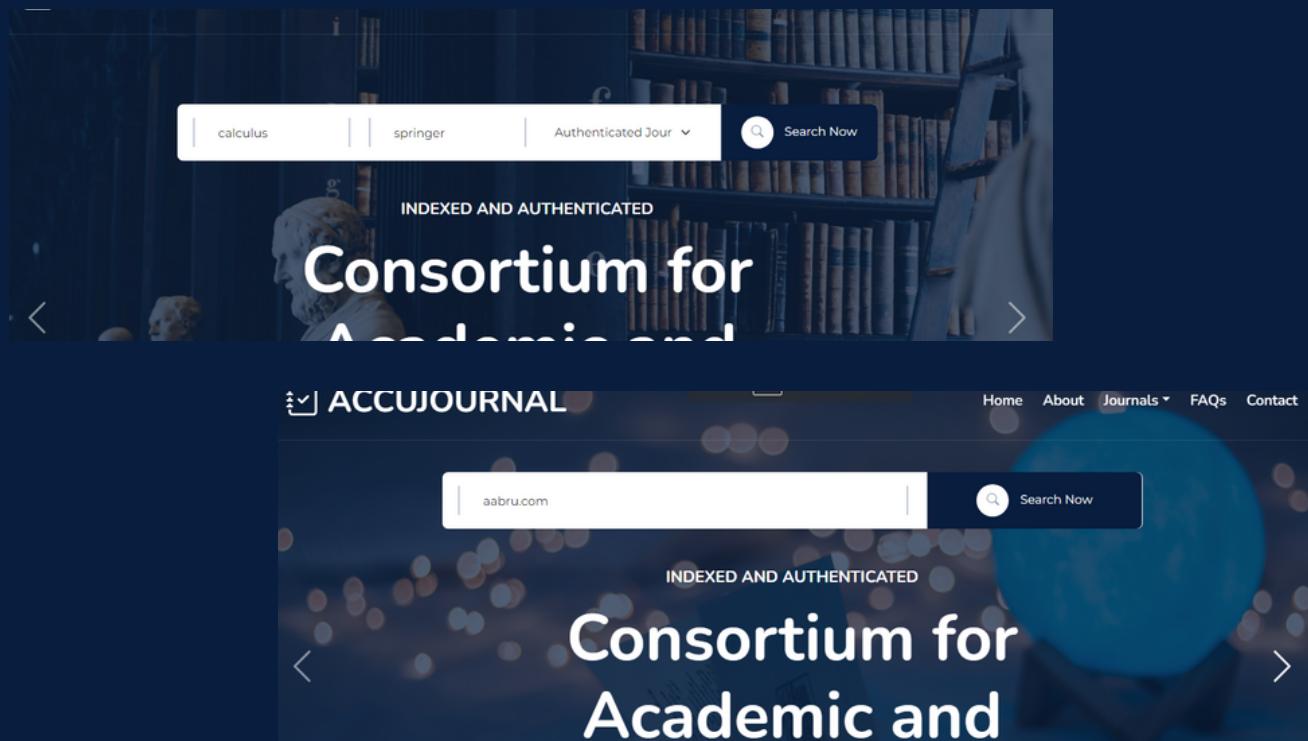
## 2. Login and Sign up :

A screenshot of the ACCUJOURNAL login and sign-up pages. The left side shows the "Login" and "Sign Up" buttons. Below them are fields for "Username / Mail ID" and "Password", each with a placeholder text. A "Login" button is located at the bottom of this section. A link "Don't have an account?" is positioned below the password field. The right side shows the same fields for "Mail ID", "Password", and "Confirm Password", each with a placeholder text. A "SignUp" button is located at the bottom of this section. A link "Already a Member?" is positioned below the "Confirm Password" field. The bottom of the page features the ACCUJOURNAL logo.

### 3. Index Page :



### 4. Search Engine :



# SCREENFLOW

## 4. Feedback and Complaints :

The screenshot shows a web page titled "ACCUJOURNAL". At the top right are links for "Home", "About", "Journals ▾", "FAQs", and "Contact". Below the title, there are two main sections: "Feedback?" and "Compliant?". The "Feedback?" section contains fields for "Your Name", "Your Email", "Subject", and "Message", followed by a blue "Send Feedback" button. The "Compliant?" section contains fields for "Your Name", "Your Email", "Website name/link (if available)", "Journal name", and "What is the issue? Is it a fake journal?", followed by a blue "Send Compliant" button.

## 5. Results :

The screenshot shows a table titled "Your Query Results" with columns: ISSN, Journal title, Publisher address, Publisher name, Web of Science Categories, and eISSN. There are five rows of data:

ISSN	Journal title	Publisher address	Publisher name	Web of Science Categories	eISSN
0944-2669	CALCULUS OF VARIATIONS AND PARTIAL DIFFERENTIAL EQUATIONS	TIERGARTENSTRASSE 17, HEIDELBERG, GERMANY, D-69121	SPRINGER HEIDELBERG	Mathematics, Applied   Mathematics	1432-0835
1311-0454	FRACTIONAL CALCULUS AND APPLIED ANALYSIS	CAMPUS, 4 CRINAN ST, LONDON, ENGLAND, N1 9XW	SPRINGERNATURE	Mathematics, Applied   Mathematics, Interdisciplinary Applications   Mathematics	1314-2224
0944-2669	CALCULUS OF VARIATIONS AND PARTIAL DIFFERENTIAL EQUATIONS	TIERGARTENSTRASSE 17, HEIDELBERG, GERMANY, D-69121	SPRINGER HEIDELBERG	Mathematics	1432-0835
1311-0454	FRACTIONAL CALCULUS AND APPLIED ANALYSIS	CAMPUS, 4 CRINAN ST, LONDON, ENGLAND, N1 9XW	SPRINGERNATURE	Mathematics	1314-2224

The screenshot shows a table titled "Your Url Results" with a single column of URLs:

Your Url Results
<a href="https://www.rekhtadictionary.com/meaning-of-aabru">https://www.rekhtadictionary.com/meaning-of-aabru</a>
<a href="https://www.falzasaqlain.pk/products/aabru">https://www.falzasaqlain.pk/products/aabru</a>
<a href="https://nainpreet.com/products/faliza-saqlain-aabru">https://nainpreet.com/products/faliza-saqlain-aabru</a>
<a href="https://www.rekhta.org/urdudictionary?keyword=aabru.en">https://www.rekhta.org/urdudictionary?keyword=aabru.en</a>
<a href="https://www.mesho.com/aabru/p/2zalmj">https://www.mesho.com/aabru/p/2zalmj</a>
<a href="https://unfoldself.com/product/aabru-chandbalis-blue/">https://unfoldself.com/product/aabru-chandbalis-blue/</a>
<a href="https://www.justdial.com/kapur/aabru-jewelers-sikandra/0512px512-x512-170930101943-B4j6_Bzdet">https://www.justdial.com/kapur/aabru-jewelers-sikandra/0512px512-x512-170930101943-B4j6_Bzdet</a>
<a href="https://www.shivkrupaenterprise.com/alok-suit-aabru-pure-french-crape-digital-printed-suit-online">https://www.shivkrupaenterprise.com/alok-suit-aabru-pure-french-crape-digital-printed-suit-online</a>
<a href="https://solankitextileagency.com/alok-aabru">https://solankitextileagency.com/alok-aabru</a>
<a href="https://www.flipkart.com/aabru/p/itmfa5y52euqxxx">https://www.flipkart.com/aabru/p/itmfa5y52euqxxx</a>
<a href="https://en.wikipedia.org/wiki/Aabroo_(film)">https://en.wikipedia.org/wiki/Aabroo_(film)</a>

# SCREENFLOW

---

## 6. Queries :

The screenshot shows the 'Queries?' page of the ACCUJOURNAL website. At the top, there are three contact options: a phone icon followed by 'Call us at +012 345 6789', an envelope icon followed by 'Mail us @ accujournal@gmail.com', and a location pin icon followed by 'Office Providence Street, India'. Below these are four input fields: 'Your Name', 'Your Email', 'Subject', and 'Message'. A large blue 'Send' button is at the bottom. To the right of the input fields is the ACCUJOURNAL logo, which features a stylized tree with roots and the word 'ACCUJOURNAL' written around its base.

## 7. About Us:

The screenshot shows the 'ABOUT US' page of the ACCUJOURNAL website. The background features a dark image of hands flipping through a stack of papers or books. Overlaid on this image is the word 'ABOUT US' in large, white, sans-serif capital letters. At the top, the ACCUJOURNAL logo is on the left, and 'Home' and 'Contact' links are on the right. Below the main title, there is a large white rectangular box containing text about the portal's purpose and data sources. The text states: 'The ACCUJOURNAL is a publicly available portal that includes the journals and country scientific indicators developed from the information contained in the Scopus, EBSCO, Google Scholar, UCC CARE, PubMed, SCIMAGOJR databases. These indicators can be used to assess and analyze scientific domains. Journals can be compared or analysed separately. Country rankings may also be compared or analysed separately. Journals can be grouped by subject area (27 major thematic areas), subject category (309 specific subject categories) or by country. Citation data is drawn from over 34,100 titles from more than 5,000 international publishers and country performance metrics from 239 countries worldwide.' At the bottom of the box, there is smaller text: 'A journal index, or a list of journals organized by discipline, subject, region and other factors, can be used by other researchers to search for studies and data on certain topics. As an author, publishing your research in an indexed journal increases the credibility and visibility of your'

# SCREENFLOW

## 8. Successful receipt :

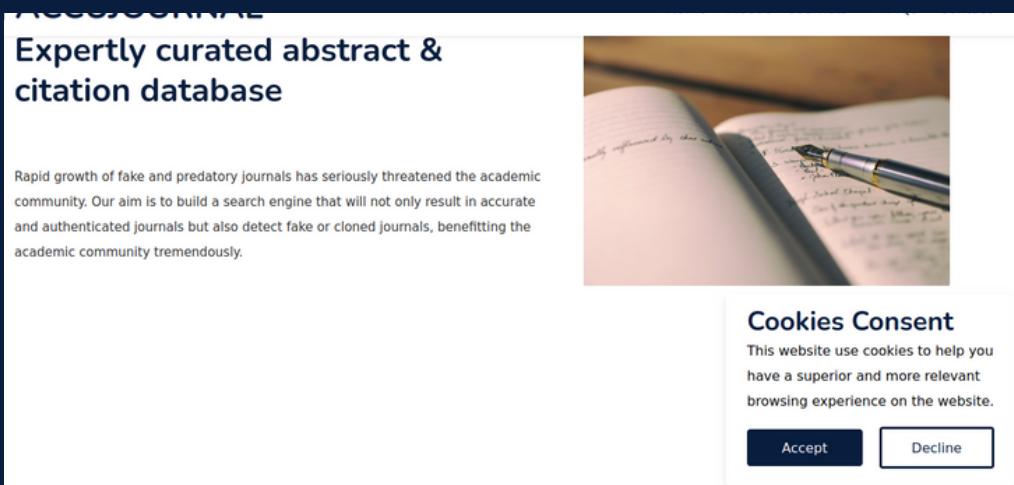
The screenshot shows the ACCUJOURNAL website. At the top, there's a navigation bar with links for Home, About, Journals, FAQs, and Contact. Below the navigation, the word "Querries?" is displayed in a large, bold font. Underneath, there are three contact options: a phone icon followed by "Call us at +012 345 6789", an envelope icon followed by "Mail us @ accujournal@gmail.com", and a location pin icon followed by "Office Providence Street, India". A "Thank You!" message is present, along with a note about message delivery and a disclaimer about fake journals. To the right of the text is the ACCUJOURNAL logo, which features a stylized tree with roots and the word "ACCUJOURNAL" written around its base.

## 9. Not found response :

The screenshot shows the ACCUJOURNAL website. The top navigation bar includes Home, About, Journals, FAQs, and Contact. Below the navigation, a banner features the word "Results" over a background image of a hand holding a book. The main content area is titled "Your Query Results" and contains a message stating "No results found." It also includes a note about reporting missing journals and a verification message. A "Thank you!" message is at the bottom. At the very bottom of the page, there are several small logos and links for indexing services like Google Scholar, CrossMark, and others.

# SCREENFLOW

## 10. Cookies Consent :



## 11. Indexes :



## 12. FAQs :

A screenshot of the AccuJournal website's FAQ section. The background shows a close-up of a person's hand holding an open book. The word 'FAQs' is overlaid in white text. Below the image, the word 'FAQs' is centered in a dark blue header bar. The main content area contains five numbered questions and their answers:

- 1. Where can one find ACCUJOURNAL?**  
You'll get accujournal at <http://192.168.34.149:5555>
- 2. What is the Accujournal?**  
The Accujournal is a trustworthy database for all the indexed and authenticated journals.
- 3. What are predatory/ dubious/ questionable journals or publishers?**  
According to Committee on Publication Ethics(COPE) "Predatory publishing is generally defined as for-profit open-access journal publication of scholarly articles without the benefit of peer review by experts in the field or the usual editorial oversight of the journals in the question."
- 4. How often Accujournal is updated?**  
Accujournal is updated quarterly i.e. on first of January, April, July, and October every year.
- 5. Why few journals are removed before completing the quarter?**

# TEST CASES

---

**Test Case 1 :**

**Name:** Login/Sign up Test

**Objective:** To verify that the Login/Sign up functionality of the software works correctly.

**Test Steps:**

- i. Launch the Home page of the website.
- ii. Click on the Login/Sign up.
- iii. Enter login credentials.
- iv. Click on the Login Button.
- v. Leads the user to Index Page after successful login.

**Expected Result:** After completing the test steps, the user should be successfully logged in and directed to the home page. The login functionality should work as expected.

**Actual Result:** Same as Expected Result. Test Case Passed.

---

# TEST CASES

---

**Test Case 2 :**

**Name:** Search Engine Test

**Objective:** To verify that the Search engine of the software works correctly.

**Test Steps:**

- i. Go to the Index page of the website.
- ii. Enter either the ISSN/title, publisher, or both in the search engine and click on Search Now.
- iii. You get the particulars of all the journals in the database with the entered title along with their ISSN, publisher name, and their category.
- iv. If you couldn't get the desired details, change the Authenticated Journals to Cloned/Fake Journals.
- v. If you find the entered journal in this database, it means that that specific journal is a fake journal.

**Expected Result:** After completing the test steps, the user should successfully be able to get the desired and lucid results.

**Actual Result:** Same as Expected Result. Test Case Passed.

---

# TEST CASES

---

Test Case 3 :

Name: Customer Complaint/Feedback test

Objective: To verify that the entered Complaint/Feedback by the users is stored in the database and the users get it's acknowledgement correctly.

Test Steps:

- i. Go to the Contact page of the website.
- ii. Enter the Name, Mail id, Subject, and Message(Feedback) and click on Send Feedback.
- iii. You get the acknowledgement that the message(feedback) has been received successfully.
- iv. Enter the Name, Mail id, Website name/link(for reference), Journal name(which is to be re-verified), the issue the user is facing and click on Send Complaint.
- v. You get the acknowledgement that the message(complaint) has been received successfully.

Expected Result: After completing the test steps, the user should successfully receive the acknowledgement(that the message has been received successfully) and the sent Feedback or Complaint is to be stored in the database systematically.

Actual Result: Same as Expected Result. Test Case Passed.

---

# REFERENCES & SOURCES

---

w3schools.com

<https://chat.openai.com>

<https://mjl.clarivate.com>

<https://www.mongodb.com/docs>

unsplash.com

<https://colorhunt.co>

<https://icons.getbootstrap.com>

<https://themewagon.github.io/startup2>

<https://www.udemy.com/course/the-complete-web-development-bootcamp/>



Thank You Swami.