

IMPLEMENTATION OF MVC PATTERN IN CONTENT MANAGEMENT SYSTEM USING CODEIGNITER AS SKELETON FRAMEWORK.

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

Rahul Singh(38110443)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MARCH - 2022



SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Rahul Singh(38110443)** who carried out the project entitled "IMPLEMENTATION OF MVC PATTERN IN CONTENT MANAGEMENT SYSTEM USING CODEIGNITER AS SKELETON FRAMEWORK." under my supervision.

Internal Guide
Dr. G.Kalaiarasi M.E,Ph.D

Head of the Department
Dr. L. Lakshmanan M.E., Ph.D.

Submitted for Viva voce Examination held on_____

Internal Examiner

External Examiner

DECLARATION

I **Rahul Singh(38110443)** here by declare that the Project Report entitled“ IMPLEMENTATION OF MVC PATTERN IN CONTENT MANAGEMENT SYSTEM USING CODEIGNITER AS SKELETON FRAMEWORK.”

done by me under the guidance of Dr./Prof./Mr./Ms.KalaiarasInternal) is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering / Technology degree in COMPUTER SCIENCE AND ENGINEERING.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D, Dean**, School of Computing **Dr. L. Lakshmanan M.E., Ph.D. ,** and **Dr.S.Vigneshwari M.E., Ph.D. Heads** of the Department of Computer Science and Engineering for providing necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Dr./Mr./MsKalaiaarasi for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Content Management System is computer software that helps and allows the user to create and publish digital content on the World Wide Web. It helps the user to collaborate in editing and managing the content, allowing multiple contributors to publish the subject theme. In this paper, the attempt has been made to design and develop a better Content Management System from scratch using the technique of implementation of MVC OOP on PHP using the CodeIgniter skeleton Framework. This research paper will describe in theory the creation of a web application that will allow for the creation, posting, publishing, and also retrieval of content from the database with proper security. It will further explain the set of policies and procedures along with a supporting catalog for content submissions, review, and approval. Personal dashboards or personal tools will be available for each category of their users for their proposed categories of work. The Skeleton Framework used will be PHP CodeIgniter which is used to develop web applications using the simple and elegant toolkit. The framework, or often called the web development framework is a structure that supports the development of dynamic websites and web applications. We will implement various design practices and algorithms in order to achieve often reusable codes customizable designs and various manageable resource services, and API-related tasks. In this paper, it will be discussed in detail about the various requirements including hardware requirements, software requirements, and server requirements in order to achieve the desired objective. This research looks at

current practices to help developers' better grasp how to use PHP frameworks in web development apps. Our approach of generating CRUD (Create, Read, Update, and Delete) operations is used to validate this proposal.

Keywords – CodeIgniter, Model-View-Controller, PHP, Content Management System, CRUD OperationsOOP.

CHAPTER	NAME	Page Numbers
1	Introduction	6-7
2	Litrature Review	8
3	Methodlogy	9-13
4	Working with data base	14-18
5	ALGORITHM USED ARE AS FOLLOWS	18-20
6	MODULE'S WISE WITH THEIR EXPLANATION AS USED IN THE PROJECT	20-23
7	Conclusion	23-24
8	References	24-25
9	Appendices	25-32
10	Screenshots	33-34

The rapid development of web based application has affected the coding methodology. This takes our attention toward a higher need of feasibility and maintainability. Other than that, It is a general-purpose and an open source tool kit that requires minimal setup. To decrease and deal with the level of complexity in the nature of the project, various techniques for programming in PHP has been proposed. One of those is the Object Oriented Programming OOP and MVC design pattern with some short of factory design pattern. The concept used in this paper of a PHP Framework is related to Object Oriented Programming and Model View Controller. MVC is a design pattern that applies to the concept of software development and originally designed to provide multiple views of the same data for virtually modern interactive applications.

This component stores data and represent data structure relationship and dependencies with database. Model will only deal with the data base so the user does not have the authority to access the models. Model will only be available for the controller and as per the user requirements we will modify and make model classes. In order to accomplish difficult task carried out at the database end.

It is the component which represents the presentation of the data. Various devices can be used for presentation of data in which the application has multiple views of the data. Any user is allowed to view all the UI components like drop down boxes and text boxes among many other features. It will facilitate users to input their data and our system at the backend will manipulate data and again provide the data that will be required at the users end.

It is the component primarily responsible for handling the user interaction. The controller interprets the inputs from the user and informs model and view to change as required. It is represented as an information collector and acts as the glue between model and view.

Controller is the kind of main component that will follow complete business logic that we want to implement in any kind of system like here we are implementing it in Content management system.

The idea behind using MVC Pattern in PHP Framework is that the code presentation and layout will be much simpler and well separated and divided that can make the project easily manageable. It is helpful in providing an interface though which all the classes corresponding to the logical object of the application can be manipulated.

Model – This component stores data and represent data structure relationship and dependencies with database. Model will only deal with the data base so the user does not have the authority to axccess the models. Model will only be available for the controller and as per the user requirements we will modify and make model classes. In order to Accomplish difficult task carried out at the database end.

View – It is the component which represents the presentation of the data. Various devices can be used for presentation of data in which the application has multiple views of the data. Any user is allowed to view all the UI components like drop down boxes and text boxes among many other features. It will facilitate users to input their data and our system at the backend will manipulate data and again provide the data that will be required at the users end.

Controller – It is the component primarily responsible for handling the user interaction. The controller interprets the inputs from the user and informs model and view to change as required. It is represented as an information collector and acts as the glue between model and view.

Controller is the kind of main component that will follow complete business logic that we want to implement in any kind of system like here we are implementing it in Content management system.

The idea behind using MVC Pattern in PHP Framework is that the code presentation and layout will be much simpler and well separated and divided that can make the project easily manageable. It is helpful in providing an interface though which all the classes corresponding to the logical object of the application can be manipulated.

Umi Saadahetal all proposed thatthe development of projects from pure PHP takes a long time. However, a framework is often using to ease the task. The most popular is the MVC model. However, sometimes the complex design could not produce optimum performance and they mentioned the drawback as “This framework provides a very less amount of resources when compared to other frameworks.”

KhaoulaBenmoussa , Majida Laaziri ,Samira Khoulji, Mohamed Larbi Kerkeb, Abir El Yamamiyet.all all proposed that there are a lot of frameworks available for the purpose of designing. However, choosing the correct framework is not often an easy task. This literature provides a complete model to evaluate and compare the main PHP frameworks and they also mentioned the drawback as the comparison criteria are sometimes not accurate.

SUBARI, ARKHAN yet.all all proposed that Various computer applications are used to assess the success of achieving organizational goals embodied in specific measures.The drawback was mentioned as theAccuracy of the model is often questioned.

Das Somayet.all all proposed that this shows the flexibility in storing real time data. Along with that the development of NTNU- QGIS web API will provide access to various devices to simultaneous collect and store data in the database to visualize at the same time.

Moilanen, Markusyet.all proposed thatthis introduces the PHP Framework Symfony. It uses MariaDB database. The thesis covers the most essential aspects of developing said features, focusing on the critical issues and solutions to them.

MuhammadIrfanKhanyet all proposed that the techniques presented in this article will help you efficiently and effectively create large-scale applications that are otherwise difficult to manage. This framework methodology was developed to implement application development using the Codeigniter 2.1.0 framework.

Natalya Prokofyevayet all proposed that this document provides an overview of the popularity of various PHP programming frameworks (CakePHP2, CodeIgniter, Symfony2, Yii, PhalconPHP) and comparisons based on different criteria. The tests were run using the ab.exe (Apache Benchmark) tool that came with the Apache web server.

One of those is the Object Oriented Programming OOP and MVC design pattern with some short of factory design pattern. The methodology used in this paper of a PHP Framework is related to Object Oriented Programming and Model View Controller. MVC is a design pattern that applies to the concept of software development and originally designed to provide multiple views of the same data for virtually modern interactive applications. This component stores data and represent data structure relationship and dependencies with database. Model will only deal with the data base so the user does not have the authority to access the models. Model will only be available for the controller and as per the user requirements we will modify and make model classes. In order to accomplish difficult task carried out at the database end. It is the component which represents the presentation of the data. Various devices can be used for presentation of data in which the application has multiple views of the data. Any user is allowed to view all the UI components like drop down boxes and text boxes among many other features. It will facilitate users to input their data and our system at the backend will manipulate data and again provide the data that will be required at the users end.

It is the component primarily responsible for handling the user interaction. The controller interprets the inputs from the user and informs model and view to change as required. It is represented as an information collector and acts as the glue between model and view.

Controller is the kind of main component that will follow complete business logic that we want to implement in any kind of system like here we are implementing it in Content management system.

The idea behind using MVC Pattern in PHP Framework is that the code presentation and layout will be much simpler and well separated and divided that can make the project easily manageable. It is helpful in providing an interface through which all the classes corresponding to the logical object of the application can be manipulated.

The integration of frameworks has revolutionized the generation of web applications, and CodeIgniter is one of these frameworks that facilitate the construction of PHP web applications.

In this paper the architecture that has been used is standard MVC architecture. It is beneficial for easy visualization, enhanced security, modularity, and overall project performance. The **DRY** (Don't Repeat Yourself) concept is used herein. It decreases the number of lines in the code and hence provides better performance. DRY Concept is extremely helpful in eliminating redundancies in the process as well as in logic. One of the key goals of the DRY principle is to make code more maintainable and increase

reusability of the same code in different scenario at all stages of its development. When the DRY principle is followed, we are able to update code in one place and have the change implemented to all occurrences of the code in question automatically.

We shall attempt to provide optimized security in both front end and back end and in backend we will be using pre defined security features or CodeIgniter and in the front end we will define our security guidelines and security codes that will be responsible for the perfect output for the clients or the users end. As a result of better optimization the burden on the hardware of the user decreases significantly. Additionally, it also helps in the project performance enhancement. Apart from these, better and strict implementation of the MVC pattern provides increased project modularity and better visualization of the system. It helps the project become more user-friendly.

Software requirements

1. Windows 7 and above. The latest operating systems make it easier and feasible to achieve the objectives of our proposed system. For the optimal performance it is suggested that the user makes the use of latest operating systems at their disposal.
2. PHP 7.3 or above
3. XAMPP server for local testing and deployment
4. MySQL – It is an open source relational database management system.

Server requirements

1. PHP version 7.3 or newer is required, with the **intl** extension and **mbstring** extension installed.
2. The following PHP extensions should be enabled on our server:
3. PHP-json,Php-xml.
4. In order to use the CURL Request, We will also need libcurl installed.
5. A database is required for most web application programming. Currently supported databases are:
6. a) MySQL (5.1+) via the *MySQLi* or *PDO* driver.

The programming languages and other frameworks used in our proposed work are listed below –

1. **HTML** – We have used HTML (Hypertext Markup Language) which will primarily function in the front end. It is designed for the documents to be displayed in a web browser. HTML will be used for the basic designing for the front end that will be available for the client or to the servers end.
2. **CSS** - CSS is the extension to the HTML which modifies the design and display of the web pages. In our proposed work we have used CSS to beautify the view. Enhancing the view has a positive impact on the user making the project more user-friendly user-interfaces.
3. **JavaScript** – JavaScript is used for the motion and also for the functionality at the client side.
4. **PHP** - It is server side scripting language used to design and implement web service as well as it is used inside the framework CodeIgniter skeleton framework.
5. **PHP CodeIgniter** – CodeIgniter is a framework of PHP which is highly secured. It is mostly famous for having a very small footprint. CodeIgniter is also helpful in handling errors, customizability, security, better User Interface, and it is based on MVC Pattern. It also provides active record classes for handling all the queries very securely and to eliminate some short of errors that will occur at the clients end.
6. **SQL** – It has been used for the querying of the data base which is present in the data base.

Content Management System is a software platform whose main purpose is to automate the tasks required to manage and publish content, such as uploading and formatting website content. Basically, a CMS makes it easy for people of all tech levels to manage their website. The CMS takes input through the user-friendly interface and remodels it into a website that works for the visitors.

In our proposed system the users, who are the visitors on the website would just be able to view the posts and like it as well. However, the subscribers who have made an account on the website would be able to post their contents and comment on posts. For an added layer of security, the content posted by the subscriber would not be directly posted on the website. It will go for a review to the Admin or the Manager. Upon further

acceptance the post will be published on the website. This is because we know that java script will be executed as the as it gets the DOM (Document . READY FUNCTION).)

Implementation of MVC discussed in depth.

We have connected the CodeIgniter with our website database. Just after entering the CodeIgniter interface we see the folder named 'assets'. In the assets folder all the folders including admin, manger, subscriber, and home are present.

- In the admin folder we can see all the files of admin dashboard and the css and js files related to it.
- In the manager folder, all the css and js files of manager are present.
- Same is the case with subscriber folder which includes all the css and js files of subscriber.

In the Home folder, the files are present of css and js including the assets folder of Home and the contact form.

In the **Config** folder, we find the several files including **Autoload** and **database**.

- **Autoload** basically keeps the files preloaded for our visit in the same section again.
- In the **database** file, we can find the three main stored information. The username, password, and the database.

This Project is a CMS(Content Management System) based Software in which the we have divided them to 4 parts.The first one is the Users.The users are the ones who just visit the websites.The second one is the Subscribers.Subscribers are the ones who have created the account in our website.The third one is the Manager who can manage the subscribers as well as the manager can manage all the poems and can activate all the poems and deactivate the poems.The Fourth one we have a Admin who can manage all the managers as well as the subscribers.Admin can also view all the poems and can activate and deactivate all the poems.

As the users visits the website they will find a home page which will consists of a header in which it have the search module and the menu (Home, Contact, Login) bellow which we have a news corner and after the news corner we have got all poems where the

poems can be seen one by one in pagination. In the poems section the poems can be liked as we can comment and can also be read more. Below the poems it has got a space in which the trending poem can be seen according to the likes and the comments which is named as popular posts. After the popular post section we have a section named as latest post in which the users will be able to see the recently posted poems. In the view page we also have an option to view the poems category wise. We also have the module names as search module in which we can search the poems according to our needs and view them. The users can also subscribe to news letter if they want to get the notification and stay updated with us. We have tried our level best to make it as user friendly as well as seo friendly giving all the necessary links like Instagram page as well as the facebook links and few more meta tags for better seo.

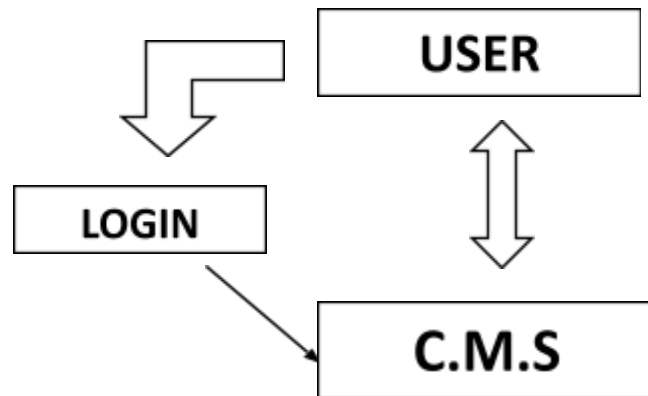
Once the user creates the account he will have the status as deactivated. Until and unless the user is activated by the admin or the manager. Once the admin or the manager activates the user then the user can log-in. In this we have a common login from which the user can log-in to their dashboard and add their own poems. The importance of making a common log-in is just to get the code minimized and we can follow a good concept of DRY which is Don't Repeat Yourself which states that once the code is written we need to reuse the same type of codes. As the subscriber gets its dashboard after it has logged in successfully he has two options to upload his own poems for which CK editor has been used for better and best performance and he can view the home page from the session itself and can change the profile details if he wants to change and update it. He/She can also view his/her all poems and can also activate and deactivate the poem that he/she is not willing to show to the users. Now after subscribers we have a manager who can manage the subscriber in the manager's dashboard the manager can add new categories can add new subscribers and also can view all the poems and can also do its CRUD(Create Read Update Delete) Active and Deactive operations. We have an Admin who can manage all the Managers, Subscribers, All Poems, System settings, Add Poems, Master Entries for the required options or fields and contact us. He can perform all the crud operations (Create Read Update Delete) at the same time activate and deactivate. The admin has also got the rights and authority to add admin add managers, add subscribers as well as add poems. He will also be able to see the people who contact them via contact us page. Since it is totally dynamic we have implemented website setting in which the admin can change anything on the view part like (Page Title) from his own dashboard.

When the subscriber is logged in then the session is set during the login so that we have a good track of the user.

Working of the Database

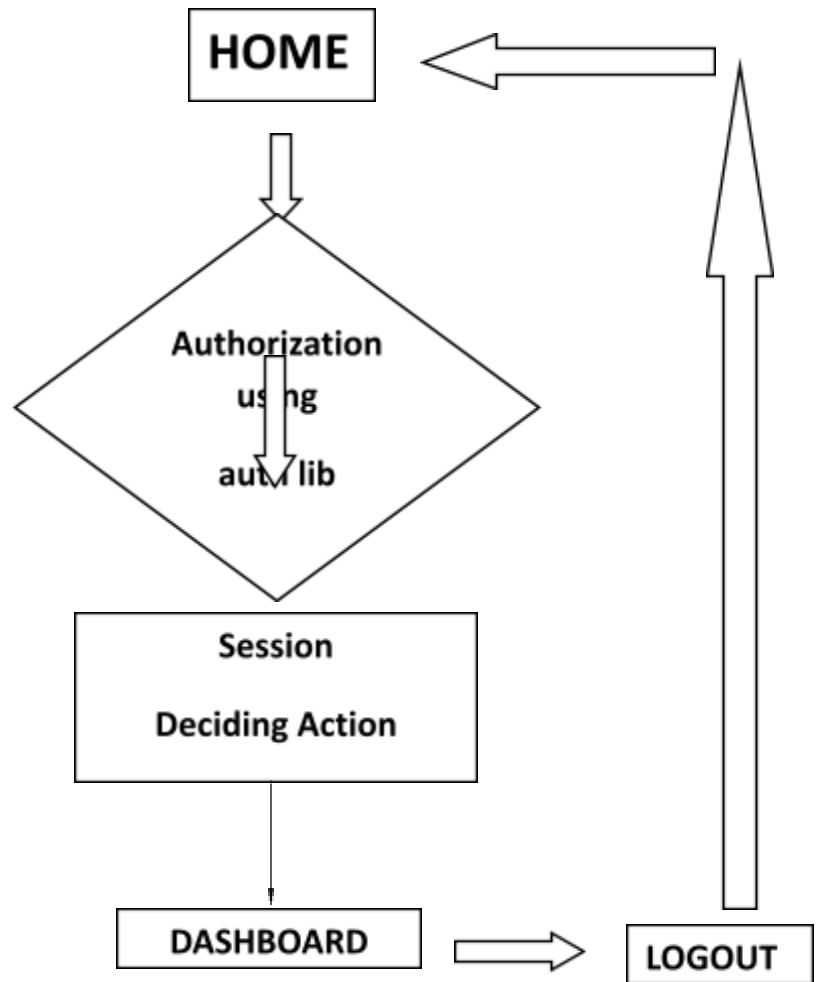
The working of the Database can be conveniently explained using the Data Flow Diagrams or DFD.

Level 1 DFD:



The user can visit the Content Management System by either as a visitor or as a subscriber by logging in.

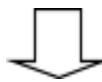
Level 2 DFD:



After coming to the home screen the visitor will be allowed to go through an authorization process using the third party software `auth.lib`. After getting authorized a session will be started to decide if visitor is an admin, manager, or subscriber. After that he will be redirected to the concerned dashboard. After operation their CRUD Operations the user will be logged out and return to the home screen.

Level 3 DFD:

LOGIN



ADMIN

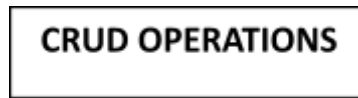


MANAGER



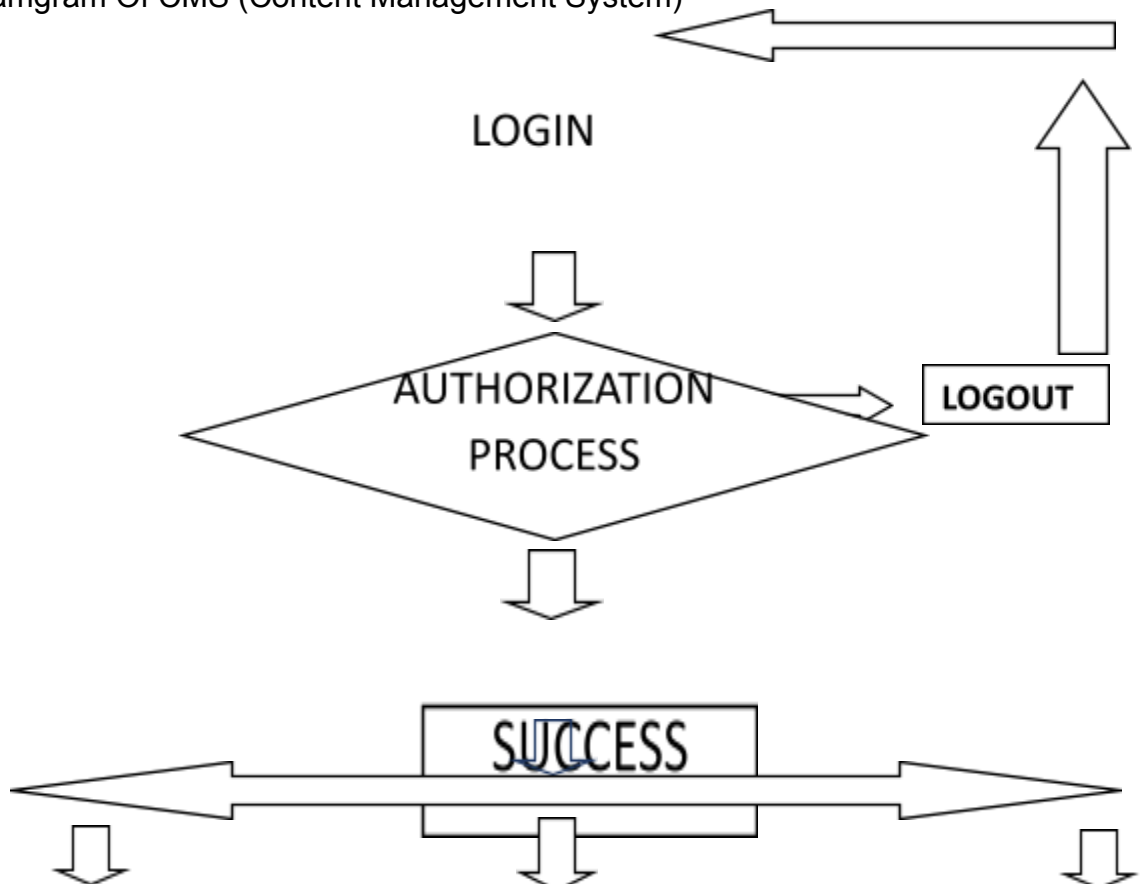
SUBSCRIBERS

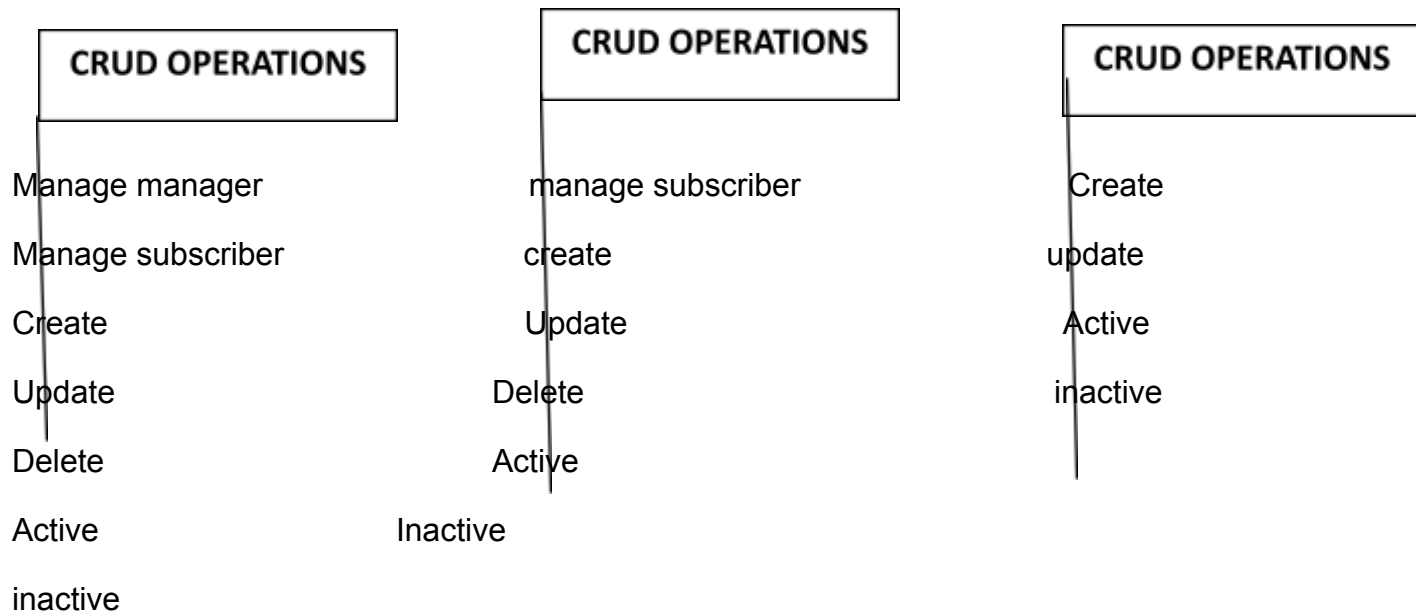




After the visitor logs in on the website, they will have to go through an authorization process. Upon successful authorization, the system will redirect them to the appropriate dashboard. The admin will have a separate dashboard, the manager will have a separate dashboard and the subscribers will be redirected to a different dashboard. All three of them will be allowed to perform their CRUD Operations separately without each other's interference.

ER Diagram Of CMS (Content Management System)





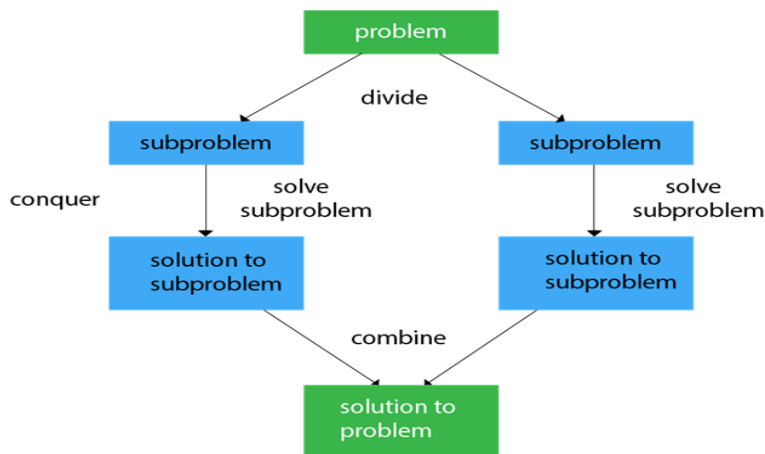
ALGORITHM USED ARE AS FOLLOWS

- **CSRF TOKENS:-** A CSRF Token is a secret, unique and unpredictable value a server-side application generates in order to protect CSRF vulnerable resources.
- **DON'T REPEAT YOURSELF (DRY) Concepts:-** Dry concept is used so that the codes ones written is a function is used again and again.
- **SEARCHING ALGORITHMS:-** Searching Algorithms are designed to check for an element or retrieve an element from any data structure where it is stored. Based on the type of search operation, these algorithms are generally classified into two categories:
 1. **Sequential Search:** In this, the list or array is traversed sequentially and every element is checked. For example: Linear search.
 2. **Interval Search:** These algorithms are specifically designed for searching in sorted data-structures. These type of searching algorithms are much more efficient than Linear Search as they repeatedly target the center of the search structure and divide the search space in half. For Example: Binary Search.

- **DIVIDE AND CONQUER ALGORITHMS(BROKEN CODES INTO SMALLER FUNCTIONS):-**

Both merge sort and quicksort employ a common algorithmic paradigm based on recursion. This paradigm, **divide-and-conquer**, breaks a problem into subproblems that are similar to the original problem, recursively solves the subproblems, and finally combines the solutions to the subproblems to solve the original problem. Because divide-and-conquer solves subproblems recursively, each subproblem must be smaller than the original problem, and there must be a base case for subproblems. You should think of a divide-and-conquer algorithm as having three parts:

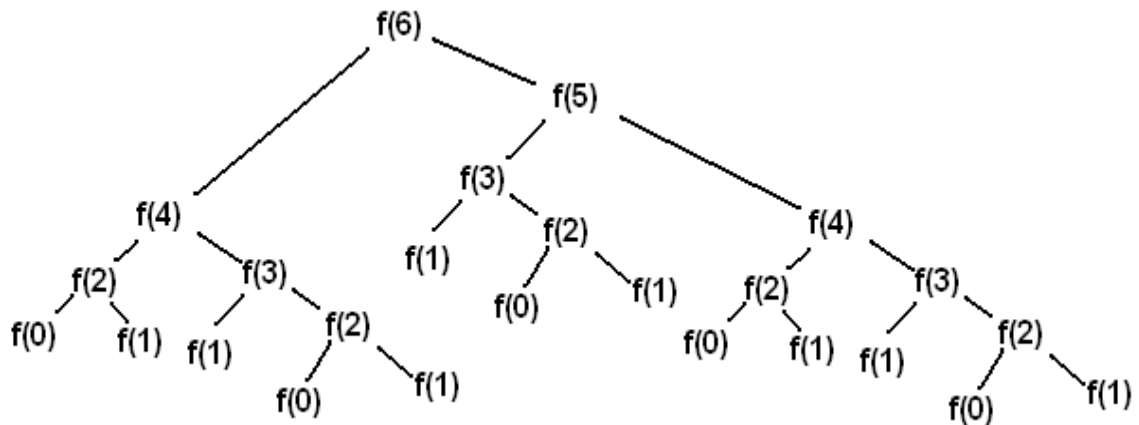
1. **Divide** the problem into a number of subproblems that are smaller instances of the same problem.
2. **Conquer** the subproblems by solving them recursively. If they are small enough, solve the subproblems as base cases.
3. **Combine** the solutions to the subproblems into the solution for the original problem.



- **DYNAMIC PROGRAMMING ALGORITHMS**

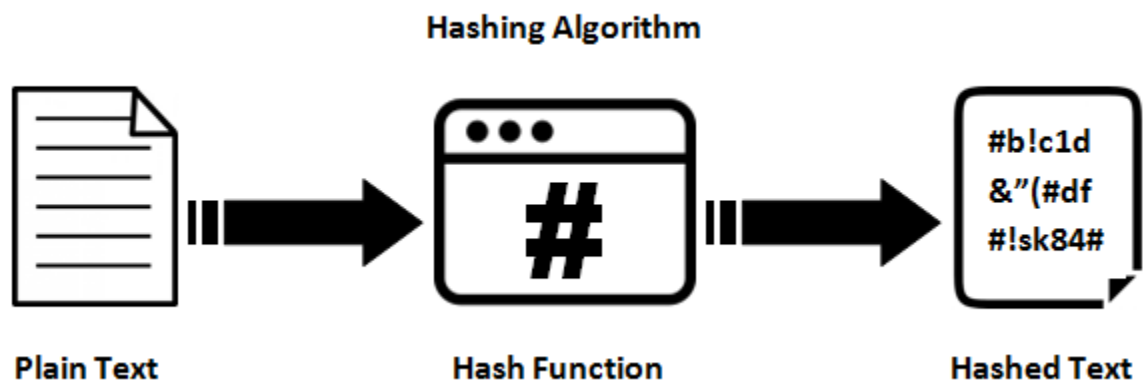
Dynamic programming is a technique that breaks the problems into sub-problems, and saves the result for future purposes so that we do not need to compute the result again. The subproblems are optimized to optimize the overall solution is known as optimal substructure property. The main use of dynamic

programming is to solve optimization problems. Here, optimization problems mean that when we are trying to find out the minimum or the maximum solution of a problem. The dynamic programming guarantees to find the optimal solution of a problem if the solution exists.



- **SHA-256 HASHING TECHNIQUES**

SHA 256 is a part of the SHA 2 family of algorithms, where SHA stands for Secure Hash Algorithm. Published in 2001, it was a joint effort between the NSA and NIST to introduce a successor to the SHA 1 family, which was slowly losing strength against brute force attacks. The significance of the 256 in the name stands for the final hash digest value, i.e. irrespective of the size of plaintext/cleartext, the hash value will always be 256 bits.



MODULE'S WISE WITH THEIR EXPLAINATION AS USED IN THE PROJECT

Login Module:- When users access the system through Portal Direct Entry, they're considered guests until they log in. The *Login Module* is a portal module that allows users to type a user name and password to log in.

Post Or Blog:-The Blog Module allows authorized users to maintain a blog. Blogs are a series of posts that are time stamped and are typically viewed by date as you would view a journal. Blog entries can be public or private to the site members, depending on which roles have access to view content.

Contact Module:-The Contact module allows site visitors to send emails to other authenticated users and to the administrator.

Search Module:-The search module lets users search for specific content on your site. You can search both for users and for particular words. When you are on the "Content" tab of Search, you will be able to search for words appearing in the default.

Categories Moduel:- The Categories Moduel consists of the entire the categorized poem (Ex: Love, Sad, Happy etc...)

Footer Module:- The Footer Module consists of the entire footer family such as("About , T&C, Contact, Sitemap, Graviance, Report us").

Admin Module:- The Admin Module has Got the full rights to manage the manager and subscriber as well as to activate and deactivate the users and the managers. He can also view all the poems.

Manager Module:- The Manager Module has got the rights to manage only the subscribers from its own manager pannel. He can also view the poems only which he has allowed / or permitted.

Subscriber Module:- The Manager Module has got the rights to manage only the subscribers from its own subscriber pannel. He can also view the poems only which he has posted.

Cross Site Scripting(XSS Attack)

An XSS (Cross Site Scripting) attack is undeniably one of most frequent cause of web application breakdown. An XSS attack involves a hacker inserting a malicious URL into the browser to compromise the application's security. A built-in XSS filter in CodeIgniter is immediately initialized.

For the protection from an XSS or Cross Site Scripting Attack we have used the CSRF Tokens in our proposed model. CSRF (**Cross Site Request Forgery**) token is a gigantic, impossible-to-guess random number that the website will include on its own web page when providing it to the user. Each time they serve a page to anyone, it's different. The hacker won't be able to guess the token, won't be able to persuade the web browser to yield it (if the browser is functioning correctly), and thus won't be able to make a valid request, because requests with the invalid token (or no token) will be rejected by the website.

Generating the Hashing Technique

Hashing is a technique used in database management to locate needed data on the disc without requiring an index structure.

The hashing method is used to index and retrieve things in a database since searching for a specific item using a shorter hashed key rather than the original value is faster.

In our proposed model, we have used the SHA-256 Hashing technique. It is a cryptographic hash algorithm that generates a unique bit.

The Use of SOAP API

SOAP is a common communication protocol that allows programmes running on different operating systems, such as Linux and Windows, to communicate using HTTP and its XML format. Accounts, passwords, leads, and custom objects can all be created, recovered, updated, and deleted using SOAP-based APIs.

Over twenty different types of requests are available, making it simple for API developers to manage their accounts, do precise searches, and much more. After then, these can be used with any language that supports web services.

SOAP APIs take advantage of web-based protocols such as HTTP and XML, which are currently supported by all operating systems, allowing developers to simply operate web services and receive responses regardless of language or platform.

Implementation of AGILE Methodology

An agile methodology is a form of software development that is iterative. Agile technique takes a short time span of 1 to 4 weeks for every iteration. The agile development approach is tailored to meet the changing needs of the business. It distributes software more quickly and with fewer changes. In the Agile methodology, there are two distinct roles.

1. The Scrum Master - . The Scrum Master is a team leader and facilitator who assists team members in adhering to agile methods so that they can achieve their obligations and meet the needs of their clients.
2. Product Owner - They are these people from a business standpoint, the Product Owner is the person in charge of the product.

CONCLUSION

We suggested a technique in this study for modeling web applications using the CodeIgniter PHP framework. We used the model driven approach principles to create these applications in order to attain this goal. This strategy, which employs a modelling and transformation process, benefits application portability and quality while reducing cost and time. This method has proven to be more effective by allowing the creation of a comprehensive Meta model that specifies an MVC CodeIgniter application. In this we have also used all kinds of backend validation for the backend security. We have used the concepts of (DRY) Don't Repeat Yourself for the code

optimization and we have also normalized the data base till 4th DFD for minimizing redundancy (duplicate data) and to ensure only related data is stored in each table. We are using a hashing technique (SHA-256) for the password to have a secured account of the all the account holders. We are using the variable submit for each and every post that is submitted, if only a variable \$param gets the the variable submit them it will proceed or else it gives error_404. If it finds \$param as submit it will check if the post is empty or not if the post is not empty then it will proceed further. We have also used an algorithm of CSRF tokens, as well as html skip entities to get rid from XSS attack. We have introduced a new concept as when the subscriber posts their own Blogs/Poems it will go to the admin or the manager for them to accept or decline the post so that we can give more security too the data base.

References

- Subari, A., S. Manan, and E. Ariyanto. "Design of E-office system in vocational school Diponegoro University using code igniter framework." In *IOP Conference Series: Materials Science and Engineering*, vol. 801, no. 1, p. 012141. IOP Publishing, 2020.
- A R KHAN "Lunarpages. Most Popular PHP Frameworks of 2018, July 11 2018. Available at <http://lunarpages.com/most-popular-php-frameworks-of-2018/>."
- R. Das and L. P. Saikia. Comparison of procedural php with codeigniter and laravel framework. *International Journal of Current Trends in Engineering & Research*, 2(6):42–8, 2016.
- Umi Saadah Usage of server-side programming languages for websites. Available at <https://w3techs.com/technologies/overview/programminglanguage/all> Accessed: 22-Aug-2018.
- X. Li, S. Karnan, and J. A. Chishti. An empirical study of three php frameworks. In *Systems and Informatics (ICSAI), 2017 4th International Conference on*, pages 1636–1640. IEEE, 2017.
- S. Pandey. Development of a web application (idomain) in the codeigniter framework: Hosting accounts management system development. 2016.
- N. Prokofyeva and V. Boltunova. Analysis and practical application of php frameworks in development of web information systems. *Procedia Computer Science*, 104:51–56, 2017.

- KhaoulaBenmoussa, Majida Laaziri ,SamiraKhoulji,Mohamed Larbi Kerkeb,Abir El Yamami PHP has generated an explosion of PHP frameworks. There is a big debate about what the best PHP frameworks are, because the simple fact is that not all frameworks are built for everyone. 2015.
- Muhammad Irfan Khan Core PHP Vs PHP Framework,Best Practices, Nov 29 2017. Available at <http://nmgtechnologies.com/blog/corephp-vs-php-framework.html>.

❖ Appendices

Login Process

View.php

```
<form method="post" action="<?=base_url('login/mylogin/submit');?>">
```

Controller.php

```
public function mylogin($param="")
{
    //check for param is empty or not
    if(empty($param)):
        show_404();
    else:
        //check $param is submit or not
        if($param=='submit'):
```

```

        //check post is empty or not
        if(empty($_POST)):
            show_404();
        else:

$this->form_validation->set_rules('username','UserName','required|trim');

$this->form_validation->set_rules('password','Password','required|trim');
        if($this->form_validation->run()===FALSE):
            redirect('login','refresh');
        else:
            $username=$this->input->post('username');
            $password=$this->input->post('password');
            //candidates credentials
            $user_data=$this->login_model->login($username,$password);
            if(empty($user_data)):
                //show flashdata that login is unsuccessful
                redirect('login','refresh');
            else:
                //show flashdata that login is successful

redirect($this->session->userdata['user_type']."/".$this->session->userdata['user_type']."_dashbaord",'refresh');

            endif;
        endif;
    endif;
else:
    show_404();
endif;
endif;

```

```
}
```

Model.php

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Login_model extends CI_Model
{
    Privatefunctionmysession($user_id,$username,$f_name,$l_name,$user_category,$email,
    $phone,$profile_pic,$created_by,$created_by_id,$created_by_ip,$status)
    {
        $session_data = array
        (
            $user_category.'_user_id'      => $user_id,
            $user_category.'_username'      => $username,
            $user_category.'_f_name'        => $f_name,
            $user_category.'_l_name'        => $l_name,
            'user_type'                     =>
$user_category,
            $user_category.'_email'         => $email,
            $user_category.'_phone_no'      => $phone,
            $user_category.'_profile_pic'   => $profile_pic,
            $user_category.'_created_by'    => $created_by,
            $user_category.'_created_by_id' => $created_by_id,
```

```

        $user_category['_created_by_ip']    => $created_by_ip,
        $user_category['_login_at']        => date("Y-m-d H:i:s"),
        $user_category['_logged_in']       => true,
        $user_category['_status']          => $status
    );

    $this->session->set_userdata($session_data);

    return $this->session->userdata;

}

public function login($username,$password)
{
    $user_data=array();
    if(!empty($username)&&!empty($password)):
        $login_status=$this->validate_login($username,$password);
        //return $login_status;
        if(!empty($login_status)):
            $user_data=$login_status;
        endif;
    endif;
    return $user_data;
}

private function login_user_type($username,$password)
{
    $user_type="";
    if(!empty($username)&&!empty($password))

```

```

        {
            $credential
            array('username'=>$username,'password'=>md5($password));
            $user_arr=array('admin','manager','subscriber');
foreach($user_arr as $table):
    $my_data_arr=$this->db->get_where($table, $credential)->row_array();
    if(!empty($my_data_arr)):
        $user_type=$table;
    endif;
endforeach;
    }

    return $user_type;
}

/**This function is used to validate login and for creating session */
private function validate_login($username,$password)
{
    $my_session_data=array();

    if(($username!="") && ($password!="")):

        $get_user_type = $this->login_user_type($username,$password);

        //return $get_user_type;exit();

        if(!empty($get_user_type)):

            $credential = array('username' => $username, 'password'
=>md5($password), 'pwd' => $password, 'status'=>'active');

            $my_query= $this->db->get_where($get_user_type, $credential);

            if($my_query->num_rows()>0):

                $row = $my_query->row();

```

```
$mysession_data=$this->mysession($row->id,$row->username,$row->f_name,$row->l_name,$
get_user_type,$row->email_id,$row->phone_no,$row->profile_pic,$row->created_by,$row->cre
ated_by_id,$row->created_by_ip,$row->status);
```

```
endif;
```

```
endif;
```

```
endif;
```

```
return $mysession_data;
```

```
}
```

```
}
```

SEARCH POST

Controller.php

```
public function search_posts($param="")
```

```
{
```

```
if(empty($param)):
```

```
show_404();
```

```
else:
```

```
if($param=='submit'):
```

```
if(empty($_POST)):
```

```
$this->common_home();
```

```
else:
```

```
$this->form_validation->set_rules('key','Search
Item','required|trim');
```

```
if($this->form_validation->run()===FALSE):
```

```
$this->common_home();
```

```

else:

    $search=$this->input->post('key');

$post_data=$this->home_model->search_posts($search);

    if(empty($post_data)):

        $data['poems']=array();

    else:

        $data['all_settings']=$this->get_my_single_array_from_multi_array($this->home_model->get_all_settings(),'content_type','content_description');

        $data['latest_news']=$this->home_model->get_latest_news();

        $data['latest_poem']=$this->home_model->get_latest_poem();

        $data['poem_count']=$this->home_model->get_poem_count();

        $data['poems']=$post_data;

        $this->search_view($data);

    endif;

endif;

endif;

else:

    show_404();

endif;

endif;

}

```

Model.php

```

public function search_posts($search_item=FALSE)

{

```

```

$my_res_arr=array();

$this->db->like('blog_id', $search_item,'both');

$this->db->or_like('blog_title', $search_item,'both');

$this->db->or_like('blog_slug', $search_item,'both');

$this->db->or_like('blog_content', $search_item,'both');

$this->db->or_like('blog_type', $search_item,'both');

$query = $this->db->get('blog_post');

$res=$query->result_array();

if(!empty($res)):

foreach($res as $r):

    $r['comments']=$this->get_all_comments_by_blog_id($r['blog_id']);

    $r['likes']=$this->get_all_blog_like_by_blog_id($r['blog_id']);

    $my_res_arr[]=$r;

endforeach;

endif;

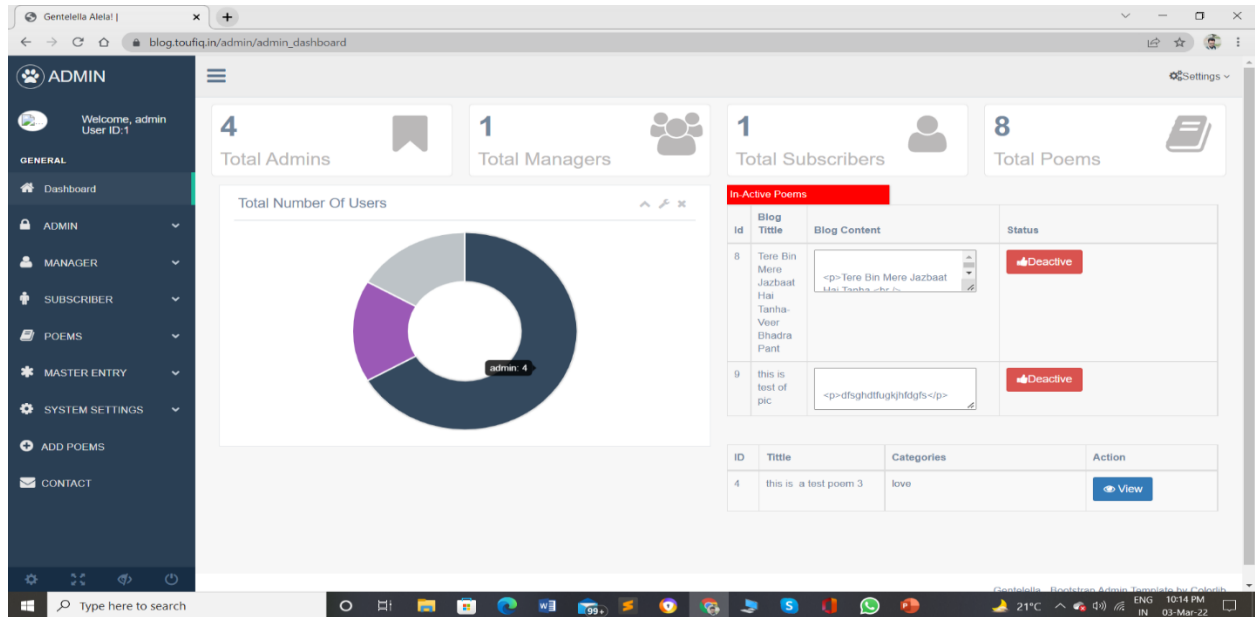
return $my_res_arr;

}

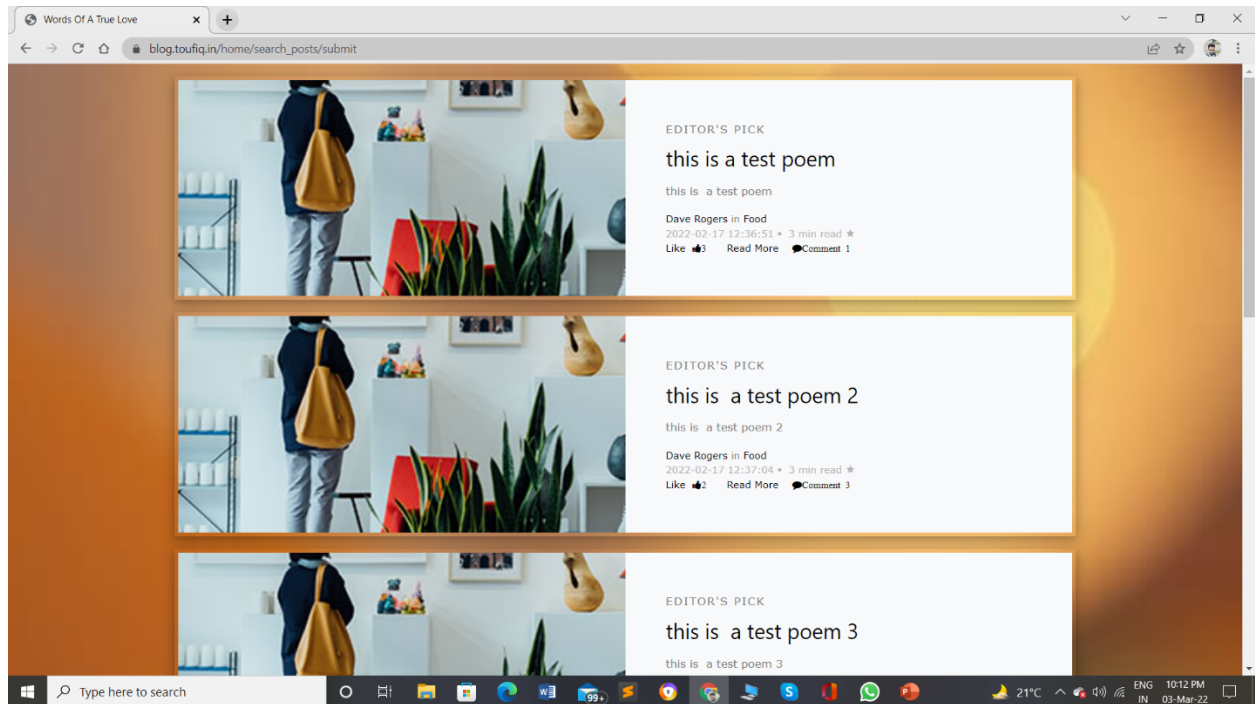
```

ADMIN DASHBOARD

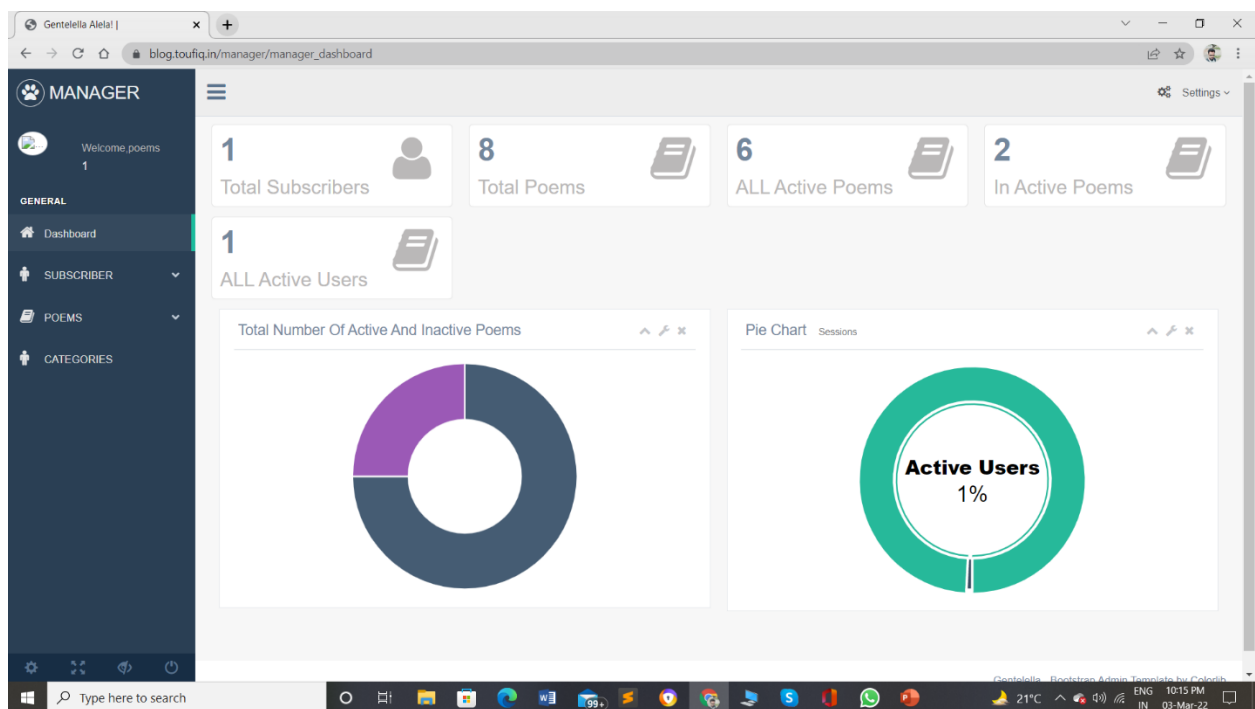
FRONTVIEW



Search View



Manager Dashboard



SUBSCRIBER DASHBOARD

