

Virtual Algorithm Vault: A Web Application

BY

Nazmus Sakib

ID: 181-15-10817

AND

Jubayer Ahmmud Shuvo

ID: 181-15-10807

AND

Md. Saharia Khan Mehedi

ID: 181-15-10889

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Saiful Islam

Sr. Lecturer

Department of CSE

Daffodil International University

Co-Supervised By

Dr. Sheak Rashed Haider Noori

Associate Professor & Associate Head

Department of CSE

Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

JANUARY 2022

APPROVAL

This Project titled “**Algo Vault**”, submitted by Nazmus Sakib(181-15-10817) and Jubayer Ahmmud Shuvo(181-15-10807), and Md. Saharia Khan Mehedi(181-15-10889) to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on ***date***.

BOARD OF EXAMINERS



Chairman

Dr. S.M Aminul Haque

Associate Professor and Associate Head

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University



Internal Examiner

Naznin Sultana

Assistant Professor

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University



Internal Examiner

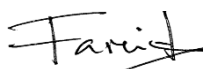
Raja Tariqul Hasan Tusher

Senior Lecturer

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University



External Examiner

Dr. Dewan Md. Farid

Professor

Department of Computer Science and Engineering

United International University

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Name, Designation, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:



Saiful Islam

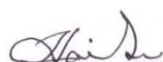
Senior Lecturer

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University

Co-Supervised by:



Dr. Sheak Rashed Haider Noori

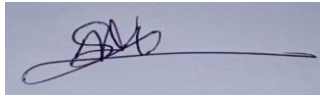
Associate Professor and Associate Head

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University

Submitted by:

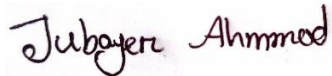


Nazmus Sakib

ID: -181-15-10817

Department of CSE

Daffodil International University

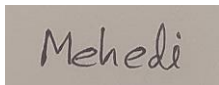


Jubayer Ahmmad Shuvo

ID: -181-15-10807

Department of CSE

Daffodil International University



Md. Saharia Khan Mehedi

ID: -181-15-10889

Department of CSE

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to Almighty God for His divine blessing making us possible to complete the final year project/internship successfully.

We are really grateful and wish our profound indebtedness to **Saiful Islam, Sr. Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Field name*” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to **Supervisor, Saiful Islam, Sr. Lecturer**, and Head, Department of CSE, for his kind help to finish our project and also to other faculty members and the staff of the CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discussion while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

In this digital era with easy access to the internet, everything is easy. If you want to learn anything from the internet you can do it with the help of the web application accessing them via the internet. Nowadays web applications play a vital role in our day-to-day life. But for programmers, especially beginners, need the help of other experienced persons to guide them. This sometimes leads to the downfall of many beginners for lack of proper or any guidance at all. The developed system is a web application aimed at helping programmers learn and practice data structures and algorithms. To guide the learners on what to learn, we have developed a web application. The front-end was implemented using HTML5, CSS3, and Bootstrap. On the other hand, Python, Django, sqlite3 was used in the back-end. In addition, a mobile application will be developed in the future. To guide the student AI is used in the web application. This system makes students' lives easier by guiding them even if he/she is not from the CSE background. Trainers from different regions can upload their tutorials for the students. This project will help the students to start their programming journey.

TABLE OF CONTENTS

CONTENTS

PAGE

Board of examiners	i
Declaration	ii-iii
Acknowledgments	Iv
Abstract	v

CHAPTER

Chapter 1: Introduction

1.1 Introduction	1
1.2 Motivation	2
1.3 Objectives	2
1.4 Expected Outcome	3
1.5 Report Layout	3-4

Chapter 2: Background

2.1 Introduction	5
2.2 Related Works	5-7
2.3 Comparative studies	7-8
2.4 Scope of the problem	8
2.5 Challenges	8-9

Chapter 3: Requirements Specification

3.1 Business Process Model	10-11
3.2 Requirement Collection and Analysis	11-12
3.3 Use Case Model	12-13
3.4 Use Case Description	13-18
3.5 Entity Relationship Diagram	18-19
3.6 Logical Data Model	19-20
3.7 Design Requirements	20-21

Chapter 4: Design Specification

4.1 Front-End Design	22
4.2 Back-End Design	23
4.3 Interaction Design and UX	24
4.4 Implementation Requirements	24-25

Chapter 5: Implementation and Testing

5.1 Implementation of Database	26
5.2 Implementation of Front-End Design	26-34
5.3 Implementation Interaction	36
5.4 Testing Implementation	36-37
5.5 Test Results and Report	38

Chapter 6: Conclusion and Future Scope

6.1 Discussion and Conclusion	39
6.2 Scope of Future Development	40

References	41
-------------------	----

APPENDICES

Appendix A: Project Reflection	42
--------------------------------	----

List of Figure

Figure 2.2.1: Related Work CP-Algorithm Page	6
Figure 2.2.2: Related Work ShafaetsPlanet page	7
Figure 3.1.1: The Business Process Model for the Proposed Project	10
3.3.1: The Use Case Diagram for the System	12
Figure 3.5.1: Entity Relationship Diagram	19
Figure 3.6.1: Logical Data Model of the system	20
Figure 5.1.1: Database Table	26
Figure 5.2.1: Register Page	27

Figure 5.2.2: Login Page	28
Figure 5.2.3: Home Page	28
Figure 5.2.4: Profile Page	29
Figure 5.2.5: Tutorial Page	29
Figure 5.2.6.1: Tutorial Details Page	30
Figure 5.2.6.2: Comment Section Page	30
Figure 5.2.7: About Us Page	31
Figure 5.2.8: Contact Us Page	31
Figure 5.2.9: Approved Post Page	32
Figure 5.2.10: Pending Post Page	32
Figure 5.2.11: Review Post Page	33
Figure 5.2.12: Create Post Page	33
Figure 5.2.13.1: Doughnut Chart Page	34
Figure 5.2.13.2: Recommendation Page	34

List of Table

Table 3.4.1: Use case Description of New Registration	13-14
Table 3.4.2: Use case Description of Login	14
Table 3.4.3: Use case Description of Forgot Password	14-15
Table 3.4.4: Use case Description of Update Profile	15
Table 3.4.5: Use case Description of Contact Us	15-26
Table 3.4.6: Use case Description of Submit Tutorial	16
Table 3.4.7: Use case Description of Review Tutorial	16-17
Table 3.4.8: Use case Description of Delete Tutorial	17
Table 3.4.9: Use case Description of Read tutorials	17
Table 3.4.10: Use case Description of Comment	18
Table 5.4.1: Testing Implementation	35-37

CHAPTER 1

INTRODUCTION

1.1 Introduction

In the path of programming and problem solving, mentors play a huge role for their mentees. In the beginning, a new programmer wouldn't always know what to do or how to plan their learning paths. That is where mentors come as saviors. But it isn't always easy to find a good and experienced mentor. Depending on the mentor, a programmer may become very good and go far or they might give up midway being demotivated and depressed. That is why we decided to build a system that would act as a mentor for programmers and will always be available to everybody. As such lowering the dependency on mentors for beginners.

In this proposed project, we have a few sections. One of them is the admin section. admins can modify, add, delete and review the contents of the web application. There will be a superuser who can add or remove admins. Admins can modify their personal details.

Another is for normal users. They can read tutorials, edit their personal info and get guidelines. All users can add tutorials and some problems for those tutorials. Admins will review them and if suitable, these tutorials will be added to the website.

The tutorial section is where all the tutorials can be found. There will be some problems to be solved for all tutorials. Each tutorial will have relevant topic-related tags. There will be a comment section for each tutorial and readers can add comments there.

Another section will be for recommendations. Recommendations will be made based on the users' previously learned topics and solved problems. AI will be used for this.

In short, in this system, users will add tutorials on topics they are fluent in. Read tutorials on different topics and get problem suggestions on those topics. They will also get recommendations on what to learn next based on their past activities. the AI will ensure that the recommendations are as best as possible.

1.2 Motivation

Firstly, during our second semester of university when we started programming, we didn't know how to start or after learning a topic what to do next. We struggled a lot until we got mentors.

Secondly, after getting mentors, we became too dependent on them. But they couldn't always spare time for us as they were busy themselves and we felt lost at those times.

Thirdly, I've seen many people stopping midway because they didn't have mentors and didn't know how to move forward from a certain point. As a result, they started trying out random topics and eventually gave up.

Fourthly, I've also seen people who didn't get proper guidelines or got poor guidelines from inexperienced persons. Which eventually led to them giving up.

Finally, after facing all these circumstances, we felt it would have been better if there were a system that could guide us properly. We searched the Internet for solutions. But we didn't find any such system. So, we decided to try to build such a system and started to study and research about it. After completing our initial study, we saw that it was possible to build such a system and started working on it.

1.3 Objectives

In this project, our main objective is to develop a system for programmers which will guide them as a mentor on what they should learn after one particular topic. Some specific objectives are needed for this project.

- It's very important to analyze the system. We need to collect data, identify the system requirements and evaluate this data and requirements.
- Next is to design the model for the learning system. This is a web-based application. So, firstly design the Front-end part for user interaction and next is to implement the Back-end using python and Django framework.
- Next is to add tutorials and relevant problems to the website and train the AI.
- The last part is to test and validate the system to ensure that it runs successfully.

1.4 Expected Outcome

- In every learning process everyone needs a mentor who can guide them. This platform will guide them as a mentor which will help them fulfill their need of a mentor.
- In this web application there will be an auto-guidance system that makes students' learning process easy and smooth. The auto system will tell them what to learn next after completing one lesson.
- There will be a lot of tutorials for the students. After completion of a particular tutorial, there will be some problem sets that should be solved by the students.

1.5 Report Layout

Report Layout means specific information about every chapter of the project. As our project is a development project, we will describe our project in six chapters. These six chapters are Introduction, Background, Requirement Specification, Design Specification, Implementation, Testing, and the last chapter is the conclusion and future scope.

In the First Chapter, we just described the total view of our project including Introduction, Motivation, Objectives, Expected Outcome, and Report Layout.

In the Second Chapter, illustrates information about the background of our project like related works, Comparative Studies, the Scope of the problem, and what kinds of challenges we were faced to create the routine.

In the Third Chapter, Describes some requirements and functionality of our project. The business process model and use case modeling were described how to work the routine exactly. In addition, requirement collection and design requirements were specified which types of data needed for this project.

In the Fourth Chapter, Design Specification includes Front-end Design, Back-end Design, Interaction Design, and UX and Implementation Requirements. The types of language used in the Front-end and Back-end will also be described in this section.

In the Fifth Chapter, after completing the Front-end and Back-end design then the next part is implementation and testing. How to implement using a database and how

to user interaction with the system are given here. In addition, the testing result and reports are also viewed in this section.

In the Sixth Chapter, which is the last part of this project including two sections- Conclusion and Future Scope. The future scope section describes in detail the features that can add in the future with the system.

CHAPTER 2

INTRODUCTION

2.1 Introduction

Almost every CSE student wants to learn problem-solving. But things are complicated for the newcomers. Because in the path of learning it's hard to get a mentor. And we all know that if someone had a mentor for them it would be of great help for them to learn. For that reason, we are developing a web app where the students will get the guidance of a mentor and also get recommendations on what to do next.

In our system, there are several sections. One of them is the admin section. Admin has the superpower. Admins can modify, add, delete and review the contents of the web application. There will be a superuser who can add or remove admins. Admins can modify their personal details.

Basically, most people think that programming is only for science background students and those who are from computer science backgrounds are able to write code in different programming languages. But this boundary is broken and from any kind of background, people can learn programming and if we look at different IT companies there are a lot of people working in the industry from non-science backgrounds.

Even many university students lost their path on programming for the lack of a mentor. If they use our website, they can easily get mentoring on what to do after finishing one chapter. Even the user can upload a tutorial on a specific topic and then some problem will be given which will be solved by the learners.

The automated system will play a vital role in our website. It will help the students most who want to learn to solve problems. In this system, with proper guidance, their programming journey will be much easier.

2.2 Related Works

This work might be complicated in many steps collecting information, calculating their track, maintaining the tutorials, and the problems based on the tutorials. For this process

to implement the system the requirements are too complicated. Because there's a lot of educational websites or programming websites to learn to program. But the problem they face most while learning programming is that after watching a full tutorial or reading a full blog what to learn next and which problem to solve based on their learning.

For example,



Figure 2.2.1: Related work CP-Algorithm page

link: <https://cp-algorithms.com>

This site is in English but there's no suggestion for the user. cp-algorithms is a site that contains English-translated tutorials. The original site is emaxx.com which is in Russian. this site is in English and it doesn't have any recommendation system.



Figure 2.2.2: Related work Shafaetsplanet page

link: <http://www.shafaetsplanet.com/?p=1437>

This site is in Bangla but they also don't give any suggestions so users who are here need to learn things randomly which is a disadvantage for them.

And because of this confusion, they lost track and they didn't continue their practice. As a result, they are out from learning programming. For solving this problem, we made a system that will guide them as a teacher and keep them on track as a mentor in their learning process. In this way, it will help them in their career development. And because of this confusion, they lost track and they didn't continue their practice. As a result, they are out from learning programming. For solving this problem, we made a system that will guide them as a teacher and keep them on track as a mentor in their learning process. In this way, it will help them in their career development.

2.3 Comparative Studies

We know that every system has some quality and some drawbacks. Our project named "Algo Vault" is a learning system. In our project, there's an automated system for guidance but also a manual system available for the admins and superusers. Because

reviewing tutorials should be handled manually and the tutorial quality should be checked by the admins.

Our project is better than others for some reason. Here users not only get the benefit of a mentor but also can practice for their problems and in the comment section they can give their lesson-related problems and someone who is better in that can answer this in the comment section.

There are many facilities in our project for the users. The admin team will always be concerned about the data of the user and always look at the quality of the tutorials which will be provided for the learners. The learners can easily find the problems based on their activity and can practice a lot of problems according to the activity. They can also participate in various programming contests based on their learning.

2.4 Scope of the Problem

The project “Algo Vault” is for learners who want to learn programming with proper guidelines. This will be helpful for every university student who is with a computer science background as well as non-science background students. Because our proposed system is very easy to use for the user to keep them on track. Easy to find the topic of what they want to learn and what to learn next the automated system will tell them. There’s also the manual system in our system for the admin panel that can edit, delete, and modify the system. However, our project is web-based, so we keep the user interface simple for all the users so that they can operate them easily. We think that our proposed system will help the other educational websites to make their sites automated.

2.5 Challenges

Our proposed system is for auto-guidance as a mentor in the learning process. To develop such a system, we need to go through a lot of difficulties. Auto guiding a student on what to learn next after learning a previous topic is tough enough. Our system collects the data that is learned by the user and based on the user activity it gives suggestions on what to do next. Some challenges are -

- Designing easy to use UI for the users
- Learning to create responsive web pages
- Learning Python and Django framework for backend
- Collecting and analyzing data of the users
- Making sure users get proper suggestions
- Making sure that the same category suggestions do not always appear and it is properly distributed among all the categories

CHAPTER 3

REQUIREMENTS SPECIFICATION

3.1 Business Process Model

The elaboration of BPM is Business Process Model. Business Process Model refers to the graphical user interface. It indicates how the process exactly works.

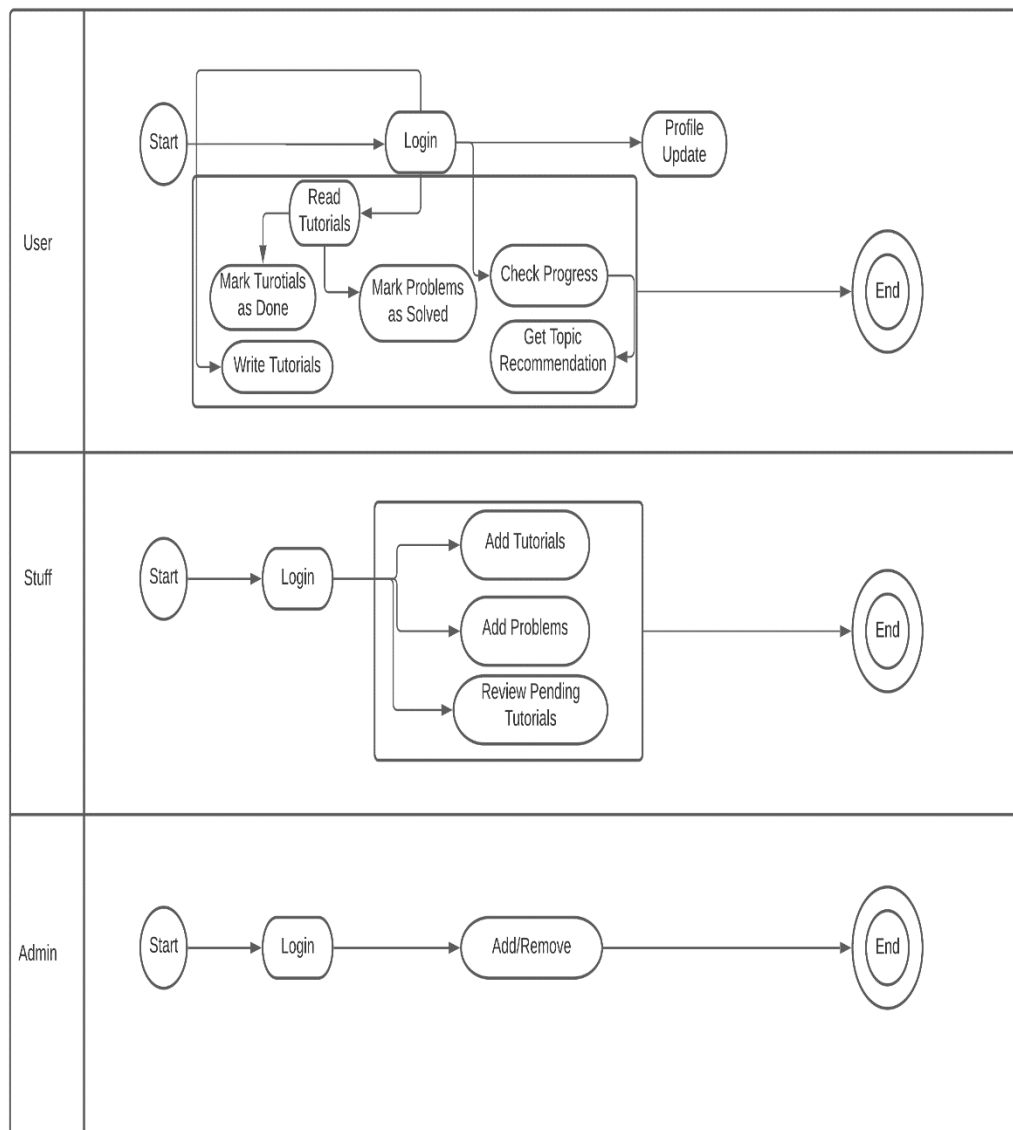


Figure 3.1.1: The Business Process Model for the Proposed Project

BPM provides you with a clear thought about your idea. BPM likewise gives you the consistency of your application. It reduces the wasteful capacity. BPM helps the project to decrease the cost, increment speed, and increment the nature of the process.

In the BPM, every process started at the start node and finished the process at the end node. In this figure 3.1.1, we can see that admin, user and staff started with the start node and finished with the end node. We can definitely say that Admin can maintain the full process, and staff can add tutorials/problems and also review the pending tutorials. On the other hand, the user can read tutorials, mark tutorials done also with mark problems as solved. The user also can check their progress and get their recommendation.

Therefore, Business Process Model easily describes the full project in a simple graphical interface.

3.2 Requirements Collection and Analysis

From the discussions till now, we now have a general idea about what the requirements are. They are as follows:

We need a system where a user can read and submit tutorials and get recommendations. We would also need a page to show the progress of the user. After logging in the user can access these features. And logged-out users are able to see the tutorial articles.

When a user acts as a general user, they will be able to read and post tutorials. After posting tutorials, it will need to be reviewed by staff or admin user. The user can also check his progress and update his profile.

When a staff take a general user, they can access and do everything a general user can. Besides that, staff can approve, edit or delete an article.

For recommendations, we will have to build an algorithm that can properly give suggestions to the users. For this we would have to do thorough research on topic suggestions.

Also, for controlling all these processes, we will use the Django admin panel. Which only the admin and staff has access to.

We would have to choose a development model for the application. We will use agile method for the dev process.

3.3 Use Case Model

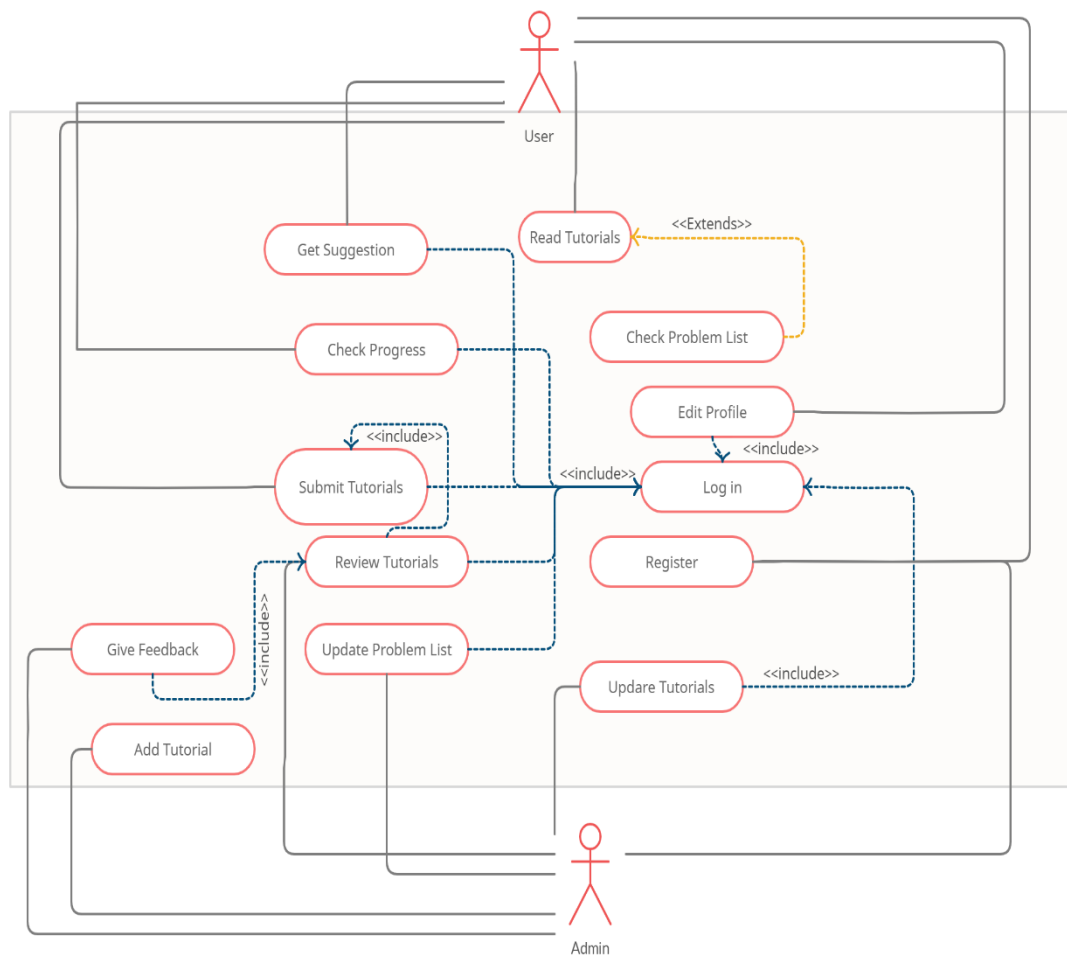


Figure 3.3.1: The Use-Case Diagram for the system

Use case diagram is one kind of scenario, which in your system or framework connects with the general people, organization, and external system. Normally users and applications in various circumstances use cases can abridge the whole project for the user easily. We can see the entire project in a solitary diagram using a use case diagram. Use case diagram helps to develop the project. The oval shape represents the use case and the stick figure represents the actor or user in the process, The proper use case diagram and activities are used to reach the actual destination.

In this project, users have to log in to save their data and progress. Without logging in they only can see some functionality that doesn't need to save. They can view what

they need after logging in and what they actually want to learn. They will get suggestions based on their activity which helps them to keep them on track and learn efficiently. Here users can add tutorials and also, they can solve the problem of others in the comment box. Admin can review the tutorials and also update the problem list. After reviewing the tutorial admin can approve the post also admin can see the pending post for approval.

3.4 Use Case Description

There is some model given in below:

Table 3.4.1: Use case description of New Registration

1	Use Case Name:	Register
2	Pre-Condition:	None
3	Actor:	All User
4	Internal Path:	4.1: Click on Register Button 4.2: Enter Username 4.3: Enter Full Name 4.4: Enter Email 4.5: Enter Password 4.6 Confirm Password 4.7: Click Submit button 4.8: Confirm email by clicking the link sent to the email
5	Exceptional Path:	4.2.1: Username contains characters other than letters and digits 4.2.2: Username already in use 4.4.1: Invalid email 4.4.2: Email already in use 4.5.1: Password must contain both letters and digits. 4.5.2: Password is less than 6 characters 4.6.1: Passwords do not match

Table 3.4.2: Use case description of Login

1	Use Case Name:	Login
2	Pre-Condition:	Register
3	Actor:	All User
4	Internal Path:	4.1: Click the login button 4.2: Enter Username 4.3: Enter Password 4.4: Click Submit
5	Exceptional Path:	4.2.1: Wrong Username 4.2.2: Wrong Password

Table 3.4.3: Use case description of Forgot Password

1	Use Case Name:	Forget Password
2	Pre-Condition:	Registration
3	Actor:	All user
4	Internal Path:	4.1: Click on the login button 4.2: Click on Forgot Password? 4.3: Enter email address 4.4: Click submit 4.5: Click the link sent to the email 4.6: Enter a new password 4.7: RE-enter new password
5	Exceptional Path:	4.3.1: Invalid email address 4.3.2: Email not registered 4.6.1: Invalid password format 4.7.1: Passwords do not match

Table 3.4.4: Use case description of New Registration

1	Use Case Name:	Update Profile
2	Pre-Condition:	Login
3	Actor:	All user
4	Internal Path:	4.1: Click on the profile button 4.2: Enter Name or leave empty 4.3: Enter email or leave empty 4.4: Enter current password 4.5 Enter a new password or leave empty 4.6: Re-enter new password
5	Exceptional Path:	4.3.1: Invalid Email 4.4.1: Incorrect Password 4.5.1: wrong password format 4.6.1: Passwords doesn't match

Table 3.4.5: Use case description of Contact Us

1	Use Case Name:	Contact Us
2	Pre-Condition:	None
3	Actor:	General user
4	Internal Path:	4.1: Click on Contact Us 4.2: Enter Name 4.3: Enter Email 4.4: Enter Subject 4.5: Enter message
5	Exceptional Path:	4.2.1: Empty field 4.3.1: Empty field 4.3.2: Invalid email 4.4.1: empty field 4.5.1: empty field

Table 3.4.6: Use case description of Submit Tutorial

1	Use Case Name:	Submit tutorial
2	Pre-Condition:	Login
3	Actor:	All user
4	Internal Path:	4.1: Click the profile icon 4.2: Click my tutorials 4.3: Click add tutorials 4.4: Enter title 4.5: Enter tags 4.6: Enter details 4.7: Add suggested problems
5	Exceptional Path:	4.4.1: Empty field 4.5.1: Empty field 4.6.1: Empty field 4.7.1: Empty field

Table 3.4.7: Use case description of Review Tutorial

1	Use Case Name:	Review tutorial
2	Pre-Condition:	Login
3	Actor:	Stuff, Admin
4	Internal Path:	4.1: Click the profile icon 4.2: Click pending tutorials 4.3: Click tutorial title 4.4: Click approve or disapprove
5	Exceptional Path:	4.3.1: Tutorial does not exist

Table 3.4.8: Use case description of Delete Tutorial

1	Use Case Name:	Delete Tutorial
2	Pre-Condition:	Login
3	Actor:	Stuff, Admin
4	Internal Path:	4.1: Click tutorials 4.2: Click tutorial title 4.3: Click delete 4.4: Enter password 4.5: Click confirm
5	Exceptional Path:	4.2.1: Page doesn't exist 4.4.1: Incorrect password

Table 3.4.9: Use case description of Tutorial Read

1	Use Case Name:	Tutorial read
2	Pre-Condition:	Login
3	Actor:	All user
4	Internal Path:	4.1: Click tutorials 4.2: Click tutorial title 4.3: Click Mark as Done
5	Exceptional Path:	4.2.1: Page does not exist

Table 3.4.10: Use case description of Comment

1	Use Case Name:	Comment
2	Pre-Condition:	Login
3	Actor:	All user
4	Internal Path:	4.1: Click tutorials 4.2: Click tutorial title 4.3: Write a comment in the comment section

		4.4: Click the comment
5	Exceptional Path:	4.3.1: Empty field

Table 3.4.1: Use case description of Recommendation

1	Use Case Name:	Recommendation
2	Pre-Condition:	Login
3	Actor:	General user
4	Internal Path:	4.1: Click the profile icon 4.2: Click my progress
5	Exceptional Path:	

3.5 Entity Relationship Diagram

The entity-relationship diagram is one kind of summary of all data structures of the system. For this system, the entity-relationship diagram represents the database that is important for this system. The relationship between tables is also shown with the entities in this diagram.

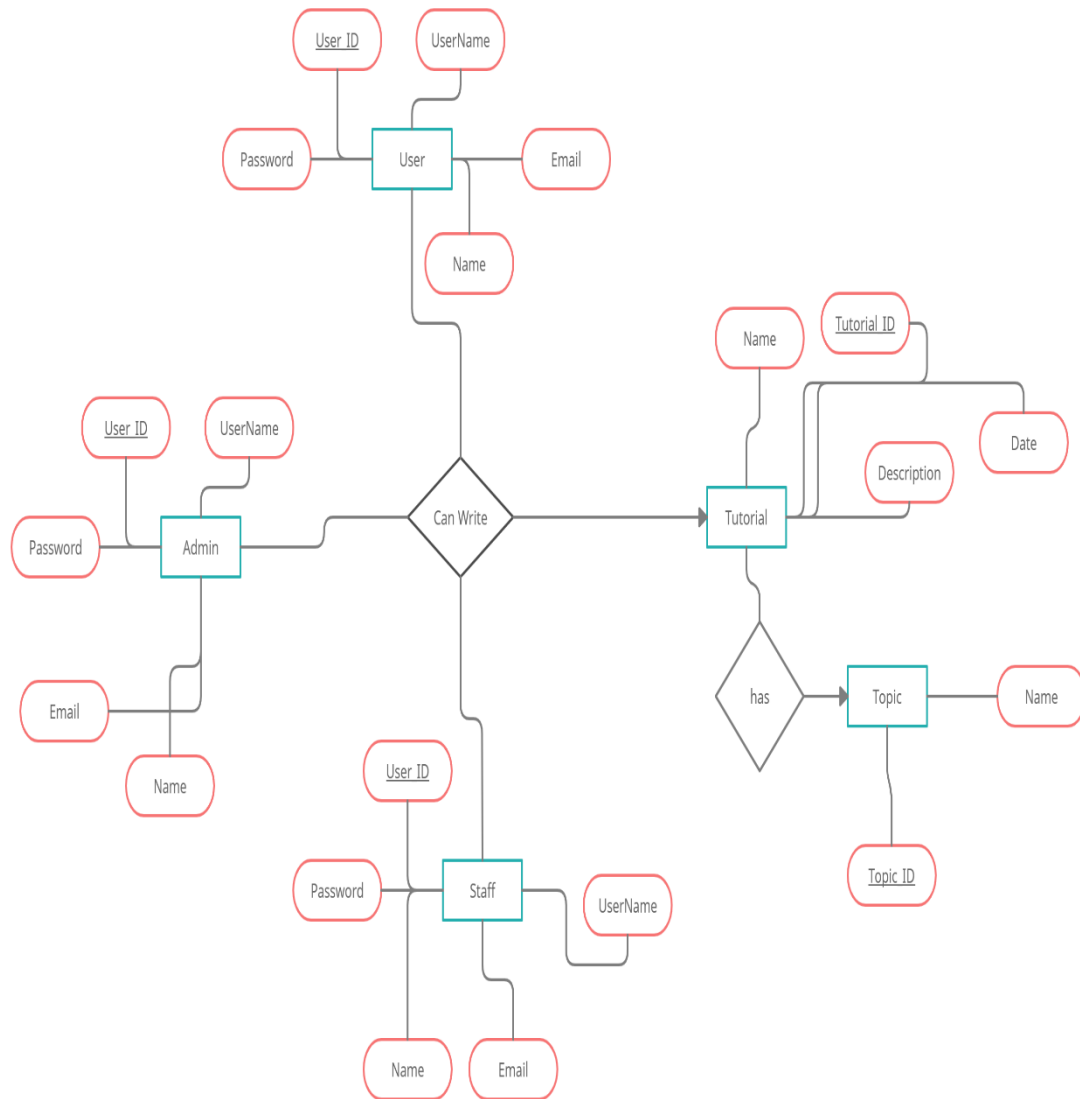


Figure 3.5.1: Entity Relationship Diagram

3.6 Logical Data Model

The logical data model is the process of representing data architecture and organization in a graphical way. The logical data model provides all the entity relations of a system. It is specific problem domain expressed independently of a particular database management product or storage technology.

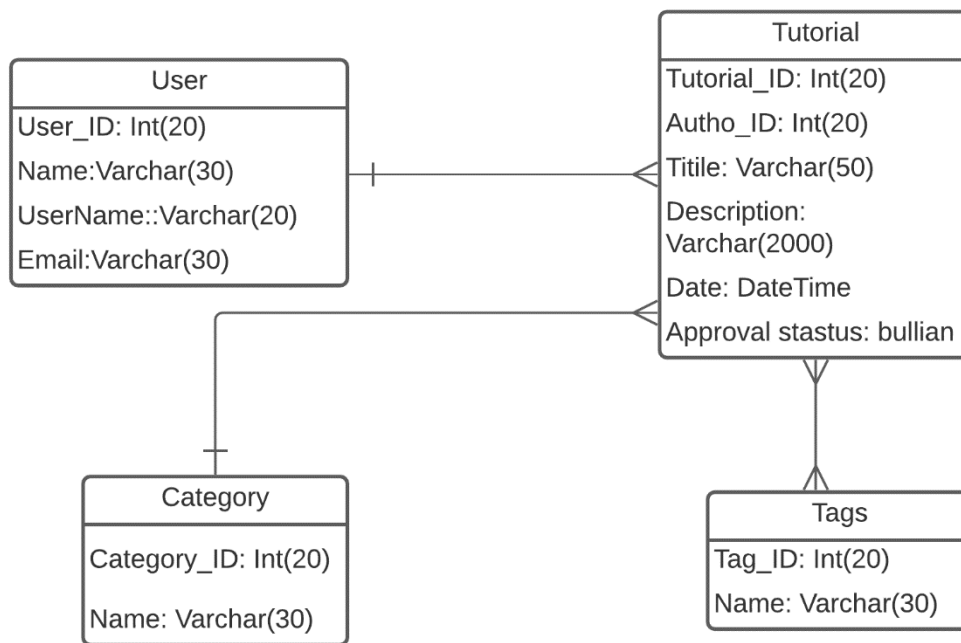


Figure 3.6.1: Logical Data Model of the System

3.7 Design Requirements

There are some requirements to generate the “Algo Vault” system. They are hardware requirements and software requirements.

Hardware Requirements:

Hardware requirements or specification means what kind of hardware is needed to operate the project.

- Processor: Platinum III or Higher.
- Ram: 512MB or more.
- Hard Disk: 500MB or more.
- Screen Resolution at least 640*480 or above.
- Required Internet connection.

Software Requirements:

Though the system is a web-based application, so the client-side is required.

- Operating System: Windows/Linux.
- Database: SQLite.
- Sublime Text for editing.
- Inter Browser for View.
- VS Code and PyCharm for editing.

Language

- HTML
- CSS
- Bootstrap
- Python

Framework

- Django Framework

CHAPTER 4

DESIGN SPECIFICATION

4.1 Front-end Design

The front-end is one kind of application that the user interacts with the application directly. Front-end converts the data to the graphical interface for the user to see and interact with the data easily.

Front-end is the viewer site of the user. Our project “Algo Vault” is one kind of web-based application. This application also has two-part. One is the front-end and another is the back-end.

The Front-end part includes some language. They are HTML, CSS, JS(JavaScript). HTML refers to the Hypertext Markup Language and CSS refers to the Cascading style sheet. On the other hand, JavaScript is one kind of scripting language.

Html and CSS are used for creating a structure for the application. HTML can derive the header, footer. In addition, how the text is displayed in the application and how the other thing on the website will appear. The front-end design tells us how easy the system is for the user and how interactive the system is.

There’s a lot of designers who are responsible for the front-end implementation whom we called front-end designer.

HTML and CSS are the main structural elements of a web. Without knowing these no one can create a website. HTML is like the foundation of a building and CSS is like how the building looks. And for more functionality in a website, we use JS.

After completing the front-end part, we just see the user interface that how the system looks, and after clicking a specific button it works properly or go that pointed page or not.

4.2 Back-end Design

The back-end is also a program that indirectly supports the front-end site. The back-end is also called the server-side. Because it helps with how the site works, updates and changes. The back-end is a program that users cannot see in the browser. It can only be modified by the administrator. Users only can use the functionality of the system but actually what is happening behind the system to get the user data accurately is only maintained in the back-end.

We have already been told that our project name is “Algo Vault” which is basically a web-based application. The application works dynamically. If one needs automation then he needs a backend server. Without a back-end system it is impossible. To run the backend system perfectly there’s a lot of programmers whom we call backend engineers.

Back-end usually consists of three parts. They are an application, server, and database. In this system, all information is stored in the database and then run by a server.

Back-end also controls the security of the application. Dynamic sites are always changing updates in real-time. For dynamic sites, a database is required to work properly.

Django is a high-level Python web framework that enables the 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.

Django helps us write software that is:

- Complete
- Versatile
- Secure
- Scalable
- Maintainable
- Portable

4.3 Interaction Design and UX

Goal-Driven Design: User research is very important to know the real scenarios and related problems. That is the actual person who used the system. What we want to achieve through our persona should be well defined by user stories. User stories give logic to our thoughts making the design perfect.

User Expectations: This is the same case when a user comes to an application and performs some actions expecting some reaction from the product. If a user fills some form and clicks 'Done', he expects some confirmation alert or an error message if something is wrong with any of the fields, he expects some alert or error message.

Usability: There should be less attention to figuring out how to use the application so that the user can focus on completing the task. The product should be very easy to use and easy to understand its functions.

Utility: It is the most important factor focusing on "Does the user find the function useful?" or "Is the function important or is it designed to give a high-quality UI factor only?" The function should solve some problem or purpose then only it is useful to the user. There should be a very logical reasoning behind the function to complete a valuable task.

Affordance and Signifier: Function that speaks for itself and suggests its own use its affordance. Signifier hints at the affordance of the particular function and suggests how to use it.

4.4 Implementation Requirements

Requirements for implementing Front-End: This system is implemented with web programming. For this, I have used VS Code to write code and the languages are HTML, CSS, Bootstrap

Requirements for implementing Back-End: To implement the database I have used SQLite database with Django.

Summary of the languages, Frameworks, and tools

- HTML
- CSS
- Bootstrap
- JavaScript
- Django
- SQLite

Chapter 5

IMPLEMENTATION AND TESTING

5.1 Implementation of Database

This is a web-based project and it stores data in a database. We have used sqlite3 for the database. We have used Django built-in models to create the database. We have used ORM to implement this database. So, for queries, we have used the built-in functions of Django. When a user logs in or registers or updates their profile, the actions are checked on the database at that time. Figure 5.1.1 shows the tables and contents of the database.

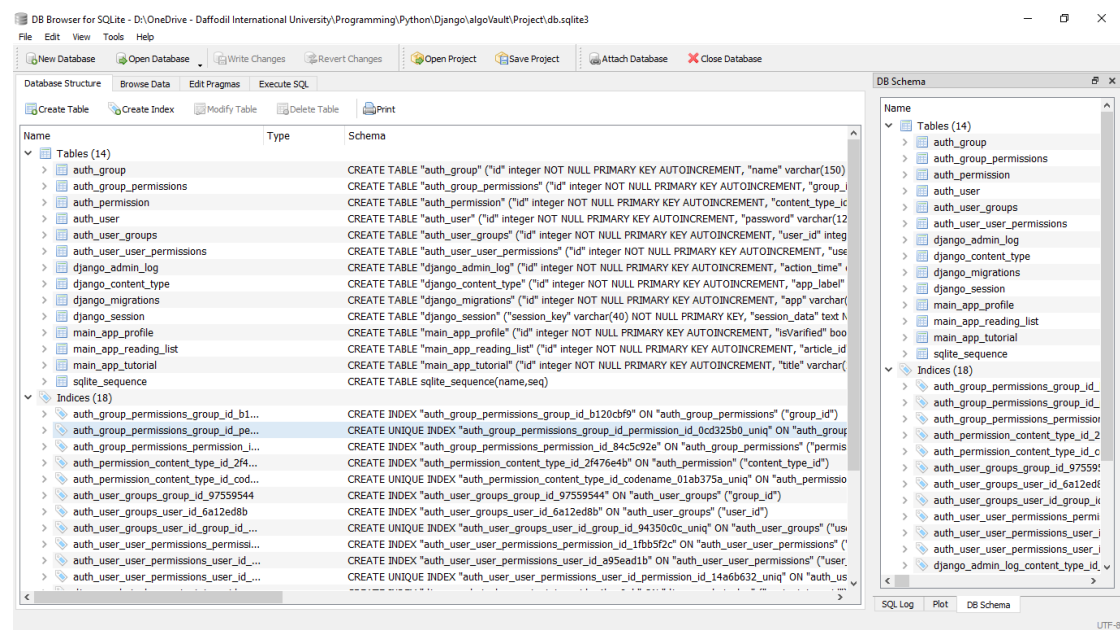


Figure 5.1.1: Database Table

5.2 Implementation of Front-End Design

Front-end design is the most important part of a web application. If the website's user interface looks critical, complex, and not user friendly then the user will not be happy to use it and the user will look out for another option. Also, page speed is very important for a web application. Nowadays everything is very fast so its very important for a web application fast as possible otherwise users will get bored and they will leave the site.

Keeping these in mind we have tried to keep out the application as simple as possible and clean. We have tried to create a user interface that is really user-friendly and responsive. It is very important too that the website is responsive because nowadays users are very diverse tile. Nowadays users are accessing websites from different types of devices. They are accessing the website from a mobile, laptop, tablet, etc. and all of these screen sizes are not the same. And the responsive site is the best solution for this problem.

- ❖ On this website, there is mainly part of users and admins.
- ❖ General users need to register first at the beginning to use the benefit of the website. And the admin has direct access to the website to control the website.
- ❖ General users can log in to the website with their registered email address and password.
- ❖ Admin needs to approve the tutorials which were uploaded.
- ❖ Admin can remove any users if he wants.

5.2.1 Register Page

Figure 5.2.1.1 shows the Registration page. This page is where a user registers for the website.

A user needs to enter a unique username, his full name, unique email and password to register for the site.

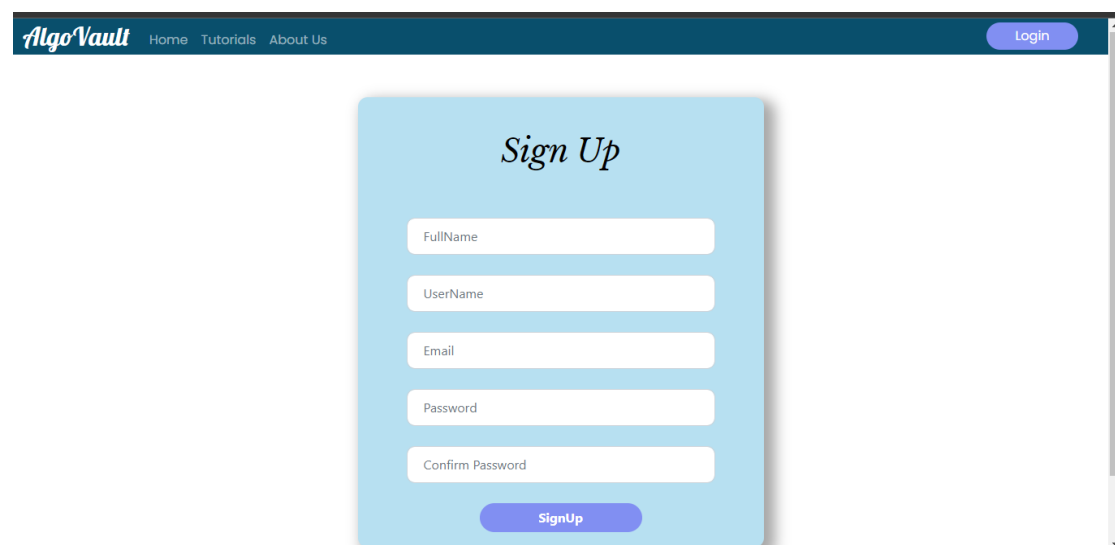


Figure 5.2.1: Register page

5.2.2.1 Login Page

Figure 5.2.2.1 shows the login page. All the users have to insert their credentials here to log in and access the features of the site.

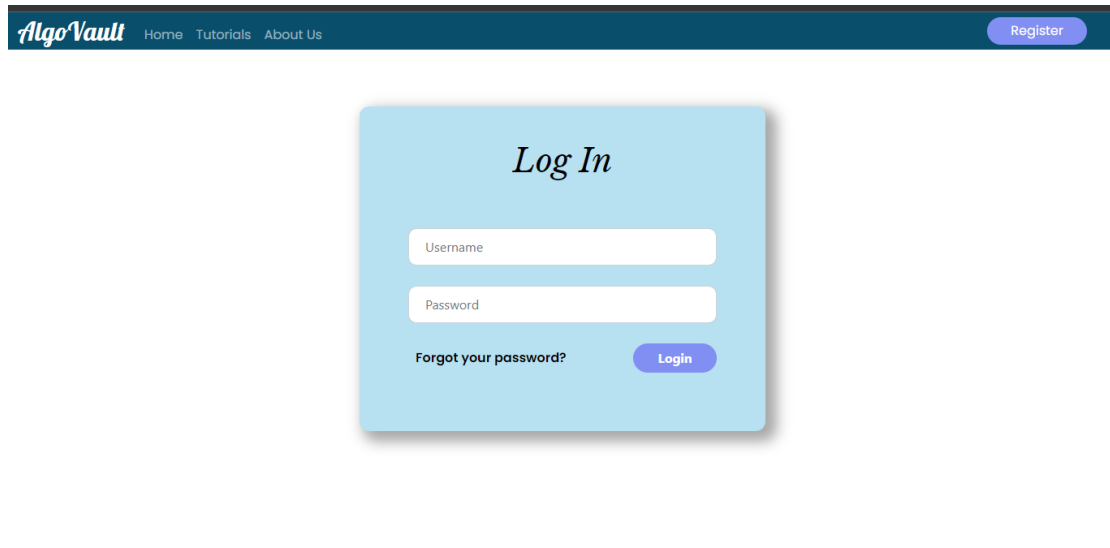


Figure 5.2.2: Login page

5.2.3 Home

Figure 5.2.3.1 shows the home page of our website. In this page people will see the latest article posts. And by clicking the more button of an article will take them to the detailed page.

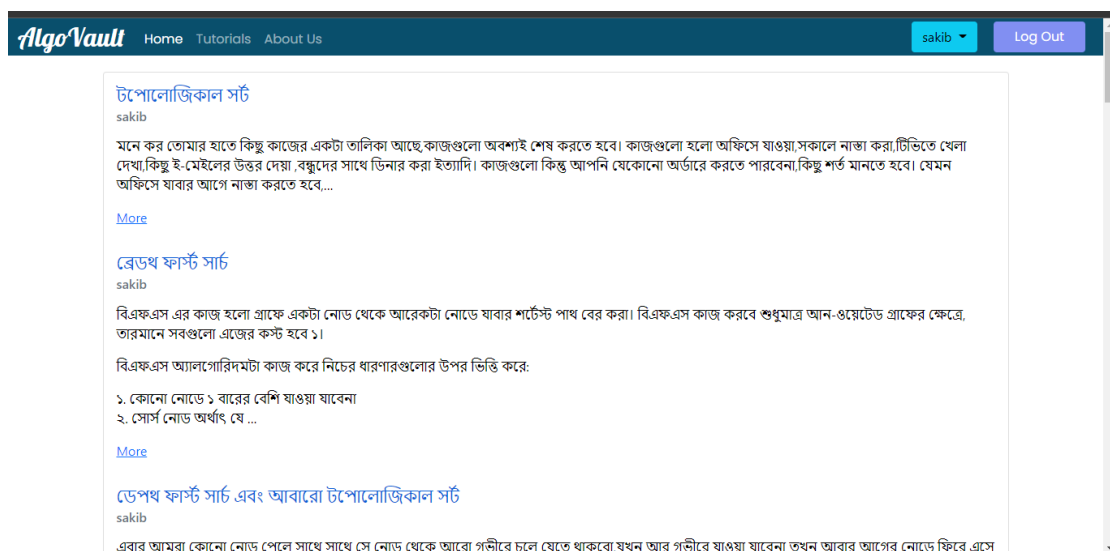
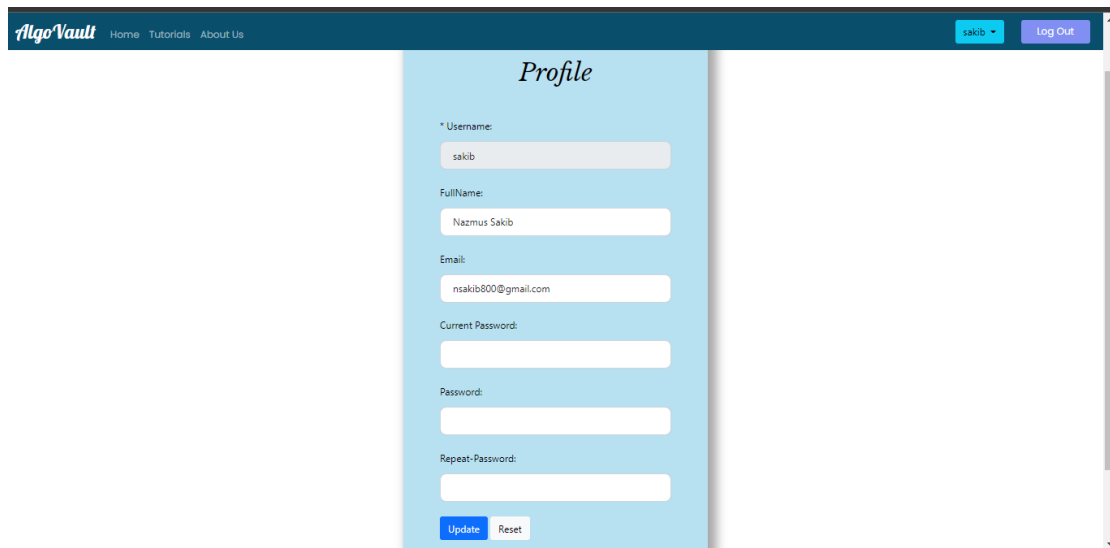


Figure 5.2.3.1: Home page

5.2.4 Profile Page

Figure 5.2.4.1 show the profile page. On this page, a user can also update their profile and password. After entering the updated information, the user would have to enter their password and click the update button. That way it will be updated



The screenshot displays the 'Profile' page of the AlgoVault application. The page features a dark blue header with the 'AlgoVault' logo, navigation links for 'Home', 'Tutorials', and 'About Us', and a user profile section showing 'sakib' with a 'Log Out' button. The main content area is a light blue vertical panel titled 'Profile' in italics. It contains several form fields: 'Username' (pre-filled with 'sakib'), 'FullName' (pre-filled with 'Nazmus Sakib'), 'Email' (pre-filled with 'nsakib800@gmail.com'), 'Current Password', 'Password', and 'Repeat-Password'. At the bottom of this panel are two buttons: 'Update' (in blue) and 'Reset' (in white).

Figure 5.2.4.1 Profile page.

5.2.5 Tutorial Page

Figure 5.2.5.1 shows the tutorial page. This is where the tutorial list can be found. If a user clicks one of the tutorial titles, they will be taken to the details page of that tutorial.



Figure 5.2.5.1 Tutorial page

5.2.6 Tutorial Details Page

Figures 5.2.6.1 and 5.2.6.2 show the tutorial details page. This is where an article is described in detail. Figure 5.2.6.2 shows the comment section of the page. An user can comment or reply to others comments here. They can also give a reaction to the article. The mark as done button is for logged in users only. They can click the button and mark the article as read.



Figure 5.2.6.1 Tutorial details page

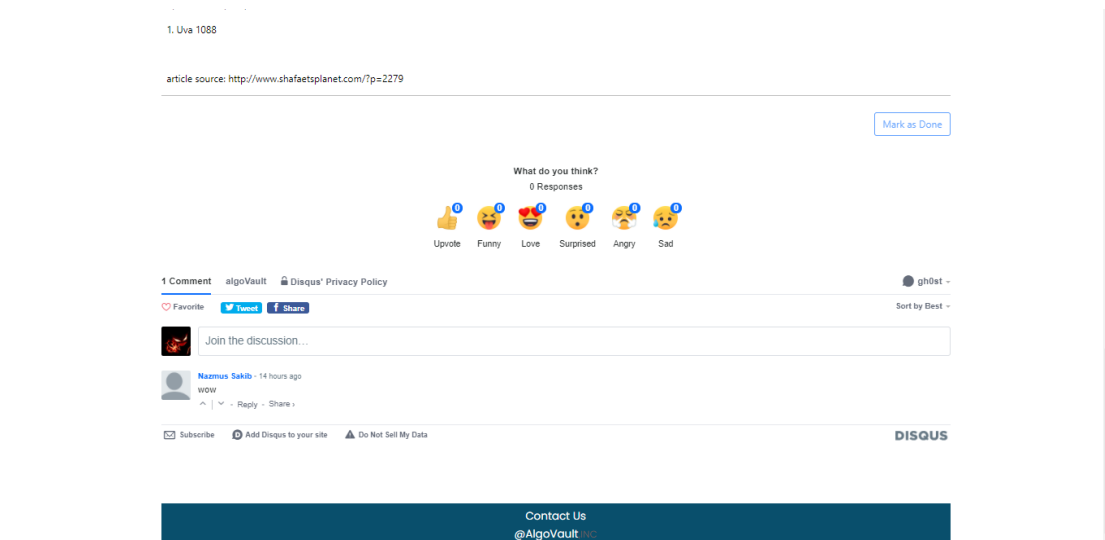


Figure 5.2.6.2 Comment section

5.2.7 About Us Page

Figure 5.2.7.1 shows the about us page. In this page, some pieces of information and contacts of the developers are shown.

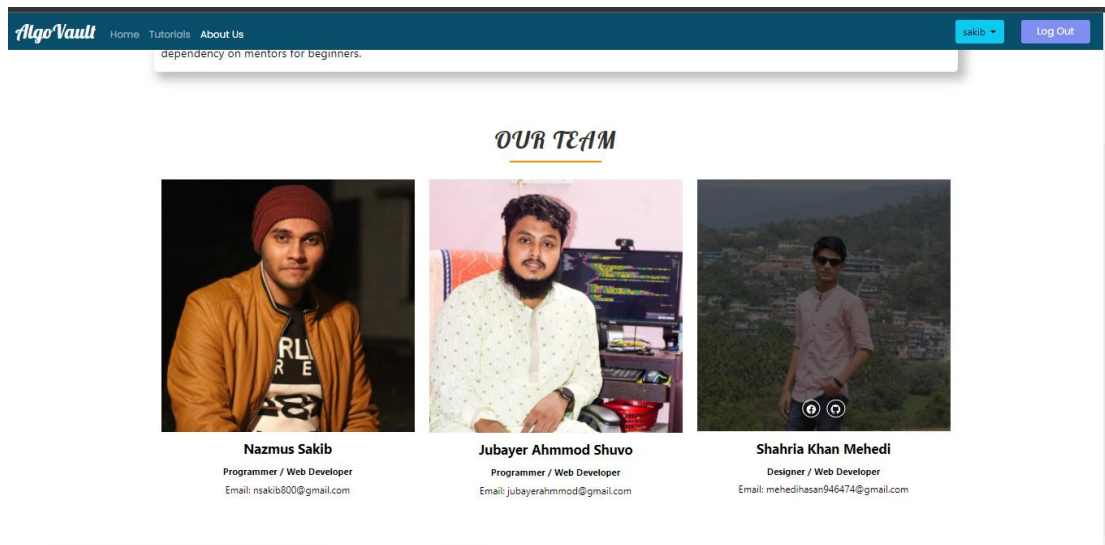


Figure 5.2.7.1 About us page

5.2.8 Contact Us Page

Figure 5.2.8.1 shows the contact page. Users can enter their details and a message for the admins and send them. Through this page they can contact the admins.

Figure 5.2.8.1 Contact us page

5.2.9 Approved Posts Page

Figure 5.2.9.1 shows the page where a user can see the posts that they have written and be approved.

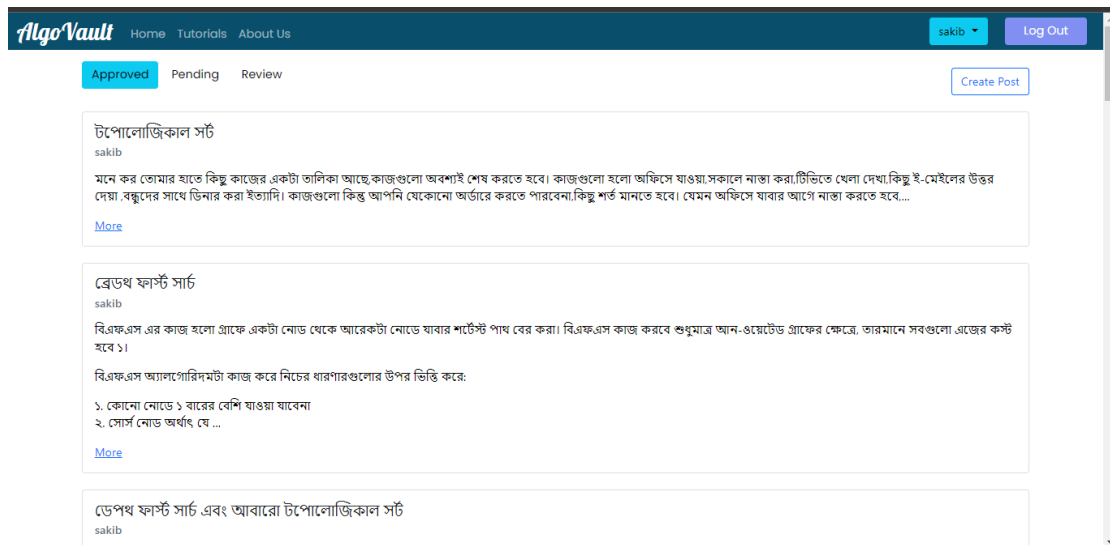


Figure 5.2.9.1 Approved posts page

5.2.10 Pending Posts Page

Figure 5.2.10.1 shows the page where a user can see the posts that they have written and are pending approval.

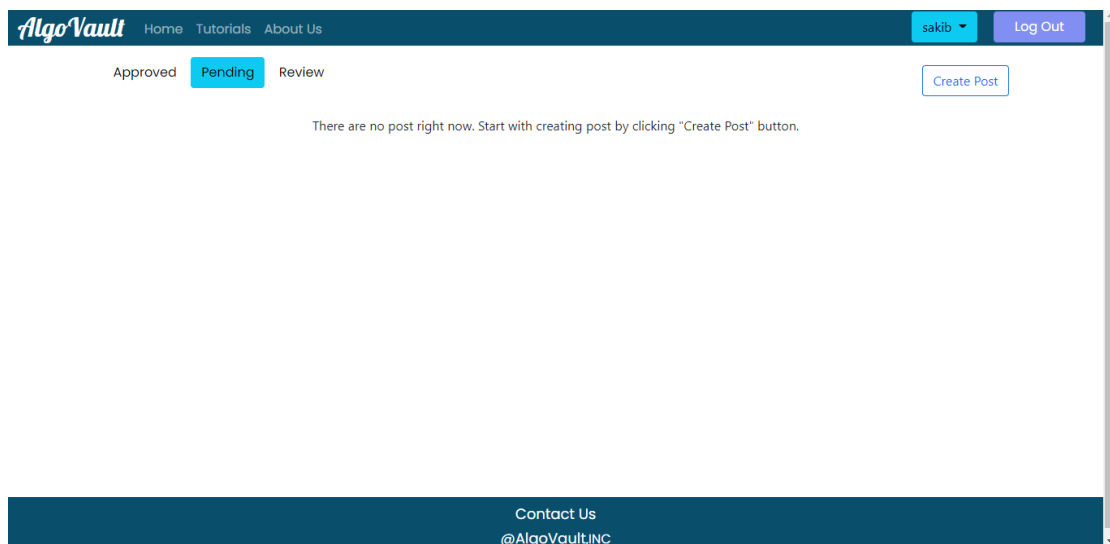


Figure 5.2.10: Pending Posts Page

5.2.11 Review Posts Page

Figure 5.2.11.1 shows the page where a user can see the posts that need to be approved. This page is only available to the admin and staff users.

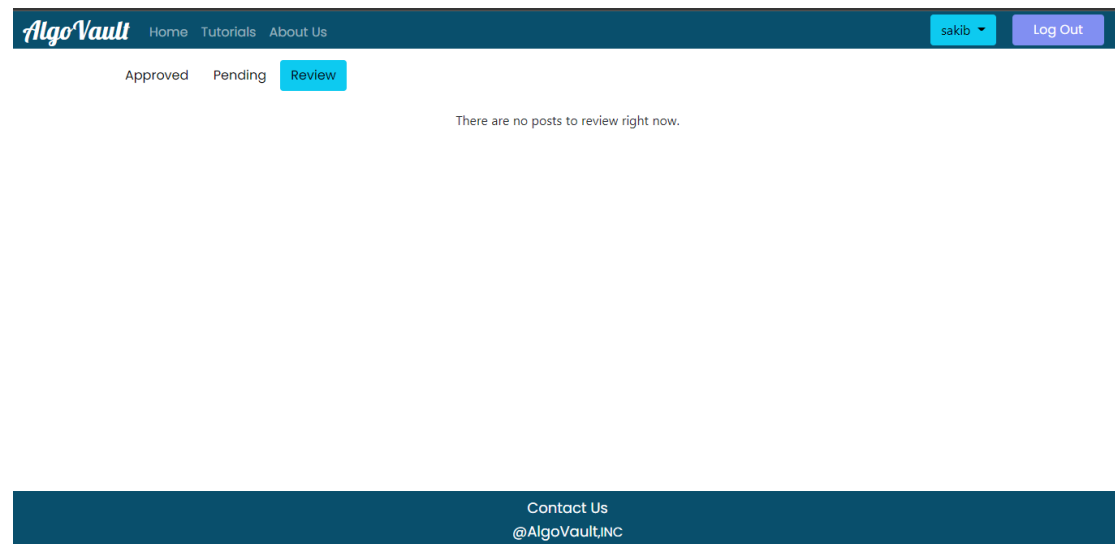


Figure 5.2.11: Review Post page

5.2.12 Create Post Page

Figure 5.2.12.1 shows the create post page. Here a user can create a new post. To create a post, they need to enter the title, select tag and level and write the article and submit it.

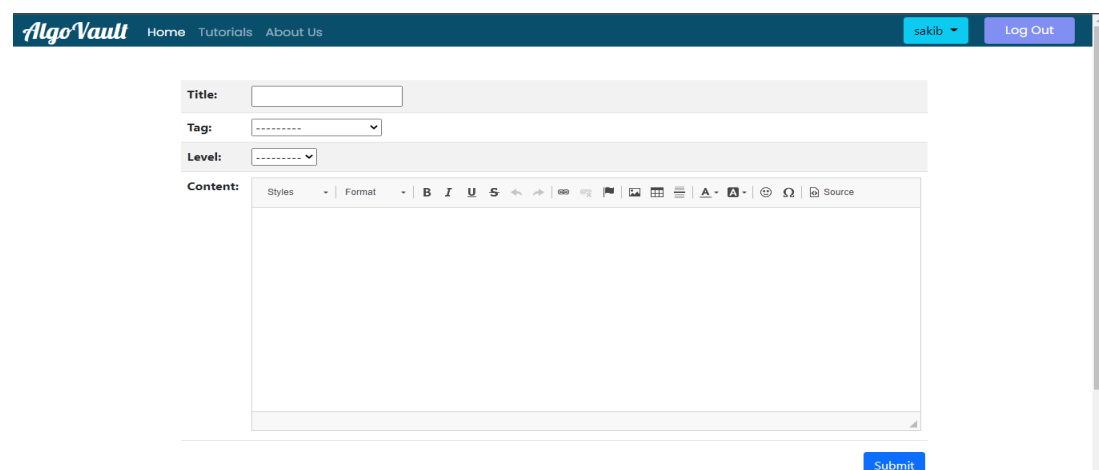


Figure 5.2.12.1: Create Post Page

5.2.13 Progress Page

Figure 5.2.13.1 shows the doughnut chart of user progress. It shows the percentage of each topic the user has covered.

Figure 5.2.13.2 shows the recommendations section. This is where an user can get recommendations for what topic to learn next.

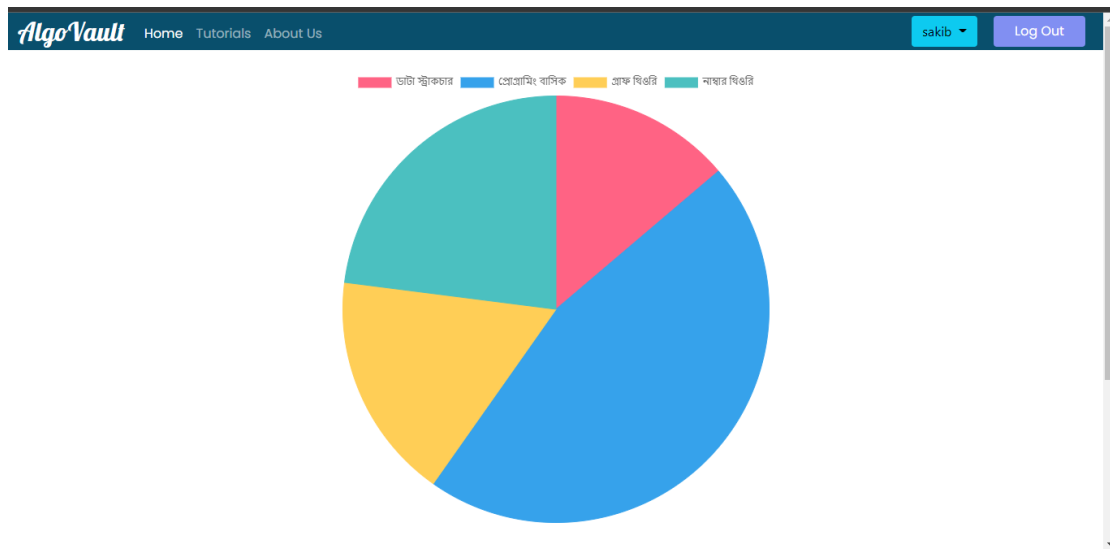


Figure 5.2.13.1 Doughnut chart

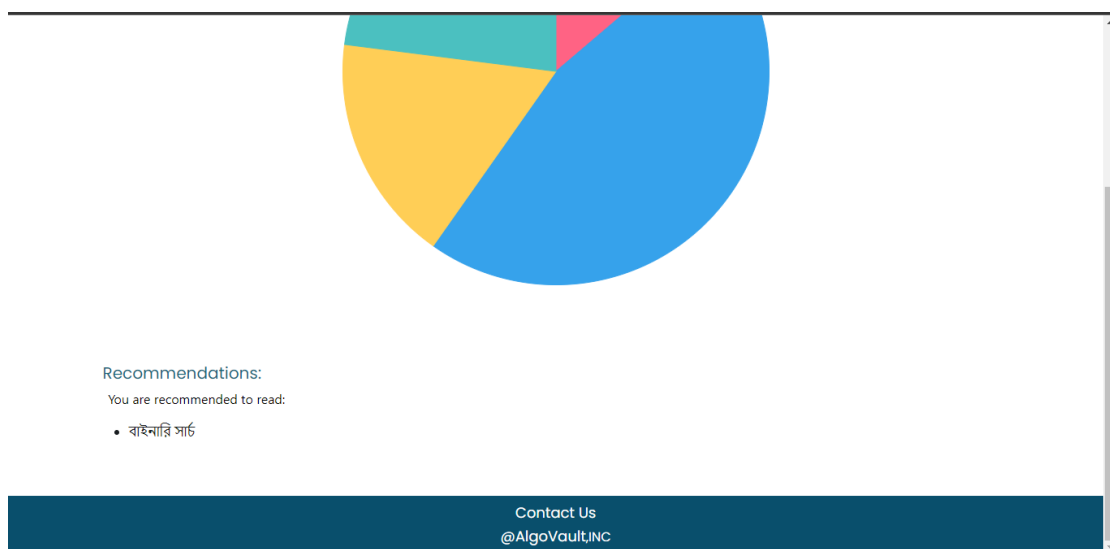


Figure 5.2.13.2 Recommendations

5.3 Implementation Interaction

We have tried to make our website as user-friendly as possible. People will like this application easily. They can find what they are looking for easily. With the interactive looks, the website is too much friend for the user.

5.4 Testing Implementation

In this section, the use case to be tested during functional testing are listed and testing steps and expected results are explained in detail.

Table 5.4.1: Testing Implementation

Test Case	Test Input	Expected Outcome	Obtained Outcome	Pass/Fail
1. User Registration	User must fill all the fields in the registration page	Registered successfully	Registered successfully	Pass
2. User Login	Users can log in via browser by username and password	Logged in successfully	Logged in successfully	Pass
3. Admin Login	Admin can login via browser by username and password	Logged in successfully	Logged in successfully	Pass
4. Staff Login	Staff can login via browser by username and password	Logged in successfully	Logged in successfully	Pass
5. Update Profile	User can update profile information	Updated successfully	Updated successfully	Pass
6. Contact Admins	User can contact admins by entering their email, subject, and message	Message sent successfully	Message sent successfully	Pass
7. View a Tutorial list	Users can see the entire tutorial list by browsing the page	The tutorial page loaded successfully	The tutorial page loaded successfully	Pass

8. View Tutorial Details	Users can see the details of a tutorial by clicking the tutorial title from the list	Details loaded successfully	Details loaded successfully	Pass
9. Comment	Users can comment on a tutorial.	Comment posted successfully	Comment posted successfully	Pass
10. React	Users can give reactions to a tutorial	Reaction posted successfully	Reaction posted successfully	Pass
11. Mark Tutorial as Read	Click Mark as done button to mark tutorial as done	Marked successfully	Marked successfully	Pass
12. Submit new Tutorial	Submit a new tutorial by entering details	Submission successful	Submission successful	Pass
13. View submitted tutorial details	View the details by clicking my posts button	Page loaded successfully	Page loaded successfully	Pass
14. Edit Post	Edit post by clicking the button and updating the posts	Post updated successfully	Post updated successfully	Pass
15. Approve Post	Approve a submitted post by clicking the button on the posting page	Post approved successfully	Post approved successfully	Pass
16. View approved Posts	View approved posts by browsing the page	Page loaded successfully	Page loaded successfully	Pass
17. View Pending Posts	View Pending posts by browsing the page	Page loaded successfully	Page loaded successfully	Pass

18. View Posts to be reviewed	View review needed posts by browsing the page	Page loaded successfully	Page loaded successfully	Pass
19. View progress	View a doughnut chart of progress	Page loaded successfully	Page loaded successfully	Pass
20. Get recommendations	Get topic recommendation of what to learn next	Got proper recommendation	Got proper recommendation	Pass

5.5 Test Results and Reports

After the project work completion, we have tested our projects many times from different devices and by different users. Our goal was to check whether the project is working fine or not. After testing several times, we were glad to say that the project is working smartly and flawlessly. A report is a paper that actually describes that test result and defines whether the system is ready to use or not.

Few test categories were performed for this project:

- Registration test successfully.
- The login system works fine.
- The profile update process is working fine.
- Website speed and loading.
- Comment posting working fine.
- Overall performance and usability.
- The admin panel works fine.
- The recommendation is working well.

Chapter 6

CONCLUSION AND FUTURE SCOPE

6.1 Discussion and Conclusion

We have developed an educational system which is called “Algo Vault” which is mainly a programming learning system. In this system, users can find too many benefits in their learning path. It is difficult for a student to keep him from derailing when learning a programming language. Because while learning it is mandatory for a learner to keep him in touch with a mentor. So, focusing on that thing we developed a system that can make the learning process for a student easier. Also, that keeps the activity track and give the user suggestion based on the previous activity on what to do next. While developing this system we keep focusing that the user can easily co-operate with the system. In this system, there's an option to upload tutorials but the tutorials must be approved by the super admin. Super admins can edit, delete and manage data in this system. If the admin wants, he can change anything in the system which will affect the whole system.

To guide the student on what to do next, information must be entered into the system and also give permission to the automated system to keep track of their activity. The whole suggestion process is completed by the super admin. Like what students want to learn and after learning that specific thing what suggestions give next.

This automated system is helpful for the newbies who want to learn to program but they have no guide or mentor. This system reduces their time finding a mentor also saves money which one should pay for their mentor.

We hope that every learner who wants to learn cleverly how to program will use our system.

Not only the learners but also the other sites which students use to learn things will use our automated system concept to give proper suggestions to guide their users. Because till now they have an only system where they can learn but not get any kind of suggestion based on their activity on what to learn next after completing a specific lesson.

6.2 Scope for Future Development

We think that our Developed system is not used by computer science students but also the students who are not from computer science backgrounds and that is our future scope.

There are many scopes of this system in the near future. Some are-

- This is now only web-based but we want it for the mobile user. For this reason, we will develop an Android application and IOS application.
- We will develop an online judge here.
- We will use an automated system that can collect users' other online judge data.
- Increase more functionalities for the users.
- To enhance the ease of use and capability.

References

- [1] For HTML, CSS& Bootstrap W3Schools, available at <<<https://www.w3schools.com/>>>, last accessed on 22-8-2021 at 03:19am
- [2] For learning HTML FreeCodeCamp , available at <<<https://www.freecodecamp.org/learn/>>>, last accessed on 2-5-2021 at 03:19pm
- [3] For learning web development Codecademy, available at <<<https://www.codecademy.com/>>>, last accessed on 12-09-2021 at 00:11am
- [4] For learning development EDx, available at <<<https://www.edx.org/>>>, last accessed on 10-12-2020 at 03:19am
- [5] For learning CSS Coursera, available at <<<https://www.coursera.org/>>>, last accessed on 31-12-2020 at 03:44am
- [6] For learning purpose GreekForGreeks , available at <<<https://www.geeksforgeeks.org/>>>, last accessed on 18-11-2021 at 10:19pm
- [7] For solving problem Stackoverflow, available at <<<https://stackoverflow.com/>>>, last accessed on 21-12-2021 at 8:32am
- [8] For designing Bootstrapstudio, available at <<<https://bootstrapstudio.io/>>>, last accessed on 22-2-2021 at 09:19pm
- [9] For the expected design Bootstrap, available at <<<https://getbootstrap.com/>>>, last accessed on 15-9-2021 at 01:41am
- [10] For learning purpose Onemonth, available at <<<https://onemonth.com/>>>, last accessed on 23-9-2021 at 07:19pm
- [11] For learning web Educative, available at <<<https://www.educative.io/>>>, last accessed on 5-10-2021 at 08:55pm
- [12] Learning development purpose Frontendmasters, available at <<<https://frontendmasters.com/>>>, last accessed on 11-12-2021 at 01:10am
- [13] Designing Pages in Figma, available at <<<https://frontendmasters.com/>>>, last accessed on 27-2-2021 at 02:33am

APPENDICES

Appendix A: Project Reflection

HTML: HTML (Hypertext Markup Language) is the code that is used to structure a web page and its content. For example, content could be structured within a set of paragraphs, a list of bulleted points, or using images and data tables.

CSS: CSS is the language for describing the presentation of Web pages, including colors, layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers. CSS is independent of HTML and can be used with any XML-based markup language. The separation of HTML from CSS makes it easier to maintain sites, share style sheets across pages, and tailor pages to different environments. This is referred to as the separation of structure (or: content) from the presentation.

Bootstrap: Bootstrap is a free front-end framework for faster and easier web development. Bootstrap includes HTML and CSS-based design templates for typography, forms, buttons, tables, navigation, modals, image carousels, and many others, as well as optional JavaScript plugins. Bootstrap also gives you the ability to easily create responsive designs.

JavaScript: JavaScript is a text-based programming language used both on the client-side and server-side that allows you to make web pages interactive. Where HTML and CSS are languages that give structure and style to web pages, JavaScript gives web pages interactive elements that engage a user.

Django: Django is a high-level Python web framework that enables the 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.

SQLite: SQLite is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. It is a database, which is zero-configured, which means like other databases you do not need to configure it in your system.

SQLite engine is not a standalone process like other databases, you can link it statically or dynamically as per your requirement with your application. SQLite accesses its storage files directly.

URL: A URL (Uniform Resource Locator) is a unique identifier used to locate a resource on the Internet. It is also referred to as a web address. URLs consist of multiple parts -- including a protocol and domain name -- that tell a web browser how and where to retrieve a resource.

UI & UX: User interface design or user interface engineering is the design of the user and UX design refers to user experience design, while UI design stands for user interface design. Both of these are crucial to an IT product and need to work closely together.

Turnitin Originality Report

Processed on: 02-Jan-2022 17:46 +06

ID: 1736856262

Word Count: 8371

Submitted: 1

FinalReport By Nazmus Sakib

Similarity Index		Similarity by Source	
29%		Internet Sources:	26%
		Publications:	4%
		Student Papers:	21%

[exclude quoted](#) [exclude bibliography](#) [exclude small matches](#) mode:

[Change mode](#) [print](#) [refresh](#) [download](#)

3% match (student papers from 20-Jan-2021) Class: Fall 2020 - Fall 2023 Assignment: SI Paper ID: 1490618914	✖
3% match (Internet from 29-Nov-2020) https://medium.com/strategic-designs/5-pillars-of-interaction-design-ixd-every-designer-must-know-about-a-very-big-ux-factor-5350e1706b0	✖
2% match (student papers from 30-Jun-2020) Class: Fall 2020 - Fall 2023 Assignment: SI Paper ID: 1351777985	✖
1% match (student papers from 12-Jan-2021) Class: Fall 2020 - Fall 2023	