



**Daffodil**  
*International*  
**University**

**Project Title**

**Tutor Finder System**

**Submitted by:**

Mohammad Rabius Sani

172-35-2208

Department of Software Engineering

Daffodil International University Approval

**Supervised by:**

Md. Shohel Arman

Assistant Professor

Department of Software Engineering

Daffodil International University

**Version-01**

APPROVAL

This Project titled on "Tutor Finder", submitted by **Mohammad Rabius Sani (ID: 172-35-2208)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS



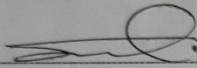
Chairman

Dr. Imran Mahmud

Head and Associate Professor

Department of Software Engineering

Faculty of Science and Information Technology Daffodil International University



Internal Examiner 1

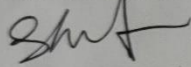
Md. Khaled sohel

Assistant Professor

Department of Software Engineering

Faculty of Science and Information Technology

Daffodil International University



Internal Examiner 2

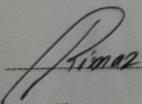
Md. Shohel Arman

Assistant Professor

Department of Software Engineering

Faculty of Science and Information Technology

Daffodil International University



External Examiner

Rimaz Khan

Managing Director

Tecognize Solution Limited

## DECLARATION

I hereby declare that, I have taken this project under the supervision of **Md. Shohel Arman**, Assistant Professor Department of Software Engineering, Daffodil International University. I additionally claim that this assignment doesn't had been submitted some other place for award of any degree.

*Rabius Sani*

-----  
**Mohammad Rabius Sani**

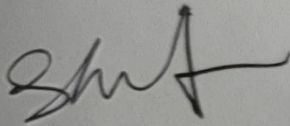
Id: 172-35-2208

Batch: 23

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University



-----  
Certified by:

**Md. Shohel Arman**

Assistant Professor

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

## **ACKNOWLEDGEMENT**

**ALHAMDULILLAH** & thanks to my **ALLAH** for everything. Then I would love to thanks my Parents & Teacher for their support. From my University, I have learned a lot about software engineering & also ethics, politeness & morality. I hope and believe that, they will help me to build my career.

I specially thanked to my supervisor **Md. Shohel Arman**, for gave me chance to work under him in my final project. He was very helpful & supportive. With him, I have done my project with a new experience and stress free.

Finally, my heartiest thanks to friends, and well-wishers for being with me and encourage me to complete my final project

## Contents

	<b>Page No</b>
Chapter 1: Introduction.....	1
1.1 Project Overview.....	1
1.2 Project Purpose.....	2
1.2.1 Background.....	2
1.2.2 Benefits.....	2
1.2.3 Goal.....	3
1.3 Stakeholders.....	3
1.4 Project Schedule.....	4
1.4.1 Gantt Chart.....	4
1.4.2 Release Plan/Milestone.....	4
Chapter 2: Functional Requirement Specification.....	5
2.1 Functional Requirements.....	5
2.2 Non-Functional Requirements.....	6
Chapter 3: System Analysis.....	6
3.1 Use Case Diagram.....	6
3.1.1 Use case Diagram for User.....	7
3.2 USE Case Description.....	7
3.3 Sequence Diagram.....	11
Chapter 4: System Design Specification.....	12
4.1.1 Activity Diagram For admin.....	12
4.1.2 Activity Diagram For student.....	13
4.1.3 Activity Diagram For Tutor.....	13
4.1.4 DataFlow Diagra.....	14
4.2 ER Diagram.....	14
Chapter 5: Development.....	15
5.1 Tools & Technology.....	15
5.1.1 HTML.....	15
5.1.2 CSS.....	16
5.1.3 PHP.....	16
5.1.4 Laravel Framework.....	16
5.1.5 MySQL.....	17
5.1.6 XAMPP .....	17

5.1.7 JavaScript.....	17
Chapter 6: System Testing.....	18
6.1 Introduction.....	18
6.2 Testing Strategies.....	18
6.3 Feature to be tested.....	18
6.4 Test case table.....	19
6.4.1 Test case for login.....	19
6.4.2 Test case for page&navigation.....	19
6.4.3 Test case for Sign up.....	19
6.4.4 Test case for forget password.....	20
6.4.5 Test case for adding course& chat.....	20
6.4.6 Test case for all other function .....	21
Chapter 7: User manual.....	22
7.1 Home Page.....	22
7.2 User Registration page.....	22
7.3 User login page.....	23
7.4 Dashboard.....	23
7.5 Course table.....	24
7.6 Student table.....	24
7.7 Services.....	25
7.8 About team.....	25
7.9 Subscribe now.....	26
7.10 Feature.....	26
7.11 Contact us.....	27
Chapter 8: Project Summary.....	28
8.1 Limitations.....	28
8.2 Obstacles And Achievements.....	28
8.3 Future Scope.....	28
8.4 References.....	29

**List of Figures:**

<b>Figure No:</b>	<b>Name of Figures:</b>	<b>Page No:</b>
1.1	Gantt Chart	4
3.1.1	Use case diagram	7
3.3	Sequence Diagram	11
4.1.1	Activity Diagram for admin	12
4.1.2	Activity Diagram for student & tutor	13
4.1.3	Dataflow Diagram	14
4.3	ER Diagram	14
7.1	Home Page	22
7.2	User Registration Page	22
7.3	User Login Page	23
7.4	Dashboard	23
7.5	Course table	24
7.6	Student table	24
7.7	Services	25
7.8	About team	25
7.9	Subscribe now	26
7.10	feature	26
7.11	Contact us	27

# Chapter-1

## Introduction

### 1.1 Project Overview:

The project is approaching to the solution of a basic problem of hiring a tutor. It will help tutor and student to communicate easily.

For student, it will display nearest location of tutors. They will able to send request to tutor. They will be able to search tutor subject wise, location wise. Tutor list will be shown as students subject wise with tutor rating. Student can give rating and review to their teachers only.

For tutor, it will display requests made by the students. They can be accepted or rejected student request. They can add their subject which they want to teach. They also can search and send request to student.

In admin side, they will be able to manage the whole system. They can add, remove and edit any kind of information. They can see how many user registered. Super admin can add sub admin. Sub admin have some limitation.



## 1.2 Project Purpose

### 1.2.1 Background:

Recently, there are too many tuition media in online. Most them are fake. So, many of teacher & students are victim of them. So, I decided to solve this problem and Create a platform where Teacher and student can communicate directly. Then I develop this Tutor Finder system, which is user friendly and secure.

### 1.2.2 Benefits of Project:

This system can solve all problems for Students and Teacher who wants tutor or tuition. The **“Tutor Finder System”** can provides all the facilities with very low cost and very fast then present. Due to automated system, it doesn't need much maintenance cost. For this reason it is more coast efficient.

- It saves our time.
- Teacher or Student can be searched easily in user local area.
- Can get best teacher with rating.
- Best teacher with low cost.
- It saves money, physical and mental harassment.

### 1.2.3 Goals:

Now a days, most valuable thing for people are time & money. By proper using Them we can get a bright future.

In the digital world, everyone is love to use internet. Most then love to spent time in social media. Where there is positive and negative site. Now everyone has a smartphone to use social media. They communicate with each other by it. So I decided to make my system as social media, where only student and teacher can communicate each other.

My system is built for student and teacher to provide them a good platform. Where they can search for their desire tutor or tuition subject & location wise. It will be a best solution for unemployment problem in our country. Rating system will help tutor find tuition easily.

### 1.3 Stakeholders

**Admin:** Admin module consists of a web panel. Admin can monitor the activities and perform tasks that are authorized for them. There can be more than one admin; they all can have different responsibilities. There will be a master admin, it can add or remove admin and can also view the logs of the other admin.

**User (student/teacher):** As the system is web based. It has two users tutor and student. Both user can search and request to each other. Both user can chat with each other. Until they agreed, their info will be private. Student can rate tutor. User can accept or reject request.

## 1.4 Project Schedule

### 1.4.1 Gantt Chart:

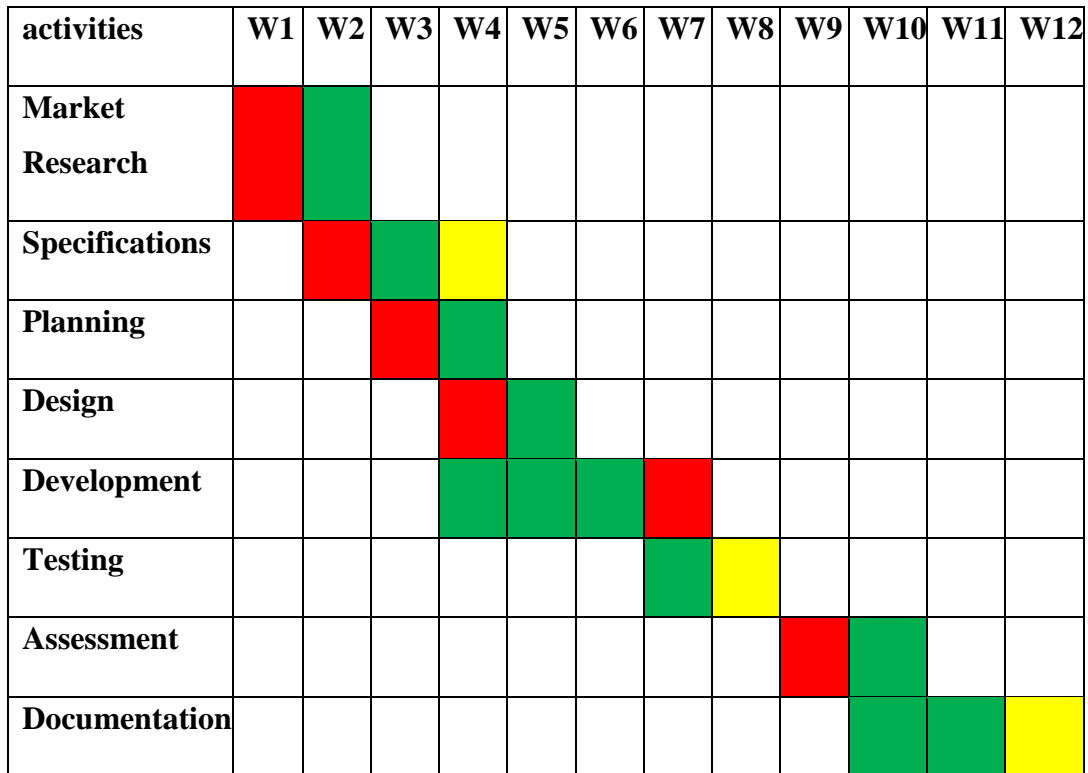


Figure: 1.1

### 1.4.2 Release Plan:

Serial	Task Name	Time
1	Requirement analysis	2 Week
2	Project proposal	1 Week
3	Database design	2 Week
4	User interface design	1 Week
5	Implementation	3 Week
6	Testing	1 Week
7	Evaluating the project	2 Week

## Chapter-2

### Functional requirement and specification

#### 2.1 Functional Requirements:

Functional requirements define a system or its component. It specifies “what should the system do?” User specifies functional requirement. It is mandatory to meet these requirements. It is captured in the use case.

The functional requirements of “**Tutor Finder**” are:

- **Registration**
- **Log in**
- **Verification**
- **Change password**
- **Courses**
- **Payment**
- **Reviews**
- **Settings**
- **Searching**
- **Request**
- **Request–response**
- **Chat**
- **Reviews**
- **About**

## 2.2 Non-Functional Requirements:

- lowest System response time
- interval time between the submission of a tutor request, date/location and information to student shall be less than 5 min
- interval time between the determined and the actual confirmation shall be 8min
- 10 tutor requests for every 1 hour.
- System recovery time after unexpected failures shall be less than 30 minutes.

## Chapter-3

### System Analysis:

#### 3.1 Use Case Diagram:

Use-case diagrams **describe the high-level functions and scope of a system**. These diagrams also identify the interactions between the system and its actors. The use cases and actors in use-case diagrams describe what the system does and how the actors use it, but not how the system operates internally.

UCDs have only 4 major elements: The actors that the system you are describing interacts with, the system itself, the use cases, or services, that the system knows how to perform, and the lines that represent relationships between these elements.

- UCDs only summarizes **relationships** between use actors, cases & systems.
- UCDs does **not show the order**.

### 3.1.1 Use case diagram for User, Admin:

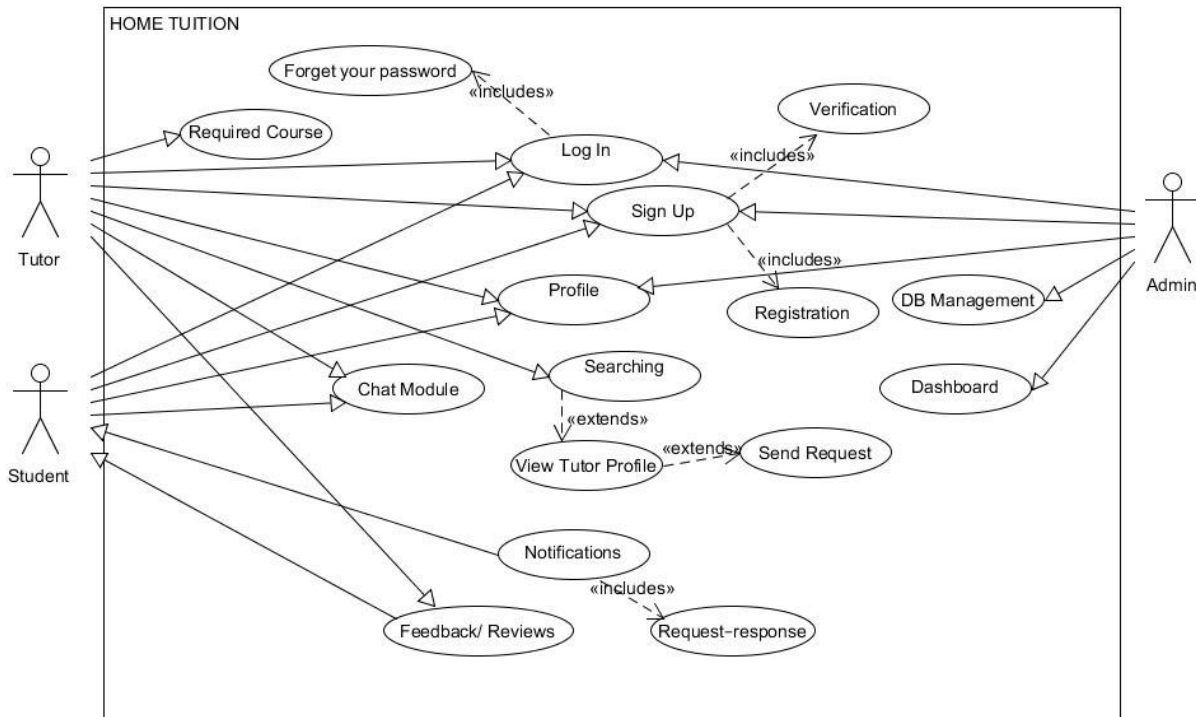


Figure: 3.1.1 (Use case diagram for User and Admin)

### 3.2 USE Case Description:

#### 1. Admin Module:

In admin side, they will be able to manage the whole system. They can add, remove and edit any kind of information. They can see how many user registered. Super admin can add sub admin. Sub admin have some limitation. They can add, remove and monitor full system. Admin will keep all the payment history.

**1.1 Login:** If a person has assigned for application he/she can able to use and manage the application through admin panel. If admin forgot the password they can change their password.

**1.1.1 Forget your password:** Admin can change their password through email that provided at the time of assigning. With this email admin can update their password. A new tab will display on web browser to edit password.

- 1.2 Profile:** In profile section, Admin can manage or update their personal information and password.
- 1.3 Dashboard:** This is the most powerful section of the admin panel where dashboards make it possible to monitor the progress of data, and reflect each change that has happened to it. It provide valuable knowledge that must be cut to cater to different needs, but that doesn't mean to show all data on the screen only important data such as teachers logs, student logs, recent reviews, new registrations and other information.
- 1.4 Admin:** In this section head admin can set different roles of different sub admin. Head admin can add or remove sub admin with defining their roles. Role can be changeable.
- 1.5 Tutors:** In this section, Admin or editor can verify the tutor or can block or unblock the tutor on the basis of rating and reviews.
- 1.6 Courses:** In this section, Admin/editor can manage course list and can add or remove courses as per demand. Admin can also view courses that requested by teacher and recommended by student.
- 1.7 Students:** In this section, Admin or editor can verify the student or can block or unblock the student on the basis of rating and reviews.
- 1.8 Reviews:** In this section, Admin/Editor/Moderator can view, edit or delete the reviews if found anything offensive. Student and tutor rating will be maintaining separately in order for future enhancement.
- 1.9 Settings:** In this section, Admin can edit in system settings such as new updates and feature can be done in this section.

## **2. Student Module:**

Student is the one who can generate request to the concern tutor as own in an application. But it contains a whole process which can initiate with the registration process. Following are the function of student module which is listed below:

- 2.1 Login:** registered students can able to use all features of the application. Otherwise they need to sign up. They can change their pass if they forget it

- 2.1.1 Forget your password:** for change their password, they need to give the email they use in registration. Then they can change or update their password. A new tab will display on application that allow user for new password.
- 2.2 Sign Up:** By clicking this button student can switch from main screen to registration screen.
- 2.2.1 Registration:** In this activity a form is shown having some required field which should be filled such as name, email address, password, date of birth and qualification etc.
- 2.2.2 Verification:** here student is required a verification code that will be generated by application and will be send to the student with the provided verification process (Email Verification and/or SMS Verification). Student can enter the provided code in order to complete the process. After that student can move toward the home screen of the application.
- 2.3 Searching:** Student can search and view a tutor according to their location or course wise. Student can also open the tutor profile by clicking on map
- 2.4 Tutor Profile:** Student can view tutor qualification and other information but not the private information such as contact number and resident address. Student can message tutor thorough application.
- 2.5 Chat:** Chat module will help the student to send a message to tutor. They can also maintain their chat history by deleting a conversation.
- 2.6 Courses:** Student can view courses from list. Student can request to admin panel for addition of new courses in given course list
- 2.7 Feedback/Reviews:** Student can give their feedback and reviews related to the tutor and company. For the future benefits
- 2.8 Settings:** They can update or remove their personal information. It include password change
- 2.9 About:** This is the section where student can get information about the developers and the company. It will help the user to understand the usability of application.



### 3. **Tutor Module:**

Tutor can accept or delete request which were sent by the student. But it contains a whole process which can be start with the registration process. Following are the function of student module which is listed below.

**3.1 Login:** Registered Tutor can able to use all features of the application. Otherwise, they need to register first. Forgotten pass can be recovered by forget password.

**3.1.1 Forget your password:** for change their password, they need to give the email they use in registration. Then they can change or update their password. A new tab will display on application that allow user for new password.

**3.2 Sign Up:** By clicking this button tutor can switch from main screen to registration activity/screen

**3.2.1 Registration:** In this activity a form is shown having some required field which should be filled such as name, email address, password, date of birth and qualification etc.

**3.2.2 Payment:** At the time of registration, Application will charge a specific amount for registering a tutor. Without payment tutor will not be register. Payment schedule can be on monthly basis or yearly.

**3.2.3 Verification:** here student is required a verification code that will be generated by application and will be send to the student with the provided verification process (Email Verification and/or SMS Verification). Student can enter the provided code in order to complete the process. After that student can move toward the home screen of the application.

**3.3 Request–response:** Tutor will response to student request.

**3.4 Chat:** Chat module will help the tutor to reply the message of student. They can also maintain their chat history by deleting a conversation.

**3.5 Courses:** Tutor can view their courses, update and remove the courses from list appeared in their profile. Tutor can request to admin for addition of new courses in given course list

**3.6 Feedback/Reviews:** Tutor can give their feedback and reviews related to the student and company. For the future benefits.

**3.7 Settings:** Tutor can maintain their personal information. Such as update their profile or qualification and password.

**3.8 About:** This is the section where tutor can get information about the developers and the company. It will help the user to understand the usability of application

### 3.3 Sequence Diagram

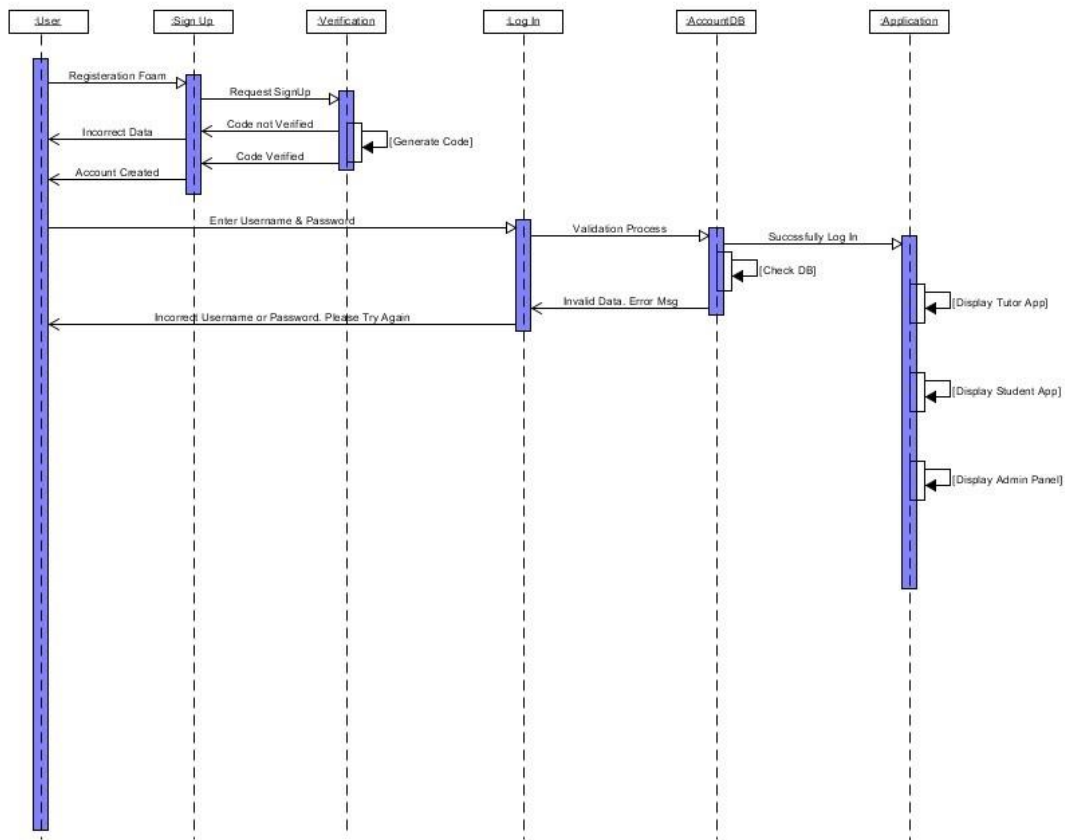


Figure : Sequence Diagram

# Chapter-4

## System Design specification

### 4.1 Activity Diagram:

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. An activity diagram is used by developers **to understand the flow of programs on a high level**. It also enables them to figure out constraints and conditions that cause particular events.

#### 4.1.1 Activity Diagram for Admin:

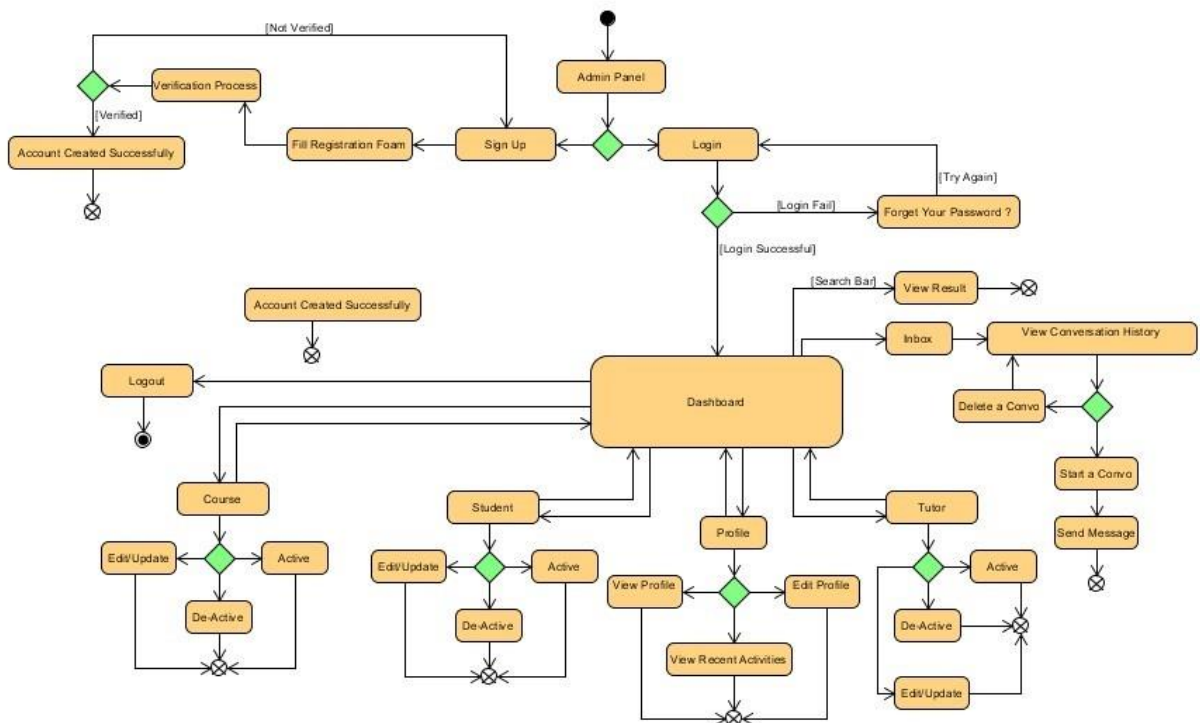


Figure: 4.1.1(Activity Diagram for Admin)

### 4.1.2 Activity Diagram for Student:

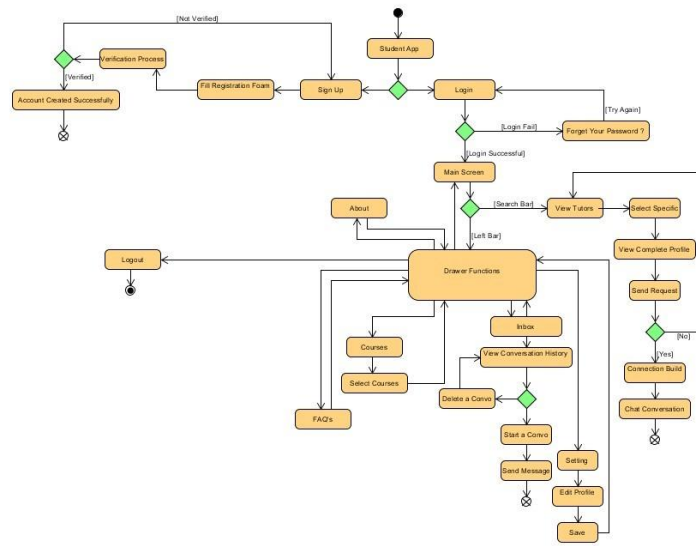


Figure: 4.1.2(Activity Diagram for student)

### 4.1.3 Activity Diagram for Tutor:

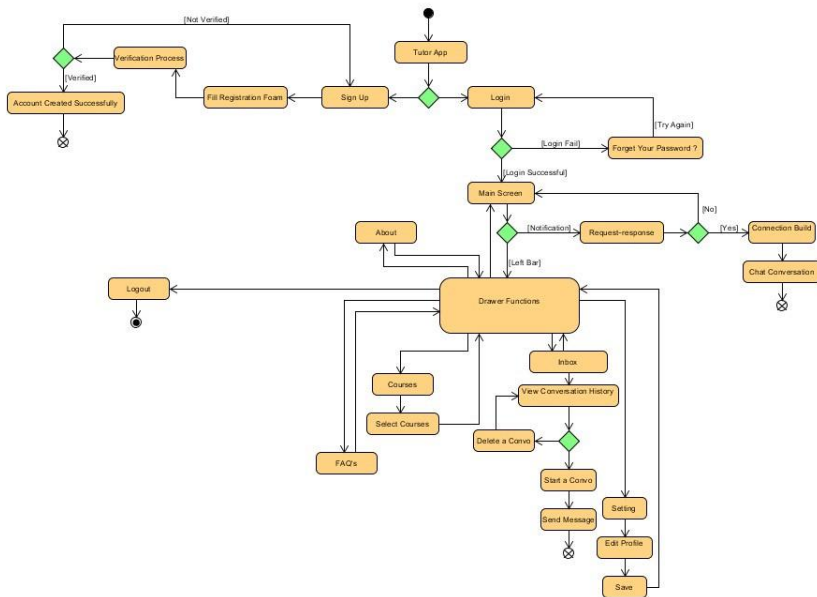


Figure: 4.1.3(Activity Diagram for Tutor)

### 4.1.3 DataFlow Diagram:

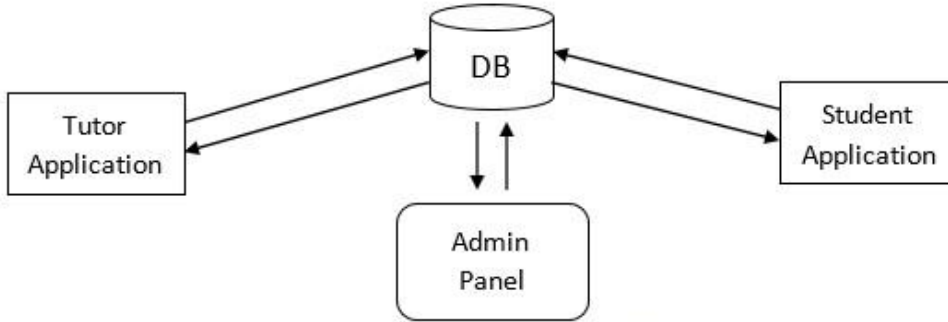


Figure 5-1 Data Flow Diagram (DFD)

Figure: 4.1.3(DataFlow Diagram

### 4.2 ER Diagram:

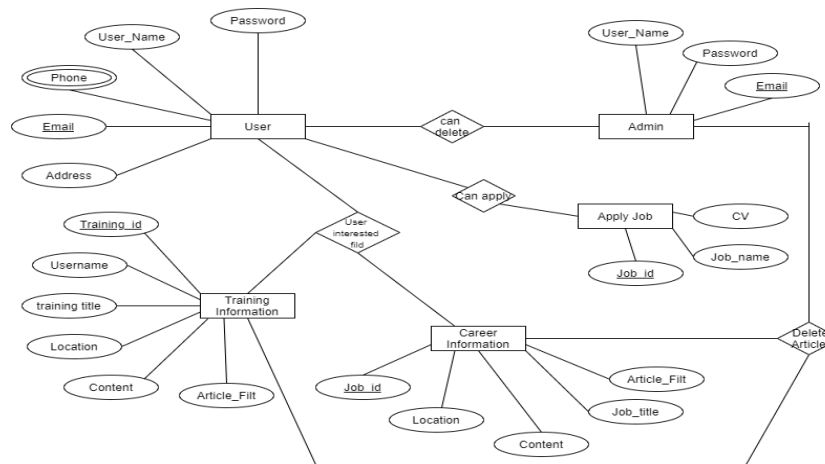


Figure: 4.2(ER Diagram)

## Chapter-5

### Development

#### 5.1 Tools & Technology:

In this project I used open source tools. So that I can cut the cost of this project.

- HTML,
- CSS
- JavaScript,
- Bootstraps
- Apache server
- PHP & Laravel Framework for backend
- MySQL

#### 5.1.1 HTML

Html is used for frontend in this project. As it is a markup language which is a set of markup tags.

I used it because:

- Easy to learns.
- Easy to use.
- Supported by all browser.
- Cost effective.

### 5.1.2 CSS

As CSS or Cascading Style Sheets is used to style and layout web pages. I also used it for style my web page what I done before with html

### 5.1.3 PHP

For backend there are many language. But php is one of the best among then. Because, we can

- Generate dynamic page content.
- Create, open, read, write, delete, and close files on the server.
- Collect form data.
- Send and receive cookies.
- Add, delete, and modify data in database.
- Be used to control user-access.
- Encrypt data.

Many more we can do with PhP. So for that I chose this language

### 5.1.4 Laravel Framework

It is a **PHP framework**, which is open source. It is easy to understand. Model-view-controller design pattern is followed by it. It can reuses the existing components of different frameworks which helps in creating a web application. The web application thus designed is more structured and pragmatic..

### 5.1.5 MySQL

One of the best web application server is MySQL. Which run the web application. I decide to use it beause:

- It can be use for small and big project
- It is easy to use and very fast.
- It use standard sql .
- a number of platforms use for this compile.
- It is free.

All data are stored in table in mysql sever.

### 5.1.6 XAMPP s

- It is free and open source
- Its cross-platform web server's solution stack package
- It interpreters for scripts written in the PHP and Perl programming languages.
- It is simple & lightweight
- it works on every operating system.

### 5.1.7 JavaScript

- It is **the Programming Language for the Web.**
- It can update and change both HTML and CSS.
- It can calculate, manipulate and validate data.



## Chapter-6

### System Testing

#### 6.1 Introduction:

Testing is a method to check the software, to ensure it is error free. It actually done for to identify errors, gaps or missing requirements. Which will be solved before launch.

#### 6.2 Testing Strategies:

Software testing methodologies are the various strategies or approaches used to test an application to ensure it behaves and looks as expected.

The purpose of a test strategy is to provide a rational deduction from organizational, high-level objectives to actual test activities to meet those objectives from a quality assurance perspective.

#### 6.3 Feature to be tested:

Features Id	Features Name	Description	Involves User
01	Registration	User must be needed registration first.	Users
02	Login	Login as authenticated User and Admin.	Users, Admin
03	Submit training and career info	User can Submit training and career information after login in the system.	User
04	Give Report	User can give report others users post	User
05	Update Profile	User update her own profile in the system.	User
06	Reset password	User and Admin will not Change their password	Admin, User
07	Delete User	Admin will delete User account for fake post.	Admin
08	Delete reported article	Admin will delete User reported article	Admin
09	Logout	Logout from the system.	Admin, User
10	Search tutor / tuition	Search for tutor and apply	User

## 6.4 Test case table:

**Objective:** To check if login functionality is working perfectly.

Test ID	Test case/test script	Attribute and value	Description	Expected result	Result
HT-031	Login as a 'Student'.	Username: 03343625155 Password: 1234	Logging in the student application.	Student should be logged in.	Pass
HT-032	Login as a 'Tutor'.	Username: 03343625155 Password: 1234	Logging in the tutor application.	Tutor should be logged in.	Pass

**Objective:** To check if the correct page with the correct navigation bar is loaded.

Test ID	Test case/test script	Attribute and value	Description	Expected result	Result
HT-031	Search the course from the search bar.	Null	Searching the courses from the student application.	Searched courses are shown on the screen.	Pass
HT-032	Search the courses from search bar.	Null	Searching different courses from the system to get registered in.	Searched courses are shown on the screen.	Pass

**Objective:** To check if signup functionality is working properly.

Test ID	Test case/test script	Attribute and value	Description	Expected result	Result
HT-031	Signup from the Student application.	Null	Signing up as student from student application by filling the form correctly.	New account for the user should be created.	Pass
HT-032	Signup from the tutor application.	Null	Signing up as tutor from tutor application by filling the form	New account for the user should be created.	Pass

**Objective:** To check if forget password functionality is working perfectly.

Test ID	Test case/test script	Attribute and value	Description	Expected result	Result
HT-031	Student forget password.	Null	If the student forgets the password, they can reset it with this functionality.	Open another window that takes phone number and a code is sent.	Pass
HT-032	Tutor forget password.	Null	If the tutor forgets the password, they can reset it with this functionality.	Open another window that takes phone number and a code is sent.	Pass

**Objective:** To check if adding courses functionality is working perfectly.

Test ID	Test case/test script	Attribute and value	Description	Expected result	Result
HT-031	The tutor adds courses that they want to teach.	Null	The new courses are added to the tutor's profile.	The new courses should be added to the tutor's profile.	Pass

**Objective:** To check if chat functionality is working perfectly.

Test ID	Test case/test script	Attribute and value	Description	Expected result	Result
HT-031	Student chat with the tutor.	Null	The chat from student to the tutor through the chat module.	The student message should be received by the tutor application.	Pass
HT-032	Login as a 'Tutor Member.	Null	The chat from tutor to the student through the chat module.	The tutor message should be received by the student application.	Pass

**Objective:** To check if all other functionalities are working perfectly.

<b>Test ID</b>	<b>Test case/test script</b>	<b>Attribute and value</b>	<b>Description</b>	<b>Expected result</b>	<b>Result</b>
HT-031	Tutor changing their status (online/ offline)	Null	The tutor change their status from the buttons on their main home screen.	The status of tutor will change from the current status.	Pass
HT-032	Tutor changing their availability	Null	The tutor change their availability from the availability activity.	The tutor's available timings should change.	Pass
HT-033	Logout	Null	The user can log out from the application.	The user should be logged out of the application.	Pass

## Chapter-7

### User manual

#### 7.1 Home Page:



Figure: 7.1(Home Page

#### 7.2 Registration page:

The image shows a registration form titled 'Register in Tutor Finder'. The form is set against a light blue background. At the top of the form is the title 'Register in Tutor Finder'. Below the title is the text 'Registering For (Please Click Below)' followed by three radio button options: 'Student', 'Guardian', and 'Tutor'. The form then contains several input fields: 'Fullname', 'Date of Birth' (with a date format 'mm/dd/yyyy' and a calendar icon), 'Institution Name', 'Email', 'Contact Number', 'NID Number', and 'Address'. At the bottom of the form is a green 'Register' button.

Figure: 7.2(User Registration Page)

### 7.3 login page:

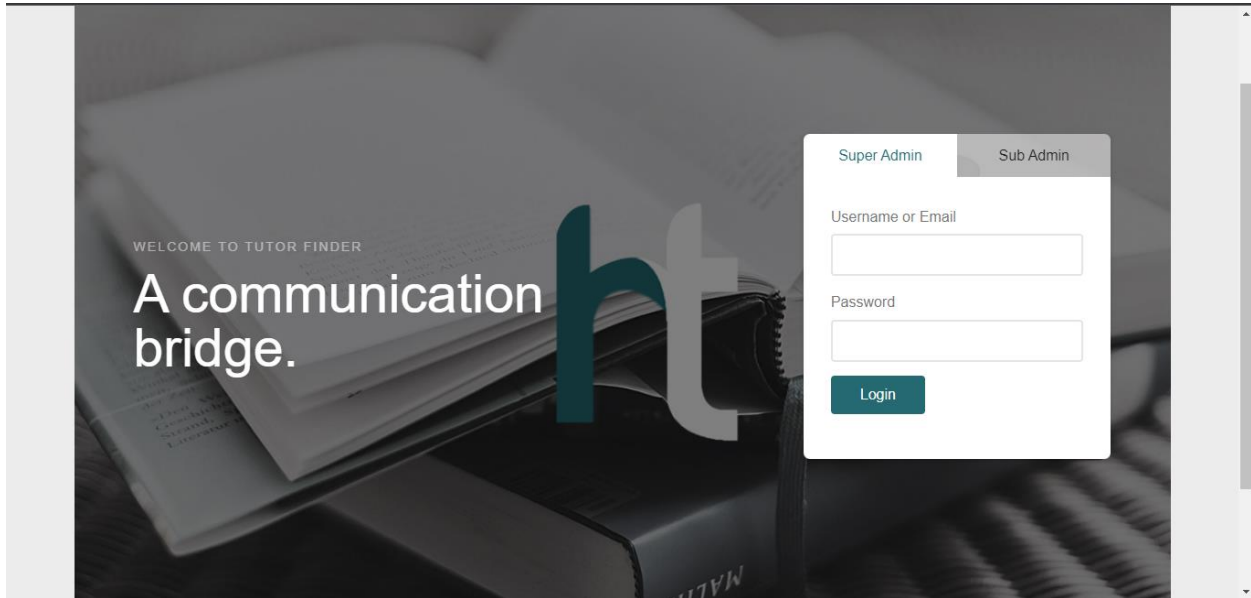


Figure: 7.3(Login page)

### 7.4 Dashboard:

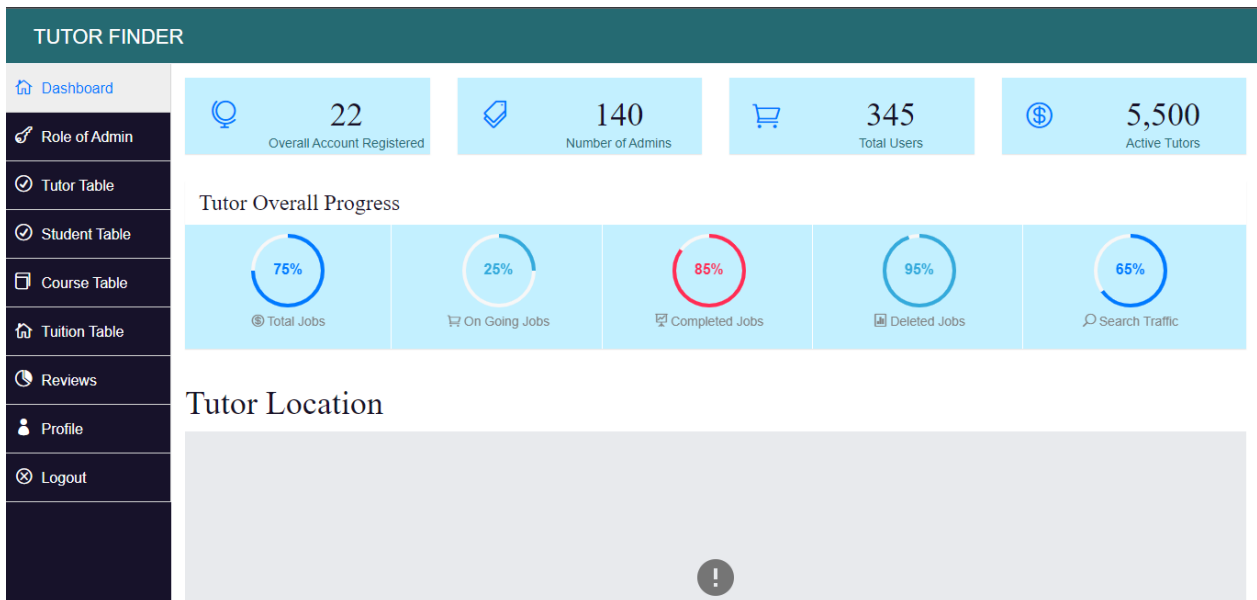


Figure: 7.4(Dashboard)

## 7.5 Course Table:

**TUTOR FINDER** Search




Dashboard  
Role of Admin  
Tutor Table  
Student Table  
**Course Table**  
Review  
Profile  
Logout

### Course Registration

Tutor Id  
Course Id

**Add Courses**

### Course Information

Course Id	Course Name	Type of Course	Description	Action
				  

C:/Users/dcl/Desktop/Smart-tutor-Finder---Admin-Panel-master/./home.html

Figure: 7.5(Course table)

## 7.6 Student Table:

**HOME TUITION** Search

Dashboard  
Role of Admin  
Tutor Table  
**Student Table**  
Course Table  
Review  
Profile  
Logout

### Student Information




Student Id	Student Name	Phone Number	Email Address	Action
				  

Figure: 7.6(Student Table)

## 7.7 Service:

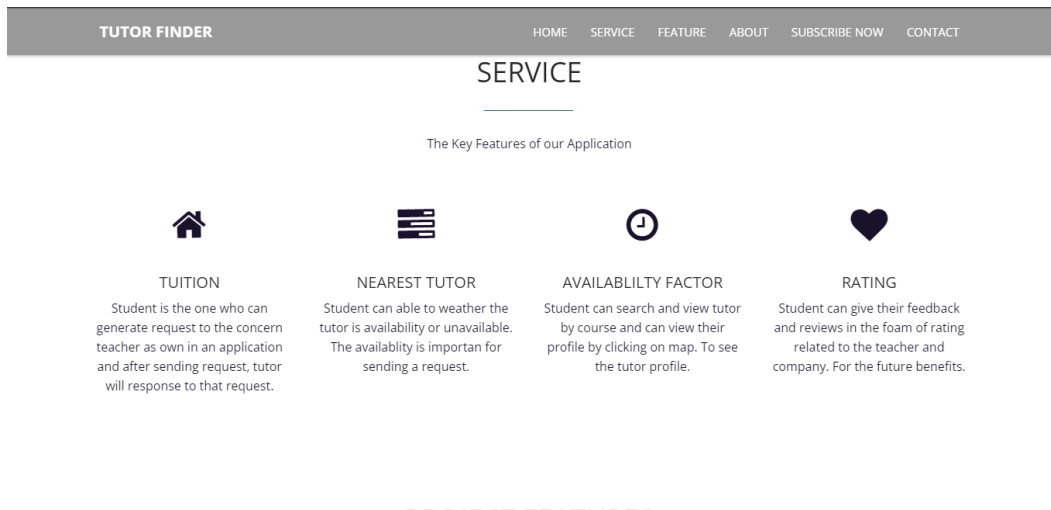


Figure: 7.7(Service)

## 7.8 About Team:

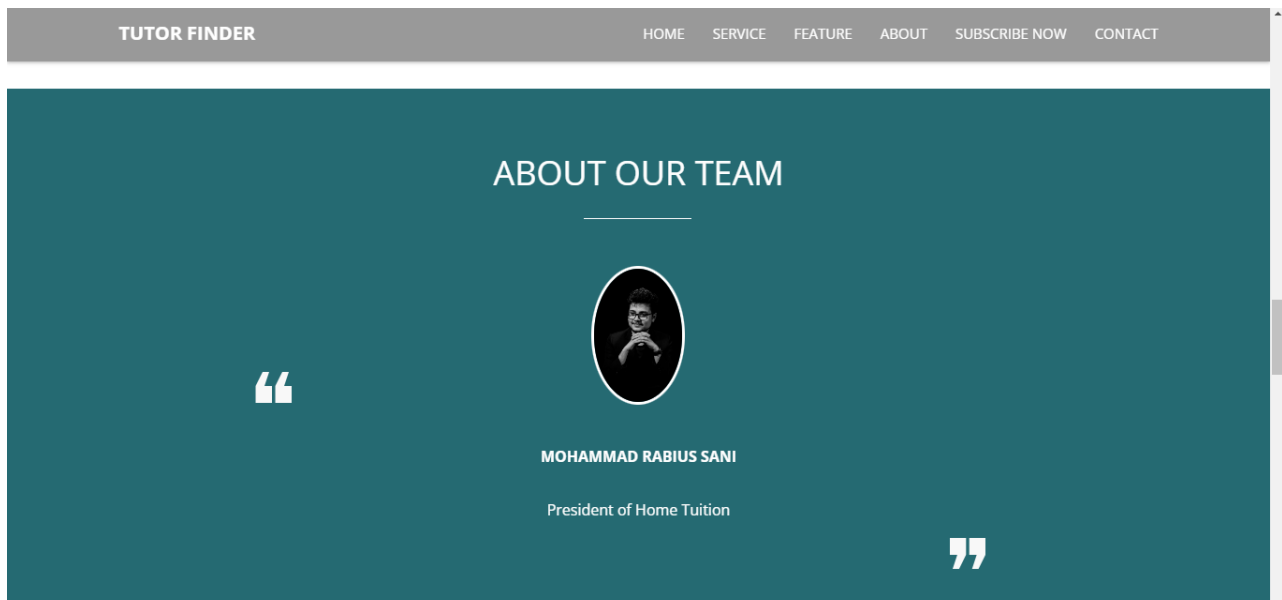


Figure: 7.8(About Team)



## 7.9 SUBSCRIBE NOW:

TUTOR FINDER HOME SERVICE FEATURE ABOUT SUBSCRIBE NOW CONTACT

### SUBSCRIBE NOW

Our subscription price for tutors.

SILVER	GOLD	DIAMOND
100 Tk	500 Tk	1,000 Tk
Weekly	Monthly	Yearly
No Bonus Points	Trail Bonus Points	All Bonus Points

C:/Users/dcl/Desktop/Smart-tutor-Finder---Admin-Panel-master/.../login.html

Figure: 7.9(SUBSCRIBE NOW)

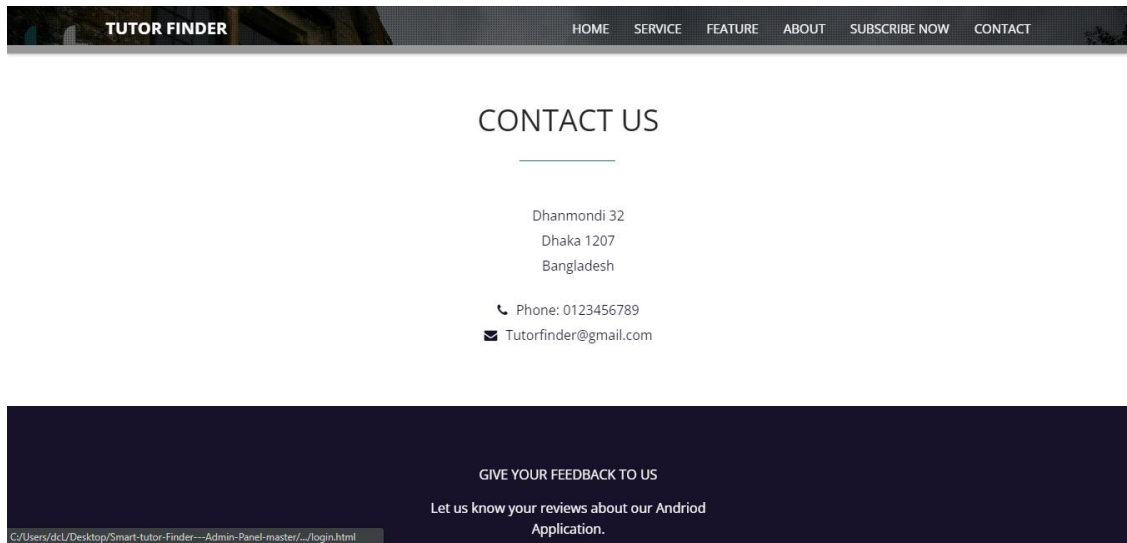
## 7.10 Feature:

TUTOR FINDER PROJECT FEATURES HOME SERVICE FEATURE ABOUT SUBSCRIBE NOW CONTACT

Following are the main main features of our project "Tutor Finder". Which will entertain two users and provide various types of facilities.

Figure: 7.10(Feature)

## 7.11 CONTACT US:



**Figure: 7.11(CONTACT US)**

## Chapter-8

### 8.1 Project Summary:

#### 8.1.1 Summary:

From the starting of this project, I start to collect requirement. Then when I collected all my requirement, I submit my project proposal. After that I start design my UI for my project. After completing designing, I start developing frontend with html, css, JavaScript. Then i stored data in database which is very important. Then I start backend with php to complete my project.

After completing all, I know testing is more important. So I did that and solve all my problem.

### 8.2 Limitations:

A project without any limitations is really hard. I tried my best. But there still some limitation. Such as: -

- 1 Can't identify fake id automatic
- 2 Not fully ready automatic payment.

### 8.3 Obstacles and Achievements:

During this project, I have face some obstacles. Though it's a common matter and solving obstacles will help to get achievement. So 1<sup>st</sup> I tried to solve them alone. But when I was unable to solve them, I took my supervisor help. He helped me to get out from them always. In this project my best achievement is, now i have confidence to do project alone.

### 8.4 Future Scope:

There are some future scope in this project. Such as:

- 1 Add parent portal
- 2 Add online course
- 3 Develop as mobile app

## 8.5 References:

To develop my project, I have taken help from many places. Some reference are given bellow:

- [www.w3schools.com](http://www.w3schools.com)
- [www.jquery.com](http://www.jquery.com)
- <http://bdtutors.com/>
- <https://getbootstrap.com>
- <http://www.hometutorbd.com>

# Plagiarism Result



Project Report Library

to me, Md.Shohel ▾

Dec 17, 2022, 10:51 AM (1 day ago)



Dear Student,

Your Plagiarism Result is 25% for details Please see the attachment file.

**Please read the instruction:**

- **Report to be arranged according to the Page Numbering:**
  - a. Preliminary pages must be in lower case roman numerals e.g. i, ii, iii.
  - b. All pages of the main body or from chapter one will be numbered in Arabic numerals e.g. 1, 2, 3.
  - c. All pages have to be arranged according to the table of contents
- 2. Format: The report should be in **ONE FILE** and **PDF** document.
- 3. Copyright Note: Write ©**Daffodil International University** at footer
- For Library Clearance please fill up your information in Internship Portal. Five fields must be completed as like- ID, Name, Department, Project/Internship Title & Supervisor Name.  
[http://internship.daffodilvarsity.edu.bd/index.php?app=applicant\\_login](http://internship.daffodilvarsity.edu.bd/index.php?app=applicant_login)
- **Please attach the supervisor & your signature in the Approval and Declaration page.**
- When you send us a new document, just send a reply to all. Don't create/send new mail.
- **If needed please contact the following Officer**
- Badhan Hubert Corraya-01981323203, Md. Mostafizur Rahman-01847334818, Ms. Umme Ahasan-01847334816, Md. Dulal Uddin: 01847334802, Ms. Syeda Aklima-01713493041

\*\*\*