“Colabs-Resolving Problems”

Project Report submitted in partial fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

IN

Computer Science and Engineering

By

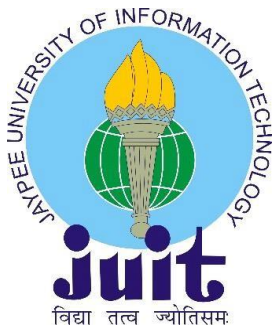
Shriya Chopra (151235)
Shubham (151244)
Jatin Walia (151251)
Sahil Gaur (151255)
Deepanshu Singla (151314)
Jaipreet Chandok (151320)
Kunal Kumar (151336)

UNDER THE GUIDANCE OF

Miss Divya J

Systems Engineer - Education, Training & Assessment, ETA, Infosys Mysore

To



Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat, Solan-173234

Himachal Pradesh

Candidate's Declaration

I hereby declare that the work presented in report entitled "Colabs-Resolving your problems" in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering** submitted in the department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried over a period from 4th February 2019 to 17th May 2019 under the supervision of **Miss. Divya J, Systems Engineer - Education, Training & Assessment, ETA, Infosys Mysore**

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Shriya Chopra (151235)
Shubham (151244)
Jatin Walia (151251)
Sahil Gaur (151255)
Deepanshu Singla (151314)
Jaipreet Chandok (151320)
Kunal Kumar (151336)

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

_____

Supervisor Name: Miss. Divya J

Designation: Systems Engineer

Department: ETA

Date:

ACKNOWLEDGEMENT

We are grateful to every members of **ETA department of Infosys,** Mysore for their continuous guidance and supervision as well as the CCD for providing us with all the necessary software and hardware requirements and the information to complete our project.

We would also like to express our gratitude towards **Miss Divya J**, Systems Engineer, ETA, for her continuous guidance and co-operation and helping us to tackle the complicated parts of our project.

We are also grateful to faculty of Computer Science Department of **Jaypee University of Information Technology, Waknaghat** for their guidance over the last 4 years and nurturing us into professionals.

Shriya Chopra (151235)
Shubham (151244)
Jatin Walia (151251)
Sahil Gaur (151255)
Deepanshu Singla (151314)
Jaipreet Chandok (151320)
Kunal Kumar (151336)

TABLE OF CONTENT

Abstract

The application “Colabs-Resolving your Problems” aims to reduce the dependency of Experts/Teachers being present physically after official hours to guide Trainee/Students. Certain common queries which are already answered can be used by the other people who need the same, etc. As an end user a trainee can post a question/query/doubt and get it answered by the experts. A trainee can also answer to a post if he/she knows the appropriate solution for the same. The experts can suggest a suitable solution for each posts or can approve/reject the solutions suggested by the other trainees.

A leader board has been maintained for various statistics like trainee with more number of correct solution for queries. There are all the functionalities which you would expect in any fully functioning website.

Chapter-1 INTRODUCTION

1.1 Introduction

With “Colabs-Resolving your problems” we wanted to create a platform which could provide trainees with a tool which will help them to solve their problems in a short span of time with the help of their colleagues and some experts. We already know about the importance of similar type of websites like “Stack Overflow” where enthusiastic and professional programmers meet to find innovative solutions and “Quora” where individuals can discuss about wide range of topics and where users can provide their opinion on a given topic. Our project will help trainees to post their problems on our online application instead of having to clear their doubts by having to physically go to the respective course expert. The question of the trainees can be answered by either a trainee or an expert in the respective field. The answers provided can then be marked irrelevant by other trainees if the answer is incorrect/vague/ambiguous, the marked answers will then be sent to the expert and he will take the final decision whether the answer is correct or incorrect. The expert is also vested with the power to block users who are creating nuisance on the application.

1.2 Problem Statement

1000’s of trainees undergo campus training during a single session/semester, and every one of those trainees have different problems in different sectors of technology which requires experts from all those different sectors to guide these trainees. Such kind of service is obsolete and impractical due to the cost and number of experts needed to achieve such a task. Hence there is a need to find a better way to solve this problem.

Thus arises the need for a web portal where all the trainees can post their queries and get them answered by experts or peers in category the question has been posted. Experts will have a major role as they will act as admin and can block users and mark questions/answers as irrelevant. Also a proper log will be maintained about each and every activity in the web portal including keeping track on who is the proficient of the trainee and rewarding the top 3 trainees with certificates based on daily, monthly and overall rating.

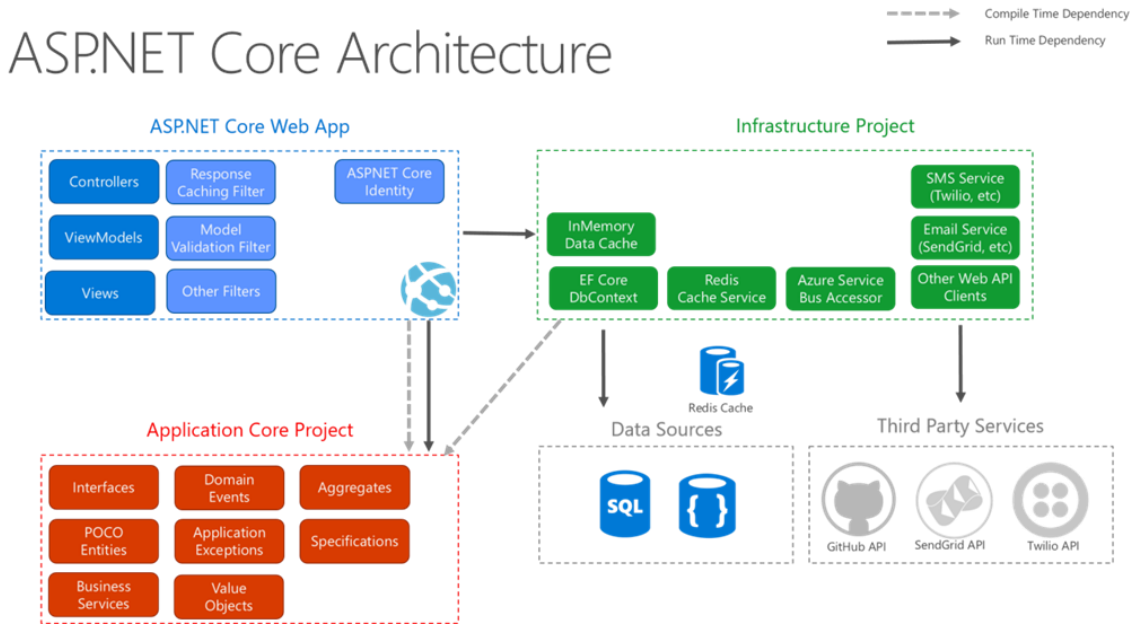
1.1 Objectives

Our Objective is to create a user-friendly and secure application where the trainees/students can interact with the experts/teachers in a professional manner and get all their queries answered efficiently. We aim to reduce the trainee's dependency on physical presence of an expert to guide him.

1.2 Methodologies

Since we are creating a web-application, there are various technological STACKS which have to be considered and we have to decide which stack suites the SRS (Software Requirements and Specification) best. In our case we have decided to use **ASP.NET Core** for backend-purposes and Bootstrap + Semantic UI for front-end.

ASP.NET Core is a cross-platform, open-source framework which provides high performance for building Web applications and Cloud applications. It can also be used to create apps based on IoT and mobile backend.



1.2.1 Sql Server 2012

It is one of the major ODBMS (Operational database management system) and is the crown-jewel of Microsoft's industry leading data platform. It has a highly efficient database Engine (which stores, processes and secures data) and it provides its user with a controlled access and very fast processing so that it can cope with the consumption requirements of even most of the large-scale application.

Even though Machine Learning has not been integrated in our project, Sql Server supports machine learning into most workflows. The concept of stored procedures can drastically reduce the size of any code if used wisely.

All operation which are invoked on SQL Server 2012 are done via TDS which is Microsoft-defined format, and is used for transferring data between the client and the server. Database consists of various tabled with various columns just like any other relational database. It supports all different datatypes (int, varchar, char, decimal, date, datetime, password, bit) which one would expect and also some other highly useful ones. User-defined datatypes and also allowed along with the highly useful stored procedure which helps reducing redundancy and repetition in code. Since the database can I shared using SSMS (Sql Server Management Studio), it makes it very comfortable for the dev Team to collaborate their progress and ensures a steady flow in the project compared to some other databases.

1.2.2 Semantics UI

Semantics is a framework used for front-end development and is powered by LESS (Leaner Style with block-formatting syntax) and jQuery and it provides a very clean and elegant user experience. Semantic UI is structured differently than most other frameworks and uses 5 categories so that the developer can easily use and re-use the different Component. The five Components are Element (buttons, inputs and other standard elemnets), Collection (forms etc.), View (website content), Module (JavaScript function like making menu responsive etc.), Behaviors.

1.2.3 MVC

MVC (Models-View-Control) is one of the most widely used framework in web development and is the fundamental pillar of any good website. The basic concept of MVC is to segregate the web application into three different components, all of which can make or break any projects.

1.4.3.1 Models

Models which are all of the database and database algorithms and logics and the programmer has to design and maintain during the course of extensible project such as ours.

For example, in our Case there is a User Object which will manipulate Answers, Question objects and other Objects will make use of information in the User Object. One such scenario is the leaderboard object making use of user object to calculate the top 3 trainees. There are various other objects and around 50 scenarios which have to carefully monitored to ensure proper functionality of the project since failure in even one of these scenarios can result in domino effect causing the entire project to fall down.

Clean model classes have to be created and they must be collaborated with our database. Data Annotation also must be defined using rules and attributes so that they are applied to the client side of the software. Entirety of the business login must be contained in the model, along with any implementation and logic. When the project is build, the controller will create and populate all the View-Model instances.

1.4.3.2 Views

View is front-end of the website(User-Interface). Using Views, we can control how the data which is received from the model can be displayed to the user and we also have to control our data using view. For example, in our application there is a button to post a question which make a form pop-up, using that form, we have to interact with our database so that the question is updated in the database. In ASP.NET MVC, different methods in a controller can yield different views, that is why in the Views folder of MVC, there are multiple sub-

folder with same name as controller. For Example, both trainee and expert controller has a view called viewAnswers and our program needs to know which view is being called and when. We have used Razor View Engine in our views (.cshtml), which is combination of html tags and server side, in case we want to write our C# code in the file, we can use @ character to represent that.

There is also the concept of partial views which have been widely used in our application. A partial view is simply a view with the only difference being its re-usability, it helps in eliminating repetition in the code by creating a partial view if user is using the same view in multiple pages, one such example is that of the navigation bar, which is something that is repeated in every page, so rather than writing the code of navigation bar over and over again in each page, a partial view can be created once, and then re-used multiple times.

1.4.3.3 Controllers

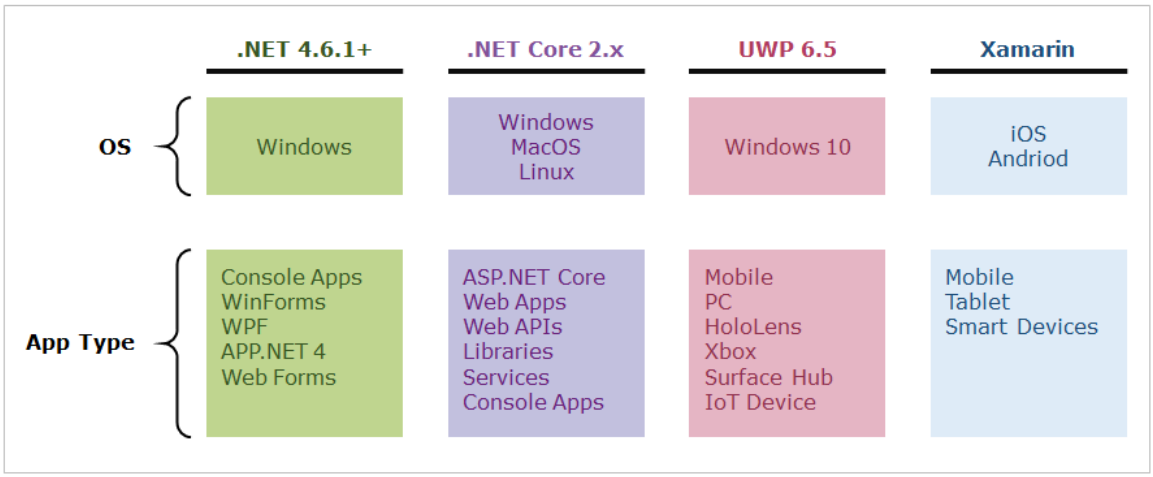
Controllers are the central piece and core of any MVC Application, if your website is a Hotel, then controllers is the main receptionist/manager. Controllers interact with any http/s request which is coming through the channel and makes all the decisions regarding which model to select, and then take data from that respective model to a particular view which user is expecting to be directed to. Essentially, controllers simply control the entire traffic through the application and maintain the overall flow. Most websites map requests to disk files on the server but in our case, MVC maps URLs to methods

Controller in ASP.NET MVC Architecture is a class which has been derived from the base class System.Web.Mvc.Controller, in this architecture every class name must be concatenated with “Controller”. For example, if your controller name is User, then it must be names UserController in the class.

1.4.4 Entity Framework Core

Entity framework enables working with data using objects of various user-defined classes without over interacting too much with database columns, it is open-source and an ORM Framework, ORM meaning object relational mapping, it enables working with higher level of abstraction. Compared to traditional applications, it involves lesser code and programmer can maintain the application almost effortlessly.

According to Microsoft, "Entity Framework is O/RM (Object-relation mapper) that enable .NET programmers/developers to interact with database using .NET objects and it further eliminates the need for data-access layer code which programmer would have had to write.". Now we don't need to write the lengthy code to access and database and we can simply get and .edmx format code automatically. We do now not need to understand the square to apply the entity framework.



1.4.5 HTML5

HTML5 (Hyper-text markup language) is used to describe web pages. It enables developers to do most of the things without additional software like browser plugins, from integrating animations to sound (howler JavaScript), it can be used design even the most complicated applications that run on browser. All browsers (Chrome, Safari, Opera, Firefox, Edge) support html5.

It consists of series of short codes that can be written in simple text-file. These short codes are known as tags and every html element is wrapped around it. For example, if u have a heading, it is wrapped around a tag like <h1> “Heading” </h1>. Similarly, there are various other tags like form, input, button, p, strong, italics and many more.

1.4.6 Cascading Style Sheet

CSS is tailor-made for a sole purpose of managing your application’s appearance in an efficient manner, maintaining a proper structure of each and every HTML element, whether it an image, a button or even a simple text, css can change appearance of any html element.

A CSS can be applied to a web page in 3 ways:

1. External CSS: A Separate file must be created and link to this file must be provided in the head section of your html file, in the external file all your styles are provided by selecting them by either classes or ids. Most Experts recommend this as it is easier to manage in case there are any changes to be made in the future.

2. Internal CSS: All the styling is done in the same html file, inside the head tag, a style tag is provided and elements can be selected directly or by using classes or ids. This is not recommended as all code is in a single file and harder to maintain.

3. Inline CSS: This is only used when there is very little styling need to be done, styles can be provided directed inside the tag using style attribute.

1.4.7 JavaScript and AJAX

JavaScript is a dynamin language which is renowned for its lightweight and it is well known for utilizing as a bits of site pages. It is a deciphered programming language. Client side JavaScript is the most broadly perceived kind of the language. DOM Manipulation is a feature in which files are used or referenced by HTML document for the code to be deciphered by the program. It suggests that a page need not be a static HTML, yet can fuse extends that partner with the customer, control the program, and capably make HTML content.

The JavaScript client side provides various positive conditions over standard server-side substance. JavaScript can be used to check if the customer has entered a correct email address (Format) in a structure field. The JavaScript code is executed when the user introduces the structure, and just if all of the entries are considerable, they would be submitted to the Web Server.

Chapter-2 LITERATURE SURVEY

2.1 Introduction to Literature Survey

Since there is now a flood of technologies which can be used to develop web application, one must research various technologies to determine which development environment is suited best for the project under consideration.

2.2 Summary of Lex Courses

2.2.1

Title: Implementing Database Using SQL Server 2012

Authors: Bavani Srinivasan, Prateem Chakraborty, Girija Prasad, Neha Thakur

Publishing: Lex

Summary: SQL Server also referred to as Microsoft SQL Server or MSSQL is a relational database management system which is developed by Microsoft. It is used for storing the data and accessing/retrieving the stored data from the database whenever requested by some application. SQL Server gives a very competitive competition to its competitors like MySQL or Oracle Database. With the help of SQL Server, we can create or develop databases for applications that belong to different domains such as retail, health care, financial services, education, insurance, marketing, etc. SQL Server provides us with the capability to declare any number of variables, functions of different types such as scalar functions, inline table-valued functions, multi-statement table valued functions, etc and user-defined stored procedures. It also provides the developer with the ability to do exception handling in functions as well as in stored procedures. SQL Server Management Studio (SSMS) acts as the main Graphical User Interface (GUI) for SQL Server. It was included as a tool in the SQL Server 2005 and its later versions for the purpose that it will help in the configuration, management, and administration of all the components that are there within the SQL Server. It can support the 32-bit as well as 64-bit environments. It helps its users in visualizing and analyzing the query plans and thus helps them in optimizing the database performance further. It can be used to create a new database as well as alter/modify an existing database schema if a user wants to add or modify some tables or indexes. It also has a query window which provides its users with a GUI based interface where they can write and execute their queries.

2.2.2

Title: Entity Framework Core 2.0

Authors: Kumudayani Krishnappa, Swetha Sree Shridar, Varun Verma

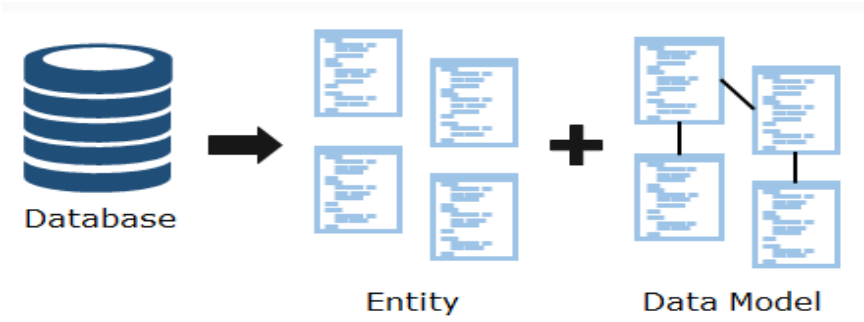
Publishing: Lex

Summary: Entity Framework Core (EF Core) is the latest version of Entity Framework from Microsoft. It is an open-source framework which is being used by the developers around the world. It is designed in such a way that it is lightweight, extensible and supports the cross-platform development as a part of Microsoft's .NET Core Framework. It is also simple to use and has much better performance than earlier versions of Entity Framework. Entity Framework Core is an object-relational mapping (ORM) framework. Object-relational mapping enables the developers to work with data in an object-oriented way by performing the work required to map between objects defined in an application's programming language and data stored in relational data sources. In other words, it is an enhancement to .NET that provides developers with an automatic mechanism for storing and accessing the data from the relational databases easily. We are using Entity Framework Core instead of Entity Framework because Entity Framework can be used to develop and deploy applications only on Windows platform but Entity Framework Core can be used to develop and deploy applications across various platforms such as Linux, macOS, Android, and Windows. We can develop mobile applications on Xamarine for Android, iOS and Windows can also be developed using Entity Framework Core.

We learned that there are two approaches for building/developing a Data Access Layer(DAL) for a .NET Core based application which are as follows:

Database-First Approach

In the Database-First Approach with the help of Entity Framework Core API and Entity Framework Core commands we create an Entity Data Model(EDM) based on an existing database. In the EDM, the Entity represents a C# class which is used for representing a table in the existing database and the Data Model represents the relationship between the C# classes.



Scaffolding is done in order to create an Entity Data Model by executing the scaffold-DbContext command in the Package Manager Console or the CLI and its syntax is as follows:

Scaffold-DbContext [-Connection] <String> [-Provider] <String> [-OutputDir <String>] [-Context <String>] [-Schemas <String []>] [-Tables <String []>] [-DataAnnotations] [-Force] [-Project <String>] [-StartupProject <String>] [<CommonParameters>]

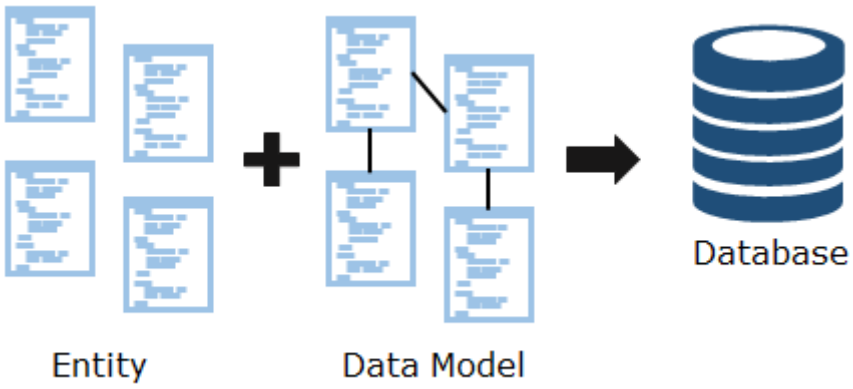
In this command the parameters that are mandatory are:

1. The -Connection<String> parameter is used to establish a connection with the existing database.

2. The -Provider <String> parameter represents the database provider name that has all the run-time libraries required to interact with a specific database. In our case it is Sql Server.

Code-First Approach

In the Code-First Approach, we first create all the required entity classes and context with the help of any language like C# or VB.NET which are .NET compatible. With the help of the EF Core API, a database schema and tables are created based on the conventions and configurations provided in the entity classes.



The Code-First Approach is a domain-driven approach as the total focus of this approach is to create the entity classes for the application domain rather than designing the database first and then creating the classes and model that match the database design as we do in database-first approach.

In our training, we learned how we can build/develop a Data Access Layer(DAL) for a .NET Core based application. We also learned how we can build/develop a Data Access Layer(DAL) for a .NET Core based application using the Database-First Approach.

2.2.3

Title: Presentation Tier Using ASP.NET Core

Authors: Aswin Thiyagarajan, Kaarthika Nataraj, Bavani Srinivasan, Antonita A.

Publishing: Lex

Summary: ASP.NET Core MVC is a rich framework which is used by the developers all over the world for building or developing web applications and API's using the Model-View-Controller design pattern. The advantage that ASP.NET Core holds over ASP.NET is that it can run fluently on platforms like Linux, macOS, and Windows whereas ASP.NET can run only on ASP.NET. During our training, we learned how we can develop the presentation tier of an enterprise application using ASP.NET Core MVC which will help us in creating web application that can be deployed on cloud or on premise.

Model-View-Controller is an architectural design pattern which is used for developing enterprise applications. It is considered one of the best design approach because it separates different aspects of an application such as input logic, business logic and user-interface logic with side by side providing a loose coupling between these elements. Now this separation of these elements provides us with easier change management and high testability of the application. The reason we have used the MVC design pattern is that it is difficult to make any changes in the non-MVC applications because the presentation and the business logic are tightly coupled but we can do the same changes with minimal coding efforts by using the MVC design pattern as in MVC based applications the presentation layer and the business logic are loosely coupled. So the ASP.NET Core framework with the MVC (Model-View-Container) design pattern approach is one of the best choices that we can use to implement various requirements specified in our SRS (Software Requirements Specification) as we want to create an application that works fluently even at peak loads and if ever we need to accommodate any changes then we should be able to do so with minimal effort.

Chapter-3 SYSTEM DEVELOPMENT

3.1 Functional Requirements

3.1.1 Requirements in Scope

REGISTER:

- Trainees have to register themselves to use the application
- Official email id should be used while registering
- Trainees can give course preferences while registering, which can be used for notifying them when a query is posted for the same
- Logging in should be mandatory for posting a question and answering
- Experts should have a different interface with consistent data communication between the application

TRAINEES:

- Trainees can see all the posted queries and answer
- Filtering the queries based on courses/topics should be made available
- An option of 'Report irrelevant' should be available, that can be used for reporting inappropriate questions/answer
- 'Notify me' option for questions that are yet to be answered
- Trainee will be notified once the answers are available for the question which he/she has selected 'Notify me' option
- They can post both question and its corresponding solution
- Appreciating a specific post can be done by an option of giving a star (*)
- Answering a query and posting a query requires a trainee to login
- Query with solution or query or answering a query, should have a status of 'Submitted to Experts'
- The status changes from 'Submitted to Experts' to 'Verified' when the same is verified by the experts
- Trainees whose queries/answers that are reported as 'Report irrelevant' with valid reason may or may not be blocked for further use of the application (decided by the experts)

EXPERTS:

- Experts can approve the posts from the trainees
- Rejection of posts happen when the same post exists already
- 'Report irrelevant' option should be taken care for incorrect queries/answers, irrelevant topics, unpleasantly words/statements etc.
- Blocking of trainees can happen for valid reasons like usage of abusive words etc
- Experts can rate a question/answer on a scale of 5
- Experts has to be notified whenever a new post is posted
- Screen for viewing all the posts should be available
- Filtering all the available posts based on criteria's like approved, rejected, all etc
- Experts can generate a report of the happenings in the application
- Certificate should be generated for the trainees who are topped in leader board

GENERAL:

- Leader board which showcases the statistics
- Example: star of the day, star of the month, overall star performer
- The following categories can be used for deciding the performers number of posts, rating received from experts, appreciation stars etc
- Certificates can be downloaded only by the logged in trainees who are also in the top 3 of any leader board stats

REQUIREMENTS OUT OF SCOPE

- Posting a question in the video format
- Coding environment for practicing in order to get the solution for queries

USER INTERFACE

- Should be easy to understand and user friendly
- Should adapt to different screen sizes

3.1.2 Non-Functional Requirements

SECURITY

- Application should be available only within intranet. Should not be available in VPN too.

AUDIT TRAIL

- Logs should be maintained for the changes being done to questions/queries. The timestamp and person responsible should be saved for both of them.

HARDWARE AND SOFTWARE REQUIREMENTS

Following requirements can be provided by the developer based upon the selected technology.

DEPLOYMENT ENVIRONMENT REQUIREMENTS

Hardware requirements

- Processor/RAM/HDD : 2 or 4GB RAM/NA
- Web server : Any suitable servers can be used
- Database Server : Any suitable servers can be used

Development Environment Requirements

- IDE : Eclipse with neon and above

Chapter-4 PROJECT DEVELOPMENT

4.1 Database Implementation

- User Table:
    1. UserName: Name using which user will register/login.
    2. FirstName: First Name of user.
    3. LastName: Last Name of user.
    4. DateofBirth: Date of birth of user.
    5. Password: Account password of user.
    6. Preferences: All the Categories which user prefers (java, html etc.)
    7. RoleId: 1 meaning trainee and 2 meaning expert.
    8. AccessId: Wheather the user is blocked or not.
    9. CorrectAnswers: Number of Answers of user verified by expert.
    10. CreatedOn: The date user is created on.
    11. Flag: Manages notification.
    12. QuestionIds: Manages all the questions which have been marked for notification.

- Questions Table
    1. QuestionId: Primary Key, Id of Question
    2. UserName: Foreign Key, Defines the owner/poster of question.
    3. Content: Title and explanation of question.
    4. Categories: In which category/s the question has been posted.
    5. CreatedOn: Date on which question has been posted
    6. ModifiedOn: If the question has been edited.
    7. RelevanceStatus: Wheather the question is relevant or not.
    8. QuestionStatus: Wheather it is answered or not.
    9. ExpertStatus: Wheather the question is approved, submitted to experts or rejected.
    10. Appreciation stars: Average of stars given to question by trainees
    11. ExpertRating: Average rating given by experts

- Answers Table
    1. AnswerId: Primary Key, Id of Answer.
    2. QuestionId: Foreign Key, Id of question to which it is an answer to.
    3. UserName: Foreign Key, Owner/Poster of the answer.
    4. Content: The Actual answer.
    5. CreatedOn: Date and Time of submission.
    6. ModifiedOn: Date and Time of when answer was edited last.
    7. ExpertStatus: Wheather the answer is approved, submitted to experts or rejected.
    8. RelevanceStatus: Relevant or irrelevant.
    9. AppreciationStars: Average of stars given by trainee to a particular answer.
    10. ExpertRating: Average Rating given by expert to an answer.

- Categories Table
    1. CategoryId: Primary Key, Id of category
    2. CategoryName: Name of category.

- IrrelevantQuestions Table
    1. IrreleventQid: Primary Key, Id of question marked irrelevant.
    2. QuestionId: Id of original question
    3. MarkedBy: User which has reported the particular question.
    4. MarkedOn: Date and time on which question has been reported
    5. Reason: The person using must explain his reasons for reporting question.
    6. Description: In case the reason is not pre-defined.

- IrrelevantAnswer Table
    1. IrreleventAid: Primary Key, Id of question marked irrelevant.
    2. AnswerId: Id of original answer.
    3. MarkedBy: User which has reported the particular answer.
    4. MarkedOn: Date and time on which answer has been reported.
    5. Reason: The person using must explain his reasons for reporting answer.
    6. Description: In case the reason is not pre-defined.

- AnswersRatingLog_Trainee Table
    1. LogId: Id of the log.
    2. AnswerId: Id of answer which has been rated.
    3. UserName: User who has given the rating.
    4. AppreciationStar: Rating given (Trainee)

- AnswersRatingLog_Expert Table
    1. LogId: Id of the log.
    2. AnswerId: Id of answer which has been rated.
    3. UserName: User who has given the rating.
    4. ExpertRating: Rating given (Expert)

- QuestionsRatingLog_Trainee Table
    1. LogId: Id of the log.
    2. QuestionId: Id of question which has been rated.
    3. UserName: User who has given the rating.
    4. AppreciationStar: Rating given (Trainee)

- QuestionsRatingLog_Expert Table
    1. LogId: Id of the log.
    2. QuestionId: Id of question which has been rated.
    3. UserName: User who has given the rating.
    4. ExpertRating: Rating given (Expert)

4.1.1 Table Implementation

S.No	Field Name	Type	Size	Keys
1.	UserName	Varchar	40	Primary
2.	First Name	Varchar	30	NOT NULL
3.	Last Name	Varchar	30	NOT NULL
4.	DOB	Date	-	NOT NULL
5.	Password	Varchar	20	NOT NULL
6.	Preferences	Varchar	Max	NOT NULL
7.	RoleId	Tiny Int	-	NOT NULL
8.	Access Id	Tiny Int	-	-
9.	CorrectAnswers	Int	-	-
10.	CreatedOn	Date	-	NOT NULL
11.	Flag	Int	-	NOT NULL
12.	QuestionIds	Varchar	Max	-

User Table

S.No	Field Name	Type	Size	Keys
1.	QuestionId	Int	-	Primary
2.	UserName	Varchar	40	Foreign
3.	Content	Varchar	200	NOT NULL
4.	Categories	Varchar	Max	NOT NULL
5.	CreatedOn	Datetime	-	NOT NULL
6.	ModifiedOn	Datetime	-	NOT NULL
7.	RelevanceStatus	Varchar	15	NOT NULL
8.	QuestionStatus	Varchar	20	-
9.	ExpertStatus	Varchar	30	-
10.	AppreciationStar	Decimal	(2,1)	NOT NULL
11.	ExpertRating	Decimal	(2,1)	NOT NULL

Questions Table

S.No	Field Name	Type	Size	Keys
1.	AnswerId	Int	-	Primary
2.	QuestionId	Int	-	Foreign
3.	UserName	Varchar	40	Foreign
4.	Content	Varchar	1000	NOT NULL
5.	CreatedOn	Datetime	-	NOT NULL
6.	ModifiedOn	Datetime	-	NOT NULL
7.	RelevanceStatus	Varchar	30	NOT NULL
8.	ExpertStatus	Varchar	30	NOT NULL
9.	AppreciationStar	Decimal	(2,1)	NOT NULL
10.	ExpertRating	Decimal	(2,1)	NOT NULL

Answers Table

S.No	Field Name	Type	Size	Keys
1.	IrrelevantQid	Int	-	Primary
2.	QuestionId	Int	-	Foreign
3.	MarkedBy	Varchar	40	NOT NULL
4.	MarkedOn	Datetime	1000	NOT NULL
5.	Reason	Varchar	200	NOT NULL
6.	Description	Varchar	1500	NOT NULL

Irrelevant Questions Table

S.No	Field Name	Type	Size	Keys
1.	IrrelevantAid	Int	-	Primary
2.	AnswerId	Int	-	Foreign
3.	MarkedBy	Varchar	40	Foreign
4.	MarkedOn	Datetime	1000	NOT NULL
5.	Reason	Varchar	200	NOT NULL
6.	Description	Varchar	1500	NOT NULL

Irrelevant Answer Table

S.No	Field Name	Type	Size	Keys
1.	LogId	Int	-	Primary
2.	QuestionId	Int	-	Foreign
3.	UserName	Varchar	40	Foreign
4.	ExpertRating	Decimal	(2,1)	NOT NULL

Questions Rating Expert

S.No	Field Name	Type	Size	Keys
1.	LogId	Int	-	Primary
2.	QuestionId	Int	-	Foreign
3.	UserName	Varchar	40	Foreign
4.	Appreciation Star	Decimal	(2,1)	NOT NULL

Questions Rating Trainee

S.No	Field Name	Type	Size	Keys
1.	LogId	Int	-	Primary
2.	AnswerId	Int	-	Foreign
3.	UserName	Varchar	40	Foreign
4.	Expert Rating	Decimal	(2,1)	NOT NULL

Answer Rating Expert

S.No	Field Name	Type	Size	Keys
1.	LogId	Int	-	Primary
2.	QuestionId	Int	-	Foreign
3.	UserName	Varchar	40	Foreign
4.	Appreciation Star	Decimal	(2,1)	NOT NULL

Answer Rating Trainee

4.2 DAL (Data Access Layer)

Data Access layer is created by scaffolding the database. Scaffolding is done to create the basic CRUD Operations against your database.

4.2.1 READ OPERATIONS

- GetAllCategories: Returns list of categories
- GetUser: Find a specific user from the database.
- GetAllQuestions: Returns list of all question in database.
- GetQuestionById: Finds a specific question and return it.
- GetAllAnswers: Returns list of all answers in database.
- GetAnswersById: Finds all answers and return it.
- GetFirstAnswerById: Return First answer of a question.
- GetAllUsersByCategory: Returns a list of all users who follow a particular.
- GetAllQuestionByCategory: Returns all question in a category
- GetQuestionRatingsByExpert: Returns average rating of a question (given by expert).
- GetRatingStatusForExpertByQuestion: Returns wheather the question has been rated or not.
- GetAnswerRatingByExpert: Similar to question rating, it returns expert rating of a particular answer.
- GetAnswerRatingByTrainee: Returns Appreciation star (decimal) given by trainee to a particular answer.
- GetQuestionRatingsByTrainee: Returns Appreciation star (decimal) given by a trainee to a particular question.
- GetQuestionIdsByUser: Returns list of questions which have been marked by user to notify when there is any change.
- GetFlagForUser: Returns number of notifications for a user
- GetLeadersofAllTime: Return top users of all time
- GetLeaderMonthly: Return list of top user in a 30-day range of when function is called.

4.2.2 CREATE OPERATIONS

- AddNewUser: Adds a new user to database
- AddNotifyMe: Adds a flag for notifications in user database.
- AddNewQuestion: Adds new question posted to the database.
- AddNewAnswer: Adds new answers which have been posted to the database.
- AddQuestionRatingByTrainee: Appreciation stars given by trainee are added in decimal form.
- AddQuestionRatingByExpert: Expert Rating given by expert is added to database in decimal form.

4.2.3 UPDATE OPERATIONS

- UpdateFlagOfUser: This is automatically called whenever something of interest to the trainee/expert is changed (For example, a new question/answer of particular category posted), and flag count is updated.
- UpdateQuestionIdsofUser: Whenever a user marks something for notification, this function is called to update the questionids marked by user.
- EditQuestion: Used to update content of question
- EditAnswer: Used to update content of answer
- BlockUser: Used to update the Access Id of a particular user.
- MarkIrrelevantQuestion: Used to update the Expert Status of question to Rejected whenever an expert updates it.
- MarkIrrelevantAnswer: Used to update the Expert Status of answer to Rejected whenever an expert updates it.
- ApproveQuestion: Used to update the Expert status of question of verified whenever it is updated by the expert.
- ApproveAnswer: Used to update the Expert status of answer of verified whenever it is updated by the expert.

4.3 CONTROLLERS

Controllers control the flow of the application execution and returns different actions based on the request. Controllers provide a constant communication between View and the Model.

There are 3 controllers in our project:

1. User Controller
2. Trainee Controller
3. Expert Controller

4.3.1 User Controller

This controller is responsible for various functionalities such as

- Login / Logout functionality
- Register User functionality
- Validating the users

4.3.2 Trainee Controller

This controller is responsible for various functionalities such as

- Trainee can post questions, which will be answered by experts or their peers.
- Trainee can also post answers to questions asked by other users.
- Auto-complete function for categories is implemented when user enters any text.
- User can sort by categories.
- Functionality to view answers posted by trainee/expert
- Functionality to provide users with the notifications related to their category
- Provide option to the user to mark question/answer irrelevant
- Leaderboard which provides us with the top rated trainee which are calculated using the expert and appreciation stars having the option for weekly and monthly leaderboard.
- Option to download Certificate for the top 3 rated trainee

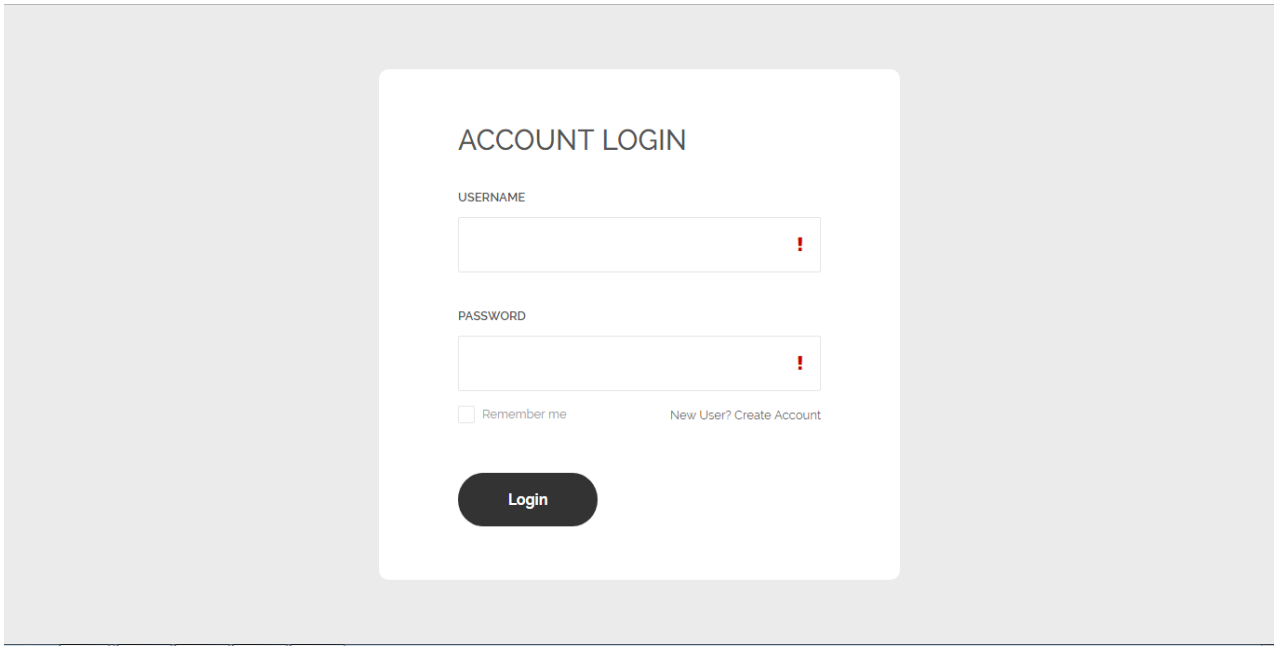
4.3.3 Expert Controller

This controller is responsible for various functionalities such as

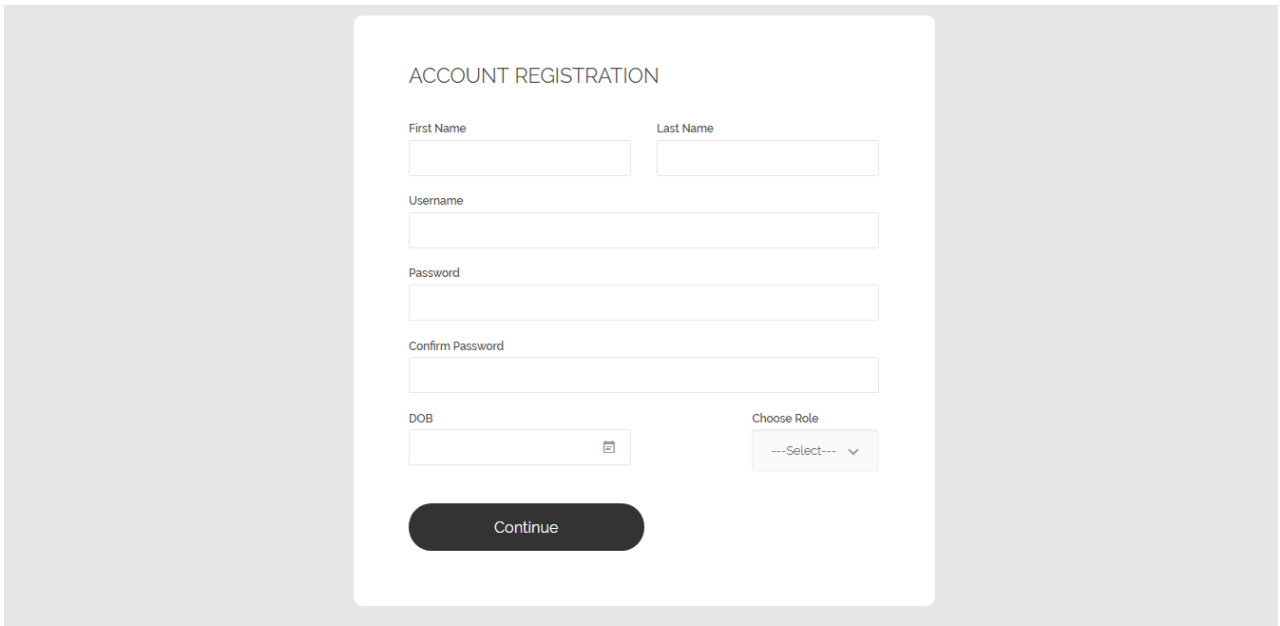
- Expert approving and rating the question
- Expert marking the reported questions as relevant or irrelevant
- Posting, Viewing and Editing the answers
- Marking the question as irrelevant or duplicate
- Marking the answer irrelevant
- Approving the question and answers
- Expert can view any user and also block any user
- Leaderboard which provides us with the top rated expert which are calculated using the expert and appreciation stars both for weekly and monthly leaderboard
- Option to download Certificate for the top 3 rated trainee

4.4 Views

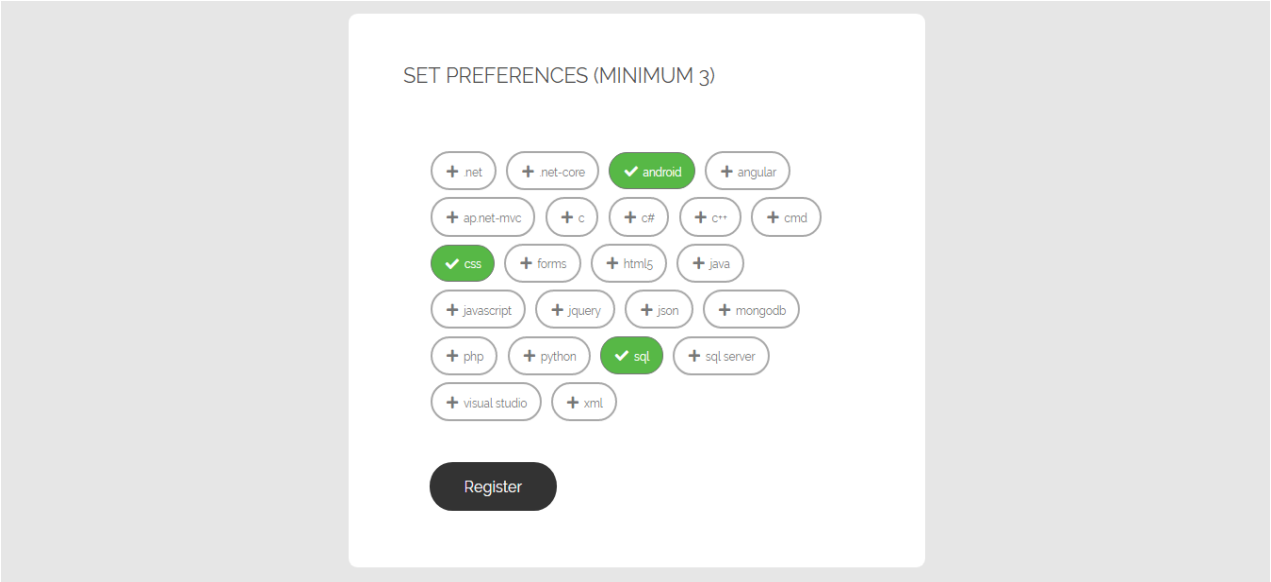
4.4.1 Login Page



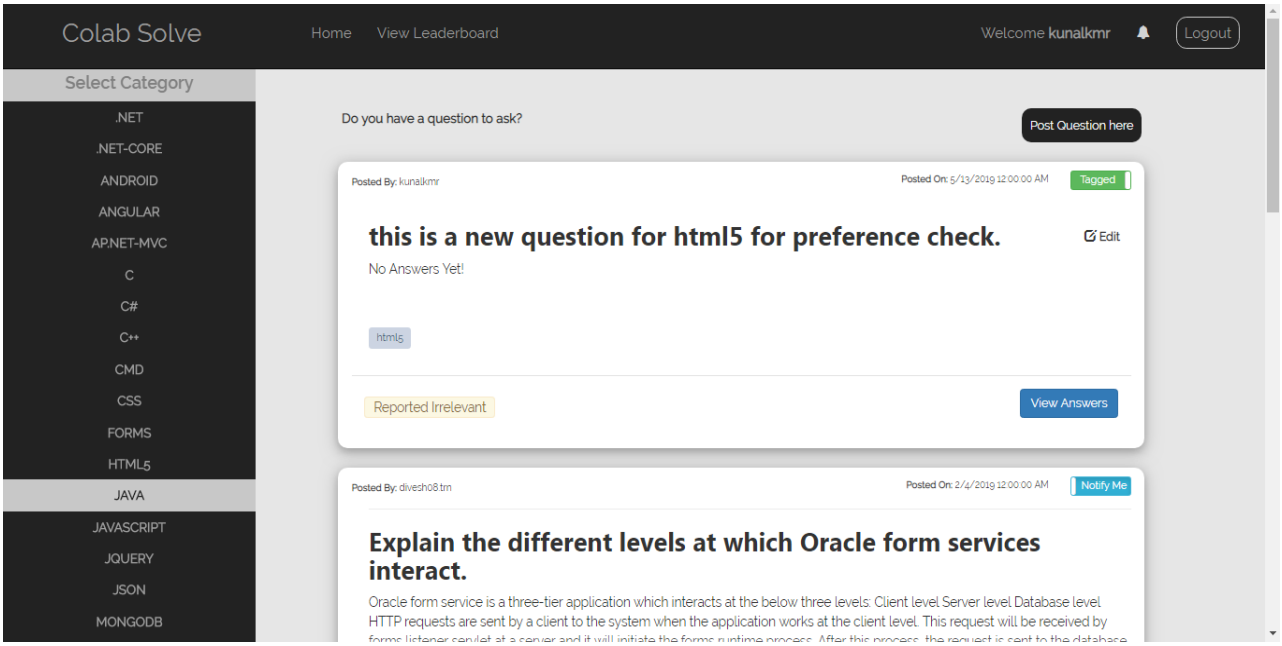
4.4.2 Register Page



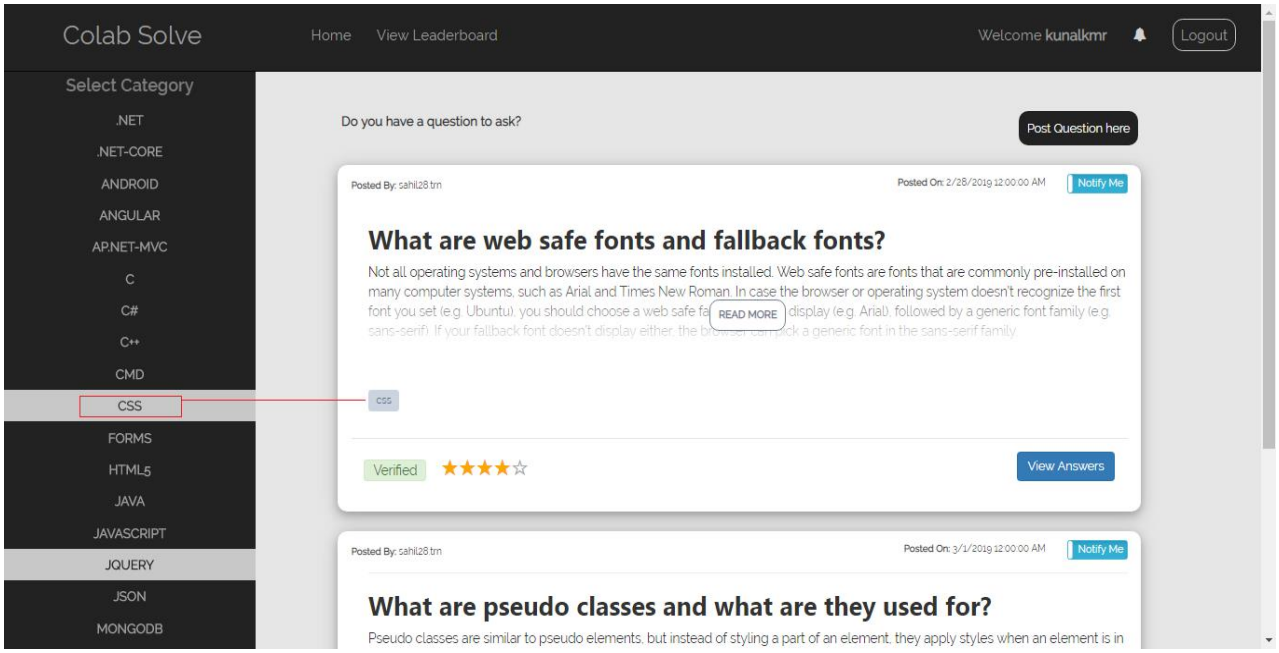
4.4.2 Register Phase 2



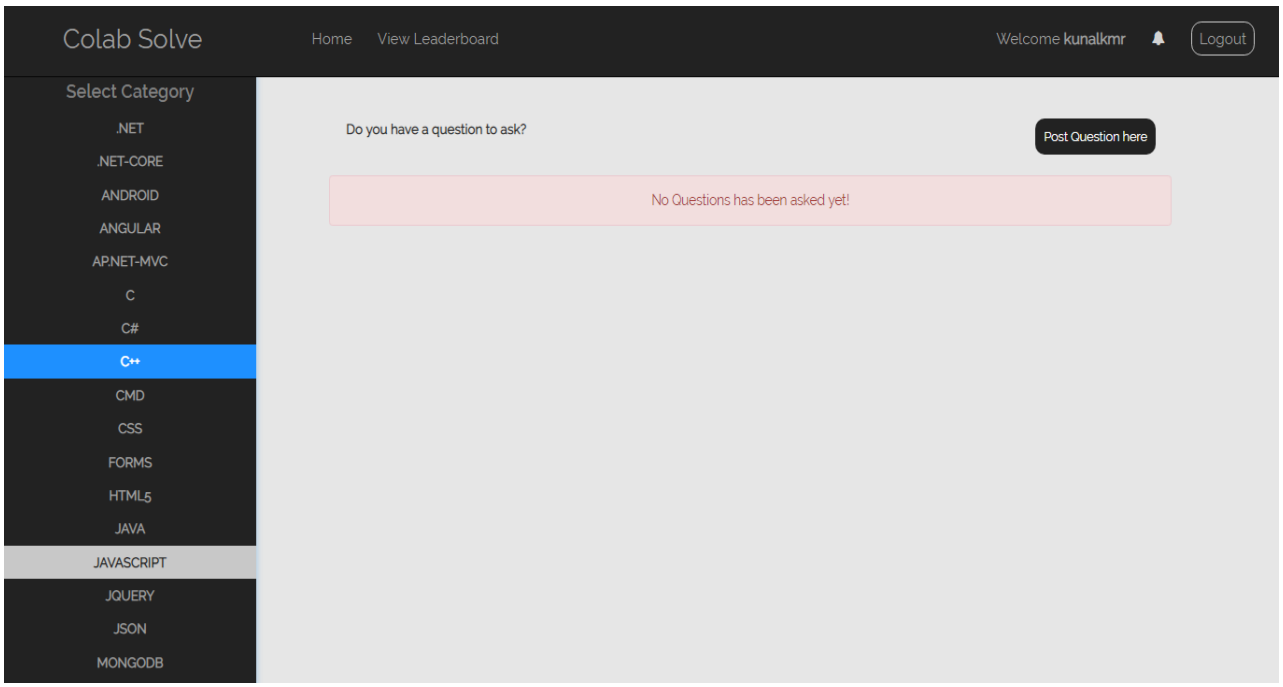
4.4.3 Trainee Home Page



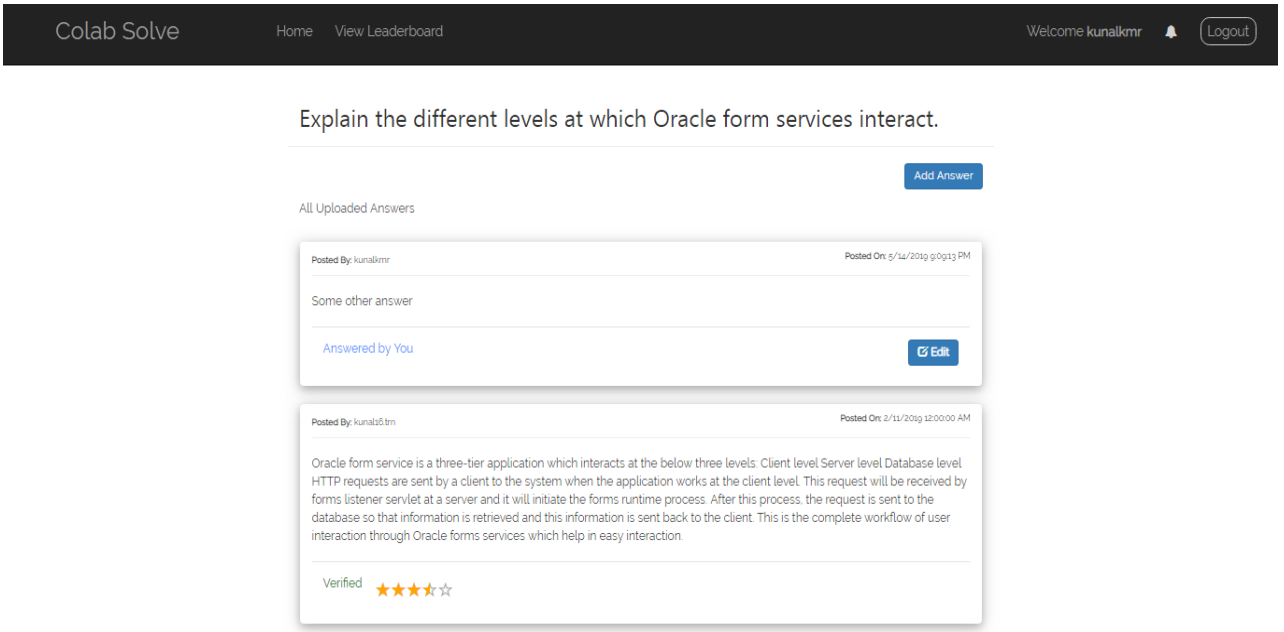
4.4.4 Trainee Selecting Specific Category



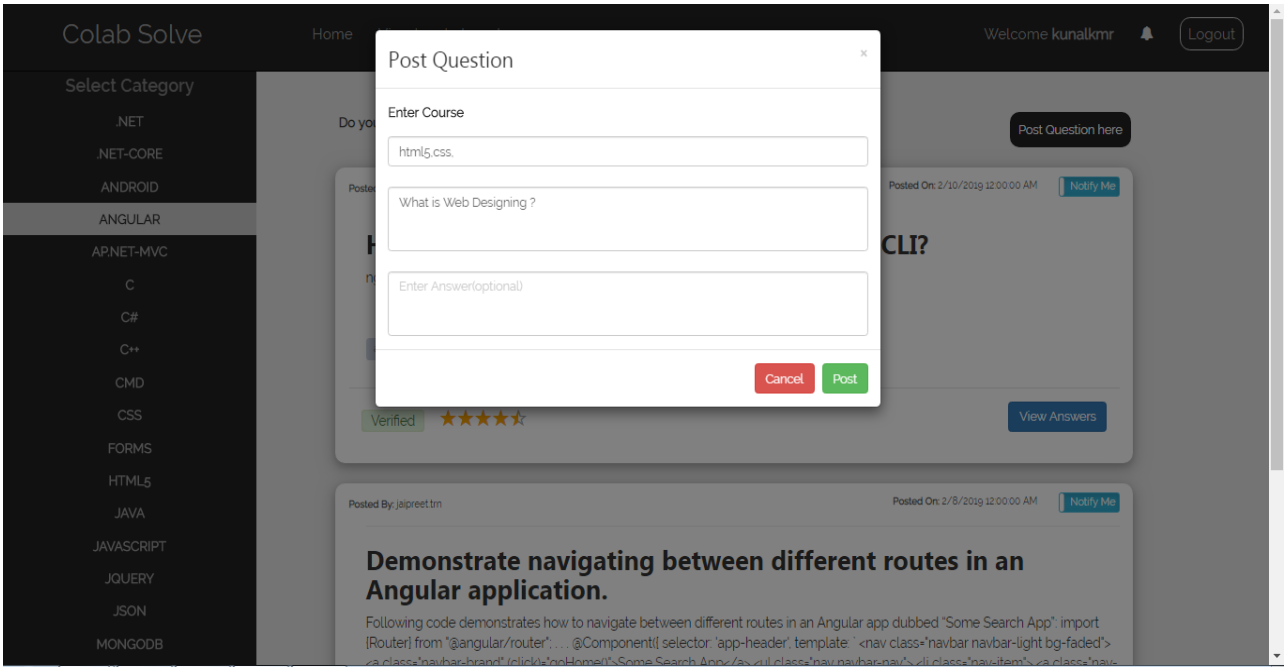
4.4.5 Category in which question is not posted



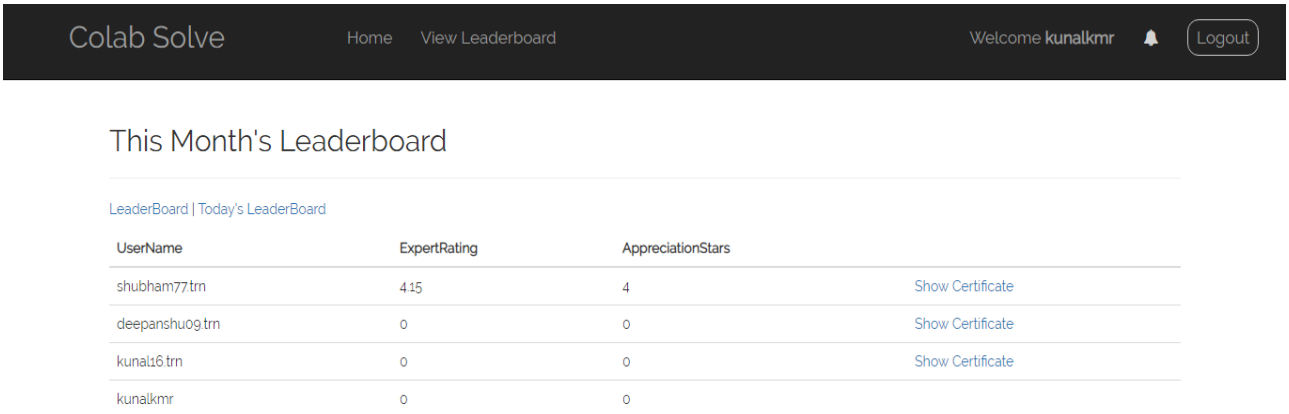
4.4.6 View Answers



4.4.7 Posting Question



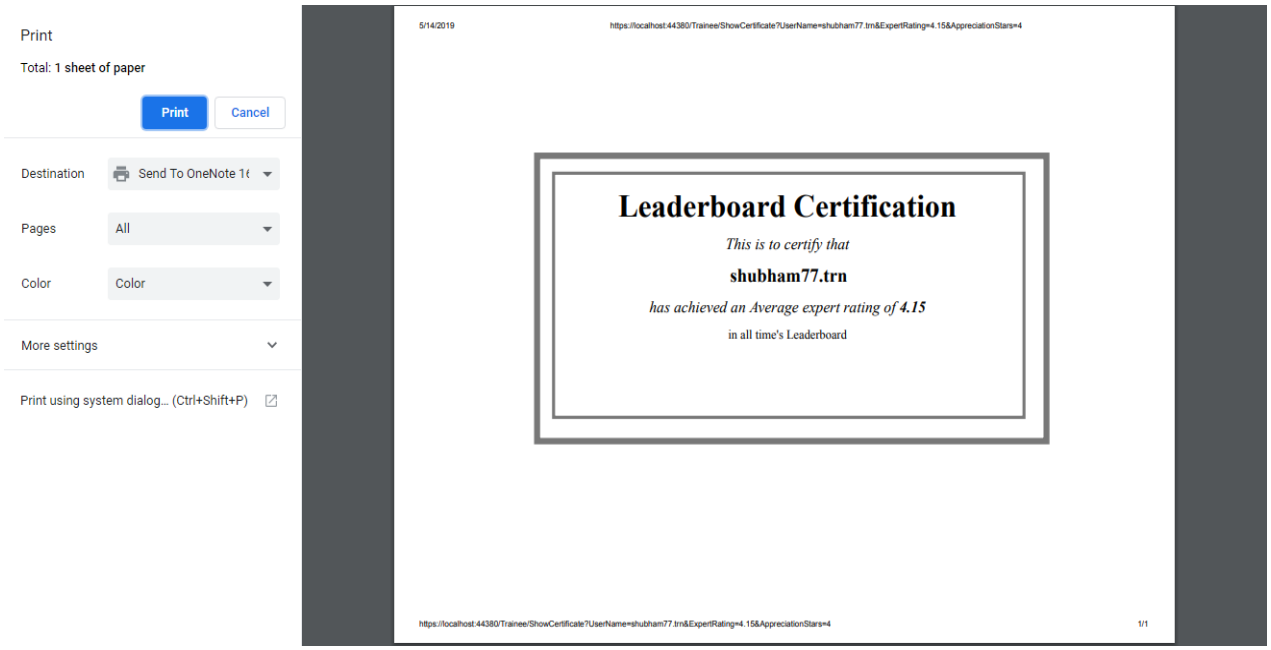
4.4.8 View Leaderboard



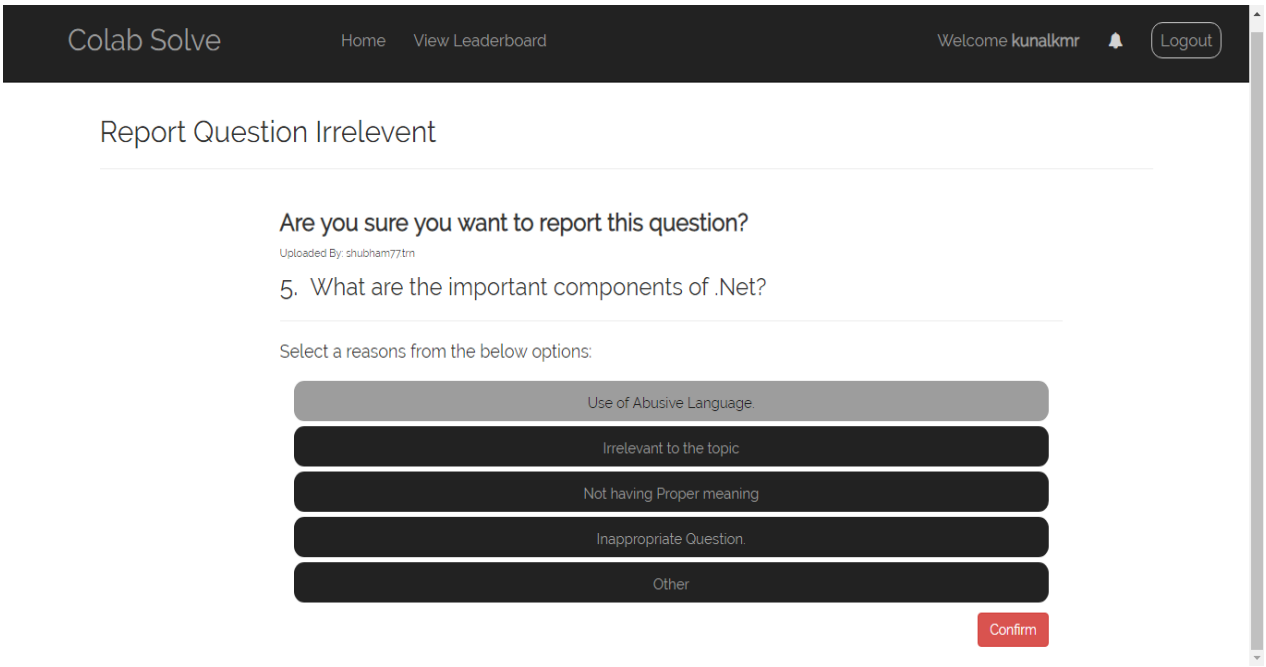
4.4.9 Certificate

4.4.10 Download Certificate

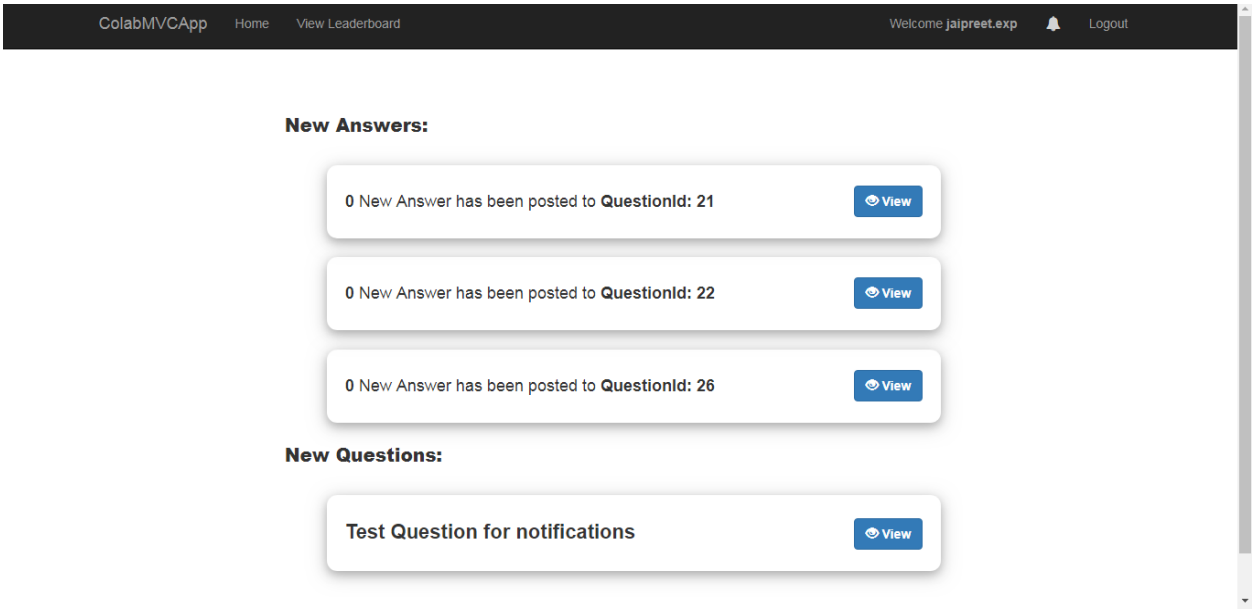
s



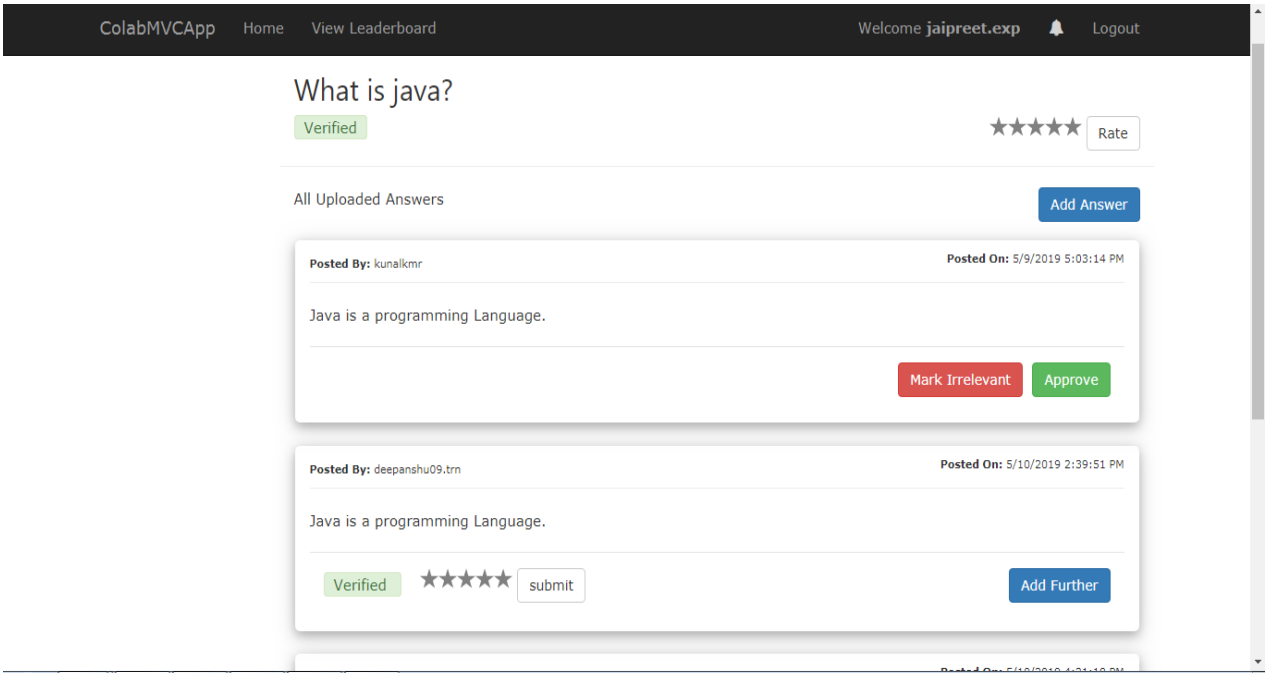
4.4.11 Reporting Question



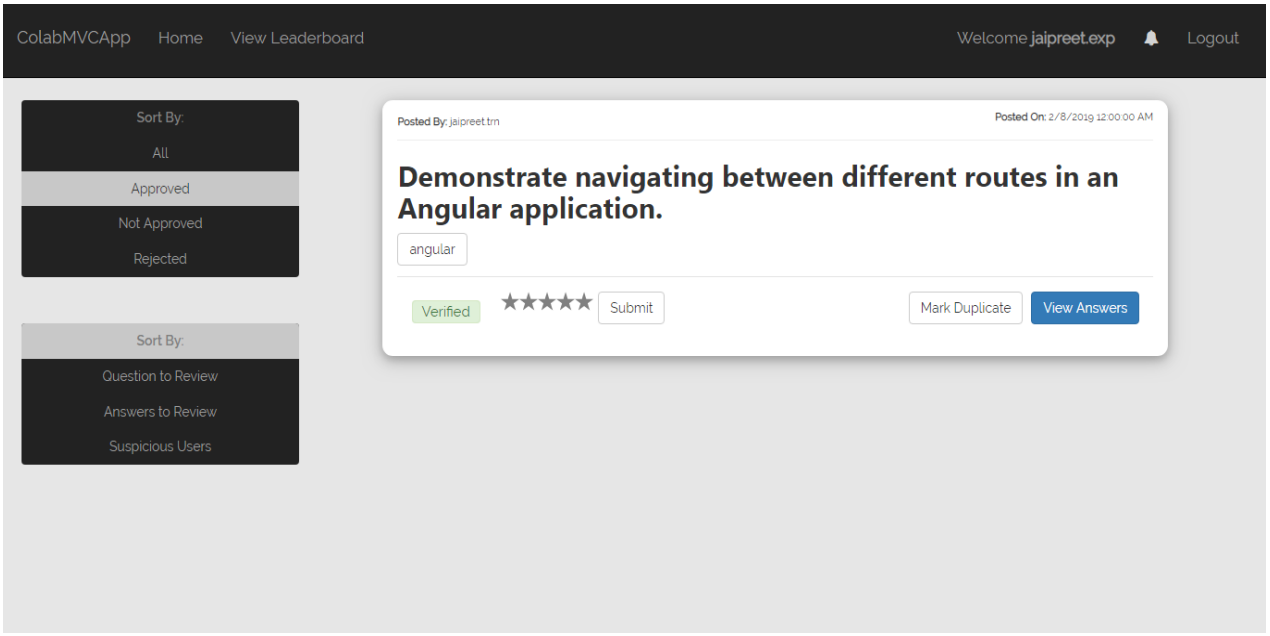
4.4.12 Expert Notification



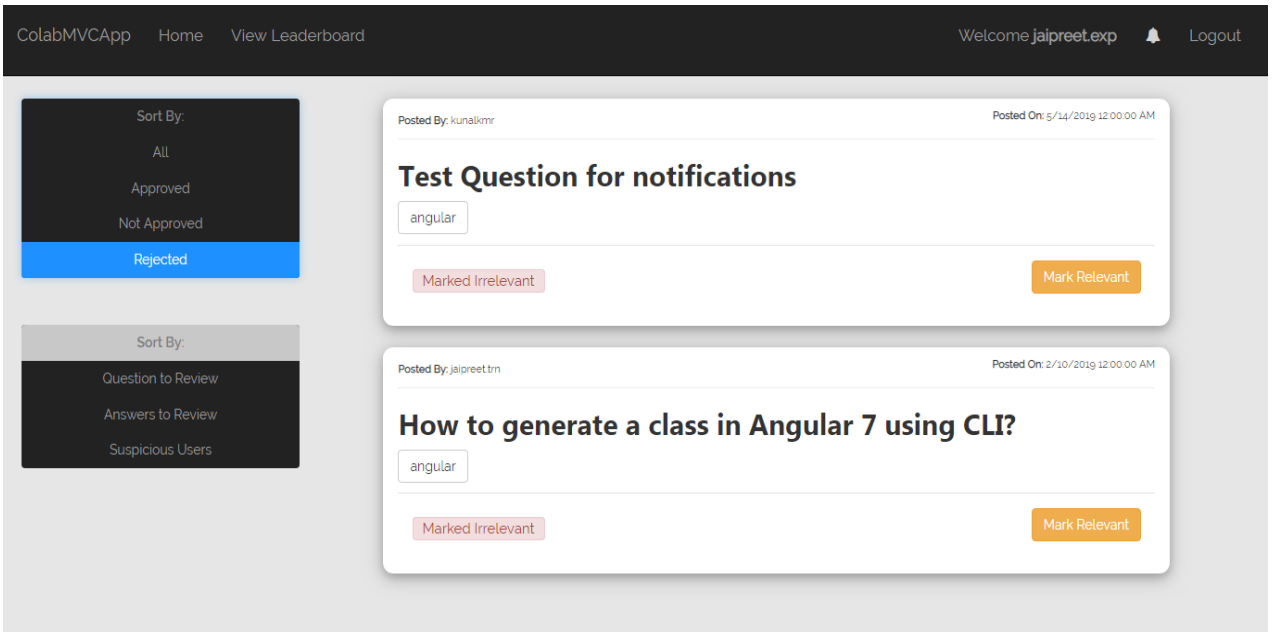
4.4.13 Expert Notification Detailed



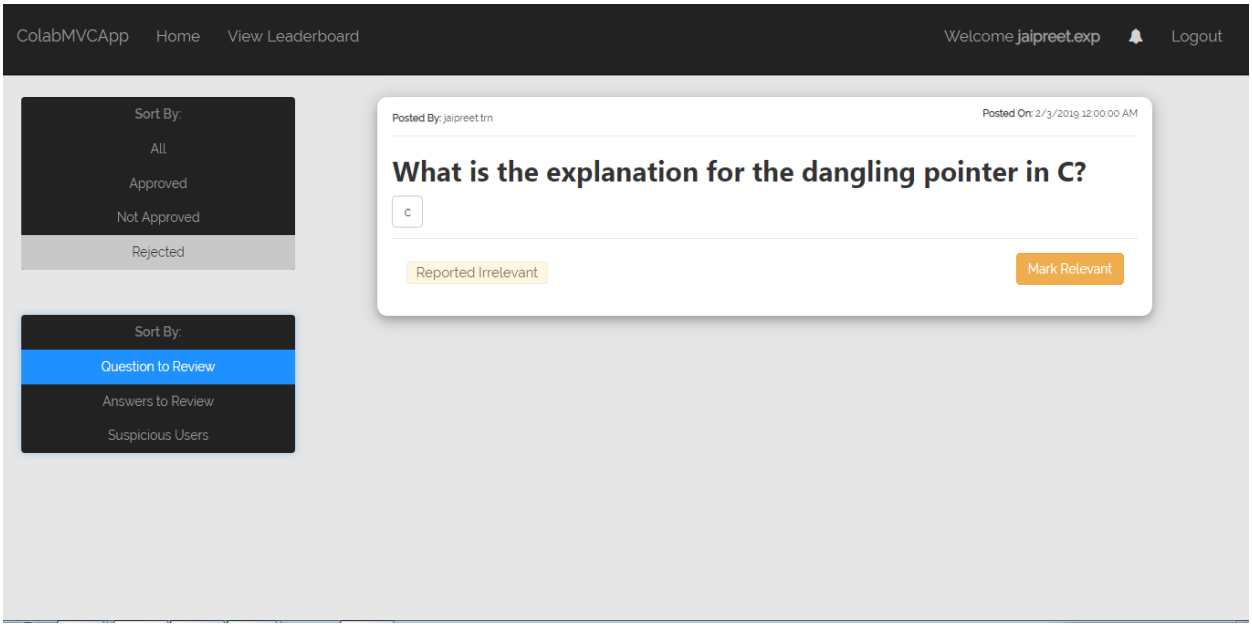
4.4.14 Expert Layout-Approved



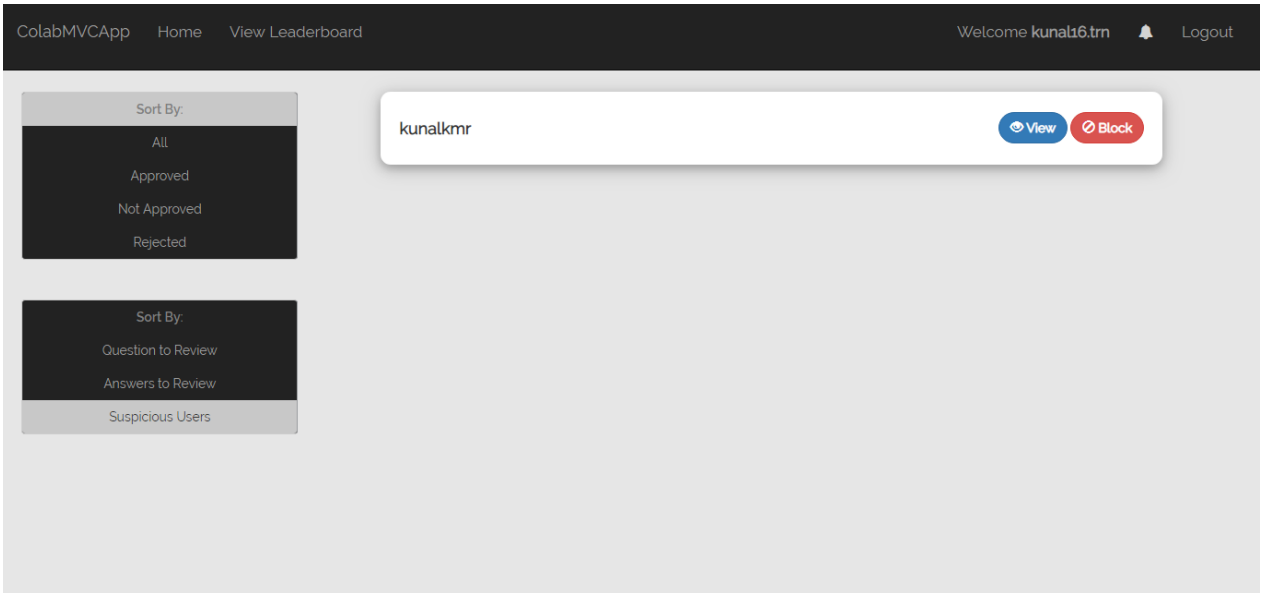
4.4.15 Expert-Layout-Rejected



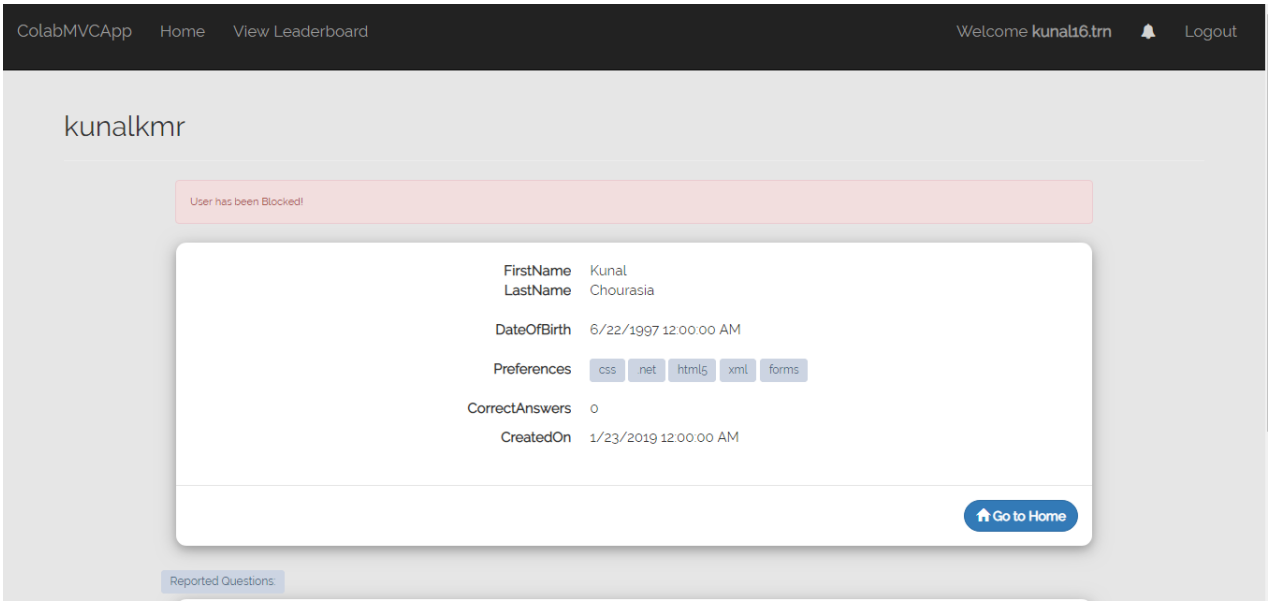
4.4.16 Questions Under Review



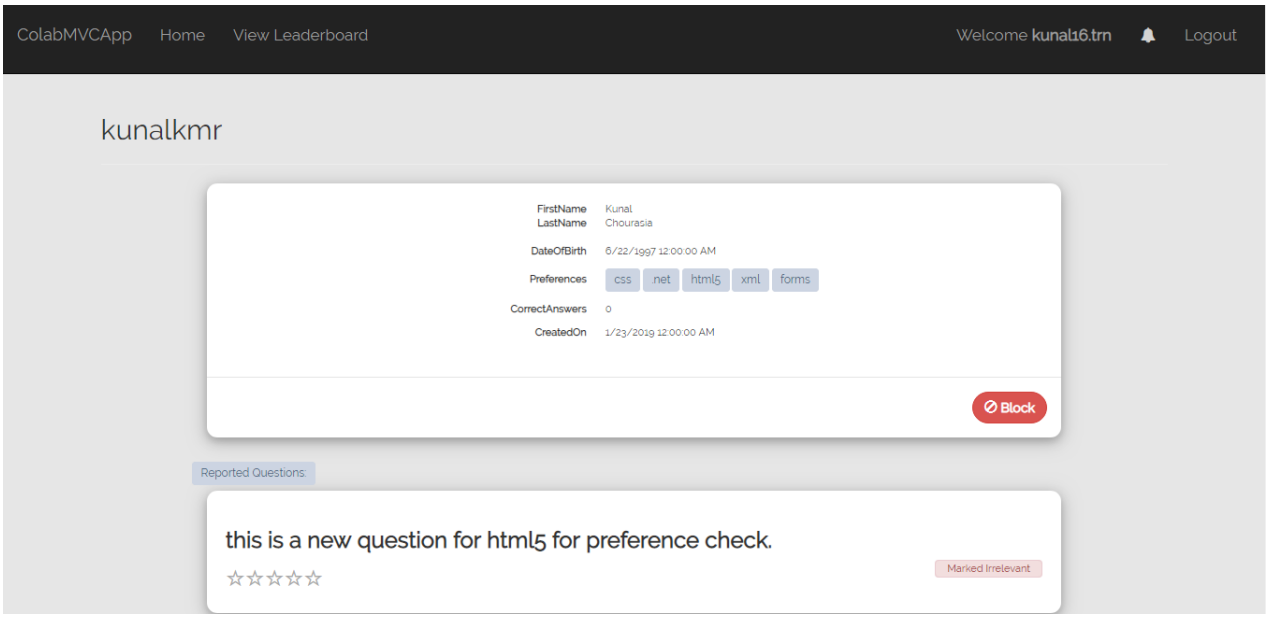
4.4.17 Block User View



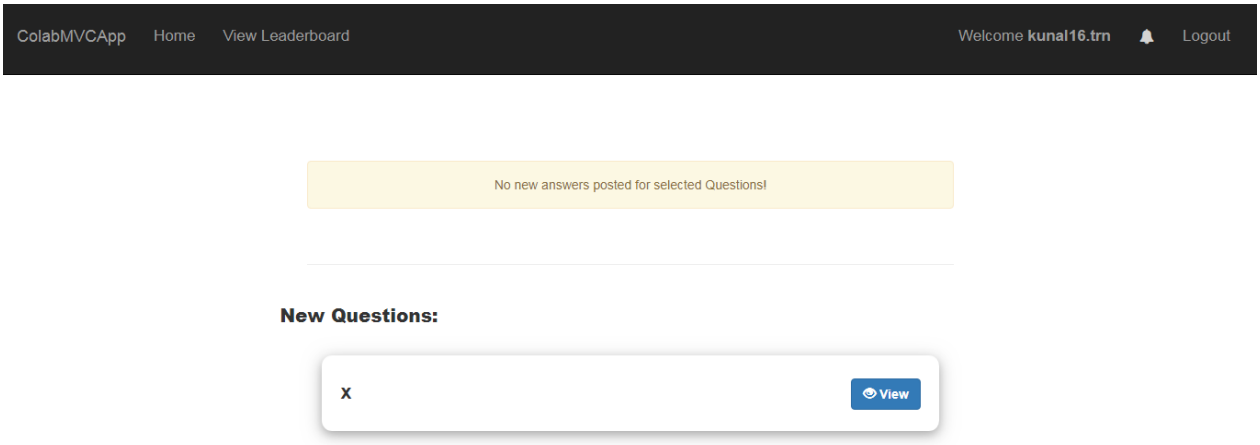
4.4.18 Block User-Details



4.4.19 User Profile



4.4.20 Expert Notifications



Chapter 5 Performance Analysis

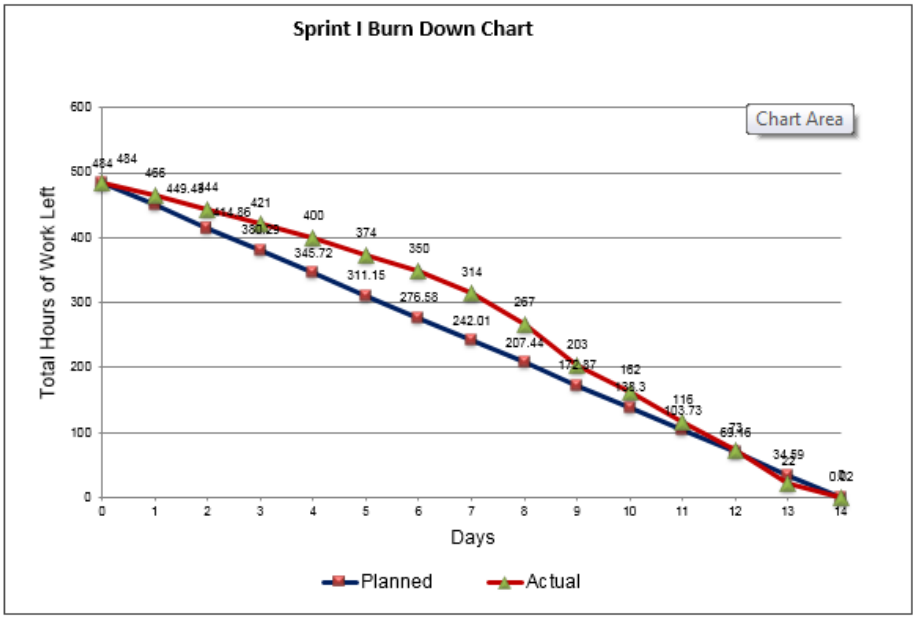
5.1 Agile Methodology

Agile technique is a practice that includes consistent cycle of undertaking improvement and support just as testing and act of spontaneity all through the product advancement life cycle of venture. In this technique, not at all like Waterfall model, the advancement and testing exercises are simultaneous and it might change as the prerequisite changes. Since in some random task, the necessities are liable to change with the client's interest, or fresh introduction of modules and so forth.

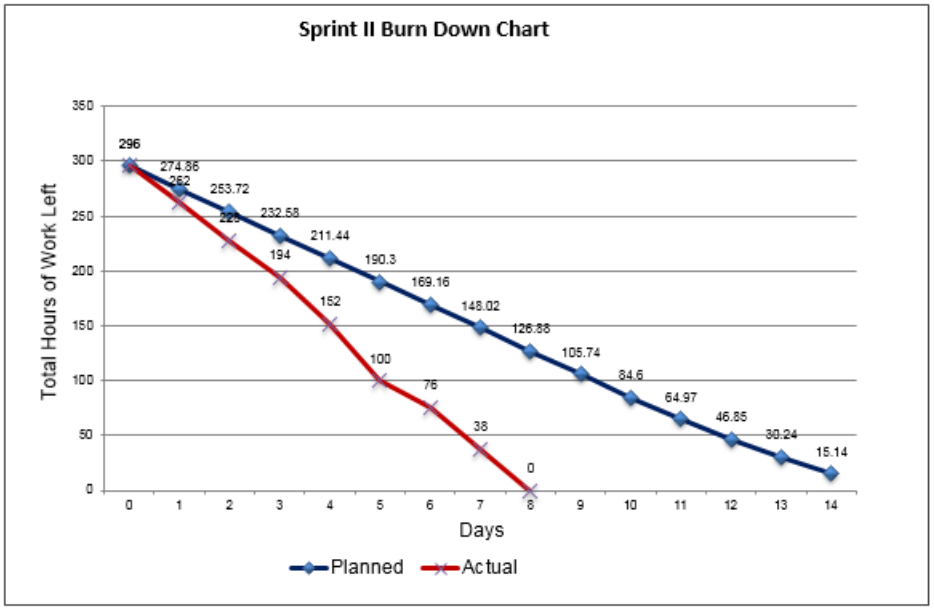
In this strategy, the entire life cycle of the venture is separated into Sprints, where you can partition the undertaking and toward the finish of the particular run, the diagram portrays the exhibition of that venture. Each Sprint comprises of Product excess spread sheet which contains client stories masterminded in sequential request. Every one of Sprint 1 and Sprint 2 are time bound with fitting due date. Each dash is of fourteen days. It is essentially made when one is working in a group or a gathering of individuals. It is principally used to break down individual exertion when working in a group and tells about the advancement of one. Dashboard indicates graphical portrayal of deferral or advancement concerning arranged exertion. This line is investigated regarding ideal line.

The nimble advancement lifecycle comprises of four center components:

1. Stand Meeting of Team Members with Scrum Master
2. Arranging venture into little pieces for example Run 1 and Sprint 2
3. Scope quantification
4. Retrospection
5. Dashboard



Sprint 1 Progress
COMPLETED



Sprint 2 Progress
DAY 9

5.2 Test Planning

5.2.1 Register

S.No	Test Case	Expected Result	Actual Result
1.	User: Jaipreet@gmail.com Pass: ace	FAIL (.trn required)	FAIL
2.	User: Jaipreet.trn@gmail.com Pass: ace	FAIL (Password annotation)	FAIL
3.	User: Jaipreet.trn@gmail.com Pass : Hawk@eye09	PASS	PASS
4.	User: Jaipreet.trn@gmail.com Pass : Hawk@eye09	FAIL (user already exists)	FAIL
5.	User: kunalkmr Pass: kunalkmr Confirm Pass: kunalkumar	FAIL (pass don't match)	FAIL
6.	User: kunalkmr Pass: kunalkmr Categories: java, css	FAIL (atleast 3 categories must be selected)	PASS
7.	User: kunalkmr Categories: java, css, html	PASS	PASS
8.	User: kunalkmr DOB: 17th May 2019	FAIL (dob must be less than current date)	FAIL

5.2.2 Login Page

S.No	Test Case	Expected Result	Actual Result
1.	User: Jaipreet@gmail.com Pass: ace	FAIL (wrong password)	FAIL
2.	User: Jaipreet1@gmail.com Pass: ace	FAIL (user does not exist)	FAIL
3.	User: Jaipreet.trn@gmail.com Pass : Hawk@eye09	PASS	PASS

5.2.3 Trainee Page

S.No	Action	Controller+Method	Result
1.	Post Question	Trainee Add Question	Question is posted in the database
2.	Click on any category	Trainee Sort By Category	Questions of that category displayed
3.	View Answers	Trainee Get Answers	All the answers of that question are displayed
4.	Click on notification	Trainee GetNotificationByFlag	All notifications for that particular user are displayed
5.	Click on Mark Irrelevent	Trainee Mark Irrelevant	Popup asking the user for his reasons of marking the question.
6.	Submission of Reason (Mark irrelevent)	Trainee Mark Irrelevant	Question marked and hidden until an expert decides on its authenticity.
7.	Click on Leaderboard	Trainee GetOverallLeader	A table is displayed for overall leader
8.	Click Leaderboard Daily Category	Trainee GetDailyLeader	A table is displayed for today.

9.	Click leaderboard Monthly category	Trainee GetMonthlyLeader	A table is displayed for last month
10.	Click on get certificate	Trainee Get Certificate	Certificate is displayed with an option to download.
11.	Click on download certificate	Trainee Windows.print()	Chrome pop-up pre-selected for download as pdf
12.	Click on add categories	Trainee Add Categories	A popup which enables user to update his categories
13.	Click on edit	Trainee Update Question/Answer	A trainee will be able to edit answer or question answered or asked by him.
14	Click on logout	Trainee Logout	Successfully logout.

5.2.4 Expert Page

S.No	Action	Controller+Method	Result
1.	Post Answer	Expert Add Answer	Answer is posted in the database
2.	Click on any category	Expert Sort By Category	Questions of that category (only is expert is pf that category) displayed
3.	View Answers	Expert Get Answers	All the answers of that question are displayed
4.	Click on notification	Expert GetNotificationByFlag	Notifications regarding which new

			question/ answer has been posted, or which user is suspicious
5.	Click on Mark Irrelevent	Expert Mark Irrelevant	The Trainee Status is changed to expert can then block it.
6.	Click on block user	Expert Block user	Access of that user is denied and he can longer login with his account
7.	Click on Leaderboard	Expert GetOverallLeader	A table is displayed for overall leader so that he can generate their certificate
8.	Click Leaderboard Daily Category	Expert GetDailyLeader	A table is displayed for today so that he can generate their certificate
9.	Click leaderboard Monthly category	Expert GetMonthlyLeader	A table is displayed for last month so that he can generate their certificates
10.	Click on View Approved/Rejected	Expert GetQuesionByStatus	Question which have already approved/rejected are displayed for assessment
11.	Click on Blocked user	Expert GetBlockedUser	List of all users who have blocked.
12.	Click on review answer/question	Expert GetNewQuestions/Answers	All the latest posts for that expert's categories

Chapter 6 CONCLUSIONS

Our application “Colabs-Resolving Problems” was up to the expectations of our mentors/idea-owner. All the user requirements were fulfilled and the application works well in the intranet environment of the organization. The Application has been well tested in different erroneous conditions and the errors have been handled properly in debugging phase. The Users can access the application from two different computers simultaneously and can login and logout successfully without facing any session error. The web Application works perfectly fine in different conditions of respective systems. It can be improved further and can be made more attractive and can offer additional functionalities in terms of analyzing the car breakdowns.

FUTURE SCOPE:

Since this is a web application, there are a loads of tiny additions which could be integrated in our application. From integrating attachment, so that videos and images can be posted and displayed which would make our application more interactive a user friendly. There is also a possibility of adding compilers, which is something most of the big websites of our category have implemented.

REFERENCES

[1] http://www.w3schools.com

[2] Design and Implementation of an MVC-Based Architecture for E-Commerce Applications by E. Althammer and W. Pree Published date 5th september 2013.

[3] Evaluation of the Model-View-Controller design pattern when applied to a heterogeneous application to distribute newspaper textual content to mobile devices”, Sakib Supple

[4] A Database and Web Application Based on MVC Architecture. Selfa, D.M., Carrillo, M., and Rocio Boone, M. Puebla, Mexico: IEEE, IEEE Int. Conf. on Electronics, Communications, and Computers (CONIELECOMP 2006).

[5] Domain Driven Web Development With WebJinn. Kojarski, S. and Lorenz, D.H. Anaheim, CA : ACM, OOPSLA 2003.

[6] https://www.c-sharpcorner.com/

[7] https://stackoverflow.com/