

CHAPTER I

INTRODUCTION

1.1 Overview

QnA Is All About Two Things: Questions and Answers

The question component of QnA is straightforward but shouldn't be overlooked. Asking questions is just as important as responding to them. You wouldn't use your business's Facebook account to only talk about you, right?

QnA works the same way. You have to be engaging with both your responses and your questions. Wording questions so that someone would be interested in answering them is critical. Viral questions can drive as much traffic to your site as a popular response can.

As for giving answers, there are many schools of thought about how to appropriately respond to a QnA question. Some resources will tell you short and succinct answers are ideal. While others will tell you, longer answers are important.

Ultimately, I believe in treating QnA like you would treat a blog post. The answer should be exactly as long as it needs to be to answer the question. You want to provide enough detail to properly answer the question without filling the response with unnecessary fluff. Here's my quick list of best practices for providing answers on Quora.

QnA is a site where people post answers to your questions. Its advantage lies in its community of reputable experts. I've found the quantity of questions to be quite diverse —QnA covers a wide variety of topics and interests, and is bound to have something for your curiosity. You can also choose to search for answers to specific questions right in the search bar on every page. One special thing about QnA is that you can create your own category, if you don't find relevant category for your question you can create your own.

1.2 Problem Statement

QnA is a knowledge exchange platform. Which can be share by anyone who has knowledge in something. Obviously sharing knowledge is a part of wisdom. It will reach to the people who has desire to acquire knowledge.

The basics of using QnA are as follows: you can choose to either browse questions and answers written by other Users, or ask a question of your own and wait for it to be answered. You can leave comments on answers if you wish, those are the basics.

QnA site can be home to hundreds of people legitimately searching for genuine answers to their most pressing concerns. That makes them a perfect avenue for finding highly targeted traffic with tremendous opportunities for conversion.

1.3 Objective of Project

We didn't actually say that this is a market that we want to be part of, and we weren't actually focused on the other Q&A sites. The way we think about this is there's actually a lot of information that's still in people's heads that's not on the internet. And when you think about it you would say that probably 90% of the information that people have is still in their heads, not on the internet. So we're trying to get that information out of people's heads, so it's not on sources that are hard to access on the internet, and get it into a really useful format to make a valuable database.

1.4 Application or Scope

QnA is a site where people post answers to your questions. It also allows you to follow Topics, People, and specific Questions, which is great for keeping up with trends and questions that you never ran into yet. Its advantage lies in its community of reputable experts. I've found the quantity of questions to be quite diverse —QnA covers a wide variety of topics and interests, and is bound to have something for your curiosity. You can also choose to search for answers to specific questions right in the search bar on every page

QnA is the Question and Answer web application and it is the one of the best platform or place where you can share your knowledge and get the best answer for your question.

1. In QnA application you just not become a good reader but also becomes a good writer.
2. It also aids you to replace the bad qualities like Unusual use of WhatsApp, Fb, Instagram, Skype or many other worst applications from your daily schedule.
3. It always finds very effective answer for your questions which is written on the basis of true experience and event.

Improves communication: It is one of those things intrigues me to write here.

Vocabulary gets sky-rocketed: Who doesn't need a good vocabulary. One can flaunt vocabulary and impress others, improve your choice of words and create a lasting impression over listeners. Vocabulary is something which can be an add-on for anyone ranging from student to professional.

Fun, entertainment and nice leisure activity: For someone like me who is not a bibliophile or a passionate reader to fall for Quora truly means that it surely gasses some magic, magic in words, magic in stories, magic in writing. More than that, all of it is in a very interactive and interesting way to kill boredom.

Knowledge gaining site: It clearly depicts how knowledge can be transferred in an effective as well as intriguing manner. World is full of knowledge but why choose the boring way when learning can be fun. Also, as Quora says it provides the best answer to any Question.

Life advices and experiences: The best of all, QnA users are a hell lot experienced in almost every field and the tips, experiences they pass on is priceless.

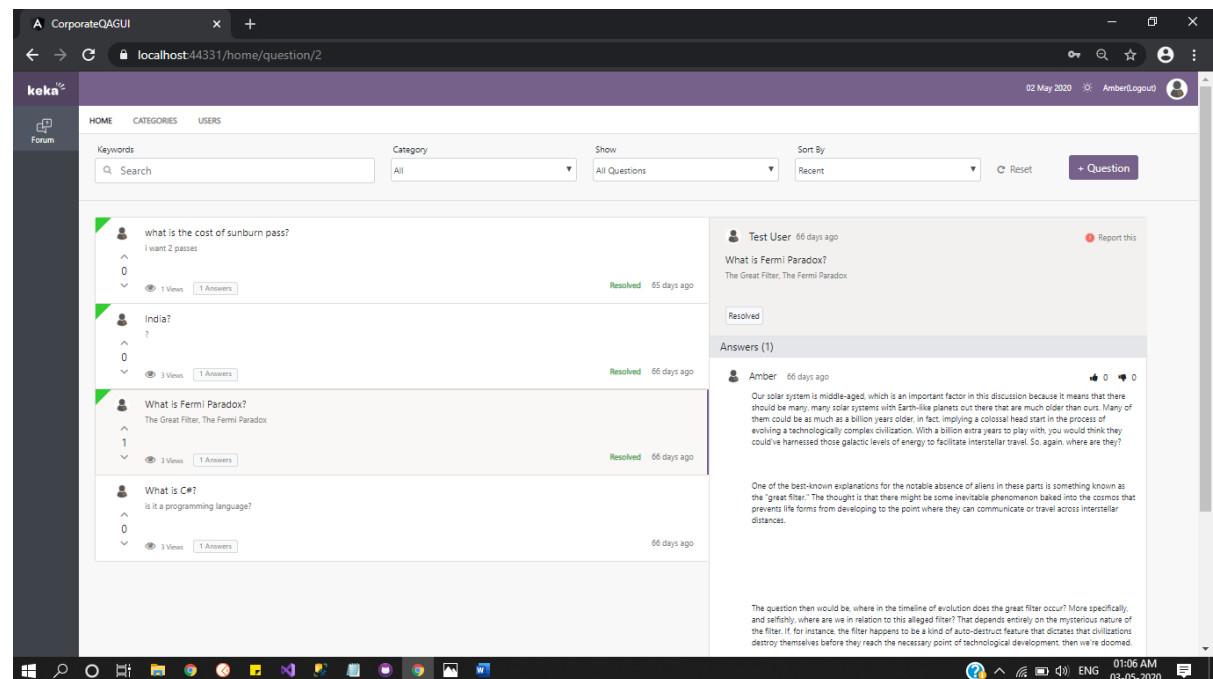


Figure 1: Question & Answer Page

1.5 Organization of Report

Chapter I - Introduction

Here in the introduction, we go through an overview of project and problem statements in which various problems are discussed by covering all the objectives and applications.

Chapter II – Literature Survey

According to research and drawbacks of QnA WEBSITES they are still in their infancy, yet the technology itself is decades old.

Chapter III – Methodology

The methodology is the systematic, theoretical analysis of the methods applied to a field of QnA websites. We go through the background of the project and platform used like Visual Studios MVC and database in the project.

Chapter IV – Implementation

In the implementation, we discussed how our QnA Web application is required in this technical system. We explore the different functions involved in the development of the web application. Step by step instructions is included to help understand the project.

Chapter V – Result

The outcome of our app is discussed along with its features. We have pictorially depicted how our web application functions and how to interact with the queries of users.

Chapter VI – User Manual

In the user manual, we explain how to use our web application discussing the software requirements as well as hardware requirements of the project.

Chapter VII – Conclusion & Future Scope

In QnA application you just not become a good reader but also becomes a good writer. It also aids you to replace the bad qualities like Unusual use of WhatsApp, Fb, Instagram, Skype or many other worst applications from your daily schedule. It helps you to achieve the level of maturity. It always finds very effective answer for your questions which is written on the basis of true experience and event.

Chapter VIII – References

We cite all the resources that helped us create this report for our project. The citations were taken from research papers published by different universities and data researchers who are studying and developing the QNA web application.

CHAPTER II

Literature Survey

2.1 History

Q&A, also Q and A, generally abbreviates the phrase "question and answer", and refers to a period of time or an occasion that follows a type of interview format, often public, when an individual "answers questions that are asked by a reporter, by the people in an audience, etc." It may also refer to:

Q&A (Symantec), database and word processing software product for PCs in the 1980s and 1990s

Q&A website, a site with a searchable database of questions, with answers contributed by users

Q&A software is online software that attempts to answer questions asked by users (Q&A stands for "question and answer"). Q&A software is frequently integrated by large and specialist corporations and tends to be implemented as a community that allows users in similar fields to discuss questions and provide answers to common and specialist questions.

There are numerous examples of Q&A software in both open source and SaaS formats, including Qhub, OSQA, Question2Answer, and Stack Exchange. Standalone Q&A communities such as Quora or Yahoo! Answers aren't traditionally in either category while operating in the Q&A market.[1]

Q&A software is often provided to corporate and specialist sites, so the site and its users can be asked questions as well as provide or receive expert answers to them. This kind of software is particularly useful for responding to questions regarding specific industries. Users may learn by regularly answering questions or exchanging views with other industry specialists using the website.

In the late 1990s, a free online service called Answer Point provided by Ask Jeeves, was launched, allowing users to ask questions and with the help of other people, have them answered. The slogan of the service, "The Ask Jeeves Answer Point is the place where you can ask and answer questions. Have a question? Post it! Know the answer? Post it!",[2] indicated the main function of it, which inspired the creation of later Q&A sites. The last archived version of the Ask Point was from late 2001 when it still allowed registration.

Since then, more and more sites have begun to offer Q&A services. Google launched its Q&A service called Google Questions and Answers in August 2001 which used Google staffers to answer questions by e-mail. A flat fee (US\$3.00) was involved for an answer. In April 2002, Google launched Google Answers, which allowed users to post answers to questions, to replace its predecessor. Google Answers cost askers \$2 to \$200 for an accepted answer. By late December 2006, it was fully closed to new activity.

In early 2000s, Yahoo! launched its online Q&A service called Ask Yahoo!, which was later replaced by the beta version of Yahoo! Answers on December 8, 2005. Ask Yahoo! was discontinued in March 2006. Yahoo! Answers give members the chance to earn points, thus encouraging user participation. To support countries using non-English characters, Yahoo! Answers operate different platforms in some Asian countries, such as Yahoo! Chiebukuro in Japan and as Yahoo! Knowledge in Korea, Taiwan, China, and Hong Kong.

Quora was founded in June 2009, while the website was made available to the public on June 21, 2010. Users can collaborate by editing questions and suggesting edits to other users' answers.

From 2010 with the widespread use of smartphones and tablets, there is an increasing number of Q&A sites that decide to launch mobile applications. Popular Q&A sites like Yahoo! Answers and Quora have launched their own mobile applications. There is also a booming of new Q&A software such as Canvass which purely rely on mobile applications as their service channel.

2.2 Mechanism

Motivations

The social capital theory, social exchange theory, and social cognitive theory explain why users continuously contribute knowledge to online social Q&A communities. On the other hand, many feel hesitant to contribute due to fear of criticism or of misleading the online community members. The contributors can have intrinsic or extrinsic motivation to contribute. Further, the motivations to share knowledge can be categorized into individual-based and organization or website based motivations. Website based motivations (extrinsic) include rewards and incentives to the contributions like upvotes or coupons. Individual-based motivations (intrinsic) would constitute factors like a belief in knowledge ownership, individual characteristics, interpersonal trust, and will for justice. Specifically, two kinds of motivations drive people to participate in online Q&A: Why do people ask questions, and answer questions?

Motivation for asking:

While there are other ways of fulfilling information needs, more people have been asking questions in online Q&A services. What motivates people to take advantage of those systems? And what do users expect from using systems? Choi (2014) surveyed 200 people who actively asked questions on Yahoo! Answers, and asked them the motivations and expectations of asking behaviors in online Q&A system. The five most significant factors of asking behaviors in the survey are:

- Learning; Self-education through acquiring information
- Having fun asking a question
- Seeking advice or opinions for making decisions
- Finding relevant information
- Gaining a sense of security through knowledge

Expectation from askers:

The next step regarding the cycle of question and answering behaviors is for askers to assess answers to see if answers meet their expectations from information needs. In the same survey examined by Choi (2014) the most desired factor was "Additional or alternative information (4.03 / 5)", followed by "Accurate and complete information". It is interesting that factors related to information itself surpassed "social and emotional support (2.47 / 5)" which is related to affective needs.

Motivation for answering: Answering behaviors have raised the curiosity among researchers, because there will be no explicit compensations against the activity. Raban and Harper (2008) asked people who answered questions on social Q&A about the motivation of their answering behavior as well in the same survey. The top ranked motivations are:

- Reputation enhancement
- Enjoyment in helping others
- Reciprocity
- Knowledge self-efficacy
- Satisfaction
- Confirmation
- Continuance intention

Crowdsourcing

Q&A software uses a sourcing model called crowdsourcing to obtain answers to the questions that are posted. Crowdsourcing is the act of outsourcing work to an undefined, networked labor using an open call for participation, and it is used to support the activities on most Q&A software (such as Stack Exchange, Quora, etc.). This technique allows the user to obtain answers from a large community, with lower costs and defect rates to the developers.

The user asks their question which can be answered by any member of that online community. The answers then go through an assessment, wherein the good answers are upvoted and the bad answers are downvoted by the users (or a similar parallel on different platforms). The members of the online community can find a question that needs answering to and answer that question.

This type of crowd participation in a Q&A platform has the potential to increase the knowledge of individual users, thus improving and speeding up their work process. The abundance of information to which users are exposed through this system is effectively changing the way people collaborate, communicate and learn in online communities.

Some other defining features of a Q&A software includes:

- differentiation of questions and answers (features that affect answers should not affect the question)
- differentiation of answers (replies to the question) and comments (replies to an answer)
- user voting of answers
- sorting of answers by votes and questions by answered status
- approval of an answer
- question tagging and tag search
- marking a question as a duplicate of another one (if the event occurs)

CHAPTER III

Methodology

3.1 OVERVIEW OF METHODOLOGY

The establishment and use of sound engineering principles in order to obtain economically developed software that is reliable and works efficiently on real machines is called software engineering.

The methodology will be prototype model of system development life cycle which iterative and will involve the following stages of system analysis, system design, implementation/development. Testing and validation. In the analysis phase, existing system will be evaluated.

Problems with the current system will be identified. This will be done by interviewing user of the system and consulting with human resource. The new system requirements are defined. In particular, the deficiencies in the existing system must be addressed with specific proposals for improvement. The proposed system will be designed with the help of DFDs and ERDs in order to identify the relationship between the different entities in the system. The new system will be implemented using tools such as flow charts, data flow diagrams, and design manipulation languages such as PHP to design the user interface and enable connection to the database and SQL to design the relational tables in the database.

When the new system is up it will be tested and validated in order to find out whether it is doing what it was meant for. Maintenance will be kept up rigorously at all times. Users of the system will be kept up-to-date concerning the latest modifications and procedures.

Software engineering is the discipline whose aim is:

1. Production of quality software
2. Software that is delivered on time
3. Cost within the budget
4. Satisfies all requirements.

Software process is the way in which we produce the software. Apart from hiring smart, knowledgeable engineers and buying the latest development tools, effective software development process is also needed, so that engineers can systematically use the best technical and managerial practices to successfully complete their projects.

Software life cycle is the series of identifiable stages that a software product undergoes during its lifetime. A software lifecycle model is a descriptive and diagrammatic representation of the software life cycle. A life cycle model represents all the activities required to make a software product transit through its lifecycle phases. It also captures the order in which these activities are to be taken.

Life Cycle Models

There are various life cycle models to improve the software processes.

- WATERFALL MODEL
- PROTOTYPE MODEL
- ITERATIVE ENHANCEMENT MODEL
- EVOLUTIONARY MODEL
- SPIRAL MODEL
- AGILE MODEL.
- DevOps

3.2 PROJECT PLATFORM USED IN PROJECT

3.2.1 Visual studio

Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that expand the functionality at almost every level—including adding support for source control systems (like Subversion and Git) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Azure DevOps client: Team Explorer).

3.2.2 Azure DevOps

Azure DevOps is a Software as a service (SaaS) platform from Microsoft that provides an end-to-end DevOps toolchain for developing and deploying software. It also integrates with most leading tools on the market and is a great option for orchestrating a DevOps toolchain. At DevOps Group, we have lots of customers who have found Azure DevOps fits their needs irrespective of their language, platform or cloud.

3.3 PROPOSED METHODOLOGY

The proposed methodology that we use in our project is agile model. Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like –

- Planning
- Requirements Analysis
- Design
- Coding
- Unit Testing and
- Acceptance Testing.

At the end of the iteration, a working product is displayed to the customer and important stakeholders.

Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In Agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

The Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.

The most popular Agile methods include Rational Unified Process (1994), Scrum (1995), Crystal Clear, Extreme Programming (1996), Adaptive Software Development, Feature Driven Development, and Dynamic Systems Development Method (DSDM) (1995). These are now collectively referred to as Agile Methodologies, after the Agile Manifesto was published in 2001.

The graphical illustration of the Agile Model is as follows–

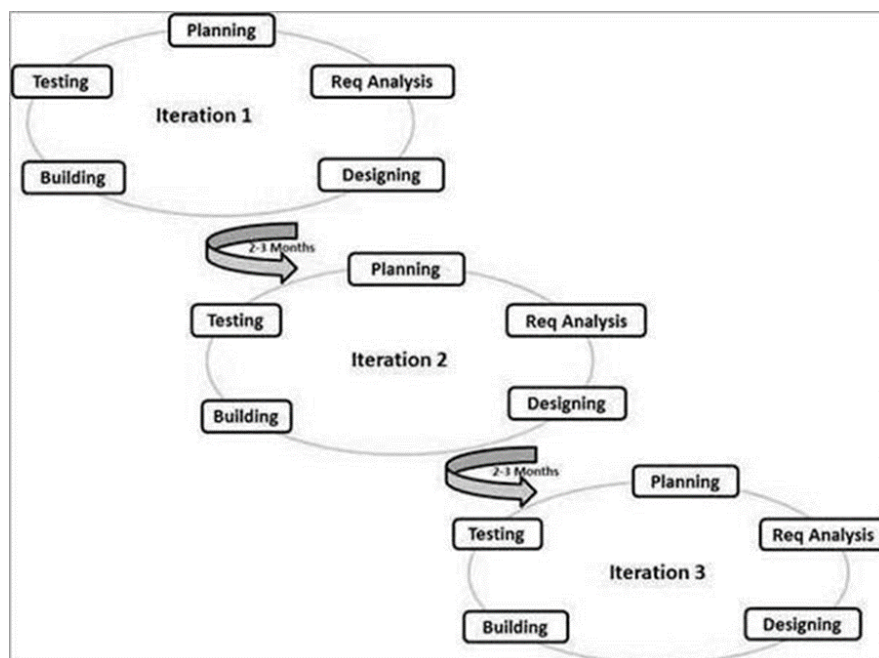


Figure 2: Agile Model

Following are the Agile Manifesto principles –

- **Individuals and interactions**– In Agile development, self-organization and motivation are important, as are interactions like co-location and pair programming.
- **Working software**– Demo working software is considered the best means of communication with the customers to understand their requirements, instead of just depending on documentation.
- **Customer collaboration**– As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous customer interaction is very important to get proper product requirements.
- **Responding to change**– Agile Development is focused on quick responses to change and continuous development.

Agile Vs Traditional SDLC Models

Agile is based on the adaptive software development methods, whereas the traditional SDLC models like the waterfall model is based on a predictive approach. Predictive teams in the traditional SDLC models usually work with detailed planning and have a complete forecast of the exact tasks and features to be delivered in the next few months or during the product life cycle. Predictive methods entirely depend on the requirement analysis and planning done in the beginning of cycle. Any changes to be incorporated go through a strict change control management and prioritization.

Agile uses an adaptive approach where there is no detailed planning and there is clarity on future tasks only in respect of what features need to be developed. There is feature driven development and the team adapts to the changing product requirements dynamically. The product is tested very frequently, through the release iterations, minimizing the risk of any major failures in future.

Customer Interaction is the backbone of this Agile methodology, and open communication with minimum documentation are the typical features of Agile development environment. The agile teams work in close collaboration with each other and are most often located in the same geographical location. Agile methods are being widely accepted in the software world recently. However, this method may not always be suitable for all products.

Agile Model - Pros and Cons

Agile methods are being widely accepted in the software world recently. However, this method may not always be suitable for all products. Here are some pros and cons of the Agile model.

The advantages of the Agile Model are as follows –

- Is a very realistic approach to software development.
- Promotes teamwork and cross training.
- Functionality can be developed rapidly and demonstrated.
- Resource requirements are minimum.
- Suitable for fixed or changing requirements
- Delivers early partial working solutions.
- Good model for environments that change steadily.
- Minimal rules, documentation easily employed.
- Enables concurrent development and delivery within an overall planned context.
- Little or no planning required.
- Easy to manage.
- Gives flexibility to developers.

The disadvantages of the Agile Model are as follows –

- Not suitable for handling complex dependencies.
- More risk of sustainability, maintainability and extensibility.
- An overall plan, an agile leader and agile PM practice is a must without which it will not work.
- Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines.
- Depends heavily on customer interaction, so if customer is not clear, team can be driven in the wrong direction.
- There is a very high individual dependency, since there is minimum documentation generated.

Transfer of technology to new team members may be quite challenging due to lack of documentation.

3.6 DIAGRAMS

3.6.1 ER Diagram

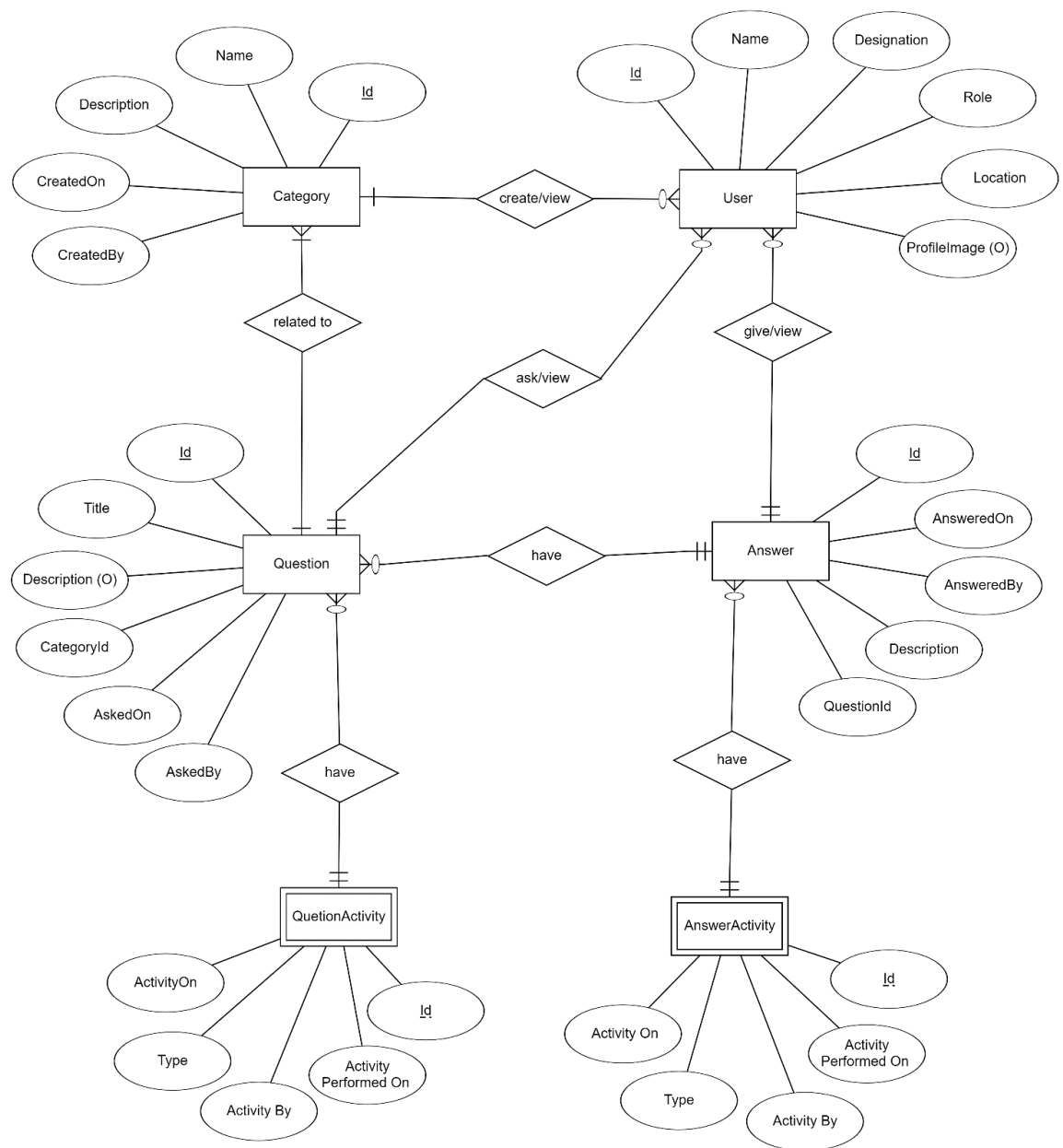


Figure 3: ER Diagram

CHAPTER IV

Implementation

4.1 FUNCTIONS WITH EXPLANATION:

QnA is a site where people post answers to your questions. It also allows you to follow Topics, People, and specific Questions, which is great for keeping up with trends and questions that you never ran into yet. Its advantage lies in its community of reputable experts. I've found the quantity of questions to be quite diverse —QnA covers a wide variety of topics and interests, and is bound to have something for your curiosity. You can also choose to search for answers to specific questions right in the search bar on every page

QnA is the Question and Answer web application and it is the one of the best platform or place where you can share your knowledge and get the best answer for your question.

1. In QnA application you just not become a good reader but also becomes a good writer.
2. It also aids you to replace the bad qualities like Unusual use of WhatsApp, Fb, Instagram, Skype or many other worst applications from your daily schedule.
3. It always finds very effective answer for your questions which is written on the basis of true experience and event.

4.2 CODING WITH EXPLANATION

4.2.1. Models

User Information Model

```
public class UserInfo
{
    public long Id { get; set; }
    public string Username { get; set; }
    public string ProfileImage { get; set; }
    public string Designation { get; set; }
    public string Role { get; set; }
    public string Location { get; set; }
    public long Likes { get; set; }
    public long Dislikes { get; set; }
    public long QuestionsAsked { get; set; }
    public long QuestionsAnswered { get; set; }
    public long QuestionsSolved { get; set; }
}
```

Question Model

```
public class Question
{
    public long Id { get; set; }
    public string Title { get; set; }
    public string Description { get; set; }
    public long CategoryId { get; set; }
    public long AskedBy { get; set; }
    public DateTime AskedOn { get; set; }
}
```

Answer Model

```
public class Answer
{
    public long Id { get; set; }
    public long QuestionId { get; set; }
    public long AnsweredBy { get; set; }
    public string Description { get; set; }
    public DateTime AnsweredOn { get; set; }
}
```

Category Model

```
public class Category
{
    public long Id { get; set; }
    public string Name { get; set; }
    public string Description { get; set; }
    public long CreatedBy { get; set; }
    public DateTime CreatedOn { get; set; }
}
```

4.2.2 Services**Question Services**

```
public class QuestionServices
{
    private IDbConnection _connection;
    public QuestionServices(IDbConnection connection)
    {
        _connection = connection;
    }

    public Question AddQuestion(Question question)
    {
        using (IDbConnection Db = _connection)
        {
```

```

        string sQuery = "INSERT INTO Question (Title,Description,CategoryId,AskedBy,AskedOn)
VALUES(@Title,@Description,@CategoryId,@AskedBy,@AskedOn)";
        Db.Execute(sQuery, question.MapTo<Data.Question>());
        return Db.Query<Data.Question>("Select * from Question where
Description=@Description", new { Description = question.Description
}).FirstOrDefault().MapTo<Question>();
    }
}

public IEnumerable<QuestionInfo> GetAllQuestions(long userId,SearchFilters filters)
{
    using(IDbConnection Db = _connection)
    {
        if(filters.SearchKey == null)
        {
            filters.SearchKey = "";
        }
        string sQuery = $"SELECT * FROM QUESTIONSVIEW WHERE (TITLE LIKE
'{filters.SearchKey}%' OR DESCRIPTION LIKE '{filters.SearchKey}%' )";
        if (filters.CategoryId != 0)
        {
            sQuery = $"{sQuery} AND CATEGORYID = @CATEGORYID";
        }

        if (filters.Show == ShowOption.MyQuestions)
        {
            sQuery = $"{sQuery} AND ASKEDBYID = @USERID";
        }
        else if(filters.Show == ShowOption.MyParticipation)
        {
            sQuery = $"{sQuery} AND (SELECT COUNT(*) FROM ANSWER WHERE QUESTIONID =
QUESTIONSVIEW.ID AND ANSWEREDBY = @USERID)>0 ";
        }

        if (filters.SortBy == SortByOption.Recent)
        {
            sQuery = $"{sQuery} ORDER BY ASKEDON DESC";
        }
        return Db.Query<Data.QuestionInfo>(sQuery, new { USERID = userId, CATEGORYID =
filters.CategoryId }).MapCollectionTo<QuestionInfo>();
    }
}

public IEnumerable<AnsweredQuestionInfo> GetMyAnsweredQuestions(long id)
{
    using(IDbConnection Db = _connection)

```



```

        {
            string sQuery = "SELECT * FROM ParticipationView where AnsweredById = @Id ORDER
BY AnsweredOn DESC";
            return Db.Query<Data.AnsweredQuestionInfo>(sQuery, new { Id = id
}).MapCollectionTo<AnsweredQuestionInfo>();
        }
    }
}

```

Answer Services

public class [AnswerServices](#)

```

{
    private IDbConnection _connection;
    public AnswerServices(IDbConnection connection)
    {
        _connection = connection;
    }
    public Answer AddAnswer(Answer answer)
    {
        using(IDbConnection Db = _connection)
        {
            string sQuery = "INSERT INTO Answer
(QuestionId,AnsweredBy,Description,AnsweredOn)
VALUES(@QuestionId,@AnsweredBy,@Description,@AnsweredOn)";
            Db.Execute(sQuery, answer.MapTo<Data.Answer>());
            return Db.Query<Data.Answer>("Select * from Answer where Description =
@Description", new { Description = answer.Description }).FirstOrDefault().MapTo<Answer>();
        }
    }

    public IEnumerable<AnswerInfo> GetAllAnswers(long id)
    {
        using(IDbConnection Db = _connection)
        {
            string sQuery = "SELECT * FROM AnswersView where AnsweredFor=@Id";
            return Db.Query<Data.AnswerInfo>(sQuery,new { Id = id
}).MapCollectionTo<AnswerInfo>();
        }
    }

    public AnswerInfo GetAnswer(long id)
    {
        using(IDbConnection Db = _connection)
        {
            string sQuery = "SELECT * FROM AnswersView where Id=@Id";
            return Db.Query<Data.AnswerInfo>(sQuery, new { Id = id
}).FirstOrDefault().MapTo<AnswerInfo>();
        }
    }
}

```

Category Services

```

public class CategoryServices: ICategoryServices
{
    private IDbConnection _connection;
    public CategoryServices(IDbConnection connection)
    {
        _connection = connection;
    }

    public IEnumerable<CategoryInfo> GetAllCategories()
    {
        using (IDbConnection Db = _connection)
        {
            string sQuery = "SELECT * FROM CategoriesView";
            return Db.Query<Data.CategoryInfo>(sQuery).MapCollectionTo<CategoryInfo>();
        }
    }

    public Category AddCategory(Category category)
    {
        using (IDbConnection Db = _connection)
        {
            string sQuery = "INSERT INTO Categories (Name , Description , CreatedBy, CreatedOn)
VALUES(@Name , @Description,@CreatedBy, @CreatedOn)";
            var NewCategory = category.MapTo<Data.Category>();
            Db.Execute(sQuery, NewCategory);
            return Db.Query<Data.Category>("SELECT * FROM Categories WHERE Name =@Name",
new { Name = NewCategory.Name }).FirstOrDefault().MapTo<Category>();
        }
    }
}

```

4.3 Google Account Integration**Integrating Google Sign-In into your web app**

Google Sign-In manages the OAuth 2.0 flow and token lifecycle, simplifying your integration with Google APIs. A user always has the option to revoke access to an application at any time.

Create authorization credentials

Any application that uses OAuth 2.0 to access Google APIs must have authorization credentials that identify the application to Google's OAuth 2.0 server. The following steps explain how to create credentials for your project. Your applications can then use the credentials to access APIs that you have enabled for that project.

1. Go to the Credentials page.
2. Click **Create credentials > OAuth client ID**.
3. Select the **Web application** application type.
4. Name your OAuth 2.0 client and click **Create**

After configuration is complete, take note of the client ID that was created. You will need the client ID to complete the next steps. (A client secret is also created, but you need it only for server-side operations.)

Load the Google Platform Library

You must include the Google Platform Library on your web pages that integrate Google Sign-In.

```
<script src="https://apis.google.com/js/platform.js" async
defer></script>
```

Specify your app's client ID

Specify the client ID you created for your app in the Google Developers Console with the `google-signin-client_id` meta element.

```
<meta name="google-signin-client_id"
content="YOUR_CLIENT_ID.apps.googleusercontent.com">
```

Note: You can also specify your app's client ID with the `client_id` parameter of the `gapi.auth2.init()` method.

Add a Google Sign-In button

The easiest way to add a Google Sign-In button to your site is to use an automatically rendered sign-in button. With only a few lines of code, you can add a button that automatically configures itself to have the appropriate text, logo, and colors for the sign-in state of the user and the scopes you request.

To create a Google Sign-In button that uses the default settings, add a `div` element with the class `g-signin2` to your sign-in page:

```
<div class="g-signin2" data-onsuccess="onSignIn"></div>
```

The following is an example of the default Google Sign-In button:

Get profile information

After you have signed in a user with Google using the default scopes, you can access the user's Google ID, name, profile URL, and email address.

To retrieve profile information for a user, use the `getBasicProfile()` method.

```
function onSignIn(googleUser) {
  var profile = googleUser.getBasicProfile();
  console.log('ID: ' + profile.getId()); // Do not send to your backend! Use an ID token instead.
  console.log('Name: ' + profile.getName());
  console.log('Image URL: ' + profile.getImageUrl());
  console.log('Email: ' + profile.getEmail()); // This is null if the 'email' scope is not present.
}
```

Note: By default, the `fetch_basic_profile` parameter of `gapi.auth2.init()` is set to `true`, which will automatically add `'email profile openid'` as scope. **Important:** Do not use the Google IDs returned by `getId()` or the user's profile information to communicate the currently signed in user to your backend server. Instead, [send ID tokens](#), which can be securely validated on the server.

Sign out a user

You can enable users to sign out of your app without signing out of Google by adding a sign-out button or link to your site. To create a sign-out link, attach a function that calls the `GoogleAuth.signOut()` method to the link's onclick event.

```
<a href="#" onclick="signOut();">Sign out</a>
<script>
function signOut() {
  var auth2 = gapi.auth2.getAuthInstance();
  auth2.signOut().then(function () {
    console.log('User signed out.');
```

CHAPTER V

Result

5.1 Test Cases and Result

To conclude we summarize the various results that have been obtained. As a first step, we researched the technologies used in web applications and we selected angular and .net core to develop our web app Truba QnA. With the help of these two frameworks, we were able to successfully create our web app with maximum efficiency.

After lot of research we found out that angular and dot Net core has everything which was needed in our projects. They both are open source and widely used in market and are known for their efficiency. Angular is used to create SPA (single page application). It uses TypeScript language which is basically a super set of vanilla javascript. It provides extra functionality than javascript.

Both Angular and .Net core have extensive community support which helped us in creating an efficient web application.

5.2 Analytic Discussion

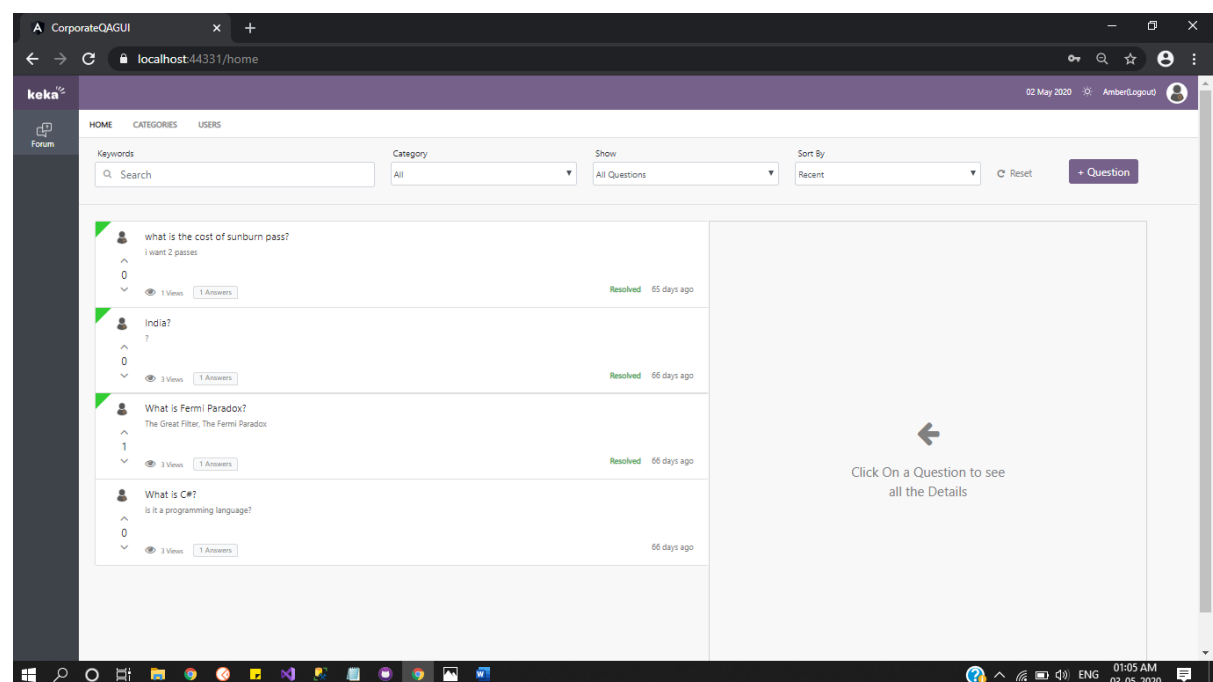


Figure 4: Home Screen

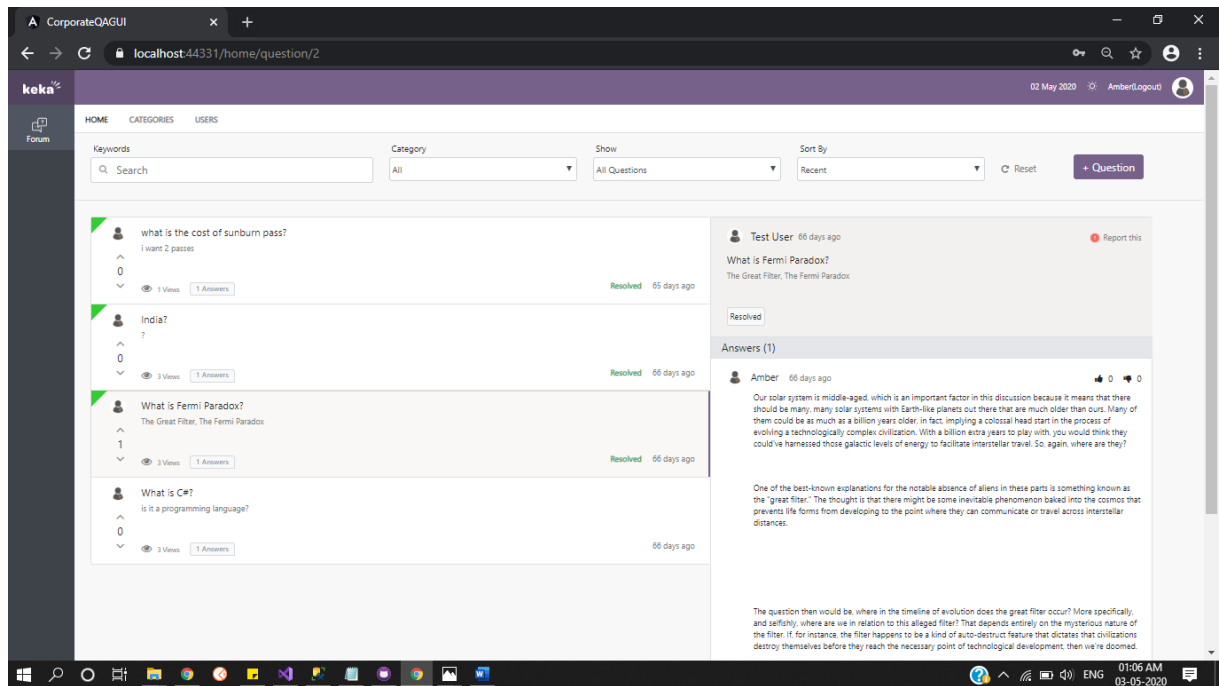


Figure 5: Question's Answers Screen

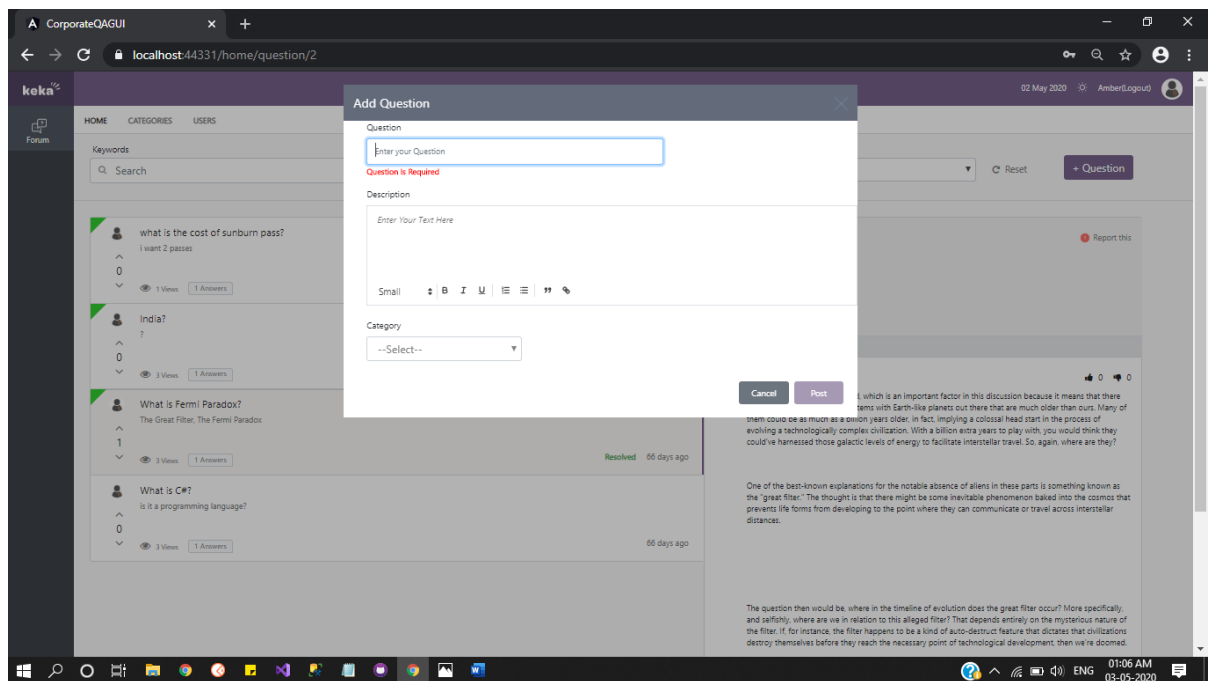


Figure 6: Add Question

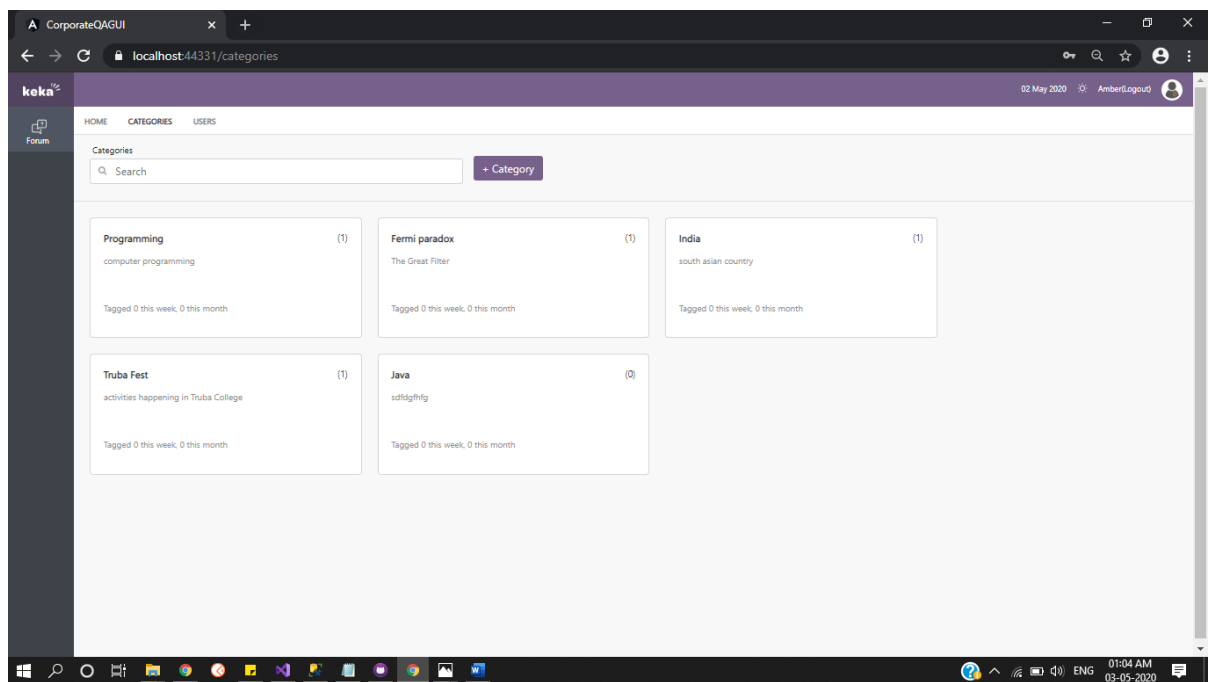


Figure 7: Add Category

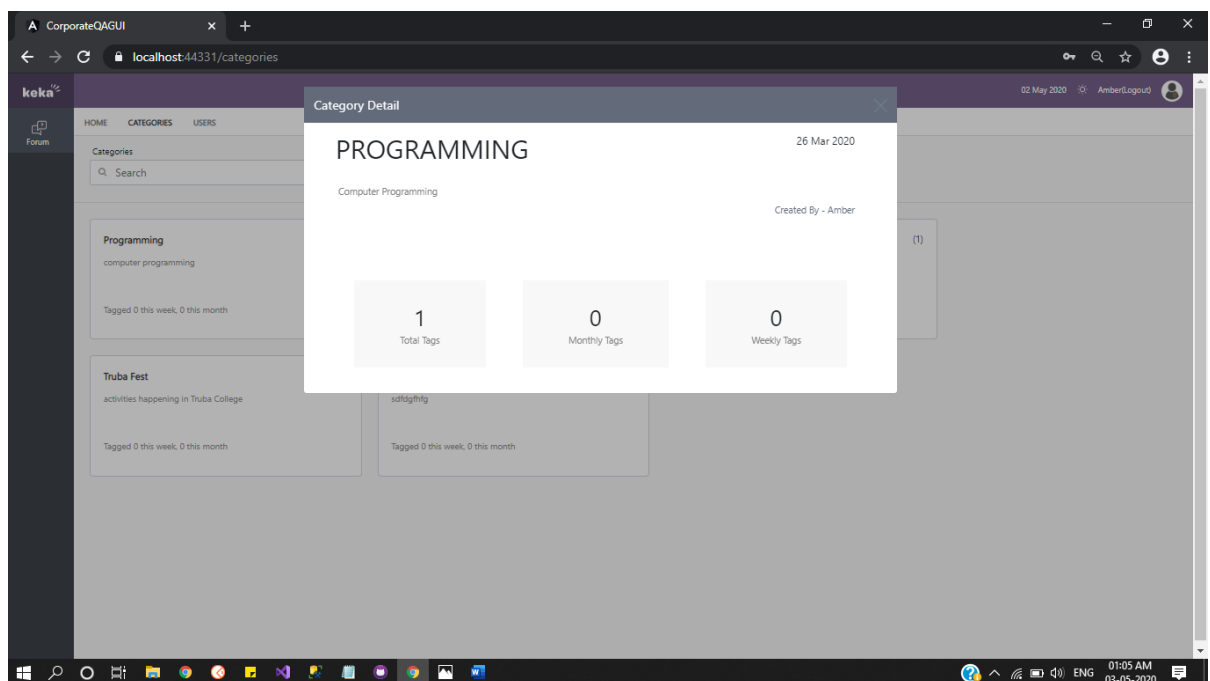


Figure 8: Category Information

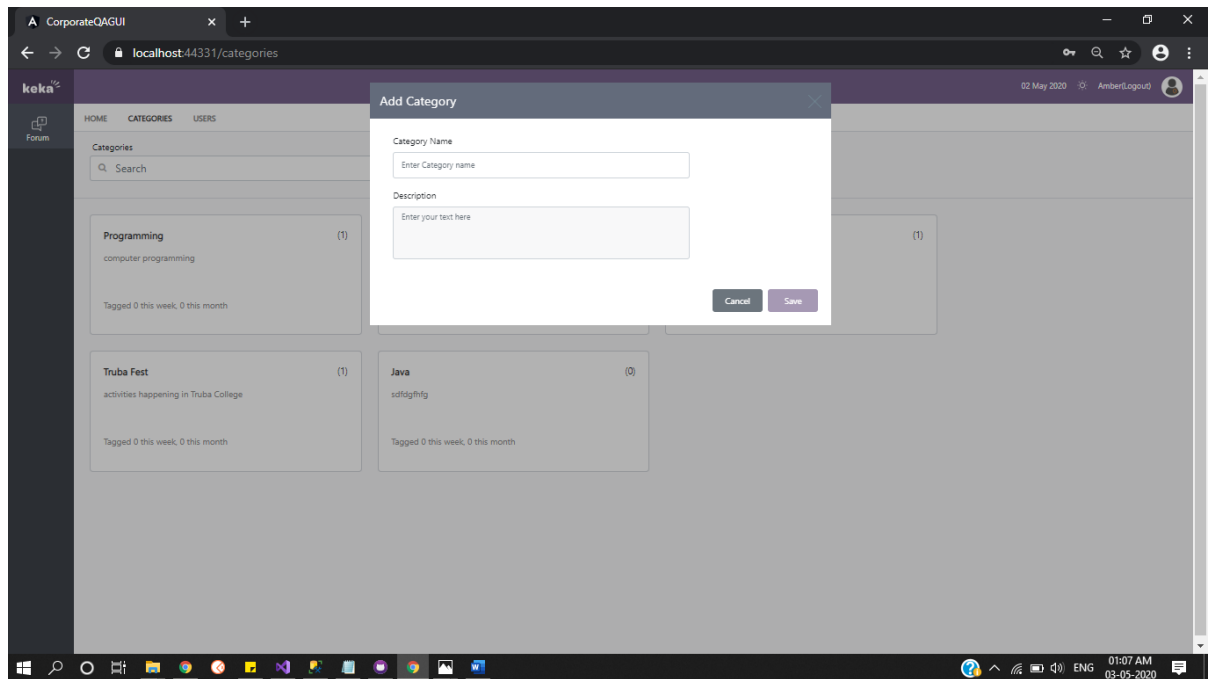


Figure 9: Add Category

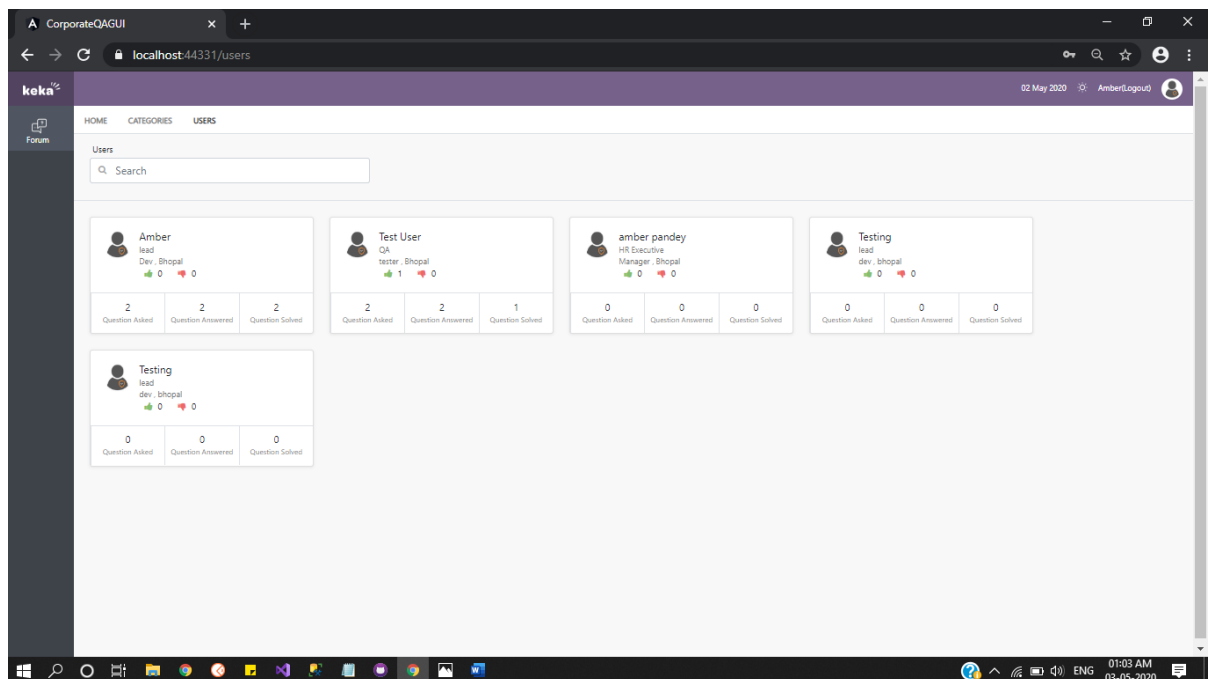


Figure 10: Users

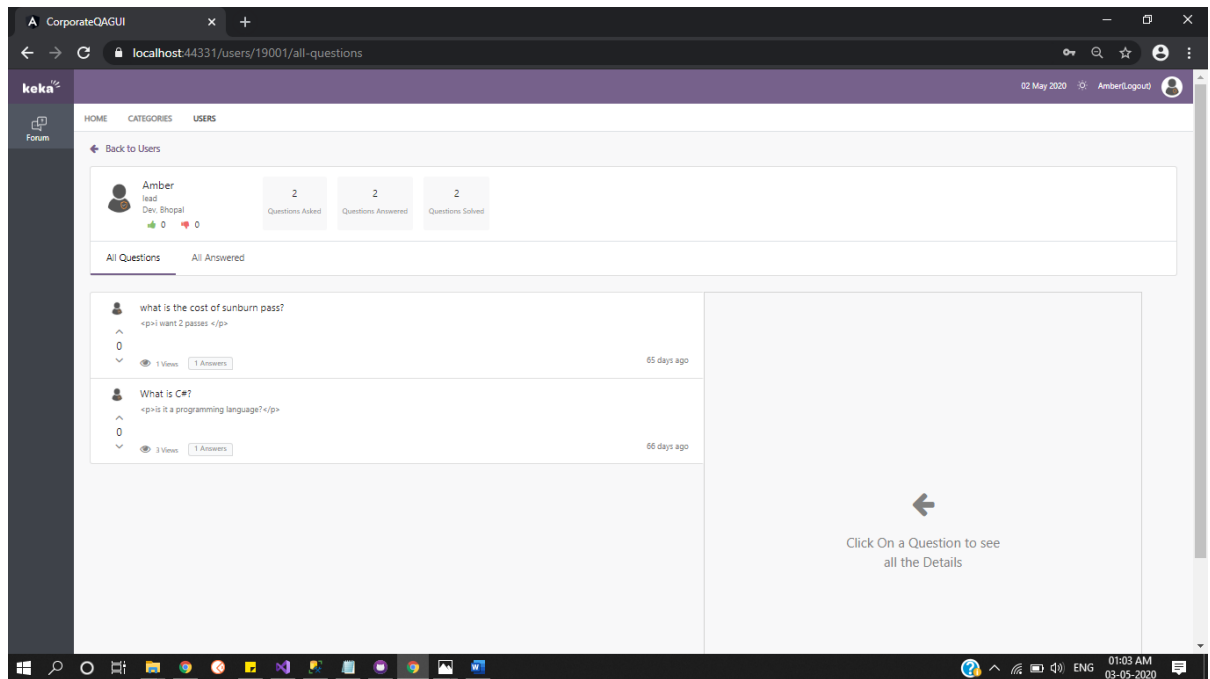


Figure 10: User's Information

5.3 Database Design

5.3.1 Tables

LAPPY.CorporateQnA - dbo.Question			
	Column Name	Data Type	Allow Nulls
🔑	Id	int	<input type="checkbox"/>
	Title	varchar(100)	<input type="checkbox"/>
	Description	varchar(MAX)	<input type="checkbox"/>
	CategoryId	int	<input type="checkbox"/>
	AskedBy	int	<input type="checkbox"/>
	AskedOn	datetime	<input type="checkbox"/>
			<input type="checkbox"/>

Figure 11: Question Table

LAPPY.CorporateQnA - dbo.Answer			
	Column Name	Data Type	Allow Nulls
PK	Id	bigint	<input type="checkbox"/>
	QuestionId	bigint	<input type="checkbox"/>
	AnsweredBy	bigint	<input type="checkbox"/>
	Description	varchar(MAX)	<input type="checkbox"/>
	AnsweredOn	datetime	<input type="checkbox"/>
			<input type="checkbox"/>

Figure 12: Answer

LAPPY.CorporateQn...bo.AnswerActivity			
	Column Name	Data Type	Allow Nulls
PK	Id	bigint	<input type="checkbox"/>
	ActivityPerformedOn	bigint	<input type="checkbox"/>
	ActivityBy	bigint	<input type="checkbox"/>
	Type	smallint	<input type="checkbox"/>
	ActivityOn	datetime	<input type="checkbox"/>
			<input type="checkbox"/>

Figure 13: AnswerActivity

LAPPY.CorporateQn...o.QuestionActivity			
	Column Name	Data Type	Allow Nulls
PK	Id	bigint	<input type="checkbox"/>
	ActivityPerformedOn	bigint	<input type="checkbox"/>
	ActivityBy	bigint	<input type="checkbox"/>
	Type	smallint	<input type="checkbox"/>
	ActivityOn	datetime	<input type="checkbox"/>
			<input type="checkbox"/>

Figure 14: Question Activity

LAPPY.CorporateQnA - dbo.Users* ↗ ✕			
	Column Name	Data Type	Allow Nulls
🔑	Id	bigint	<input type="checkbox"/>
	Name	varchar(50)	<input type="checkbox"/>
▶	Designation	varchar(50)	<input type="checkbox"/>
	Role	varchar(20)	<input type="checkbox"/>
	Location	varchar(20)	<input type="checkbox"/>
	ProfileImage	varchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Figure 15: Users

LAPPY.CorporateQnA - dbo.Categories ↗ ✕			
	Column Name	Data Type	Allow Nulls
🔑	Id	bigint	<input type="checkbox"/>
	Name	nchar(100)	<input type="checkbox"/>
	Description	nvarchar(MAX)	<input type="checkbox"/>
	CreatedBy	int	<input checked="" type="checkbox"/>
	CreatedOn	datetime	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Figure 16:Categories

5.3.2 Views

Answers View

```

CREATE VIEW [dbo].[AnswersView]
AS
SELECT
A.Id,
A.QuestionId
AS AnsweredFor,
A.AnsweredBy AS AnsweredById,
(SELECT [Name] FROM dbo.Users AS U WHERE (Id = A.AnsweredBy)) AS AnsweredBy,
(SELECT ProfileImage FROM dbo.Users WHERE (Id = A.AnsweredBy)) AS ProfileImage,
A.Description,
(SELECT COUNT(Id) AS Expr1 FROM dbo.AnswerActivity AS AA WHERE
(ActivityPerformedOn = A.Id) AND (Type = 1)) AS Likes,
(SELECT COUNT(Id) AS Expr1 FROM dbo.AnswerActivity AS AA WHERE
(ActivityPerformedOn = A.Id) AND (Type = 2)) AS Dislikes,
DATEADD(HOUR, 5, DATEADD(MINUTE, 30, A.AnsweredOn)) AS AnsweredOn
FROM
dbo.Question AS Q
INNER JOIN
dbo.Answer AS A ON Q.Id = A.QuestionId
GO

```

Categories View

```

CREATE VIEW [dbo].[CategoriesView]
AS SELECT
    C.Id as Id,
    C.Name as Name,
    C.Description as Description,
    C.CreatedBy as CreatedBy,
    C.CreatedOn as CreatedOn,
    (Select count(Q.Id) from Question Q where Q.CategoryId = C.Id ) as
TotalTagged,
    (Select count(Q.Id) from Question Q where Q.CategoryId=C.Id and Q.AskedOn>=
DATEADD(DAY,-7,GETDATE())) as WeeklyTagged,
    (Select count(Q.Id) from Question Q where Q.CategoryId = C.Id and
Q.AskedOn>= DATEADD(DAY,-30,GETDATE())) as MonthlyTagged,
    (Select U.Name from Users U where U.Id = C.CreatedBy) as CreatedByPersonName
from Categories as C;
GO

```

Participation View

```

CREATE view [dbo].[ParticipationView]
as select
    QV.Id as Id,
    QV.ProfileImage as ProfileImage,
    QV.AskedBy as AskedBy,
    QV.AskedBy as AskedBy,
    QV.Title as Title,
    QV.Description as QuestionDescription,
    QV.Upvotes as Upvotes,
    QV.TotalAnswers as TotalAnswers,
    QV.TotalViews as TotalViews,
    AV.Id as AnswerId,
    AV.AnsweredBy as AnsweredBy,
    AV.Description as Description,
    AV.AnsweredOn as AnsweredOn
from QuestionsView QV inner join AnswersView AV on QV.Id = AV.AnsweredFor;
GO

```

Questions View

```

CREATE VIEW [dbo].[QuestionsView]
AS
SELECT
    Q.Id,
    U.Id AS AskedById,
    U.[Name] AS AskedBy,
    U.ProfileImage,
    Q.Title,
    Q.[Description],
    Q.CategoryId,
    (SELECT COUNT(Id) AS Expr1 FROM QuestionActivity AS QA WHERE (ActivityPerformedOn
= Q.Id) AND (Type = 1)) -
    (SELECT COUNT(Id) AS Expr1 FROM QuestionActivity AS QA WHERE (ActivityPerformedOn
= Q.Id) AND (Type = 2)) AS Upvotes,
    (SELECT COUNT(DISTINCT ActivityBy) AS Expr1 FROM QuestionActivity AS QA WHERE
(ActivityPerformedOn = Q.Id) AND (Type = 0)) AS TotalViews,
    (SELECT COUNT(Id) AS Expr1 FROM Answer AS A WHERE (QuestionId = Q.Id)) AS
TotalAnswers,
    DATEADD(HOUR, 5, DATEADD(MINUTE, 30, Q.AskedOn)) AS AskedOn,
    (SELECT COUNT(dbo.AnswerActivity.Id) AS Expr1

```

```

FROM AnswerActivity
INNER JOIN
(SELECT Id AS AnswerId FROM Answer WHERE (QuestionId = Q.id)) AS SelectedAnswers
ON dbo.AnswerActivity.ActivityPerformedOn = SelectedAnswers.AnswerId
WHERE (dbo.AnswerActivity.Type = 4)) AS Resolved
FROM dbo.Question AS Q
INNER JOIN Users AS U ON Q.AskedBy = U.Id
GO

```

Users View

```

CREATE VIEW [dbo].[UsersView]
AS SELECT
    U.Id as Id,
    U.Name as Username,
    U.ProfileImage as ProfileImage,
    U.Designation as Designation,
    U.Role as Role,
    U.Location as Location,
    (Select count(QA.Id) from QuestionActivity QA where QA.ActivityPerformedOn
in (Select Q.Id from Question Q where Q.AskedBy=U.Id) and QA.Type=1) as Likes,
    (Select count(QA.Id) from QuestionActivity QA where QA.ActivityPerformedOn
in (Select Q.Id from Question Q where Q.AskedBy=U.Id) and QA.Type=2) as Dislikes,
    (Select count(Q.Id) from Question Q where Q.AskedBy = U.Id) as
QuestionsAsked,
    (Select count(A.Id) from Answer A where A.AnsweredBy = U.Id) as
QuestionsAnswered,
    (Select count (AA.Id) from AnswerActivity AA left join Answer A on A.Id =
AA.ActivityPerformedOn where AA.Type = 4 AND U.Id = A.AnsweredBy) as
QuestionsSolved
FROM Users as U;
GO

```

CHAPTER VI

User Manual

6.1 Software Requirements

The software requirement in this project includes:

For Development

- Angular 8: Open-source framework with highly advanced JavaScript support
- Node.js
- MS-SQL server
- Visual Studio 2019 community: It is a free community version available for all.
- Web Browser: Internet Explorer, Mozilla Firefox, Google Chrome, etc.
- Window 7 and above

For App User

- Web Browser

6.2 Hardware Requirements

Sever Side:

- **Operating System:** windows 10
- **Processor:** 3.0 Ghz dual core or Higher
- **Ram:** 12Gb+
- **Hard Drive:** 90Gb

Client Side:

- **Operating System:** Windows XP or above, MAC or UNIX.
- **Processor:** Pentium III or 2.0 GHz or higher.
- **RAM:** 256 Mb or more

6.3 Steps to Run Project:

- Open Visual studio
- Open QnA.sln
- Configure ms-sql server and QnA.sln to connect web app with database.

- Install npm packages: use command “npm install”
- Build solution: press “Ctrl + B”,
- Go to “QnA App” folder in QnA project and build angular app: use command “npm build --watch”
- After build press “ctrl + F5” in visual studio to run IIS server
- By default, it will start running in local environment in default browser.

CHAPTER VII

Conclusion & Future Scope

7.1 Conclusion

- Truba QnA is very effective for solving queries and finding a suitable solution for the related query according to end user.
- Truba QnA enables to segregate the queries based on their search keyword.
- Truba QnA makes searching of answers based on their categories.
- Truba QnA enables users to add their own answer to the queries.
- User can upvote the answers which are appropriate according to them.
- User who has upload the query can select the best answer from the list of answers given by other users.
- A User can see other users details.
- Users can add their own category related to their query.
- Integrated google authentication ; users can login using google account.

7.2 Future work

- We are looking forward to add a new feature of sub categories to the category section.
- In future a user will be able to add media (images, gifs, videos) in an answer.
- In future a user will be able to add comments on the solution of the respected queries.
- We are looking forward to add a discussion forum where users can interact with each other.
- Integration of Outlook and Facebook authentication features are in pipeline.

CHAPTER VIII

References

1. Microsoft <https://docs.microsoft.com/en-us/dotnet/csharp/>
2. Angular <https://angular.io/guide/architecture>
3. C# <https://www.c-sharpcorner.com/>
4. W3schools <https://www.w3schools.com>
5. Wikipedia www.wikipedia.com
6. www.mdbootstrap.com
7. www.tutorialspoint.com
8. www.stackoverflow.com
9. www.github.com
10. <https://developers.google.com/>
11. www.w3docs.com