Crowd Sourced Utility Application

K.MarySudha Rani[1], SaiTej Dandge[2], Pallavi Yenigalla[3]

[1]Asst.Prof, CSE Dept, CBIT
[2]BE 4/4, CSE Dept,CBIT
[3]BE 4/4, CSE Dept, CBIT

Abstract: Crowdsourcing has emerged as new learning and online collaborative paradigm, in which crowds of people can collaborate and complete a specific task. This project enables users to connect and share Mcq's, Blogs with tags included. It provides a graphical analysis of the performance of each individual based on their attempts and also a comparative performance with others. It provides a social network among the people to chat and share ideas among them so that they can share their knowledge and ideas over the application. This is scalable as it is deployed on the cloud and the resources allocated can be increased as the number of users of the application increase. It also suggests questions based on the pattern of search or type of questions which the user answers. Overall this project on the cloud helps people to interact and gain knowledge through questionnaires and answering questions, which leads to the sharing of one's information with everyone.

Keywords: crowdsourcing, cloud, social network.

1. Introduction

The term "crowdsourcing" has been around for nearly a decade, originally coined in 2006 by wired writer Jeff Howe. He defined it as the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people. In other words, businesses tap into the "power of the crowd" rather than carrying out tasks themselves. Crowdsourcing is a specific sourcing model in which individuals or organizations use contributions from internet users to obtain needed services or ideas. This mode of sourcing divides work between participants to achieve a cumulative result. Advantages of using crowdsourcing may include improved costs, speed, quality, flexibility, scalability, or diversity. Companies have caught on to the attention surrounding crowdsourcing, and have sought out innovative ways to get their valued customers involved in production and promotion processes. Crowdsourcing brings people together online, so there aren't any geographical constraints to get in the way. Crowdsourcing generally occurs through social media, so businesses can gain insightinto what consumers are talking about, interested in, and where their talents lie. In this paper, the crowdsourcing model and its suitability to develop a utility application is explored, and a novel framework is presented.

2. Crowdsourced utility application

This application acts as a platform to share all knowledge in terms of multiple choice questions and articles to the world. Users are notified with a push notification whenever new question or article added to their field of interest.

Users can upload multiple choice questions including tags with answer they know. Questions can be classified into multiple categories. Users can also add subject questions to find out answer from other users. Graphical analysis of each user can be seen, to know user's field of interest and his active log. While uploading question/article, user can include images related to respective content in the application and this would help.


https://play.google.com/store/apps/details?id=com.tdevelopers.questo

3.1 Cloud Service

Cloud computing is a type of Internet-based computing that provides shared computer processing resources and data to computers and other devices on demand. It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable computing resources (e.g., computer networks, servers, storage, applications and services), which can be rapidly provisioned and released with minimal management effort. Cloud computing and storage solutions provide users and enterprises with various capabilities to store and process their data in either privately owned, or third-party data centers that may be located far from the user– ranging in distance.

Cloud computing relies on sharing of resources to achieve coherence and economy of scale. This application uses Platform as a service (PaaS) cloud model. In a PaaS model, a cloud provider delivers hardware and software tools -- usually those needed for application development -- to its users as a service. PaaS does not typically replace a business entire infrastructure. Instead, a business relies on PaaS providers for key services, such as Java development or application hosting. A PaaS provider, ports all the underlying computing and software; users only need to log in and start using the platform – usually through a Web browser interface. A PaaS provider hosts the hardware and software on its own infrastructure. As a result, PaaS frees users from having to install in-house hardware and software to develop or run a new application.

Figure 1 : Cloud Architecture
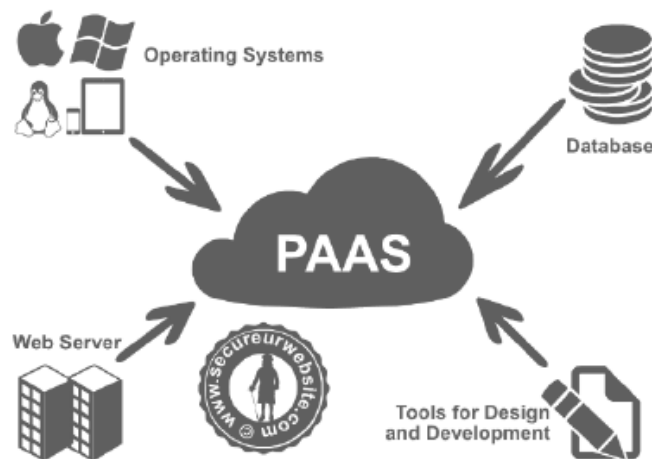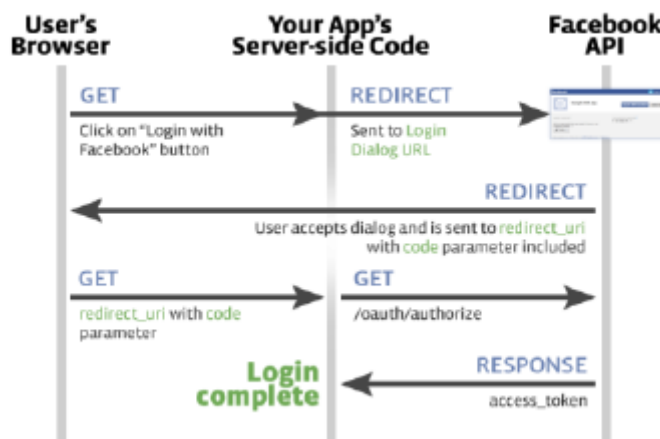

3.2 Facebook Authentication and OAuth 2.0

Figure 2: Facebook Login API

Facebook Login for Apps is a fast and convenient way for people to create accounts and log into the app across multiple platforms.

The Facebook SDK for Android enables people to sign into the developer's app with Facebook Login. When people log into the app with Facebook they can grant permissions to the app so that the developer can retrieve information or perform actions on Facebook on their behalf.

This application uses facebook login to authenticate users. It uses only public profile of user which is as given default permission by facebooksdk.

3.3 Database Interaction using Firebase Real Time Database
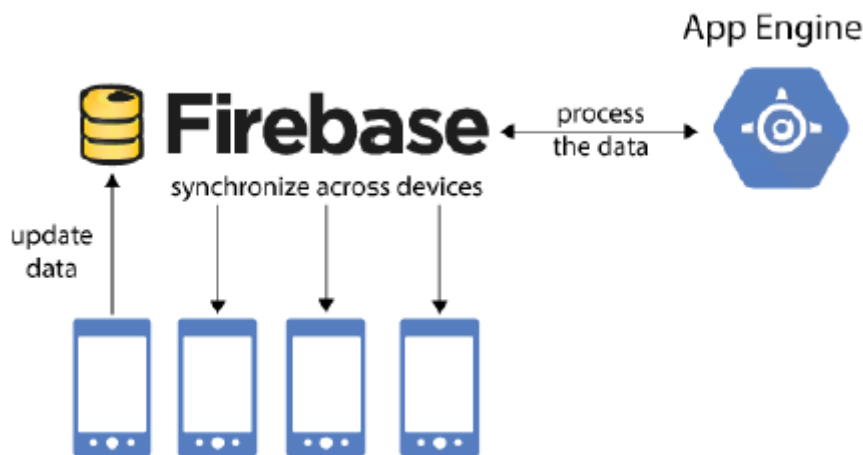

Figure 3: Firebase RealTime Database

The Firebase Real-time Database is a cloud-hosted database. Data is stored as JSON and synchronized in real-time to every connected client. Instead of typical HTTP requests, the Firebase Real-time Database uses data synchronization—every time data changes, any connected device receives that update within milliseconds. It provides collaborative and immersive experiences without thinking about networking code. When the device regains connection, the Real-time Database synchronizes the local data changes with the remote updates that occurred while the client was offline, merging any conflicts automatically.

Instead of REST api calls, this application use Firebase Real Time database for reading and writing operations. This NoSQL database is horizontally scalable with schema less structure, Uploading part involves adding new question (multiple choice question), article, tag. This uploading part is only accessible after being authenticated in the application using OAuth 2.0 provided.

Some parts of this application like Comments of any digital artifact is needed to be synchronized in real time.

Transactions are used to update like count and view count. Without transactions, likes and views of particular question or article may not be consistent and this may lead to misunderstanding.

3.4 Firebase Cloud Storage

Figure 4: Storing Images in Firebase Cloud Storage
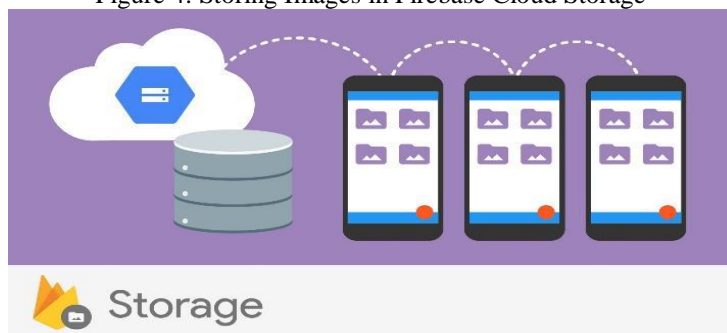
Cloud Storage for Firebase is a powerful, simple, and cost-effective object storage service built for Google scale. The Firebase SDKs for Cloud Storage add Google security to file uploads and downloads for Firebase apps, regardless of network quality. Developers can use Firebase SDKs to store images, audio, video, or other user-generated content. Google Cloud Storage is used to access the same files on the server. Some questions, articles may have images embedded. In this application, cloud storage is used to store images. All the images of different objects are uploaded into storage bucket. After successful url of uploaded image is obtained which is stored as attribute in table.
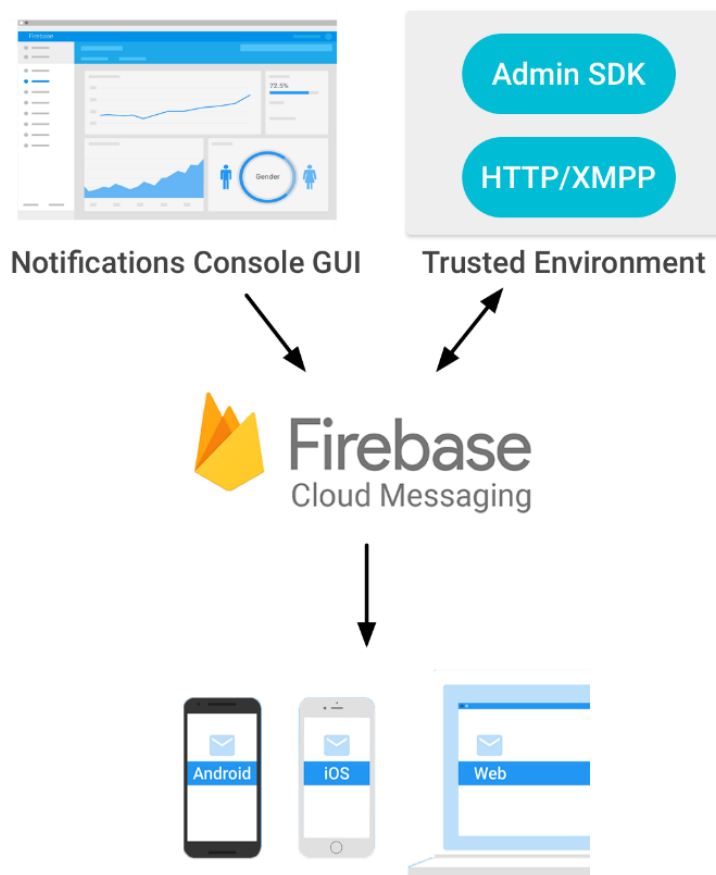
3.5 Notifications

Firebase Cloud Messaging (FCM) offers a broad range of messaging options and capabilities. The information in this page is intended to help understand notification messages and data messages and what you can do with them using FCM's options.

Some of these options are available only through the Firebase Cloud Messaging server implementation. Each device will have its own unique token id.This application runs service on client side and checks for token id continuously

Whenever other user likes or comments on particular post, the owner of post gets notified by a push notification. First the owner's tokenid is fetched from database and HTTP call is made to GCM server by passing token id as argument to that call.

Figure 5: Firebase Cloud Messaging

When the owner reconnects to internet, he will get data from GCM server which is shown as push notification.
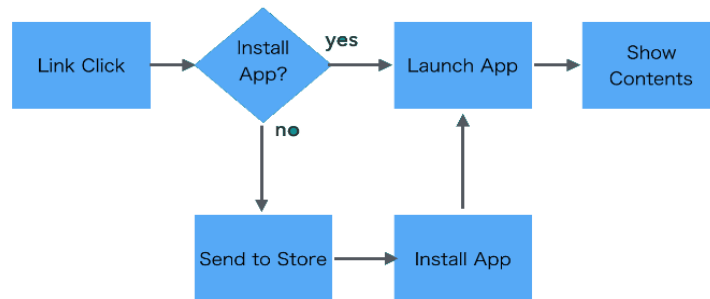
3.6 Firebase Dynamic Links

Figure 6: Firebase Dynamic Links working

Firebase Dynamic Links are links that work on multiple platforms, and whether or not the app is already installed. If a user opens a Dynamic Link on iOS or Android, they can be taken directly to the linked content in your native app.

If a user opens a Dynamic Link on Android and doesn't have your app installed, the user can be prompted to install it; then, after installation, the app starts and can access the link.

Dynamic Link is created by forming a URL by adding Dynamic Link parameters to a domain specific to your app. These parameters specify the links to open, depending on the user's platform and whether the app is installed. Shareable content in this application are shared using these reliable dynamic links, when the user clicks these dynamic link, they are redirected to shared content, instead of showing them main screen

3.7 Picasso Image Loader

Picasso is an image library for Android. It's created and maintained by Square, and caters to image loading and processing. It simplifies the process of displaying images from external locations. Many common pitfalls of image loading on Android are handled automatically by Picasso such as Handling ImageViewrecycling and download cancelation in an adapter, complex image transformations with minimal memory use, automatic memory and disk caching. This application uses this loader to fetch image as bitmap from storage bucket.Url of image which is present as attribute in object is sent as an argument to image loader. This loader loads picture from url asynchronously and returns bitmap which is used to display image.

3.8 Palette Vibrant Color Detection

Figure 7: Image Swatches fetched using palette extractor

A helper class to extract prominent colors from an image. A number of colors with different profiles are extracted from the image such as Vibrant, Vibrant Dark, Vibrant Light ,Muted,

Muted Dark, Muted Light. Palette. Builder supports both synchronous and asynchronous generation which doesn't affect UI. This application uses palette to get vibrant color from image. Bitmap of image is passed as an argument to this, palette extracts the vibrant color from bitmap asynchronously and returns back the color which is set as background of that imageview

4. Comparison with Existing Systems

4.1 Quora

Quora is an existing system that connects to everything the user wants to know about. Quora aims to be a place to write new content and share content from the web. People and their interests are organized so that valid and valuable information is found to the users. This system implements only subjective data and does not include the multiple-choice questions which can be very fruitful for educational purpose such as preparing for the GRE.

The crowd sourced utility application also provides a comparative graphical analysis which helps the users to monitor their performances by answering questions, this feature not included in quora.

4.2 Stack Exchange

Stack Exchange is a network of question-and-answer websites on topics in varied fields, each site covering a specific topic, where questions, answers, and users are subject to a reputation award process. The sites are modelled after Stack Overflow, a Q&A site for computer programming questions that was the original site in this network. The reputation system allows the sites to be self-moderating. This application does not include the questionnaire in the multiple choice format.

5. Results

The results and discussion explains the outcomes obtained by completion of project.

Fig 8 illustrates the analytics of the user's involvement in the application. These analytics is displayed on the server side to the developer so as to track the progress of the application.

Fig 9 illustrates the analytics of user's app usage, the analytics displayed are the division of users based on country, users who installed who have different OS versions and based on the users handset model.

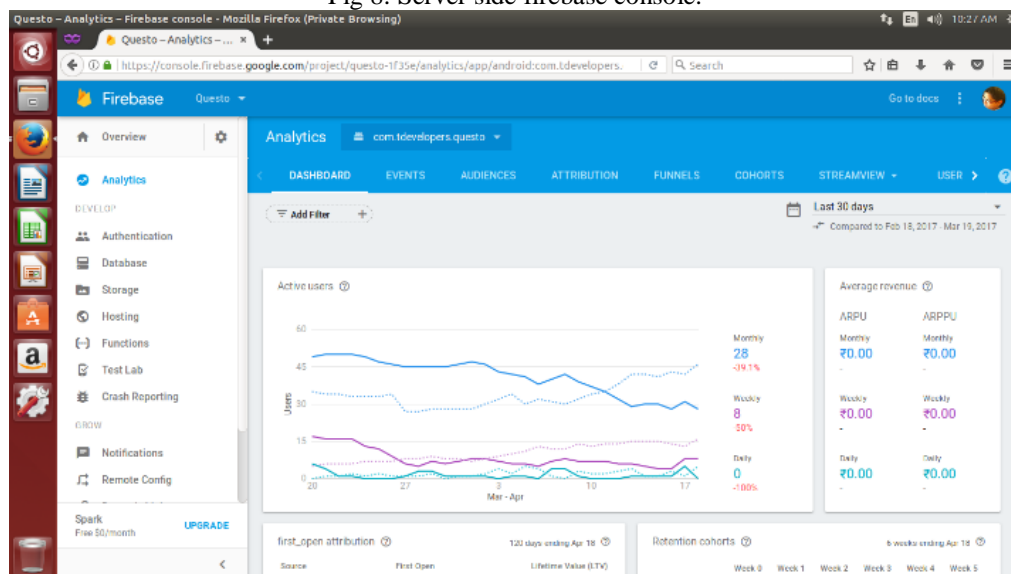Fig 8: Server side firebase console:

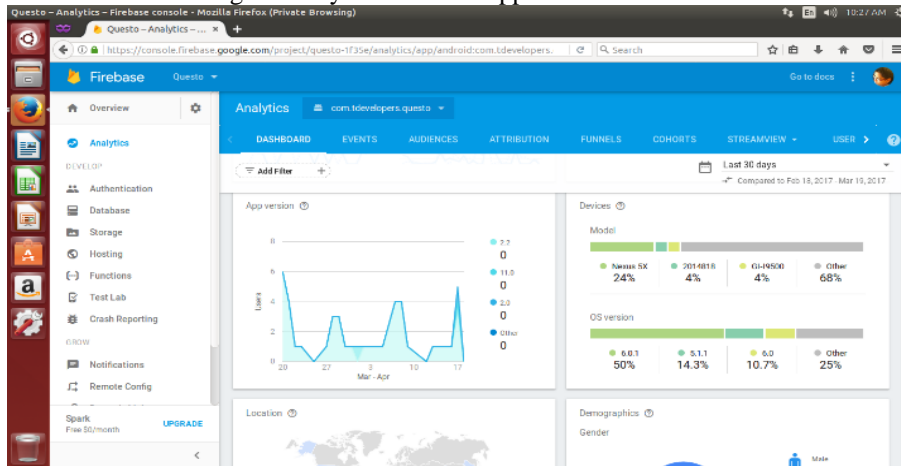Fig 9: Analytics based on app and OS versions

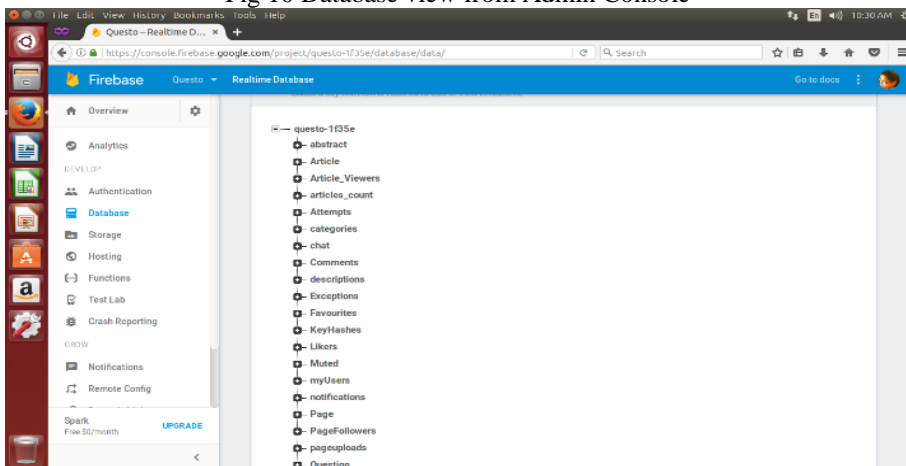Fig 10 Database view from Admin Console

Fig 10 illustrates the database as modules that is stored on the server side. The data is stored under several categories in the database such as likers, users information, question uploads, chats, favourites, notifications, comments, number of viewers.
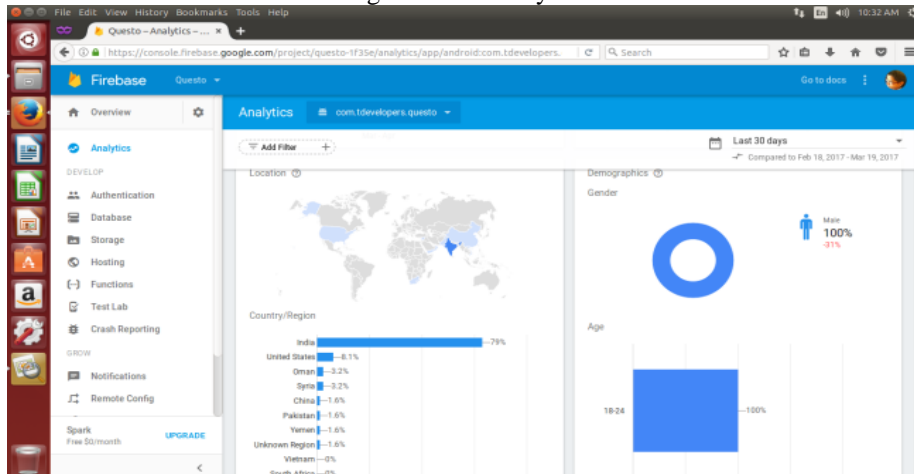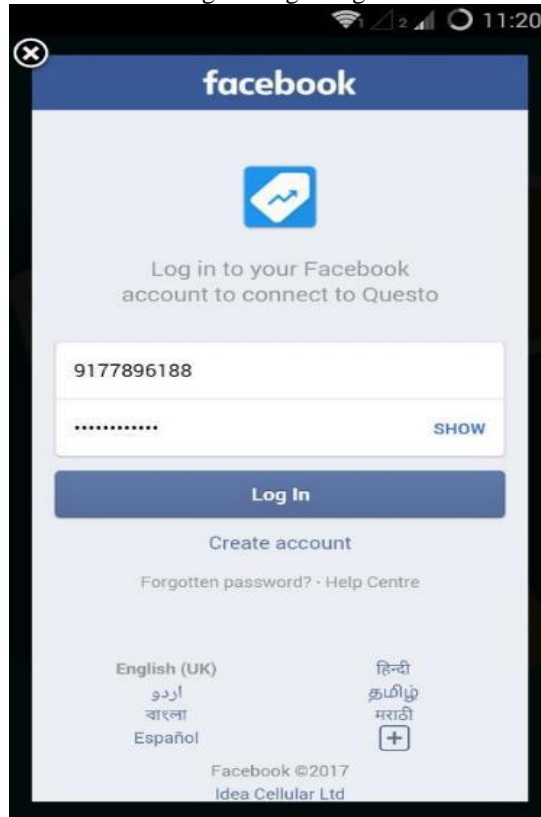
Fig 11. User Analytics

Fig 11 illustrates the graphical analysis of the users who use the app divided under different countries. It also illustrates the number of male users and female users using the app. Analytics of the age group of users using the application is also displayed.

Fig 12 Login Page

In Fig 12, the user has access to all the information and questions in the app by logging in. This is done through facebook to provide security. The user is prompted to provide his facebook login details such as username and password which is then authenticated as a registered user or not.
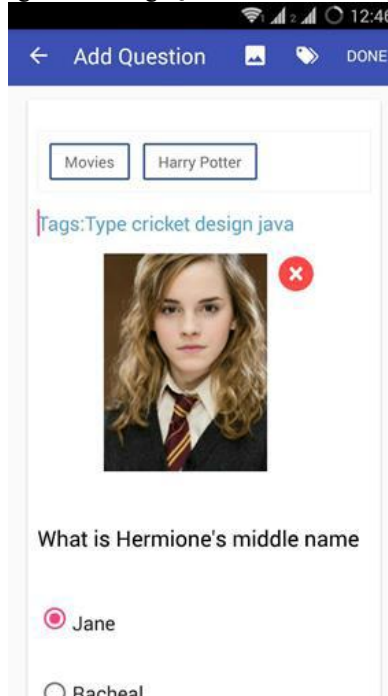
Fig 13: Home Screen
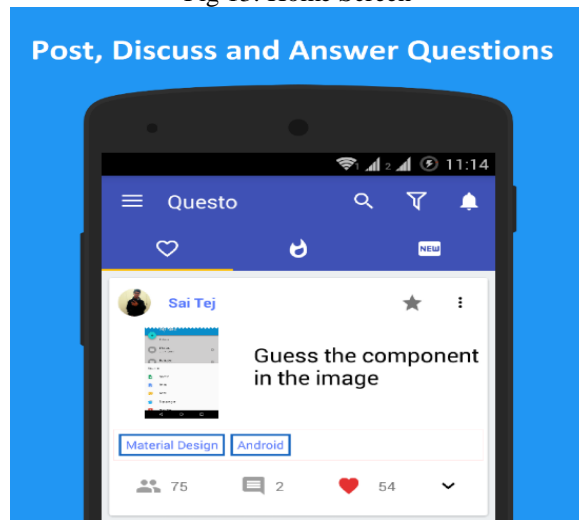
Fig 13 displays the user's home screen which employs the layouts used and gives the user the feed of all the questions, posts, articles posted by the user's friends.

Fig 14: Adding Questions with MCQ's

The Fig 14 displays the related questions attached to the tag the user enters. Posts, articles, MCQ's related to the concerned tag are displayed to the user.
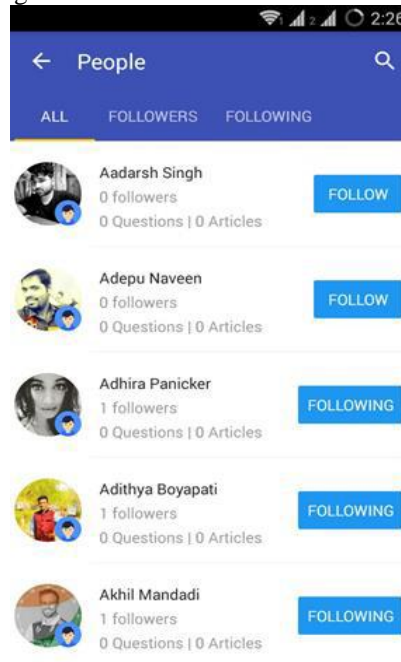
Fig 15: Friends- followers and following

Fig 15 displays the friends, follow requests from other users, the information of the users that a user is following all under the toolbar options All, Followers and Following.

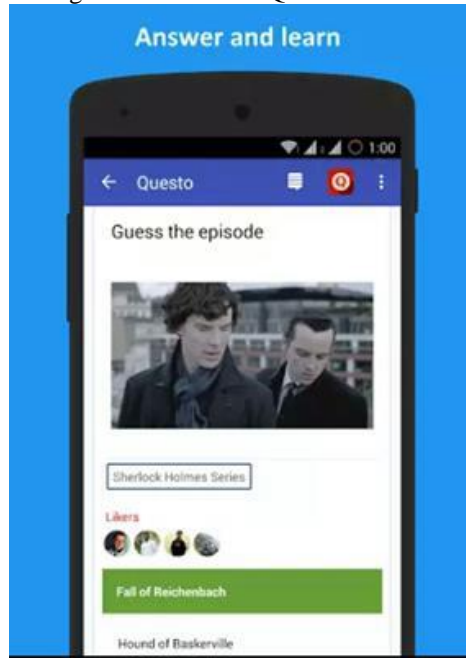Fig 16 Likes for the Question Posted

Fig 16 illustrates the question related data such as the number of likes to a given question, the names of the users that like the question, all which is established in real time without refreshing
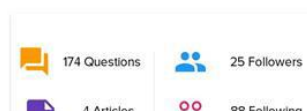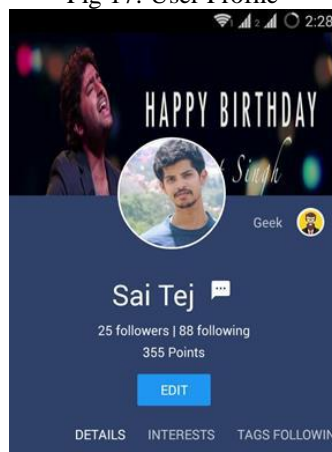
Fig 17: User Profile

Fig 17 illustrates the abstract view of a user's profile that includes information such as a particular user's following, data of the people the user follows, points awarded to the user based on his answering, additional details of the user, the user's interests and the tags the user follows.
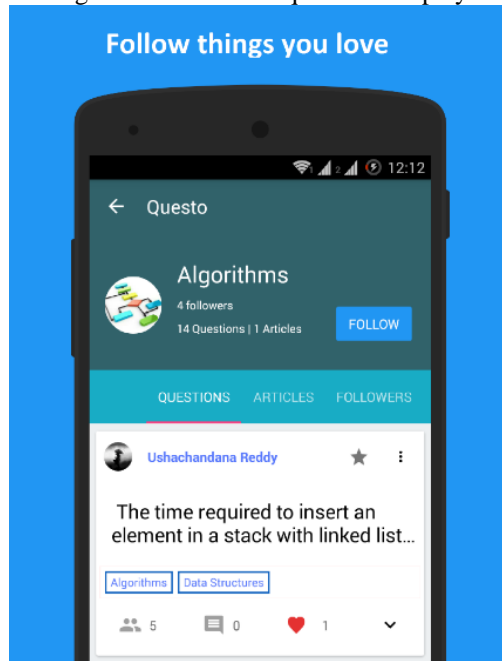
Fig 18: Liked related questions display

Fig 18 the related profiles of the answers that we like by clicking on the heart symbol are displayed, where in the details of all the questions, articles and followers of that profile are displayed. If the user is interested, he can follow such accounts.
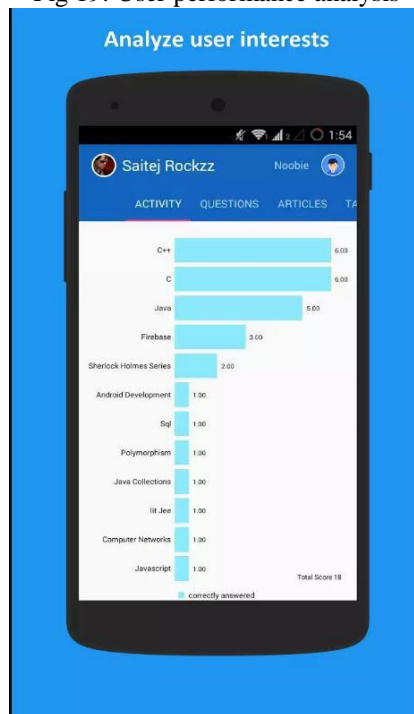
Fig 19: User performance analysis

Fig 19 illustrates the performance analysis in the form of a graph based on the questions the user answer. It provides a comparative analysis of the different domain questions that the user attempts.

6. Conclusion

We have presented an application using crowdsourcing technique, which provides users to interact with other users, post MCQ's, blogs, articles and images. All the data is stored on the cloud - The Firebase real time database which is a platform as a service cloud model. Uploading questions, the likes count, full duplex chat between users happens in real time. The application has been successfully implemented with all the specifications mentioned and deployed on the google play store by the name Questo. There have been around 500 downloads of the Questo app up to date. In future data analytics can be used to analyze the user's interests to suggest similar questions. Also, a method to filter the irrelevant data posted by users as questionnaire, images, blogs or articles can be filtered and reported as spam, to reduce the wrong usage of the application. Voice based question could be added to engage users more into the application.

References

[1]. C. Eickhoff, A. P. de Vries "How crowdsourcable is your task" Proceedings of the Workshop on Crowdsourcing for Search and Data Mining (CSDM 2011) WSDM 2011.
[2]. Android studio, android-developers.googleblog.com
[3]. Firebase console, firebase.google.com
[4]. design.google.com
[5]. Guangyu Zou, Alvaro Gil, Marina Tharayil, "An agent-based model for crowdsourcing systems", Simulation Conference (WSC) 2014 Winter, pp. 407-418, 2014.
[6]. NileshPadhariya, KshamaRaichura, "Crowdlearning: An incentive-based learning platform for crowd", Contemporary Computing (IC3) 2014 Seventh International Conference on, pp. 44-49, 2014.
[7]. Android Debug Bridge, 2015, [online] Available: htttp://developer.android.com/tools/help/adb.html.
[8]. M. M. Fuad, D. Deb, "Design and Development of a Mobile Classroom Response System for Interactive Problem Solving", Proceedings of 26th International Conference on Software Engineering and Knowledge Engineering (SEKE), pp. 49-52, July 1-3.
[9]. "Developerdocs-Firebase",Firebase.com,[online]Available: https://www.firebase.com/docs/.
[10]. General purpose crowdsourcing techniques, http://ieeexplore.ieee.org/document/7456302/