CAPSTONE PROJECT REPORT

(Project Term August-December 2018)

DevHub

Submitted by

Sahil Sharma	**Registration Number :11509689**
Akash Sharma	**Registration Number :11504536**
Anmol Kalra	**Registration Number :11509520**

Project Group Number : CSERGC0004

Course Code : CSE-439

Under the Guidance of

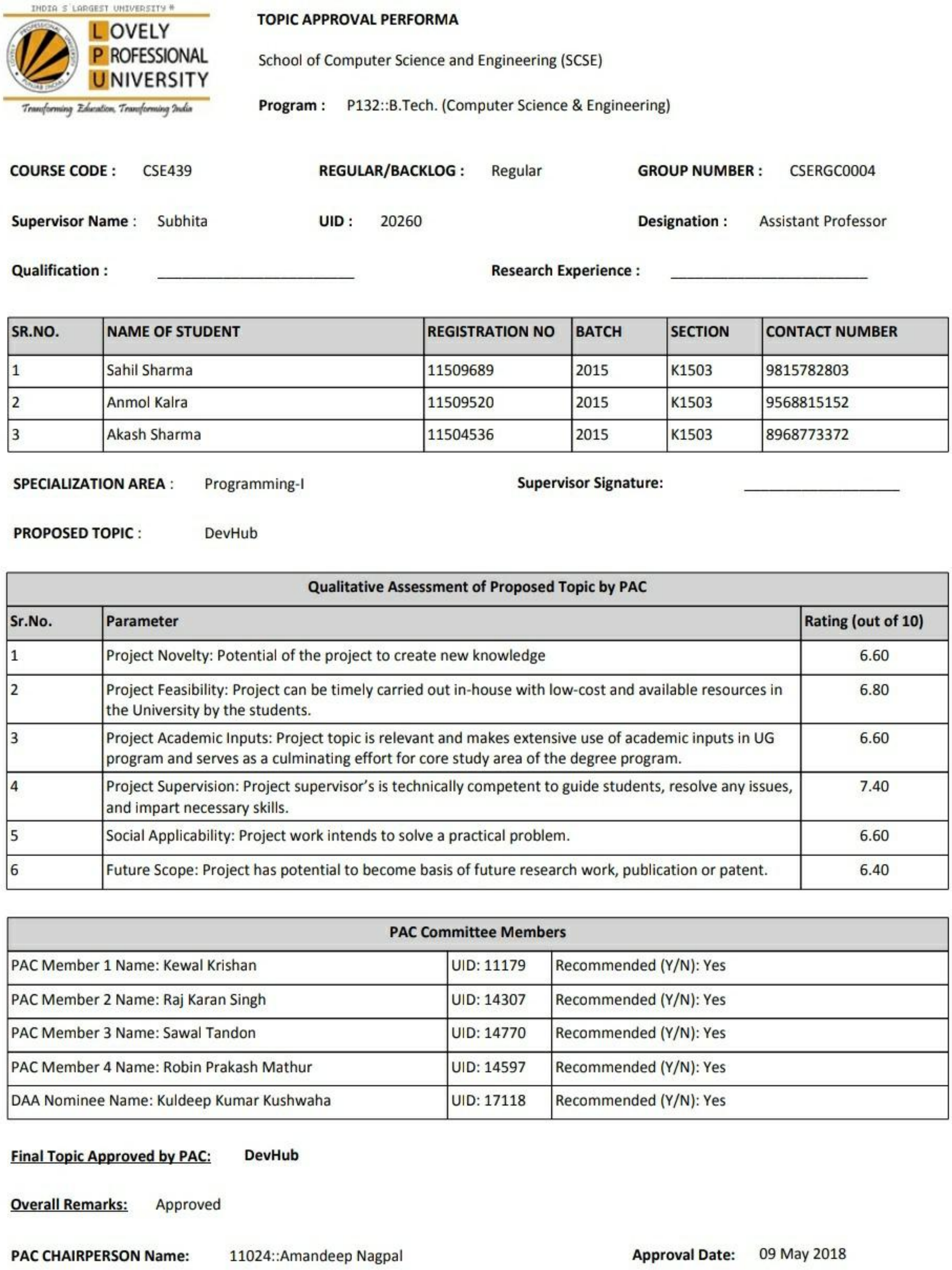
Ms. Subhita Menon

Assisstant Professor

School of Computer Science and Engineering

LOVELY PROFESSIONAL UNIVERSITY

Transforming Education Transforming India



INDIA'S LARGEST UNIVERSITY

LOVELY PROFESSIONAL UNIVERSITY

Transforming Education, Transforming India

TOPIC APPROVAL PERFORMA

School of Computer Science and Engineering (SCSE)

Program : P132::B.Tech. (Computer Science & Engineering)

COURSE CODE : CSE439 **REGULAR/BACKLOG :** Regular **GROUP NUMBER :** CSERGC0004

Supervisor Name : Subhita **UID :** 20260 **Designation :** Assistant Professor

Qualification : ____________________ **Research Experience :** ____________________

SR.NO.	NAME OF STUDENT	REGISTRATION NO	BATCH	SECTION	CONTACT NUMBER
1	Sahil Sharma	11509689	2015	K1503	9815782803
2	Anmol Kalra	11509520	2015	K1503	9568815152
3	Akash Sharma	11504536	2015	K1503	8968773372

SPECIALIZATION AREA : Programming-I **Supervisor Signature:** ________________

PROPOSED TOPIC : DevHub

Qualitative Assessment of Proposed Topic by PAC		
Sr.No.	**Parameter**	**Rating (out of 10)**
1	Project Novelty: Potential of the project to create new knowledge	6.60
2	Project Feasibility: Project can be timely carried out in-house with low-cost and available resources in the University by the students.	6.80
3	Project Academic Inputs: Project topic is relevant and makes extensive use of academic inputs in UG program and serves as a culminating effort for core study area of the degree program.	6.60
4	Project Supervision: Project supervisor's is technically competent to guide students, resolve any issues, and impart necessary skills.	7.40
5	Social Applicability: Project work intends to solve a practical problem.	6.60
6	Future Scope: Project has potential to become basis of future research work, publication or patent.	6.40

PAC Committee Members		
PAC Member 1 Name: Kewal Krishan	UID: 11179	Recommended (Y/N): Yes
PAC Member 2 Name: Raj Karan Singh	UID: 14307	Recommended (Y/N): Yes
PAC Member 3 Name: Sawal Tandon	UID: 14770	Recommended (Y/N): Yes
PAC Member 4 Name: Robin Prakash Mathur	UID: 14597	Recommended (Y/N): Yes
DAA Nominee Name: Kuldeep Kumar Kushwaha	UID: 17118	Recommended (Y/N): Yes

Final Topic Approved by PAC: **DevHub**

Overall Remarks: Approved

PAC CHAIRPERSON Name: 11024::Amandeep Nagpal **Approval Date:** 09 May 2018

Declaration

We hereby declare that the capstone work entitled DevHub Using Android Application Development is an authentic record of our own work carried out in B.Tech degree in Computer Science and Engineering from Lovely Professional University, Phagwara, under the guidance of Ms. Subhita Menon, during January to May 2018. All the information furnished in this capstone report is based on our own intensive work and is genuine.

Project Group Number : CSERGC0004

Sahil Sharma
11509689

Akash Sharma
11504536

Anmol Kalra
11509520

Date:

Date:

Date:

CERTIFICATE

This is to certify that the declaration statement made by this group of students is correct to the best of my knowledge and belief. They have completed this Capstone Project under my guidance and supervision. The present work is the result of their original investigation, effort and study. No part of the work has ever been submitted for any other degree at any University. The Capstone Project is fit for the submission and partial fulfillment of the conditions for the award of B.Tech degree in Computer Science and Engineering from Lovely Professional University, Phagwara.

Ms. Subhita Menon

Assistant Professor

School of Computer Science and Engineering, Lovely Professional
University, Phagwara, Punjab.

Date :

ACKNOWLEDGEMENT

We take this opportunity to present our votes of thanks to all those guideposts who really acted as lightening pillars to enlighten our way throughout this project that has led to successful and satisfactory completion of this study.

We are grateful to Ms. Subhita Menon for providing us with an opportunity to undertake this capstone project and providing us with all the facilities. We are highly thankful to mam for her active support, valuable time and advice, whole-hearted guidance, sincere cooperation, and pains-taking involvement during the study and in completing the assignment of preparing the said case study within the time stipulated.

Lastly, we are thankful to all those, particularly the various friends, who have been instrumental in creating proper, healthy, and conductive environment and including new and fresh innovative ideas for us during the capstone project, without their help, it would have been extremely difficult for us to prepare it in a time bound framework.

TABLE OF CONTENTS

1. INTRODUCTION

1.1 DESCRIPTION

DevHub is a native Android Application developed for android devices. The application serves as a platform for users to share their projects and collaborating with different developers to work on their projects in different domains and having access to share their problems and find the answer to them by asking then in the domain specific discussion forum. It provides online chatting and discussion forums which includes Individual chat (one to one chatting) and Group chat options and a section where the access to the chat has been given to all the people who have joined that domain. It has different modules which are a News module which keeps the track of technical updates happening in the world of modern technology, a quiz module with three different sections and difficulty levels where users can test their programming languages capability and a module for learning which refers to different websites for learning different technologies, programming languages and other technical things.

1.2 OBJECTIVES

1. The main idea behind this project is to gather the people of different domain together so that they could collaborate with each other and together work on a project on a single platform through a mobile application.
2. In this application users can add their projects and can get information of other projects and work as a team in competition of the project.
3. Users can resolve their issue through discussion in the group chat discussion forum.
4. Users can get information of different domains in single hand.
5. User can get updated with the current technical news.

2. PROFILE OF THE PROBLEM

The purpose of this application is to fulfill the requirements of students and developers today on a single application based platform, where they can share their project ideas through chat activities one can share their ideas with an individual person or can share with a group.

User can get information regarding important coding competitions and hackathons going on periodically throughout the year and he can judge himself by reading the eligibility criteria and information regarding event pattern..

Users can learn different programming languages through the functionality of quick quiz. Moreover, user can get the information about various technologies.

User can get information of the latest technological trends, which have been updated whenever a new update or product related to technology releases in the market.

Every module of the application has an important part and totally differ from each other that provides a variety of things to be experienced in the application. Combining different features like chatting, tech feeds, competition feeds and learning modules together make it different from other applications present in the market.

Application can work on any Android operating mobile device with following specifications:

1. Minimum API level 21 i.e. Jelly Bean.
2. Minimum 1GB of RAM (Random Access Memory)
3. Good Internet connection.

3. EXISTING SYSTEM

3.1 INTRODUCTION

1.The existing system provides different technical aspects and information regarding field of programming languages, ideas sharing which is not available on a single application.

2. Different information is provided on run-time.

3. Application is user friendly and is compatible on almost every mobile device.

.

3.2 EXISTING SOFTWARE

1. Users can read the latest technical trending information with the trending module which refreshes on a regular basis.

2. There are six different modules in application which meets different requirements of user regarding technical information.

3. Most applications and websites do not provide the option of chatting one to one with other users, An individual can post his or hers queries globally to share among the different users for their solutions and suggestions.

4.Reference to different websites is made to learn different programming languages and technologies using webviews.

5.Application has a quick quiz module with questions of C, C++ and Java with easy, medium and high levels.

3.3 DFD FOR PRESENT SYSTEM

CONTEXT LEVEL DIAGRAM

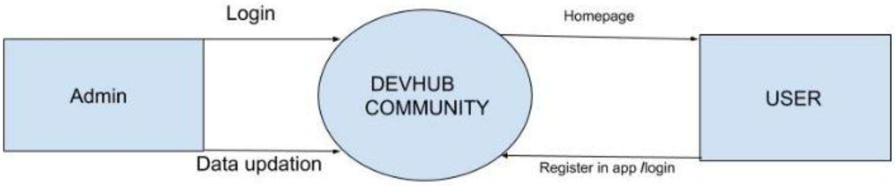


Fig. 1

FIRST LEVEL DIAGRAM

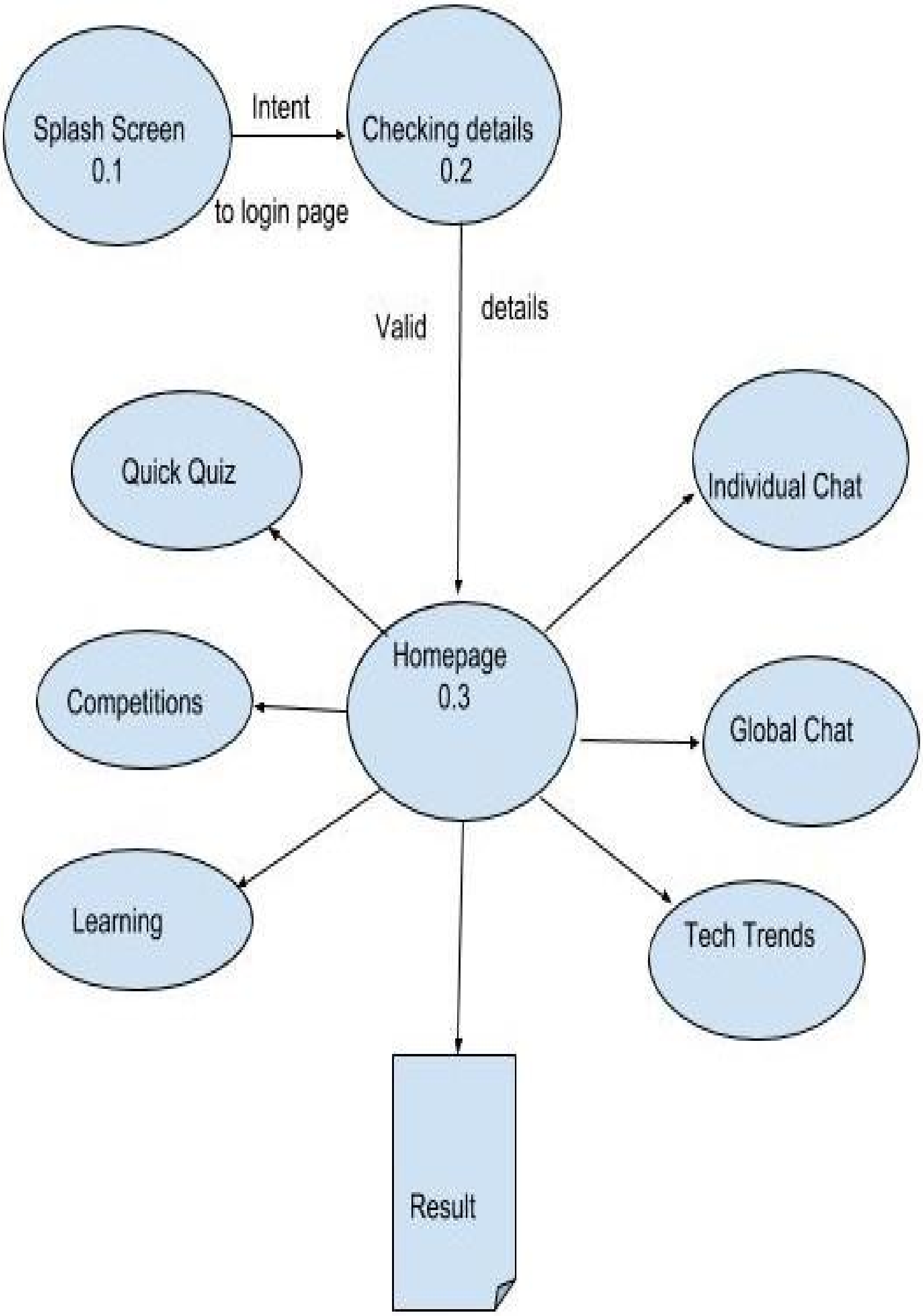


Fig. 2

SECOND LEVEL DIAGRAM

USER LEVEL

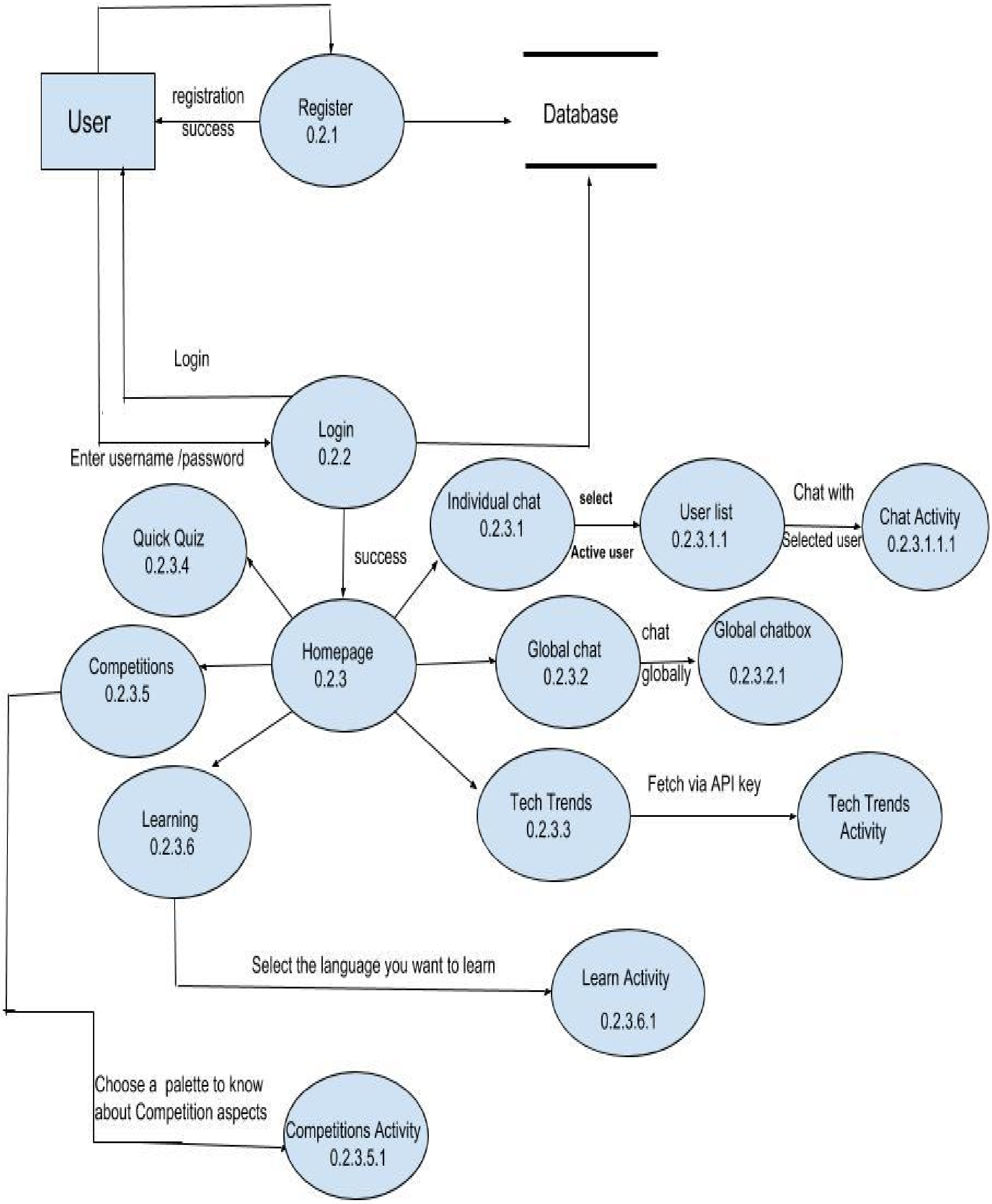
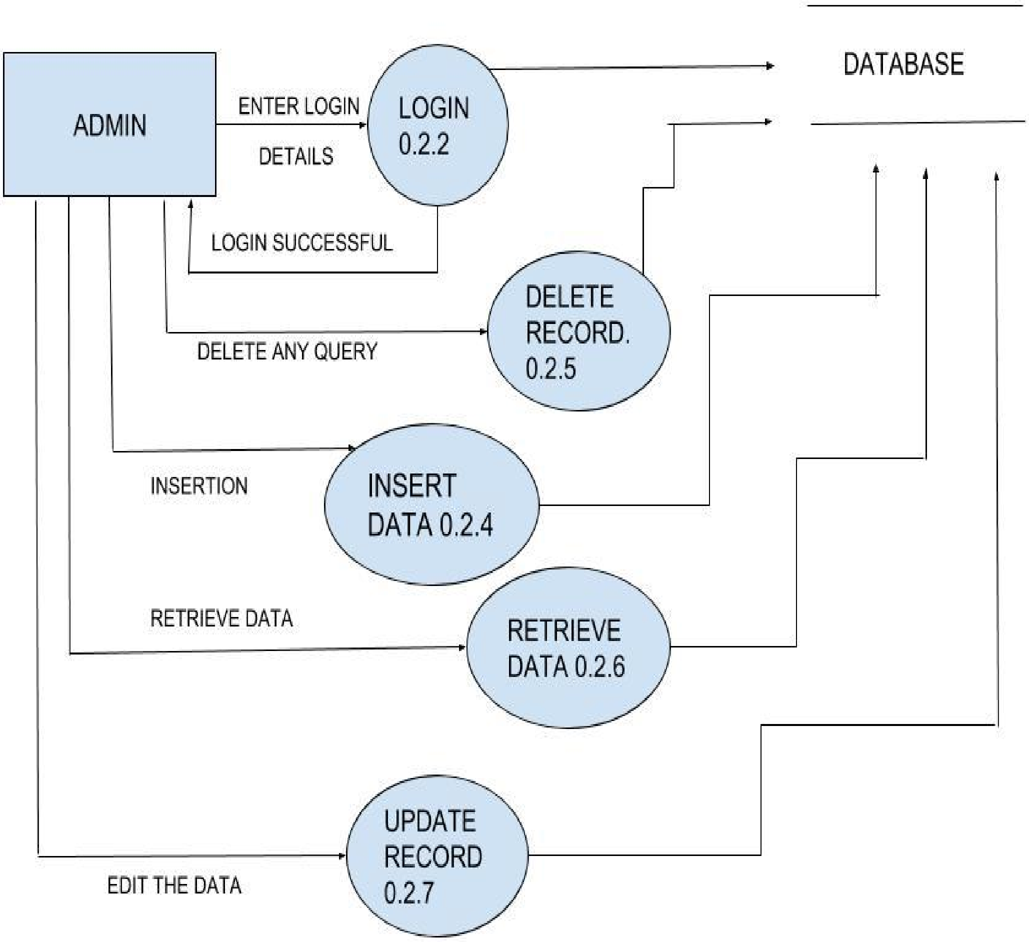


Fig. 3

ADMIN LEVEL



DATABASE USED : AWS PARSE SERVER BITNAMI v0.2.3.1

Fig. 4

3.4 WHAT'S NEW IN THE SYSTEM TO BE DEVELOPED

1. In this applications we have made a platform for the users to assess their knowledge through participating in the quiz modules. The users are given the functionality of choosing the difficulty levels in the quiz and take part in it accordingly.

2. The users can share their views on the chatting sections in two ways either through individual chatting section using one on one chatting mechanism or through global chat section i.e. sharing their views to all the users of the application at once.

3. Moreover users can get information regarding different competitions and Hackathons that take place throughout the year. Participation in these competition would help in enhancing the knowledge and capability of doing work.

4. Collapsing tool bar with problem discussion forum and comment section can be built to add more specifications of different domains.

4. PROBLEM ANALYSIS

4.1 PROBLEM DEFINITION

1. In the existing system we need to specifically get users to interact each other with their different ideas and to share their problems.

2. To get information for different competitions organizing by different companies and compiling on a single platform.

3.To make reference to different websites for the users to learn different topics through a single application only.

Example:

There are websites like stackoverflow.com which usually provides solutions to different problems asked by the user but there are very less applications which works on these areas. DevHub is a collection of different of modules which help users to build interest in programming and computer science.

4.2 FEASIBILITY ANALYSIS

The feasibility for the project is done on the basis of the parameters like time required, workers needed, hardware and software requirements, cost involved in the building of this application.

Resources that are required to build up this project and make the testing system live on the internet are:

Time – There are about 3-4 months to build up this project, starting from mid-January 2018 to end of November.

Workers – The team must consist of the members skilled in coding of Java and must be familiar with the concepts of Android Application Development.

Hardware – A system with the following minimum configurations is needed for the project to be built on time and with no constraints from hardware:

1. Laptop with minimum 4 GB RAM.

2. In case of absence of emulator an android mobile device can be used for checking the working of the application connecting through a USB Cable.

3. Software - Android Studio.

This project was found to be completely feasible after the analysis done on the basis of the above mention parameters.

4.3 PROJECT PLAN

The application DevHub is made to work on internet connection only. This would require the groundwork to be done before starting the projects. Certain calculations are done on the basis of the available resources for the project.

1. There are 3 members in the group. The workload is to be divided according to the areas of interest.

2. Time available to complete the project is 3-4 months.

3. Modules such as coding, testing and designing should are done more carefully and more time is spent on these to ensure quality of the project.

5. SOFTWARE REQUIREMENT ANALYSIS

5.1 INTRODUCTION

To build a certain project, there is a requirement of some specific hardware and software that must be available whenever needed. The analysis of the requirements to complete the project, firstly, can be shown on the minimum basis which are just enough to complete the project.

If the minimum requirements of hardware and software are not provide, then it is not possible to complete that project.

But for the developing of a project to be smooth and without crashes and delays, there is a need to provide recommended or suggested requirements for the system. These requirements are carried out by system analyst and publish to the whole team.

5.2 GENERAL DESCRIPTION

The minimum hardware requirements for the project are as follows:

1. CPU with Intel Core i3 with 2.50 GHZ processor.

2. Hard disk capacity of 40 GB.

3. Memory of 4GB of RAM.

4. Monitor for display.

5. Keyboard and a mouse.

The minimum software requirements for the project are:

Server side:

Front End – XML (Extensible Markup Language)

Back End – AWS (Amazon Web Services) EC2 instance (T2.micro), Parse Server certified by Bitnami 2-7-2-0 on Ubuntu 16-04-Autogen by AWS mp.

Tool End – Android Studio

Client Side:

Operating System – Android

Run Time Environment - Android

Minimum requirements is barely some basic needs to complete the project. But to complete the project with no hick-ups there are recommended requirements for the current project.

Recommended Requirements:

The recommended hardware requirements for the project are as follows:

1. CPU of Intel i3 processor with above 2.5 GHZ clock cycle.

2. Hard disk space free should be above 20 GB

3. Memory of 4 GB of RAM

4. Monitor for display of high resolution

5. Keyboard and a mouse.

The recommended software requirements for the project are:
Server side:

1.Front End – XML (Extensible Markup Language)

2. Back End – AWS (Amazon Web Services) EC2 instance (T2.micro), Parse Server certified by Bitnami 2-7-2-0 on Ubuntu 16-04-Autogen by AWS mp.

3. Tool End – Android Studio

Client Side:

1. Operating System – Android

2. Run Time Environment - Android

6. DESIGN

6.1 SYSTEM DESIGN FLOW DIAGRAM

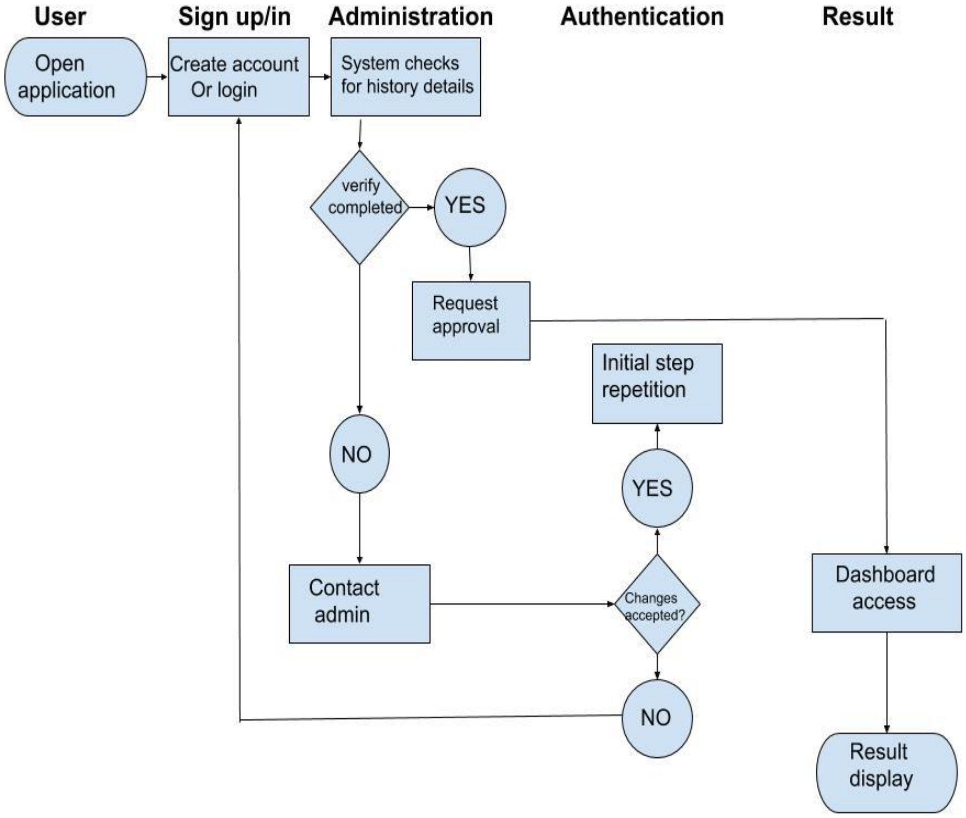
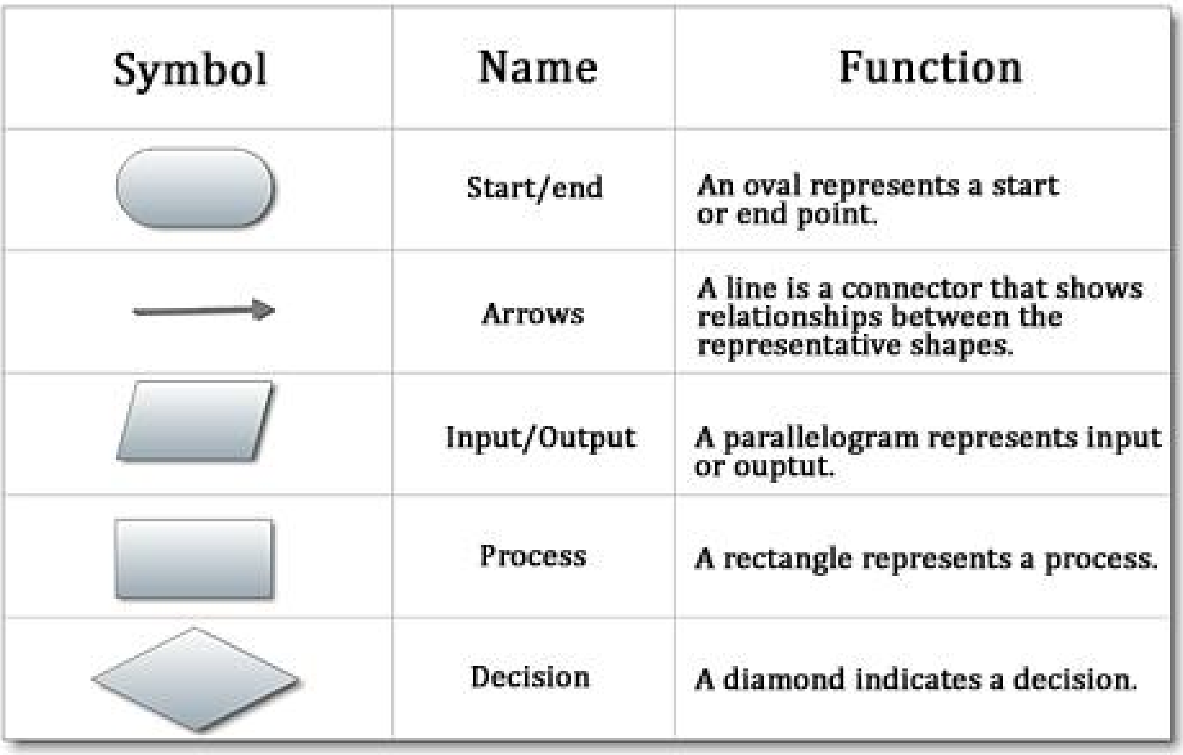


Fig. 5

6.2 DESIGN NOTATIONS



Symbol	Name	Function
⬭	Start/end	An oval represents a start or end point.
→	Arrows	A line is a connector that shows relationships between the representative shapes.
▱	Input/Output	A parallelogram represents input or ouptut.
▭	Process	A rectangle represents a process.
◇	Decision	A diamond indicates a decision.

Fig. 6

6.3 DETAILED DESIGN

CONTEXT LEVEL DESIGN

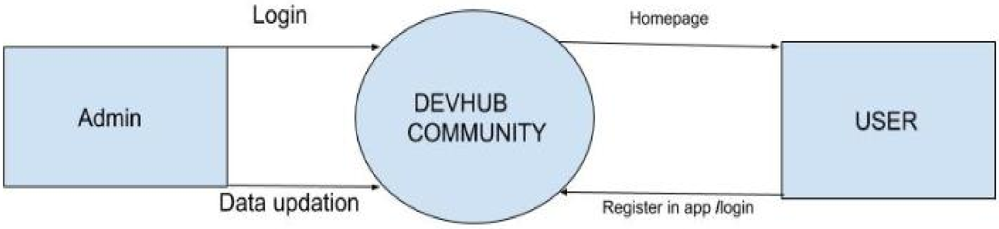


Fig. 7

FIRST LEVEL DESIGN

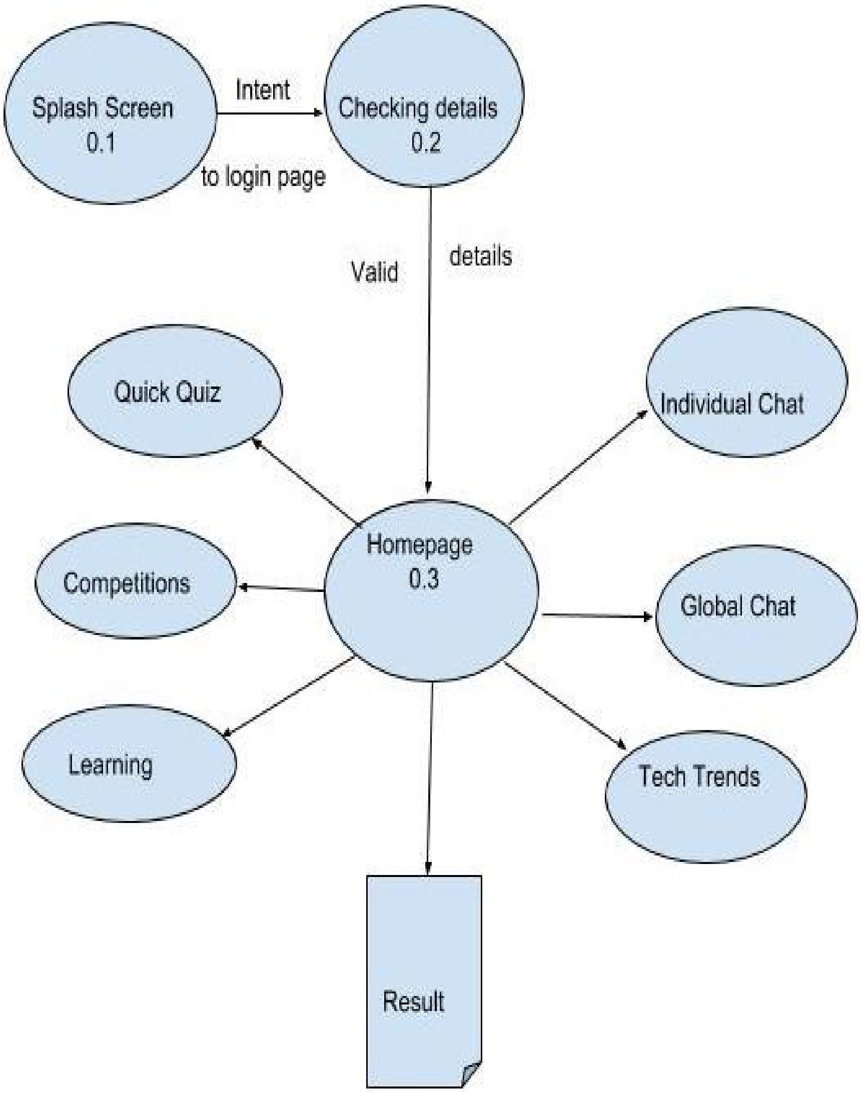


Fig. 8

SECOND LEVEL DESIGN

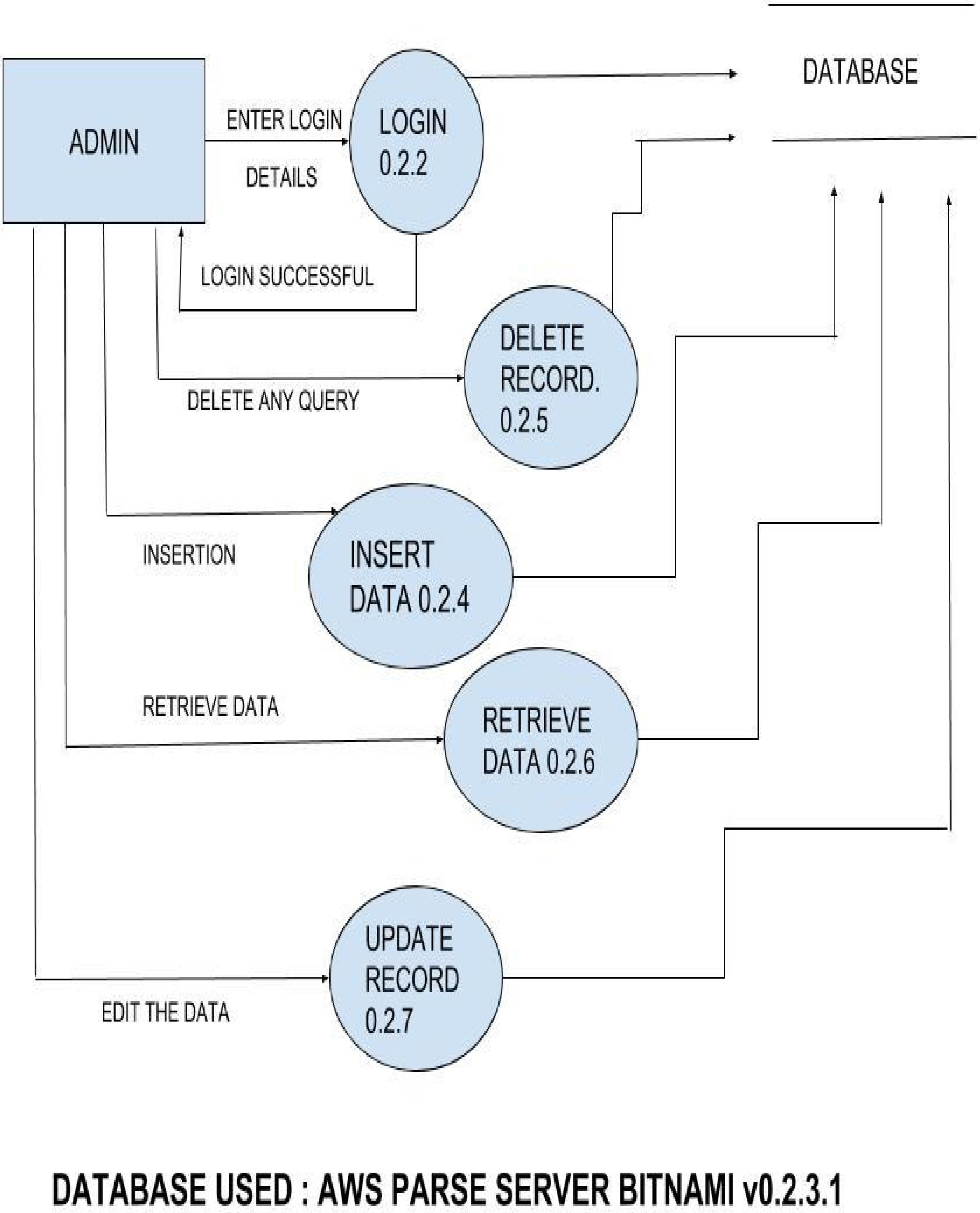


Fig. 9

6.4 FLOW CHARTS

6.4.1 REGISTRATION MODULE

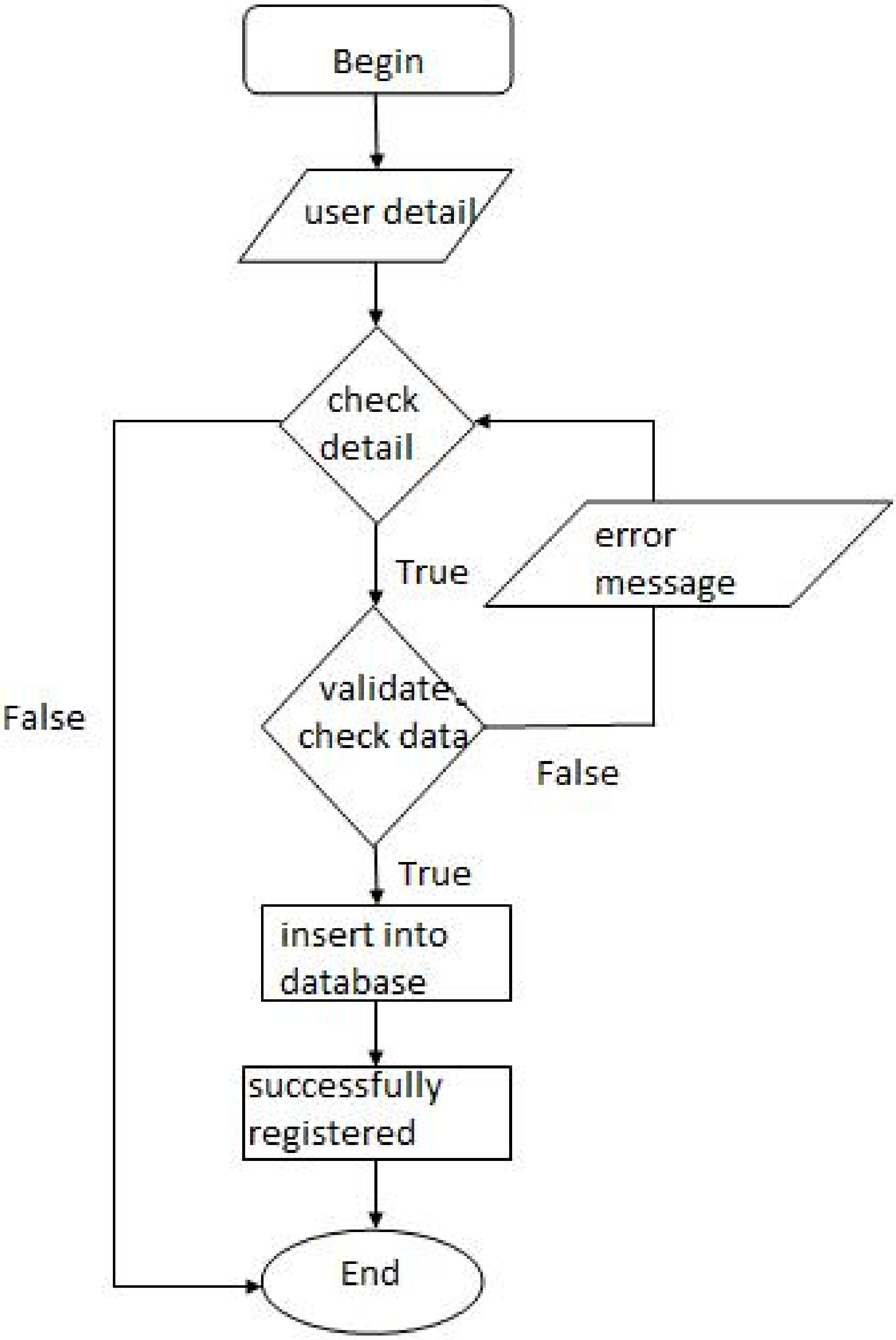


Fig. 10

6.4.2 LOGOUT MODULE

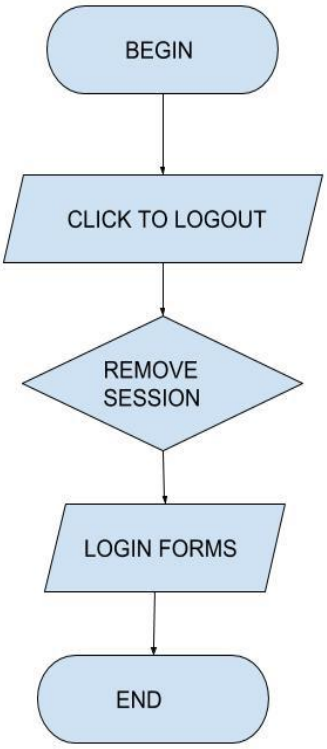


Fig. 11

6.5 PSEUDOCODE

6.5.1 SIGN UP

```
1.  GET(USERNAME, PASSWORD)
    1.1. SAVE IN BACKGROUND(USERNAME, PASSWORD).
    1.2. ADD TO SERVER(USERNAME,PASSWORD)
    1.3. WAIT FOR RESPONSE VALUE(NULL)
    1.4. PRINT VALIDATION ERROR:
            1.4.1 IF:VALIDATION ERROR VALUE NOT EQUALS(NULL).
    1.5. SAVE DATA SUCCESSFULLY.
```

6.5.2 LOGIN

```
1.  GET EDITTEXT(USERNAME,PASSWORD).
2.  FIND IN BACKGROUND(USERNAME,PASSWORD).
3.  GET RESPONSE RESULT VALUE:
    3.1 IF:NULL THEN ALLOW LOGGED IN.
    3.2 ELSE: LOGIN FAILED.
```

6.5.3 HOME PAGE

```
1. SELECT(CARD VIEW)
2. IF:CARD VIEW(VALUE ( INDIVIDUAL CHAT))
   2.1. REDIRECT (INDIVIDUAL CHAT ACTIVITY)
   ELSE IF:CARD VIEW(VALUE(GROUP CHAT))
   2.2. REDIRECT(GROUP CHAT ACTIVITY)
   ELSE IF:CARD VIEW(VALUE(LEARNING))
   2.3. REDIRECT(LEARNING ACTIVITY))
   ELSE IF:CARD VIEW(VALUE(QUICK QUIZ))
   2.4. REDIRECT(QUICK QUIZ ACTIVITY))
   ELSE IF:CARD VIEW(VALUE(TECH TRENDS))
   2.5. REDIRECT(TECH TRENDS ACTIVITY)
   ELSE IF:CARD VIEW(VALUE(COMPETITIONS))
   2.6. REDIRECT(COMPETITIONS ACTIVITY))
```

7. TESTING

7.1 FUNCTIONAL TESTING

Functional testing (also known as black-box testing) is the process of verifying that a system or system component adheres to the specification that defines its requirements. Functional testing can be performed at the system level or the unit level.

To perform functional testing, you typically create a set of input/outcome relationships that verify whether each specification requirement is implemented correctly. At least one test case should be created for each entry in the specification document.

These test cases should test the various boundary conditions for each entry. After the test suite is ready, you execute the test cases and verify whether the correct outcomes are produced.

In our project the functional testing is simply performed by running the project on mobile devices. If the project successfully runs as desired, functional testing is successful. The project was run on multiple devices to check the compatibility among them.

In functional testing each of the six modules i.e. Group chat, Individual Chat, Learning, Tech Trends, Competitions and Quick Quiz are tested according to the mechanism of integration testing. These modules are build separately at first, and then Integrated one by one among each other. Basically combining smaller modules to form a integrated bigger module.

7.2 STRUCTURAL TESTING

Structural testing is often referred to as ‘white box’ or ‘glass box’ or ‘clear-box testing’ because in structural testing we are interested in what is happening ‘inside the system/application’. In structural testing the testers are required to have the knowledge of the internal implementations of the code. Here the testers require knowledge of how the software is implemented, how it works. The structural testing of our project is done in the Android Studio.

It checks or tests all the coding portion of every module. It debugs and compiles the code while the project is prompted to run in browser. If there is an error, that particular line of code which contains the error is highlighted. It can then fixed and run again to check if correction made the project work and so on.

7.3 LEVELS OF TESTING

There are four types of testing which can be implemented, which are as follows:

Unit Testing

Integration Testing

System Testing

Acceptance Testing

In Unit Testing, we tested individual components like each controls for their validations to ensure that they operate correctly.

The next level is called Integration Testing. In this many unit tested modules are combined into subsystems, which are then tested. The goal here is to see if the modules can be integrated properly. This testing activity can be considered testing the design.

In System Testing, we tested whether system elements have been properly integrated and perform allocated functions to detect the errors that may result from unanticipated interactions between sub-system and system components.

Finally, in Acceptance Testing, we tested whether the system is accepted for operational use or not.

7.4 TESTING THE PROJECT

DevHub is an application that provides the functionality of chatting, learning, reading and for this the major testing has been done only to authenticate the user so that the data integrity is maintained and no misuse of the data takes place.

The testing of the project has been done for the username and password entered at the login or signup page. If the user gets the access to the account of other user than this lacks the privacy of other users. So this has been carefully tested.

The testing of this has been done on the following was and certain test cases are presented in the following table.

STRUCTURE OF TEST-CASE

S.No	Inputs	Expected Results	Actual Results	Remarks
1.	In here,users enter the different sets of username and password.	The authorized user has got the access to the whole app modules.	In actual, how the application has response according to the username and password entered.	The response according to application if it is positive or negative.

Table 1

TEST CASE OF LOGIN MODULE

Case: Login for Employee as Well as Admin					
S.No	Inputs		Expected Result	Actual Result	Remarks
1.	Username: xyz (correct)	Password: wxyz (incorrect)	Login attempt failed (password incorrect)	Login attempt was not successful	Positive
2.	Username: xyz	Password: xyz	Login attempt successful	Login attempt successful	Positive
3.	Username: no username entered	Password: xyz	Login attempt failed (password incorrect)	Login attempt was not successful	Positive
4.	Username: xyz	Password: no password entered	Login attempt failed (password incorrect)	Login attempt was not successful	Positive
5.	Username: no username entered	Password: no password entered	Login attempt failed (password incorrect)	Login attempt was not successful	Positive

Table 2

8. IMPLEMENTATION

Implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through computer programming and deployment. Many implementations may exist for a given specification or standard. For example, web browsers contain implementations of World Wide Web Consortium-recommended specifications, and software development tools contain implementations of programming languages.

PROCESS

Software process as a framework for the tasks that are required to build high quality software. Or we can say that process defines the approach that is taken as software is engineered.

PROCESS MODEL USED BY US

Spiral model

The Spiral model originally proposed by Boehm, is an evolutionary software process model that couples the iterative nature of prototyping with the controlled & systematic aspects of the linear sequential model . It provides the potential for rapid development of incremental version of the software.

Using the spiral model software is developed in a series of incremental releases. A spiral model is divided into a number of framework activities also called task regions. A spiral model contains six task regions:

1. Customer Communication: Tasks required to establish effective communication between developer & customer.

2. Planning: Tasks required to define resources, timeline & other project related information.

3. Risk analysis: Task required to access.

4. Engineering: Tasks required to build one or more representation of the application.

5. Construction & release: Task required to construct, test, install & provide user support (e.g., documentation & Training)

6. Customer evaluation: Tasks required to obtain customer feedback based on evolution of the software representation created during the engineering stage & implemented during the installation stage.

8.1 CONVERSION PLAN

To make this project live, i.e., to build APK file for the project followed:

1. Installation of Android Studio.

2. Build project with atleast level 21 API because of server connectivity .

3. Once the android studio set up successfully,ask AWS for Elastic Cloud Computing version 2(EC2 t2.micro).

4. The provider give live instance of Bitnami Parse Server Version 2 and instance id:i-0995333c6213c67cd and DNS address:ec2-18-188-58-24.

5. We can connect to AWS Server using Server address:"18.88.58.24:80/parse.

6. We have to give read and write permission and create class instances for each activity.

7. The application is now live and open for public use.

8.2 POST IMPLEMENTATION OF PROJECT AND MAINTAINANCE

The Post Implementation Review (PIR) is conducted after a project has been completed. The purpose of the PIR is to evaluate how successfully the project objectives have been met and how effective the project management practices were in keeping the project on track.

In our project the all objectives met to the requirements and it is more affective as user wants. According to the user requirements the project functionality and objectives are made according to his.

It is generally found that systems that are easy to use, require less manpower, saves the data entry and well received by people. But still the following points have to consider.

1. How have systems changed the way in which operations were performed?

2. How have systems changed the timeliness of information and reports user received?

9. PROJECT LEGACY

9.1 CURRENT STATE OF PROJECT

The current status of our project is that all modules like login, home page, admin page, test page, search page, summary page of the project are completed and their design, coding and testing are done.The application is completely developed and tested.

9.2 REMAINING AREAS OF CONCERN

There are still, after a lots of efforts, the areas of concern in the project. To make smart features in chat section by sharing files of code and enhancing the notifications to be send to the user. To reduce the loading time of modules.

9.3 TECHNICAL AND MANAGERIAL LESSONS LEARNT

We have learnt a lot of things while developing the project.

1. Working with the Android Studio.

2. Working with server side tasks(AWS).

3. Creating API calls and other functions using them.

4. Creation of notifications in android.

5. Creating and managing databases using SQLite class and AWS Parse Server.

6. Working in a team.

7. Problem Analysis and problem solving with the team mates.

10. USER MANUAL

10.1. SIGN UP

1. Open the app.

2. Click on Signup Button.

3. Enter the user id with which you want to create your account.

4. Enter the password for future authentication.

10.2. LOGIN

1. Open the app.

2. If registered user, enter username with which you have registered yourself and password.

3. If the credentials are correct, you will be logged in and redirected to Homepage.

10.3. HOME PAGE

1. After successful login, you will be redirected to Homepage.

2. Homepage contains six basic modules that includes learning, tech trends, individual chat, global chat, competitions and quick quiz.

3. Click on any option to see following results –

   3.1. Click on learning to learn different programming languages.

   3.2. Click on Individual chat to chat personally with other users.

   3.3. Click on Global chat to broadcast your message to entire community.

   3.4. Click on Competitions to see pallets contain information about important events.

   3.5. Click on quick quiz to give a short quiz to test your programming skills.

   3.6. Click on tech trends to read latest development and enhancement in technology.

10.3.1. LEARNING

1. In learning, user have option to select a programming language of his choice and learn both basic and advance concepts in detail.

2. User can click on others option to learn any other technology.

10.3.2. TECH TRENDS

1. Tech trends let user read about latest technology development and new technical products.

2. Tech trends updated automatically whenever a new technology update is released in market.

10.3.3. COMPETITIONS

1. User can see different and important hackathons and events that occur periodically and give user brief about all those events.

2. User can click any pallets to read the content related to respective event and can move up and down to see other events.

10.3.4. QUICK QUIZ

1. User can give test on certain programming languages like C, C++ and Java.

2. User can change difficulty level.

10.3.5. INDIVIDUAL CHAT

1. User can select other user from user list and redirected to message box.

2. Message box contains chat box and send button where user can type their message and send them as well as read received messages.

10.3.6. GLOBAL CHAT

1. User can broadcast his/her problem to all other users.

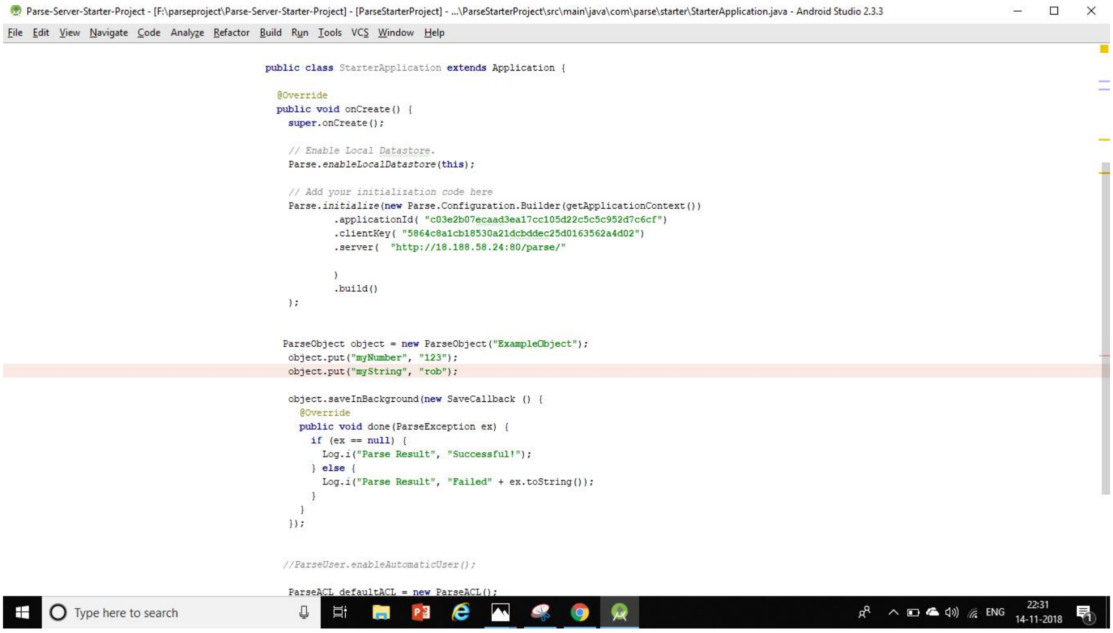
2. Any user is able to answer that general query.

11. SOURCE CODE

11.1 APPLICATION CODE

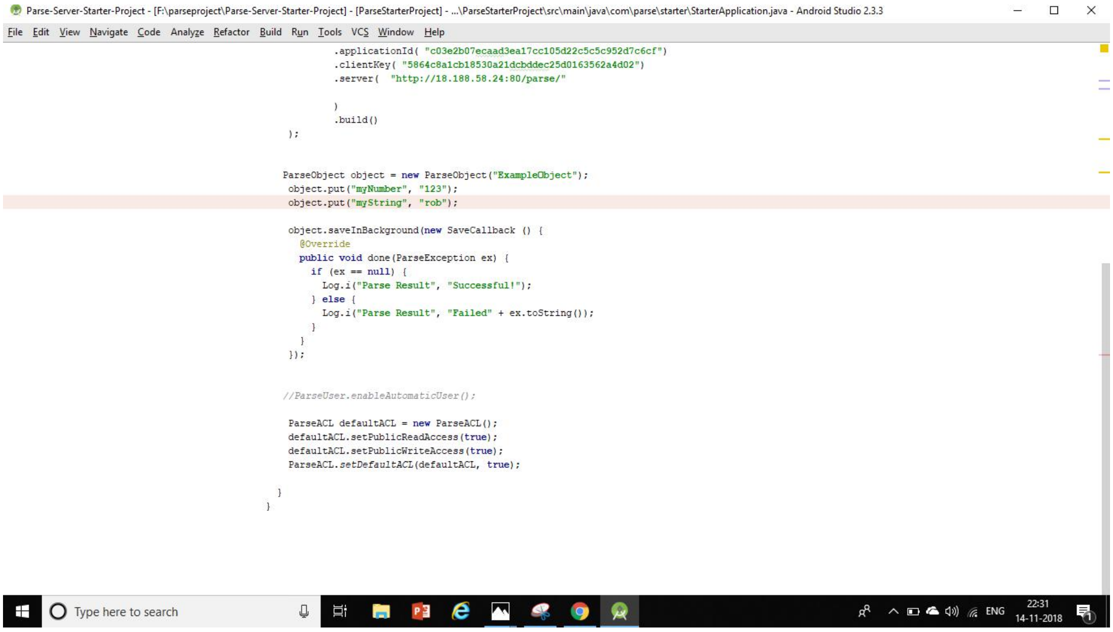
1. CONNECTION SETUP

In this activity the application is connected to the server using instance id and master key.


Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\StarterApplication.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:31
14-11-2018

```java
public class StarterApplication extends Application {

  @Override
  public void onCreate() {
    super.onCreate();

    // Enable Local Datastore.
    Parse.enableLocalDatastore(this);

    // Add your initialization code here
    Parse.initialize(new Parse.Configuration.Builder(getApplicationContext())
            .applicationId( "c03e2b07ecaad3ea17cc105d22c5c5c952d7c6cf")
            .clientKey( "5864c8a1cb18530a21dcbddec25d0163562a4d02")
            .server(  "http://18.188.58.24:80/parse/"

            )
            .build()
    );


   ParseObject object = new ParseObject("ExampleObject");
    object.put("myNumber", "123");
    object.put("myString", "rob");

    object.saveInBackground(new SaveCallback () {
      @Override
      public void done(ParseException ex) {
        if (ex == null) {
          Log.i("Parse Result", "Successful!");
        } else {
          Log.i("Parse Result", "Failed" + ex.toString());
        }
      }
    });


   //ParseUser.enableAutomaticUser();

    ParseACL defaultACL = new ParseACL();
```

Fig. 12

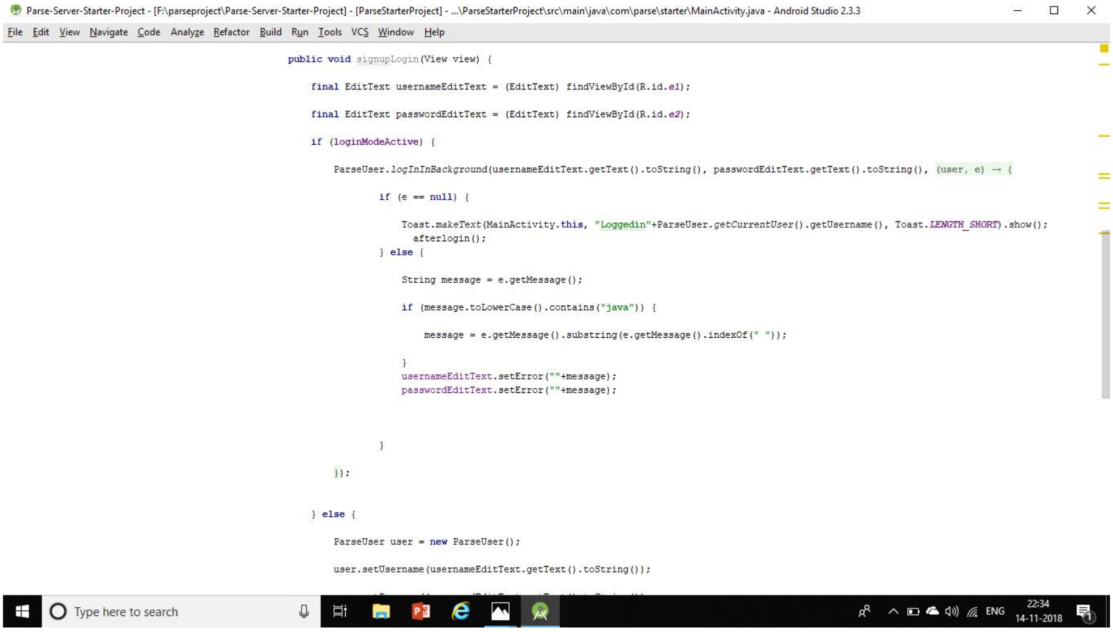

Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\StarterApplication.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:31
14-11-2018

```java
            .applicationId( "c03e2b07ecaad3ea17cc105d22c5c5c952d7c6cf")
            .clientKey( "5864c8a1cb18530a21dcbddec25d0163562a4d02")
            .server(  "http://18.188.58.24:80/parse/"

            )
            .build()
    );


   ParseObject object = new ParseObject("ExampleObject");
    object.put("myNumber", "123");
    object.put("myString", "rob");

    object.saveInBackground(new SaveCallback () {
      @Override
      public void done(ParseException ex) {
        if (ex == null) {
          Log.i("Parse Result", "Successful!");
        } else {
          Log.i("Parse Result", "Failed" + ex.toString());
        }
      }
    });


   //ParseUser.enableAutomaticUser();

    ParseACL defaultACL = new ParseACL();
    defaultACL.setPublicReadAccess(true);
    defaultACL.setPublicWriteAccess(true);
    ParseACL.setDefaultACL(defaultACL, true);

  }
}
```

Fig. 13

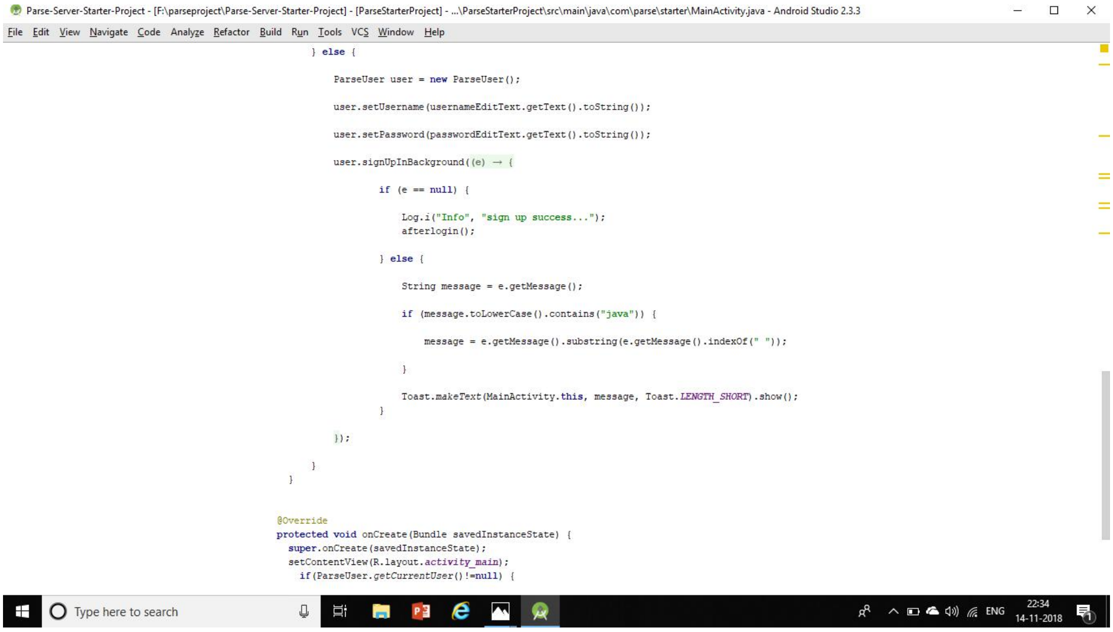
2. LOGIN/ SIGNUP ACTIVITY

User can either login and sign-up by using their username and password. The entered username is validated using AWS security services.


Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\MainActivity.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:34
14-11-2018

```java
public void signupLogin(View view) {

    final EditText usernameEditText = (EditText) findViewById(R.id.e1);

    final EditText passwordEditText = (EditText) findViewById(R.id.e2);

    if (loginModeActive) {

        ParseUser.logInInBackground(usernameEditText.getText().toString(), passwordEditText.getText().toString(), (user, e) → {

                if (e == null) {

                    Toast.makeText(MainActivity.this, "Loggedin"+ParseUser.getCurrentUser().getUsername(), Toast.LENGTH_SHORT).show();
                      afterlogin();
                } else {

                    String message = e.getMessage();

                    if (message.toLowerCase().contains("java")) {

                        message = e.getMessage().substring(e.getMessage().indexOf(" "));

                    }
                    usernameEditText.setError(""+message);
                    passwordEditText.setError(""+message);



                }

        });

    } else {

        ParseUser user = new ParseUser();

        user.setUsername(usernameEditText.getText().toString());
```

Fig. 14

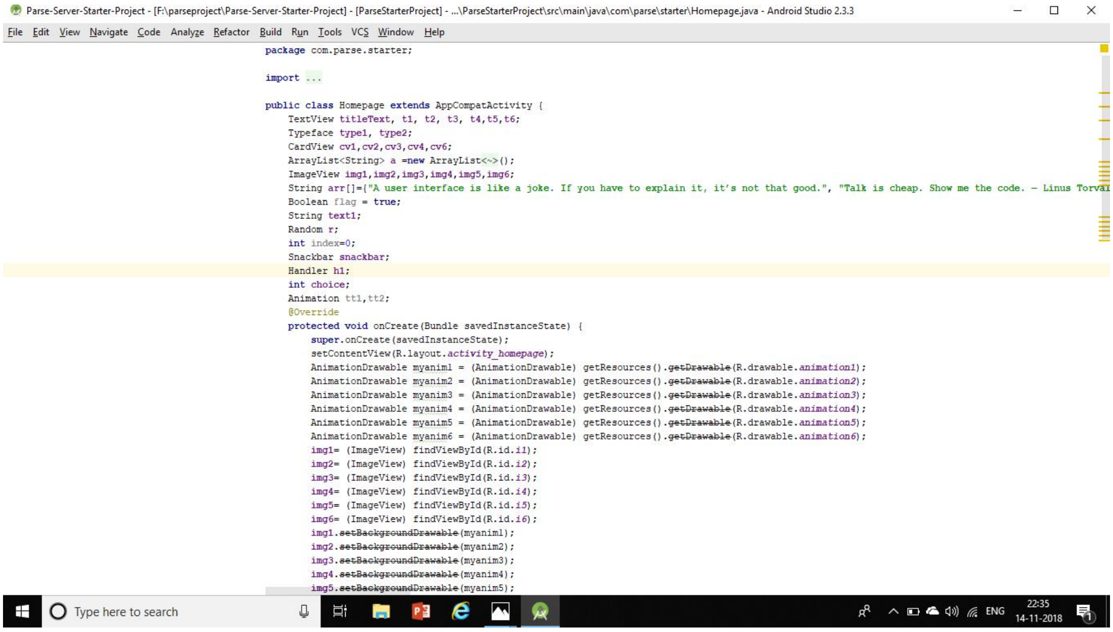

Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\MainActivity.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:34
14-11-2018

```java
      } else {

          ParseUser user = new ParseUser();

          user.setUsername(usernameEditText.getText().toString());

          user.setPassword(passwordEditText.getText().toString());

          user.signUpInBackground((e) → {

                  if (e == null) {

                      Log.i("Info", "sign up success...");
                      afterlogin();

                  } else {

                      String message = e.getMessage();

                      if (message.toLowerCase().contains("java")) {

                          message = e.getMessage().substring(e.getMessage().indexOf(" "));

                      }

                      Toast.makeText(MainActivity.this, message, Toast.LENGTH_SHORT).show();
                  }

          });

      }
  }


@Override
protected void onCreate(Bundle savedInstanceState) {
  super.onCreate(savedInstanceState);
  setContentView(R.layout.activity_main);
    if(ParseUser.getCurrentUser()!=null) {
```

Fig. 15

3. HOMEPAGE

This activity contains a set of card-views which intent us towards different modules of the application.



Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\Homepage.java - Android Studio 2.3.3

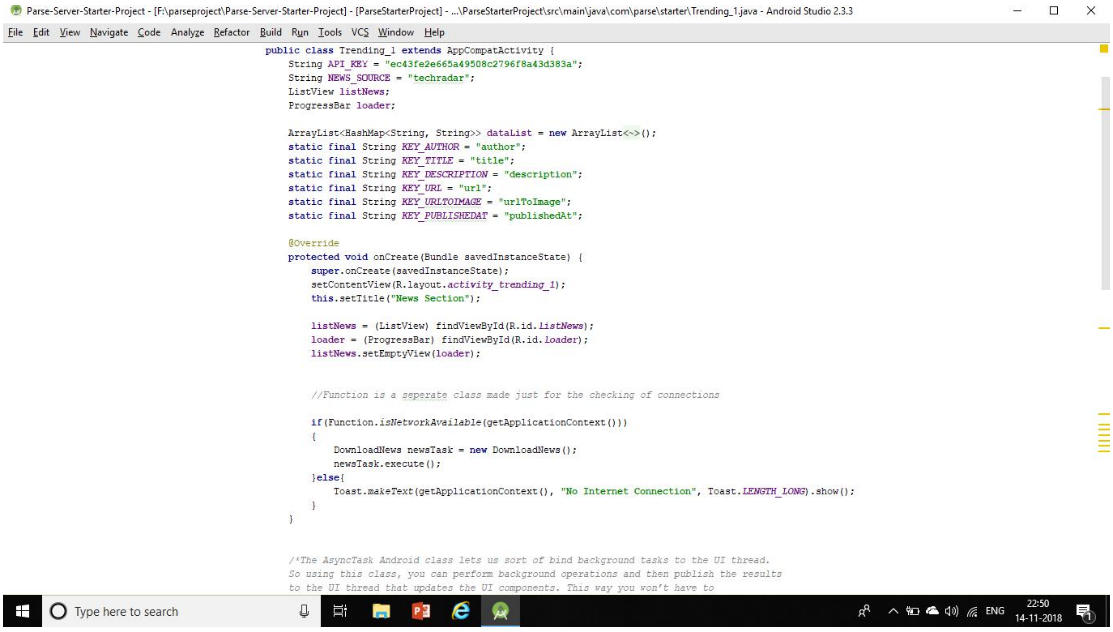
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:35
14-11-2018

```
package com.parse.starter;

import ...

public class Homepage extends AppCompatActivity {
    TextView titleText, t1, t2, t3, t4,t5,t6;
    Typeface type1, type2;
    CardView cv1,cv2,cv3,cv4,cv6;
    ArrayList<String> a =new ArrayList<~>();
    ImageView img1,img2,img3,img4,img5,img6;
    String arr[]={"A user interface is like a joke. If you have to explain it, it’s not that good.", "Talk is cheap. Show me the code. – Linus Torval
    Boolean flag = true;
    String text1;
    Random r;
    int index=0;
    Snackbar snackbar;
    Handler h1;
    int choice;
    Animation tt1,tt2;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_homepage);
        AnimationDrawable myaniml = (AnimationDrawable) getResources().~~getDrawable~~(R.drawable.animation1);
        AnimationDrawable myanim2 = (AnimationDrawable) getResources().~~getDrawable~~(R.drawable.animation2);
        AnimationDrawable myanim3 = (AnimationDrawable) getResources().~~getDrawable~~(R.drawable.animation3);
        AnimationDrawable myanim4 = (AnimationDrawable) getResources().~~getDrawable~~(R.drawable.animation4);
        AnimationDrawable myanim5 = (AnimationDrawable) getResources().~~getDrawable~~(R.drawable.animation5);
        AnimationDrawable myanim6 = (AnimationDrawable) getResources().~~getDrawable~~(R.drawable.animation6);
        img1= (ImageView) findViewById(R.id.i1);
        img2= (ImageView) findViewById(R.id.i2);
        img3= (ImageView) findViewById(R.id.i3);
        img4= (ImageView) findViewById(R.id.i4);
        img5= (ImageView) findViewById(R.id.i5);
        img6= (ImageView) findViewById(R.id.i6);
        img1.~~setBackgroundDrawable~~(myaniml);
        img2.~~setBackgroundDrawable~~(myanim2);
        img3.~~setBackgroundDrawable~~(myanim3);
        img4.~~setBackgroundDrawable~~(myanim4);
        img5.~~setBackgroundDrawable~~(myanim5);
```

Fig. 16

4. TECH TRENDS

In this activity the latest technology trends are retrieved through a unique API key and shown in a list.

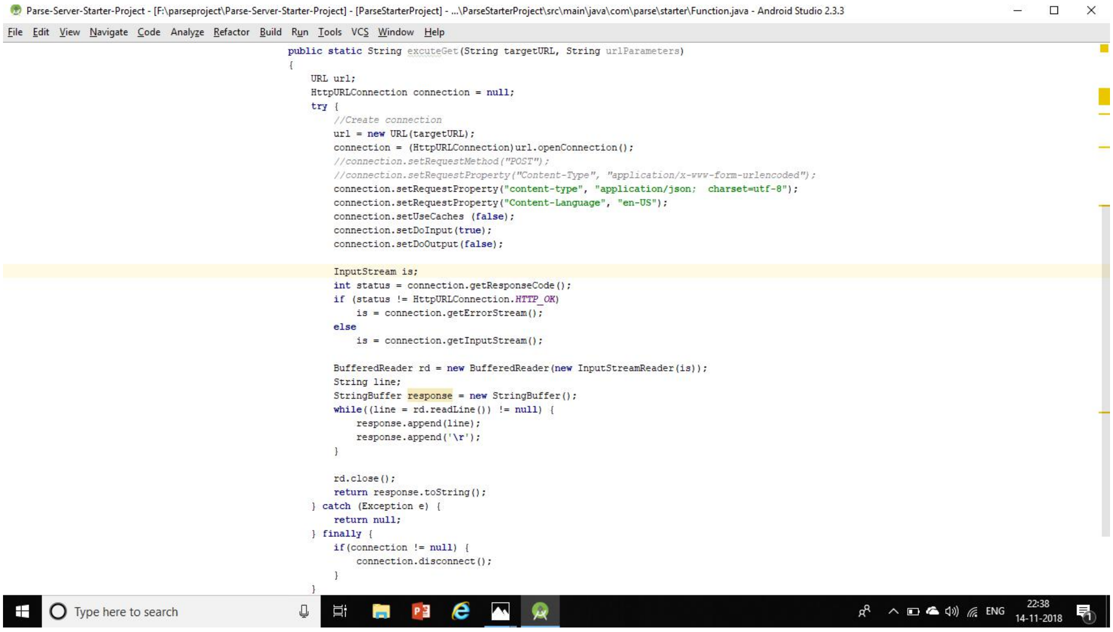


Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\Trending_1.java - Android Studio 2.3.3

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:50
14-11-2018

```
public class Trending_1 extends AppCompatActivity {
    String API_KEY = "ec43fe2e665a49508c2796f8a43d383a";
    String NEWS_SOURCE = "techradar";
    ListView listNews;
    ProgressBar loader;

    ArrayList<HashMap<String, String>> dataList = new ArrayList<~>();
    static final String KEY_AUTHOR = "author";
    static final String KEY_TITLE = "title";
    static final String KEY_DESCRIPTION = "description";
    static final String KEY_URL = "url";
    static final String KEY_URLTOIMAGE = "urlToImage";
    static final String KEY_PUBLISHEDAT = "publishedAt";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_trending_1);
        this.setTitle("News Section");

        listNews = (ListView) findViewById(R.id.listNews);
        loader = (ProgressBar) findViewById(R.id.loader);
        listNews.setEmptyView(loader);


        //Function is a seperate class made just for the checking of connections

        if(Function.isNetworkAvailable(getApplicationContext()))
        {
            DownloadNews newsTask = new DownloadNews();
            newsTask.execute();
        }else{
            Toast.makeText(getApplicationContext(), "No Internet Connection", Toast.LENGTH_LONG).show();
        }
    }


    /*The AsyncTask Android class lets us sort of bind background tasks to the UI thread.
    So using this class, you can perform background operations and then publish the results
    to the UI thread that updates the UI components. This way you won't have to
```

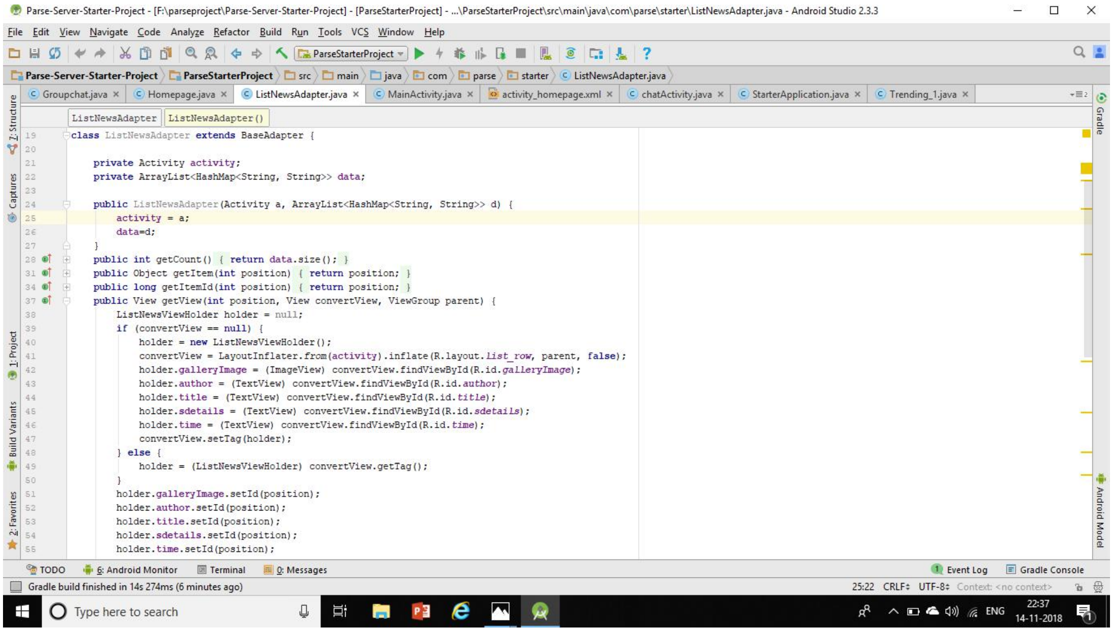
Fig. 17



Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\Function.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:38
14-11-2018

```java
public static String excuteGet(String targetURL, String urlParameters)
{
    URL url;
    HttpURLConnection connection = null;
    try {
        //Create connection
        url = new URL(targetURL);
        connection = (HttpURLConnection)url.openConnection();
        //connection.setRequestMethod("POST");
        //connection.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");
        connection.setRequestProperty("content-type", "application/json;  charset=utf-8");
        connection.setRequestProperty("Content-Language", "en-US");
        connection.setUseCaches (false);
        connection.setDoInput(true);
        connection.setDoOutput(false);

        InputStream is;
        int status = connection.getResponseCode();
        if (status != HttpURLConnection.HTTP_OK)
            is = connection.getErrorStream();
        else
            is = connection.getInputStream();

        BufferedReader rd = new BufferedReader(new InputStreamReader(is));
        String line;
        StringBuffer response = new StringBuffer();
        while((line = rd.readLine()) != null) {
            response.append(line);
            response.append('\r');
        }

        rd.close();
        return response.toString();
    } catch (Exception e) {
        return null;
    } finally {
        if(connection != null) {
            connection.disconnect();
        }
    }
```

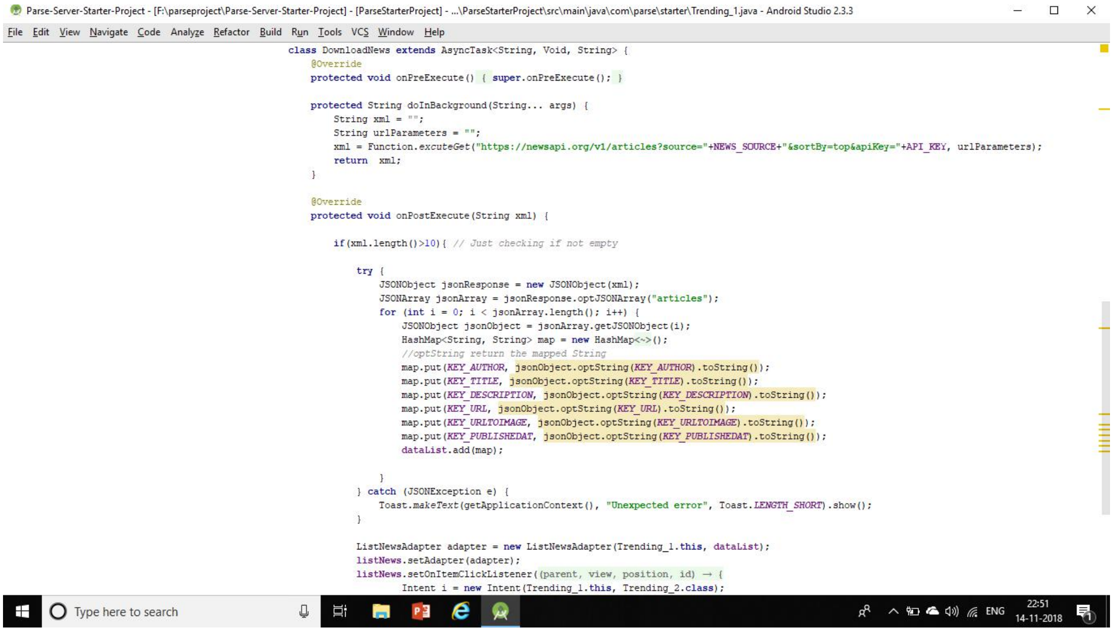
Fig. 18



```java
class ListNewsAdapter extends BaseAdapter {

    private Activity activity;
    private ArrayList<HashMap<String, String>> data;

    public ListNewsAdapter(Activity a, ArrayList<HashMap<String, String>> d) {
        activity = a;
        data=d;
    }
    public int getCount() { return data.size(); }
    public Object getItem(int position) { return position; }
    public long getItemId(int position) { return position; }
    public View getView(int position, View convertView, ViewGroup parent) {
        ListNewsViewHolder holder = null;
        if (convertView == null) {
            holder = new ListNewsViewHolder();
            convertView = LayoutInflater.from(activity).inflate(R.layout.list_row, parent, false);
            holder.galleryImage = (ImageView) convertView.findViewById(R.id.galleryImage);
            holder.author = (TextView) convertView.findViewById(R.id.author);
            holder.title = (TextView) convertView.findViewById(R.id.title);
            holder.sdetails = (TextView) convertView.findViewById(R.id.sdetails);
            holder.time = (TextView) convertView.findViewById(R.id.time);
            convertView.setTag(holder);
        } else {
            holder = (ListNewsViewHolder) convertView.getTag();
        }
        holder.galleryImage.setId(position);
        holder.author.setId(position);
        holder.title.setId(position);
        holder.sdetails.setId(position);
        holder.time.setId(position);
```

Gradle build finished in 14s 274ms (6 minutes ago)

Fig. 19

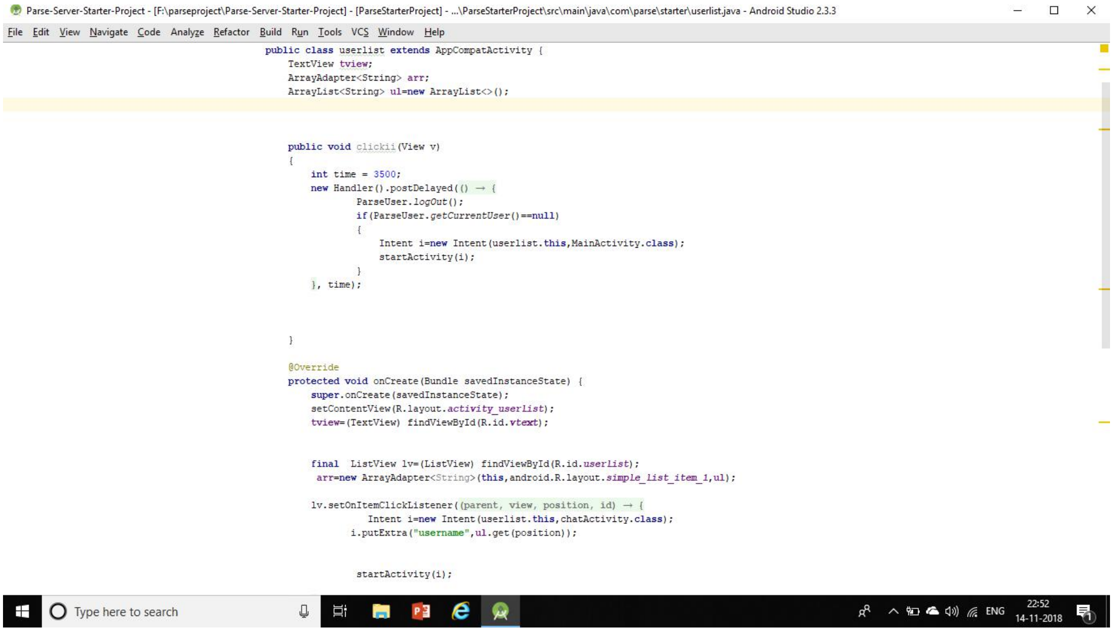

Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\Trending_1.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:51
14-11-2018

```java
class DownloadNews extends AsyncTask<String, Void, String> {
    @Override
    protected void onPreExecute() { super.onPreExecute(); }

    protected String doInBackground(String... args) {
        String xml = "";
        String urlParameters = "";
        xml = Function.excuteGet("https://newsapi.org/v1/articles?source="+NEWS_SOURCE+"&sortBy=top&apiKey="+API_KEY, urlParameters);
        return  xml;
    }

    @Override
    protected void onPostExecute(String xml) {

        if(xml.length()>10){ // Just checking if not empty

            try {
                JSONObject jsonResponse = new JSONObject(xml);
                JSONArray jsonArray = jsonResponse.optJSONArray("articles");
                for (int i = 0; i < jsonArray.length(); i++) {
                    JSONObject jsonObject = jsonArray.getJSONObject(i);
                    HashMap<String, String> map = new HashMap<~>();
                    //optString return the mapped String
                    map.put(KEY_AUTHOR, jsonObject.optString(KEY_AUTHOR).toString());
                    map.put(KEY_TITLE, jsonObject.optString(KEY_TITLE).toString());
                    map.put(KEY_DESCRIPTION, jsonObject.optString(KEY_DESCRIPTION).toString());
                    map.put(KEY_URL, jsonObject.optString(KEY_URL).toString());
                    map.put(KEY_URLTOIMAGE, jsonObject.optString(KEY_URLTOIMAGE).toString());
                    map.put(KEY_PUBLISHEDAT, jsonObject.optString(KEY_PUBLISHEDAT).toString());
                    dataList.add(map);

                }
            } catch (JSONException e) {
                Toast.makeText(getApplicationContext(), "Unexpected error", Toast.LENGTH_SHORT).show();
            }

            ListNewsAdapter adapter = new ListNewsAdapter(Trending_1.this, dataList);
            listNews.setAdapter(adapter);
            listNews.setOnItemClickListener((parent, view, position, id) → {
                    Intent i = new Intent(Trending_1.this, Trending_2.class);
```

Fig. 20

5. USER LIST

All the registered users of the application are shown through this activity.

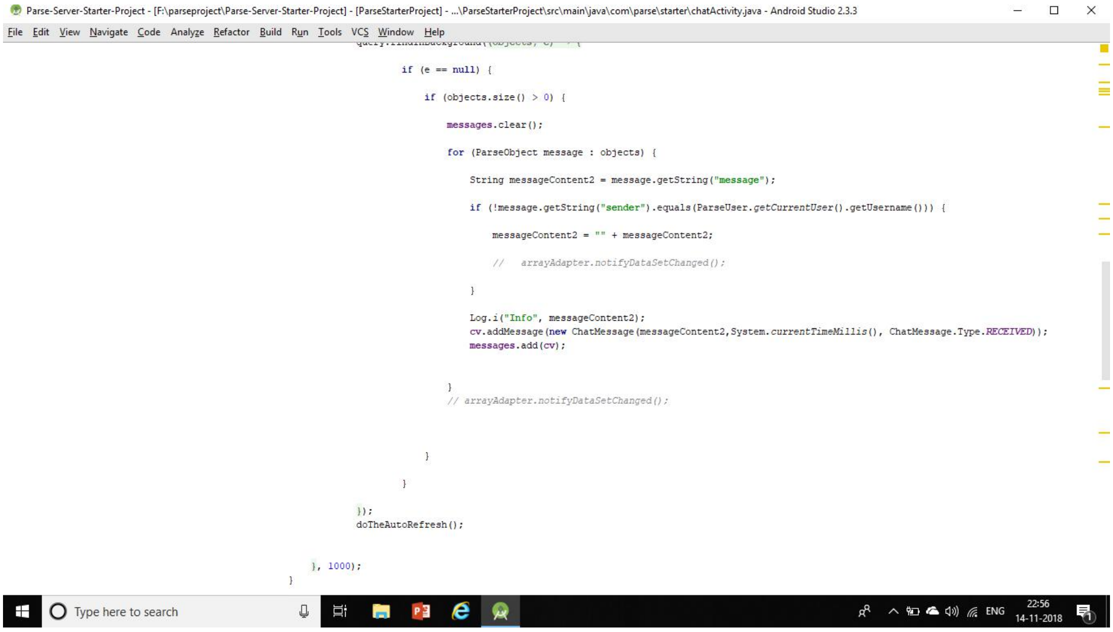

Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\userlist.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:52
14-11-2018

```java
public class userlist extends AppCompatActivity {
    TextView tview;
    ArrayAdapter<String> arr;
    ArrayList<String> ul=new ArrayList<>();



    public void clickii(View v)
    {
        int time = 3500;
        new Handler().postDelayed(() → {
                ParseUser.logOut();
                if(ParseUser.getCurrentUser()==null)
                {
                    Intent i=new Intent(userlist.this,MainActivity.class);
                    startActivity(i);
                }
        }, time);



    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_userlist);
        tview=(TextView) findViewById(R.id.vtext);


        final  ListView lv=(ListView) findViewById(R.id.userlist);
         arr=new ArrayAdapter<String>(this,android.R.layout.simple_list_item_1,ul);

        lv.setOnItemClickListener((parent, view, position, id) → {
                  Intent i=new Intent(userlist.this,chatActivity.class);
               i.putExtra("username",ul.get(position));


                startActivity(i);
```

Fig. 21

6. INDIVIDUAL CHATBOX

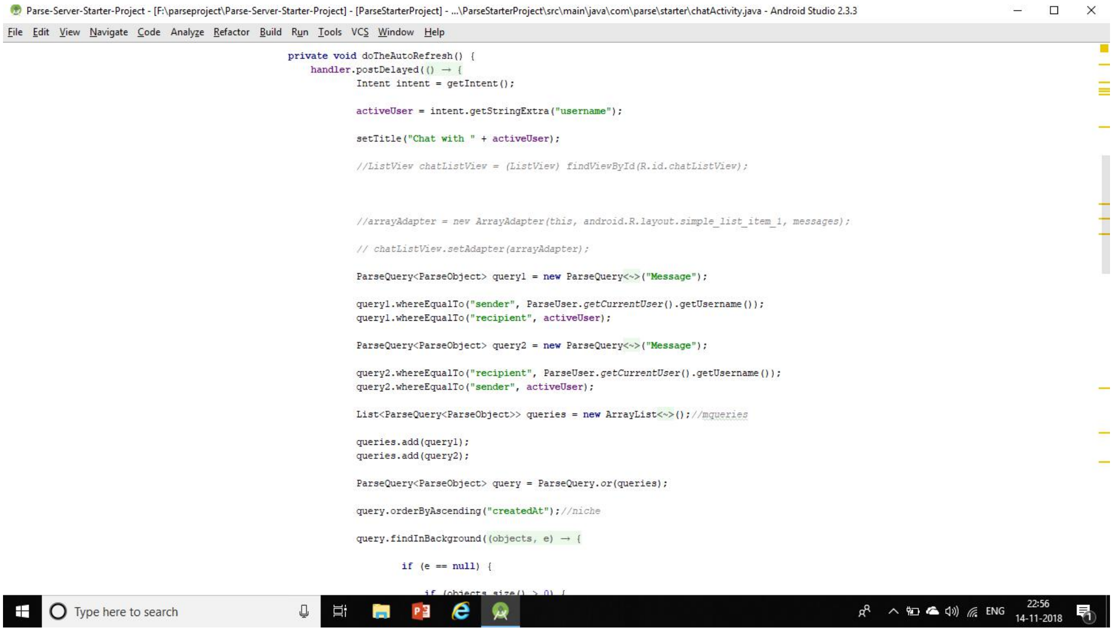
User can chat with any other registered user through this activity.



Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\chatActivity.java - Android Studio 2.3.3

```
            query.findInBackground((objects, e) → {

                    if (e == null) {

                        if (objects.size() > 0) {

                            messages.clear();

                            for (ParseObject message : objects) {

                                String messageContent2 = message.getString("message");

                                if (!message.getString("sender").equals(ParseUser.getCurrentUser().getUsername())) {

                                    messageContent2 = "" + messageContent2;

                                    //   arrayAdapter.notifyDataSetChanged();

                                }

                                Log.i("Info", messageContent2);
                                cv.addMessage(new ChatMessage(messageContent2,System.currentTimeMillis(), ChatMessage.Type.RECEIVED));
                                messages.add(cv);


                            }
                            // arrayAdapter.notifyDataSetChanged();



                        }

                    }

            });
            doTheAutoRefresh();


    }, 1000);
}
```

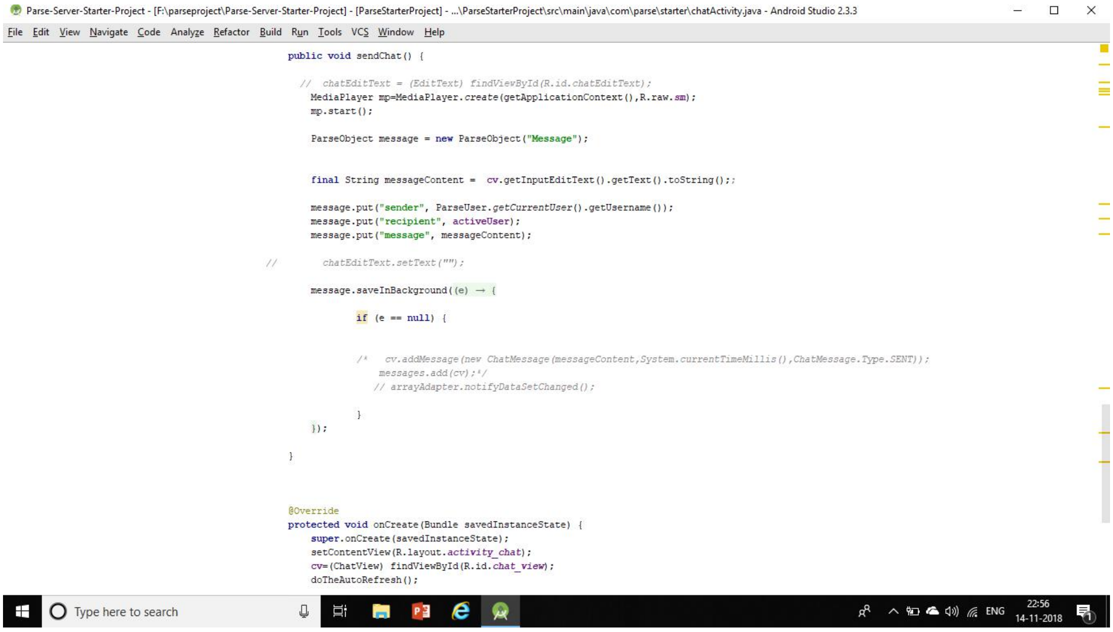
Fig. 22



Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\chatActivity.java - Android Studio 2.3.3

```
private void doTheAutoRefresh() {
    handler.postDelayed(() → {
            Intent intent = getIntent();

            activeUser = intent.getStringExtra("username");

            setTitle("Chat with " + activeUser);

            //ListView chatListView = (ListView) findViewById(R.id.chatListView);



            //arrayAdapter = new ArrayAdapter(this, android.R.layout.simple_list_item_1, messages);

            // chatListView.setAdapter(arrayAdapter);

            ParseQuery<ParseObject> query1 = new ParseQuery<~>("Message");

            query1.whereEqualTo("sender", ParseUser.getCurrentUser().getUsername());
            query1.whereEqualTo("recipient", activeUser);

            ParseQuery<ParseObject> query2 = new ParseQuery<~>("Message");

            query2.whereEqualTo("recipient", ParseUser.getCurrentUser().getUsername());
            query2.whereEqualTo("sender", activeUser);

            List<ParseQuery<ParseObject>> queries = new ArrayList<~>();//mqueries

            queries.add(query1);
            queries.add(query2);

            ParseQuery<ParseObject> query = ParseQuery.or(queries);

            query.orderByAscending("createdAt");//niche

            query.findInBackground((objects, e) → {

                    if (e == null) {

                        if (objects.size() > 0) {
```

Fig. 23

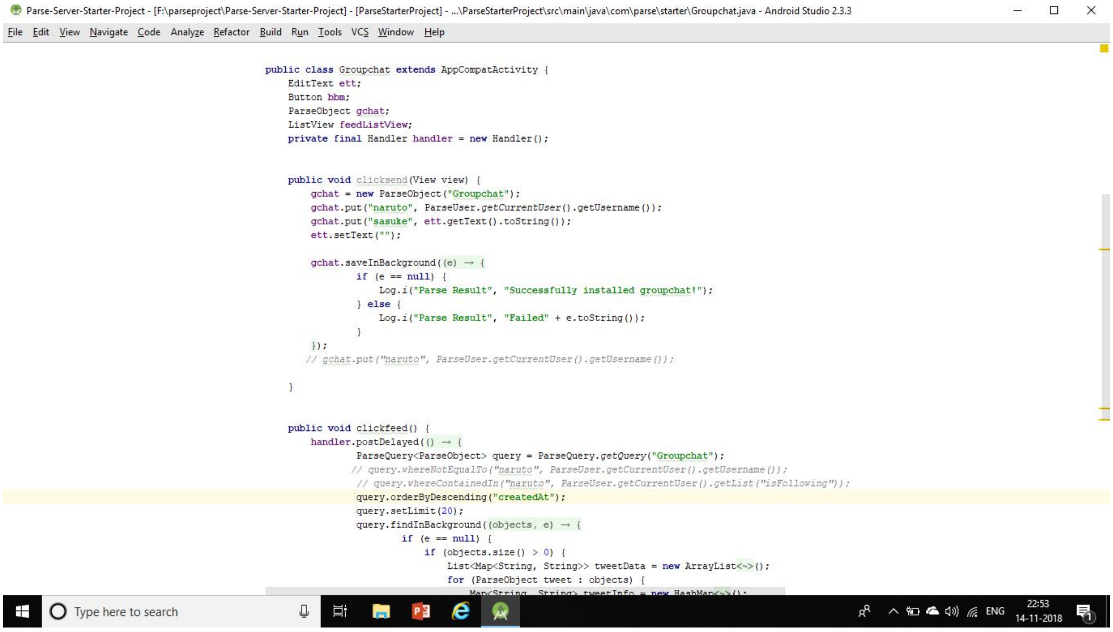

Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\chatActivity.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:56
14-11-2018

```java
    public void sendChat() {

      //  chatEditText = (EditText) findViewById(R.id.chatEditText);
        MediaPlayer mp=MediaPlayer.create(getApplicationContext(),R.raw.sm);
        mp.start();

        ParseObject message = new ParseObject("Message");


        final String messageContent =  cv.getInputEditText().getText().toString();;

        message.put("sender", ParseUser.getCurrentUser().getUsername());
        message.put("recipient", activeUser);
        message.put("message", messageContent);

//          chatEditText.setText("");

        message.saveInBackground((e) → {

                if (e == null) {


                /*   cv.addMessage(new ChatMessage(messageContent,System.currentTimeMillis(),ChatMessage.Type.SENT));
                    messages.add(cv);*/
                   // arrayAdapter.notifyDataSetChanged();

                }
        });

    }



    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_chat);
        cv=(ChatView) findViewById(R.id.chat_view);
        doTheAutoRefresh();
```

Fig. 24

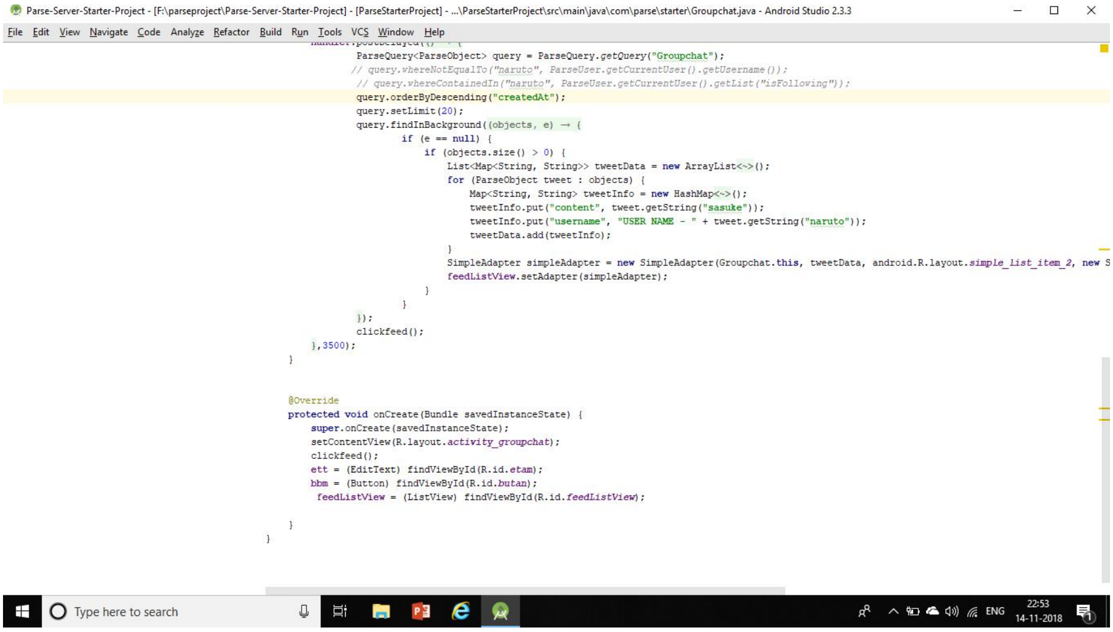
7. GLOBAL CHATBOX

This is the activity where all the users of the application are able to collectively chat with each other.


Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\Groupchat.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:53
14-11-2018

```java
public class Groupchat extends AppCompatActivity {
    EditText ett;
    Button bbm;
    ParseObject gchat;
    ListView feedListView;
    private final Handler handler = new Handler();


    public void clicksend(View view) {
        gchat = new ParseObject("Groupchat");
        gchat.put("naruto", ParseUser.getCurrentUser().getUsername());
        gchat.put("sasuke", ett.getText().toString());
        ett.setText("");

        gchat.saveInBackground((e) → {
                if (e == null) {
                    Log.i("Parse Result", "Successfully installed groupchat!");
                } else {
                    Log.i("Parse Result", "Failed" + e.toString());
                }
        });
       // gchat.put("naruto", ParseUser.getCurrentUser().getUsername());

    }


    public void clickfeed() {
        handler.postDelayed(() → {
                ParseQuery<ParseObject> query = ParseQuery.getQuery("Groupchat");
               // query.whereNotEqualTo("naruto", ParseUser.getCurrentUser().getUsername());
                // query.whereContainedIn("naruto", ParseUser.getCurrentUser().getList("isFollowing"));
                query.orderByDescending("createdAt");
                query.setLimit(20);
                query.findInBackground((objects, e) → {
                        if (e == null) {
                            if (objects.size() > 0) {
                                List<Map<String, String>> tweetData = new ArrayList<~>();
                                for (ParseObject tweet : objects) {
                                    Map<String, String> tweetInfo = new HashMap<~>();
```

Fig. 25

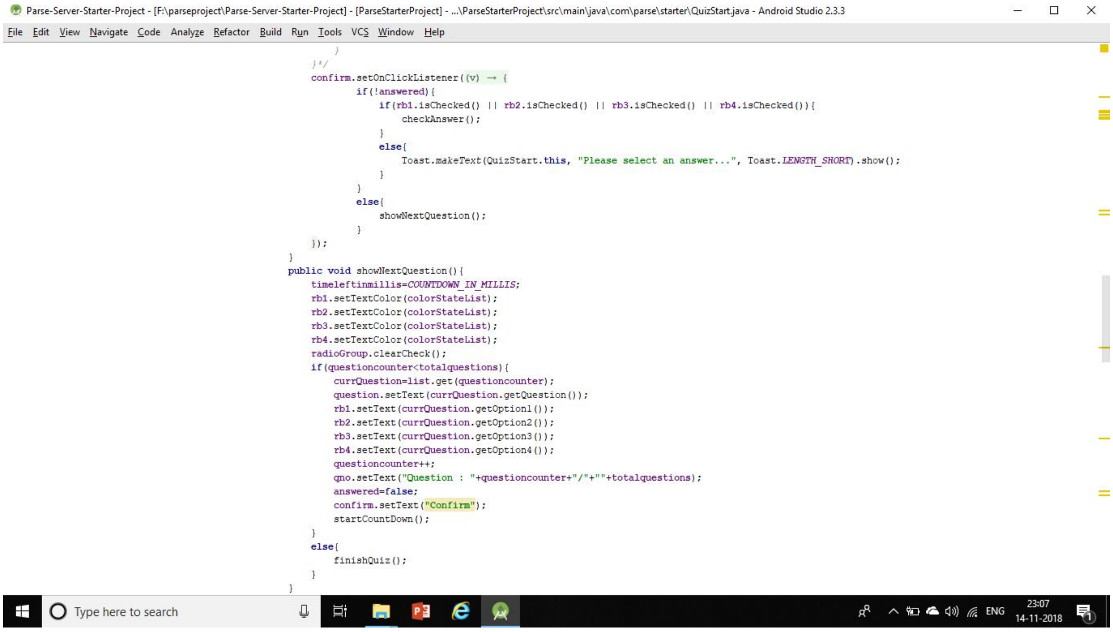

Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\Groupchat.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:53
14-11-2018

```java
        handler.postDelayed(() → {
                ParseQuery<ParseObject> query = ParseQuery.getQuery("Groupchat");
               // query.whereNotEqualTo("naruto", ParseUser.getCurrentUser().getUsername());
                // query.whereContainedIn("naruto", ParseUser.getCurrentUser().getList("isFollowing"));
                query.orderByDescending("createdAt");
                query.setLimit(20);
                query.findInBackground((objects, e) → {
                        if (e == null) {
                            if (objects.size() > 0) {
                                List<Map<String, String>> tweetData = new ArrayList<~>();
                                for (ParseObject tweet : objects) {
                                    Map<String, String> tweetInfo = new HashMap<~>();
                                    tweetInfo.put("content", tweet.getString("sasuke"));
                                    tweetInfo.put("username", "USER NAME - " + tweet.getString("naruto"));
                                    tweetData.add(tweetInfo);
                                }
                                SimpleAdapter simpleAdapter = new SimpleAdapter(Groupchat.this, tweetData, android.R.layout.simple_list_item_2, new S
                                feedListView.setAdapter(simpleAdapter);
                            }
                        }
                });
                clickfeed();
        },3500);
    }


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_groupchat);
        clickfeed();
        ett = (EditText) findViewById(R.id.etam);
        bbm = (Button) findViewById(R.id.butan);
         feedListView = (ListView) findViewById(R.id.feedListView);

    }
}
```

Fig. 26

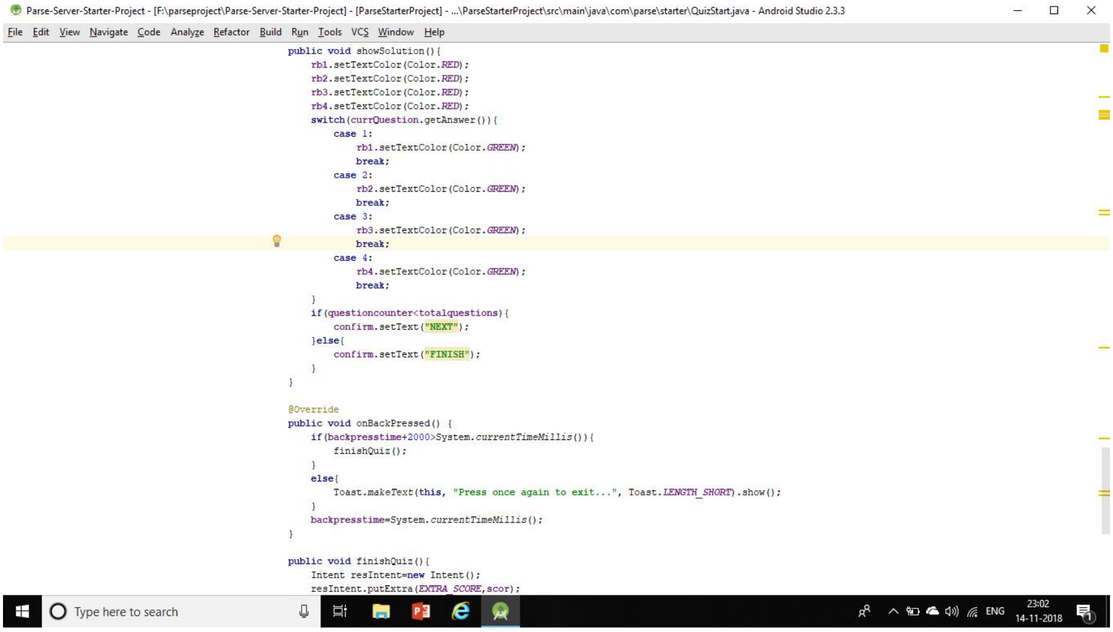
8. QUICK QUIZ

This activity provides the user with the functionality of testing his knowledge with participating in quizzes based on different programming languages selecting the difficulty level of their choice.


Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\QuizStart.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
23:07
14-11-2018

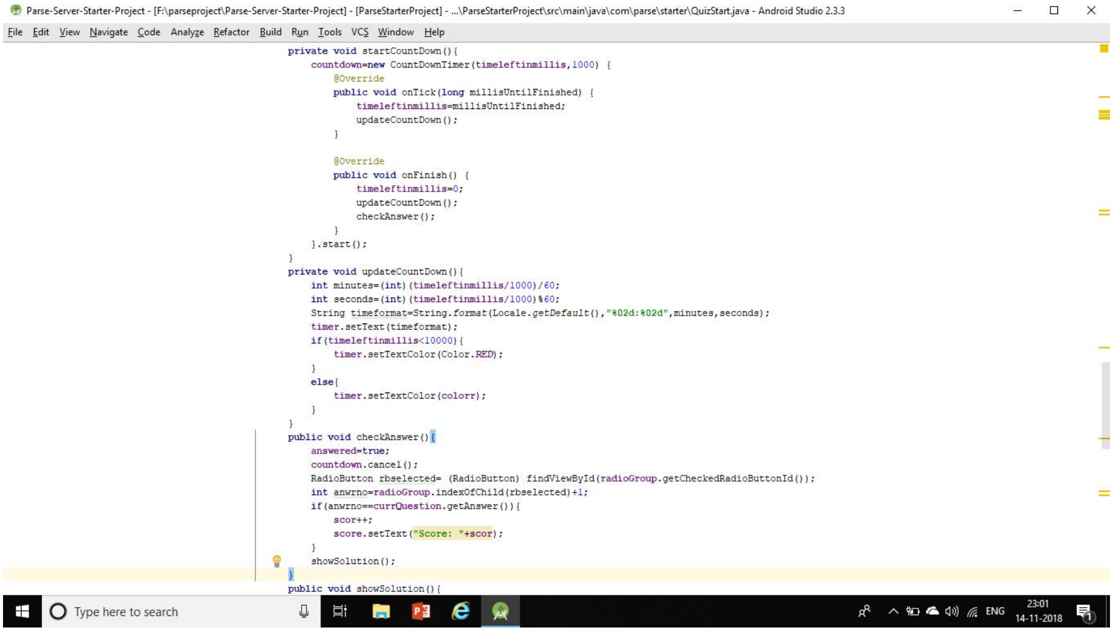
```java
            }
        }*/
        confirm.setOnClickListener((v) → {
                if(!answered){
                    if(rb1.isChecked() || rb2.isChecked() || rb3.isChecked() || rb4.isChecked()){
                        checkAnswer();
                    }
                    else{
                        Toast.makeText(QuizStart.this, "Please select an answer...", Toast.LENGTH_SHORT).show();
                    }
                }
                else{
                    showNextQuestion();
                }
        });
    }
    public void showNextQuestion(){
        timeleftinmillis=COUNTDOWN_IN_MILLIS;
        rb1.setTextColor(colorStateList);
        rb2.setTextColor(colorStateList);
        rb3.setTextColor(colorStateList);
        rb4.setTextColor(colorStateList);
        radioGroup.clearCheck();
        if(questioncounter<totalquestions){
            currQuestion=list.get(questioncounter);
            question.setText(currQuestion.getQuestion());
            rb1.setText(currQuestion.getOption1());
            rb2.setText(currQuestion.getOption2());
            rb3.setText(currQuestion.getOption3());
            rb4.setText(currQuestion.getOption4());
            questioncounter++;
            qno.setText("Question : "+questioncounter+"/"+""+totalquestions);
            answered=false;
            confirm.setText("Confirm");
            startCountDown();
        }
        else{
            finishQuiz();
        }
    }
```

Fig. 27


Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\QuizStart.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
23:02
14-11-2018

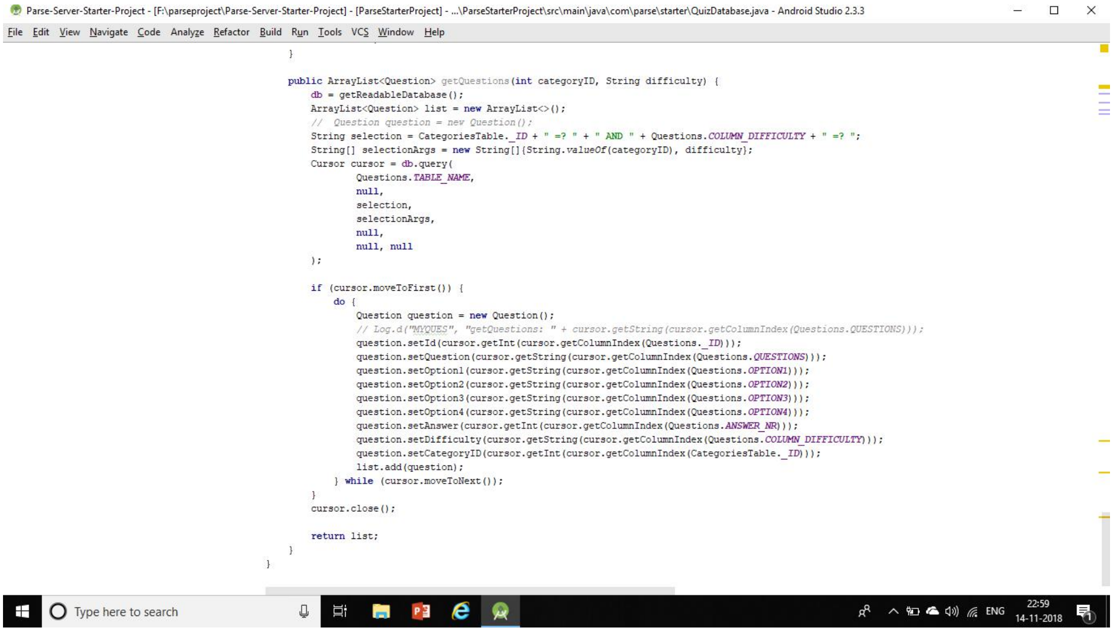
```
public void showSolution(){
    rb1.setTextColor(Color.RED);
    rb2.setTextColor(Color.RED);
    rb3.setTextColor(Color.RED);
    rb4.setTextColor(Color.RED);
    switch(currQuestion.getAnswer()){
        case 1:
            rb1.setTextColor(Color.GREEN);
            break;
        case 2:
            rb2.setTextColor(Color.GREEN);
            break;
        case 3:
            rb3.setTextColor(Color.GREEN);
            break;
        case 4:
            rb4.setTextColor(Color.GREEN);
            break;
    }
    if(questioncounter<totalquestions){
        confirm.setText("NEXT");
    }else{
        confirm.setText("FINISH");
    }
}

@Override
public void onBackPressed() {
    if(backpresstime+2000>System.currentTimeMillis()){
        finishQuiz();
    }
    else{
        Toast.makeText(this, "Press once again to exit...", Toast.LENGTH_SHORT).show();
    }
    backpresstime=System.currentTimeMillis();
}

public void finishQuiz(){
    Intent resIntent=new Intent();
    resIntent.putExtra(EXTRA_SCORE,scor);
```

Fig. 28


Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\QuizStart.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
23:01
14-11-2018

```
private void startCountDown(){
    countdown=new CountDownTimer(timeleftinmillis,1000) {
        @Override
        public void onTick(long millisUntilFinished) {
            timeleftinmillis=millisUntilFinished;
            updateCountDown();
        }

        @Override
        public void onFinish() {
            timeleftinmillis=0;
            updateCountDown();
            checkAnswer();
        }
    }.start();
}
private void updateCountDown(){
    int minutes=(int)(timeleftinmillis/1000)/60;
    int seconds=(int)(timeleftinmillis/1000)%60;
    String timeformat=String.format(Locale.getDefault(),"%02d:%02d",minutes,seconds);
    timer.setText(timeformat);
    if(timeleftinmillis<10000){
        timer.setTextColor(Color.RED);
    }
    else{
        timer.setTextColor(colorr);
    }
}
public void checkAnswer(){
    answered=true;
    countdown.cancel();
    RadioButton rbselected= (RadioButton) findViewById(radioGroup.getCheckedRadioButtonId());
    int anwrno=radioGroup.indexOfChild(rbselected)+1;
    if(anwrno==currQuestion.getAnswer()){
        scor++;
        score.setText("Score: "+scor);
    }
    showSolution();
}
public void showSolution(){
```

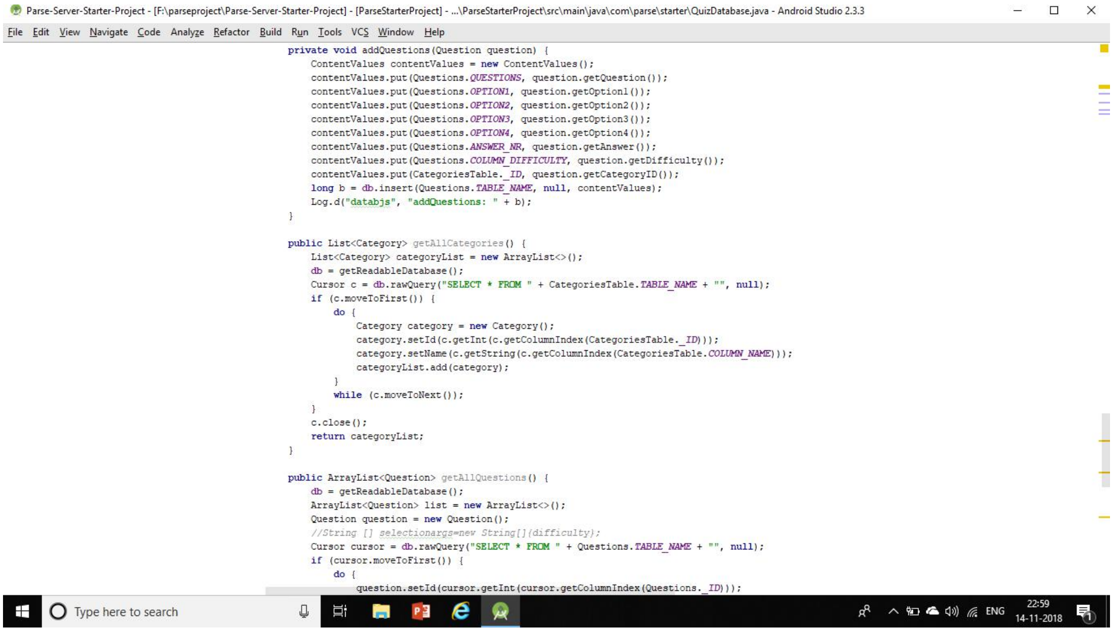
Fig. 29



Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\QuizDatabase.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:59
14-11-2018

```java
    }

    public ArrayList<Question> getQuestions(int categoryID, String difficulty) {
        db = getReadableDatabase();
        ArrayList<Question> list = new ArrayList<>();
        //  Question question = new Question();
        String selection = CategoriesTable._ID + " =? " + " AND " + Questions.COLUMN_DIFFICULTY + " =? ";
        String[] selectionArgs = new String[]{String.valueOf(categoryID), difficulty};
        Cursor cursor = db.query(
                Questions.TABLE_NAME,
                null,
                selection,
                selectionArgs,
                null,
                null, null
        );

        if (cursor.moveToFirst()) {
            do {
                Question question = new Question();
                // Log.d("MYQUES", "getQuestions: " + cursor.getString(cursor.getColumnIndex(Questions.QUESTIONS)));
                question.setId(cursor.getInt(cursor.getColumnIndex(Questions._ID)));
                question.setQuestion(cursor.getString(cursor.getColumnIndex(Questions.QUESTIONS)));
                question.setOption1(cursor.getString(cursor.getColumnIndex(Questions.OPTION1)));
                question.setOption2(cursor.getString(cursor.getColumnIndex(Questions.OPTION2)));
                question.setOption3(cursor.getString(cursor.getColumnIndex(Questions.OPTION3)));
                question.setOption4(cursor.getString(cursor.getColumnIndex(Questions.OPTION4)));
                question.setAnswer(cursor.getInt(cursor.getColumnIndex(Questions.ANSWER_NR)));
                question.setDifficulty(cursor.getString(cursor.getColumnIndex(Questions.COLUMN_DIFFICULTY)));
                question.setCategoryID(cursor.getInt(cursor.getColumnIndex(CategoriesTable._ID)));
                list.add(question);
            } while (cursor.moveToNext());
        }
        cursor.close();

        return list;
    }
}
```

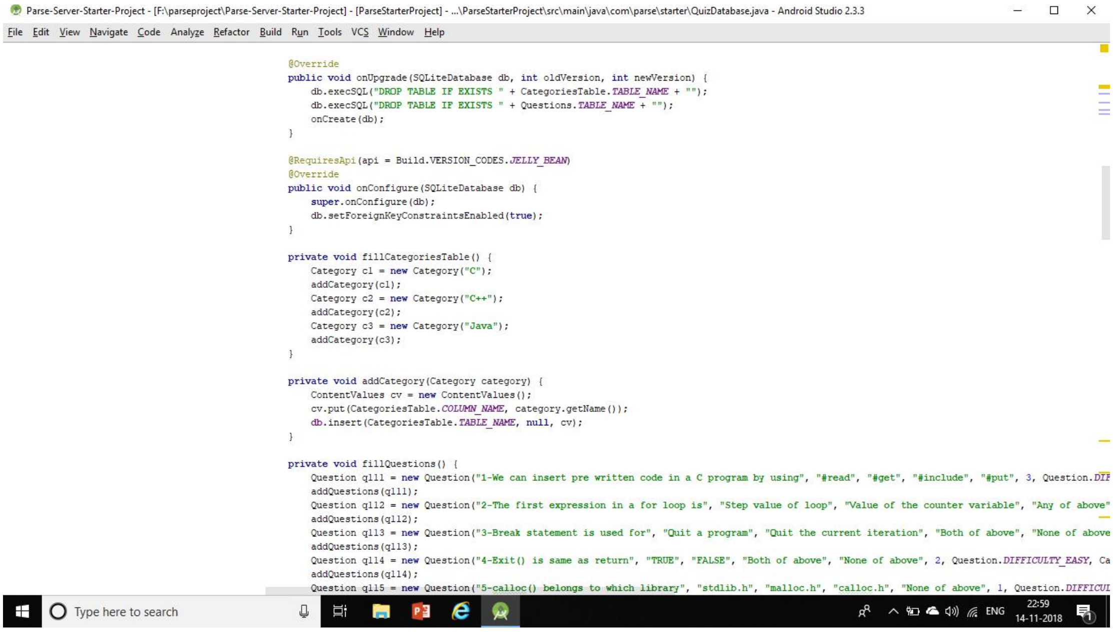
Fig. 30



Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\QuizDatabase.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:59
14-11-2018

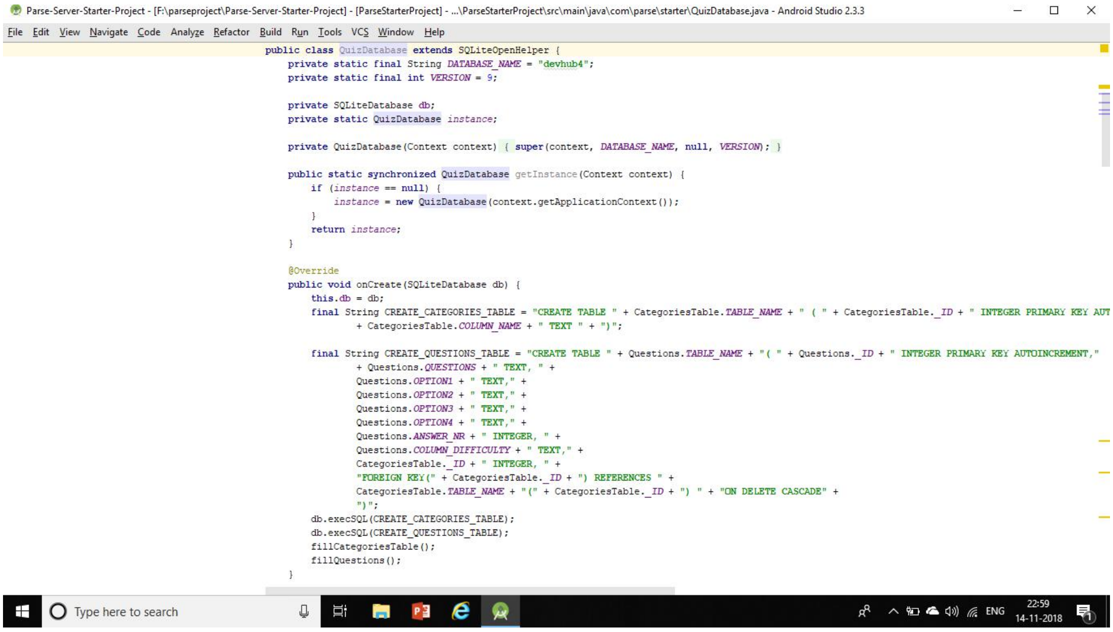
```java
    private void addQuestions(Question question) {
        ContentValues contentValues = new ContentValues();
        contentValues.put(Questions.QUESTIONS, question.getQuestion());
        contentValues.put(Questions.OPTION1, question.getOption1());
        contentValues.put(Questions.OPTION2, question.getOption2());
        contentValues.put(Questions.OPTION3, question.getOption3());
        contentValues.put(Questions.OPTION4, question.getOption4());
        contentValues.put(Questions.ANSWER_NR, question.getAnswer());
        contentValues.put(Questions.COLUMN_DIFFICULTY, question.getDifficulty());
        contentValues.put(CategoriesTable._ID, question.getCategoryID());
        long b = db.insert(Questions.TABLE_NAME, null, contentValues);
        Log.d("databjs", "addQuestions: " + b);
    }

    public List<Category> getAllCategories() {
        List<Category> categoryList = new ArrayList<>();
        db = getReadableDatabase();
        Cursor c = db.rawQuery("SELECT * FROM " + CategoriesTable.TABLE_NAME + "", null);
        if (c.moveToFirst()) {
            do {
                Category category = new Category();
                category.setId(c.getInt(c.getColumnIndex(CategoriesTable._ID)));
                category.setName(c.getString(c.getColumnIndex(CategoriesTable.COLUMN_NAME)));
                categoryList.add(category);
            }
            while (c.moveToNext());
        }
        c.close();
        return categoryList;
    }

    public ArrayList<Question> getAllQuestions() {
        db = getReadableDatabase();
        ArrayList<Question> list = new ArrayList<>();
        Question question = new Question();
        //String [] selectionargs=new String[]{difficulty};
        Cursor cursor = db.rawQuery("SELECT * FROM " + Questions.TABLE_NAME + "", null);
        if (cursor.moveToFirst()) {
            do {
                question.setId(cursor.getInt(cursor.getColumnIndex(Questions._ID)));
```

Fig. 31


Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\QuizDatabase.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:59
14-11-2018

```java
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + CategoriesTable.TABLE_NAME + "");
        db.execSQL("DROP TABLE IF EXISTS " + Questions.TABLE_NAME + "");
        onCreate(db);
    }

    @RequiresApi(api = Build.VERSION_CODES.JELLY_BEAN)
    @Override
    public void onConfigure(SQLiteDatabase db) {
        super.onConfigure(db);
        db.setForeignKeyConstraintsEnabled(true);
    }

    private void fillCategoriesTable() {
        Category c1 = new Category("C");
        addCategory(c1);
        Category c2 = new Category("C++");
        addCategory(c2);
        Category c3 = new Category("Java");
        addCategory(c3);
    }

    private void addCategory(Category category) {
        ContentValues cv = new ContentValues();
        cv.put(CategoriesTable.COLUMN_NAME, category.getName());
        db.insert(CategoriesTable.TABLE_NAME, null, cv);
    }

    private void fillQuestions() {
        Question q111 = new Question("1-We can insert pre written code in a C program by using", "#read", "#get", "#include", "#put", 3, Question.DIF
        addQuestions(q111);
        Question q112 = new Question("2-The first expression in a for loop is", "Step value of loop", "Value of the counter variable", "Any of above"
        addQuestions(q112);
        Question q113 = new Question("3-Break statement is used for", "Quit a program", "Quit the current iteration", "Both of above", "None of above
        addQuestions(q113);
        Question q114 = new Question("4-Exit() is same as return", "TRUE", "FALSE", "Both of above", "None of above", 2, Question.DIFFICULTY_EASY, Ca
        addQuestions(q114);
        Question q115 = new Question("5-calloc() belongs to which library", "stdlib.h", "malloc.h", "calloc.h", "None of above", 1, Question.DIFFICUL
```

Fig. 32

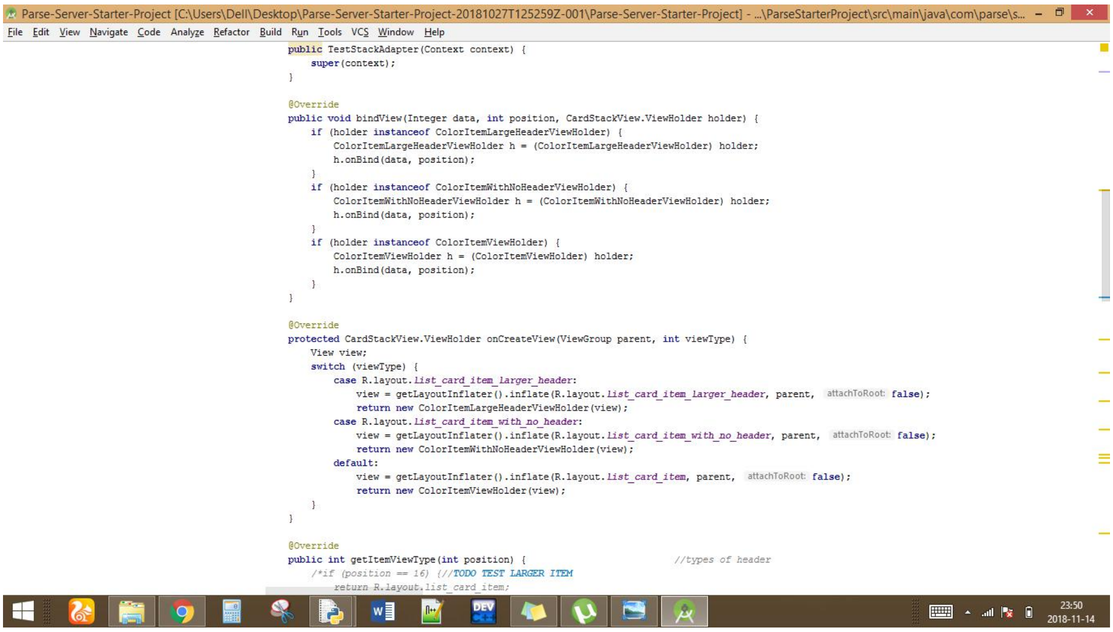

Parse-Server-Starter-Project - [F:\parseproject\Parse-Server-Starter-Project] - [ParseStarterProject] - ...\ParseStarterProject\src\main\java\com\parse\starter\QuizDatabase.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Type here to search
ENG
22:59
14-11-2018

```java
public class QuizDatabase extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "devhub4";
    private static final int VERSION = 9;

    private SQLiteDatabase db;
    private static QuizDatabase instance;

    private QuizDatabase(Context context) { super(context, DATABASE_NAME, null, VERSION); }

    public static synchronized QuizDatabase getInstance(Context context) {
        if (instance == null) {
            instance = new QuizDatabase(context.getApplicationContext());
        }
        return instance;
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        this.db = db;
        final String CREATE_CATEGORIES_TABLE = "CREATE TABLE " + CategoriesTable.TABLE_NAME + " ( " + CategoriesTable._ID + " INTEGER PRIMARY KEY AUT
                + CategoriesTable.COLUMN_NAME + " TEXT " + ")";

        final String CREATE_QUESTIONS_TABLE = "CREATE TABLE " + Questions.TABLE_NAME + "( " + Questions._ID + " INTEGER PRIMARY KEY AUTOINCREMENT,"
                + Questions.QUESTIONS + " TEXT, " +
                Questions.OPTION1 + " TEXT," +
                Questions.OPTION2 + " TEXT," +
                Questions.OPTION3 + " TEXT," +
                Questions.OPTION4 + " TEXT," +
                Questions.ANSWER_NR + " INTEGER, " +
                Questions.COLUMN_DIFFICULTY + " TEXT," +
                CategoriesTable._ID + " INTEGER, " +
                "FOREIGN KEY(" + CategoriesTable._ID + ") REFERENCES " +
                CategoriesTable.TABLE_NAME + "(" + CategoriesTable._ID + ") " + "ON DELETE CASCADE" +
                ")";
        db.execSQL(CREATE_CATEGORIES_TABLE);
        db.execSQL(CREATE_QUESTIONS_TABLE);
        fillCategoriesTable();
        fillQuestions();
    }
```

Fig. 33

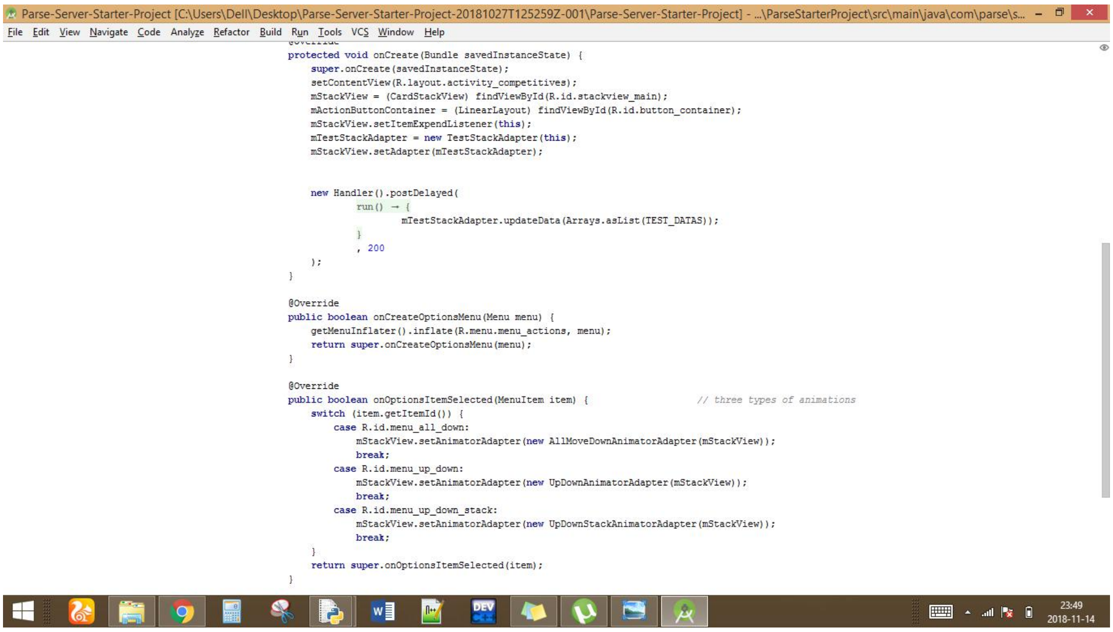
9. COMPETITIONS ACTIVITY

In this activity a list of different contests and hackathons are presented in the form of pallets of card-views that provide information about those competitions.


Parse-Server-Starter-Project [C:\Users\Dell\Desktop\Parse-Server-Starter-Project-20181027T125259Z-001\Parse-Server-Starter-Project] - ...\ParseStarterProject\src\main\java\com\parse\s...
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
23:50
2018-11-14

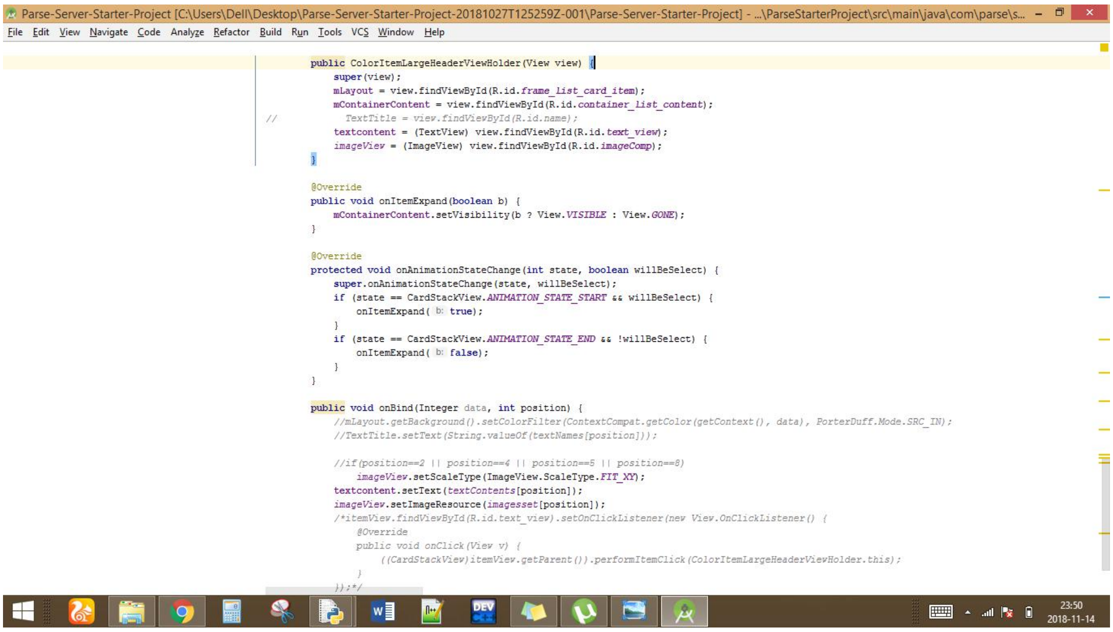
```java
public TestStackAdapter(Context context) {
    super(context);
}

@Override
public void bindView(Integer data, int position, CardStackView.ViewHolder holder) {
    if (holder instanceof ColorItemLargeHeaderViewHolder) {
        ColorItemLargeHeaderViewHolder h = (ColorItemLargeHeaderViewHolder) holder;
        h.onBind(data, position);
    }
    if (holder instanceof ColorItemWithNoHeaderViewHolder) {
        ColorItemWithNoHeaderViewHolder h = (ColorItemWithNoHeaderViewHolder) holder;
        h.onBind(data, position);
    }
    if (holder instanceof ColorItemViewHolder) {
        ColorItemViewHolder h = (ColorItemViewHolder) holder;
        h.onBind(data, position);
    }
}

@Override
protected CardStackView.ViewHolder onCreateView(ViewGroup parent, int viewType) {
    View view;
    switch (viewType) {
        case R.layout.list_card_item_larger_header:
            view = getLayoutInflater().inflate(R.layout.list_card_item_larger_header, parent,  attachToRoot: false);
            return new ColorItemLargeHeaderViewHolder(view);
        case R.layout.list_card_item_with_no_header:
            view = getLayoutInflater().inflate(R.layout.list_card_item_with_no_header, parent,  attachToRoot: false);
            return new ColorItemWithNoHeaderViewHolder(view);
        default:
            view = getLayoutInflater().inflate(R.layout.list_card_item, parent,  attachToRoot: false);
            return new ColorItemViewHolder(view);
    }
}

@Override
public int getItemViewType(int position) {                          //types of header
    /*if (position == 16) {//TODO TEST LARGER ITEM
        return R.layout.list_card_item;
```

Fig. 34


Parse-Server-Starter-Project [C:\Users\Dell\Desktop\Parse-Server-Starter-Project-20181027T125259Z-001\Parse-Server-Starter-Project] - ...\ParseStarterProject\src\main\java\com\parse\s...
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
23:49
2018-11-14

```java
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_competitives);
    mStackView = (CardStackView) findViewById(R.id.stackview_main);
    mActionButtonContainer = (LinearLayout) findViewById(R.id.button_container);
    mStackView.setItemExpendListener(this);
    mTestStackAdapter = new TestStackAdapter(this);
    mStackView.setAdapter(mTestStackAdapter);


    new Handler().postDelayed(
            run() → {
                    mTestStackAdapter.updateData(Arrays.asList(TEST_DATAS));
            }
            , 200
    );
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_actions, menu);
    return super.onCreateOptionsMenu(menu);
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {                   // three types of animations
    switch (item.getItemId()) {
        case R.id.menu_all_down:
            mStackView.setAnimatorAdapter(new AllMoveDownAnimatorAdapter(mStackView));
            break;
        case R.id.menu_up_down:
            mStackView.setAnimatorAdapter(new UpDownAnimatorAdapter(mStackView));
            break;
        case R.id.menu_up_down_stack:
            mStackView.setAnimatorAdapter(new UpDownStackAnimatorAdapter(mStackView));
            break;
    }
    return super.onOptionsItemSelected(item);
}
```

Fig. 35

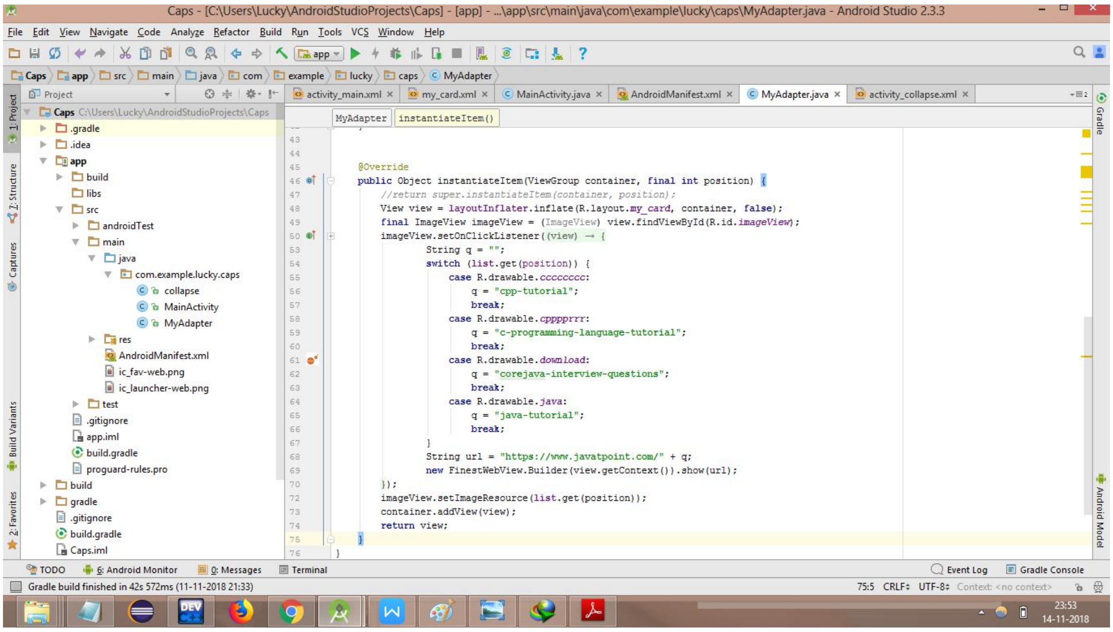

Parse-Server-Starter-Project [C:\Users\Dell\Desktop\Parse-Server-Starter-Project-20181027T125259Z-001\Parse-Server-Starter-Project] - ...\ParseStarterProject\src\main\java\com\parse\s...
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
23:50
2018-11-14

```
    public ColorItemLargeHeaderViewHolder(View view) {
        super(view);
        mLayout = view.findViewById(R.id.frame_list_card_item);
        mContainerContent = view.findViewById(R.id.container_list_content);
//          TextTitle = view.findViewById(R.id.name);
        textcontent = (TextView) view.findViewById(R.id.text_view);
        imageView = (ImageView) view.findViewById(R.id.imageComp);
    }

    @Override
    public void onItemExpand(boolean b) {
        mContainerContent.setVisibility(b ? View.VISIBLE : View.GONE);
    }

    @Override
    protected void onAnimationStateChange(int state, boolean willBeSelect) {
        super.onAnimationStateChange(state, willBeSelect);
        if (state == CardStackView.ANIMATION_STATE_START && willBeSelect) {
            onItemExpand( b: true);
        }
        if (state == CardStackView.ANIMATION_STATE_END && !willBeSelect) {
            onItemExpand( b: false);
        }
    }

    public void onBind(Integer data, int position) {
        //mLayout.getBackground().setColorFilter(ContextCompat.getColor(getContext(), data), PorterDuff.Mode.SRC_IN);
        //TextTitle.setText(String.valueOf(textNames[position]));

        //if(position==2 || position==4 || position==5 || position==8)
            imageView.setScaleType(ImageView.ScaleType.FIT_XY);
        textcontent.setText(textContents[position]);
        imageView.setImageResource(imagesset[position]);
        /*itemView.findViewById(R.id.text_view).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                ((CardStackView)itemView.getParent()).performItemClick(ColorItemLargeHeaderViewHolder.this);
            }
        });*/
```

Fig. 36

10. LEARN ACTIVITY

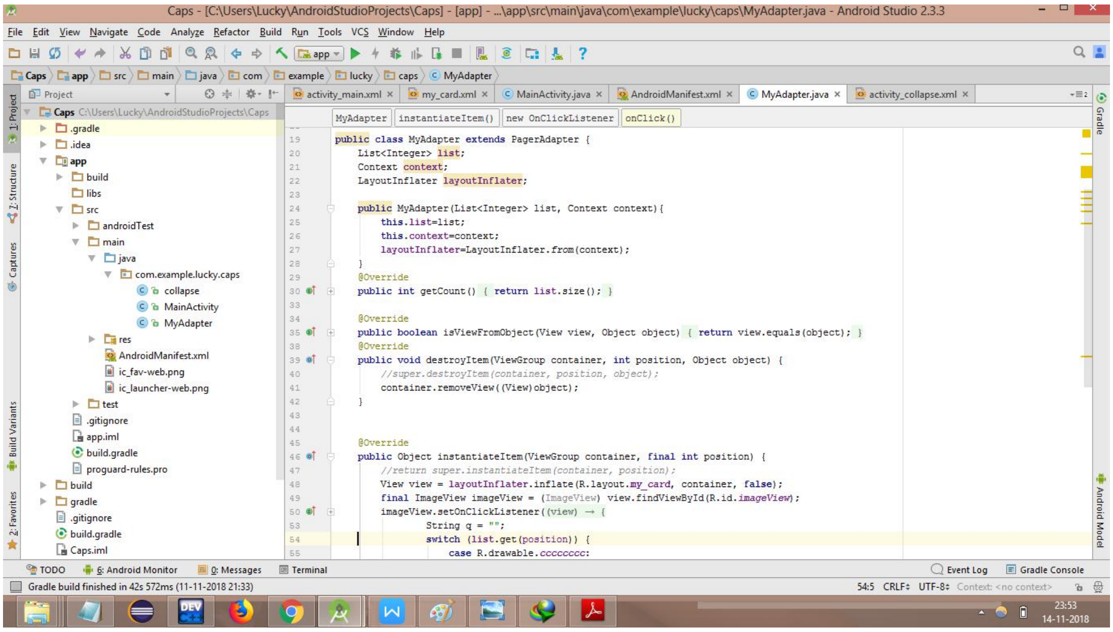
User can learn the languages of their interest by selecting any module.


Caps - [C:\Users\Lucky\AndroidStudioProjects\Caps] - [app] - ...\app\src\main\java\com\example\lucky\caps\MyAdapter.java - Android Studio 2.3.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Caps app src main java com example lucky caps MyAdapter
activity_main.xml my_card.xml MainActivity.java AndroidManifest.xml MyAdapter.java activity_collapse.xml
Project
Caps C:\Users\Lucky\AndroidStudioProjects\Caps
.gradle
.idea
app
build
libs
src
androidTest
main
java
com.example.lucky.caps
collapse
MainActivity
MyAdapter
res
AndroidManifest.xml
ic_fav-web.png
ic_launcher-web.png
test
.gitignore
app.iml
build.gradle
proguard-rules.pro
build
gradle
.gitignore
build.gradle
Caps.iml
MyAdapter instantiateItem()
TODO 6: Android Monitor 0: Messages Terminal Event Log Gradle Console
75:5 CRLF UTF-8 Context: <no context>
23:53
14-11-2018

```
    @Override
    public Object instantiateItem(ViewGroup container, final int position) {
        //return super.instantiateItem(container, position);
        View view = layoutInflater.inflate(R.layout.my_card, container, false);
        final ImageView imageView = (ImageView) view.findViewById(R.id.imageView);
        imageView.setOnClickListener((view) → {
                String q = "";
                switch (list.get(position)) {
                    case R.drawable.cccccccc:
                        q = "cpp-tutorial";
                        break;
                    case R.drawable.cppprrr:
                        q = "c-programming-language-tutorial";
                        break;
                    case R.drawable.download:
                        q = "corejava-interview-questions";
                        break;
                    case R.drawable.java:
                        q = "java-tutorial";
                        break;
                }
                String url = "https://www.javatpoint.com/" + q;
                new FinestWebView.Builder(view.getContext()).show(url);
        });
        imageView.setImageResource(list.get(position));
        container.addView(view);
        return view;
    }
}
```

Gradle build finished in 42s 572ms (11-11-2018 21:33)

Fig. 37



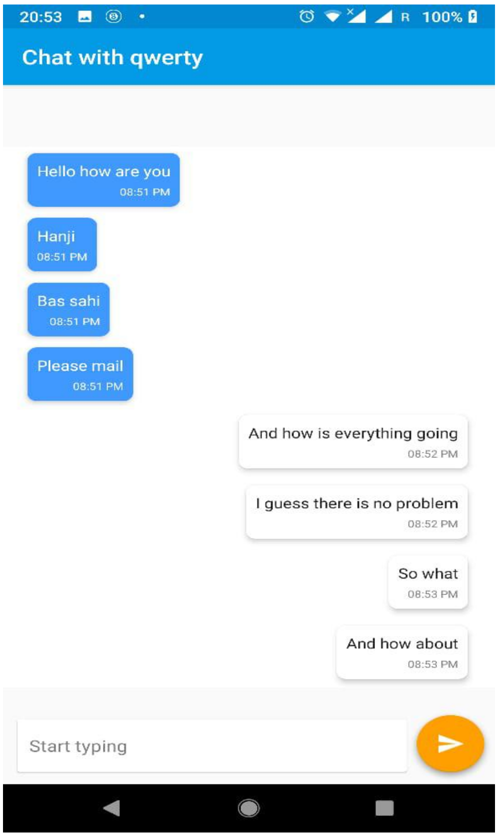
Caps - [C:\Users\Lucky\AndroidStudioProjects\Caps] - [app] - ...\app\src\main\java\com\example\lucky\caps\MyAdapter.java - Android Studio 2.3.3

```java
public class MyAdapter extends PagerAdapter {
    List<Integer> list;
    Context context;
    LayoutInflater layoutInflater;

    public MyAdapter(List<Integer> list, Context context){
        this.list=list;
        this.context=context;
        layoutInflater=LayoutInflater.from(context);
    }
    @Override
    public int getCount() { return list.size(); }

    @Override
    public boolean isViewFromObject(View view, Object object) { return view.equals(object); }
    @Override
    public void destroyItem(ViewGroup container, int position, Object object) {
        //super.destroyItem(container, position, object);
        container.removeView((View)object);
    }


    @Override
    public Object instantiateItem(ViewGroup container, final int position) {
        //return super.instantiateItem(container, position);
        View view = layoutInflater.inflate(R.layout.my_card, container, false);
        final ImageView imageView = (ImageView) view.findViewById(R.id.imageView);
        imageView.setOnClickListener((view) → {
                String q = "";
                switch (list.get(position)) {
                    case R.drawable.cccccccc:
```

Gradle build finished in 42s 572ms (11-11-2018 21:33)

Fig. 38

11.2 SCREENSHOTS



Chat with qwerty

Hello how are you
08:51 PM

Hanji
08:51 PM

Bas sahi
08:51 PM

Please mail
08:51 PM

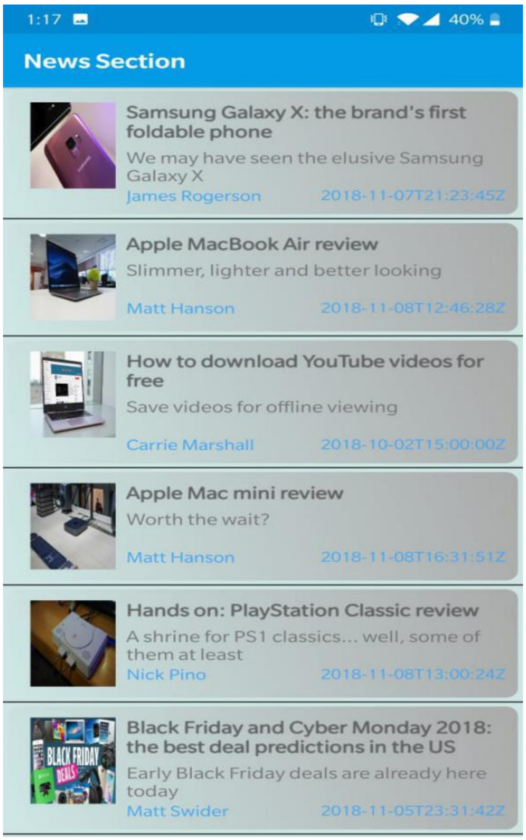
And how is everything going
08:52 PM

I guess there is no problem
08:52 PM

So what
08:53 PM

And how about
08:53 PM

Start typing



News Section

Samsung Galaxy X: the brand's first foldable phone
We may have seen the elusive Samsung Galaxy X
James Rogerson 2018-11-07T21:23:45Z

Apple MacBook Air review
Slimmer, lighter and better looking
Matt Hanson 2018-11-08T12:46:28Z

How to download YouTube videos for free
Save videos for offline viewing
Carrie Marshall 2018-10-02T15:00:00Z

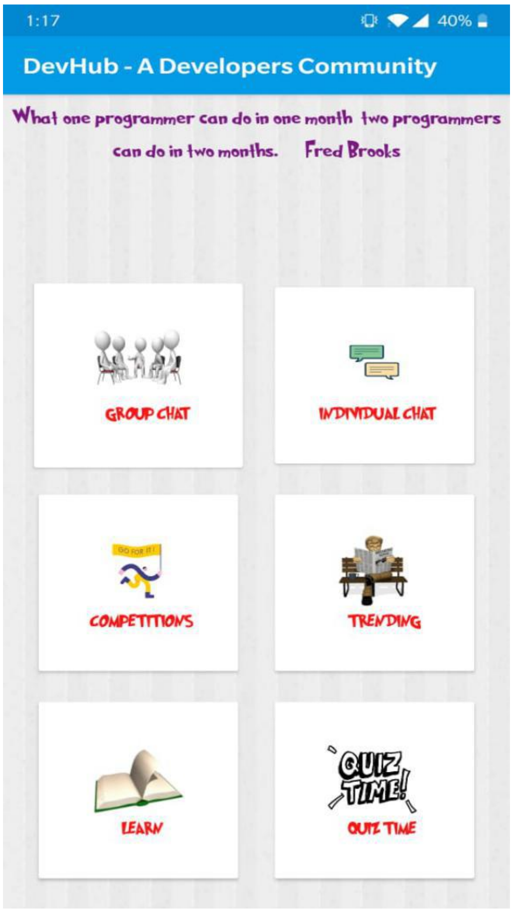
Apple Mac mini review
Worth the wait?
Matt Hanson 2018-11-08T16:31:51Z

Hands on: PlayStation Classic review
A shrine for PS1 classics... well, some of them at least
Nick Pino 2018-11-08T13:00:24Z

Black Friday and Cyber Monday 2018: the best deal predictions in the US
Early Black Friday deals are already here today
Matt Swider 2018-11-05T23:31:42Z

Fig. 39

Fig. 40

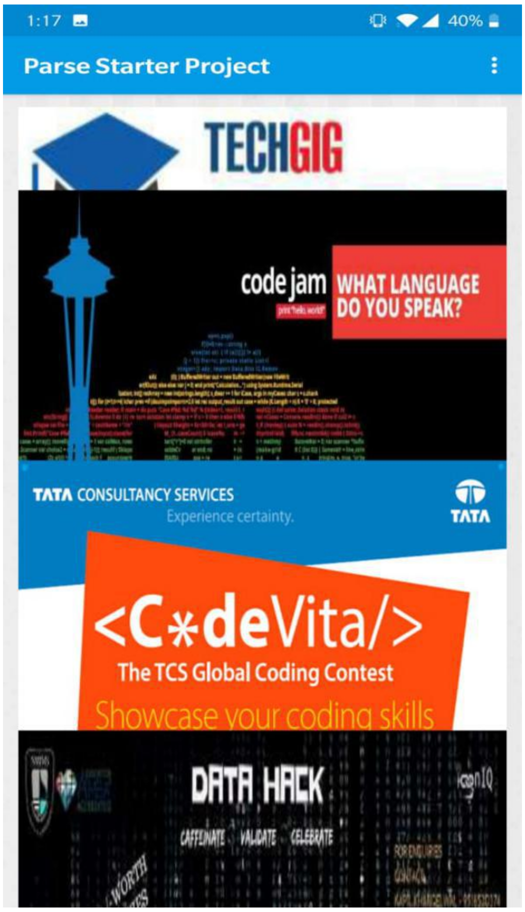


1:17 40%

DevHub - A Developers Community

What one programmer can do in one month two programmers can do in two months. Fred Brooks

Fig. 41

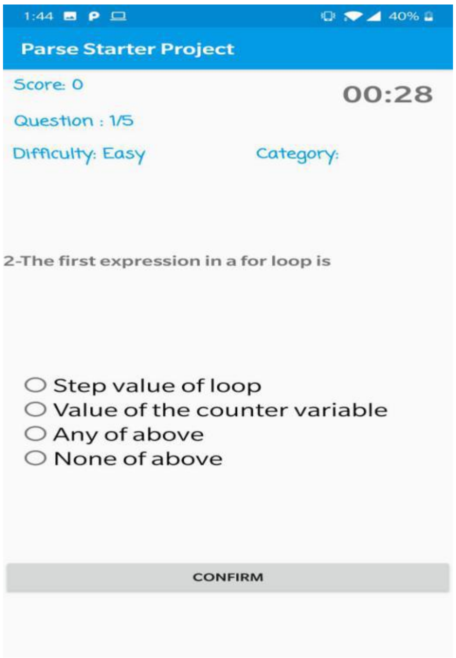


1:17 40%

Parse Starter Project

TECHGIG
code jam
WHAT LANGUAGE DO YOU SPEAK?
TATA CONSULTANCY SERVICES
Experience certainty.
TATA
<C*deVita/>
The TCS Global Coding Contest
Showcase your coding skills
DATA HACK
CAFFEINATE VALIDATE CELEBRATE

Fig. 42



1:44 40%

Parse Starter Project

Score: 0

00:28

Question : 1/5

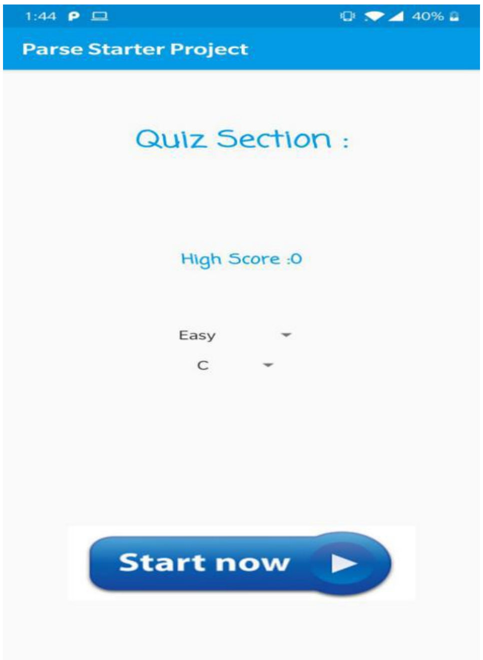
Difficulty: Easy

Category:

2-The first expression in a for loop is

○ Step value of loop
○ Value of the counter variable
○ Any of above
○ None of above

Fig. 43



1:44 40%

Parse Starter Project

Quiz Section :

High Score :0

Easy

C

Fig. 44

12. BIBLIOGRAPHY

1. **https://developers.google.com/training/android/**

2. **https://stackoverflow.com/questions/tagged/android**

3. **https://www.androidhive.info/**

4. **https://www.tutorialspoint.com/android/android_sqlite_database.htm**

5. **https://www.journaldev.com/9438/android-sqlite-database-example-tutorial**

6. **https://github.com/TheFinestArtist/TheFinestArtist-Android**