UNIVERSITY INSTITUTE OF ENGINEERING AND TECHNOLOGY
PANJAB UNIVERSITY, CHANDIGARH

PROJECT: JOB MATCHERS

Computer Science and Engineering
(2019-2023)

Mentor:

Dr. Sarbjeet Singh

Submitted By:

Muskan Mittal (UE193075)
Nancy (UE193077)
Nischay Wadhwa (UE193083)
BE CSE Section - 2

CERTIFICATE

I hereby certify that the work which is being submitted in this project work titled "**JOB MATCHERS**" in partial fulfillment of the requirement for the award of the degree of "**Bachelor of Engineering in Computer Science and Engineering**" submitted in UIET, Panjab University, Chandigarh, is an authentic record of my work carried out under the supervision of **Dr. Sarbjeet Singh** and refers to other researchers work which is duly listed in the reference section. The matter presented in this project work has not been submitted for the award of any other degree of this or any other university.

Muskan Mittal, Nancy, Nischay
(Name of Student)
Roll No: **UE193075, UE193077, UE193083**

This is to certify that the statements made above by the candidate are correct and true to the best of my knowledge.

Dr. Sarbjeet Singh
Professor,
CSE, UIET, Panjab University,
Chandigarh – 160014

DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING

VISION: To be recognized as an eminent department in Computer Science and Engineering education and research for the benefit of society globally.

MISSION:

- To sustain world-class computing infrastructure for the enhancement of technical knowledge in the field of Computer Science and Engineering.
- To excel in research and innovation for the discovery of new knowledge and technologies.
- To produce technocrats, entrepreneurs, and business leaders of the future.
- To foster human values for national growth and life-long learning amongst all the stakeholders.

PROGRAMME:B.E. CSE (UG PROGRAMME)

PROGRAMME EDUCATIONAL OBJECTIVES:

I. Graduates will work as software professionals in industry of repute.

II. Graduates will pursue higher studies and research in engineering and management disciplines.

III. Graduates will work as entrepreneurs by establishing startups to take up projects for societal and environmental causes.

PROGRAMME OUTCOMES:

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to different computer science & Engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts and demonstrate the knowledge of and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and team work: function effectively as a member of a team assembled to undertake a common goal in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

COURSE OUTCOMES (CO):

CO1: Apply the knowledge from previous semesters to undertake and solve a real-life problem

CO2: Illustrate the solution after identifying various objectives of the problem undertaken

CO3: Devise an organized action plan along with all the team members

CO4: Develop a solution using the appropriate methodology and tools available

CO5: Communicate and demonstrate the work through the structured report and oral presentation

ABSTRACT

Job Searching on a website takes a lot of time. One has to go through different job portals to search for job openings which takes around 20-30 minutes everyday. To make this task easy we have designed an app which not only tells the various job openings but also provides a link to the application page of that job.

With Job Matchers we want to make the task of looking for job openings easier, reliable and time efficient. So that the person can focus only on those job openings which are relevant to him/her. The person will then be provided with the details of his area of interest for the job.

TABLE OF CONTENTS

DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING 2

PROGRAMME EDUCATIONAL OBJECTIVES 2

PROGRAMME OUTCOMES 3

COURSE OUTCOMES (CO) 4

ABSTRACT 5

TABLE OF CONTENTS 6

INTRODUCTION 7

MOTIVATION 7

PROPOSED SYSTEM 8

SOFTWARE REQUIREMENTS SPECIFICATION DOCUMENT 9

1. DATA REQUIREMENTS 9

2. FUNCTIONAL REQUIREMENTS 9

3. PERFORMANCE REQUIREMENTS 10

4. SYSTEM REQUIREMENTS 10

5. TESTING AND MAINTAINABILITY REQUIREMENTS 10

DESIGNING THE PRODUCT ARCHITECTURE 11

1. UML CLASS DIAGRAM 11
2. USE CASE DIAGRAM 13
3. SEQUENCE DIAGRAM 14

IMPLEMENTATION 19

TESTING 23

MAINTENANCE 29

RESULTS AND DISCUSSIONS 30

FUTURE SCOPE 33

CONCLUSION 34

REFERENCES 35

Course Exit Survey 36

INTRODUCTION

MOTIVATION

Freshers look out for off campus job openings on various websites on a daily basis. Also, working professionals who want to switch their jobs also look out for recent job openings in companies. For this they need to search for jobs on various websites.

But, the task of visiting a website and searching for relevant jobs is time consuming. People need to spend 30-40 minutes to gather relevant job openings. So, to make this time efficient as well as reliable we thought of developing an android app 'Job Matchers'.

'Job Matchers' helps users to find recent and relevant job openings on just a single search click. Users only need to specify their job requirements and they will get a list of jobs they are interested in. They will be redirected to apply link. This will reduce the time for searching for a job. Also in this way users can apply to more jobs in a single day.

PROPOSED SYSTEM

The proposed solution is an Android Application, "Job Matchers" which tries to reduce time of search. Why a Mobile Application? A Mobile Application is a set of software which is designed and developed in such a way that it can run on any mobile device, smartphone or tablet. Job Mathers is an Android mobile application which runs on an Android platform with Internet support. In this project, the Job Matchers Android application helps people find job openings relevant to them. The user will be asked to fill job role, job location and job type. After clicking the search button a list of relevant jobs is shown. Job details include job role, company name, location, job type and the portal name through which job has been taken from.

Then the user can click on a particular job and will be redirected to apply a link of the job. In this way, Job Matchers provides reliable and relevant jobs and their information.

SOFTWARE REQUIREMENTS SPECIFICATION DOCUMENT

1. DATA REQUIREMENTS

The set of data that is involved in any project is defined using data requirements. For this project, the main data required is the job role, job location and job type to search for any current openings. Without this information the application is not able to find jobs data.

2. FUNCTIONAL REQUIREMENTS

Functional requirements are properties that must exist in the final system. For any mobile application, we need to download the application from the play store. The application could be either free or paid depending upon the store or merchant. To use the application,

- The user needs to enter details of job.
- Then he/she will be shown a list of jobs related to them.
- Then they will be redirected to page with apply link for a particular job.

3. PERFORMANCE REQUIREMENTS

Response time, scalability, platform dependencies, tolerance are the performance requirements that should be considered when developing any system. The application or system should be able to respond quickly when the user interacts with the application. The application should be developed in such a way that it should be scalable enough to accept new features when we want to expand the application complexity. The application should run in all the specified software and hardware requirements from the design phase of the project. Also, the tolerance rate (fault tolerance) of the application should be at a higher level in case of network issues, connectivity issues, and when the application crashes or stops. It should be able to deliver the information about any of those issues to the user when the system is no longer able to provide results when the user wants.

4. SYSTEM REQUIREMENTS

The application should be installed into a device, system or any machine in such a way that it should have basic requirements like supporting software and hardware of the device, accessing in-built software, say camera for mobile device, internet permissions, and potential security issues such as virus or any malware detection.

5. TESTING AND MAINTAINABILITY REQUIREMENTS

Apart from designing and developing the application, maintaining the application is one of the important characteristics. The developer/owner of the application should be concerned about the maintenance of the application by fixing the issues. The solution for fixing such type of issues when the application crashes by using any of the “Crash Reporting Service” that are available for mobile applications.

Following needs to be checked for maintenance of the app.

- The API used should work well in the future as well.
- The job openings listed should be recent and the data related to job shown to user is reliable.

DESIGNING THE PRODUCT ARCHITECTURE

A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

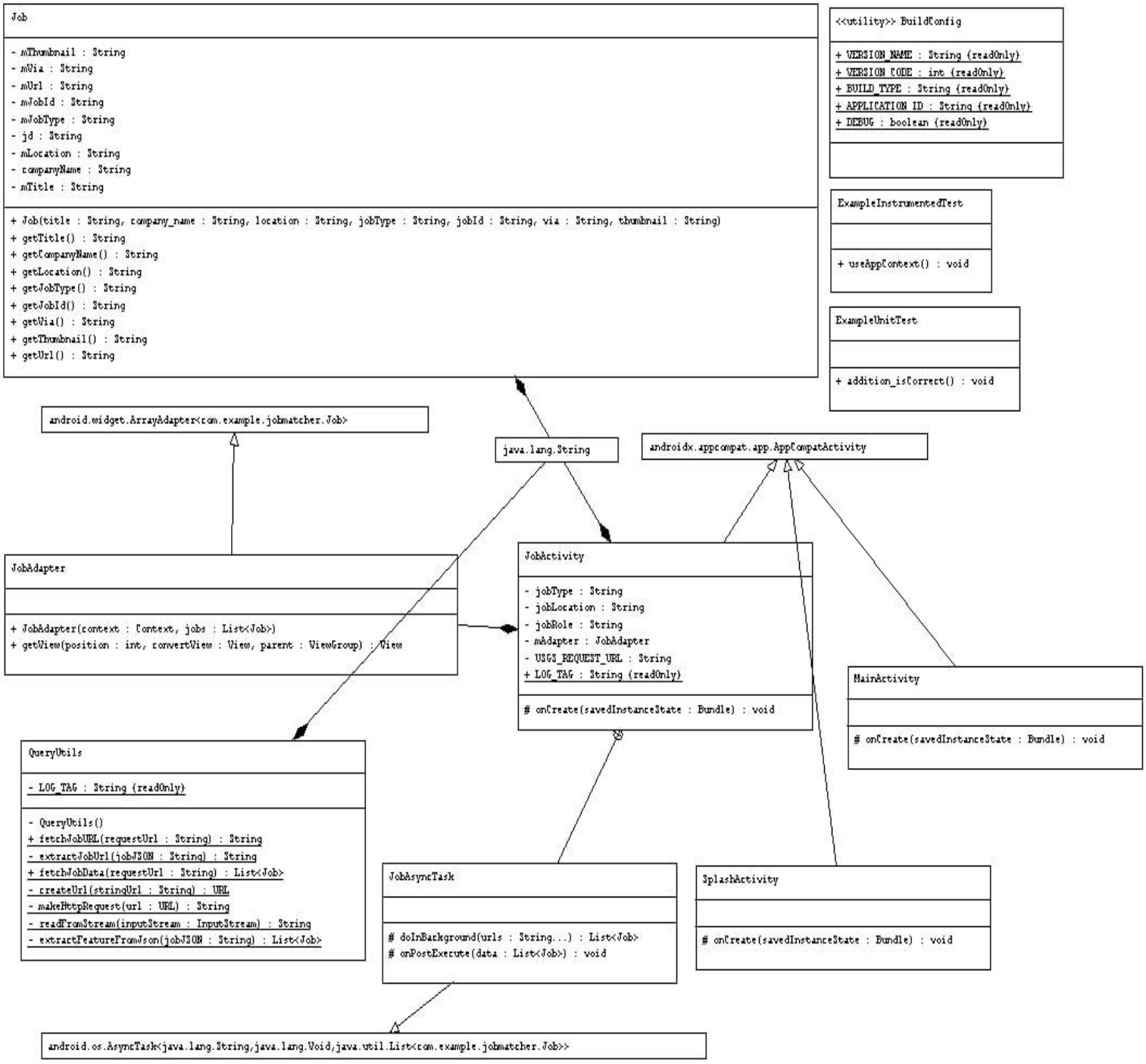
In this third phase, the system and software design documents are prepared as per the requirement specification document. This helps define overall system architecture.

This design phase serves as input for the next phase of the model.

UML CLASS DIAGRAM

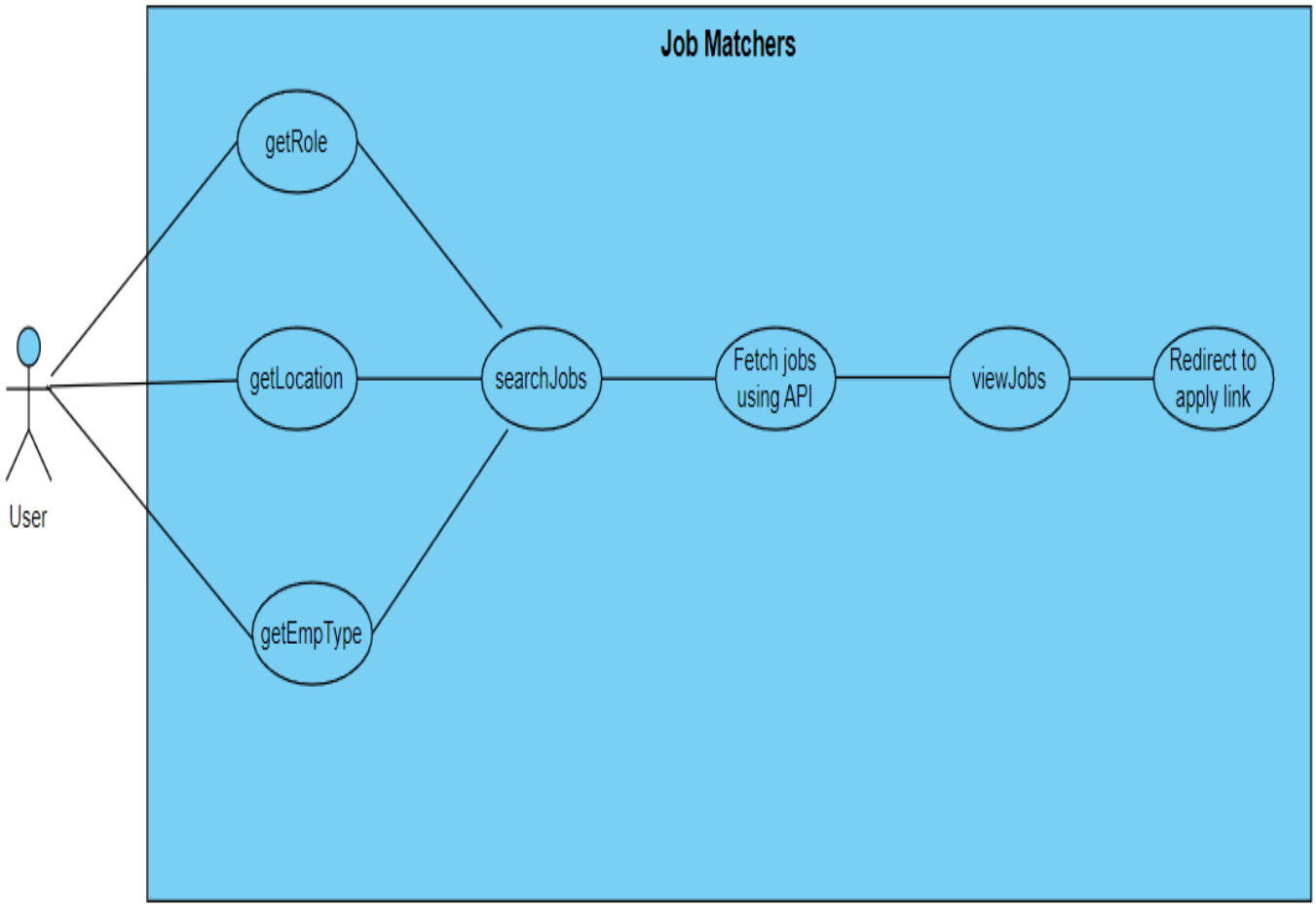
Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.


Job
- mThumbnail : String
- mVia : String
- mUrl : String
- mJobId : String
- mJobType : String
- jd : String
- mLocation : String
- companyName : String
- mTitle : String
+ Job(title : String, company_name : String, location : String, jobType : String, jobId : String, via : String, thumbnail : String)
+ getTitle() : String
+ getCompanyName() : String
+ getLocation() : String
+ getJobType() : String
+ getJobId() : String
+ getVia() : String
+ getThumbnail() : String
+ getUrl() : String
<<utility>> BuildConfig
+ VERSION_NAME : String {readOnly}
+ VERSION_CODE : int {readOnly}
+ BUILD_TYPE : String {readOnly}
+ APPLICATION_ID : String {readOnly}
+ DEBUG : boolean {readOnly}
ExampleInstrumentedTest
+ useAppContext() : void
ExampleUnitTest
+ addition_isCorrect() : void
android.widget.ArrayAdapter<com.example.jobmatcher.Job>
java.lang.String
androidx.appcompat.app.AppCompatActivity
JobAdapter
+ JobAdapter(context : Context, jobs : List<Job>)
+ getView(position : int, convertView : View, parent : ViewGroup) : View
JobActivity
- jobType : String
- jobLocation : String
- jobRole : String
- mAdapter : JobAdapter
- USGS_REQUEST_URL : String
+ LOG_TAG : String {readOnly}
# onCreate(savedInstanceState : Bundle) : void
MainActivity
onCreate(savedInstanceState : Bundle) : void
QueryUtils
- LOG_TAG : String {readOnly}
- QueryUtils()
+ fetchJobURL(requestUrl : String) : String
- extractJobUrl(jobJSON : String) : String
+ fetchJobData(requestUrl : String) : List<Job>
- createUrl(stringUrl : String) : URL
- makeHttpRequest(url : URL) : String
- readFromStream(inputStream : InputStream) : String
- extractFeatureFromJson(jobJSON : String) : List<Job>
JobAsyncTask
doInBackground(urls : String...) : List<Job>
onPostExecute(data : List<Job>) : void
SplashActivity
onCreate(savedInstanceState : Bundle) : void
android.os.AsyncTask<java.lang.String,java.lang.Void,java.util.List<com.example.jobmatcher.Job>>

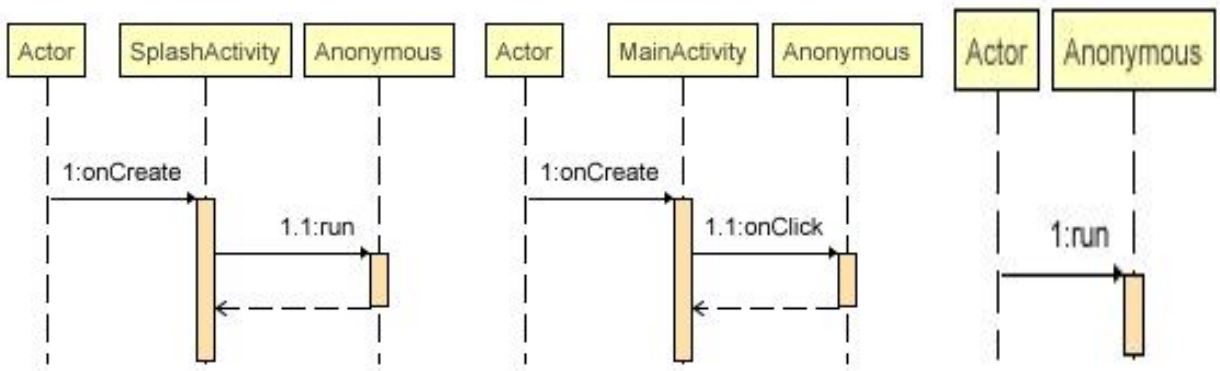
USE CASE DIAGRAM

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.

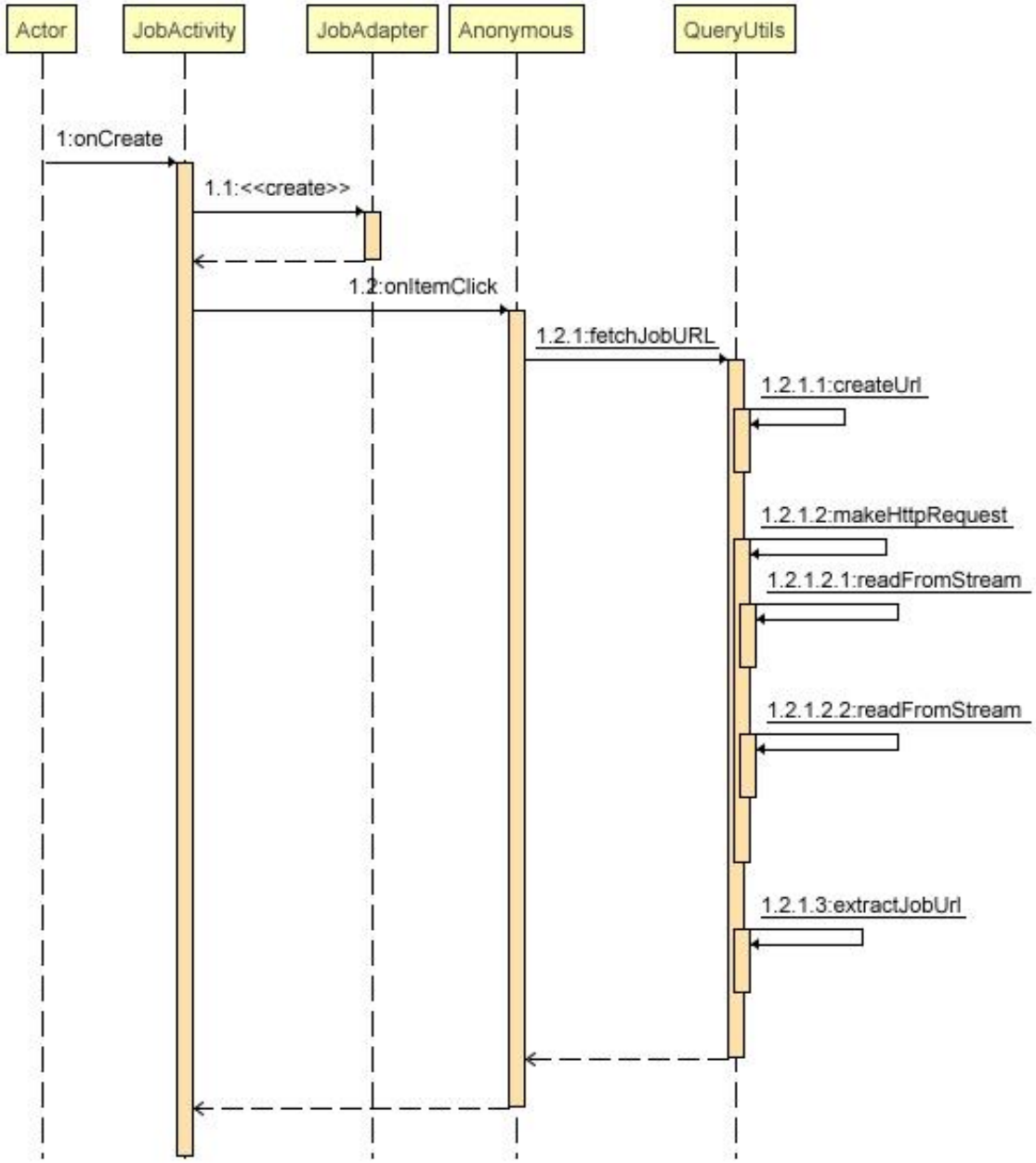


SEQUENCE DIAGRAM

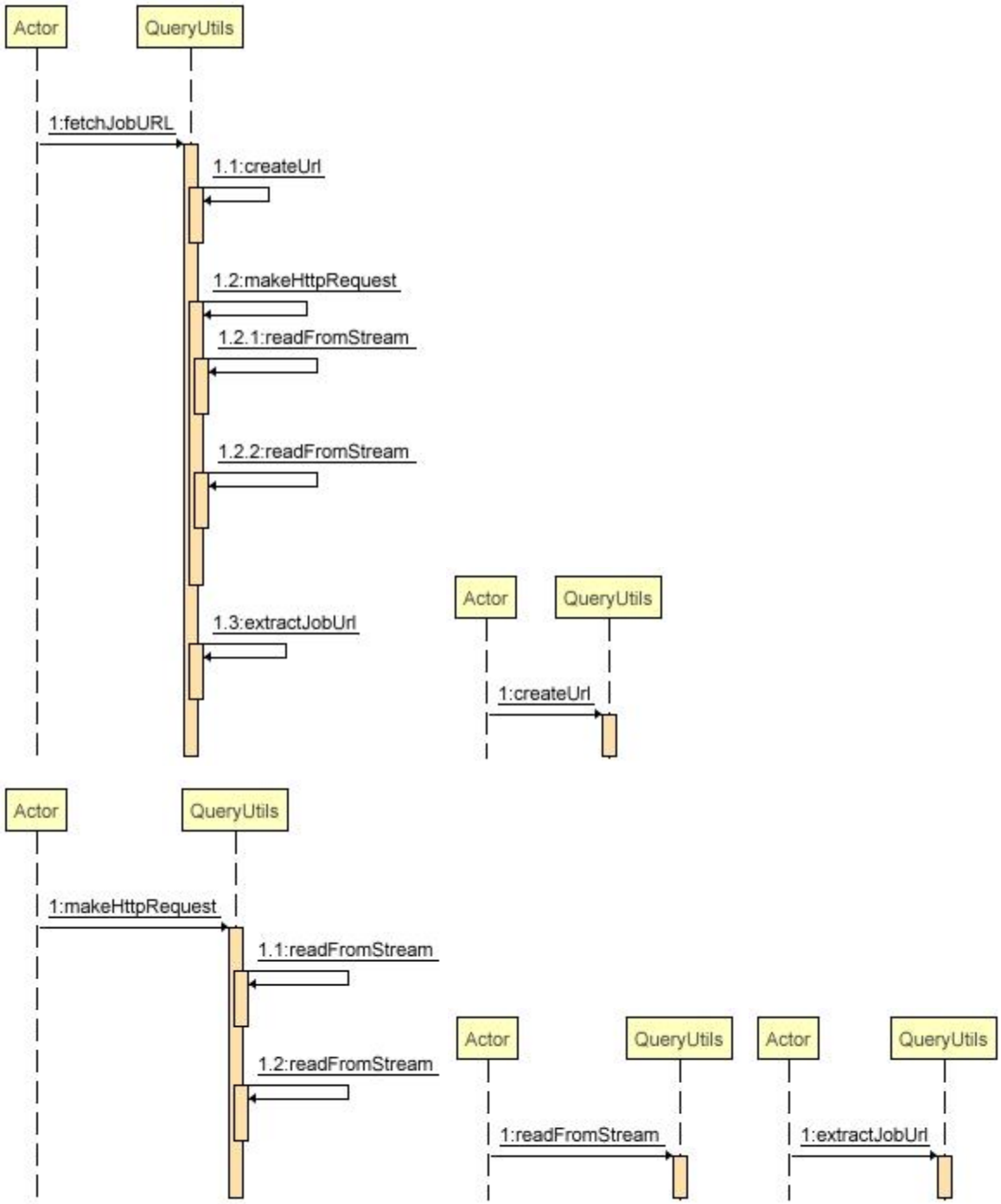
The sequence diagram represents the flow of messages in the system and is also termed as an event diagram. It helps in envisioning several dynamic scenarios. It portrays the communication between any two lifelines as a time-ordered sequence of events, such that these lifelines took part at the run time.

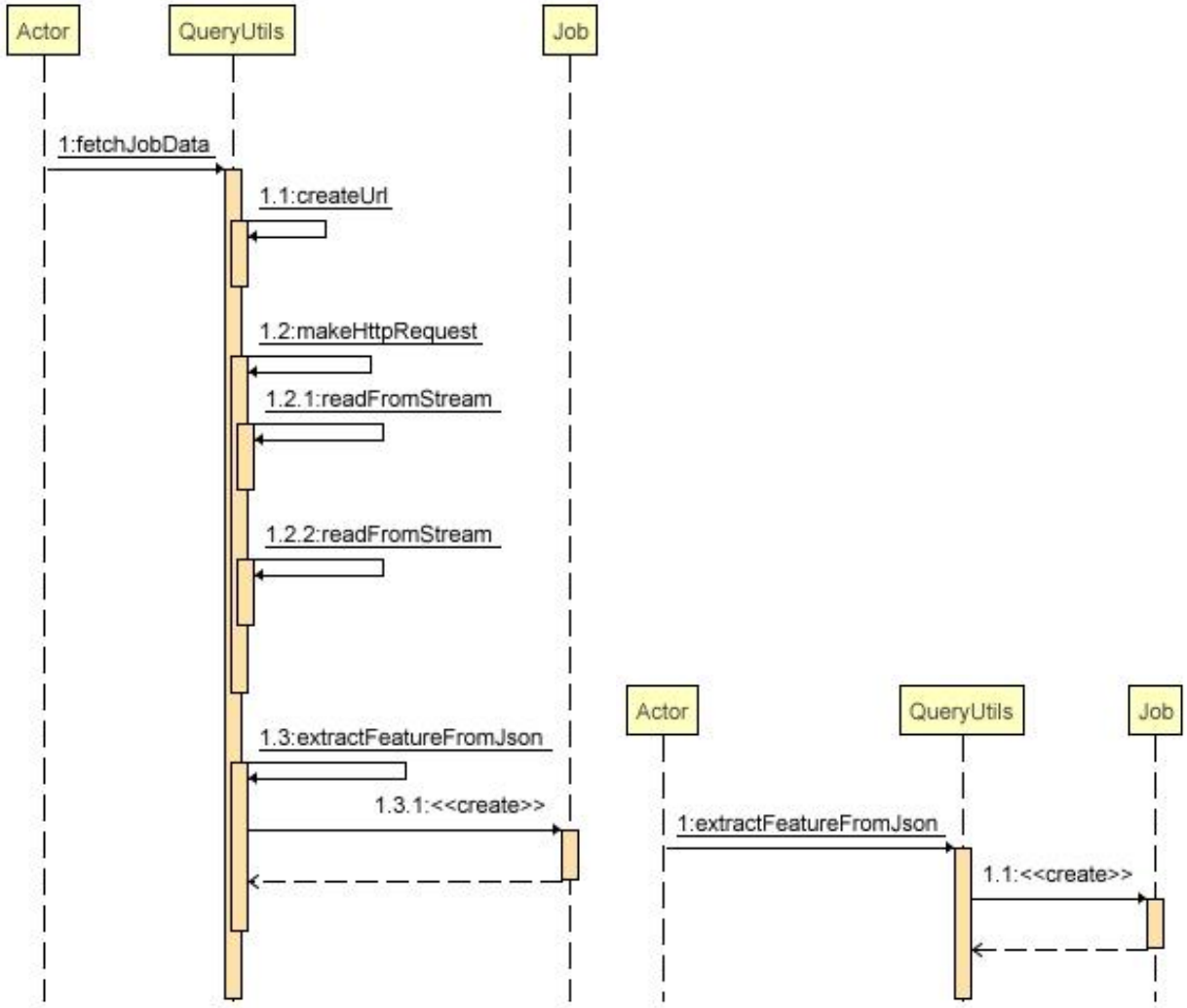


The sequence of the system starts from the Splash Activity . The system shows the splash screen by displaying the logo of the app JOB MATCHERS . When the app is ready, the splash screen is dismissed and the app is displayed . Next come the Main Activity which is responsible for taking the inputs from the user for job role, location and the type of employment they need . The search button is used for searching the results for given inputs .

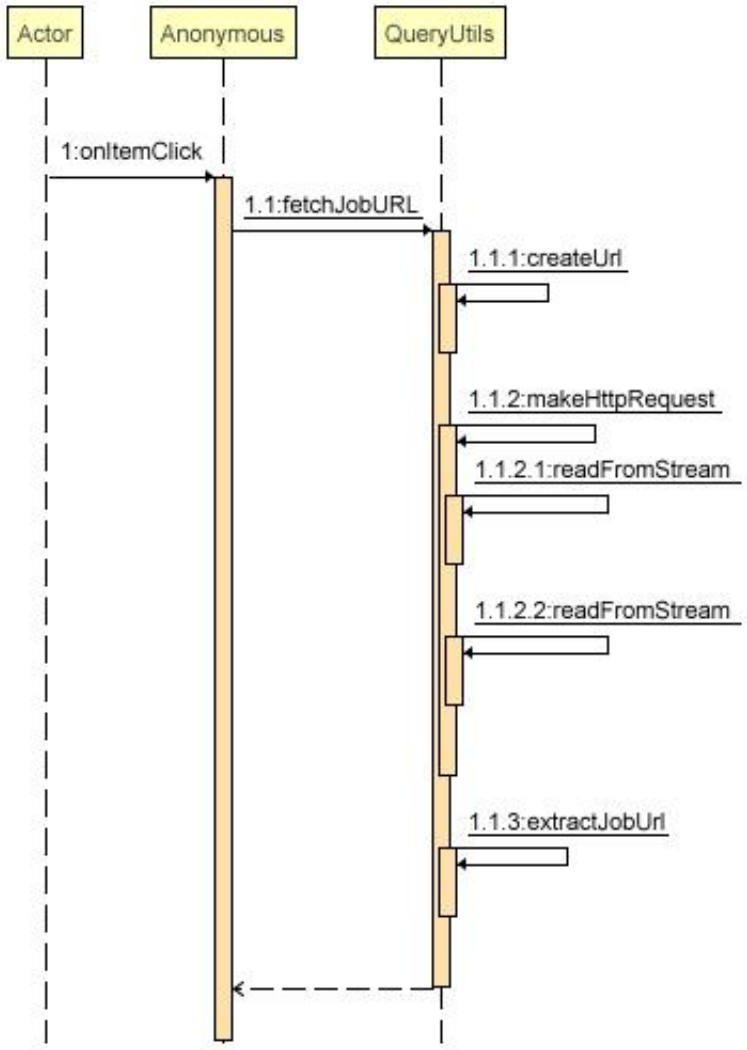


After the Search button is clicked , an API call is made to the first API from which we fetch the desired information and using the jobID from the first API we make a call to the second API for getting the jobURL. The jobAdapter Activity is responsible for adding each job in the listview to display the list of jobs. The Onclick() method in Job Activity is responsible for sending the user to the job opening in the browser .


Actor
QueryUtils
1:fetchJobURL
1.1:createUrl
1.2:makeHttpRequest
1.2.1:readFromStream
1.2.2:readFromStream
1.3:extractJobUrl
Actor
QueryUtils
1:createUrl
Actor
QueryUtils
1:makeHttpRequest
1.1:readFromStream
1.2:readFromStream
Actor
QueryUtils
1:readFromStream
Actor
QueryUtils
1:extractJobUrl



Functions in the Query Utils are the helper functions to facilitate the proper functioning of the system.


Actor
Anonymous
QueryUtils
1:onItemClick
1.1:fetchJobURL
1.1.1:createUrl
1.1.2:makeHttpRequest
1.1.2.1:readFromStream
1.1.2.2:readFromStream
1.1.3:extractJobUrl

IMPLEMENTATION

Introduction to programming languages, IDE's, tools and technologies used for this implementation.

Programming Languages:

- **Java**

  As the project is developing an Android Application, the default programming language is Java. All Android applications are built using Java in Android Studio or Eclipse. Java is a popular and widely used language throughout the world. Java is one of the powerful programming languages like C, C++. developed by Sun Microsystems which has many powerful features as described below. It is object oriented programming language, one of the important hierarchies in the programming languages which is used to implement real time applications, it provides for code reusability, it has a platform independence feature including any virtual machines(Write Once Read Everywhere). It is more secure as the compilers are designed efficiently to figure out any kind of errors.

  - ➢ Java has been used for creating all the modules like Job, JobActivity, JobAdapter, MainActivity, QueryUtils.
  - ➢ We have created a splash screen using JAVA.
  - ➢ JAVA has been used as the backend language for this whole project.

- **XML**

  XML stands for eXtensible Markup Language, which is a way of describing data using a text-based document. Because XML is extensible and very flexible, it's used for many different things, including defining the UI layout of Android apps.

  - ➢ We have designed the user interface part using XML for the whole app.
  - ➢ XML has been used as a frontend language for this project.

- ➢ XML provides different kinds of layouts like linear, constraint, relative and according to the design requirements we have used these layouts in the app.
- ➢ XML also provides text views, edit views, image views which makes it very convenient for a developer to design the layout of an app.

IDE's, Tools and Technologies

- **Android Studio:**
  Android Studio is exclusively designed for developing Android applications. It consists of all Android SDK tools to design, develop, maintain, test, debug and publish our app. The IDE is designed very efficiently which makes the developer's job easy. All java files, layout files (for design) are integrated into a single project easily. After the completion of project, the whole application could be put as an .APK (Android Package) file, in which we can run that APK file in any device and use the application.

Other main tools include Android SDK, SDK Manager, and Gradle Build.

- **Android Software Development Kit (SDK):**
  One of the main tools used in developing android applications, as it packages many core features into one SDK and it can be used in the application easily. This helps us to avoid writing a lot of code, and building applications faster.

- **Gradle Build:**
  Gradle Scripts are the recent feature that is added to Android Studio. It is basically an automated build system which is used to automate the various phases involved in designing an application that includes design, development, test, debug, and publish. We need to configure the project and modules by mentioning all the supported jar files, SDK's, version name, level, compiled SDK version, build tools version. to ensure that the developed app is compatible with the testing device/emulator. Gradle is also similar to Ant and Maven which helps in maintaining java projects (repositories).

- **SDK Manager:**
  It is one of the main tools to maintain the updates of all the installed components required to run the project. It also notifies us when the project is not compatible with the device or any other compatibility issues and to download any component that is required.

- **Material Design:**
  Material design is a comprehensive guide for visual, motion, and interaction design across platforms and devices. To use material design in your Android apps, follow the guidelines defined in the material design specification and use the new components and styles available in the material design support library.
  We have followed material design for designing the layout of android app.

Security and Permissions in Android

Security notions in Android are quite high. Whenever a new Android Application is created, a unique user and group ID. This makes the maintenance of the application in an easier way to avoid any security or privacy issues. As the application is created uniquely, it becomes private and no one can access other applications.
It is included within the tags as it is an XML file. Permissions are automatically created for the basic applications at the time when we create the application. If the app uses a higher level API or SDK we must explicitly mention the permissions inside uses-permissions tag to access the features or components.

TESTING

Black Box testing:

In this project, sample test cases are written and manual testing is done to check the functionality of the application. Black box testing is a technique of software testing which examines the functionality of software without peering into its internal structure or coding. The primary source of black box testing is a specification of requirements that is stated by the customer.

In this method, tester selects a function and gives input value to examine its functionality, and checks whether the function is giving expected output or not. If the function produces correct output, then it is passed in testing, otherwise failed.

Following are two test cases for which we will test the app:

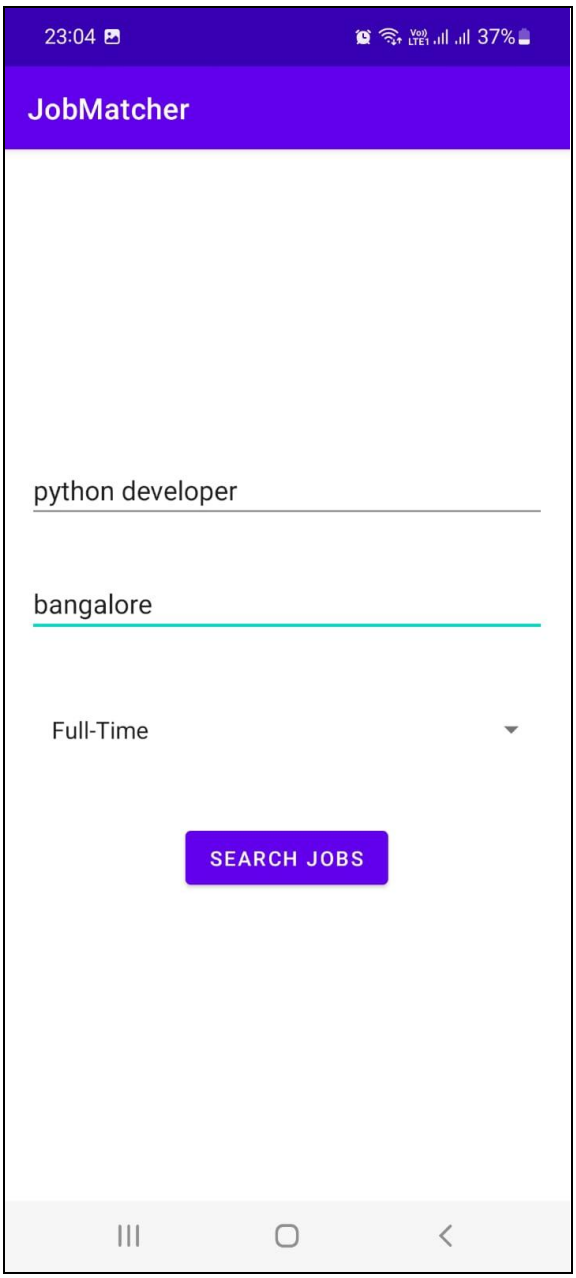
1. Adding python developer as job role, Bangalore as location, and Full time role.

   This will output a list of related jobs.

2. Adding python developer as job role, Bangalore as location, and Full time role.

   This will output a list of related jobs.

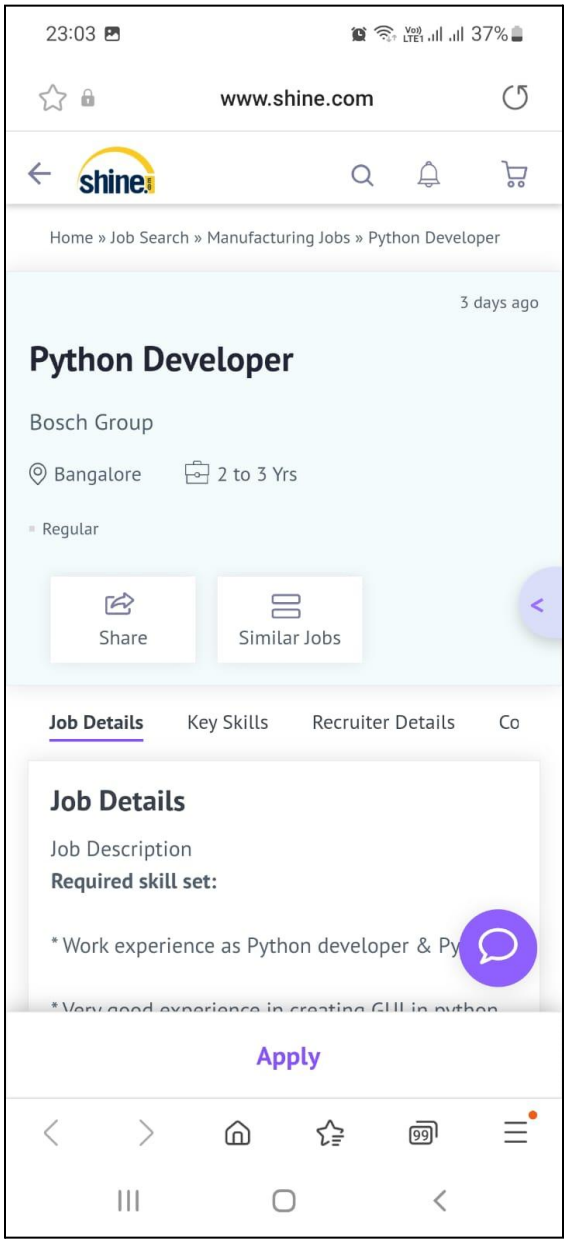
For Test 1



Enter Data

List of Jobs

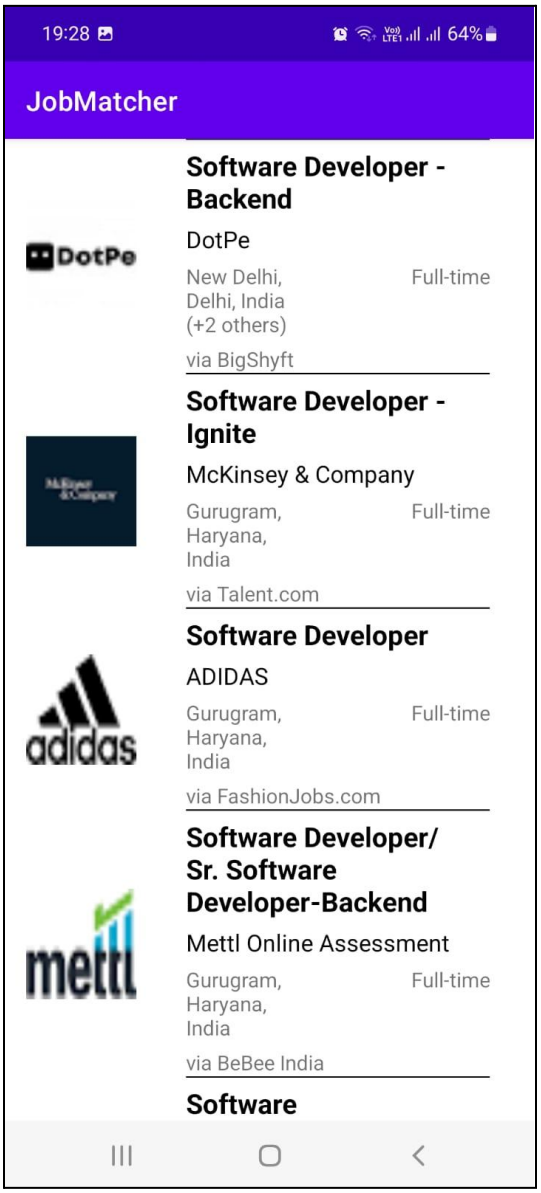
List of Jobs



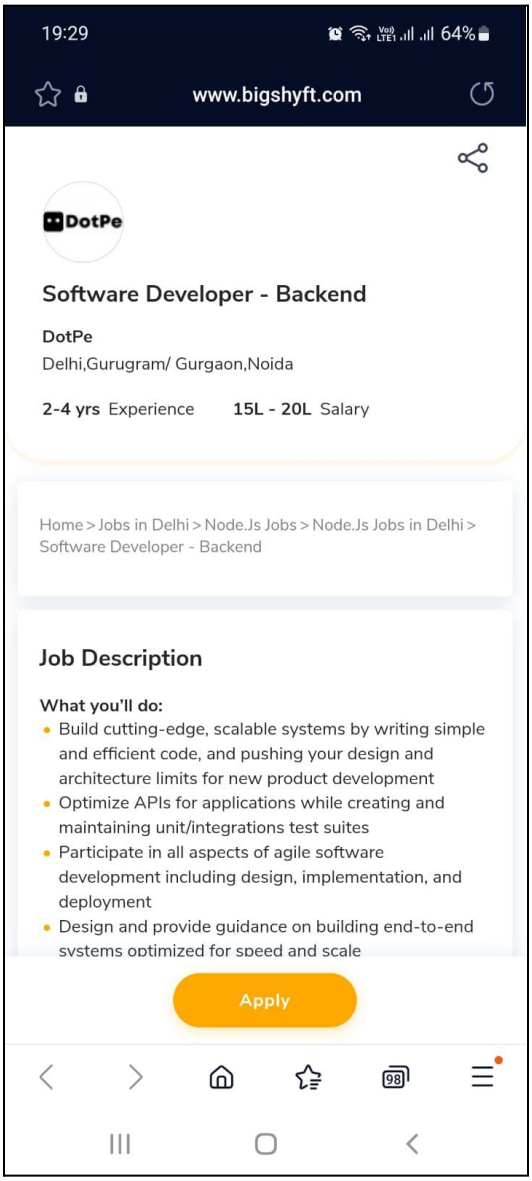
Redirected to Job Link Page

For Test 2

Enter Data



List of Jobs

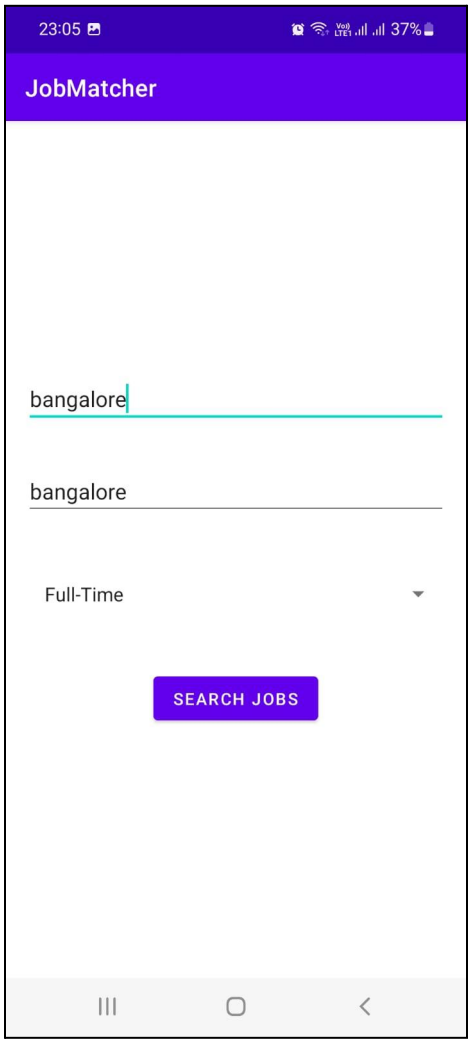


Redirected to Job Link Page

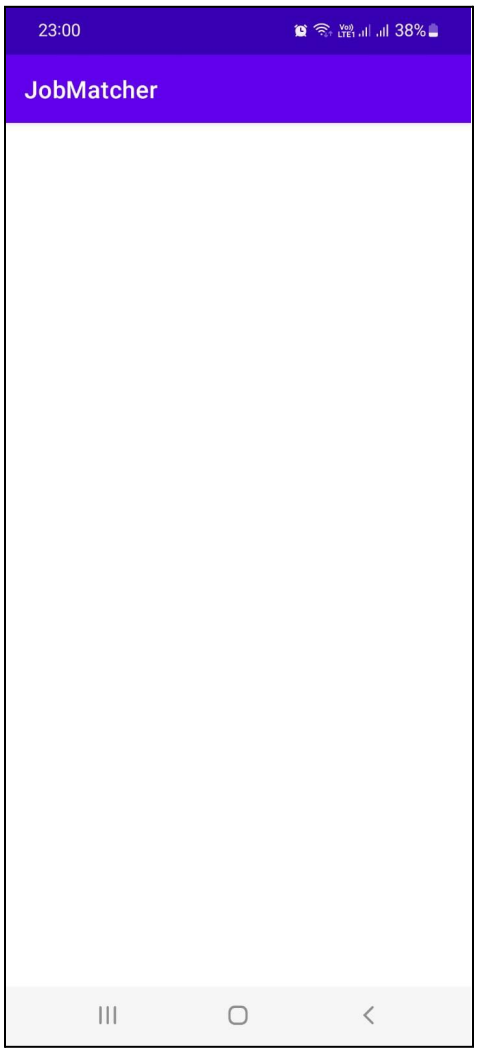
White Box testing:

Once the application meets the user requirements and functionalities according to the test cases, its internal logic are completely tested to ensure that the application does not have any logical errors or issues.It tests internal coding and infrastructure of a software focus on checking of predefined inputs against expected and desired outputs. It is based on inner workings of an application and revolves around internal structure testing. In this type of testing programming skills are required to design test cases. The primary goal of white box testing is to focus on the flow of inputs and outputs through the software and strengthening the security of the software.

For this test we enter job location in place of job role, so in output we do not get any list of jobs as there is no logical error in the app.



Enter data



No Job is shown

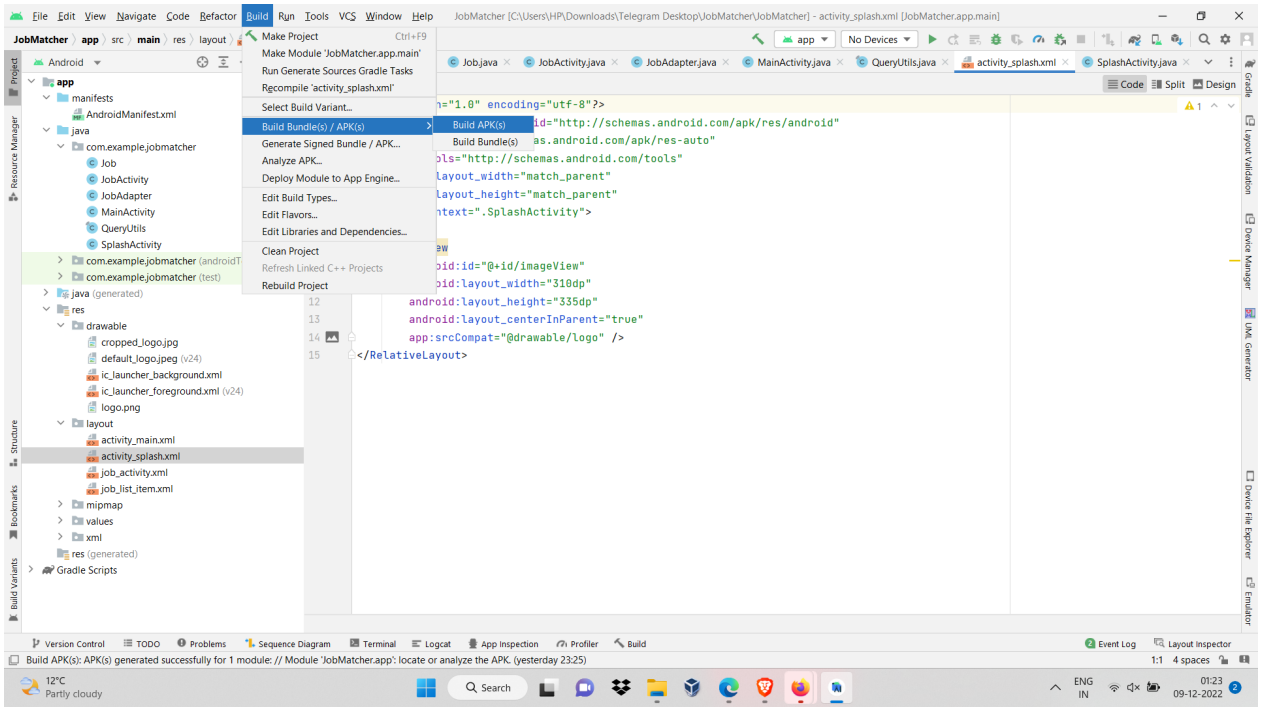
System Testing:

It refers to checking whether the system in which the application is built meets the necessary requirements like software support. For example: In this project, we have checked whether the device in which the application developed is compatible with the software (Android Studio)

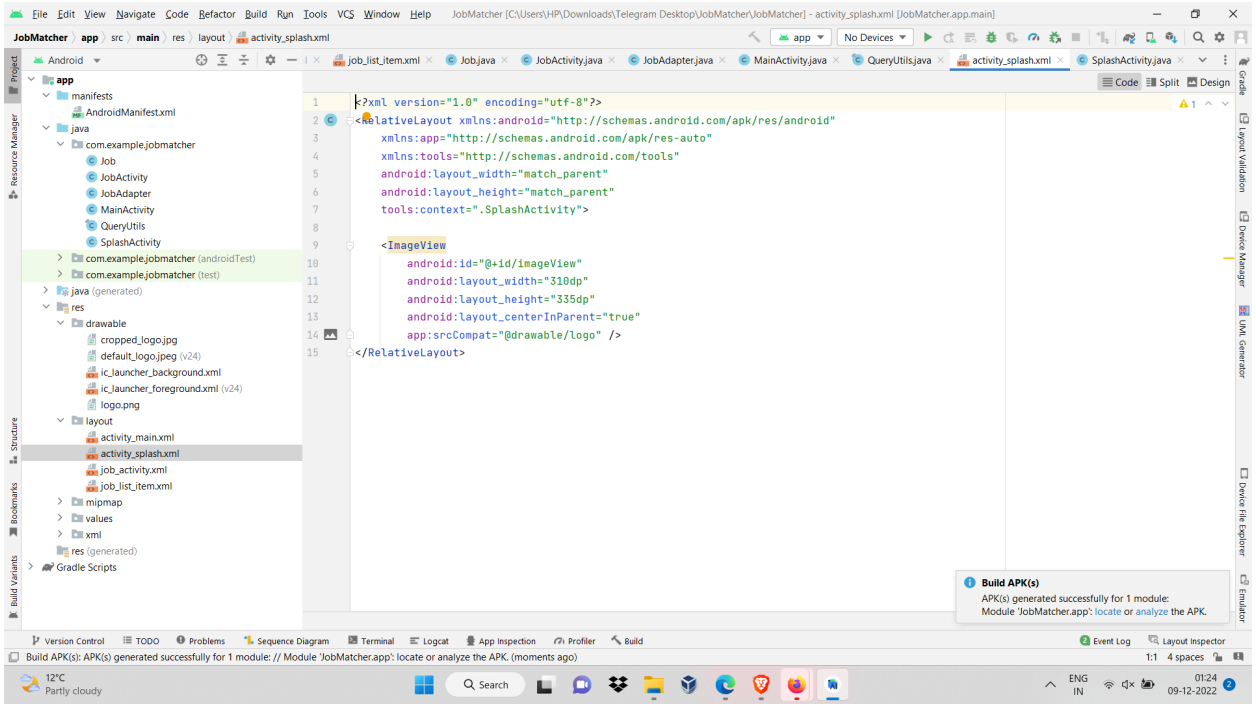
End to End Testing:

Tested the complete environment of application by connecting the device with different machines, installing as an APK file, with the database and in local network. End-to-end testing is a technique used to test whether the flow of an application right from start to finish is behaving as expected. The purpose of performing end-to-end testing is to identify system dependencies and to ensure that the data integrity is maintained between various system components and systems.

Below figure represents the built of APK:



Build APK option



This image shows that apk has been successfully created

MAINTENANCE

Apart from designing and developing the application, maintaining the application is one of the important characteristics. The developer/owner of the application should be concerned about the maintenance of the application by fixing the issues. The solution for fixing such type of issues when the application crashes by using any of the “Crash Reporting Service” that are available for mobile applications. Following need to be checked for maintenance of the app.

- The API used should work well in the future as well.

- The job openings listed should be recent and the data related to job shown to user is reliable.

RESULTS AND DISCUSSIONS

To make the application interactive, different controls have been used and designed using the layout file. Following are the important controls that are designed and used in this application:

Description of Features and the Approach

Front Page of app

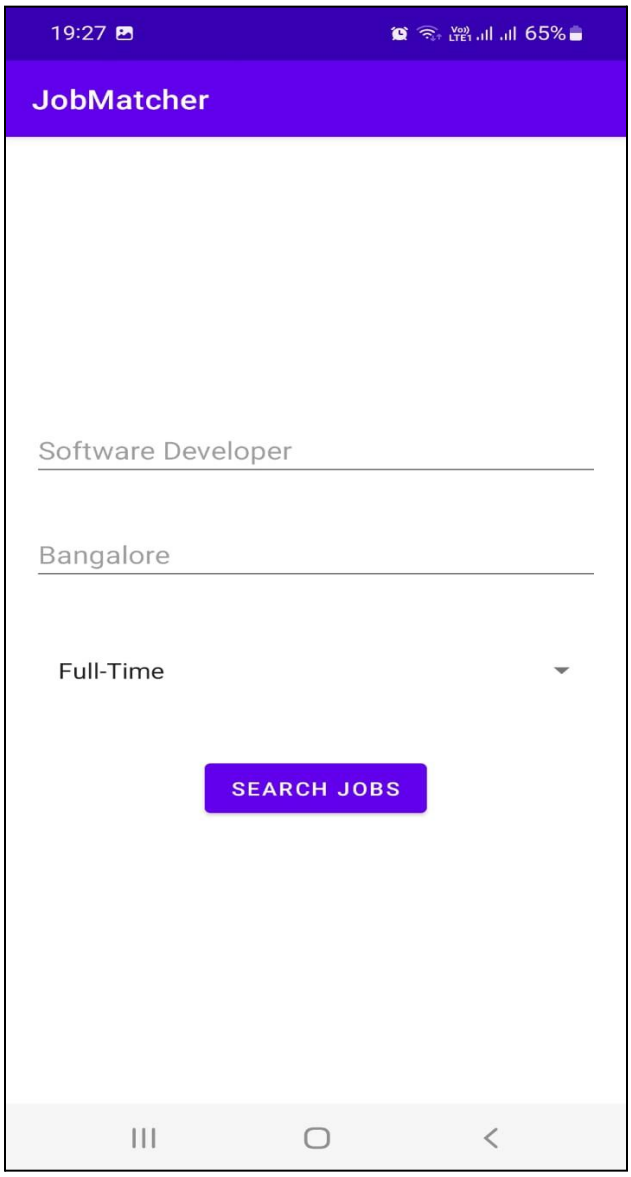
This image shows the front page of the app which displays the logo and name of the app.

This screen is also known as splash screen.

We have created a different activity known as splash activity for this page.

This page will remain for around 2000ms.

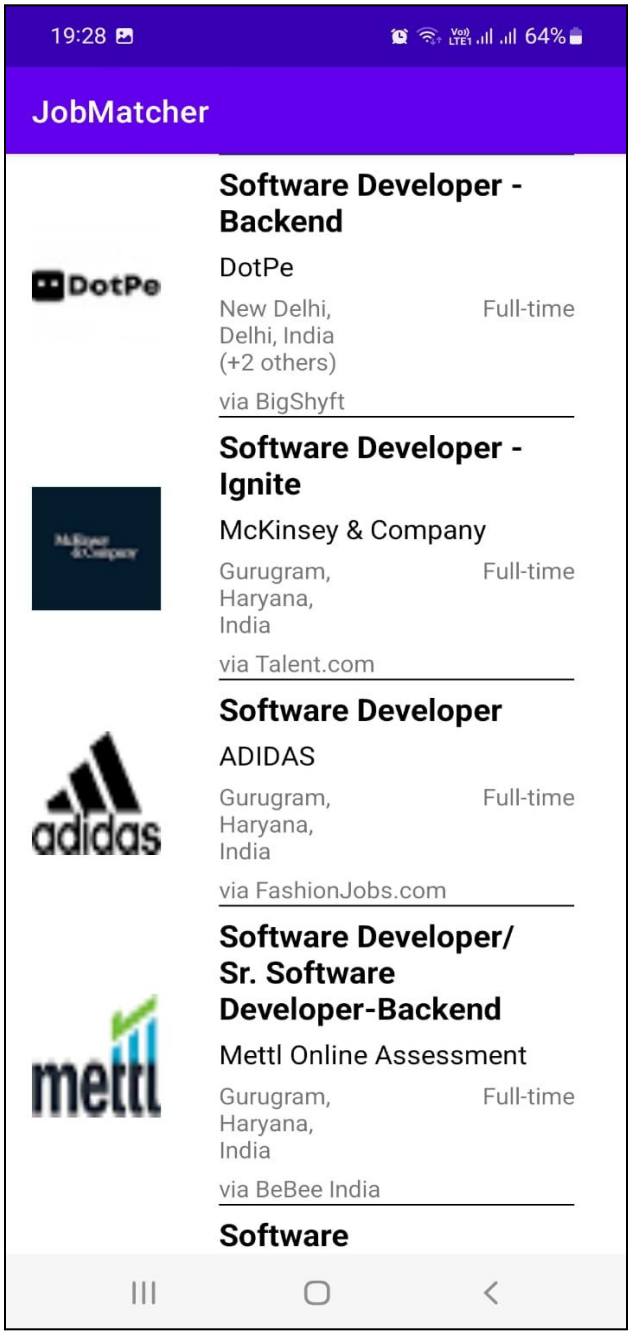
Its UI is defined in activity_spash file which follows a linear layout and also includes an image view of the logo.



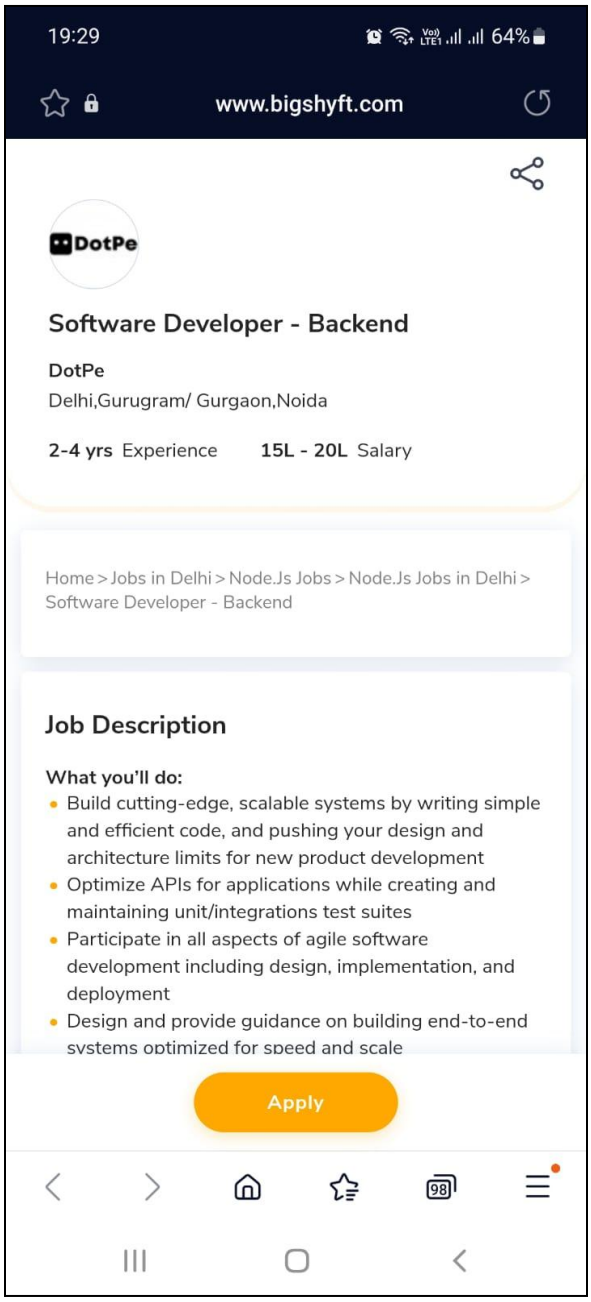
First page of app

Enter details for searching jobs

This is the first page of app which allow user to enter the role, location and type of job one is looking for.



List of jobs



Redirect to application page of job

Once we click the search job button on the previous page, we get a list of recent job openings. On clicking any job, it redirects to the apply link of that job as shown in right hand side image.

FUTURE SCOPE

- Job Matchers future scope is to provide every user the best matching jobs related to them by gathering more information from user related to job.
- This project can be further enhanced to create a profile of the user. The data related to his education will be gathered through his profile. Then the users will be suggested jobs based on their profiles.
- Another aim is to help users save job openings to check them later. This will help users to check any jobs at a later time.
- Users will be notified and reminded to apply to a job if there is a deadline to apply. This will help them if they forget to apply to a job.
- Moreover, implementing this application in iOS platform as an iPhone App is one of the important considerations as we have many iPhone users.

CONCLUSION

We as a team have learnt a lot from this project on how to develop an Android Application. Also, we explored Google API's. We made use of these APIs to fetch required data. We learnt how to enhance the user interface of an Android App using Material Design.

Through Job Matchers less time will be spent on searching for jobs.So that the person can focus only on those job openings which are relevant to him/her. The person will then be provided with the details of his area of interest for the job.

While working on the project, we learnt on how to collaborate with each other and manage the tasks to be completed for the project. We will continue to work on the application in future to increase its functionality. So that Job Matchers is able to find more reliable and relevant jobs to the user.

REFERENCES

- Android Official Documentation
  https://developer.android.com/guide/components/fundamentals
- Android Development Course
  - User Interface:
    https://www.udacity.com/course/android-basics-user-interface--ud834
  - User Input:
    https://www.udacity.com/course/android-basics-user-input--ud836
  - Multiscreen Apps:
    https://www.udacity.com/course/android-basics-multiscreen-apps--ud839
  - Networking course:
    https://www.udacity.com/course/android-basics-networking--ud843
- Material Design:
  https://www.udacity.com/course/material-design-for-android-developers--ud862
- APIs:
  - https://serpapi.com/google-jobs-api
  - https://serpapi.com/google-jobs-listing-api

Department of Computer Science and Engineering

UIET, Panjab University, Chandigarh

Course Exit Survey

Dear Student

The attainment of course outcome after the completion of the course is required as it would help in the continuous improvement of Course Outcomes (CO). This course exit survey would enable us to know as to what extend the subject under consideration and the teaching methodology that have been practiced in the institution have contributed towards the attainment of course outcomes. Hence you are asked to provide the attainment level on scale of Very High (5), High (4), Medium(3), Satisfactory (2), Low (1) for the given course outcomes.

Name of the student: Muskan Mittal **Roll No:** UE193075	**Year/Semester** **7th Semester (December 2022)**	**Academic Year 2022 - 2023**
Course Name:	**Project - I**	
Course Code:	**CS - 757**	
Teacher Name:	Dr. Sarbjeet Singh	

Course Outcomes		**Course Outcome Attainment**				
		Very High (5)	**High (4)**	**Medium (3)**	**Satisfactory (2)**	**Low (1)**
CO1	Apply the knowledge from previous semesters to		✔			

	undertake and solve a real-life problem					
CO2	Illustrate the solution after identifying various objectives of the problem undertaken		✔			
CO3	Devise an organized action plan along with all the team members		✔			
CO4	Develop a solution using the appropriate methodology and tools available			✔		
CO5	Communicate and demonstrate the work through the structured report and oral presentation		✔			

Suggestions for Improvement

_____

_____

_____

Muskan Mittal

Signature of Student

Department of Computer Science and Engineering

UIET, Panjab University, Chandigarh

Course Exit Survey

Dear Student

The attainment of course outcome after the completion of the course is required as it would help in the continuous improvement of Course Outcomes (CO). This course exit survey would enable us to know as to what extend the subject under consideration and the teaching methodology that have been practiced in the institution have contributed towards the attainment of course outcomes. Hence you are asked to provide the attainment level on scale of Very High (5), High (4), Medium(3), Satisfactory (2), Low (1) for the given course outcomes.

Name of the student: Nancy **Roll No:** UE193077	**Year/Semester 7th Semester (December 2022)**	**Academic Year 2022 - 2023**
Course Name:	**Project - I**	
Course Code:	**CS - 757**	
Teacher Name:	Dr. Sarbjeet Singh	

Course Outcomes		**Course Outcome Attainment**				
		Very High (5)	**High (4)**	**Medium (3)**	**Satisfactory (2)**	**Low (1)**
CO1	Apply the knowledge from previous semesters to			✔		

	undertake and solve a real-life problem					
CO2	Illustrate the solution after identifying various objectives of the problem undertaken		✔			
CO3	Devise an organized action plan along with all the team members			✔		
CO4	Develop a solution using the appropriate methodology and tools available		✔			
CO5	Communicate and demonstrate the work through the structured report and oral presentation		✔			

Suggestions for Improvement

_____

_____

_____

Nancy

Signature of Student

Department of Computer Science and Engineering

UIET, Panjab University, Chandigarh

Course Exit Survey

Dear Student

The attainment of course outcome after the completion of the course is required as it would help in the continuous improvement of Course Outcomes (CO). This course exit survey would enable us to know as to what extend the subject under consideration and the teaching methodology that have been practiced in the institution have contributed towards the attainment of course outcomes. Hence you are asked to provide the attainment level on scale of Very High (5), High (4), Medium(3), Satisfactory (2), Low (1) for the given course outcomes.

Name of the student: Nischay Wadhwa **Roll No:** UE193083	**Year/Semester** **7th Semester (December 2022)**	**Academic Year** **2022 - 2023**
Course Name:	**Project - I**	
Course Code:	**CS - 757**	
Teacher Name:	Dr. Sarbjeet Singh	

Course Outcomes		**Course Outcome Attainment**				
		Very High (5)	**High (4)**	**Medium (3)**	**Satisfactory (2)**	**Low (1)**
CO1	Apply the knowledge from previous semesters to		✔			

	undertake and solve a real-life problem					
CO2	Illustrate the solution after identifying various objectives of the problem undertaken		✔			
CO3	Devise an organized action plan along with all the team members			✔		
CO4	Develop a solution using the appropriate methodology and tools available		✔			
CO5	Communicate and demonstrate the work through the structured report and oral presentation		✔			

Suggestions for Improvement

_____

_____

_____

Nischay Wadhwa

Signature of Student