

“Tour Takers”: A GUIDE HIRING APPLICATION

Submitted By

Md. Imran Hossain

172-15-10109

S.M. Forhad Hossain

172-15-10117

Prinon Bhowmik

172-15-9977

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering.

Supervised By:

Ms. Mousumi Zaman Bonny

Sr. Lecturer

Daffodil International University

Co-Supervised By:

Ms. RefathAraHossain

Lecturer

Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

Dhaka, Bangladesh

September 2020

APPROVAL

This Project titled **Tour Takers**, submitted by **Md. Imran Hossain** and **S.M. Forhad Hossain** and **Prinon Bhowmik** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering (BSc) and approved as to its style and contents. The presentation has been held on *08/10/2020*.

BOARD OF EXAMINERS



Dr. Syed Akhter Hossain

Chairman

Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



Nazmun Nessa Moon

Internal Examiner

Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



Gazi Zahirul Islam

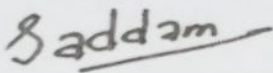
Internal Examiner

Assistant Professor

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University



Dr. Md. Saddam Hossain

External Examiner

Assistant Professor

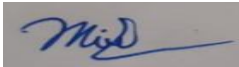
Department of Computer Science and Engineering

United International University

DECLARATION

I hereby declare that, this project has been done by us under the supervision of **Ms. Mousumi Zaman Bonny, Sr. Lecturer, Department of CSE** Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere forward of any degree or diploma.

Supervised by:



Ms. Mousumi Zaman Bonny
Sr. Lecturer
Daffodil International University

Co-Supervised by:

Ms. RefathAraHossain
Lecturer
Daffodil International University

Submitted by:

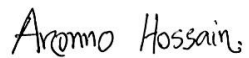


Md Imran Hossain

ID: -172-15-10109

Department of CSE

Daffodil International University

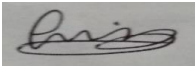


S.M. Forhad Hossain

ID: -172-15-10117

Department of CSE

Daffodil International University



Prinon Bhowmik

ID: -172-15-9977

Department of CSE

Daffodil International University

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to almighty God for his divine blessing make us possible to complete the final year project successfully.

We are grateful and wish my profound my indebtedness to **Ms. Moushumi Zaman Bonny**, Sr. Lecturer Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field “Android Development” to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Prof. Dr. Syed Akhter Hossain, Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

Finally, we must acknowledge with due respect the constant support and patients of my parents, without them any of our work wouldn't happen.

ABSTRACT

In today's fast-changing business environment, it's extremely important to be able to respond to everyone in the most effective and timely manner. If any tourists wish to get tourism service by online system and have service within specific range of amount and higher security by local guides.

Tour Takers is a guide sharing android application, which provides various registered guides in searched locality and pay by cash exact amount considered by system. This project also provides weather updates of selected location for any kind of weather obstacles and also provides hourly forecast for 7days.

In order to develop a guide hiring app, a number of Technologies must be studied and understood. These include multi-tiered architecture, cloud server and real-time database handling, implementation technologies such as **JAVA**, programming language (**Android Studio**) and real-time databases **Firestore**, many third-party libraries such as **Google Map, Retrofit, Picasso, Lottie Animation**. This is a project with the objective to develop a basic android application where a guide is provided with light weight android app and also to know about the technologies used to develop such an application.

TABLES OF CONTENTS

CONTENTS	PAGE
Approval	ii
Board of examiners	ii-iii
Declaration	iv-v
Acknowledgements	vi
Abstract	vii
CHAPTER	
Chapter 1: Introduction	1
1.1 Introduction	1
1.2 Motivation	1
1.3 Objective	1
1.4 Expected Outcome	2
Chapter 2: Background	2
2.1 Introduction	2
2.2 Related Worked	2
2.3 Comparative Studies	3
2.4 Scope of the problem	3
2.5 Challenges	3
Chapter 3: Design Specification	4-49
3.1 Front-end Design and Details (Tour Taker)	4-29
3.2 Front-end Design and Details (Tour Takers for Guide)	30-42
3.3 Back-end Design and Code	43-45
3.4 Data Base	46-49
3.5 Implementation Requirement	49
Chapter 4: Conclusion and Future Scope	50
4.1 Discussion	50
4.2 Future Scope	50

LIST OF FIGURES

FIGURES	PAGE NO
Figure 3.1.1: Sign-Up Page	4
Figure3.1.2: First-time sign-up	5
Figure 3.1.3: Checking e-mail	5
Figure 3.1.4: Sing up page	6
Figure 3.1.5: Enter all Information	6
Figure 3.1.6: Signing up	7
Figure 3.1.7: E-mail not verify yet	7
Figure 3.1.8: Verifying e-mail address.	8
Figure 3.1.9: Verified email address	8
Figure 3.1.10: Main Activity	9
Figure 3.1.11: Navigation Bar	10
Figure 3.1.12: User profile and edit option	11
Figure 3.1.13: Edit user name.	11
Figure 3.1.14: Uploading User Profile Image.	12
Figure 3.1.15: Uploaded User Image	12
Figure 3.1.16: Select Tourist District	13
Figure 3.1.17: Selected place image show	13
Figure 3.1.18: Short Information of places	14
Figure 3.1.19: Selecting tourist places	14
Figure 3.1.20: Select start and end date	15
Figure 3.1.21: Show weather	15
Figure 3.1.22: Select meeting location	16
Figure 3.1.23: Show location	16
Figure 3.1.24: Finding Guide	17
Figure 3.1.25: Tour Crate	17
Figure 3.1.26: Guide profile details	18
Figure 3.1.27: Guide message box	18

Figure 3.1.28: Event Fragment	19
Figure 3.1.29: Enter all information	20
Figure 3.1.30: Searching Guide	20
Figure 3.1.31: Going Event Fragment	21
Figure 3.1.32: All event list	21
Figure 3.1.33: Searching event	22
Figure 3.1.34: Select My event option	22
Figure 3.1.35: Recycler View	23
Figure 3.1.36: Event details (1)	23
Figure 3.1.37: Event details (2)	24
Figure 3.1.38: Member list	24
Figure 3.1.39: Event details (Non-admin)	25
Figure 3.1.40: Event details (Non-admin)	25
Figure 3.1.41: Admin details	26
Figure 3.1.42: Join event.	26
Figure 3.1.43: List of members	27
Figure 3.1.44: Comment box for members	27
Figure 3.1.45: Going to reply box	28
Figure 3.1.46: Reply message	28
Figure 3.1.47: Going weather	29
Figure 3.1.48: Weather fragment	29
Figure 3.2.1: Guide Sign-Up Page	30
Figure 3.2.2: First-time sign-up	31
Figure 3.2.3: Checking e-mail	31
Figure 3.2.4: Sing up page	32
Figure 3.2.5: Signing up	33
Figure 3.2.6: E-mail not verify yet	33
Figure 3.2.7: Verifying e-mail address	34
Figure 3.2.8: Verified email address	34

Figure 3.2.9: Location Permission	35
Figure 3.2.10: Battery Optimization	35
Figure 3.2.11: Always run permission	36
Figure 3.2.12: Allow display	36
Figure 3.2.13: Guide is Offline	37
Figure 3.2.14: Guide is Online	37
Figure 3.2.15: Navigation Bar	38
Figure 3.2.16: Guides Profile	38
Figure 3.2.17: Guide profile edit option	39
Figure 3.2.18: New Request Coming	40
Figure 3.2.19: Accepted Tour	40
Figure 3.2.20: Tourist Details	41
Figure 3.2.21: Message box for guide	41
Figure 3.2.23: Going Weather Fragment	42
Figure 3.2.24: Guide Fragment Activity	42
Figure 3.3.1: Tour Takers Main Activity Class (1)	43
Figure 3.3.2: Tour Takers Main Activity Class (2)	43
Figure 3.3.3: XML Code and design	44
Figure 3.3.4: Main Class of Tour Takers Guide app	45
Figure 3.3.5: Sign Up Layout of Tour Takers Guide	45
Figure 3.4.1: Firebase Database	46
Figure 3.4.2: User Profile Database	47
Figure 3.4.3: Guide Profile Database	48
Figure 3.4.4: Event Details Database	48

Chapter 1: Introduction

1.1 Introduction:

As an acknowledged and used market model, app-based system is rapidly gaining ground. Increasing business strategies are introducing a app platform that offers tools for guide hiring and manage group tour through android application.

The goal of this project is to build a tourist helping app platform where tourists can manage events to less the cost of tour by adding more members to the event and get guide services within specific charges.

1.2 Motivation:

Uber is a best example of app-based business in this era. Uber is helping many drivers in Bangladesh by earn some money by using their system. Many people are using uber now-a-days as their first faithful choice. Uber is the best ride-sharing platform in our country. We have taken the concept of uber for helping tourists from our country and foreigners by choosing the right place and way and right location charge.

1.3 Objectives:

The key goals of the tour takers growth are as follows:

1. Details of various tourism place in various locations where you want to visit.
2. Weather and hourly forecast for 7days of the selected location.
3. Tourist will login and create event or tour and search guide for searched place.
4. Therefore, the tour takers must have manual transactions, and tourists can pay manually.
5. Guides have to login with verified email id and NID card for security purpose.

1.4 Expected Outcome:

1. The main goal of this project is to help tourists by providing guides and manage group tour and guides will income some moneys by their service.
2. It will help tourists to navigate the locations they want to go by safe guide service.
3. Registered tourists can easily create event and tour.
4. Registered guides add their location and get request calls from tourists.
5. Guides should be verified with G-mail and NID card.

Chapter 2: Background

2.1 Introduction:

We are living in a world of creation. We have to stick with the world today. But people want to get tourism service online now, as well. There are many approaches to get this. Most of people of the world in this world are educated, they know about technology. Day by Day people are dependent on online. For this motivation we take a chance to make it easy for all over the world.

Our tour takers app-platform aims to make tourist's life easier and feel free to visit unknown places with team and local guideline. User can search any kinds of places in Bangladesh and get desired services.

2.2 Related Works:

It is a fascinating experience to our nation that many programmers have already produced a limited number of similar websites and mobile applications that are familiar to other people. There are some websites such as "[Couchsurfing](#)" and "[ToursByLocals](#)" where you can get guide and place information. The customer will look for all places in Bangladesh they want by using our system. People payment on cash at the end of guide service. [2],[3]

2.3 Comparative Studies:

There is some app related guiding hiring system in Bangladesh. Those apps are helpful for people. We think we want to offer these features along with additional features, such as weather updates and real-time mapping categories. People easily find places in our system. People easily get guides by our system.

We tried to create a lovely app that would be helpful to many people. Users will find details of about us. The UI design of the app would allow only the simple and required steps to perform a task that will be more effective

2.4 Scope of the Problem:

Due to financial sponsor problem. We couldn't add any payment gateway system. For corona situation we couldn't manage to contact any guides to get the actual rent of places people mostly visited.

2.5 Challenges:

Responsive mapping system is not free. So, we had to use paid Google Map API. For making a responsive design we researched many libraries for Android UI. We could manage to use some in our app.

1. Creating user friendly app with light weight is not easy for Androidx materials.
2. Firebase real-time database is easy. But cloud messaging service was light bit difficult.
3. We couldn't visit as much as places to test our app. We built it by testing purpose.
4. Most of the tourism apps in Bangladesh just provides the place details and routes. But tried to make some innovative change and tried to follow some features and concepts of Facebook. Event part of our is app is one of that.

Chapter 3: Design Specification

3.1 Front-end Design and Details (Tour Taker):

1. First install the app to a phone and open the app.
2. Then user will see the front page, where user need to input an e-mail address.

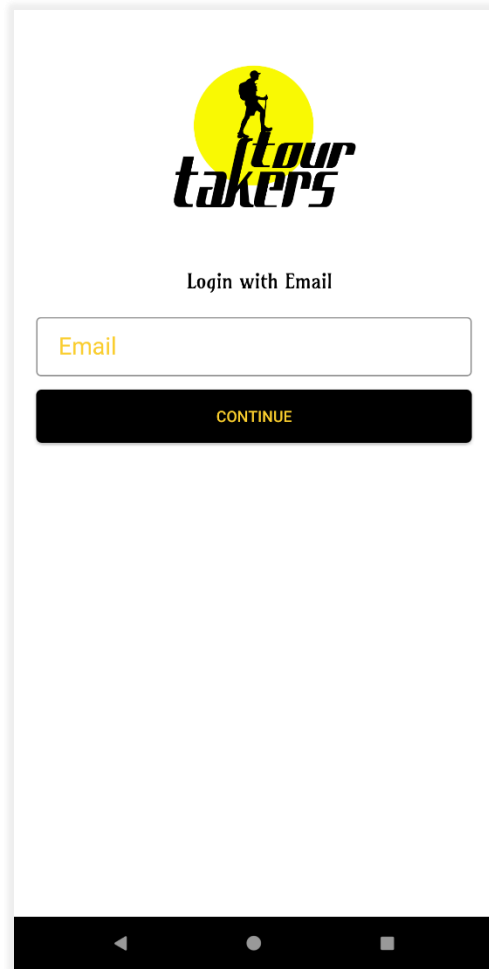


Fig 3.1.1: Sign-Up Page

3. Fig 1 we see that front page. Where user enter an e-mail id.

4. After enter the e-mail address(fig:3.1.2), system will check that email id is already stored the server or not.

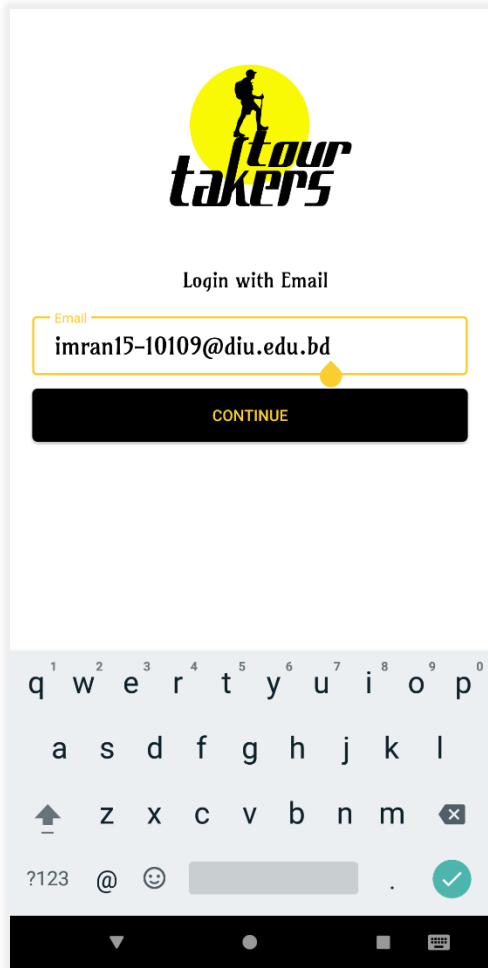


Fig 3.1.2: First-time sign-up

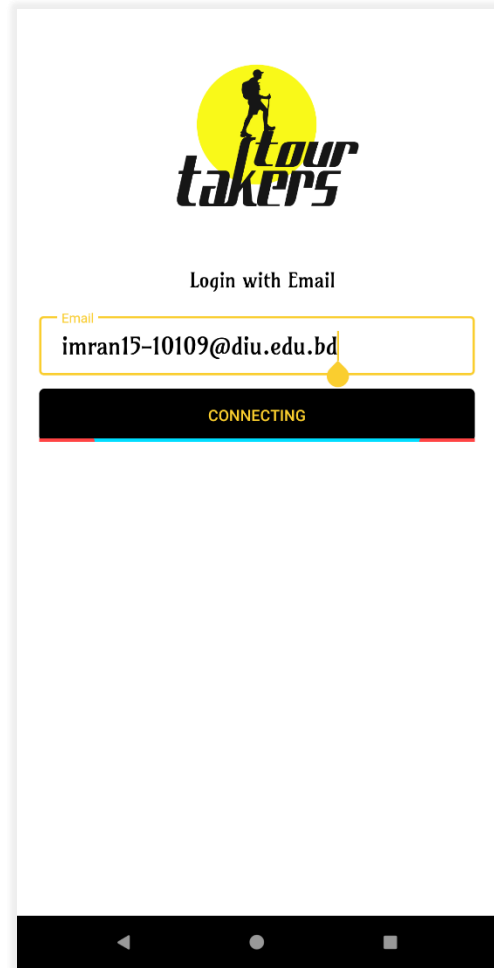
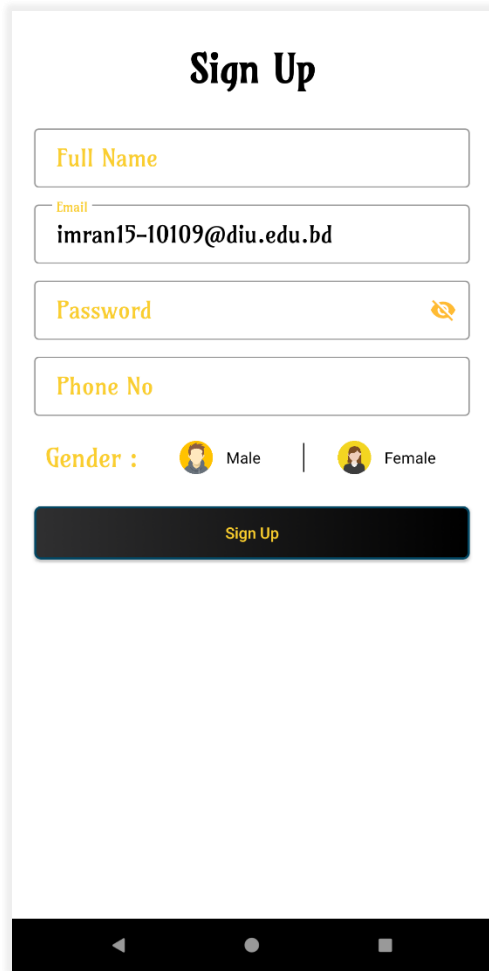


Fig 3.1.3: Checking e-mail

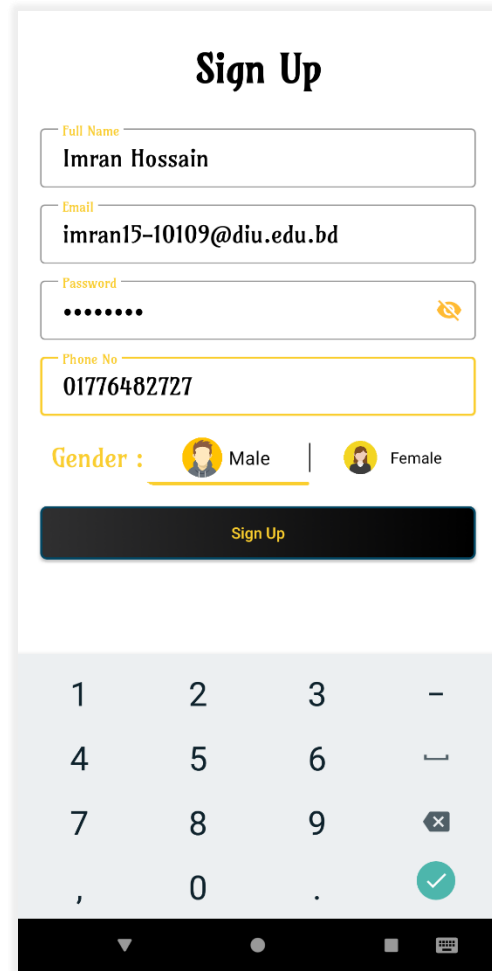
5. If the user is not registered then system will ask user to Register and show the Sign-Up activity (Fig: 3.1.3).

6. Fig:3.1.4 we can see that sign up page. User need to file all the information.
7. Fig:3.1.5 User need to enter his/her name, email, phone no, gender and very important password.



The screenshot shows a mobile application interface for a sign-up page. At the top, the title "Sign Up" is centered. Below the title are four input fields: "Full Name", "Email", "Password", and "Phone No". The "Email" field contains the text "imran15-10109@diu.edu.bd". Below the input fields is a "Gender" section with two radio button options: "Male" and "Female". At the bottom of the form is a black button with the text "Sign Up".

Fig 3.1.4: Sing up page



The screenshot shows the same mobile application interface as Fig 3.1.4, but with all fields filled. The "Full Name" field contains "Imran Hossain", the "Email" field contains "imran15-10109@diu.edu.bd", the "Password" field is masked with dots, and the "Phone No" field contains "01776482727". The "Gender" section is the same. A numeric keypad is visible at the bottom of the screen, indicating that the user is in the process of entering the phone number.

Fig 3.1.5: Enter all Information

8. After complete this registration part, when user will click the signup button a verify mail will sent to user's mail. Firebase is providing a link to verify the mail id. Before login system will check the status of that link verification. Without verify the e-mail id user can't have access to login.

9. In fig: 3.1.6 we see user signing up. After sign up when user enter his password app show a small information toast(fig:3.1.7), where it shows please verify your email id first.

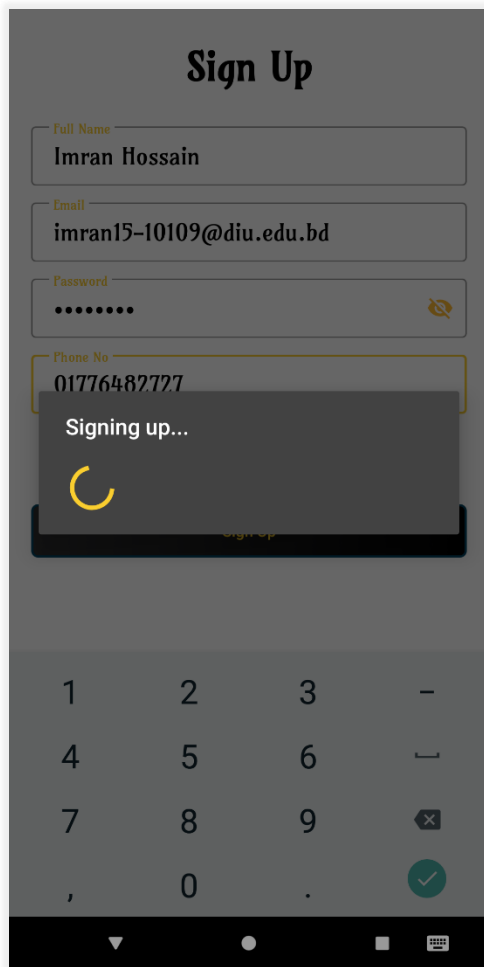


Fig 3.1.6: Signing up

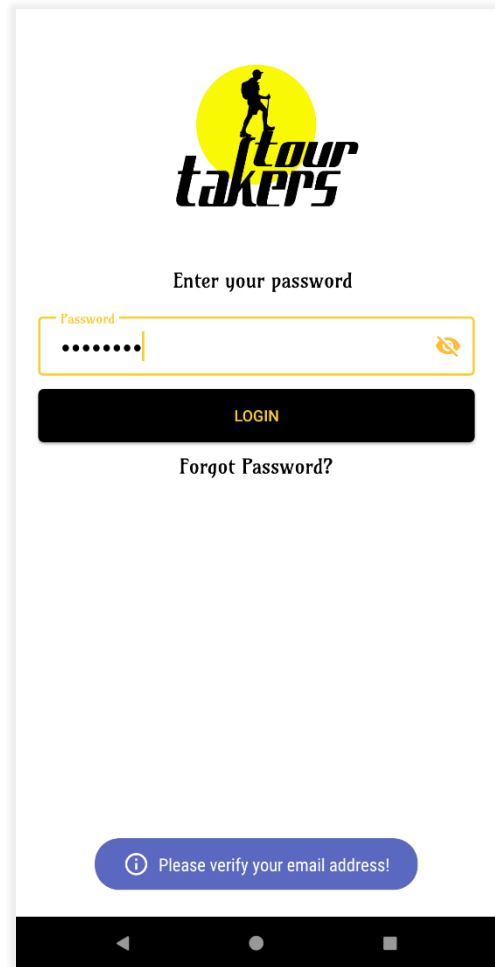


Fig 3.1.7: E-mail not verify yet

10. Then user need to go to his/her email to verify his/her e-mail.

11. System sent an URL to that e-mail. Fig:3.1.8 we see system sent an email with an URL. After click this URL User verify his/her e-mail. Verifying message also show on that email(fig: 3.1.9).

12.

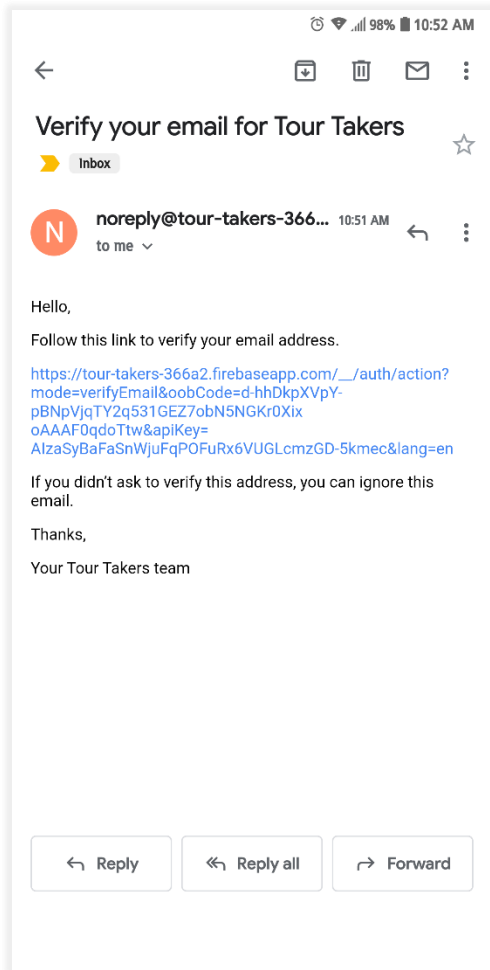


Fig 3.1.8: Verifying e-mail address.

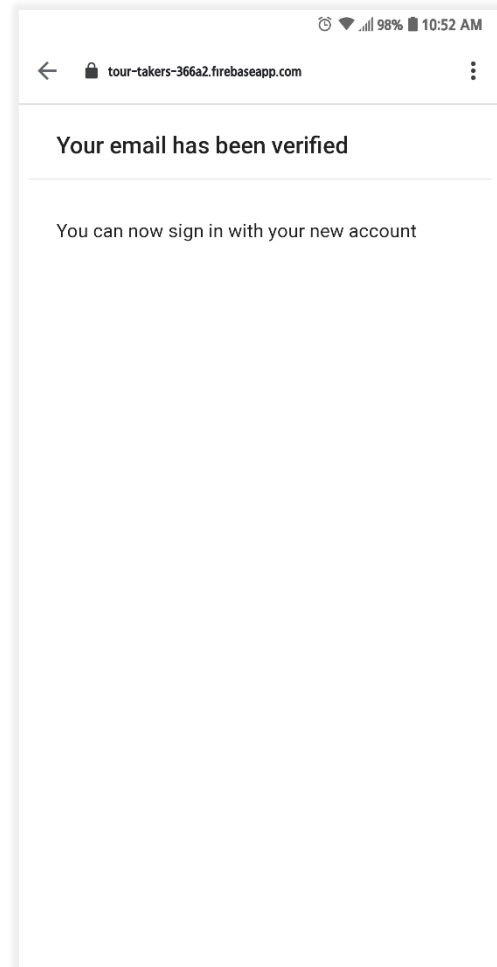


Fig 3.1.9: Verified email address

13. After verifying user go back to **Tour Taker** app.

14. And enter his/her password and Log In. This time user successfully logs in because his/her email id is already verified.

15. After login user will see the **Main Activity** (Fig:3.1.10). There user will see a form for tour.

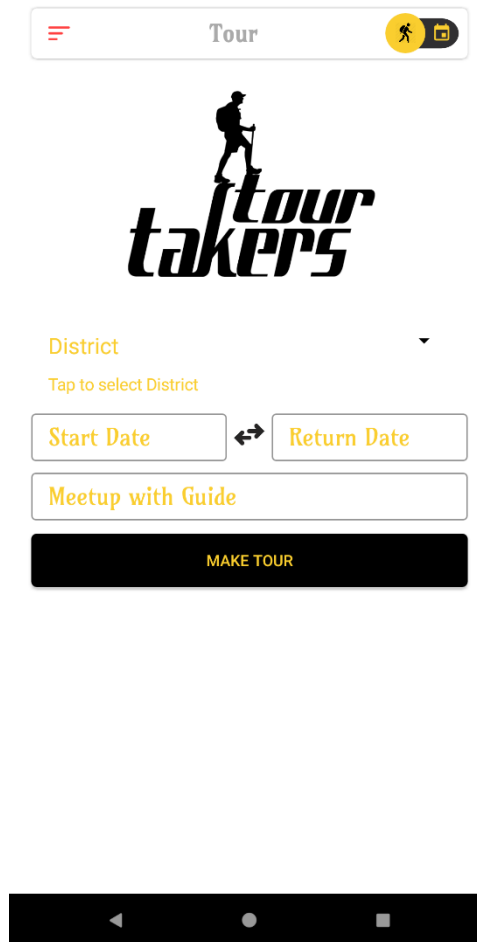


Fig 3.1.10: Main Activity

16. If user want to travel alone or family, he/she will enter his information on this fields.

17. On the top-right of this activity we can see a switch button. If anyone want to travel with a group and want to create an event then he/she will go this event page.

18. On the top-left user can see navigation button (Fig:3.1.11). On click this button user see all menu of this app on navigation bar.
19. Fig 3.1.7 we can see all menu which is Tour, Event, Weather, Terms and condition and Logout added on navigation bar.
20. We also see user short profile on top where user name, e-mail and ratings are added.

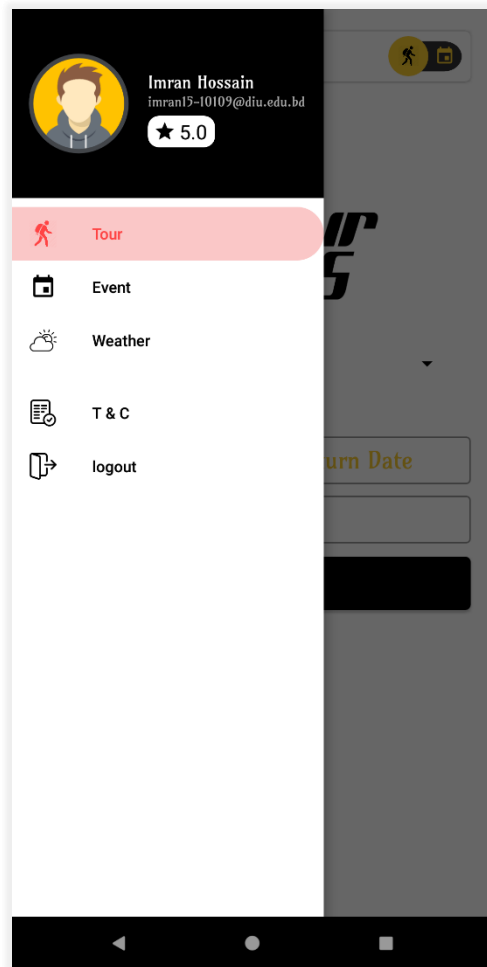


Fig 3.1.11: Navigation Bar

21. If user want to see his/her profile he/she just click the picture icon.

22. Fig 3.1.12 we can see user profile. Here user can see total tour, event and rating. This time user has not completed any tour or event so initially this will be 0. Rating also initially 0.

23. User can all see his/her name email phone number which he/she give on the registration time.

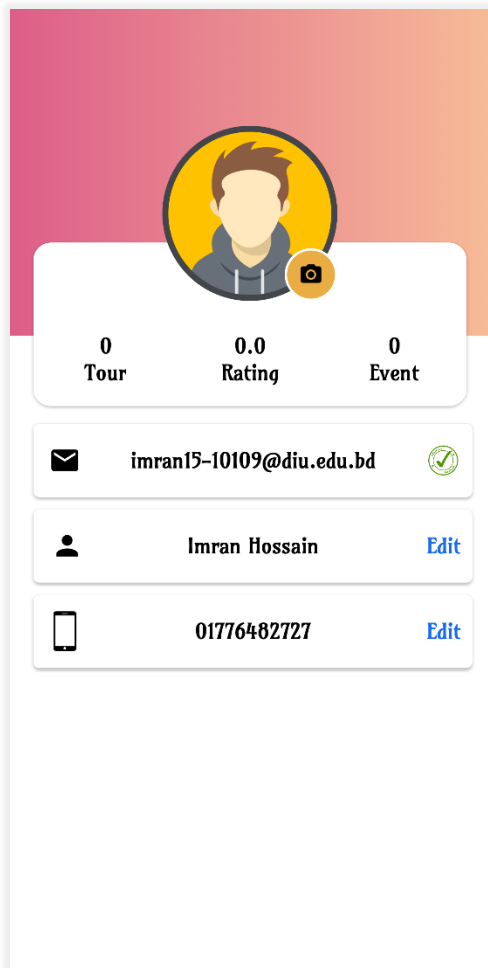


Fig3.1.12: User profile and edit option.

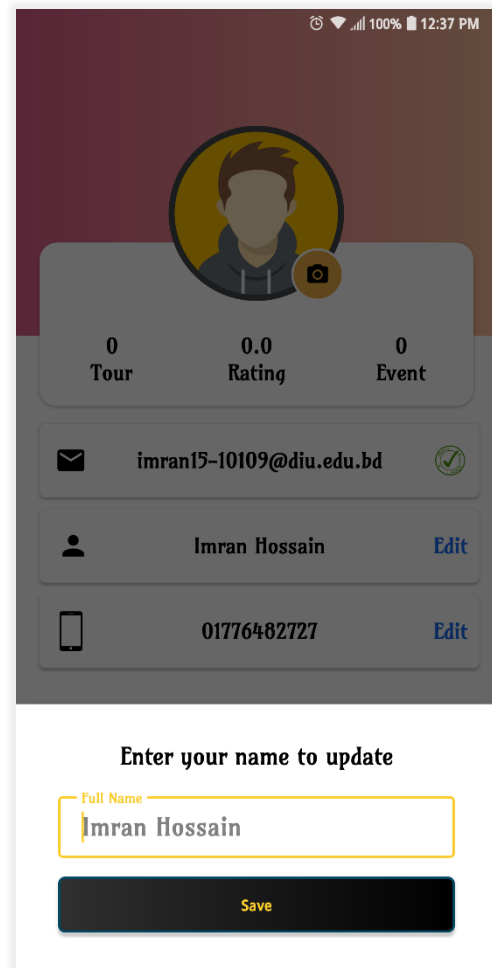


Fig 3.1.13: Edit user name.

24. If user want to edit his name or phone number, he/she can click edit (Fig:3.1.13).

25. After click edit button a bottom sheet will open from bottom to edit name. if user want to change his/her name he/she can edit his name easily.

26. After edit name click the save button it will edit and save this to database.

27. If user want to add a photo of his, he can click the default icon. This default image also set by our app. If any user registered as a male gender it will show a male gender image or user registered as a female gender it will show a default female gender image (Fig:3.1.14).

28. After click this image or small camera icon system will go his media and he can take a photo or upload an image form his phone.

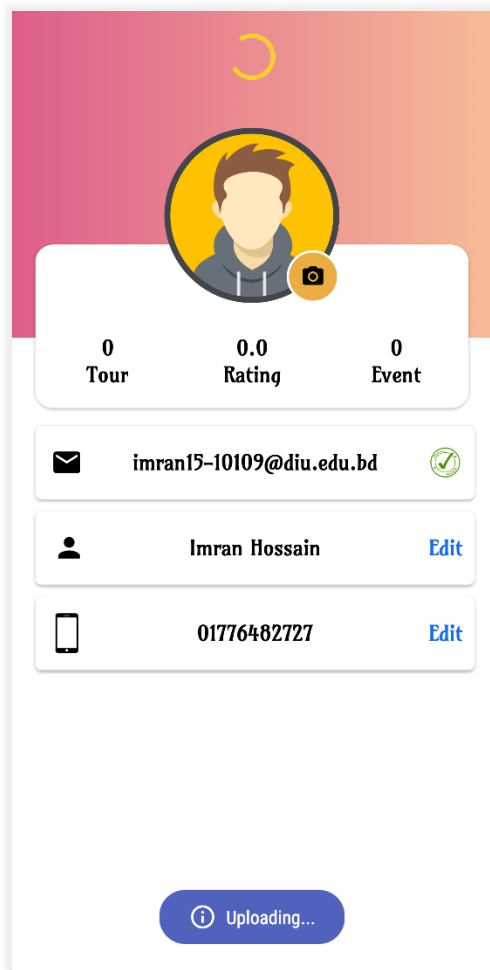


Fig3.1.14: Uploading User Profile Image.

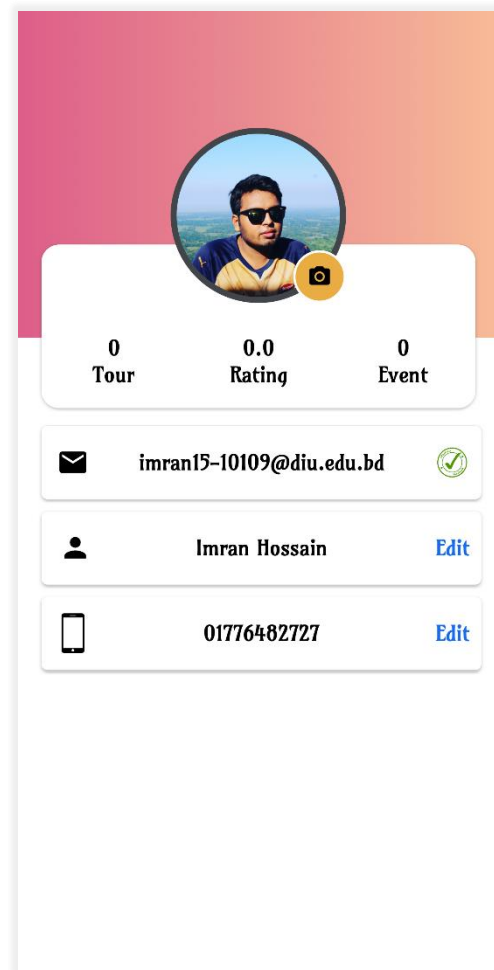


Fig 3.1.15: Uploaded User Image

29. In fig:14 we see image is uploading and finally image upload (Fig:3.1.15).

30. If user want to create a tour, he/she need to go tour activity. Then he/she need to select his tour place.

31. This time he/she will select the district name of the tourist place (Fig:3.1.16).

32. The he can see all the tourist place on selected district with photo. All photos are scrolling on top (Fig:3.1.17).

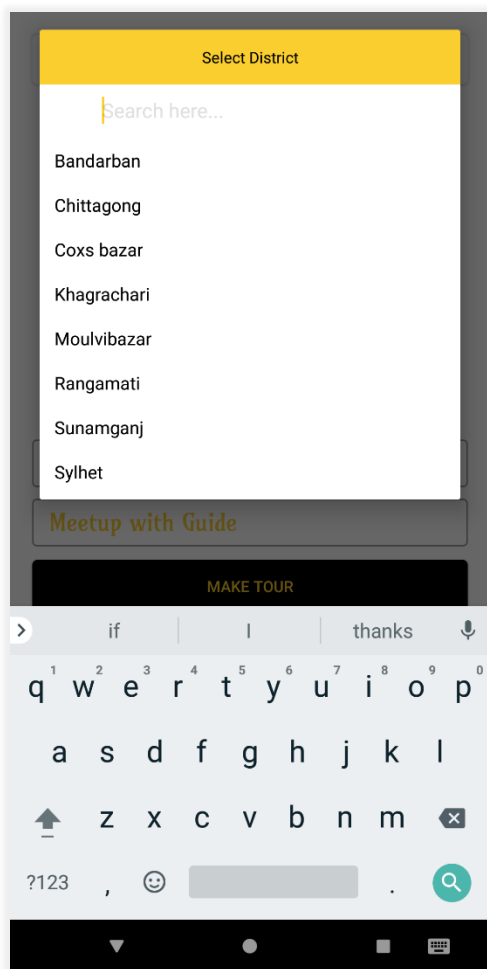


Fig3.1.16: Select Tourist District

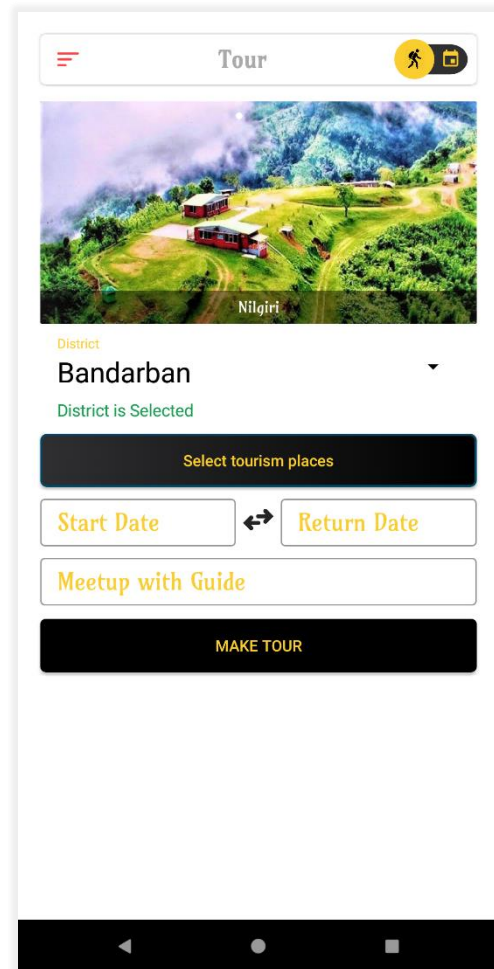


Fig 3.1.17: Selected place image show

33. If user want to know about those tourist place information, he/she will click those images, which are scrolling on top.

34. After click, app will go another activity to show about the information.

35. In fig: 3.1.18 user can see those information's. user can see all tourist place information in selected district.
36. After see those information users can know some knowledge about those places and make a right decision to where he want to travel.
37. Then tourist selected his tourism places (Fig:3.1.19).

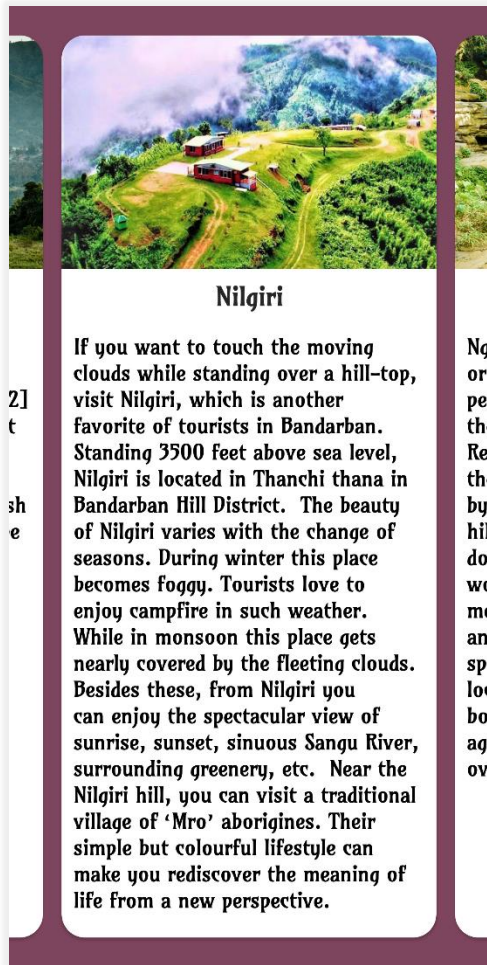


Fig3.1.18: Short Information of places

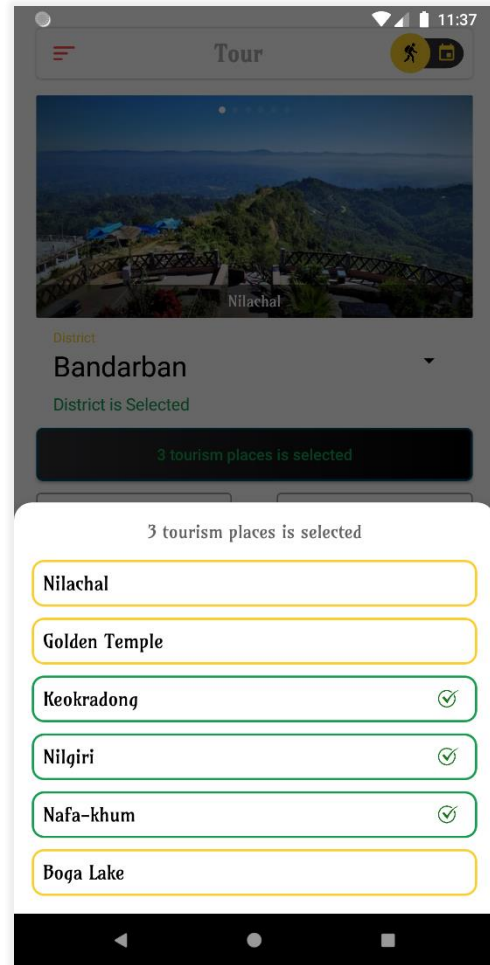


Fig 3.1.19: Selecting tourist places

38. Then user select his tour start date and end date (Fig:3.1.20).
39. After select start date and end date system will show the weather report on this district.
40. The weather report will show only those date when user want to travel (Fig:3.1.21).

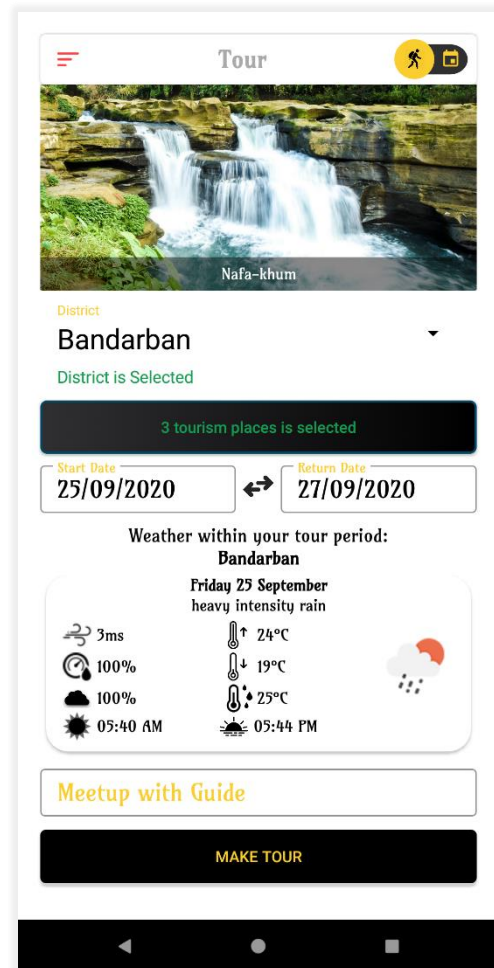
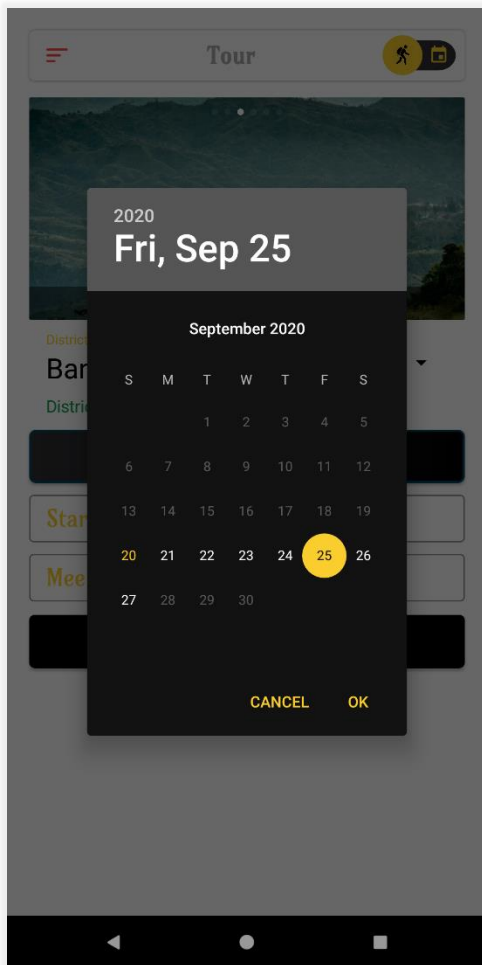


Fig3.1.20: Select start and end date

Fig 3.1.21: Show weather

41. From the weather report user will clarified that his/her travel day's weather are good or bad.
42. We are using there google weather which provide us 7 days weather.

43. Then user select guide meet up place. For that user go to map activity and select where he wants to meet with guide (Fig:3.1.22).

44. After select the meeting place user click the pick location button to confirm his meeting location.



Fig3.1.22: Select meeting location

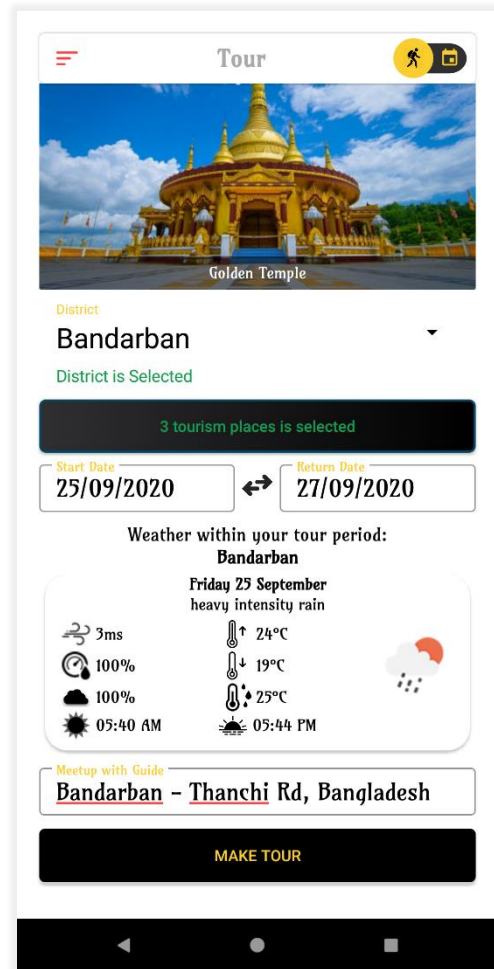


Fig 3.1.23: Show location.

45. After confirm meeting location system will set the location on meet with guide edit text (fig:3.1.23).

46. Then user click make tour. Then system will search guide on that district where user want to travel.
47. We develop a guide app for guide, who will registered as a guide. This request will go those guides who are registered for those districts.
48. If system found a guide in 1 minute than system call those guides. If any guide is accepting this request then this tour will assign for this guide. If nobody is accepting this request than it will show a toast (Fig:3.1.24). But app save this tour without assign. But latter user can request again for guide. In Fig:3.1.25 a guide accepts user request.

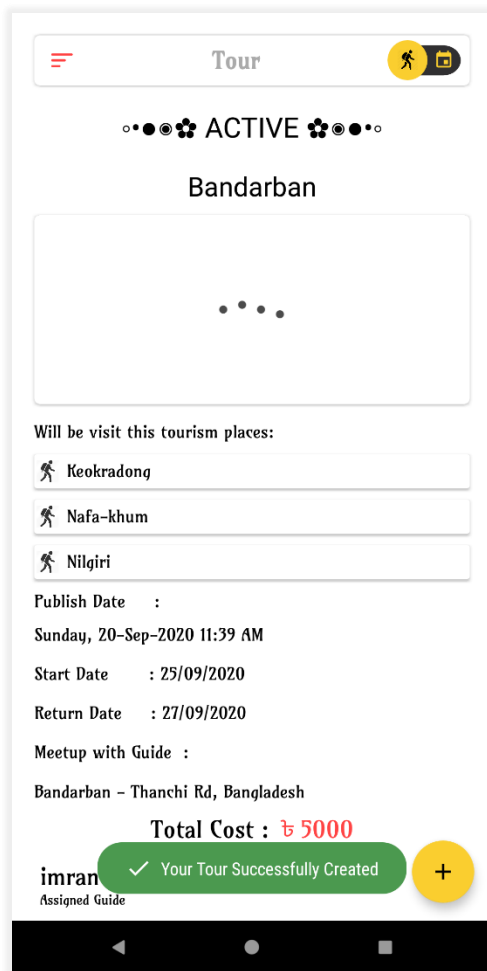


Fig 3.1.24: Finding Guide

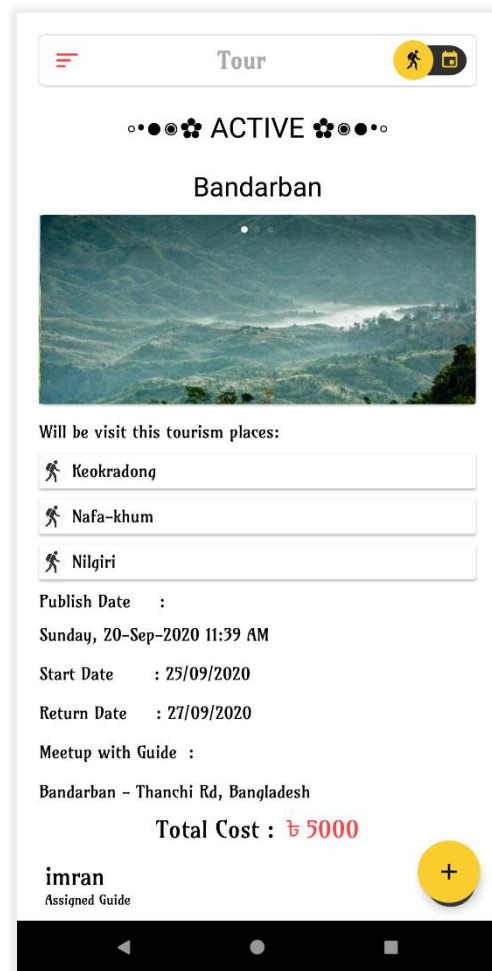


Fig 3.1.25: Tour Crate

49. This tour will show on **Main Activity** until the tour is end.

- 50. In Fig:3.1.25 user can see his/her travel details
- 51. User can see now his/her assign guide profile also.
- 52. User need to click assign guide layout. Then a popup activity show on his/her display (Fig:3.1.26). Here user can see guide profile.

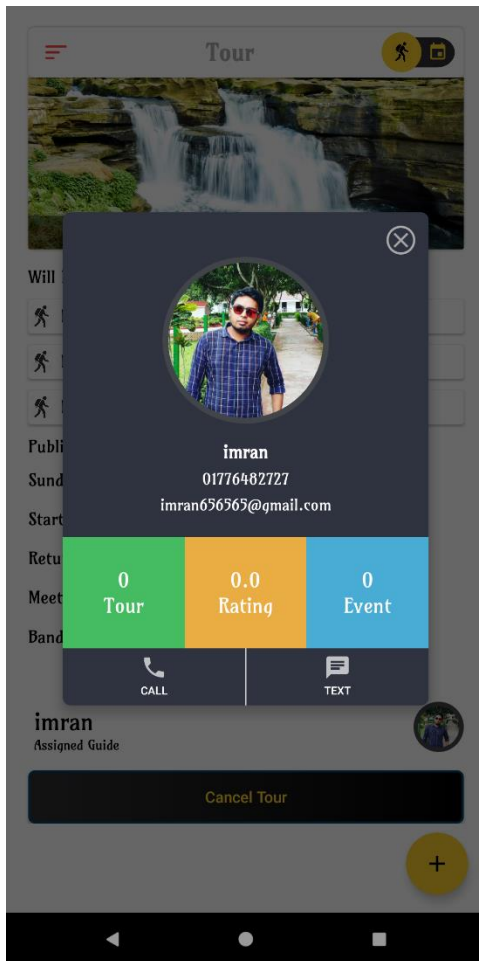


Fig 3.1.26: Guide profile details

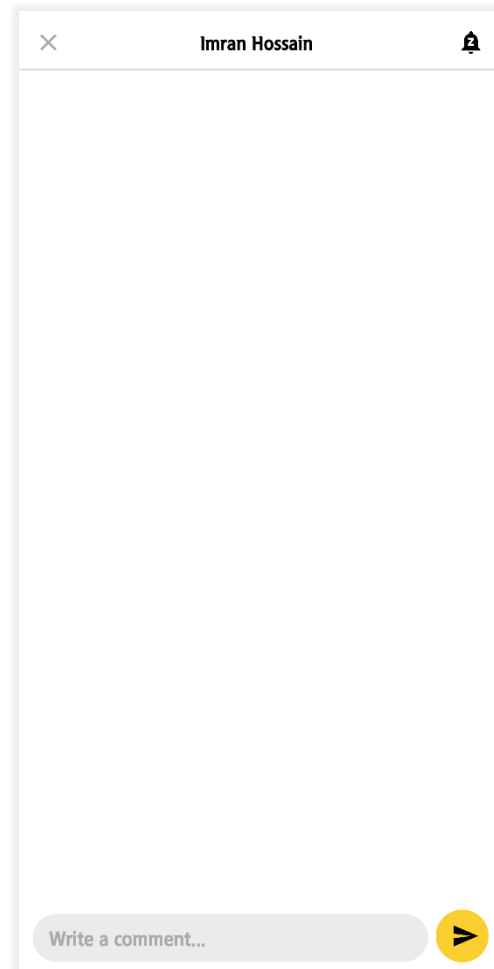


Fig 3.1.27: Guide message box.

- 53. On this popup activity user can see guide name, guide phone number, guide Total completed tour, event and his/her ratings.
- 54. User can also send message or call this guide (Fig: 3.1.27).

55. If user want to Create an event then he/she go to event option.
56. In event user see all field what user need to fill (Fig:3.1.28).
57. In this part user select district name where user travel, start date, end date, meetup with guide place, group name, total travel cost, event start time, meetup place with member and event description.

The screenshot shows a mobile application interface for creating an event. At the top, there is a header with a hamburger menu icon, the word "Event", and two icons: a person and a calendar. Below the header is the "Itour takers" logo, which features a silhouette of a hiker. The main form consists of several sections: a "District" dropdown menu with the instruction "Tap to select District"; two date input fields labeled "Start Date" and "Return Date" connected by a double-headed arrow; a "Meetup with Guide" section containing a "Group Name" input field; two input fields for "Total Cost" and "Event Time"; a "Meetup with Members" section; and an "Event Description" input field. At the bottom of the form is a prominent black button with the text "MAKE EVENT" in yellow. The entire form is set against a white background with a light gray border.

Fig3.1.28: Event Fragment

58. In event fragment user can select travel place like tour fragment. Then system will show the weather report.

59. In fig:3.1.29 user select his/her district and enter all information.
60. After giving all information user click make event button. Then system searching guide on this district where user want to travel (Fig:3.1.30).
61. If any guide accepts this request then it will show user that your guide is assign.
62. Or any guide doesn't accept this request then it will show that no guides are available at this moment, please try letter.

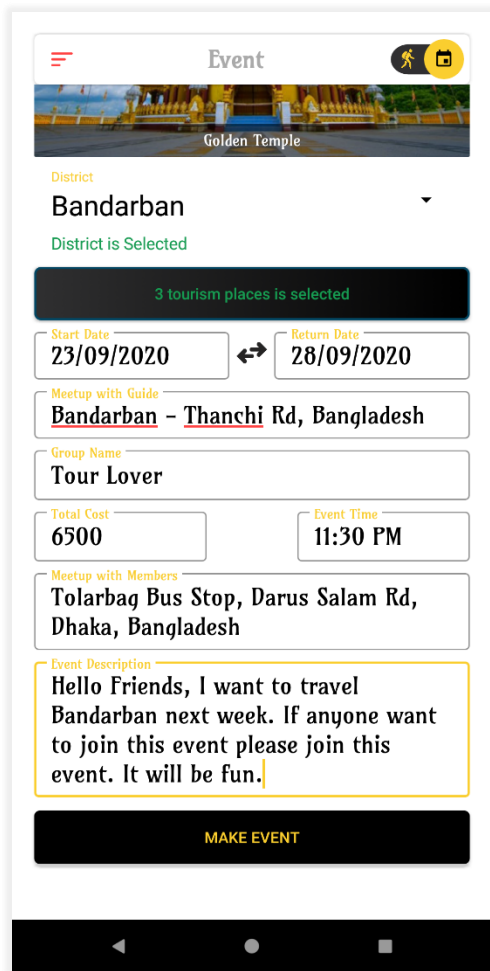


Fig3.1.29: Enter all information

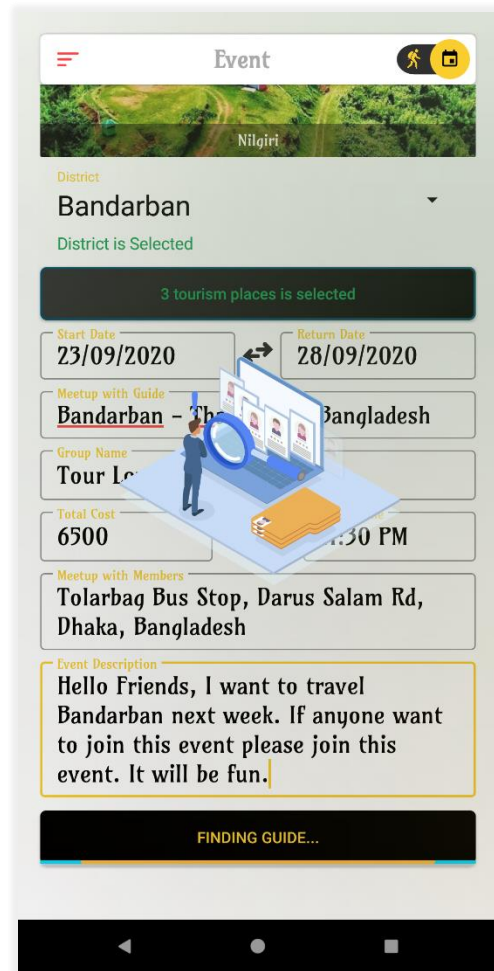


Fig 3.1.30: Searching Guide

63. Then this event will save to database. And this event will show all Tour Taker user, who already a registered member.

64. If any guide accepts this event request then it will show in guide assign part. Here user can see guide all detail. Where user see guide name, phone, email.
65. User can all so call the guide or send a message form this app.
66. To go this event fragment user, need to click navigation icon on top-left and select Event (Fig:3.1.31).
67. Then user will go event fragment where user see his/her event has created.
68. Fig:3.1.32 look like one other user also create an event with this user.

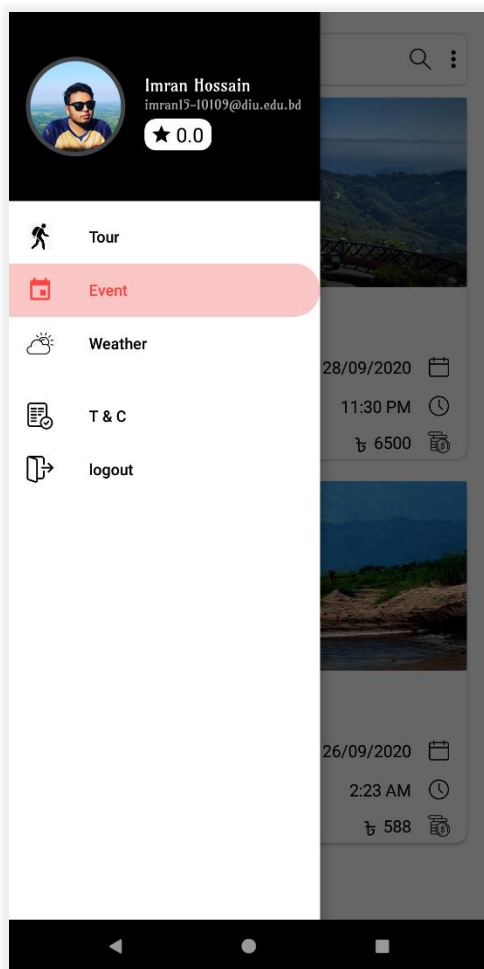


Fig3.1.31: Going Event Fragment

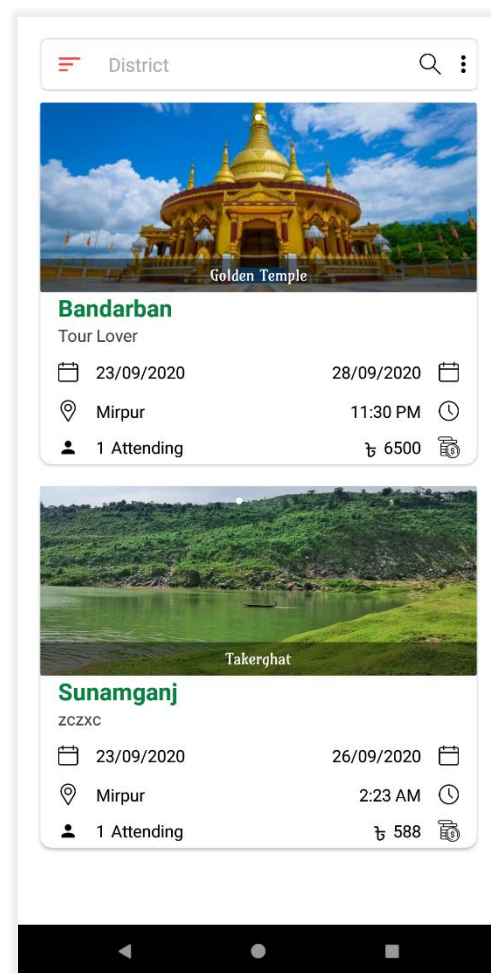


Fig 3.1.32: All event list

69. User can search event by district name or group name. that how user can filter all event list.

70. If user want to show any event which is create for Cox-bazar, user can search the district name on search option.

71. Fig 3.1.33 user search district, where he/she wants to travel.

72. There are also 2 option are available. One is all event, where user see all member event. And one is my event where only show user events, which is created by this user (Fig:3.1.34).

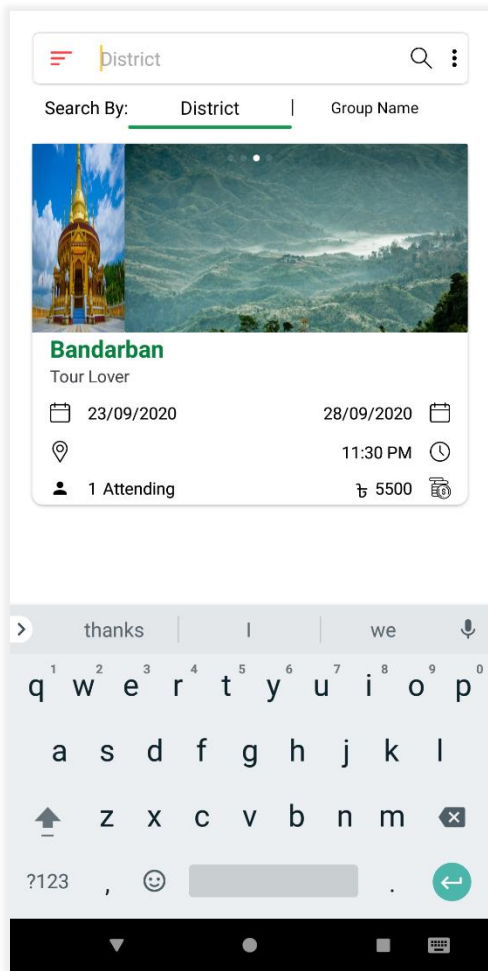


Fig3.1.33: Searching event

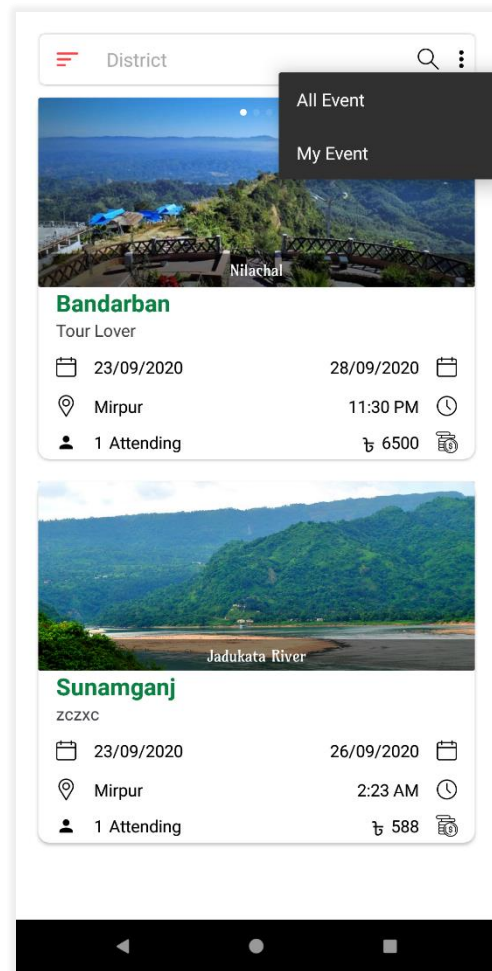


Fig 3.1.34: Select My event option

73. If user want to show only his events then he chooses My event options.

74. Default system will show all events in this fragment.

75. Fig:3.1.35 use sees only his event. In this fragment we use recycler view. There we show a short detail of this event. There we can see district name, group name, event star date, end date, meetup place, event start time, how many members join this event, and cost of this event.

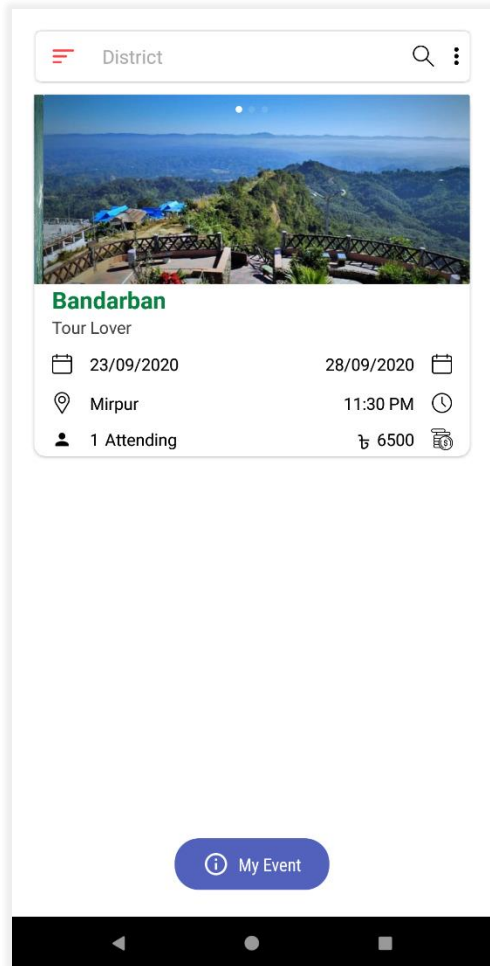


Fig 3.1.35: Recycler View

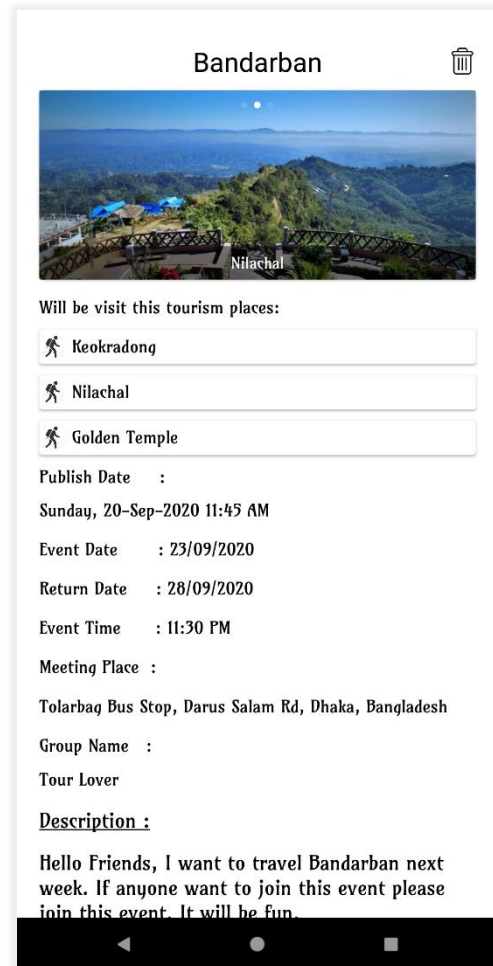


Fig 3.1.36: Event details(1)

76. If user click this recycler view then he can show all details of this event. In fig:3.1.36 we see that user full detail.

77. User can also delete this event if he/she need to. In the top corner we see that delete icon.

78. User can also update every detail if need with edit icon.

79. Fig:3.1.37 we see bottom details, where we can see admin of this event. Total attending member number, total comment.

80. If user click view member then he can see who is join this event. Initially only one member joins this event who is the admin of this event (Fig:3.1.38).



Fig3.1.37: Event details (2)

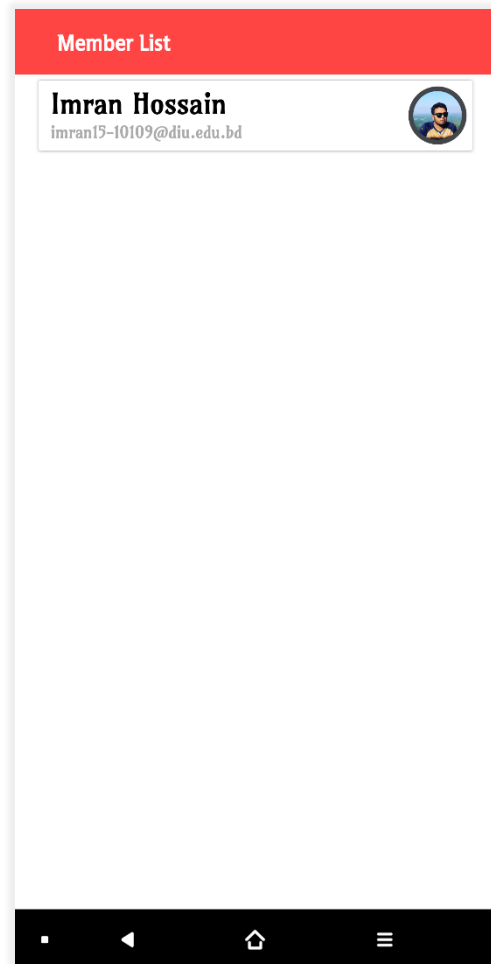


Fig 3.1.38: Member list

81. Now we will see what thing show another user app. when another user wants to see event and want to join, first he needs to go event fragment.

82. Here he can show all events. After see those event he clicks to see full details of that event.

83. Fig:3.1.39 and 3.1.40another user sees all details of that event.

84. Here user can see who is the admin of this event. And he can read all the detail of this event. If he/she want to see admin information, he can click event admin layout.

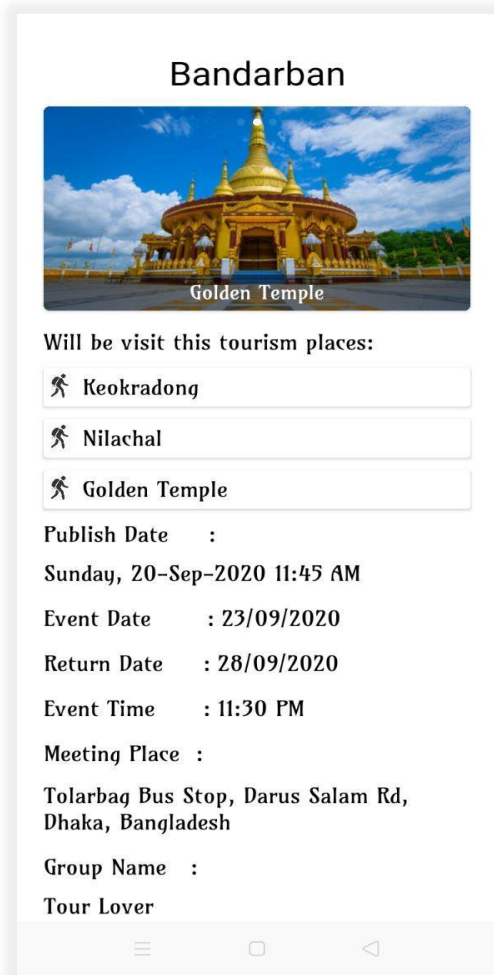


Fig 3.1.39: Event details (Non-admin)

Fig 3.1 40: Event details (Non-admin)

85. Fig:3.1.41 user can see admin profile. There he can show admin name, email, phone number.

86. User can also see admin total complete event number, total complete tour number and admin rating.

87. After see all information, if this user decides to join this event then he will click Join Event button.

88. Then a congratulation popup will come and show that you are now a member of this event (Fig:3.1.42).

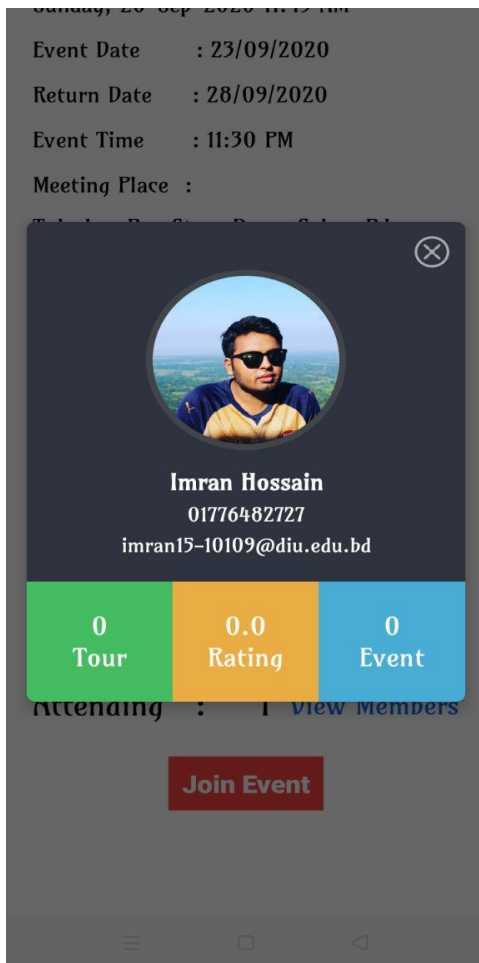


Fig 3.1.41: Admin details

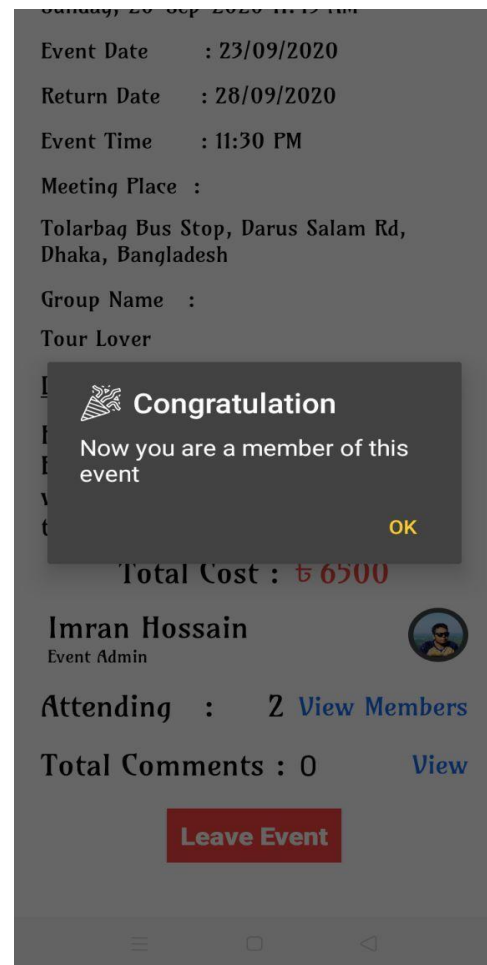


Fig 3.1.42: Join event.

89. Now system will count a new member of this event.

90. If user want to leave this event, he can. But user must leave this event before event start.

91. If event start then no member of this event can't leave this event. Also, admin cannot delete this event.

92. Member can also see all member of the event (Fig:3.1.43).
93. If user want to comment for this event, then he needsto go comment box and comment anything.
94. Fig 3.1.44 a member comments for knew some information about the event. And admin of this event is also comment.
95. If any want comment then all member of that event will also get notification.

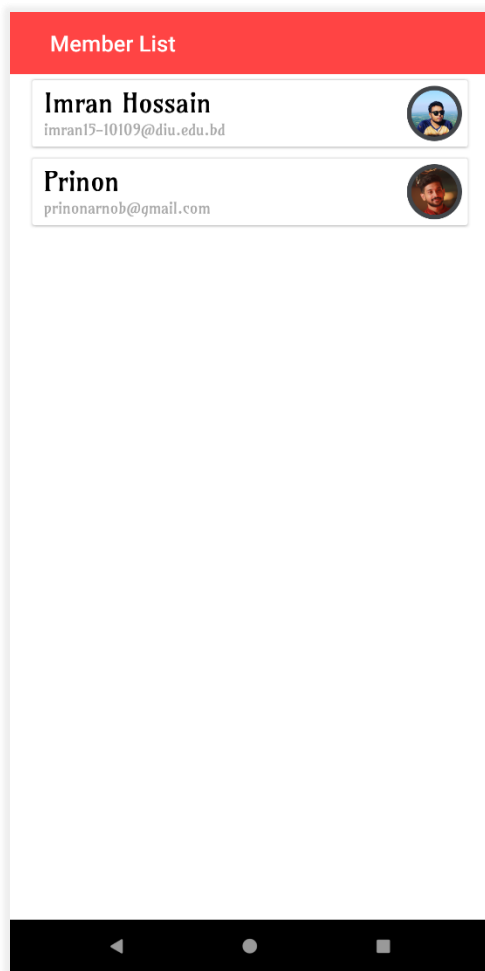


Fig 3.1.43: List of members

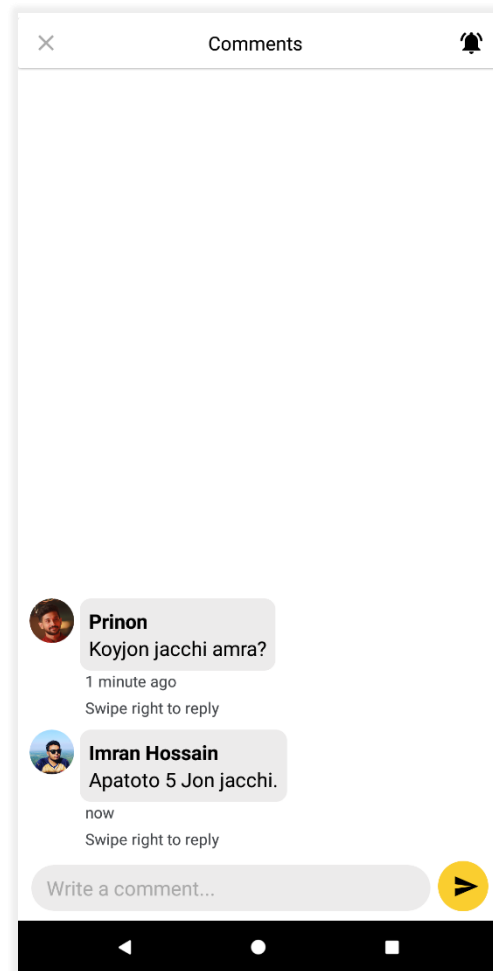


Fig 3.1.44: Comment box for members

96. If anyone wants to mute notification then, he will click notification icon and disable notification. Then this member doesn't get any notification. Member can also reply of any comment if he wants.

97. Fig:3.1.45 member is going to reply box with swipe the comment which he wants to reply. Fig:3.1.46 user reply a comment. If user want to comment any GIF file it also possible to out app.

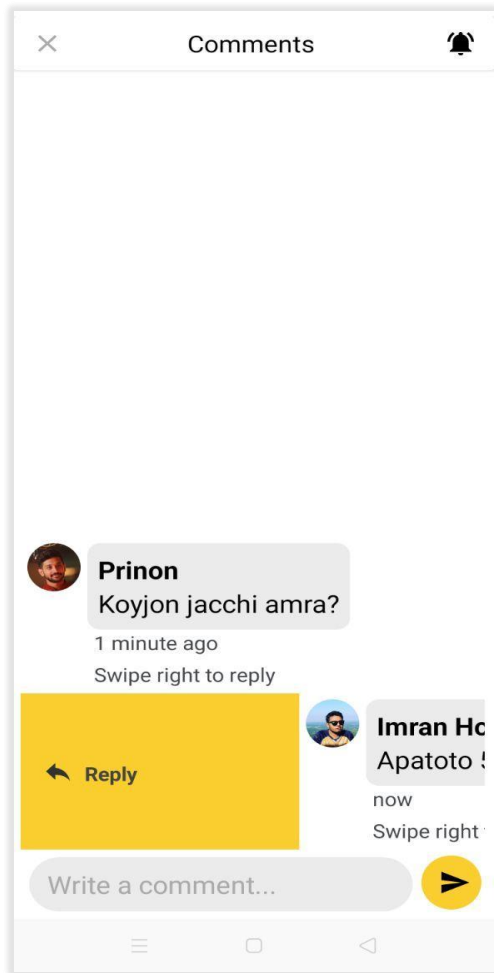


Fig 3.1.45: Going to reply box

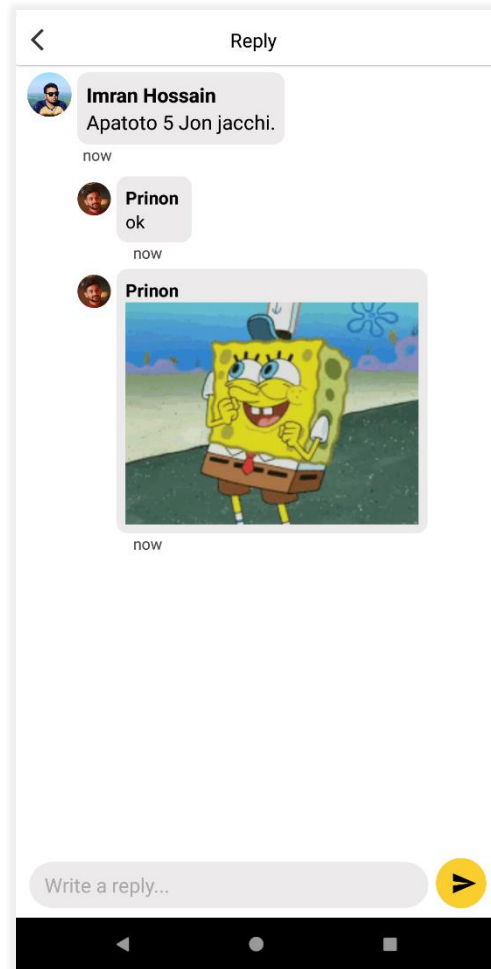


Fig 3.1.46: Reply message

98. That's all for one event. Admin and member all will comment there and can get many information about the event. In this app user can also see weather of his/her location. For see weather user will go to navigation part and go to weather.

99. Fig:3.1.47 User going to weather fragment. In weather fragment (Fig: 3.1.48) user will see current date weather. Weather will show in Celsius format. In this fragment we also show hourly forecast. We also show pressure of wind, humidity, speed of wind, sunrise, sunset, dew point, cloudiness, visibility, UV label.

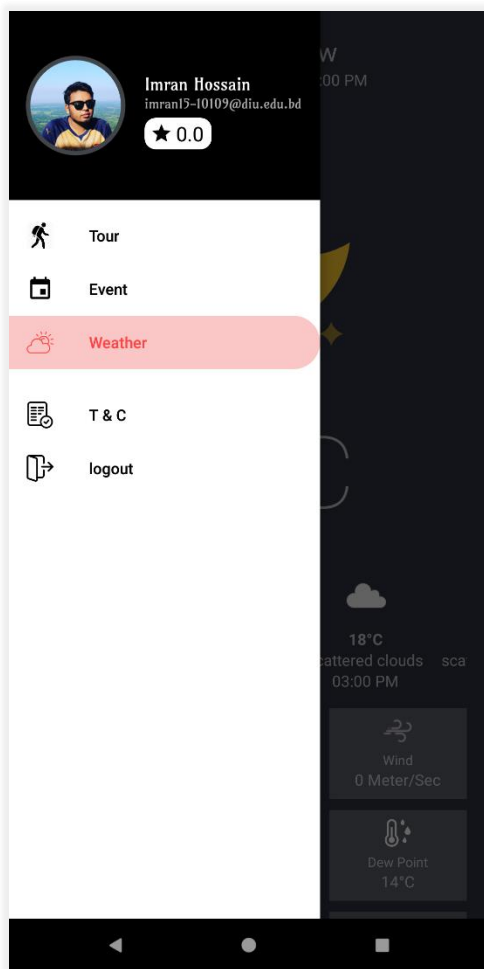


Fig 3.1.47: Going weather

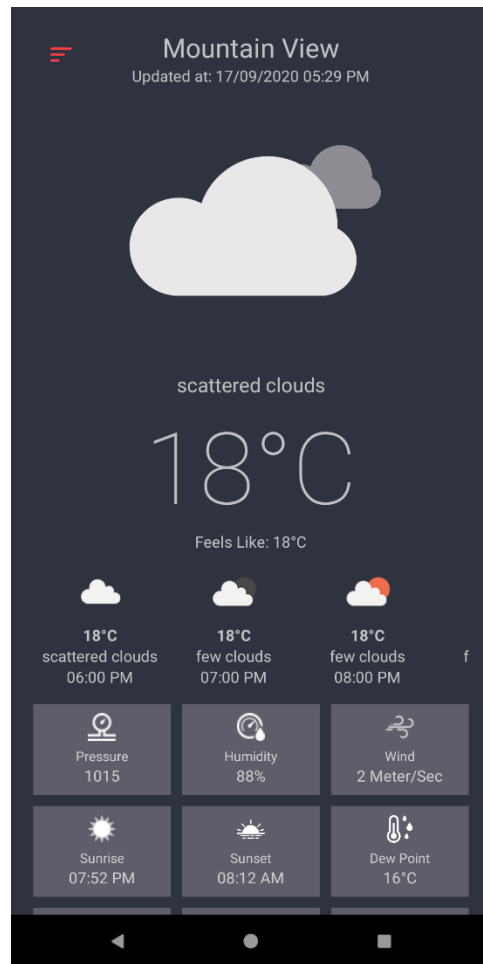


Fig 3.1.48: Weather fragment

100. That's all about **Tour Takerapp**.

3.2 Front-end Design and Details (Tour Taker for Guide):

1. First install the app to a phone and open the app.
2. Then Guide will see the front page, where guide need to input an e-mail address.

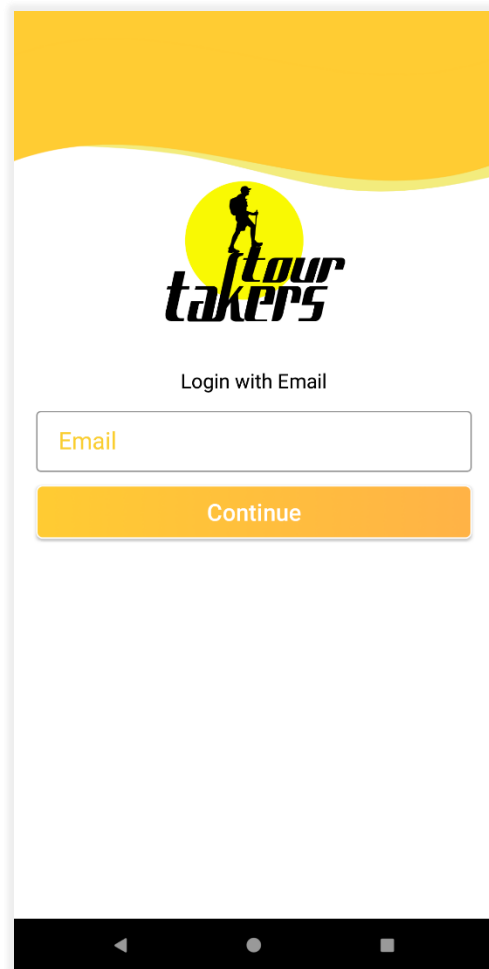


Fig 3.2.1: Guide Sign-Up Page

3. (Fig :3.2.1) We see that front page. Where guide enter an e-mail id.

4. After enter the e-mail address(fig:3.2.2), system will check that email id is already stored the server or not.

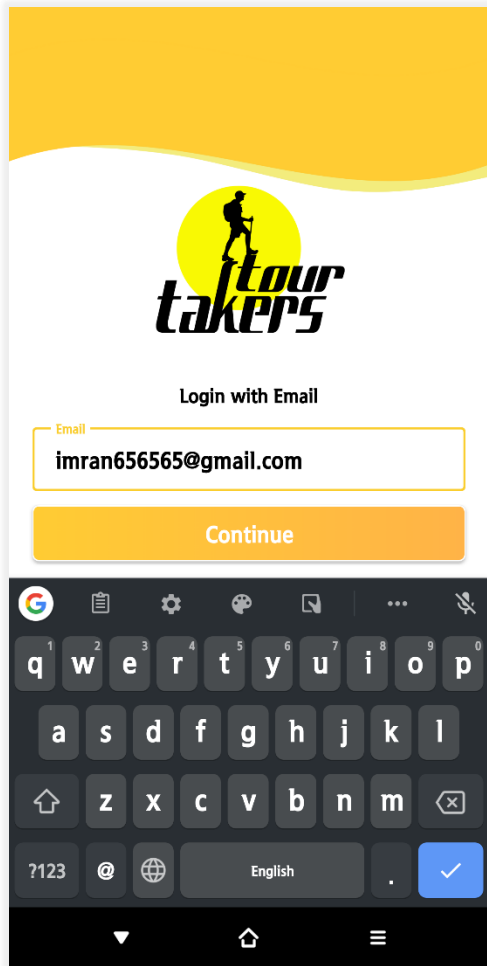


Fig 3.2.2: First-time sign-up

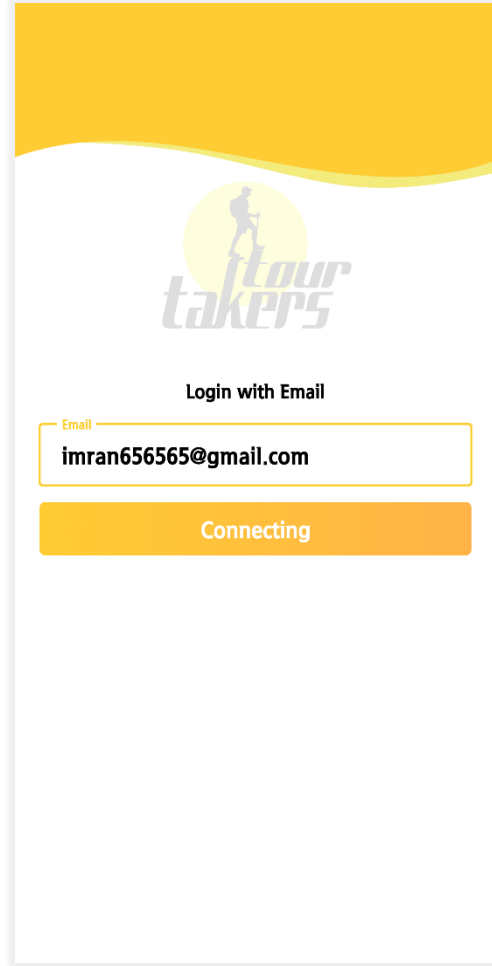


Fig 3.2.3: Checking e-mail

5. If the guide is not registered then system will ask user to Register and show the Sign-Up activity (Fig: 3.2.3).

6. Fig:3.2.4 we can see that sign up page. Guide need to file all the information.
7. Guide need to enter his/her name, email, phone no, gender, guide location and very important password.
8. Guide also give a photo of his National ID card.

Sign Up

Full Name

Email
imran656565@gmail.com

Password

Phone No

Gender : Male | Female

Select your location:

Choose your location

Government of The People's Republic of Bangladesh
NATIONAL ID CARD

National ID Verify

Sign Up

Fig 3.2.4: Sing up page

9. After complete this registration part, when guide will click the signup button a verify mail will sent to user's mail. Firebase is providing a link to verify the mail id. Before login system will check the status of that link verification. Without verify the e-mail id guide can't have access to login.

10. In fig: 3.2.5 we see user signing up. After sign up app show a small information toast (Fig:3.2.6), where it shows please verify your email id first.

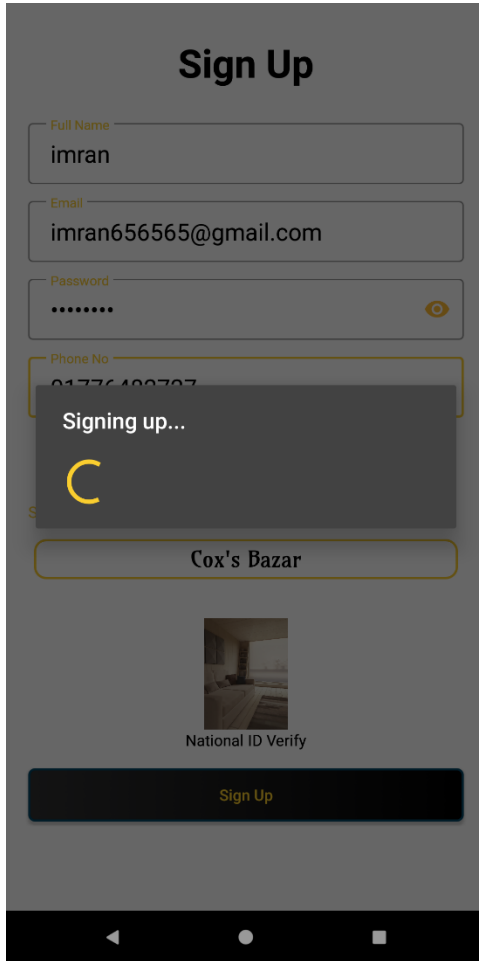


Fig 3.2.5: Signing up

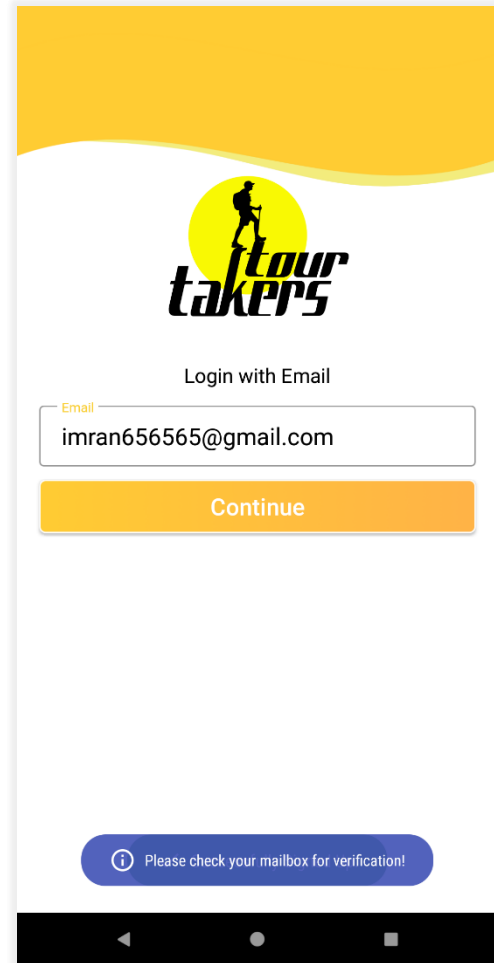


Fig 3.2.6: E-mail not verify yet

11. Then guide need to go to his/her email to verify his/her e-mail.

12. System sent an URL to that e-mail. Fig:3.2.7 we see system sent an email with an URL. After click this URL Guide verify his/her e-mail. Verifying message also show on that email (Fig: 3.2.8).

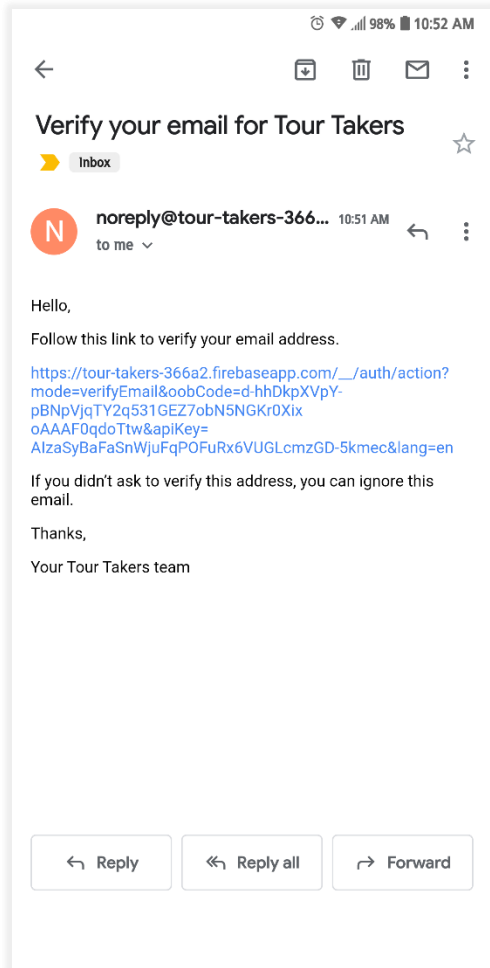


Fig 3.2.7: Verifying e-mail address.

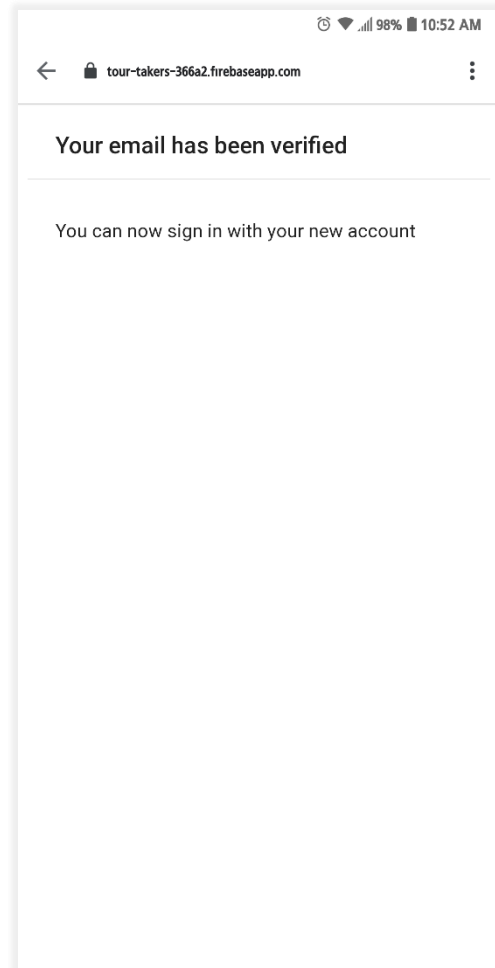


Fig 3.2.8: Verified email address

13. After verifying guide go back to **Tour Taker Guide** app.

14. And enter his/her password and Log In. This time guide successfully logs in because his/her email id is already verified.

15. After login user will see the **Main Activity** (Fig:3.2.9). There user will system want to need some permission from guide.

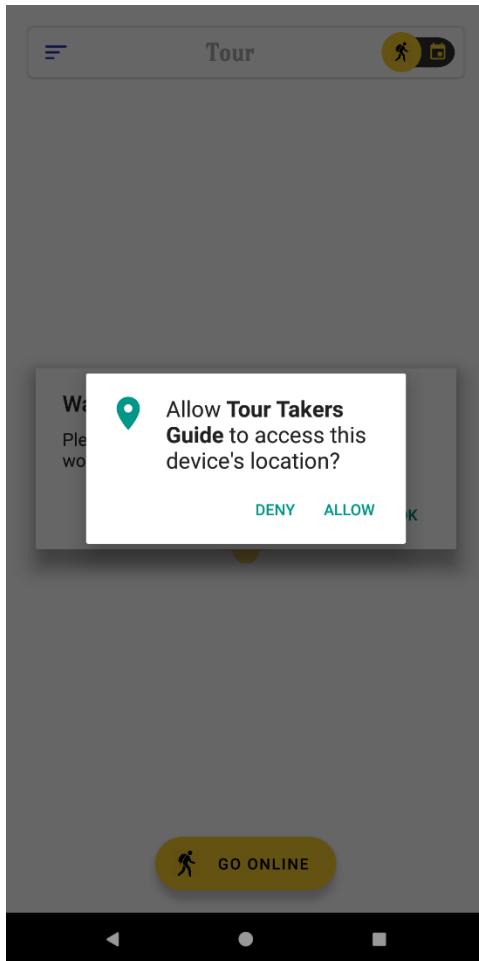


Fig 3.2.9: Location Permission

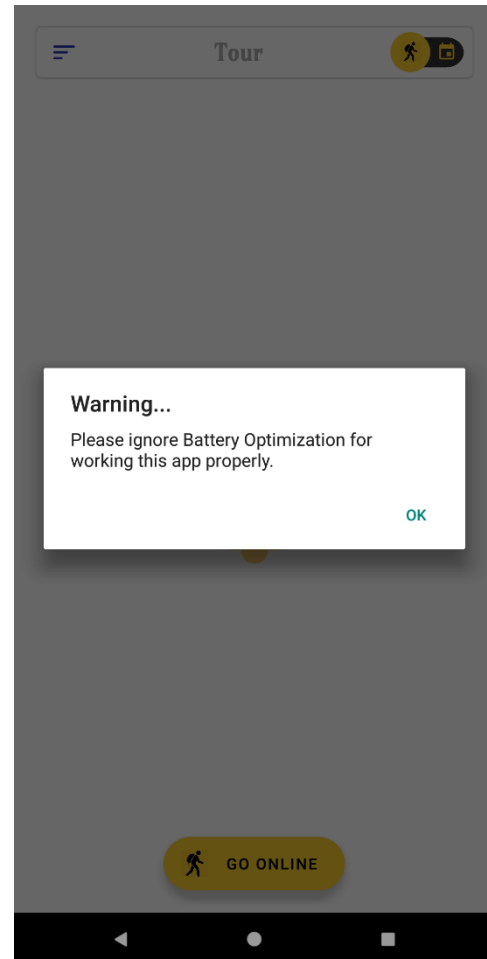


Fig 3.2.10: Battery Optimization

16. In fig 3.2.9 we see system take location permission from guide. Without location permission app can't find the guide exact location. So, guide must me allow this permission.

17. Then fig 3.2.10 system show a warning Battery Optimization for this app.

18. Then System again take a permission for always run permission (Fig 3.2.11). Because when a guide wants to take any tour or event then he needs run this app always. If he closes this app that means he is offline. So, system can't call that guide.

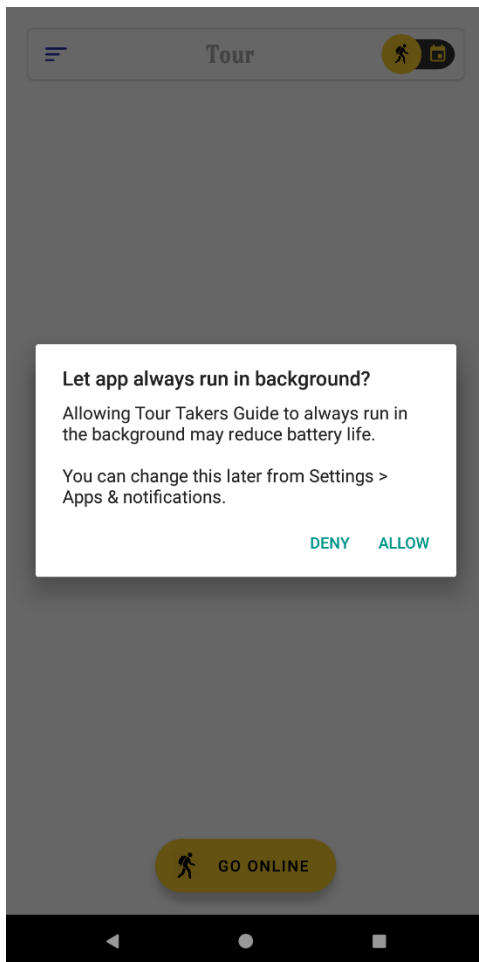


Fig 3.2.11: Always run permission

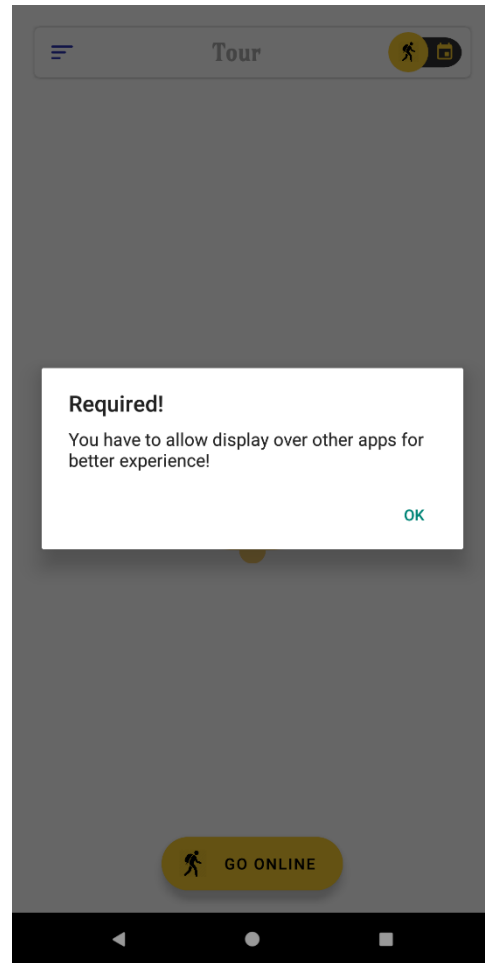


Fig 3.2.12: Allow display

19. Fig 3.2.12 System required to allow display for better experience.
20. After give all permission guide is ready for take any tour or event. If any tour or event create, and location is match with guide location then guide will get a call.

21. Fig 3.2.13 This is guide main page. We can see that at first guide has no tour or event so, guide main activity is idle.

22. Guide also show a button at bottom. That is Go Online. That means guide is off line. If guide want to take any tour, guide must be online. If guide don't want to take any tour or event for some reason then he will go offline.

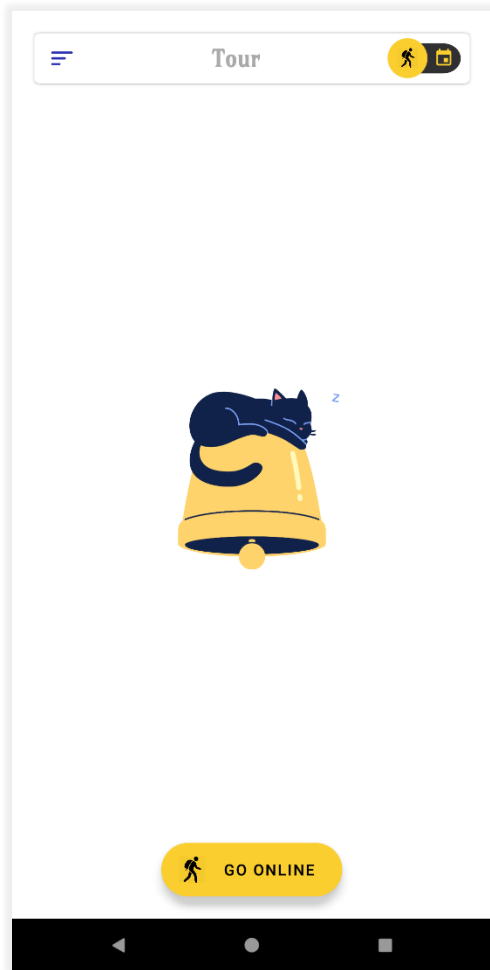


Fig 3.2.13: Guide is Offline

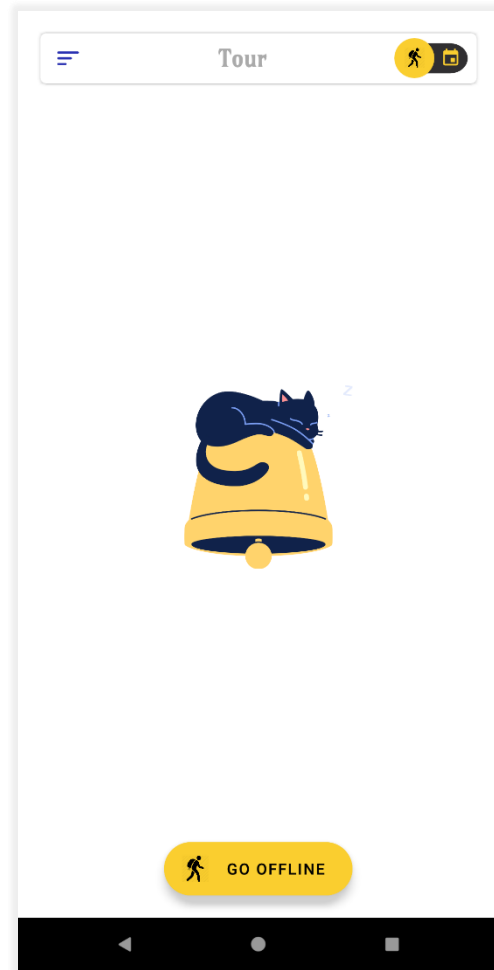


Fig 3.2.14: Guide is Online

23. On the top-left has a navigation icon. To click this icon guide can see all option at navigation bar.

24. There we can see Tours, earnings, weather, terms and conditions, settings and log out options (Fig 3.2.15).

25. If user want to see his/her profile he/she just click the picture icon (Fig 3.2.15).

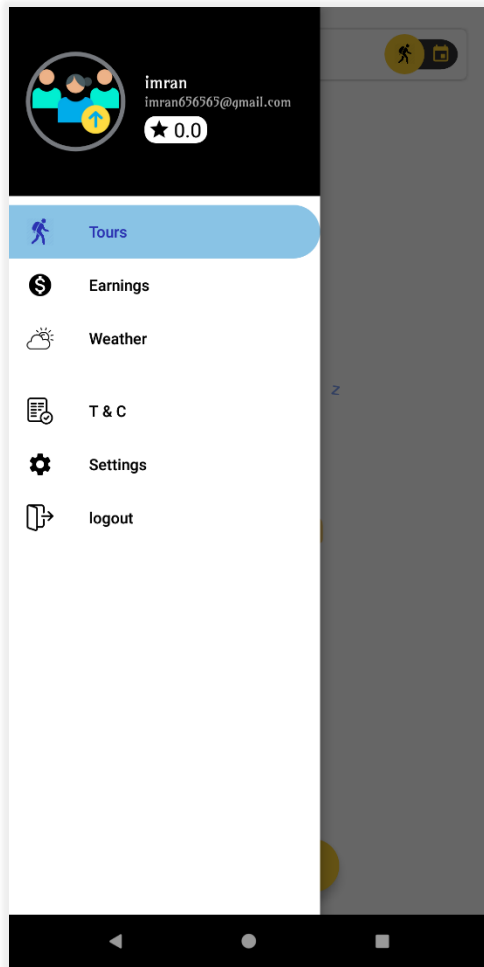


Fig 3.2.15: Navigation Bar

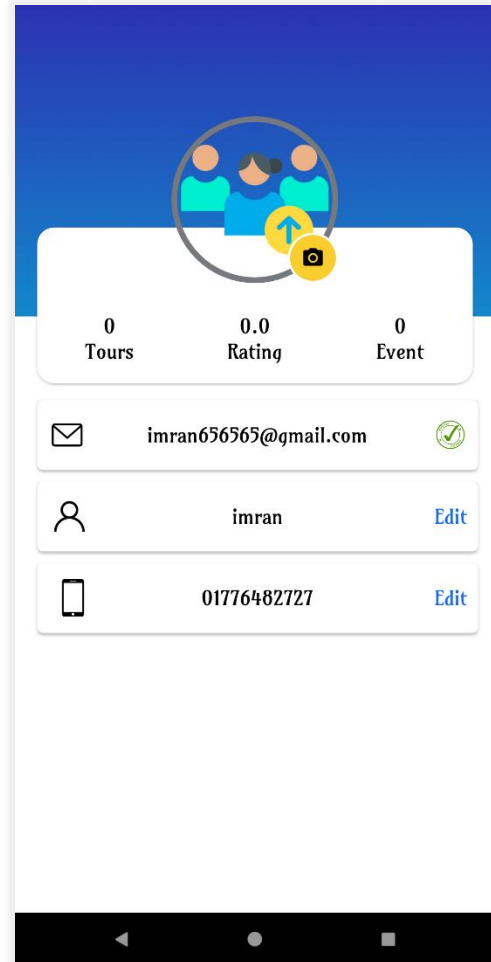


Fig 3.2.16: Guides Profile

26. Fig 3.2.16 we can see guide profile. Here guide can see total tour, event and rating. This time guide has not completed any tour or event so initially this will be 0. Rating also initially 0.

27. Guide can all see his/her name email phone number which he/she give on the registration time.

28. If guide want to edit his name or phone number, he/she can click edit (Fig:3.2.17).
29. After click edit button a bottom sheet will open from bottom to edit name. if guide want to change his/her name he/she can edit his name easily.

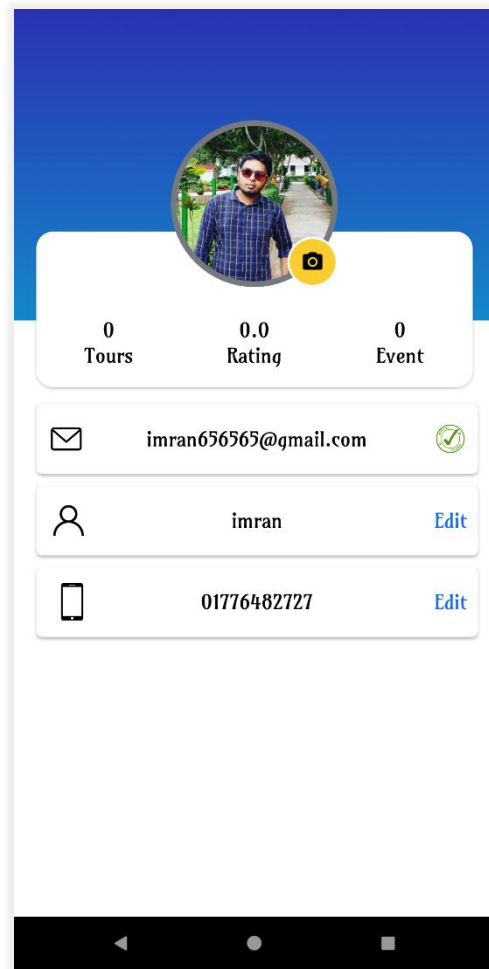


Fig3.2.17: Guide profile edit option.

30. After edit name click the save button it will edit and save this to database.
31. If guide want to add a photo of his, he can click the default icon.
32. After click this image or small camera icon system will go his media and he can browse his phone gallery or open camera for take a photo or upload an image form his phone (Fig:3.2.17).

33. If any request come to this guide than a call will came to guide app (Fig:3.2.18).
34. This call came to all guide on that district. And it will have being showing for 1 minute. If any guide accepts that request than other guides are cannot accept that request.
35. That time other guide shows a toast that will show that sorry this request is already accept by another guide.

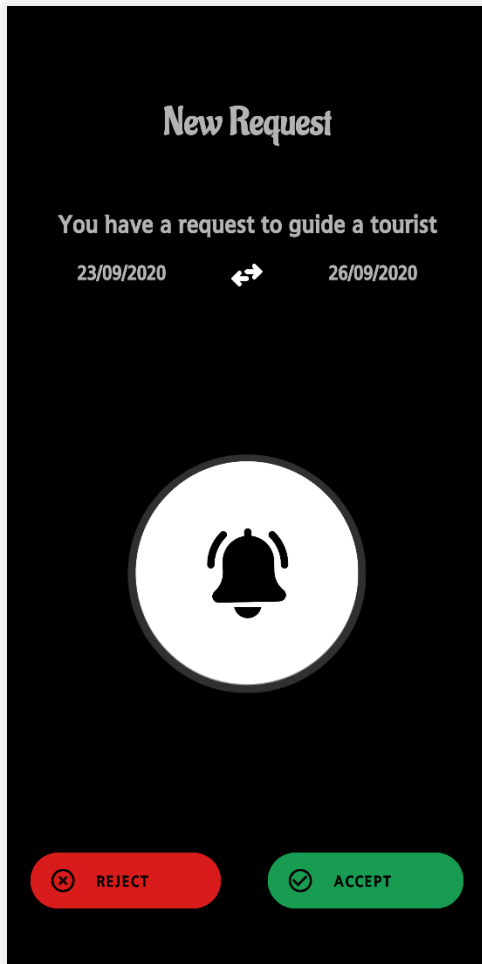


Fig3.2.18: New Request Coming

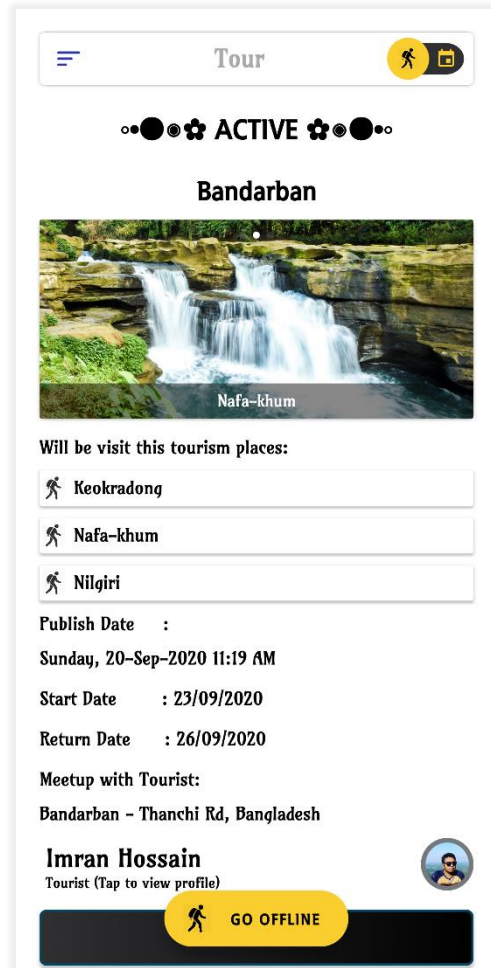


Fig 3.2.19: Accepted Tour

36. On that request activity guide will show tour start date and end date. If guide want to accepts that request then he click accept or otherwise reject button.
37. After accept this request, that tour will show on guide main activity. Where guide can see all details of this tour. That time guide will know where traveler want to go. Where the meetup place, also traveler details.

38. If guide want to see tour creator or admin details, he will just click tourist layout.
Then he will show tourist details.

39. In fig 3.2.20 guide can show tourist image, name, phone, e-mail address, tourist total number of tours, events, and rating.

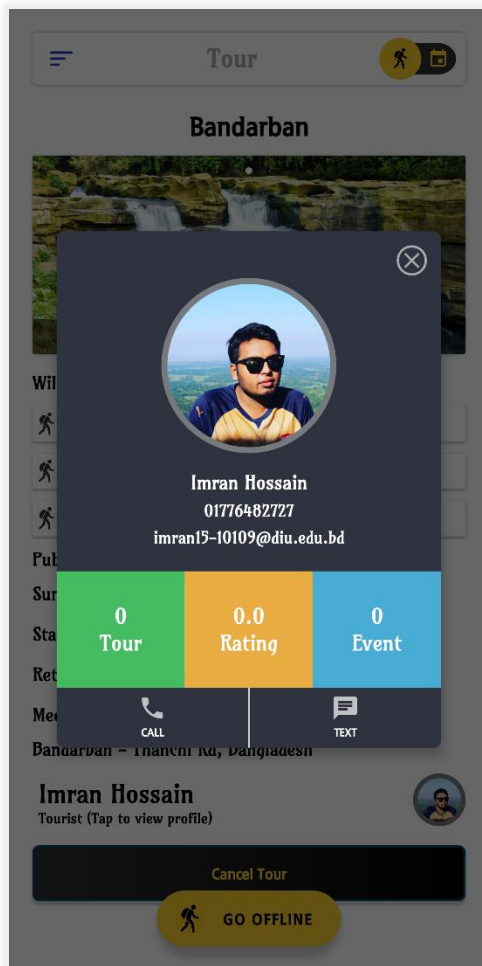


Fig 3.2.20: Tourist Details

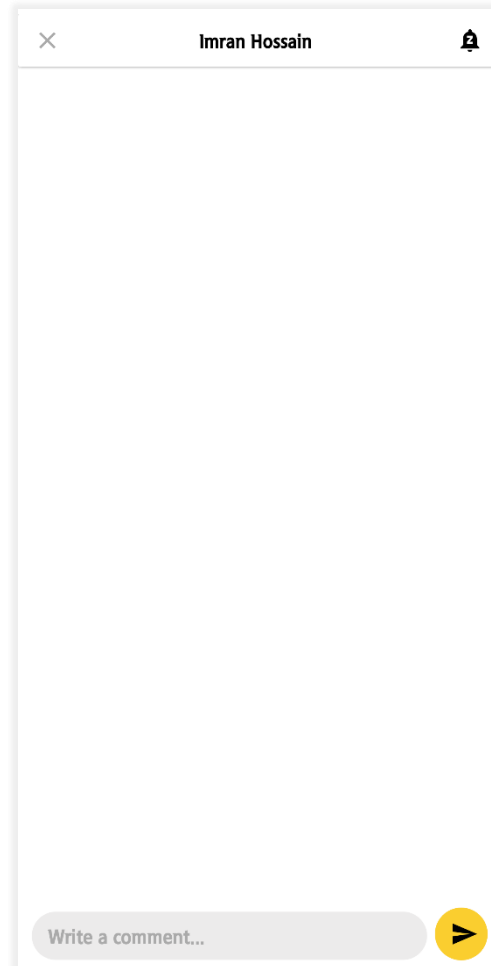


Fig 3.2.21: Message box for guide

40. Guide can also call tourist form call icon button. Guide also can text tourist.

41. Fig 3.2.21 Guide click text icon and go to chat box, where guide and tourist can chat and get extra information if they want.

42. In Guide app we also crate weather fragment. If guide want to know about the weather of his location he can.

43. If guide want to see weather then he will go to navigation bar and select weather.
The he will see his location weather.
44. We develop multiple language for guide which are Bangla and English. If guide want to switch his app language into Bangla he can. He just needs to go settings and change language.

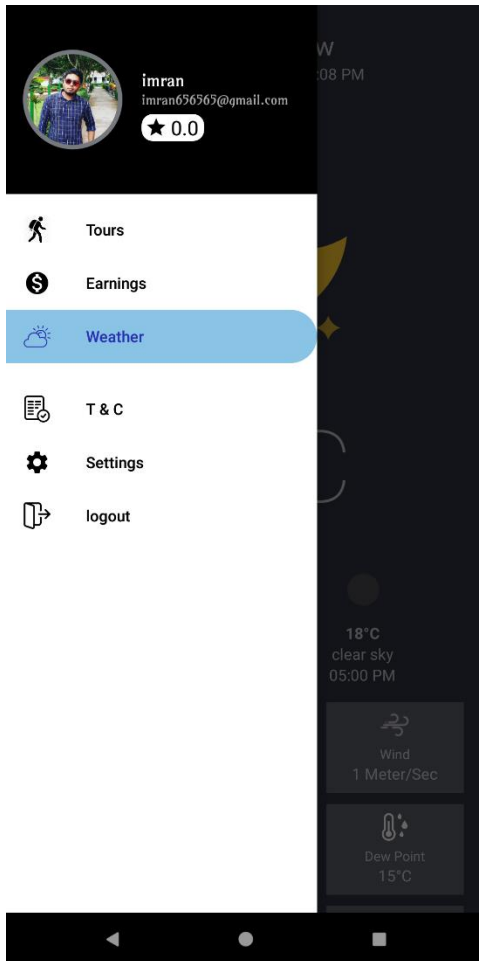


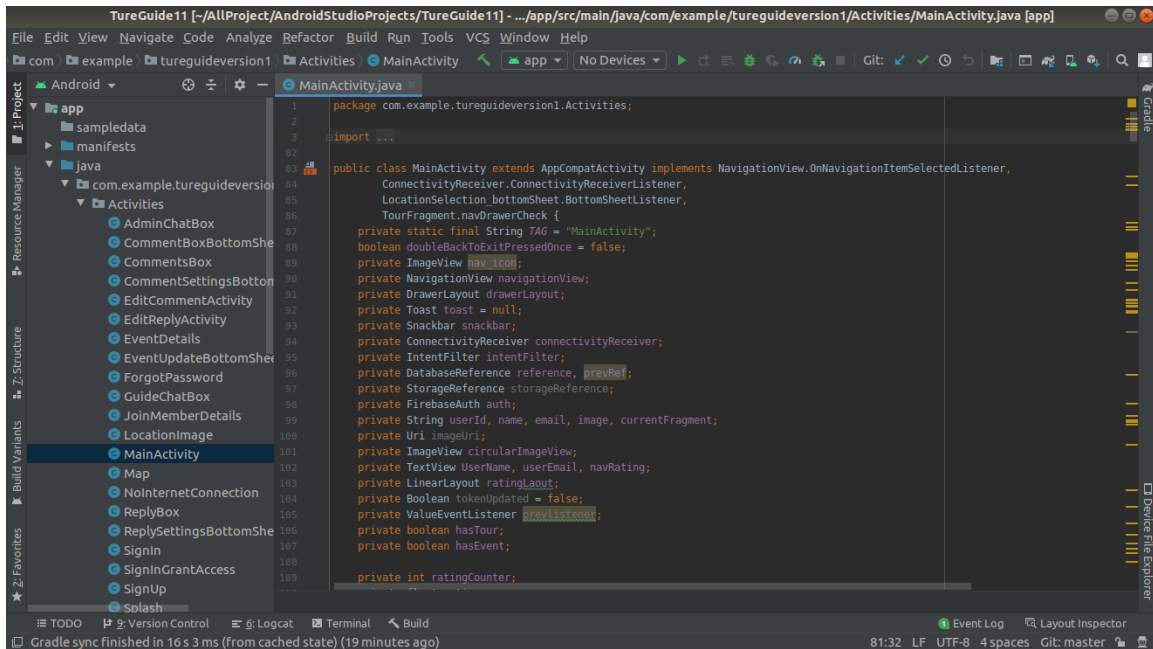
Fig 3.2.23: Going Weather Fragment



Fig 3.2.24: Guide Fragment Activity

3.3 Back-end Design and Code:

We have developed this app using android studio with java language.



```
package com.example.tureguideversion1.Activities;

import ...

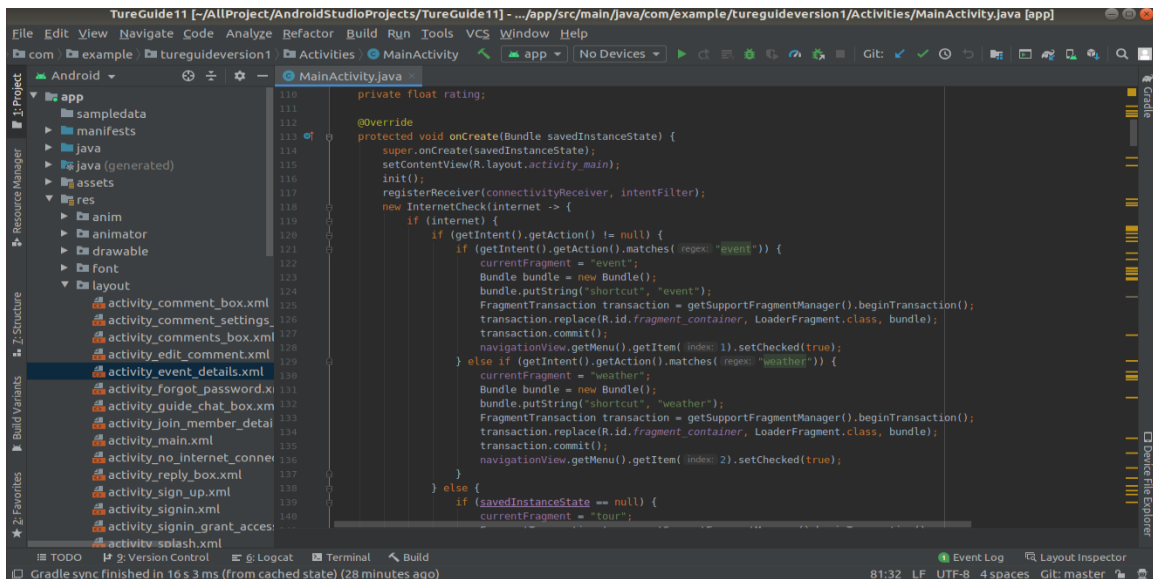
public class MainActivity extends AppCompatActivity implements NavigationView.OnNavigationItemSelectedListener,
ConnectivityReceiver.ConnectivityReceiverListener,
LocationSelection_bottomSheet.BottomSheetListener,
TourFragment.navDrawerCheck {

    private static final String TAG = "MainActivity";
    boolean doubleBackToExitPressedOnce = false;
    private ImageView nav_icon;
    private NavigationView navigationView;
    private DrawerLayout drawerLayout;
    private Toast toast = null;
    private Snackbar snackbar;
    private ConnectivityReceiver connectivityReceiver;
    private IntentFilter intentFilter;
    private DatabaseReference reference, prevRef;
    private StorageReference storageReference;
    private FirebaseAuth auth;
    private String userId, name, email, image, currentFragment;
    private Uri imageUri;
    private ImageView circularImageView;
    private TextView userName, userEmail, navRating;
    private LinearLayout ratingLayout;
    private Boolean tokenUpdated = false;
    private ValueEventListener eventListener;
    private boolean hasTour;
    private boolean hasEvent;

    private int ratingCounter;
```

Fig 3.3.1: Tour Takers Main Activity Class (1)

In Fig 3.3.1 and 3.3.2 we can see our project code, where we see main class of Tour Takers app. In the left side we can see our other class.



```
private float rating;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    init();
    registerReceiver(connectivityReceiver, intentFilter);
    new InternetCheck(internet -> {
        if (internet) {
            if (getIntent().getAction() != null) {
                if (getIntent().getAction().matches(regex: "Event")) {
                    currentFragment = "event";
                    Bundle bundle = new Bundle();
                    bundle.putString("shortcut", "event");
                    FragmentTransaction transaction = getSupportFragmentManager().beginTransaction();
                    transaction.replace(R.id.fragment_container, LoaderFragment.class, bundle);
                    transaction.commit();
                    navigationView.getMenu().getItem(1).setChecked(true);
                } else if (getIntent().getAction().matches(regex: "weather")) {
                    currentFragment = "weather";
                    Bundle bundle = new Bundle();
                    bundle.putString("shortcut", "weather");
                    FragmentTransaction transaction = getSupportFragmentManager().beginTransaction();
                    transaction.replace(R.id.fragment_container, LoaderFragment.class, bundle);
                    transaction.commit();
                    navigationView.getMenu().getItem(2).setChecked(true);
                }
            } else {
                if (savedInstanceState == null) {
                    currentFragment = "tour";
                }
            }
        }
    });
}
```

Fig 3.3.2: Tour Takers Main Activity Class (2)

In Fig 3.3.3 we see event details activity layout. Here on right we see the GUI design. And on the middle, we see xml code. We make this GUI writing those xml code.

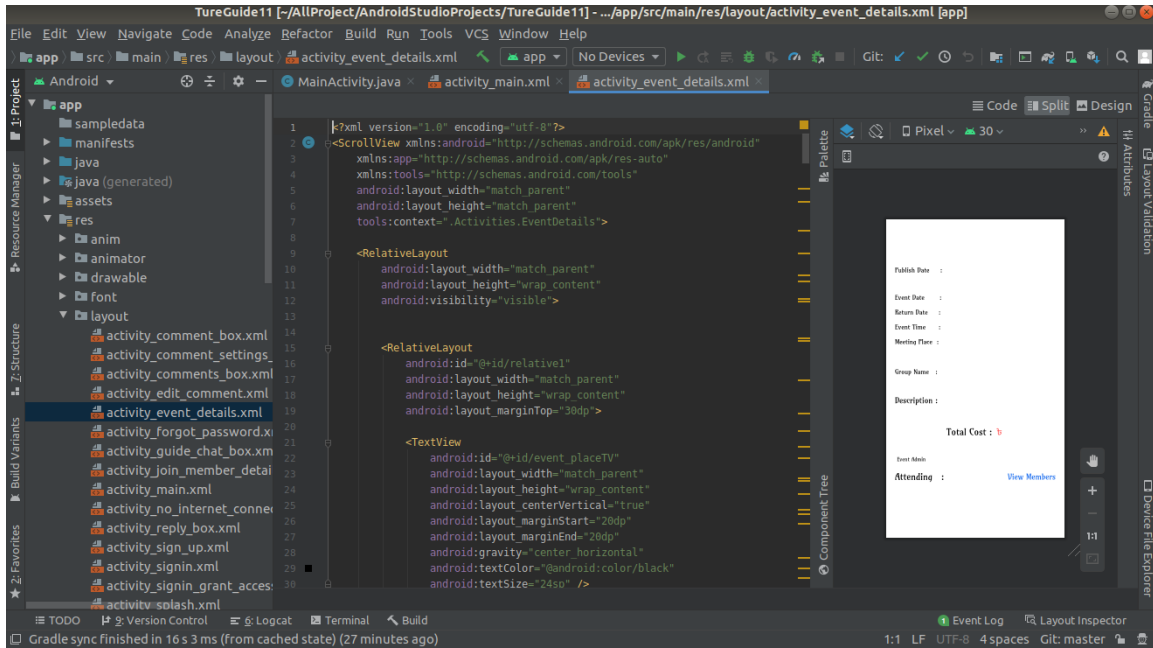


Fig 3.3.3: XML Code and design

For Tour Takers app we create many Classes, Fragments, Models, Adapters. In fig 3.3.3 on left we can see many layouts of our app.

In fig 3.3.4 we see main class of Tour Takers Guide app.

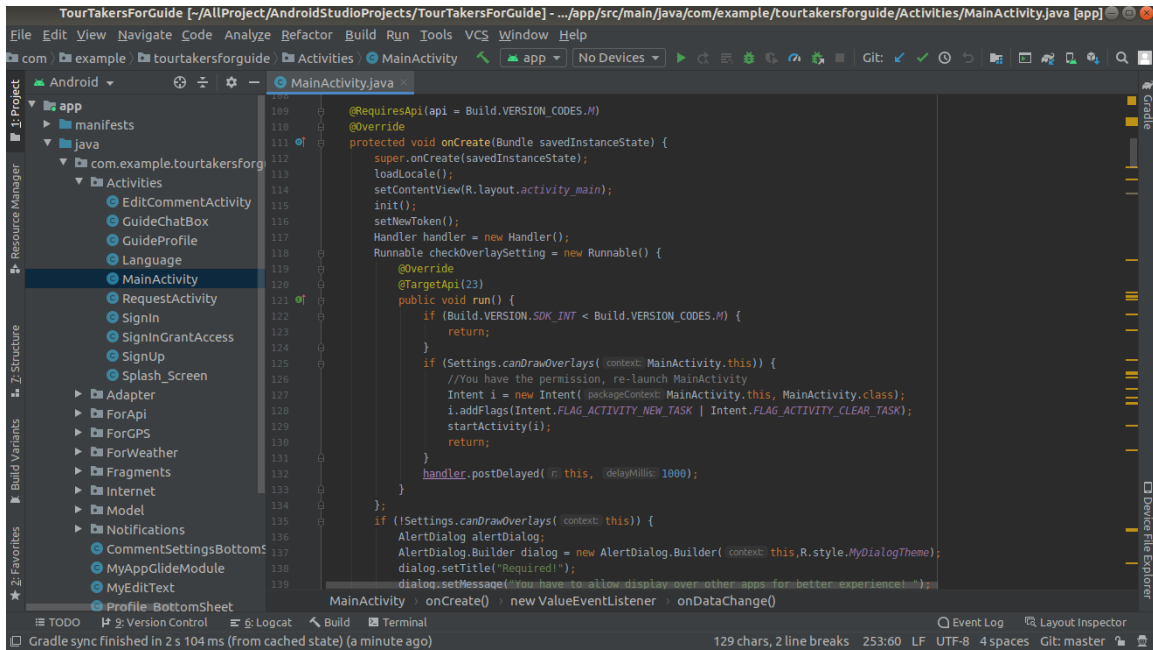


Fig 3.3.4: Main Class of Tour Takers Guide app

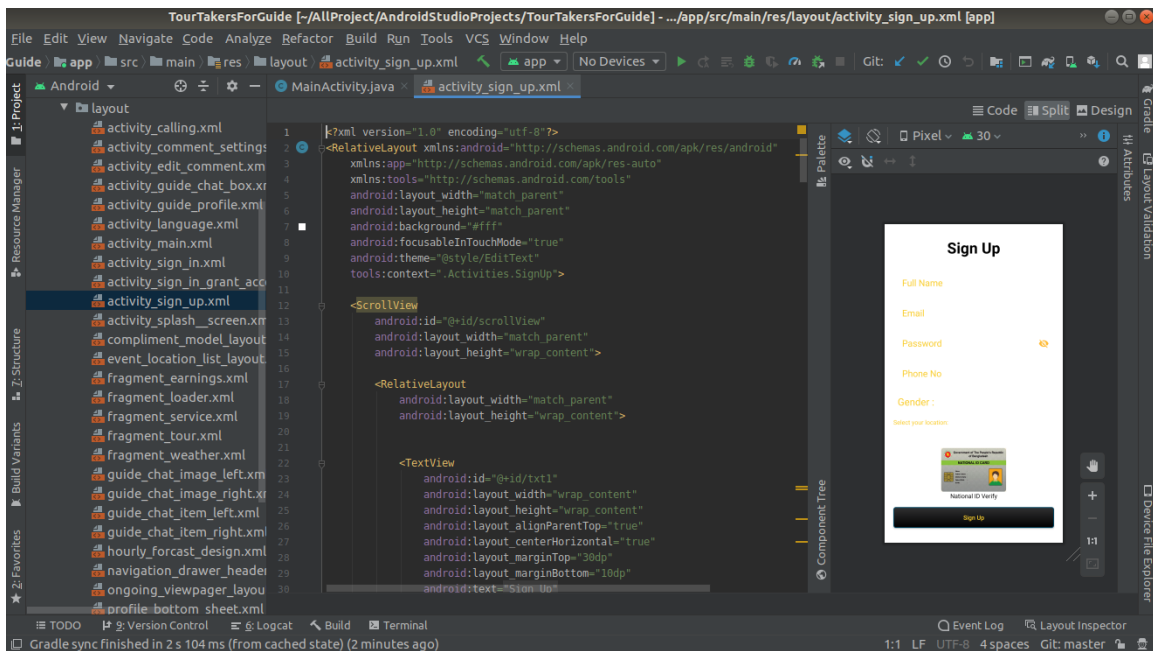


Fig 3.3.5: Sign Up Layout of Tour Takers Guide

In fig 3.3.5 we also see the layout of guide app. Here we can see the design of sign up page. Here we also see the xml code. In right we see all layout of guide app.

3.4 Data Base:

For our **Tour Taker** app, we are using **Firestore Database**. Firestore Database is a Realtime Database. This database is a cloud hosted database. Data is stored as JSON and synchronized in realtime to every connected user. When we build cross-platform apps with iOS, Android and JavaScript SDKs, all of our users share one Realtime Database instance and automatically receive updates with the newest data.

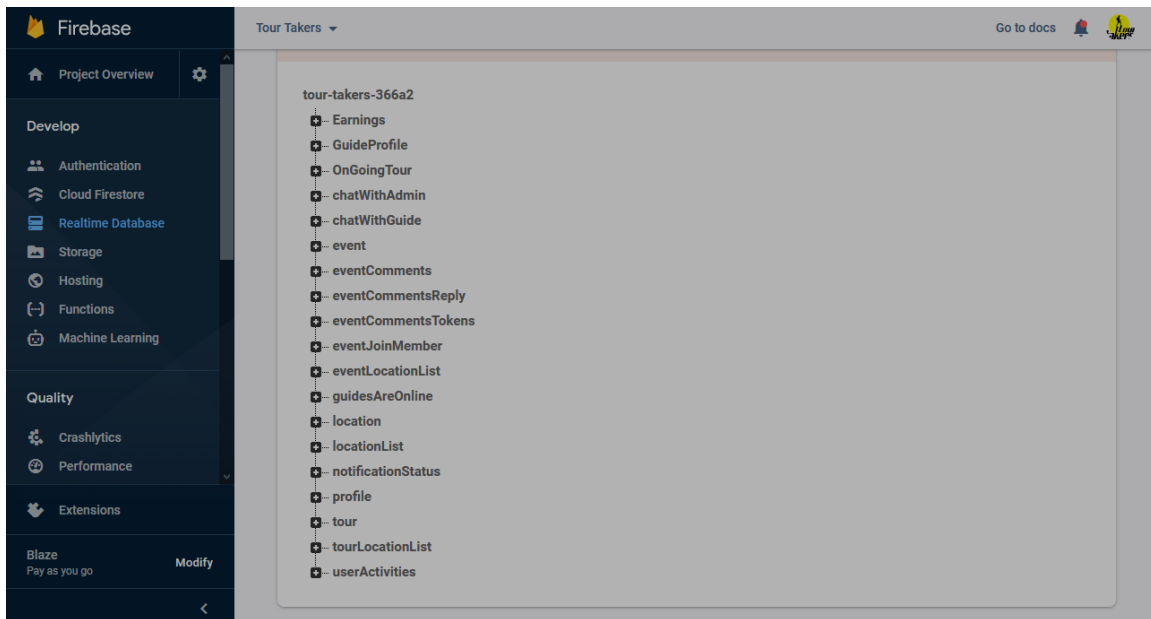


Fig 3.4.1: Firestore Database

In fig 3.4.1 we can see our tour takers database, where all table names are showing. We are also seeing some menu on the navigation bar which includes authentication, Cloud Firestore, Realtime Database, Storage, etc. Authentication authorizes user email IDs. Storage stores all image files. Realtime Database stores all data. Here we see all tables of our data. When a user first signs up, then we store his/her data in the profile table. Location tables store all locations of tourist places. Location List stores all district names of Bangladesh. Event and Tour tables store all event and tour information when a user creates this.

In fig 3.4.2 we can see user profile details. In this figure we can see when a user first signs up our app, then we take his/her some information which includes name, email, image,

phone, user gender. We also create some empty child which is user event, tour rating and rating Counter. All those child's values are initially 0. ID means user id.

Firebase Database always create a unique id for every user. And last Token child. Token child is for user device token. When user login for any device, we create a token for this device. If we send to user any notification, we are using this token.

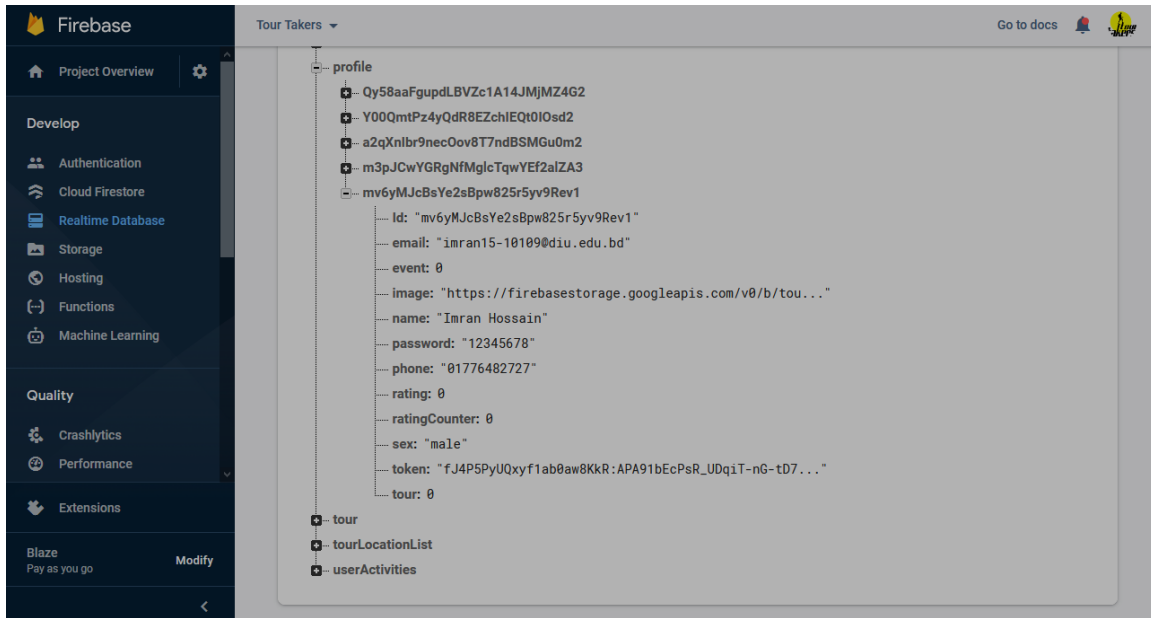


Fig 3.4.2: User Profile Database

The event and tour child are for, when a user completes a successful event or tour then it will count and add there. After complete a tour guide can rating this tourist about his behavior or how useful or anything. That will count rating child. And rating count child count how many guides get him/her. We divided rating with rating count. And get user average rating.

In Fig 3.4.3 we can see guide profile data. Here we also store guide information's. When guide first time registration our app then we collect guides data. Those data are guide name, email, guide image password, event, tour, token, id, rating and rating counter. All data are same as user data except guide National ID card image. We need this image for verify guides. We collect guide rating same way of user rating. This time user rate guide.

User Activities child store user activity. Which user create user or event and how many events or tour he creates.

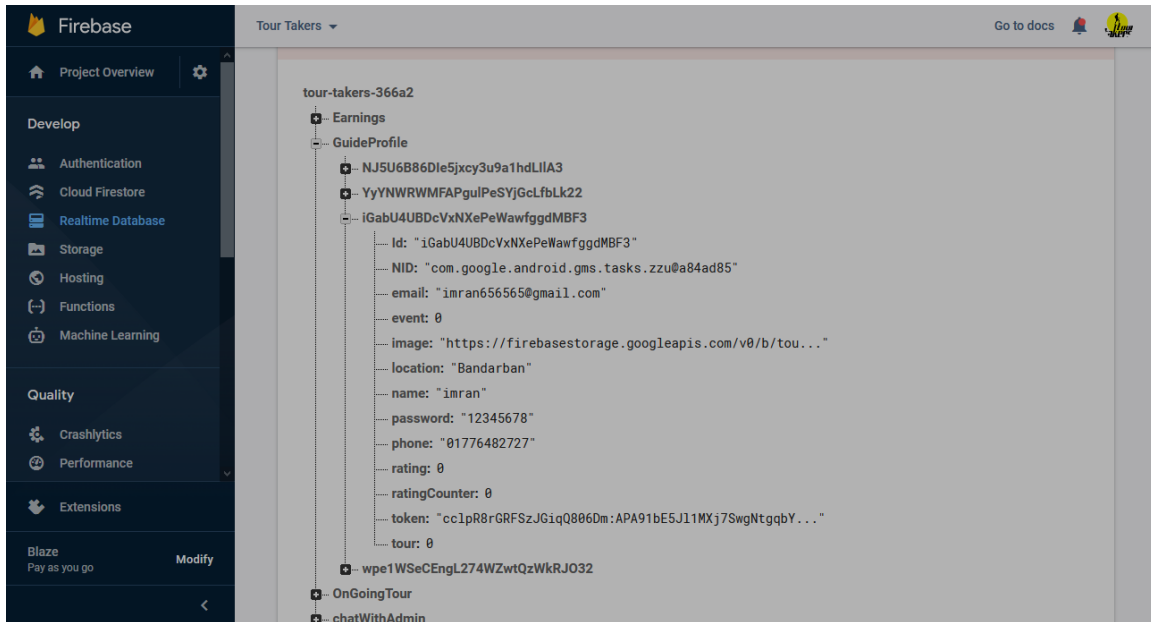


Fig 3.4.3: Guide Profile Database

In fig 3.4.4 we can see a table of event. Where we can see all information about this event. We can see eventPublisherID this id is user ID who create this event.

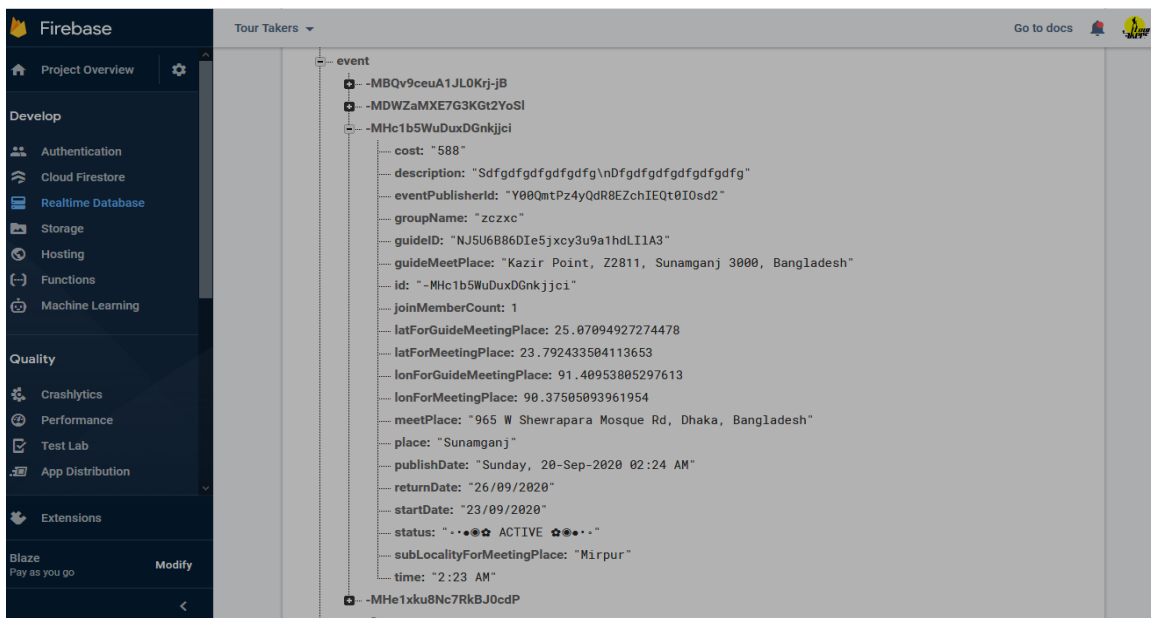


Fig 3.4.4: Event Details Database

We can see the cost of this event, tourist place, event start date, end date, meeting place, event start time, group name, description, guide meeting places all store there.

3.5 Implementation Requirement:

In our project, we have implemented many things.

- Google map
- Firebase database
- Lottie Animation
- Programming java.
- Push notification
- Crop Image
- Retrofit
- Google Map SDK
- Google Places API
- Google auto complete support fragment
- Android X
- Google Material

Chapter 4: Conclusion and Future Scope

4.1 Discussion

Finally, we have finished to our App project. After all, by the grace of Allah we just make it. Using this application people can easily travel. It is helpful for all traveler who want to travel with group or single.

We've heard a lot about the challenges of creating this app. When we started this app, we didn't know about how google map work, how firebase database work. But we learn and implement this to our app. It was a wonderful learning experience.

4.2 Future Scope

In the future, we want to lunch this app to Play Store. We want to live this app for Bangladesh. For this app many guides can earns some money. We will implement payment getaway system in future.

ORIGINALITY REPORT

12%
11%
2%
0%

SIMILARITY INDEX INTERNET SOURCES PUBLICATIONS STUDENT PAPERS

PRIMARY SOURCES		
1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	9%
2	www.researchgate.net Internet Source	1%
3	dspace.library.daffodilvarsity.edu.bd:8080 Internet Source	1%
4	hdl.handle.net Internet Source	1%
5	opus.govst.edu Internet Source	<1%
6	J HECHTMAN. "Guitar plug male connectors", Audio Wiring Guide, 2008 Publication	<1%
7	Reza MokarramAydenlou. "Types of existing buildings: detailedintroduction and seismicrehabilitation", Elsevier BV, 2020 Publication	<1%
8	Takeo Hamada, Hiroshi Kamata, StephanieHogg, and CarelMarsman. "NetworkManagement and Administration", The CRC Handbook of Modern Telecommunications, 2000. Publication	<1%

9	www.austinhomesearch.com Internet Source	<1%
10	"Sendai Port", American Society of Civil Engineers (ASCE), 2014 Publication	<1%
11	mafiadoc.com Internet Source	<1%

Exclude quotes Off

Exclude matches Off

Exclude bibliography Off