| **S.P. Mandali’s**  **Ramnarain Ruia Autonomous College**  **Matunga, Mumbai**  **Project Report**  **On**  **Blood Donation App**  **Project Guide**  **Pooja Mam**  **Project By**  **Rupesh Kumar Bhagwane**  **TYBSC**  **Department of Computer Science**  **2021-2022**  **INDEX**   | Sr No | Contents | Page No. | | --- | --- | --- | | **A** | [**Preliminary Investigation**](#bookmark=id.30j0zll) |  | | 1 | Description of Present System | 6 | | 2 | Limitations of Present System | 8 | | 3 | Proposed System | 10 | | 4 | Advantages of Proposed System | 12 | | 5 | Feasibility Study | 14 | | 6 | Stakeholders | 16 | | 7 | Project Requirements | 18 | | 8 | Technologies Used | 20 | | 9 | Gantt Chart | 22 | | **B** | [**System Analysis**](#bookmark=id.1fob9te) |  | | 10 | Project Development Methodology | 25 | | 11 | Event Table | 26 | | 12 | Use Case Diagram | 28 | | 13 | Use Case Scenario | 30 | | 14 | Activity Diagram | 35 | | 15 | Class Diagram | 49 | | 16 | Sequence Diagram | 50 | | 17 | Collaboration Diagram | 53 | | 18 | State-Chart Diagram | 59 | | **C** | [**System Design**](#bookmark=id.3znysh7) |  | | 19 | Component Diagram | 62 | | 20 | Package Diagram | 63 | | 21 | Deployment Diagram | 64 | | 22 | List of Tables with Attributes & Constraints | 65 | | **D** | [**System Coding**](#bookmark=id.2et92p0) |  | | 23 | Screen Layouts | 68 | | 24 | Coding | 78 | | 25 | Secure Coding Practices | 232 | | **E** | [**Future Enhancements**](#bookmark=id.tyjcwt) | 234 | | **F** | [**Bibliography**](#bookmark=id.3dy6vkm) | 236 | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**Preliminary Investigation**

| Description of Present System   * The Present System Resister Person as donor or Receiver. * After Resister as Receiver, they ask to wait for match blood group * They don't take or require any documented proof for identity. * In Registration, they ask medical queries related to blood donation or receive.   Limitations of Present System   * The patients or receivers cannot get the information of donor easily. * It time consuming and not gives the immediate result for search. * The system lack for providing clear vision between two entities. * No, Contact approaches like chat corner available for two entity to share or talk before physical meet. * There is no requirement for document as proof of identity to have a proper sight of being.   Proposed System  **Virtual Blood Donation App**   * Proposed system will create an android app that work over mobile. * Both Donor and Receiver can use this without going anywhere. * Receiver no need to go anywhere to search for his blood, they get matched person contact details and a chat conversation section. * According to blood group and place of location suggest the person’s.   Advantages of proposed system   * The people in need of blood can search for the donors by giving their blood group and city name. * It is very flexible and user friendly. * The person’s time and work is reduced very much which prevails in the present system. * Easy and Helpful. * The people are not limited to receive or provide services in working hours of the branch only; he is serviced 24 hours a day, 7 days of week and 365 days of the year. * There is human satisfaction contact approaches for donor & Receiver. * No issues with Authentication and security.   Feasibility study   1. **Technical Feasibility:**   The proposed system is developed using android layouts, Fragments and XML as front-end tool and Firebase as the back end. The Android Emulator or mobile is used to view the layouts that is available within android studio. All the required hardware and software are readily available in the market, Hence the app is technically feasible.   1. **Economic Feasibility:**   As the necessary hardware are available in the market at a low cost and software which are needed are open source and free, the investment is the only cost incurred and does not need any further enhancements. Hence it is economically feasible.   1. **Operational Feasibility:**   The Proposed system is operationally feasible because of the following reason.   * The customer is benefited more as most of his time is saved. The customer is serviced at his place of work. * The cost of the proposed system is almost negligible when compared to benefits are gained.  1. **Environmental Feasibility:**   The system is environmentally feasible as it doesn’t use many natural resources actually it helps in working with nature instead of against it.  Stakeholders   1. **Admin:**   Admin is the developer or the person who is owing it or in future case hospitals who can update, delete functionality in app.   1. **Donor or Receiver (patient):**   Donor: -  Donor will first register itself as normal person.  Donor has to Provide his Personal Details and document proof for identity proof for registration.  To donate the person has to opt for donation section where some health-related queries need to answer, and they will registered as donor.  Receiver (Patient): -  Receiver will first register itself as normal person.  Receiver has to Provide his Personal Details and document proof for identity proof for registration.  To Receive the person has to opt for recipient section where some health-related queries need to answer, and they will registered as receiver, and they will immediate shown up with result if person are there in with match.  Technologies used  **Software Requirements:**   | PC Operating System | Windows 10 or higher | | --- | --- | | PC Software Used | Android Studio | | PC Database Software | Firebase | | Android SDK |  | | Android Mobile | OPPO A9 | | Android Special Requirements | Google Play Services |   **Hardware Requirements:**   | PC RAM Required | 4 GB of RAM. 8 GB RAM is recommended (2 GB minimum for running SDK) | | --- | --- | | PC Disk Space Required | Requires 3GB of minimum disk space required. | | PC Screen Resolution | 1920 X 1080(recommended) | | PC Processor | Intel core i5 (minimum i3) | | Android Special Requirements | Camera, Storage Permission | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**GANTT CHART**

Gantt chart

| **Phase Title** | | **June** | | | | | | | | | | | **July** | | | | | | | | | | **August** | | | | | | | | | | | **September** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1 | | | 2 | | 3 | | 4 | | 5 | | 1 | | 2 | | 3 | 4 | | 5 | | 6 | 1 | | 2 | | | 3 | | | 4 | | 5 | 1 | 2 | 3 | | 4 | | 5 | | |
| **Project Approval** | |  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
| **Preliminary Investigation** | |  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
| **System Analysis and Design** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| **Event Table** | |  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
| **Class Diagram** | |  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
| **Use Case Diagram** | |  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
| **Use Case Specification** | |  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
| **Sequence Diagram** | |  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
| **Activity Diagram** | |  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
| **State Chart Diagram** | |  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
|  | | |  | |  | |  | |  | |  | |  | |  |  | |  | |  |  | |  | | |  | | |  | |  |  |  |  | |  | |  | | |
| **Phase Title** | October | | | | | | | | | November | | | | | | | | December | | | | | | | | January | | | | | | | | | February | | | | | | | |
|  | 1 | | 2 | 3 | | 4 | | 5 | | 1 | | 2 | | 3 | | 4 | 5 | 1 | 2 | | 3 | | 4 | 5 | | | 1 | | 2 | 3 | | 4 | | 5 | 1 | | 2 | | 3 | | 4 |  |
| **Coding** |  | |  |  | |  | |  | |  | |  | |  | |  |  |  |  | |  | |  |  | | |  | |  |  | |  | |  |  | |  | |  | |  |  |
|  | |  |  | |  | |  | |  | |  | |  | |  |  |  |  | |  | |  |  | | |  | |  |  | |  | |  |  | |  | |  | |  |  |
|  | |  |  | |  | |  | |  | |  | |  | |  |  |  |  | |  | |  |  | | |  | |  |  | |  | |  |  | |  | |  | |  |  |
| **Testing** |  | |  |  | |  | |  | |  | |  | |  | |  |  |  |  | |  | |  |  | | |  | |  |  | |  | |  |  | |  | |  | |  |  |
|  | |  |  | |  | |  | |  | |  | |  | |  |  |  |  | |  | |  |  | | |  | |  |  | |  | |  |  | |  | |  | |  |  |
|  | |  |  | |  | |  | |  | |  | |  | |  |  |  |  | |  | |  |  | | |  | |  |  | |  | |  |  | |  | |  | |  |  |
| **Implementation** |  | |  |  | |  | |  | |  | |  | |  | |  |  |  |  | |  | |  |  | | |  | |  |  | |  | |  |  | |  | |  | |  |  |
|  | |  |  | |  | |  | |  | |  | |  | |  |  |  |  | |  | |  |  | | |  | |  |  | |  | |  |  | |  | |  | |  |  |
|  | |  |  | |  | |  | |  | |  | |  | |  |  |  |  | |  | |  |  | | |  | |  |  | |  | |  |  | |  | |  | |  |  |
| **Phase Title** | March | | | | | | | | |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  | 1 | | 2 | 3 | | 4 | | 5 | |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  |
| **Report Submission** |  | |  |  | |  | |  | |  |
|  | |  |  | |  | |  | |  |
|  | |  |  | |  | |  | |  |

|  | Actual Time of  Completion |  | Expected Time of C  ompletion |  | Break |
| --- | --- | --- | --- | --- | --- |

**System**

**Analysis**

| **METHODOLOGY**  The Agile software development methodology is one of the simplest and effective processes to turn a vision for a business need into software solutions and which is to be used in this Project development from start to end. Agile is a term used to describe software development approaches that employ continual planning, learning, improvement, team collaboration, evolutionary development, and early delivery. It encourages flexible responses to change. Small projects can be implemented very quickly. For large projects, it is difficult to estimate the development time. Error can be fixed in the middle of the project. The Testers and developers work together on implementing it.    **Event Table**   1. **USER**  | **Sr. No.** | **Event** | **Trigger** | **Source** | **Activity** | **Response** | **Destination** | | --- | --- | --- | --- | --- | --- | --- | | 1. | User creates an account | Request to sign up | User | Enter details for registration | Registered | User | | 2. | User logs in to their account | Request to login | User | Verify Authentication | Logged in | User | | 3. | User wants to View Donation Details | Request to  View | User | View Donation Details | Donation details viewed | User | | 4. | User wants to View Recipients Details | Request to View | User | View Recipients Details | Recipient’s details viewed | User | | 5. | User wants to Donate Blood | Request to Fill Details. | Donor User | Filling Details | Details been recorded. | User | | 6. | User wants to Save Donation | Request to save session | Donor User | Save Donation details | Mention details Saved | User | | 7. | User wants to give preview | Request to preview | Donor User | Preview the Filled Details | Previews Details | Details | | 8. | User wants to Receive Blood | Request to Fill Details. | Recipient User | Filling Details | Details been recorded. | User | | 9. | User wants to Save Recipient | Request to save session | Recipient User | Save Donation details | Mention details Saved | User | | 10. | User wants to give matching | Request to preview | Recipient User | Preview The matching blood | Previews Details | Details | | 11. | User want to logout | Request to Logout | User | Logout | Logout | User |   USE CASE DIAGRAM        USE CASE SCENARIO   1. UseCase Specification: Donor or patient wants to register   This use case describes sign-up of their account.  Invoking Actor: Donor/Patient  Basic flow:  The usecase begins with opening the sign-up form User  must enter the username, password, Contact, DOB, Gender,  Identity Proof Document.  Pre-Condition:  Username, Password, Contact, DOB, Gender  Post-Condition:  Account is successfully created.   1. Use Case Specification: Donor or patient wants to Login.   This use case describes login of their account.  Invoking Actor: Donor/Patient  Basic Flow:  The use case begins with opening the login page.  User must enter the username/contact number & password.  Pre-Condition:  Username/Contact number & password is required for login.  Post-Condition:  Open Dashboard Successfully.   1. Use Case Specification: Donor wants to Donate Blood.   This use case describes applying for donation.  Invoking Actor: Donor  Basic Flow:  The use case begins with opening Blood Donation form.  Pre-Condition:  Donor have to fill all mandatory Details for Donation  Post-Condition:  Donation applied successfully.   1. Use Case Specification: Donor want to see Recipients Profile.   This use case describes of viewing the blood receiver Details.  Invoking Actor: Donor  Basic Flow:  The use case begins with displaying the profile  Pre-Condition:  Donor must accept or reject the Donation.  Post-Condition:  Profile Viewed Successfully.     1. Use Case Specification: Patient/Recipients wants to Receive Blood.   This use case describes applying for blood receive.  Invoking Actor: Patient/Recipients  Basic Flow:  The use case begins with opening Blood receiving form.  Pre-Condition:  Receivers have to fill all mandatory Details for have Blood.  Post-Condition:  Recipients applied successfully.   1. Use Case Specification: Patient want to search for Matched blood.   This use case describes of viewing the matched blood and place people.  Invoking Actor: Patient/Recipients  Basic Flow:  The use case begins with displaying the different person of the same blood and near place location.  Pre-Condition:  Patient/Recipient have to select person.  Post-Condition:  Matching or Searching Successfully.   1. Use Case Specification: Patient want to communicate for Matched blood.   This use case describes creating a communication corner.  Basic Flow:  The use case begins by providing Direct message space for them.  Pre-Condition:  Patient/Recipient have to Direct message to chat with person.  Post-Condition:  Communication space launched Successfully.   1. Use Case Specification: Donor needs to delete or un submit his donation.   This use case describes deletion of donation form.  Invoking actor: Donor:  Basic Flow:  The use case begins deleting donation.  Pre-Condition:  Mobile OTP verification required to delete.  Post-Condition:  Session is deleted   1. Use Case Specification: Donor has to Confirm his Donation.   This use case describes Donor has to confirm his previous donation.  Invoking Actor: Admin  Basic Flow:  The use case begins with asking Donor his/her donation.  Pre-Condition:  On Donor opening app ask for his previous donation details.  Post-Condition:  Donation Confirmation done Successfully.   1. Use Case Specification: Admin need to update Security approach or task.   This use case describes admin has to update the task, city, and document uploading space.  Invoking Actor: Admin  Basic Flow:  The use case begins with update in space for document uploading.  Pre-Condition:  The space is increased or size of uploading changed.  Post-Condition:  Updating Successfully.  Activity Diagram        Class Diagram    Sequence Diagram   * Donor      * Patient/Receiver     Collaboration Diagram      State Chart Diagram    Component Diagram    Package Diagram    Deployment Diagram    List Tables with Attributes and Constraint.   | **Table Name** | **Field Name** | **Data Type** | **Description** | | --- | --- | --- | --- | |  | Password | String | Not Null | |  | User name | String | Not Null | | Donor User | User id | String | Not Null | |  | Full Name | String | Not Null | |  | Contact | Phone | Not Null | |  | DOB | Date | Not Null | |  | Document | image | Nit Null |  | **Table Name** | **Field Name** | **Data Type** | **Description** | | --- | --- | --- | --- | |  | Password | String | Not Null | |  | User name | String | Not Null | | Patient User | User id | String | Not Null | |  | Full Name | String | Not Null | |  | Contact | Phone | Not Null | |  | DOB | Date | Not Null | |  | Document | image | Nit Null |  | **Table Name** | **Field Name** | **Data Type** | **Description** | | --- | --- | --- | --- | |  | User id | String | Not Null | |  | Full Name | String | Not Null | | Donor’s | Blood Group | String | Not Null | |  | Contact | String | Not Null | |  | Medical Details | String | Not Null | |  | Location | String | Not Null |  | **Table Name** | **Field Name** | **Data Type** | **Description** | | --- | --- | --- | --- | |  | User id | String | Not Null | |  | Full Name | String | Not Null | | Patient | Blood Group | String | Not Null | |  | Contact | String | Not Null | |  | Medical Details | String | Not Null | |  | Location | String | Not Null |  | **Table Name** | **Field Name** | **Data Type** | **Description** | | --- | --- | --- | --- | |  | Donor User id | String | Not Null | | Chat block | Donor Name | String | Not Null | |  | Patient User id | String | Not Null | |  | Patient Name | String | Not Null | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**System**

**Coding**

| Package Diagram    Deployment Diagram    List Tables with Attributes and Constraint.   | **Table Name** | **Field Name** | **Data Type** | **Description** | | --- | --- | --- | --- | |  | Password | String | Not Null | |  | User name | String | Not Null | | Donor User | User id | String | Not Null | |  | Full Name | String | Not Null | |  | Contact | Phone | Not Null | |  | DOB | Date | Not Null | |  | Document | image | Nit Null |  | **Table Name** | **Field Name** | **Data Type** | **Description** | | --- | --- | --- | --- | |  | Password | String | Not Null | |  | User name | String | Not Null | | Patient User | User id | String | Not Null | |  | Full Name | String | Not Null | |  | Contact | Phone | Not Null | |  | DOB | Date | Not Null | |  | Document | image | Nit Null |  | **Table Name** | **Field Name** | **Data Type** | **Description** | | --- | --- | --- | --- | |  | User id | String | Not Null | |  | Full Name | String | Not Null | | Donor’s | Blood Group | String | Not Null | |  | Contact | String | Not Null | |  | Medical Details | String | Not Null | |  | Location | String | Not Null |  | **Table Name** | **Field Name** | **Data Type** | **Description** | | --- | --- | --- | --- | |  | User id | String | Not Null | |  | Full Name | String | Not Null | | Patient | Blood Group | String | Not Null | |  | Contact | String | Not Null | |  | Medical Details | String | Not Null | |  | Location | String | Not Null |  | **Table Name** | **Field Name** | **Data Type** | **Description** | | --- | --- | --- | --- | |  | Donor User id | String | Not Null | | Chat block | Donor Name | String | Not Null | |  | Patient User id | String | Not Null | |  | Patient Name | String | Not Null |   **System**  **Coding**  Splash screen  D:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-12-14-054_com.example.bloodgenix.jpg  On-Boarding Screen  D:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-12-20-467_com.example.bloodgenix.jpgD:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-12-22-544_com.example.bloodgenix.jpg  D:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-12-24-480_com.example.bloodgenix.jpg  Welcome Page  D:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-12-28-259_com.example.bloodgenix.jpgD:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-12-39-845_com.example.bloodgenix.jpg  Dash-board  D:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-12-49-752_com.example.bloodgenix.jpgD:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-12-52-458_com.example.bloodgenix.jpg  D:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-12-58-846_com.example.bloodgenix.jpg  D:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-13-03-751_com.example.bloodgenix.jpg  Profile View  D:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-13-11-861_com.example.bloodgenix.jpgD:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-13-20-135_com.example.bloodgenix.jpg  D:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-13-42-446_com.example.bloodgenix.jpgE:\PRACTICAL SEM 6\Screenshot_2022-02-07-11-57-42-379_com.example.bloodgenix.jpg  Options form  E:\PRACTICAL SEM 6\Screenshot_2022-02-06-14-16-47-133_com.example.bloodgenix.jpgE:\PRACTICAL SEM 6\Screenshot_2022-02-06-14-12-47-910_com.example.bloodgenix.jpg  Chat  D:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-13-57-377_com.example.bloodgenix.jpgD:\FIANL YEAR PROJECT\Project images\Screenshot_2022-01-28-10-14-03-748_com.example.bloodgenix.jpg  Activities  **(User interface Code)**  (Manifest.xml)  <?xml version="1.0" encoding="utf-8"?> <manifest xmlns:android="http://schemas.android.com/apk/res/android"  package="com.example.bloodgenix">   <!--  The ACCESS\_COARSE/FINE\_LOCATION permissions are not required to use  Google Maps Android API v2, but you must specify either coarse or fine  location permissions for the "MyLocation" functionality.  -->  <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />  <uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" />  <uses-permission android:name="android.permission.INTERNET" />  <uses-permission android:name="android.permission.READ\_EXTERNAL\_STORAGE" />  <uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE"/>  <uses-permission android:name="com.google.android.providers.gsf.permisson.READ\_GSERVICES" />  <uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />   <application  android:allowBackup="true"  android:icon="@mipmap/ic\_launcher"  android:label="@string/app\_name"  android:requestLegacyExternalStorage="true"  android:roundIcon="@mipmap/ic\_launcher\_round"  android:supportsRtl="true"  android:theme="@style/Theme.BloodGeniX">   <!--  The API key for Google Maps-based APIs is defined as a string resource.  (See the file "res/values/google\_maps\_api.xml").  Note that the API key is linked to the encryption key used to sign the APK.  You need a different API key for each encryption key, including the release key that is used to  sign the APK for publishing.  You can define the keys for the debug and release targets in src/debug/ and src/release/.  -->  <meta-data  android:name="com.google.android.geo.API\_KEY"  android:value="@string/google\_maps\_key" />   <activity  android:name=".activities.Settings"  android:exported="false" />  <activity  android:name=".activities.About\_activity"  android:exported="false" />  <activity  android:name=".activities.fullProfileView"  android:exported="false" />  <activity  android:name=".activities.SetPassword"  android:exported="false" />  <activity  android:name=".activities.ForgetPassword"  android:exported="false" />  <activity  android:name=".activities.ChatActivity"  android:exported="false" />  <activity  android:name=".activities.User\_Profile\_View"  android:exported="false" />  <activity  android:name=".activities.BloodDnonorSearch"  android:exported="false" />  <activity  android:name=".activities.ProfileView"  android:exported="false" />  <activity  android:name=".activities.RecipientForm"  android:exported="false" />  <activity  android:name=".activities.DonationForm"  android:exported="false" />  <activity  android:name=".activities.DashBoard\_Screen"  android:exported="false" />  <activity  android:name=".activities.Otp\_Screen2"  android:exported="false" />  <activity  android:name=".activities.Otp\_Screen\_1"  android:exported="false" />  <activity  android:name=".activities.Signup\_Screen\_2"  android:exported="false" />  <activity  android:name=".activities.Signup\_Screen\_1"  android:exported="false" />  <activity  android:name=".activities.Welcome"  android:exported="false">   <!-- <intent-filter> -->  <!-- <action android:name="android.intent.action.MAIN" /> -->    <!-- <category android:name="android.intent.category.LAUNCHER" /> -->  <!-- </intent-filter> -->  </activity>  <activity  android:name=".activities.OnBoardingScreen"  android:exported="false" />  <activity  android:name=".activities.MainActivity"  android:exported="true">  <intent-filter>  <action android:name="android.intent.action.MAIN" />   <category android:name="android.intent.category.LAUNCHER" />  </intent-filter>  </activity>   <meta-data  android:name="preloaded\_fonts"  android:resource="@array/preloaded\_fonts" />  </application>  </manifest>  (MainActivity.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout  xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools"  xmlns:app="http://schemas.android.com/apk/res-auto"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.MainActivity">   <ImageView  android:id="@+id/appIcon"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="130dp"  android:layout\_marginTop="300dp"  android:layout\_marginEnd="130dp"  android:textAlignment="gravity"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  app:srcCompat="@drawable/ic\_app\_icon" />   <TextView  android:id="@+id/appTitle"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_marginStart="49dp"  android:layout\_marginEnd="49dp"  android:layout\_marginBottom="15dp"  android:autoText="true"  android:fontFamily="@font/montserrat\_bold"  android:text="BloodGeniX"  android:textAlignment="center"  android:textColor="@color/Header\_1"  android:textSize="45sp"  android:textStyle="bold"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent" /> </androidx.constraintlayout.widget.ConstraintLayout>  (OnBoardingScreen.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"   xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:gravity="center"  android:background="@drawable/ic\_welcomepage"  tools:context=".activities.OnBoardingScreen">   <androidx.viewpager.widget.ViewPager  android:id="@+id/sliderView"  android:layout\_width="match\_parent"  android:layout\_height="623dp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  tools:ignore="SpeakableTextPresentCheck">   </androidx.viewpager.widget.ViewPager>   <LinearLayout  android:id="@+id/dots"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_below="@+id/sliderView"  android:gravity="center"  android:orientation="horizontal"  app:layout\_constraintBottom\_toBottomOf="@+id/FinishButton"  app:layout\_constraintEnd\_toEndOf="@+id/FinishButton"  app:layout\_constraintStart\_toStartOf="@+id/FinishButton"  app:layout\_constraintTop\_toTopOf="@+id/FinishButton">   </LinearLayout>   <Button  android:id="@+id/FinishButton"  style="@style/Widget.MaterialComponents.Button.OutlinedButton.Icon"  android:layout\_width="100dp"  android:layout\_height="wrap\_content"  android:foregroundGravity="center"  android:text="@string/next"  android:textColor="@color/Header\_1"  android:visibility="invisible"  app:cornerRadius="15dp"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/sliderView"  app:strokeColor="@color/content\_2"  app:strokeWidth="2dp" />   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:src="@drawable/ic\_home\_indicator"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/FinishButton" />  </androidx.constraintlayout.widget.ConstraintLayout>  (Welcome.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@drawable/ic\_welcomepage"  tools:context=".activities.Welcome">     <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="0dp"  android:gravity="center"  android:orientation="vertical"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent">   <ImageView  android:id="@+id/imageView"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  app:srcCompat="@drawable/welcome\_screen\_img" />   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:gravity="start"  android:orientation="vertical">   <TextView  android:id="@+id/textView"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="25dp"  android:layout\_marginTop="25dp"  android:fontFamily="@font/poppins\_semibold"  android:lineSpacingExtra="-7dp"  android:text="@string/let\_s\_nget\_started"  android:textColor="@color/Header\_1"  android:textSize="28sp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/imageView" />   <TextView  android:id="@+id/textView2"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="25dp"  android:layout\_marginTop="10dp"  android:autoText="true"  android:fontFamily="@font/poppins"  android:lineSpacingExtra="-3dp"  android:text="@string/there\_is\_a\_hope\_of\_life\_to\_someone\_n\_in\_your\_blood\_donation"  android:textColor="@color/Header\_1"  android:textSize="20sp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView" />   </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="horizontal"  android:gravity="center"  android:layout\_marginTop="50dp">   <TextView  android:id="@+id/textView3"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="Already have an account?"  android:textColor="@color/content\_1"  android:textSize="18sp"  app:layout\_constraintEnd\_toStartOf="@+id/loginView"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView2" />   <Button  android:id="@+id/loginView"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:background="@android:color/transparent"  android:fontFamily="@font/poppins\_semibold"  android:paddingLeft="7dp"  android:paddingTop="2dp"  android:paddingRight="7dp"  android:paddingBottom="2dp"  android:text="@string/login"  android:textColor="@color/App\_colour"  android:textSize="18sp"  app:layout\_constraintStart\_toEndOf="@+id/textView3"  app:layout\_constraintTop\_toBottomOf="@+id/textView2"  tools:ignore="TouchTargetSizeCheck" />   </LinearLayout>    <TextView  android:id="@+id/textView6"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginTop="24dp"  android:fontFamily="@font/poppins"  android:gravity="center"  android:text="@string/or"  android:textColor="@color/Header\_1\_light"  android:textSize="18sp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView3" />   <Button  android:id="@+id/signUpBtn"  android:layout\_width="315dp"  android:layout\_height="60dp"  android:layout\_marginTop="24dp"  android:fontFamily="@font/poppins\_semibold"  android:text="@string/sign\_up"  android:textSize="18sp"  app:cornerRadius="10dp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView6" />   </LinearLayout>  </androidx.constraintlayout.widget.ConstraintLayout>  (Signup\_Screen\_1.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@drawable/ic\_profile\_details\_1"  tools:context=".activities.Signup\_Screen\_1">   <ScrollView  android:layout\_width="match\_parent"  android:layout\_height="match\_parent">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="vertical">   <TextView  android:id="@+id/textView4"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="32dp"  android:layout\_marginTop="50dp"  android:fontFamily="@font/poppins\_semibold"  android:text="@string/create\_account"  android:textColor="@color/Header\_1"  android:textSize="25sp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />   <androidx.constraintlayout.widget.ConstraintLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="50dp">   <de.hdodenhof.circleimageview.CircleImageView  android:id="@+id/profileImage"  android:layout\_width="144dp"  android:layout\_height="144dp"  android:layout\_marginStart="37dp"  android:layout\_marginEnd="37dp"  android:visibility="visible"  android:background="@drawable/profile\_img\_backgroud\_style"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />   <de.hdodenhof.circleimageview.CircleImageView  android:id="@+id/profileImage2"  android:layout\_width="144dp"  android:layout\_height="144dp"  android:layout\_marginStart="37dp"  android:layout\_marginEnd="37dp"  android:src="@drawable/ic\_profile\_image"  android:visibility="invisible"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />   <ImageButton  android:id="@+id/profilePickerBtn"  style="@style/Widget.MaterialComponents.Button.OutlinedButton.Icon"   android:layout\_width="89dp"  android:layout\_height="56dp"  android:layout\_marginStart="104dp"  android:layout\_marginTop="96dp"  android:src="@drawable/ic\_camera"  app:layout\_constraintEnd\_toEndOf="@+id/profileImage"  app:layout\_constraintStart\_toStartOf="@+id/profileImage"  app:layout\_constraintTop\_toTopOf="parent" />   </androidx.constraintlayout.widget.ConstraintLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="vertical"  android:layout\_marginStart="37dp"  android:layout\_marginTop="50dp"  android:layout\_marginEnd="37dp">    <!-- FullName-->  <com.google.android.material.textfield.TextInputLayout  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:hint="@string/full\_name"  app:helperText="Required\*"  app:helperTextTextColor="@color/red"  app:hintTextColor="#CC1919"   app:startIconDrawable="@drawable/ic\_baseline\_edit\_24">   <com.google.android.material.textfield.TextInputEditText  android:id="@+id/fullName"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:inputType="text"/>  </com.google.android.material.textfield.TextInputLayout>   <!-- UserName-->  <com.google.android.material.textfield.TextInputLayout  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  app:startIconDrawable="@drawable/ic\_baseline\_assignment\_ind\_24"  android:layout\_marginTop="24dp"  android:hint="User Name"  app:helperText="Required\*"  app:helperTextTextColor="@color/red"  app:hintTextColor="#CC1919">   <com.google.android.material.textfield.TextInputEditText  android:id="@+id/userName"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:inputType="text"/>  </com.google.android.material.textfield.TextInputLayout>   <!-- Email-->  <com.google.android.material.textfield.TextInputLayout  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="24dp"  app:startIconDrawable="@drawable/ic\_baseline\_email\_24"  android:hint="Email"  app:helperText="Required\*"  app:helperTextTextColor="@color/red"  app:hintTextColor="#CC1919">   <com.google.android.material.textfield.TextInputEditText  android:id="@+id/emailId"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:inputType="textEmailAddress"  android:fontFamily="@font/poppins" />  </com.google.android.material.textfield.TextInputLayout>   <!-- Password-->  <com.google.android.material.textfield.TextInputLayout  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="24dp"  android:hint="Password"  app:endIconMode="password\_toggle"  app:helperText="Required\*"  app:helperTextTextColor="@color/red"  app:startIconDrawable="@drawable/ic\_baseline\_lock\_24">   <com.google.android.material.textfield.TextInputEditText  android:id="@+id/password"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:inputType="textPassword" />  </com.google.android.material.textfield.TextInputLayout>  </LinearLayout>   <ImageButton  android:id="@+id/nextButton\_1"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="50dp"  app:cornerRadius="8dp"  android:layout\_marginEnd="37dp"  android:background="@android:color/transparent"  android:src="@drawable/ic\_nextbutton" />   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:src="@drawable/ic\_home\_indicator"  android:layout\_marginBottom="4dp"  android:layout\_gravity="center"/>   </LinearLayout>  </ScrollView>    </androidx.constraintlayout.widget.ConstraintLayout>  (Signup\_Screen\_2.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.Signup\_Screen\_2"  android:background="@drawable/ic\_profile\_details\_2">   <TextView  android:id="@+id/textView5"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="31dp"  android:layout\_marginTop="50dp"  android:fontFamily="@font/poppins\_semibold"  android:text="@string/create\_account"  android:textColor="@color/Header\_1"  android:textSize="25sp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />   <LinearLayout  android:id="@+id/linearLayout"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="38dp"  android:layout\_marginTop="50dp"  android:layout\_marginEnd="38dp"  android:orientation="vertical"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintHorizontal\_bias="0.0"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView5">   <TextView  android:id="@+id/textView7"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="@string/select\_date\_of\_birth"  android:textColor="@color/content\_1"  android:textSize="18sp" />    <DatePicker  android:id="@+id/Dob"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:calendarViewShown="false"  android:datePickerMode="spinner" />   <!-- Gender-->   <com.google.android.material.textfield.TextInputLayout  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense.ExposedDropdownMenu"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="3dp">   <AutoCompleteTextView  android:id="@+id/Gender"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:hint="@string/gender"  android:inputType="none" />  </com.google.android.material.textfield.TextInputLayout>    <!-- File Uploading-->  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="20dp"  android:orientation="horizontal">   <LinearLayout  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:gravity="center\_vertical"  android:orientation="horizontal">   <TextView  android:id="@+id/textView9"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_weight="1"  android:fontFamily="@font/poppins"  android:paddingEnd="10dp"  android:text="@string/identity\_proof"  android:textColor="@color/content\_1"  android:textSize="18sp" />  </LinearLayout>   <androidx.constraintlayout.widget.ConstraintLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="125dp">   <ImageButton  android:id="@+id/docSelection"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:background="@android:color/transparent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  app:srcCompat="@drawable/ic\_baseline\_add\_circle\_24" />   <ImageButton  android:id="@+id/docDeSelect"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:background="@android:color/transparent"  android:visibility="invisible"  app:layout\_constraintEnd\_toEndOf="@+id/docSelection"  app:layout\_constraintStart\_toStartOf="@+id/docSelection"  app:layout\_constraintTop\_toTopOf="parent"  app:srcCompat="@drawable/ic\_baseline\_remove\_circle\_24"  tools:visibility="invisible" />   </androidx.constraintlayout.widget.ConstraintLayout>   </LinearLayout>   <!-- Document name field -->  <androidx.cardview.widget.CardView  android:id="@+id/docLayout"  android:layout\_width="match\_parent"  android:layout\_height="50dp"  android:layout\_marginTop="14dp"  app:cardBackgroundColor="@color/cross"  android:visibility="invisible"  app:cardCornerRadius="8dp">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="horizontal"  android:gravity="center\_vertical">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="10dp"  android:src="@drawable/ic\_baseline\_picture\_as\_pdf\_24"/>   <TextView  android:id="@+id/docText"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:textColor="@color/content\_1"  android:textSize="15sp"  android:layout\_marginEnd="15dp"  android:layout\_marginStart="15dp"/>   </LinearLayout>   </androidx.cardview.widget.CardView>   </LinearLayout>   <ImageButton  android:id="@+id/nextButton\_2"  android:background="@android:color/transparent"  android:layout\_width="300dp"  android:layout\_height="wrap\_content"  android:layout\_marginTop="40dp"  android:src="@drawable/ic\_nextbutton"  app:layout\_constraintEnd\_toEndOf="@+id/linearLayout"  app:layout\_constraintStart\_toStartOf="@+id/linearLayout"  app:layout\_constraintTop\_toBottomOf="@+id/linearLayout" />   <ImageView  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:src="@drawable/ic\_home\_indicator"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="@+id/linearLayout"  app:layout\_constraintStart\_toStartOf="@+id/linearLayout" />  </androidx.constraintlayout.widget.ConstraintLayout>  (Otp\_Screen\_1.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.Otp\_Screen\_1">   <ScrollView  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:overScrollMode="never"  android:scrollbars="none"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:gravity="center\_horizontal"  android:orientation="vertical">   <ImageView  android:layout\_width="260dp"  android:layout\_height="200dp"  android:layout\_marginTop="82dp"  android:src="@drawable/ic\_ottp\_1" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginTop="32dp"  android:fontFamily="@font/poppins\_bold"  android:gravity="center"  android:text="Phone Number"  android:textColor="@color/Header\_1"  android:textSize="24sp" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="24dp"  android:layout\_marginTop="24dp"  android:layout\_marginEnd="24dp"  android:fontFamily="@font/poppins"  android:gravity="center"  android:text="@string/otp\_details"  android:textColor="@color/Header\_1"  android:textSize="16sp" />   <com.hbb20.CountryCodePicker  android:id="@+id/codePicker\_2"  android:layout\_width="272dp"  android:layout\_height="wrap\_content"  android:layout\_marginTop="30dp"  android:background="@drawable/text\_box\_style"  android:gravity="center" />    <EditText  android:id="@+id/mobileNumb"  android:layout\_width="match\_parent"  android:layout\_height="40dp"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  android:background="@null"  android:fontFamily="@font/poppins\_medium"  android:gravity="center"  android:hint="123456789"  android:imeOptions="actionDone"  android:importantForAutofill="no"  android:inputType="number"  android:textColor="@color/Header\_1"  android:textSize="20sp" />   <View  android:layout\_width="match\_parent"  android:layout\_height="2dp"  android:layout\_marginStart="70dp"  android:layout\_marginTop="5dp"  android:layout\_marginEnd="70dp"  android:background="@color/content\_1" />   <Button  android:id="@+id/getOtpBtn"  android:layout\_width="300dp"  android:layout\_height="60dp"  android:layout\_marginStart="37dp"  android:layout\_marginTop="100dp"  android:layout\_marginEnd="37dp"  android:fontFamily="@font/poppins\_semibold"  android:gravity="center"  android:text="Get OTP"  android:textSize="20dp"  app:cornerRadius="8dp" />   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:layout\_marginTop="20dp"  android:src="@drawable/ic\_home\_indicator" />  </LinearLayout>  </ScrollView>   </androidx.constraintlayout.widget.ConstraintLayout>  (Otp\_Screen\_2.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.Otp\_Screen2">   <ScrollView  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:overScrollMode="never"  android:scrollbars="none"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:gravity="center\_horizontal"  android:orientation="vertical">   <ImageView  android:layout\_width="260dp"  android:layout\_height="200dp"  android:layout\_marginTop="82dp"  android:layout\_gravity="center"  android:src="@drawable/ic\_otp\_2" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginTop="32dp"  android:fontFamily="@font/poppins\_bold"  android:gravity="center"  android:text="OTP Verification"  android:textColor="@color/Header\_1"  android:textSize="24sp" />   <LinearLayout  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="25dp"  android:layout\_marginTop="20dp"  android:layout\_marginEnd="25dp">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins\_medium"  android:text="@string/enter\_the\_otp\_sent\_to"  android:textSize="16dp" />   <TextView  android:id="@+id/dispNumber"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="10dp"  android:fontFamily="@font/poppins\_bold"  android:text="@string/\_91\_9768522052"  android:textSize="16dp" />  </LinearLayout>   <com.chaos.view.PinView  android:id="@+id/otpVal"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  style="@style/PinWidget.PinView"  android:layout\_marginTop="50dp"  android:layout\_marginStart="37dp"  android:layout\_marginEnd="37dp"  app:itemRadius="4dp"  android:cursorVisible="true"  android:inputType="number"  app:itemHeight="48dp"  app:itemCount="6"  app:lineColor="@color/App\_colour" />  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="horizontal"  android:layout\_marginTop="25dp"  android:layout\_marginStart="16dp"  android:gravity="center"  android:layout\_marginEnd="16dp">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="@string/didn\_t\_you\_receive\_the\_otp"  android:textSize="16sp"  android:textColor="@color/content\_2"  android:fontFamily="@font/poppins"/>   <Button  android:id="@+id/resendBtn"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:background="@android:color/transparent"  android:fontFamily="@font/poppins\_medium"  android:paddingStart="0dp"  android:paddingEnd="0dp"  android:text=" Resend OTP"  android:textAllCaps="false"  android:textColor="@color/App\_colour"  android:textSize="16sp" />  </LinearLayout>   <Button  android:id="@+id/verify"  android:layout\_width="match\_parent"  android:layout\_height="60dp"  android:text="Verify"  android:fontFamily="@font/poppins\_semibold"  android:textSize="20sp"  android:layout\_marginStart="37dp"  android:layout\_marginEnd="37dp"  android:layout\_marginTop="50dp"  app:cornerRadius="8sp"/>  <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_marginTop="21dp"  android:layout\_gravity="center"  android:src="@drawable/ic\_home\_indicator"/>  </LinearLayout>   </ScrollView> </androidx.constraintlayout.widget.ConstraintLayout>  (DashBoard\_Screen.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.drawerlayout.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  xmlns:local="http://schemas.android.com/apk/res-auto"  android:id="@+id/drawerLayout"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:fitsSystemWindows="true"  tools:context=".activities.DashBoard\_Screen"  tools:openDrawer="start">    <androidx.constraintlayout.widget.ConstraintLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent">   <androidx.appcompat.widget.Toolbar  android:id="@+id/toolbar"  android:layout\_width="match\_parent"  android:layout\_height="?attr/actionBarSize"  app:layout\_collapseMode="pin"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  android:background="@color/App\_colour"  local:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"  local:popupTheme="@style/ThemeOverlay.AppCompat.Light">   <TextView  android:id="@+id/toolBarHeader"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins\_medium"  android:textSize="15sp"  android:textColor="@color/white"  android:layout\_marginStart="10dp"/>   <ImageButton  android:id="@+id/moreInfo"  android:layout\_width="24dp"  android:layout\_height="24dp"  android:layout\_gravity="end"  android:layout\_marginEnd="32dp"  style="@style/SelectableItemBackground"  android:backgroundTint="@color/white"  android:src="@drawable/ic\_outline\_info\_24\_white" />   </androidx.appcompat.widget.Toolbar>   <fragment  android:id="@+id/main\_fragment"  android:name="androidx.navigation.fragment.NavHostFragment"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  app:defaultNavHost="true"  app:layout\_constraintBottom\_toTopOf="@id/bottom\_navigation"  android:layout\_marginTop="40dp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/toolbar"  app:navGraph="@navigation/bottom\_nav" />   <com.google.android.material.bottomnavigation.BottomNavigationView  android:id="@+id/bottom\_navigation"  android:layout\_width="match\_parent"  android:layout\_height="50dp"  android:background="?android:attr/windowBackground"  app:labelVisibilityMode="labeled"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintStart\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:menu="@menu/bottom\_menu" />   </androidx.constraintlayout.widget.ConstraintLayout>    <com.google.android.material.navigation.NavigationView  android:id="@+id/navView"  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_gravity="start"  android:paddingBottom="64dp"  app:headerLayout="@layout/header"  app:itemIconTint="@color/App\_colour"  app:itemTextColor="@color/App\_colour"  app:menu="@menu/main\_menu">   <include  layout="@layout/header"  android:id="@+id/layout\_1"  android:layout\_height="wrap\_content"  android:layout\_width="wrap\_content"  android:visibility="invisible"/>   <Button  style="@style/Widget.MaterialComponents.Button.Icon"  android:id="@+id/Logout"  android:layout\_width="match\_parent"  android:layout\_height="60dp"  android:layout\_gravity="bottom"  android:layout\_marginStart="20dp"  android:layout\_marginEnd="20dp"  android:elevation="5dp"  android:fontFamily="@font/poppins\_semibold"  android:gravity="center\_vertical"  android:text="Log out"  android:textAllCaps="false"  android:textSize="17sp"  app:cornerRadius="8dp"  app:icon="@drawable/ic\_signout"  app:iconGravity="end" />   </com.google.android.material.navigation.NavigationView>  </androidx.drawerlayout.widget.DrawerLayout>  (User\_Profile\_View.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.User\_Profile\_View">   <include  android:id="@+id/include"  layout="@layout/custom\_action\_bar" />   <androidx.swiperefreshlayout.widget.SwipeRefreshLayout  android:id="@+id/refreshLayout"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_marginTop="80dp"  app:layout\_constraintTop\_toBottomOf="@+id/include">   <ScrollView  android:layout\_width="match\_parent"  android:layout\_height="match\_parent">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_marginStart="24dp"  android:layout\_marginEnd="24dp"  android:orientation="vertical"  app:layout\_constraintTop\_toBottomOf="@+id/textView12">    <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="horizontal">   <LinearLayout  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content">   <de.hdodenhof.circleimageview.CircleImageView  android:id="@+id/userProfileImage"  android:layout\_width="80dp"  android:layout\_height="80dp"  android:elevation="2dp"  android:scaleType="centerCrop" />   <ImageButton  android:foreground="?attr/selectableItemBackground"  android:clickable="true"  android:id="@+id/picUpdate"  android:layout\_width="28dp"  android:layout\_height="28dp"  android:layout\_gravity="bottom"  android:layout\_marginStart="-10dp"  android:src="@drawable/ic\_camera\_update" />  </LinearLayout>   <LinearLayout  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="20dp"  android:orientation="vertical">   <TextView  android:id="@+id/userProfileName"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:fontFamily="@font/poppins"   android:textColor="@color/black"  android:textSize="18sp" />   <TextView  android:id="@+id/ProfileUserName"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:textColor="@color/black"  android:textSize="15sp" />  </LinearLayout>   </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="50dp"  android:orientation="vertical">   //BLOOD GROUP  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:src="@drawable/ic\_hearrt" />   <LinearLayout  android:layout\_width="230dp"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="18dp"  android:orientation="vertical">   <TextView  android:id="@+id/userProfileBlood"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"   android:textColor="@color/Header\_1"  android:textSize="14sp" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="Blood Group"  android:textColor="@color/Header\_1"  android:textSize="11sp" />  </LinearLayout>   <ImageButton  android:id="@+id/BloodBtn"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@android:color/transparent"  android:src="@drawable/ic\_pencil" />  </LinearLayout>   //Location  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="24dp">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:src="@drawable/ic\_location" />   <LinearLayout  android:layout\_width="230dp"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="18dp"  android:orientation="vertical">   <TextView  android:id="@+id/userProfileLocation"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:textColor="@color/Header\_1"  android:textSize="14sp" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="Location"  android:textColor="@color/Header\_1"  android:textSize="11sp" />  </LinearLayout>   <ImageButton  android:id="@+id/locationBtn"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@android:color/transparent"  android:src="@drawable/ic\_pencil" />  </LinearLayout>   //Contact  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="24dp">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:src="@drawable/ic\_call" />   <LinearLayout  android:layout\_width="230dp"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="18dp"  android:orientation="vertical">   <TextView  android:id="@+id/userProfileContact"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"   android:textColor="@color/Header\_1"  android:textSize="14sp" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="Contact"  android:textColor="@color/Header\_1"  android:textSize="11sp" />  </LinearLayout>   <ImageButton  android:id="@+id/ContactBtn"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@android:color/transparent"  android:src="@drawable/ic\_pencil" />  </LinearLayout>   //Email  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="24dp">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:src="@drawable/ic\_mail" />   <LinearLayout  android:layout\_width="230dp"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="18dp"  android:orientation="vertical">   <TextView  android:id="@+id/userProfileEmail"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"   android:textColor="@color/Header\_1"  android:textSize="14sp" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="Email"  android:textColor="@color/Header\_1"  android:textSize="11sp" />  </LinearLayout>   <ImageButton  android:id="@+id/EmailBtn"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@android:color/transparent"  android:src="@drawable/ic\_pencil" />  </LinearLayout>   //DOB  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="24dp">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:src="@drawable/ic\_dob" />   <LinearLayout  android:layout\_width="230dp"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="18dp"  android:orientation="vertical">   <TextView  android:id="@+id/userProfileDOB"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"   android:textColor="@color/Header\_1"  android:textSize="14sp" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="Birth Date"  android:textColor="@color/Header\_1"  android:textSize="11sp" />  </LinearLayout>   <ImageButton  android:id="@+id/DOBBtn"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@android:color/transparent"  android:src="@drawable/ic\_pencil" />  </LinearLayout>    //last bleed  <LinearLayout  android:id="@+id/bleedLayout"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="24dp"  android:visibility="invisible">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:src="@drawable/ic\_bleed" />   <LinearLayout  android:layout\_width="230dp"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="18dp"  android:orientation="vertical">   <TextView  android:id="@+id/userPersonBleed"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"   android:textColor="@color/Header\_1"  android:textSize="14sp" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="last Bleed"  android:textColor="@color/Header\_1"  android:textSize="11sp" />  </LinearLayout>   <ImageButton  android:id="@+id/BleedBtn"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@android:color/transparent"  android:src="@drawable/ic\_pencil" />  </LinearLayout>  </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="50dp"  android:orientation="horizontal">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:fontFamily="@font/poppins\_semibold"  android:text="Delete Choice"  android:gravity="center\_vertical"  android:textColor="@color/Header\_1"  android:textSize="18sp" />   <Button  android:id="@+id/deleteChoice"  android:layout\_width="80dp"  android:layout\_height="wrap\_content"  android:text="Delete"  android:layout\_marginLeft="105dp"  app:cornerRadius="8dp"  android:textSize="12sp"  android:fontFamily="@font/poppins\_semibold"  android:textAllCaps="false"/>   </LinearLayout>   </LinearLayout>   </ScrollView>  </androidx.swiperefreshlayout.widget.SwipeRefreshLayout>   </androidx.constraintlayout.widget.ConstraintLayout>  (DonationForm.xml)  <?xml version="1.0" encoding="utf-8"?> <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.DonationForm">    <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:paddingBottom="40dp">   <ImageButton  android:id="@+id/cancelBtn"  android:layout\_width="38dp"  android:layout\_height="36dp"  android:layout\_marginStart="18dp"  android:layout\_marginTop="10dp"  android:background="@drawable/cross\_style"  android:src="@drawable/ic\_cancel"/>   <TextView  android:id="@+id/textView10"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="30dp"  android:layout\_marginTop="45dp"  android:fontFamily="@font/poppins\_semibold"  android:text="@string/blood\_donation\_form"  android:textColor="@color/Header\_1"  android:textSize="24sp" />   <com.google.android.material.textfield.TextInputLayout  android:id="@+id/bloodGroupLayout"  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense.ExposedDropdownMenu"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="40dp"  android:layout\_marginEnd="37dp"  app:errorEnabled="false"  app:helperText="Required\*"  app:helperTextTextColor="@color/red">   <AutoCompleteTextView  android:id="@+id/bloodGroup"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:hint="@string/blood\_group"  android:inputType="none" />  </com.google.android.material.textfield.TextInputLayout>    <com.google.android.material.textfield.TextInputLayout  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  android:hint="Location"  app:helperText="Required\*"  app:helperTextTextColor="@color/red">   <com.google.android.material.textfield.TextInputEditText  android:id="@+id/locationDonation"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:inputType="none"/>   </com.google.android.material.textfield.TextInputLayout>    <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  android:fontFamily="@font/poppins"  android:text="Are you Diabetic?"  android:textColor="@color/content\_2"  android:textSize="18sp" />   <RadioGroup  android:id="@+id/radioGroup"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="10dp"  android:layout\_marginEnd="37dp"  android:orientation="horizontal">   <RadioButton  android:id="@+id/yes"  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_marginStart="10dp"  android:text="@string/yes"  android:textColor="@color/content\_1"  android:textSize="13sp" />   <RadioButton  android:id="@+id/no"  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_marginStart="100dp"  android:text="@string/no"  android:textColor="@color/content\_1"  android:textSize="13sp" />  </RadioGroup>   <View  android:layout\_width="wrap\_content"  android:layout\_height="2dp"  android:layout\_marginStart="37dp"  android:layout\_marginTop="2dp"  android:layout\_marginEnd="37dp"  android:background="@color/Header\_1" />   <com.google.android.material.textfield.TextInputLayout  android:id="@+id/diabeticCountLayout"  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense.ExposedDropdownMenu"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  app:helperText="Required"  app:helperTextTextColor="@color/red"  >   <AutoCompleteTextView  android:id="@+id/diabeticCount"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:hint="@string/diabetic\_count"  android:inputType="none" />   </com.google.android.material.textfield.TextInputLayout>   <com.google.android.material.textfield.TextInputLayout  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  app:helperText="Required"  app:helperTextTextColor="@color/red">   <com.google.android.material.textfield.TextInputEditText  android:id="@+id/otherSpecify"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:hint="@string/any\_other\_disease\_please\_specify"  android:inputType="textMultiLine" />  </com.google.android.material.textfield.TextInputLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="horizontal"  android:layout\_marginStart="37dp"  android:layout\_marginEnd="37dp">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="@string/weight"  android:textSize="18sp"  android:layout\_marginStart="4dp"  android:fontFamily="@font/poppins"  android:textColor="@color/content\_2"  android:layout\_gravity="center\_vertical"/>   <NumberPicker  android:id="@+id/weight"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="40dp"  android:theme="@style/NumberPickerStyle"/>  <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Kg"  android:textColor="@color/content\_1"  android:textSize="18sp"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="15dp"/>   </LinearLayout>   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  android:fontFamily="@font/poppins"  android:text="@string/did\_you\_previously\_donated"  android:textColor="@color/content\_2"  android:textSize="18sp" />   <RadioGroup  android:id="@+id/radioGroup2"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="10dp"  android:layout\_marginEnd="37dp"  android:orientation="horizontal">   <RadioButton  android:id="@+id/yes2"  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_marginStart="10dp"  android:text="@string/yes"  android:textColor="@color/content\_1"  android:textSize="13sp" />   <RadioButton  android:id="@+id/no2"  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_marginStart="100dp"  android:text="@string/no"  android:textColor="@color/content\_1"  android:textSize="13sp" />  </RadioGroup>   <View  android:layout\_width="wrap\_content"  android:layout\_height="2dp"  android:layout\_marginStart="37dp"  android:layout\_marginTop="2dp"  android:layout\_marginEnd="37dp"  android:background="@color/Header\_1" />    <com.google.android.material.textfield.TextInputLayout  android:id="@+id/PeriodLayout"  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense.ExposedDropdownMenu"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  app:helperText="Required"  app:helperTextTextColor="@color/red">   <AutoCompleteTextView  android:id="@+id/DonationPeriod"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:hint="@string/mention\_month\_passed"  android:inputType="none" />   </com.google.android.material.textfield.TextInputLayout>   <CheckBox  android:id="@+id/authorise"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  android:fontFamily="@font/poppins"  android:gravity="start"  android:onClick="checkAuthorise"  android:text="@string/authorizeValue"  android:textAlignment="textStart"  android:textColor="@color/content\_1"  android:textSize="15sp" />   <Button  android:id="@+id/applyDonation"  android:layout\_width="297dp"  android:layout\_height="60dp"  android:layout\_marginStart="37dp"  android:layout\_marginTop="100dp"  android:layout\_marginEnd="37dp"  android:defaultFocusHighlightEnabled="false"  android:drawableEnd="@drawable/ic\_baseline\_arrow\_forward\_24"  android:elevation="2dp"  android:enabled="false"  android:fontFamily="@font/poppins\_semibold"  android:text="@string/apply\_as\_donor"  android:textAlignment="textStart"  android:textAllCaps="false"  app:cornerRadius="8dp" />  </LinearLayout>   </ScrollView>  (RecipientForm.xml)  <?xml version="1.0" encoding="utf-8"?> <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.RecipientForm"  android:orientation="vertical">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:paddingBottom="40dp">   <ImageButton  android:id="@+id/cancelBtnRecipient"  android:layout\_width="38dp"  android:layout\_height="36dp"  android:layout\_marginStart="18dp"  android:layout\_marginTop="10dp"  android:background="@drawable/cross\_style"  android:src="@drawable/ic\_cancel" />   <TextView  android:id="@+id/textView10"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="30dp"  android:layout\_marginTop="45dp"  android:fontFamily="@font/poppins\_semibold"  android:text="Blood request form"  android:textColor="@color/Header\_1"  android:textSize="24sp" />   <com.google.android.material.textfield.TextInputLayout  android:id="@+id/bloodGroupLayout"  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense.ExposedDropdownMenu"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="40dp"  android:layout\_marginEnd="37dp"  app:errorEnabled="false"  app:helperText="Required\*"  app:helperTextTextColor="@color/red">   <AutoCompleteTextView  android:id="@+id/bloodGroupRecipient"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:hint="@string/blood\_group"  android:inputType="none" />  </com.google.android.material.textfield.TextInputLayout>    <com.google.android.material.textfield.TextInputLayout  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  android:hint="Location"  app:helperText="Required\*"  app:helperTextTextColor="@color/red">   <com.google.android.material.textfield.TextInputEditText  android:id="@+id/locationRecipient"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:editable="false"  android:inputType="none"/>   </com.google.android.material.textfield.TextInputLayout>   <com.google.android.material.textfield.TextInputLayout  android:id="@+id/reasonBloodLayout"  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense.ExposedDropdownMenu"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  app:errorEnabled="false"  app:helperText="Required\*"  app:helperTextTextColor="@color/red">   <AutoCompleteTextView  android:id="@+id/reasonBlood"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:hint="@string/what\_s\_the\_reason\_of\_this\_request"  android:inputType="none" />  </com.google.android.material.textfield.TextInputLayout>   <com.google.android.material.textfield.TextInputLayout  style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="30dp"  android:layout\_marginEnd="37dp"  android:hint="Write a message for potential donors"  app:helperText="Required\*"  app:helperTextTextColor="@color/red">   <com.google.android.material.textfield.TextInputEditText  android:id="@+id/requirementRecipient"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:inputType="textMultiLine"/>   </com.google.android.material.textfield.TextInputLayout>   <CheckBox  android:id="@+id/authoriseRecipient"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="50dp"  android:layout\_marginEnd="37dp"  android:fontFamily="@font/poppins"  android:gravity="start"  android:onClick="authorise"  android:text="@string/authorizeValue"  android:textAlignment="textStart"  android:textColor="@color/content\_1"  android:textSize="15sp" />   <Button  android:id="@+id/applyRecipient"  android:layout\_width="297dp"  android:layout\_height="60dp"  android:layout\_marginStart="37dp"  android:layout\_marginTop="80dp"  android:layout\_marginEnd="37dp"  android:defaultFocusHighlightEnabled="false"  android:drawableEnd="@drawable/ic\_baseline\_arrow\_forward\_24"  android:elevation="2dp"  android:enabled="false"  android:fontFamily="@font/poppins\_semibold"  android:text="Apply as Recipient"  android:textAlignment="textStart"  android:textAllCaps="false"  app:cornerRadius="8dp" />    </LinearLayout>   </ScrollView>  (BloodDonorSearch.xml)  <?xml version="1.0" encoding="utf-8"?> <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.BloodDnonorSearch">   <EditText  android:id="@+id/searchBar"  android:layout\_width="match\_parent"  android:layout\_height="60dp"  android:layout\_marginStart="25dp"  android:layout\_marginTop="50dp"  android:layout\_marginEnd="25dp"  android:background="@drawable/searchbar\_background\_style"  android:drawableStart="@drawable/ic\_baseline\_search\_24"  android:hint="Search by location"  android:inputType="text"  android:padding="16dp"  android:paddingStart="16dp" />    <androidx.recyclerview.widget.RecyclerView  android:id="@+id/searchRecycler"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_below="@+id/searchBar"  android:layout\_marginStart="16dp"  android:layout\_marginTop="50dp"  android:layout\_marginEnd="16dp"  android:padding="10dp"  app:layoutManager="androidx.recyclerview.widget.LinearLayoutManager">    </androidx.recyclerview.widget.RecyclerView> </RelativeLayout>  (ProfileView.xml)  <?xml version="1.0" encoding="utf-8"?> <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.ProfileView">   <LinearLayout  android:layout\_marginTop="30dp"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical">   <ImageButton  android:id="@+id/cancelBtnProfile"  android:layout\_width="38dp"  android:layout\_height="36dp"  android:layout\_marginStart="18dp"  android:background="@drawable/cross\_style"  android:src="@drawable/ic\_cancel" />   <de.hdodenhof.circleimageview.CircleImageView  android:id="@+id/profileImageView"  android:layout\_width="144dp"  android:layout\_height="144dp"  android:layout\_gravity="center"  android:layout\_marginTop="25dp" />   <LinearLayout  android:layout\_marginTop="15dp"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="horizontal"  android:gravity="center">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:src="@drawable/ic\_success"/>  <TextView  android:id="@+id/textView15"  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_gravity="center"  android:fontFamily="@font/poppins\_medium"  android:text="@string/verified"  android:layout\_marginStart="5dp"  android:textColor="@color/verified"  android:textSize="20sp"  android:gravity="center\_vertical"/>  </LinearLayout>   <LinearLayout  android:id="@+id/linearLayout4"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:layout\_marginTop="28dp"  android:layout\_marginEnd="20dp"  android:gravity="center"  android:padding="15dp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView15">   <ImageButton  android:id="@+id/directMessage"  android:layout\_width="30dp"  android:layout\_height="match\_parent"  android:layout\_marginStart="-10dp"  android:background="@android:color/transparent"  android:src="@drawable/ic\_directmessage" />   <ImageButton  android:id="@+id/block"  android:layout\_width="30dp"  android:layout\_height="match\_parent"  android:layout\_marginStart="45dp"  android:background="@android:color/transparent"  android:src="@drawable/ic\_block" />   <ImageButton  android:id="@+id/report"  android:layout\_width="30dp"  android:layout\_height="match\_parent"  android:layout\_marginStart="45dp"  android:background="@android:color/transparent"  android:src="@drawable/ic\_report" />  </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="37dp"  android:layout\_marginTop="50dp"  android:layout\_marginEnd="37dp"  android:orientation="vertical"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/linearLayout4">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="vertical">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="Name"  android:layout\_marginStart="5dp"  android:textColor="@color/Header\_1"  android:textSize="14sp" />   <EditText  android:id="@+id/personName"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:enabled="false"  android:fontFamily="@font/poppins\_medium"  android:inputType="none"  android:layout\_marginTop="-8dp"  android:textColor="@color/Header\_1"  android:textSize="18sp" />  </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="24dp"  android:orientation="vertical">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="Blood Group"  android:layout\_marginStart="5dp"  android:textColor="@color/Header\_1"  android:textSize="14sp" />   <EditText  android:id="@+id/personBlood"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:enabled="false"  android:fontFamily="@font/poppins\_medium"  android:inputType="none"  android:layout\_marginTop="-8dp"  android:textColor="@color/Header\_1"  android:textSize="18sp" />  </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="24dp"  android:orientation="vertical">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins"  android:text="Gender"  android:layout\_marginStart="5dp"  android:textColor="@color/Header\_1"  android:textSize="14sp" />   <EditText  android:id="@+id/personGender"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_marginTop="-8dp"  android:enabled="false"  android:fontFamily="@font/poppins\_medium"  android:inputType="none"  android:textColor="@color/Header\_1"  android:textSize="18sp" />  </LinearLayout>  </LinearLayout>  </LinearLayout>  </ScrollView>  (ChatActivity.xml)  <?xml version="1.0" encoding="utf-8"?> <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.ChatActivity">    <RelativeLayout  android:id="@+id/chatToolBar"  android:layout\_width="match\_parent"  android:layout\_height="?android:attr/actionBarSize"  android:background="@color/App\_colour">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_alignParentStart="true"  android:layout\_alignParentTop="true"  android:layout\_alignParentBottom="true"  android:layout\_marginStart="0dp"  android:layout\_marginTop="0dp"  android:layout\_marginBottom="0dp">   <ImageButton  android:id="@+id/chatBackBtn"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="15dp"  android:background="@android:color/transparent"  android:src="@drawable/ic\_baseline\_arrow\_back\_24\_white" />   <de.hdodenhof.circleimageview.CircleImageView  android:id="@+id/profileImg"  android:layout\_width="40dp"  android:layout\_height="40dp"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="10dp" />   <ImageView  android:id="@+id/activeStatus"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignTop="@+id/profileImg"  android:layout\_alignParentBottom="true"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="5dp"  android:layout\_toEndOf="@+id/profileImg"  android:src="@drawable/ic\_active"  android:visibility="invisible"/>   <TextView  android:id="@+id/receiverName"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  android:layout\_marginStart="5dp"  android:fontFamily="@font/poppins\_semibold"  android:textColor="@color/white"  android:textSize="16sp" />   </LinearLayout>   <ImageButton  android:id="@+id/dotMenuBtn"  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_alignParentEnd="true"  android:layout\_marginEnd="20dp"  android:background="@android:color/transparent"  android:src="@drawable/ic\_baseline\_more\_vert\_24" />  </RelativeLayout>    <androidx.recyclerview.widget.RecyclerView  android:id="@+id/messageAdapter"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_above="@id/ll3"  android:layout\_below="@id/chatToolBar" />  <LinearLayout  android:id="@+id/ll3"  android:layout\_alignParentBottom="true"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_margin="5dp">   <androidx.cardview.widget.CardView  android:layout\_weight="1"  android:layout\_width="match\_parent"  android:layout\_height="40dp"  app:cardBackgroundColor="@color/irregular\_shape\_colour"  app:cardCornerRadius="990dp"  android:layout\_margin="5dp">   <EditText  android:id="@+id/textMessage"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:hint="Type your message ..."  android:background="@null"  android:paddingStart="10dp"  android:paddingLeft="10dp"  android:fontFamily="@font/poppins"  android:textSize="14sp"  android:layout\_gravity="center\_vertical"/>  </androidx.cardview.widget.CardView>   <androidx.cardview.widget.CardView  android:id="@+id/sendBtn"  android:layout\_width="40dp"  android:layout\_height="40dp"  app:cardBackgroundColor="@color/App\_colour"  app:cardCornerRadius="990dp"  android:layout\_margin="5dp"  android:layout\_gravity="center\_vertical">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:background="@drawable/ic\_baseline\_send\_24"  android:layout\_gravity="center" />   </androidx.cardview.widget.CardView>  </LinearLayout>    </RelativeLayout>  (About\_activity.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.About\_activity">   <RelativeLayout  android:id="@+id/relativeLayout3"  android:layout\_width="match\_parent"  android:layout\_height="?attr/actionBarSize"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  android:background="@color/App\_colour">   <ImageButton  android:id="@+id/aboutBack"  style="@style/SelectableItemBackground"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignParentStart="true"  android:layout\_alignParentTop="true"  android:layout\_alignParentBottom="true"  android:layout\_marginStart="8dp"  android:background="@android:color/transparent"  android:src="@drawable/ic\_baseline\_arrow\_back\_24\_white" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignParentTop="true"  android:layout\_marginStart="10dp"  android:layout\_marginTop="12dp"  android:layout\_toEndOf="@+id/aboutBack"  android:fontFamily="@font/poppins\_semibold"  android:text="About Us"  android:textColor="@color/white"  android:textSize="18sp" />  </RelativeLayout>    <TextView  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="@string/aboutUs"  android:layout\_marginTop="24sp"  android:textSize="17sp"  android:padding="12dp"  android:textColor="@color/content\_1"  app:layout\_constraintTop\_toBottomOf="@+id/relativeLayout3" />  </androidx.constraintlayout.widget.ConstraintLayout>  (Settings.xml)  <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".activities.Settings">   <RelativeLayout  android:id="@+id/relativeLayout2"  android:layout\_width="match\_parent"  android:layout\_height="50dp"  android:background="@color/App\_colour"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent">   <ImageView  android:id="@+id/backButtons"  style="@style/SelectableItemBackground"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignParentStart="true"  android:layout\_alignParentTop="true"  android:layout\_marginStart="5dp"  android:layout\_marginTop="10dp"  android:src="@drawable/ic\_baseline\_arrow\_back\_24\_white" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignParentTop="true"  android:layout\_marginStart="18dp"  android:layout\_marginTop="10dp"  android:layout\_toEndOf="@+id/backButtons"  android:fontFamily="@font/poppins\_medium"  android:text="Settings"  android:textColor="@color/white"  android:textSize="18sp" />   </RelativeLayout>   <LinearLayout  android:id="@+id/linearLayout5"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="15dp"  android:layout\_marginTop="15dp"  android:layout\_marginEnd="15dp"  android:orientation="vertical"  android:paddingBottom="40dp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/relativeLayout2">   <LinearLayout  android:id="@+id/FAQLayout"  style="@style/SelectableItemBackground"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_marginTop="15dp"  android:orientation="horizontal"  android:paddingTop="15dp"  android:paddingBottom="15dp"  app:cornerRadius="8dp">   <ImageView  android:layout\_width="26dp"  android:layout\_height="26dp"  android:layout\_marginStart="5dp"  android:src="@drawable/ic\_faq" />   <TextView  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:fontFamily="@font/poppins\_medium"  android:gravity="center\_vertical"  android:text="FAQ's"  android:textColor="@color/content\_1"  android:textSize="18sp" />  </LinearLayout>   <LinearLayout  android:id="@+id/helpLayout"  style="@style/SelectableItemBackground"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_marginTop="10dp"  android:orientation="horizontal"  android:paddingTop="15dp"  android:paddingBottom="15dp">   <ImageView  android:layout\_width="26dp"  android:layout\_height="26dp"  android:layout\_marginStart="5dp"  android:src="@drawable/ic\_baseline\_live\_help\_24" />   <TextView  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:fontFamily="@font/poppins\_medium"  android:gravity="center\_vertical"  android:text="Help"  android:textColor="@color/content\_1"  android:textSize="18sp" />  </LinearLayout>  </LinearLayout>   <LinearLayout  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginBottom="60dp"  android:orientation="vertical"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:fontFamily="@font/poppins\_medium"  android:text="from"  android:textColor="@color/content\_2"  android:textSize="14sp" />   <LinearLayout  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_marginTop="-8dp">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:src="@drawable/ic\_baseline\_water\_drop\_24"/>   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:fontFamily="@font/poppins\_medium"  android:gravity="center"  android:text="BloodGenix"  android:textColor="@color/black"  android:textSize="16sp" />  </LinearLayout>   </LinearLayout> </androidx.constraintlayout.widget.ConstraintLayout>  Menu  (bottom\_menu.xml)  <?xml version="1.0" encoding="utf-8"?>  <menu xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto">   <item  android:id="@+id/home"  android:icon="@drawable/ic\_baseline\_home\_24"  android:title="Home"  app:cnb\_backgroundColor="@color/irregular\_shape\_colour" />   <item  android:id="@+id/notification"  android:icon="@drawable/ic\_baseline\_notifications\_24"  android:title="Notification"  app:cnb\_backgroundColor="@color/irregular\_shape\_colour" />   <item  android:id="@+id/map"  android:icon="@drawable/ic\_baseline\_location\_on\_24"  android:title="Map"  app:cnb\_backgroundColor="@color/irregular\_shape\_colour" />   </menu>  (chat\_menu.xml)  <?xml version="1.0" encoding="utf-8"?> <menu xmlns:android="http://schemas.android.com/apk/res/android">   <item  android:id="@+id/BlockUser"  android:title="Block User">  </item>   </menu>  (dot\_menu.xml)  <?xml version="1.0" encoding="utf-8"?> <menu xmlns:android="http://schemas.android.com/apk/res/android">   <item  android:id="@+id/forgotPassword"  android:title="Forgot Password" /> </menu>  (main\_menu.xml)  <?xml version="1.0" encoding="utf-8"?> <menu xmlns:android="http://schemas.android.com/apk/res/android">     <item android:id="@+id/profile\_view"  android:title="@string/profile"  android:icon="@drawable/ic\_baseline\_person\_outline\_24"/>   <item android:id="@+id/nav\_settings"  android:title="@string/settings"  android:icon="@drawable/ic\_setting"/>   <item android:id="@+id/nav\_about"  android:title="@string/about\_us"  android:icon="@drawable/ic\_about"/>   <item  android:id="@+id/nav\_sharing"  android:icon="@drawable/ic\_baseline\_share\_24"  android:title="@string/sharing" />   </menu>  Fragment ui  (fragment\_home.xml)  <?xml version="1.0" encoding="utf-8"?> <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:id="@+id/frameLayout"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".ui.home.HomeFragment">   <!-- *TODO: Update blank fragment layout* -->  <androidx.coordinatorlayout.widget.CoordinatorLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@color/App\_colour"  android:paddingBottom="5dp">   <com.google.android.material.appbar.AppBarLayout  android:id="@+id/AppBar"  style="@style/Theme.BloodGeniX"  android:layout\_width="match\_parent"  android:layout\_height="180dp"  android:fitsSystemWindows="true"  android:outlineAmbientShadowColor="@color/App\_colour"  android:outlineSpotShadowColor="@color/App\_colour">   <com.google.android.material.appbar.CollapsingToolbarLayout  android:id="@+id/collapsingToolbar"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:fitsSystemWindows="true"  app:contentScrim="?attr/colorPrimary"  app:layout\_scrollFlags="scroll|snap|exitUntilCollapsed">  <!-- <ImageButton--> <!-- android:id="@+id/moreInfo"--> <!-- android:layout\_width="wrap\_content"--> <!-- android:layout\_height="wrap\_content"--> <!-- android:layout\_gravity="end"--> <!-- android:layout\_marginEnd="32dp"--> <!-- style="@style/SelectableItemBackground"--> <!-- android:backgroundTint="@color/white"--> <!-- android:src="@drawable/ic\_outline\_info\_24\_white" />-->   <TextView  android:id="@+id/salutationText"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="32dp"  android:layout\_marginTop="40dp"  android:fontFamily="@font/poppins"  android:text="@string/salutation"  android:textColor="@color/white"  android:textSize="20sp"  app:layout\_constraintStart\_toStartOf="parent" />   <de.hdodenhof.circleimageview.CircleImageView  android:id="@+id/active\_person"  android:layout\_width="60dp"  android:layout\_height="60dp"  android:layout\_gravity="end"  android:layout\_marginTop="40dp"  android:layout\_marginEnd="25dp" />   <ImageView  android:layout\_width="12dp"  android:layout\_height="12dp"  android:layout\_gravity="end"  android:layout\_marginTop="85dp"  android:layout\_marginEnd="25dp"  android:src="@drawable/ic\_active" />   </com.google.android.material.appbar.CollapsingToolbarLayout>  </com.google.android.material.appbar.AppBarLayout>   <androidx.core.widget.NestedScrollView  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@drawable/stats\_style"  android:padding="20dp"  app:layout\_behavior="com.google.android.material.appbar.AppBarLayout$ScrollingViewBehavior">   <ScrollView  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:paddingBottom="30dp">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:paddingBottom="5dp">   //Donation Card view  <RelativeLayout  android:id="@+id/relativeLayout"  android:layout\_width="match\_parent"  android:layout\_height="190dp"  android:layout\_marginTop="10dp"  android:background="@android:color/transparent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent">   <RelativeLayout  android:layout\_width="match\_parent"  android:layout\_height="170dp"  android:layout\_marginTop="20dp"  android:background="@drawable/card\_style">   <TextView  android:id="@+id/reasonBloodRequest"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="13dp"  android:layout\_marginTop="13dp"  android:fontFamily="@font/poppins\_semibold"  android:text="@string/apply\_for\_donation"  android:textSize="18sp" />   <TextView  android:id="@+id/applyDetail"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_below="@+id/reasonBloodRequest"  android:layout\_alignStart="@+id/reasonBloodRequest"  android:layout\_marginTop="10dp"  android:gravity="start"  android:text="Your little effort can give others\nsecond chance to live life."  android:textSize="14sp" />   <Button  android:id="@+id/donationExpand"  style="@style/SelectableItemBackground"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_below="@+id/applyDetail"  android:layout\_marginTop="16dp"  android:fontFamily="@font/poppins\_bold"  android:text="EXPAND"  android:textColor="@color/Header\_1"  android:textSize="12sp" />   </RelativeLayout>   <ImageView  android:layout\_width="130dp"  android:layout\_height="150dp"  android:layout\_alignParentRight="true"  android:paddingStart="10dp"  android:paddingRight="-27dp"  android:src="@drawable/ic\_donationsymbol" />  </RelativeLayout>    //Receiving Card View  <RelativeLayout  android:id="@+id/relative2"  android:layout\_width="match\_parent"  android:layout\_height="190dp"  android:layout\_marginTop="20dp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/relativeLayout">   <RelativeLayout  android:layout\_width="match\_parent"  android:layout\_height="170dp"  android:layout\_alignParentTop="true"  android:layout\_marginTop="20dp"  android:background="@drawable/card\_style">   <TextView  android:id="@+id/receiveView"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignParentStart="true"  android:layout\_marginStart="13dp"  android:layout\_marginTop="13dp"  android:fontFamily="@font/poppins\_semibold"  android:text="Apply for Recipient"  android:textSize="18sp" />   <TextView  android:id="@+id/receiveDetail"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_below="@+id/receiveView"  android:layout\_alignStart="@+id/receiveView"  android:layout\_marginTop="10dp"  android:gravity="start"  android:text="Find blood donors near your\nlocation and request the needed\nblood type."  android:textSize="14sp" />   <Button  android:id="@+id/RecipientExpand"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_below="@+id/receiveDetail"  android:layout\_marginTop="14dp"  style="@style/SelectableItemBackground"  android:fontFamily="@font/poppins\_bold"  android:text="EXPAND"  android:textColor="@color/Header\_1"  android:textSize="12sp" />   </RelativeLayout>   <ImageView  android:layout\_width="130dp"  android:layout\_height="150dp"  android:layout\_alignParentRight="true"  android:layout\_marginTop="-21dp"  android:src="@drawable/ic\_recipientapply" />  </RelativeLayout>   //Search Donor card View  <RelativeLayout  android:layout\_width="match\_parent"  android:layout\_height="180dp"  android:layout\_marginTop="15dp"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/relative2">   <RelativeLayout  android:layout\_width="match\_parent"  android:layout\_height="170dp"  android:layout\_alignParentTop="true"  android:layout\_marginTop="20dp"  android:background="@drawable/card\_style">   <TextView  android:id="@+id/searchView"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignParentStart="true"  android:layout\_marginStart="13dp"  android:layout\_marginTop="13dp"  android:fontFamily="@font/poppins\_semibold"  android:text="Search Donor's"  android:textSize="18sp" />   <TextView  android:id="@+id/searchDetail"  android:layout\_width="210dp"  android:layout\_height="wrap\_content"  android:layout\_below="@+id/searchView"  android:layout\_alignStart="@+id/searchView"  android:layout\_marginTop="10dp"  android:gravity="start"  android:text="Search donor matching to blood\nand contact with them"  android:textSize="14sp" />   <Button  android:id="@+id/searchExpand"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_below="@+id/searchDetail"  android:layout\_marginTop="14dp"  style="@style/SelectableItemBackground"  android:fontFamily="@font/poppins\_bold"  android:text="EXPAND"  android:textColor="@color/Header\_1"  android:textSize="12sp" />   </RelativeLayout>   <ImageView  android:layout\_width="130dp"  android:layout\_height="150dp"  android:layout\_alignParentRight="true"  android:layout\_marginTop="-4dp"  android:src="@drawable/ic\_searchdonors" />  </RelativeLayout>   </LinearLayout>  </ScrollView>   </androidx.core.widget.NestedScrollView>   </androidx.coordinatorlayout.widget.CoordinatorLayout>  </FrameLayout>  (fragment\_notification.xml)  <?xml version="1.0" encoding="utf-8"?> <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:id="@+id/frameLayout"  tools:context=".ui.notification.NotificationFragment">   <androidx.swiperefreshlayout.widget.SwipeRefreshLayout  android:id="@+id/refreshLayout"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  app:layout\_constraintTop\_toBottomOf="@+id/frameLayout">   <!-- *TODO: Update blank fragment layout* -->  <androidx.recyclerview.widget.RecyclerView  android:id="@+id/chatsBlockView"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:layout\_marginEnd="20dp"  android:padding="15dp">   </androidx.recyclerview.widget.RecyclerView>  </androidx.swiperefreshlayout.widget.SwipeRefreshLayout>  </FrameLayout>  (fragment\_profile.xml)  <?xml version="1.0" encoding="utf-8"?> <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:layout\_marginTop="35dp"  tools:context=".ui.profile.ProfileFragment">   <!-- *TODO: Update blank fragment layout* -->  <RelativeLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content">   <EditText  android:id="@+id/location\_search"  android:layout\_width="314dp"  android:layout\_height="50dp"  android:layout\_alignParentStart="true"  android:layout\_alignParentTop="true"  android:ems="10"  android:hint="Write any place name..." />   <ImageButton  android:id="@+id/btn\_find"  android:layout\_width="wrap\_content"  style="@style/SelectableItemBackground"  android:layout\_height="wrap\_content"  android:layout\_alignParentTop="true"  android:layout\_alignParentEnd="true"  android:layout\_marginEnd="15dp"  android:background="@android:color/transparent"  android:src="@drawable/ic\_baseline\_search\_24\_formap" />   </RelativeLayout>   <fragment  xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:id="@+id/google\_map"  android:name="com.google.android.gms.maps.SupportMapFragment"  android:layout\_alignParentTop="true"  android:layout\_alignParentStart="true"  android:layout\_marginTop="20dp"/>   </LinearLayout>  Navigation  (Bottom\_nav.xml)  <?xml version="1.0" encoding="utf-8"?> <navigation xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:id="@+id/bottom\_nav"  app:startDestination="@id/home">   <fragment android:id="@+id/home"  android:label="Home"  android:name="com.example.bloodgenix.ui.home.HomeFragment"  tools:layout="@layout/fragment\_home"/>   <fragment android:id="@+id/notification"  android:label="Notification"  android:name="com.example.bloodgenix.ui.notification.NotificationFragment"  tools:layout="@layout/fragment\_notification"/>   <fragment android:id="@+id/map"  android:label="Map"  android:name="com.example.bloodgenix.ui.profile.ProfileFragment"  tools:layout="@layout/fragment\_profile"/>  </navigation>  Activities java  (MainActivity.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.os.Bundle; import android.os.Handler; import android.view.animation.Animation; import android.view.animation.AnimationUtils; import android.widget.ImageView; import android.widget.TextView;  import androidx.appcompat.app.AppCompatActivity;  import com.example.bloodgenix.R; import com.example.bloodgenix.SessionManager;  import java.util.HashMap;  public class MainActivity extends AppCompatActivity {   private static int *SPLASH\_SCREEN*=5000;  //Defining Attributes  TextView appTitle;  ImageView appIcon;  Animation topAnim, bottomAnim;  String phone="";  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);   setContentView(R.layout.*activity\_main*);   appTitle = findViewById(R.id.*appTitle*);  appIcon = findViewById(R.id.*appIcon*);   topAnim = AnimationUtils.*loadAnimation*(this,R.anim.*top\_animation*);  bottomAnim = AnimationUtils.*loadAnimation*(this,R.anim.*bottom\_animation*);   appIcon.setAnimation(topAnim);  appTitle.setAnimation(bottomAnim);   new Handler().postDelayed(new Runnable() {  @Override  public void run() {  SessionManager sessionManager = new SessionManager(MainActivity.this,"userLoginSession");  HashMap<String, String> userDetails = sessionManager.getUserDetailsFromSession();  phone = userDetails.get(SessionManager.*KEY\_PHONENUMBER*);  String IS\_login = userDetails.get(SessionManager.*IS\_LOGIN*);  if (phone == null){  Intent intent = new Intent(MainActivity.this, OnBoardingScreen.class);  startActivity(intent);  }  else{  Intent dashBoard = new Intent(MainActivity.this, DashBoard\_Screen.class);  dashBoard.putExtra("profile Values","+91 "+phone);  startActivity(dashBoard);  } // else if (sessionManager.checkLogin()){ // // Toast.makeText(MainActivity.this, IS\_login, Toast.LENGTH\_SHORT).show(); // // }  }  },*SPLASH\_SCREEN*);   } }  (AboutActivity.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.ImageButton;  import androidx.appcompat.app.AppCompatActivity;  import com.example.bloodgenix.R; import com.example.bloodgenix.SessionManager;  import java.util.HashMap;  public class About\_activity extends AppCompatActivity {   ImageButton aboutBack;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_about*);   SessionManager sessionManager = new SessionManager(this,"userLoginSession");  HashMap<String, String> userDetails = sessionManager.getUserDetailsFromSession();  String fullNumber = userDetails.get(SessionManager.*KEY\_PHONENUMBER*);    aboutBack = findViewById(R.id.*aboutBack*);  aboutBack.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent backDash = new Intent(About\_activity.this, DashBoard\_Screen.class);  backDash.putExtra("profile Values","+91 "+fullNumber);  startActivity(backDash);  finish();  }  });  } }  (BloodDonorSearch.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.os.Bundle; import android.text.Editable; import android.text.TextWatcher; import android.widget.EditText; import android.widget.ImageButton; import android.widget.Toast;  import androidx.appcompat.app.AppCompatActivity; import androidx.recyclerview.widget.RecyclerView;  import com.example.bloodgenix.myAdapter; import com.example.bloodgenix.Models.DonationDetails; import com.example.bloodgenix.R; import com.firebase.ui.database.FirebaseRecyclerOptions; import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query;  import java.util.ArrayList;  public class BloodDnonorSearch extends AppCompatActivity {   EditText searchBar;  RecyclerView searchRecycler;  String text, myNumber, bloodGroupSearch;  ArrayList<DonationDetails> donationList;  ImageButton searchBtn;  myAdapter adapter;    FirebaseDatabase database;  DatabaseReference userReference;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_blood\_dnonor\_search*);   Intent searchGroup = getIntent();  bloodGroupSearch = searchGroup.getStringExtra("Blood Group");   searchBar = findViewById(R.id.*searchBar*);  searchRecycler = findViewById(R.id.*searchRecycler*);   FirebaseRecyclerOptions<DonationDetails> options = new FirebaseRecyclerOptions.Builder<DonationDetails>()  .setQuery(FirebaseDatabase.*getInstance*().getReference().child("DonationDetails").orderByChild("blGroup").startAt(bloodGroupSearch).endAt(bloodGroupSearch+"\uf8ff"),DonationDetails.class)  .build();  adapter = new myAdapter(options);  adapter.startListening();  searchRecycler.setAdapter(adapter);  adapter.setOnItemClickListener(new myAdapter.OnItemClickListener() {  @Override  public String OnItemClick(int position, String number) {  Toast.*makeText*(BloodDnonorSearch.this, number, Toast.*LENGTH\_SHORT*).show();  Intent profileView = new Intent(BloodDnonorSearch.this, ProfileView.class);  profileView.putExtra("mobile number",number);  startActivity(profileView);  return number;  }  });   searchBar.addTextChangedListener(new TextWatcher() {  @Override  public void beforeTextChanged(CharSequence s, int start, int count, int after) {  FirebaseRecyclerOptions<DonationDetails> options = new FirebaseRecyclerOptions.Builder<DonationDetails>()  .setQuery(FirebaseDatabase.*getInstance*().getReference().child("DonationDetails").orderByChild("blGroup").startAt(bloodGroupSearch).endAt(bloodGroupSearch+"\uf8ff"),DonationDetails.class)  .build();  adapter = new myAdapter(options);  adapter.startListening();  searchRecycler.setAdapter(adapter);  adapter.setOnItemClickListener(new myAdapter.OnItemClickListener() {  @Override  public String OnItemClick(int position, String number) {  Toast.*makeText*(BloodDnonorSearch.this, number, Toast.*LENGTH\_SHORT*).show();  Intent profileView = new Intent(BloodDnonorSearch.this, ProfileView.class);  profileView.putExtra("mobile number",number);  startActivity(profileView);  return number;  }  });  }   @Override  public void onTextChanged(CharSequence s, int start, int before, int count) {   }   @Override  public void afterTextChanged(Editable s) {   filter(s.toString());  }  });  }   private int checkDonorAvailability() {  return 0;  }   private void filter(String text) {   Query searchLocationDonor = FirebaseDatabase.*getInstance*().getReference().child("DonationDetails").orderByChild("donorLocation").startAt(text).endAt(text+"\uf8ff");  if (searchLocationDonor != null){  FirebaseRecyclerOptions<DonationDetails> options = new FirebaseRecyclerOptions.Builder<DonationDetails>()  .setQuery(searchLocationDonor, DonationDetails.class)  .build();   adapter = new myAdapter(options);  adapter.startListening();  searchRecycler.setAdapter(adapter);  adapter.setOnItemClickListener(new myAdapter.OnItemClickListener() {  @Override  public String OnItemClick(int position, String number) {  Toast.*makeText*(BloodDnonorSearch.this, number, Toast.*LENGTH\_SHORT*).show();  Intent profileView = new Intent(BloodDnonorSearch.this, ProfileView.class);  profileView.putExtra("mobile number",number);  startActivity(profileView);  return number;  }  });  }else{  Toast.*makeText*(BloodDnonorSearch.this, "No Person found", Toast.*LENGTH\_SHORT*).show();  }   } }  (ChatActivity.java)  package com.example.bloodgenix.activities;  import android.app.Dialog; import android.content.Intent; import android.os.Bundle; import android.view.MenuItem; import android.view.View; import android.view.ViewGroup; import android.widget.Button; import android.widget.EditText; import android.widget.ImageButton; import android.widget.ImageView; import android.widget.PopupMenu; import android.widget.TextView; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity; import androidx.cardview.widget.CardView; import androidx.recyclerview.widget.LinearLayoutManager; import androidx.recyclerview.widget.RecyclerView;  import com.bumptech.glide.Glide; import com.example.bloodgenix.Adapters.MessagesAdapter; import com.example.bloodgenix.Models.BlockUser; import com.example.bloodgenix.Models.Messages; import com.example.bloodgenix.R; import com.example.bloodgenix.SessionManager; import com.google.android.gms.tasks.OnCompleteListener; import com.google.android.gms.tasks.Task; import com.google.firebase.auth.FirebaseAuth; import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query; import com.google.firebase.database.ValueEventListener;  import java.util.ArrayList; import java.util.Date; import java.util.HashMap;  import de.hdodenhof.circleimageview.CircleImageView;  public class ChatActivity extends AppCompatActivity implements PopupMenu.OnMenuItemClickListener {   String passedData[] = new String[5];  CircleImageView profileImg;  ImageView activeStatus;  TextView receiverName;  RecyclerView messageAdapter;  ImageButton chatBackBtn, dotMenu;  CardView sendBtn;  EditText textMessage;  public static String *senderImg*;  public static String *receiverImg*;  String senderUID;  String receiverUID, whatToDo;  String senderRoom;  String receiverRoom;  static ArrayList<Messages> *messagesArrayList*;  MessagesAdapter onlyAdapter;   FirebaseDatabase database;  DatabaseReference reference;  FirebaseAuth auth;  public String senderName;  int blockFlag;  Dialog dialog;    @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_chat*);   Intent chat = getIntent();  passedData = chat.getStringArrayExtra("chatData");  whatToDo = chat.getStringExtra("whatToDo");   profileImg = findViewById(R.id.*profileImg*);  receiverName = findViewById(R.id.*receiverName*);  textMessage = findViewById(R.id.*textMessage*);  sendBtn = findViewById(R.id.*sendBtn*);  messageAdapter = findViewById(R.id.*messageAdapter*);  chatBackBtn = findViewById(R.id.*chatBackBtn*);  dotMenu = findViewById(R.id.*dotMenuBtn*);  activeStatus = findViewById(R.id.*activeStatus*);   auth = FirebaseAuth.*getInstance*();  *receiverImg* = passedData[2];  Glide.*with*(ChatActivity.this).load(passedData[2]).into(profileImg);  receiverName.setText(passedData[1]);  *messagesArrayList* = new ArrayList<>();    chatBackBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  if (whatToDo.equals("notificationChat")) {  Intent previous = new Intent(ChatActivity.this, DashBoard\_Screen.class);  previous.putExtra("profile Values", senderUID);  startActivity(previous);  } else {  Intent previousProfile = new Intent(ChatActivity.this, ProfileView.class);  previousProfile.putExtra("mobile number", passedData[0]);  startActivity(previousProfile);  }  }  });   dotMenu.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  PopupMenu popupMenu = new PopupMenu(ChatActivity.this, v);  popupMenu.setOnMenuItemClickListener(ChatActivity.this);  popupMenu.inflate(R.menu.*chat\_menu*);  popupMenu.show();   }  });   SessionManager sessionManager = new SessionManager(ChatActivity.this, SessionManager.*SESSION\_USERSESSION*);  HashMap<String, String> senderData = sessionManager.getUserDetailsFromSession();   String senderNumber = "+91 " + senderData.get(SessionManager.*KEY\_PHONENUMBER*);  String \_number = senderNumber.substring(4, senderNumber.length());  senderUID = senderNumber;  senderName = senderData.get(SessionManager.*KEY\_FULLNAME*);  receiverUID = passedData[0];  checkStatus(receiverUID);  Query checkUserSender = FirebaseDatabase.*getInstance*().getReference("Users").orderByChild("mobileNumber").equalTo(senderNumber);  checkUserSender.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  String \_fullName = snapshot.child(\_number).child("fullName").getValue(String.class);  String \_emailId = snapshot.child(\_number).child("emailId").getValue(String.class);  String \_gender = snapshot.child(\_number).child("gender").getValue(String.class);  String \_dob = snapshot.child(\_number).child("d\_o\_b").getValue(String.class);  String \_img = snapshot.child(\_number).child("profileImg").getValue(String.class);  String \_uid = snapshot.child(\_number).child("uid").getValue(String.class);   *senderImg* = \_img;  }   @Override  public void onCancelled(@NonNull DatabaseError error) {  }  });    LinearLayoutManager linearLayoutManager = new LinearLayoutManager(this);  linearLayoutManager.setStackFromEnd(true);  messageAdapter.setLayoutManager(linearLayoutManager);  onlyAdapter = new MessagesAdapter(ChatActivity.this, *messagesArrayList*, senderUID);  messageAdapter.setAdapter(onlyAdapter);   senderRoom = senderUID + receiverUID;  receiverRoom = receiverUID + senderUID;   chatReferenceCall();    sendBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  String textmessage = textMessage.getText().toString();  dialogCreationMore();   DatabaseReference keReference = FirebaseDatabase.*getInstance*().getReference("BlockUser");  keReference.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()) {  for (DataSnapshot dataSnapshot : snapshot.getChildren()) {  String keyValue = dataSnapshot.getKey();   if (keyValue.substring(0, 14).toString().equals(senderUID)) {  dialog.show();  //BLOCK DIALOG CREATION INITIALISATION  Button CANCEL = dialog.findViewById(R.id.*CANCEL*);  Button UNBLOCK = dialog.findViewById(R.id.*UNBLOCK*);   CANCEL.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  dialog.dismiss();  }  });   UNBLOCK.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  FirebaseDatabase.*getInstance*().getReference("BlockUser").child(keyValue).removeValue().addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {  if (task.isSuccessful()) {  dialog.dismiss();  }  }  });  }  });   } else {  Toast.*makeText*(ChatActivity.this, "You have been blocked", Toast.*LENGTH\_SHORT*).show();  break;  }  }  } else {  if (textmessage.isEmpty()) {  Toast.*makeText*(ChatActivity.this, "Enter text", Toast.*LENGTH\_SHORT*).show();  return;  }  textMessage.setText("");  Date date = new Date();   Messages messages = new Messages(textmessage, senderUID, date.getTime(), receiverUID, senderName, *senderImg*, passedData[1], passedData[2]);  database = FirebaseDatabase.*getInstance*();  database.getReference().child("Chats")  .child(senderRoom)  .child("messages")  .push().setValue(messages).addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {  if (task.isSuccessful()) {  database.getReference().child("Chats")  .child(receiverRoom)  .child("messages")  .push().setValue(messages).addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {   }  });  }  }  });  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });   }  });  }   public int checkBlock() {  Query chckBlock = FirebaseDatabase.*getInstance*().getReference("BlockUser").orderByChild("receiver").equalTo(receiverUID);  chckBlock.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()) { // Toast.makeText(ChatActivity.this, "inside block", Toast.LENGTH\_SHORT).show();  blockFlag = 1;  } // }else{ // Toast.makeText(ChatActivity.this, "outside block", Toast.LENGTH\_SHORT).show(); // blockFlag = 0; // }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });  return blockFlag;  }   public void dialogCreationMore() {  dialog = new Dialog(ChatActivity.this);  dialog.setContentView(R.layout.*block\_layout\_design*);  dialog.getWindow().setLayout(ViewGroup.LayoutParams.*WRAP\_CONTENT*, ViewGroup.LayoutParams.*WRAP\_CONTENT*); // dialog.getWindow().setBackgroundDrawable(getResources().getDrawable(R.drawable.info\_layout\_background\_style));  dialog.setCancelable(false);  dialog.getWindow().getAttributes().windowAnimations = R.style.*animation*;  dialog.setCanceledOnTouchOutside(true);   }   public void checkStatus(String receiverUID) {   Query checkStatus = FirebaseDatabase.*getInstance*().getReference("PersonStatus").orderByChild("phoneNumber").equalTo(receiverUID);  checkStatus.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()) {  String status = snapshot.child(receiverUID).child("status").getValue(String.class);  if (status.equals("true")) {  activeStatus.setVisibility(View.*VISIBLE*);  } else {  activeStatus.setVisibility(View.*INVISIBLE*);  }  } else {  activeStatus.setVisibility(View.*INVISIBLE*);  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });  }   public void chatReferenceCall() {  database = FirebaseDatabase.*getInstance*();  DatabaseReference chatReference = database.getReference().child("Chats").child(senderRoom).child("messages");  chatReference.addValueEventListener(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  *messagesArrayList*.clear();  for (DataSnapshot dataSnapshot : snapshot.getChildren()) {  Messages messages = dataSnapshot.getValue(Messages.class);  *messagesArrayList*.add(messages);  }  onlyAdapter.notifyDataSetChanged();  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });  }   @Override  public boolean onMenuItemClick(MenuItem item) {  switch (item.getItemId()) {  case R.id.*BlockUser*:  reference = database.getReference().child("BlockUser").child(senderUID + receiverUID);  BlockUser block = new BlockUser(receiverUID, senderUID);  reference.setValue(block).addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {  if (task.isSuccessful()) {  Toast.*makeText*(ChatActivity.this, "Added", Toast.*LENGTH\_SHORT*).show();  }  }  });  return true;  default:  return false;  }  } }  (DashBoard\_Screen.java)  package com.example.bloodgenix.activities;  import android.app.Dialog; import android.content.Context; import android.content.Intent; import android.content.pm.ApplicationInfo; import android.net.ConnectivityManager; import android.net.Uri; import android.os.Bundle; import android.view.MenuItem; import android.view.View; import android.view.ViewGroup; import android.widget.Button; import android.widget.ImageButton; import android.widget.ImageView; import android.widget.TextView; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.ActionBarDrawerToggle; import androidx.appcompat.app.AppCompatActivity; import androidx.appcompat.widget.Toolbar; import androidx.core.view.GravityCompat; import androidx.drawerlayout.widget.DrawerLayout; import androidx.navigation.NavController; import androidx.navigation.Navigation; import androidx.navigation.ui.NavigationUI;  import com.bumptech.glide.Glide; import com.example.bloodgenix.Models.PersonStatus; import com.example.bloodgenix.R; import com.example.bloodgenix.SessionManager; import com.google.android.gms.tasks.OnCompleteListener; import com.google.android.gms.tasks.Task; import com.google.android.material.bottomnavigation.BottomNavigationView; import com.google.android.material.navigation.NavigationView; import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query; import com.google.firebase.database.ValueEventListener; import com.google.firebase.storage.StorageReference;  import java.io.File; import java.util.HashMap;  import de.hdodenhof.circleimageview.CircleImageView;  public class DashBoard\_Screen extends AppCompatActivity {   TextView salutationText;  CircleImageView active\_person;  Button donationExpand, RecipientExpand, Logout;  View layout\_1, homeLayout;   //slider layout initialisation  DrawerLayout drawerLayout;  NavigationView navigationView;  ActionBarDrawerToggle toggle;   Toolbar toolbar;  CircleImageView sliderProfile;  TextView sliderName;  Dialog dialog;  ImageButton moreInfo;  ImageView fullProfile;  String personImage;    BottomNavigationView bottomNavigation;  NavController navController;    //FireBase initialisation  FirebaseDatabase database;  DatabaseReference reference;  StorageReference storageReference;   public String full\_number;  public String fetch\_details[] = new String[15];  public String donationAdding[] = new String[3];   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_dash\_board\_screen*);    //BOTTOM NAVIGATION IMPLEMENTATION  bottomNavigation = findViewById(R.id.*bottom\_navigation*);  navController = Navigation.*findNavController*(this, R.id.*main\_fragment*);  NavigationUI.*setupWithNavController*(bottomNavigation, navController);     //DRAWER LAYOUT IMPLEMENTATION  drawerLayout = findViewById(R.id.*drawerLayout*);  navigationView = findViewById(R.id.*navView*);  toolbar = findViewById(R.id.*toolbar*); // setSupportActionBar(toolbar);   database = FirebaseDatabase.*getInstance*("https://bloodgenix-bb937-default-rtdb.firebaseio.com/");   toggle = new ActionBarDrawerToggle(DashBoard\_Screen.this, drawerLayout, toolbar, R.string.*navigation\_open*, R.string.*navigation\_close*);  drawerLayout.addDrawerListener(toggle);  toggle.syncState();   layout\_1 = findViewById(R.id.*layout\_1*);  Logout = findViewById(R.id.*Logout*);  moreInfo = findViewById(R.id.*moreInfo*);   //Extracting values  Intent val = getIntent();  full\_number = val.getStringExtra("profile Values");  profileDetailsFetch(full\_number);   dialogCreationMore();  moreInfo.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  dialog.show();  }  });   View hView = navigationView.getHeaderView(0);  sliderProfile = (CircleImageView) hView.findViewById(R.id.*sliderProfile*);  sliderName = (TextView) hView.findViewById(R.id.*sliderName*);   sliderProfile.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  dialogCreation();  dialog.show();  }  });   //Creating Status of person  createStatus(full\_number);    SessionManager sessionManager = new SessionManager(this,"userLoginSession");  HashMap<String, String> userDetails = sessionManager.getUserDetailsFromSession();   String fullName = userDetails.get(SessionManager.*KEY\_FULLNAME*);    navigationView.setNavigationItemSelectedListener(new NavigationView.OnNavigationItemSelectedListener() {  @Override  public boolean onNavigationItemSelected(@NonNull MenuItem item) {   switch (item.getItemId()) {   case R.id.*profile\_view*:  Intent profileIntent = new Intent(DashBoard\_Screen.this, User\_Profile\_View.class);  profileIntent.putExtra("mobile number", full\_number);  startActivity(profileIntent);  drawerLayout.closeDrawer(GravityCompat.*START*);  break;   case R.id.*nav\_settings*:  startActivity(new Intent(DashBoard\_Screen.this, Settings.class));  drawerLayout.closeDrawer(GravityCompat.*START*);  break;  case R.id.*nav\_about*:  startActivity(new Intent(DashBoard\_Screen.this, About\_activity.class));  drawerLayout.closeDrawer(GravityCompat.*START*);  break;  case R.id.*nav\_sharing*:   ApplicationInfo api = getApplicationContext().getApplicationInfo();  String apkpath = api.sourceDir;  Intent shareIntent = new Intent(Intent.*ACTION\_SEND*);  shareIntent.setType("application/vnd.android.package-archive");  shareIntent.putExtra("AppSharing", Uri.*fromFile*(new File(apkpath)));  startActivity(Intent.*createChooser*(shareIntent, "ShareVia"));  drawerLayout.closeDrawer(GravityCompat.*START*);  break;  default:  return false;  }   return false;  }  }); // //  Logout.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  HashMap hashMap = new HashMap();  hashMap.put("status", "false");  FirebaseDatabase.*getInstance*().getReference().child("PersonStatus").child(full\_number).updateChildren(hashMap).addOnCompleteListener(new OnCompleteListener() {  @Override  public void onComplete(@NonNull Task task) {  if (task.isSuccessful()){ // logoutUserFromSession  SessionManager sessionManager = new SessionManager(DashBoard\_Screen.this,"userLoginSession");  HashMap<String, String> userDetails = sessionManager.getUserDetailsFromSession();   Intent logout = new Intent(DashBoard\_Screen.this, Welcome.class);  sessionManager.logoutUserFromSession();  startActivity(logout);  finish();  }  }  });  }  });  }   private void createStatus(String full\_number) {  if (isNetworkConnected()){  reference = database.getReference().child("PersonStatus").child(full\_number);  PersonStatus status = new PersonStatus(full\_number, "true");  reference.setValue(status).addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {  if (task.isSuccessful()){   }else{  Toast.*makeText*(DashBoard\_Screen.this, "doesn't Added Status", Toast.*LENGTH\_SHORT*).show();  }  }  });  }else{  reference = database.getReference().child("PersonStatus").child(full\_number);  PersonStatus status = new PersonStatus(full\_number, "false");  reference.setValue(status).addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {  if (task.isSuccessful()){   }else{  Toast.*makeText*(DashBoard\_Screen.this, "doesn't Added Status", Toast.*LENGTH\_SHORT*).show();  }  }  });  }  }   private boolean isNetworkConnected() {  ConnectivityManager cm = (ConnectivityManager) getSystemService(Context.*CONNECTIVITY\_SERVICE*);   return cm.getActiveNetworkInfo() != null && cm.getActiveNetworkInfo().isConnected();  }   public void dialogCreationMore() { // ViewGroup.LayoutParams.WRAP\_CONTENT,ViewGroup.LayoutParams.WRAP\_CONTENT  dialog = new Dialog(DashBoard\_Screen.this);  dialog.setContentView(R.layout.*info\_layout*);  dialog.getWindow().setLayout(1000,ViewGroup.LayoutParams.*WRAP\_CONTENT*);  dialog.getWindow().setBackgroundDrawable(getResources().getDrawable(R.drawable.*info\_layout\_background\_style*));  dialog.setCancelable(false);  dialog.getWindow().getAttributes().windowAnimations = R.style.*animation*;  dialog.setCanceledOnTouchOutside(true);   }   public String myData() {  return full\_number;  }   //Fetching Details from FireBase  public void profileDetailsFetch(String full\_number) {  String \_number = full\_number.substring(4, full\_number.length());  Query checkUser = FirebaseDatabase.*getInstance*().getReference("Users").orderByChild("mobileNumber").equalTo(full\_number);  checkUser.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()) {  String \_fullName = snapshot.child(\_number).child("fullName").getValue(String.class);  String \_emailId = snapshot.child(\_number).child("emailId").getValue(String.class);  String \_gender = snapshot.child(\_number).child("gender").getValue(String.class);  String \_dob = snapshot.child(\_number).child("d\_o\_b").getValue(String.class);  String \_img = snapshot.child(\_number).child("profileImg").getValue(String.class);   fetch\_details[0] = \_fullName;  fetch\_details[1] = \_emailId;  fetch\_details[2] = \_dob;  fetch\_details[3] = \_gender;  fetch\_details[4] = \_number;  fetch\_details[5] = \_img;   personImage = fetch\_details[5];  Glide.*with*(layout\_1).load(fetch\_details[5]).into(sliderProfile);  sliderName.setText(\_fullName);   } else {  Toast.*makeText*(DashBoard\_Screen.this, "User doesn't exist", Toast.*LENGTH\_SHORT*).show();  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {  Toast.*makeText*(DashBoard\_Screen.this, error.getMessage(), Toast.*LENGTH\_SHORT*).show();  }  });  }   @Override  public boolean onOptionsItemSelected(@NonNull MenuItem item) {  if (toggle.onOptionsItemSelected(item)) {  return true;  }  return super.onOptionsItemSelected(item);  }   @Override  public void onBackPressed() {  if (drawerLayout.isDrawerOpen(GravityCompat.*START*)) {  drawerLayout.closeDrawer(GravityCompat.*START*);  } else {  super.onBackPressed();  }  }   private void dialogCreation() { // ViewGroup.LayoutParams.WRAP\_CONTENT,ViewGroup.LayoutParams.WRAP\_CONTENT  dialog = new Dialog(DashBoard\_Screen.this);  dialog.setContentView(R.layout.*image\_click\_zoom\_layout*);  dialog.getWindow().setLayout(ViewGroup.LayoutParams.*MATCH\_PARENT*, ViewGroup.LayoutParams.*WRAP\_CONTENT*); // dialog.getWindow().setBackgroundDrawable(getResources().getDrawable(R.drawable.info\_layout\_background\_style));  dialog.setCancelable(false);  dialog.getWindow().getAttributes().windowAnimations = R.style.*animation*;  dialog.setCanceledOnTouchOutside(true);  fullProfile = dialog.findViewById(R.id.*fullProfile*);  Glide.*with*(getApplicationContext()).load(personImage).into(fullProfile);  } }  (DonationForm.java)  package com.example.bloodgenix.activities;  import android.Manifest; import android.app.Dialog; import android.content.Intent; import android.content.pm.PackageManager; import android.location.Address; import android.location.Geocoder; import android.location.Location; import android.os.Bundle; import android.view.View; import android.view.ViewGroup; import android.widget.AdapterView; import android.widget.ArrayAdapter; import android.widget.AutoCompleteTextView; import android.widget.Button; import android.widget.CheckBox; import android.widget.ImageButton; import android.widget.NumberPicker; import android.widget.RadioButton; import android.widget.RadioGroup; import android.widget.TextView; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity; import androidx.core.app.ActivityCompat;  import com.example.bloodgenix.Models.DonationDetails; import com.example.bloodgenix.R; import com.google.android.gms.location.FusedLocationProviderClient; import com.google.android.gms.location.LocationServices; import com.google.android.gms.tasks.OnCompleteListener; import com.google.android.gms.tasks.Task; import com.google.android.material.textfield.TextInputEditText; import com.google.android.material.textfield.TextInputLayout; import com.google.firebase.auth.FirebaseAuth; import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query; import com.google.firebase.database.ValueEventListener;  import java.io.IOException; import java.util.List; import java.util.Locale;   public class DonationForm extends AppCompatActivity {   TextInputLayout diabeticCountLayout, PeriodLayout;  RadioGroup radioGroup, radioGroup2;  RadioButton radioButton, radioButton2;  ImageButton cancelBtn;  Button applyDonation;  TextInputEditText otherSpecify, locationDonation;  CheckBox authorise;  Dialog dialog;   AutoCompleteTextView bloodGroup, diabeticCount, DonationPeriod;  ArrayAdapter<String> arrayAdapterBGroup, arrayAdapterBCount, arrayAdapterDPeriod;  String phoneNumber, authUserID;  String bGroup[];  String DiabeticCount[], Weight[], DPeriod[];   public String formFillData [];  NumberPicker weight;  FusedLocationProviderClient fusedLocationProviderClientDon;   //FireBase variables  FirebaseAuth auth;  FirebaseDatabase database;  DatabaseReference reference;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_donation\_form*);   //getting values from previous activities  Intent i8 = getIntent();  formFillData = i8.getStringArrayExtra("Donation");  phoneNumber = formFillData[1];   //giving id to the respective views  diabeticCountLayout = findViewById(R.id.*diabeticCountLayout*);  locationDonation = findViewById(R.id.*locationDonation*);  PeriodLayout = findViewById(R.id.*PeriodLayout*);  bloodGroup = findViewById(R.id.*bloodGroup*);  diabeticCount = findViewById(R.id.*diabeticCount*);  radioGroup = findViewById(R.id.*radioGroup*);  cancelBtn = findViewById(R.id.*cancelBtn*);  DonationPeriod = findViewById(R.id.*DonationPeriod*);  radioGroup2 = findViewById(R.id.*radioGroup2*);  applyDonation = findViewById(R.id.*applyDonation*);  otherSpecify = findViewById(R.id.*otherSpecify*);  authorise = findViewById(R.id.*authorise*);  weight = findViewById(R.id.*weight*);   database = FirebaseDatabase.*getInstance*("https://bloodgenix-bb937-default-rtdb.firebaseio.com/");  auth = FirebaseAuth.*getInstance*();   // Setting weight max and min values  weight.setMinValue(50);  weight.setMaxValue(100);   //SETTING CURRENT LOCATION FOR PERSON  try{  fusedLocationProviderClientDon = LocationServices.*getFusedLocationProviderClient*(this);  if (ActivityCompat.*checkSelfPermission*(DonationForm.this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) == PackageManager.*PERMISSION\_GRANTED*) {  fusedLocationProviderClientDon.getLastLocation().addOnCompleteListener(new OnCompleteListener<Location>() {  @Override  public void onComplete(@NonNull Task<Location> task) {  Location location = task.getResult();  if (location != null) {  Geocoder geocoder = new Geocoder(DonationForm.this, Locale.*getDefault*());  try {  List<Address> addresses = geocoder.getFromLocation(location.getLatitude(), location.getLongitude(), 1);  if (addresses.size()>0){  for (Address adr: addresses){  if (adr.getLocality() != null && adr.getLocality().length()>0){  locationDonation.setText(adr.getLocality());  break;  }  }  }   } catch (IOException e) {  e.printStackTrace();  }  }  }  });   } else {  ActivityCompat.*requestPermissions*(DonationForm.this, new String[]{Manifest.permission.*ACCESS\_FINE\_LOCATION*}, 44);  }  }  catch (Exception e){  Toast.*makeText*(DonationForm.this, e.getMessage(), Toast.*LENGTH\_SHORT*).show();  }   cancelBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent dash = new Intent(DonationForm.this, DashBoard\_Screen.class);  dash.putExtra("profile Values", phoneNumber);  startActivity(dash);  }  });   //BLOOD GROUP DROP DOWN  bGroup = getResources().getStringArray(R.array.*bloodgroup*);  arrayAdapterBGroup = new ArrayAdapter<String>(this, R.layout.*items\_list\_blood*, bGroup);  bloodGroup.setAdapter(arrayAdapterBGroup);   //DIABETIC COUNT DROP DOWN  DiabeticCount = getResources().getStringArray(R.array.*diabeticCount*);  arrayAdapterBCount = new ArrayAdapter<String>(this, R.layout.*diabetic\_cout\_dropdown*, DiabeticCount);  diabeticCount.setAdapter(arrayAdapterBCount);  diabeticCount.setOnItemClickListener(new AdapterView.OnItemClickListener() {  @Override  public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  String value = parent.getItemAtPosition(position).toString();  if (value.equals("More than 8")){  Toast.*makeText*(DonationForm.this, "You are much diabetic", Toast.*LENGTH\_SHORT*).show();  diabeticCount.setText(null);  }  }  });   //DONATION PERIOD DROP DOWN  DPeriod = getResources().getStringArray(R.array.*periodMonth*);  arrayAdapterDPeriod = new ArrayAdapter<String>(this, R.layout.*donation\_period\_dropdown*, DPeriod);  DonationPeriod.setAdapter(arrayAdapterDPeriod);  DonationPeriod.setOnItemClickListener(new AdapterView.OnItemClickListener() {  @Override  public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  String dValue= parent.getItemAtPosition(position).toString();  if (dValue.equals("Less than 3 month")){  Toast.*makeText*(DonationForm.this, "Your donation period not completed", Toast.*LENGTH\_SHORT*).show();  DonationPeriod.setText(null);  }  }  });   //SUCCESS LAYOUT VIEWING  dialogInitialise();   radioGroup.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {  @Override  public void onCheckedChanged(RadioGroup group, int checkedId) {  radioButton = findViewById(checkedId);  if (radioButton.getText().toString().equals("yes")) {  diabeticCountLayout.setEnabled(true);  }  else {  diabeticCountLayout.setEnabled(false);  diabeticCount.setText(null);  }  }  });   radioGroup2.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {  @Override  public void onCheckedChanged(RadioGroup group, int checkedId) {  radioButton2 = findViewById(checkedId);  if (radioButton2.getText().toString().equals("yes")) {  PeriodLayout.setEnabled(true);  } else {  PeriodLayout.setEnabled(false);  DonationPeriod.setText(null);  }  }  });   applyDonation.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  int result = checkValidations();  if (result == 1){  bloodGroup.setError(null);  diabeticCount.setError(null);  otherSpecify.setError(null);  DonationPeriod.setError(null);   getUserAuthValue();  //Toast.makeText(DonationForm.this, idValue, Toast.LENGTH\_SHORT).show();  //CHECKING FOR THE USER IS APPLIED FOR DONATION OR NOT  Query recipientDetails = FirebaseDatabase.*getInstance*().getReference("RecipientDetails").orderByChild("phoneNumber").equalTo(phoneNumber);  recipientDetails.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()){  Toast.*makeText*(DonationForm.this, "You have applied for recipient before", Toast.*LENGTH\_SHORT*).show();  }else{  String wight = String.*valueOf*(weight.getValue());  reference = database.getReference().child("DonationDetails").child(phoneNumber);  DonationDetails donationDetails = new DonationDetails(authUserID,phoneNumber,bloodGroup.getText().toString(),radioButton.getText().toString(),diabeticCount.getText().toString(),otherSpecify.getText().toString(),radioButton2.getText().toString(),DonationPeriod.getText().toString(),wight,locationDonation.getText().toString(),formFillData[0], formFillData[2] );  reference.setValue(donationDetails).addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {  if (task.isSuccessful()){  dialog.show();  }  else{  dialog.dismiss();  }  }  });  }  }  @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });  }  else{  Toast.*makeText*(DonationForm.this, "Any field left empty", Toast.*LENGTH\_SHORT*).show();  }  }  });  }   private void getUserAuthValue() {   String \_number = phoneNumber.substring(4, phoneNumber.length());  Query chckUser = FirebaseDatabase.*getInstance*().getReference("Users").orderByChild("mobileNumber").equalTo(phoneNumber);  chckUser.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  authUserID = snapshot.child(\_number).child("uid").getValue(String.class);  Toast.*makeText*(DonationForm.this, "Inside authUserID", Toast.*LENGTH\_SHORT*).show();  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });  }    //DIALOG BOX INITIALISATION  public void dialogInitialise(){  dialog = new Dialog(DonationForm.this);  dialog.setContentView(R.layout.*custom\_success\_dialog\_layout*);  dialog.getWindow().setBackgroundDrawable(getDrawable(R.drawable.*background\_success\_animation*));  dialog.getWindow().setLayout(ViewGroup.LayoutParams.*WRAP\_CONTENT*,ViewGroup.LayoutParams.*WRAP\_CONTENT*);  dialog.setCancelable(false);  dialog.getWindow().getAttributes().windowAnimations = R.style.*animation*;   Button Okay, Cancel;  TextView dialogType, dialogDesc;  Okay = dialog.findViewById(R.id.*Okay*);  Cancel = dialog.findViewById(R.id.*Cancel*);  dialogType = dialog.findViewById(R.id.*dialogType*);  dialogDesc = dialog.findViewById(R.id.*dialogDesc*);   Okay.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent dash = new Intent(DonationForm.this,DashBoard\_Screen.class);  dash.putExtra("profile Values",phoneNumber);  startActivity(dash);  dialog.dismiss();  }  });   Cancel.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  dialog.dismiss();  }  });  }    //VALIDATION CHECKING FOR EMPTY FIELDS  public int checkValidations(){   int flag=0;  if (bloodGroup.getText().toString().isEmpty()) {  bloodGroup.setError("Please select");  bloodGroup.setFocusable(true);  flag=0;  }  else if (otherSpecify.getText().toString().isEmpty()) {  otherSpecify.setError("Please Enter");  otherSpecify.setFocusable(true);  flag=0;  }  else if (radioGroup.getCheckedRadioButtonId() == -1) {  Toast.*makeText*(DonationForm.this, "Please select", Toast.*LENGTH\_SHORT*).show();  flag=0;  }  else if (radioButton.getText().toString().equals("yes")) {  if (diabeticCount.getText().toString().isEmpty()) {  diabeticCount.setError("Please select");  flag=0;  }  else{  flag=1;  }  }  else if (radioGroup2.getCheckedRadioButtonId() == -1) {  Toast.*makeText*(DonationForm.this, "Please select", Toast.*LENGTH\_SHORT*).show();  flag=0;  }  else if (radioButton2.getText().toString().equals("yes")) {  if (DonationPeriod.getText().toString().isEmpty()) {  DonationPeriod.setError("Please select");  flag=0;  }  else{  flag=1;  }  }  else{  bloodGroup.setError(null);  diabeticCount.setError(null);  otherSpecify.setError(null);  DonationPeriod.setError(null);  flag=1;  }  return flag;   }    // function for applying enable and disable over apply button  public void checkAuthorise(View view) {  if (authorise.isChecked() == true) {  applyDonation.setEnabled(true);  } else {  applyDonation.setEnabled(false);  }  } }  (RecipientForm.java)  package com.example.bloodgenix.activities;  import android.Manifest; import android.app.Dialog; import android.content.Intent; import android.content.pm.PackageManager; import android.location.Address; import android.location.Geocoder; import android.location.Location; import android.os.Bundle; import android.view.View; import android.view.ViewGroup; import android.widget.ArrayAdapter; import android.widget.AutoCompleteTextView; import android.widget.Button; import android.widget.CheckBox; import android.widget.ImageButton; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity; import androidx.core.app.ActivityCompat;  import com.example.bloodgenix.Models.RecipientDetails; import com.example.bloodgenix.R; import com.google.android.gms.location.FusedLocationProviderClient; import com.google.android.gms.location.LocationServices; import com.google.android.gms.tasks.OnCompleteListener; import com.google.android.gms.tasks.Task; import com.google.android.material.textfield.TextInputEditText; import com.google.firebase.auth.FirebaseAuth; import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query; import com.google.firebase.database.ValueEventListener;  import java.io.IOException; import java.util.List; import java.util.Locale;  public class RecipientForm extends AppCompatActivity {   AutoCompleteTextView bloodGroupRecipient, reasonBlood;  TextInputEditText locationRecipient, requirementRecipient;  ArrayAdapter<String> AdapterBGroupRecipient, AdapterReasonBlood;  String bGroup[], reason[];  CheckBox authoriseRecipient;  Button applyRecipient;  ImageButton cancelBtnRecipient;  String phoneNumb, personName, personImg;  Dialog dialog;  public String formFillData [];   FusedLocationProviderClient fusedLocationProviderClient;   //FireBase variable declaration  FirebaseAuth auth;  FirebaseDatabase database;  DatabaseReference reference;  RecipientDetails recipientDetails;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_recipient\_form*);   Intent i5 = getIntent();  formFillData = i5.getStringArrayExtra("Recipient");   phoneNumb = formFillData[1];   bloodGroupRecipient = findViewById(R.id.*bloodGroupRecipient*);  locationRecipient = findViewById(R.id.*locationRecipient*);  authoriseRecipient = findViewById(R.id.*authoriseRecipient*);  applyRecipient = findViewById(R.id.*applyRecipient*);  cancelBtnRecipient = findViewById(R.id.*cancelBtnRecipient*);  requirementRecipient = findViewById(R.id.*requirementRecipient*);  reasonBlood = findViewById(R.id.*reasonBlood*);   fusedLocationProviderClient = LocationServices.*getFusedLocationProviderClient*(this);   //Blood group drop down code  bGroup = getResources().getStringArray(R.array.*bloodgroup*);  AdapterBGroupRecipient = new ArrayAdapter<String>(this, R.layout.*items\_list\_blood*, bGroup);  bloodGroupRecipient.setAdapter(AdapterBGroupRecipient);   //Reason Drop down coding  reason = getResources().getStringArray(R.array.*resonBlood*);  AdapterReasonBlood = new ArrayAdapter<String>(this,R.layout.*item\_list\_reason*,reason);  reasonBlood.setAdapter(AdapterReasonBlood);   //CURRENT LOCATION FETCHING  if (ActivityCompat.*checkSelfPermission*(RecipientForm.this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) == PackageManager.*PERMISSION\_GRANTED*) {  fusedLocationProviderClient.getLastLocation().addOnCompleteListener(new OnCompleteListener<Location>() {  @Override  public void onComplete(@NonNull Task<Location> task) {  Location location = task.getResult();  if (location != null) {  Geocoder geocoder = new Geocoder(RecipientForm.this, Locale.*getDefault*());  try {  List<Address> addresses = geocoder.getFromLocation(location.getLatitude(), location.getLongitude(), 1);  if (addresses.size()>0){  for (Address adr: addresses){  if (adr.getLocality() != null && adr.getLocality().length()>0){  locationRecipient.setText(adr.getLocality());  break;  }  }  }  } catch (IOException e) {  e.printStackTrace();  }  }  }  });   } else {  ActivityCompat.*requestPermissions*(RecipientForm.this, new String[]{Manifest.permission.*ACCESS\_FINE\_LOCATION*}, 44);  }  cancelBtnRecipient.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent donation = new Intent(RecipientForm.this,DashBoard\_Screen.class);  donation.putExtra("mobile number",phoneNumb);  startActivity(donation);  }  });    //SUCCESS LAYOUT VIEWING  dialog = new Dialog(RecipientForm.this);  dialog.setContentView(R.layout.*custom\_success\_dialog\_layout*);  dialog.getWindow().setBackgroundDrawable(getDrawable(R.drawable.*background\_success\_animation*));  dialog.getWindow().setLayout(ViewGroup.LayoutParams.*WRAP\_CONTENT*,ViewGroup.LayoutParams.*WRAP\_CONTENT*);  dialog.setCancelable(false);  dialog.getWindow().getAttributes().windowAnimations = R.style.*animation*;   Button Okay, Cancel;  Okay = dialog.findViewById(R.id.*Okay*);  Cancel = dialog.findViewById(R.id.*Cancel*);    Okay.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  database = FirebaseDatabase.*getInstance*();  auth = FirebaseAuth.*getInstance*();  reference = database.getReference().child("RecipientDetails").child(phoneNumb);  recipientDetails = new RecipientDetails(auth.getUid(), bloodGroupRecipient.getText().toString(), locationRecipient.getText().toString(), requirementRecipient.getText().toString(),reasonBlood.getText().toString(),phoneNumb, formFillData[0], formFillData[2]);  reference.setValue(recipientDetails).addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {  if (task.isSuccessful()){  Intent searchPage = new Intent(RecipientForm.this, BloodDnonorSearch.class);  searchPage.putExtra("Blood Group",bloodGroupRecipient.getText().toString());  startActivity(searchPage);  dialog.dismiss();  }else{  dialog.dismiss();  Toast.*makeText*(RecipientForm.this, "Try Again !..", Toast.*LENGTH\_SHORT*).show();  }  }  });   }  });   Cancel.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  dialog.dismiss();  }  });    applyRecipient.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  if(bloodGroupRecipient.getText().toString().isEmpty()){  bloodGroupRecipient.setError("Please Select");  }  else if(requirementRecipient.getText().toString().isEmpty()){  requirementRecipient.setError("Field left empty");  }  else if(reasonBlood.getText().toString().isEmpty()){  reasonBlood.setError("Please Select");  }  else{  bloodGroupRecipient.setError(null);  requirementRecipient.setError(null);  reasonBlood.setError(null);   Query donationDetails = FirebaseDatabase.*getInstance*().getReference("DonationDetails").orderByChild("phoneNumber").equalTo(phoneNumb);  donationDetails.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()){  Toast.*makeText*(RecipientForm.this, "You have applied for Donation", Toast.*LENGTH\_SHORT*).show();  }else{  dialog.show();  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });  }  }  });  }   public void authorise(View view){  if (authoriseRecipient.isChecked() == true) {  applyRecipient.setEnabled(true);  } else {  applyRecipient.setEnabled(false);  }  }  }  (ForgetPassword.java)  package com.example.bloodgenix.activities;  import android.app.Activity; import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.Button; import android.widget.ImageButton; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity;  import com.example.bloodgenix.R; import com.google.android.material.textfield.TextInputEditText; import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query; import com.google.firebase.database.ValueEventListener; import com.hbb20.CountryCodePicker;  import java.util.Objects;  public class ForgetPassword extends AppCompatActivity {   CountryCodePicker forgetCountryPicker;  TextInputEditText forgetMobileNo;  String countryWithPlus;  Button forgetNextBtn;  ImageButton forgotBackBtn;   String whatToDo, phoneNumberVal;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_forget\_password*);   Intent previousPage = getIntent();  whatToDo = previousPage.getStringExtra("whatToDo");  phoneNumberVal = previousPage.getStringExtra("mobile Number");   forgetCountryPicker = findViewById(R.id.*forgetCountryPicker*);  forgetMobileNo = findViewById(R.id.*forgetMobileNo*);  forgetNextBtn = findViewById(R.id.*forgetNextBtn*);  forgotBackBtn = findViewById(R.id.*forgotBackBtn*);  listners(ForgetPassword.this);   setValuestoViews(whatToDo, phoneNumberVal);   forgotBackBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  if (whatToDo.equals("UserProfile")){  Intent userPage = new Intent(ForgetPassword.this, User\_Profile\_View.class);  userPage.putExtra("mobile number",phoneNumberVal);  startActivity(userPage);  finish();  }else if(whatToDo.equals("WelcomePage")){  startActivity(new Intent(ForgetPassword.this, Welcome.class));  finish();  }   }  });   forgetNextBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  String \_number = Objects.*requireNonNull*(forgetMobileNo.getText()).toString().trim();  String full\_number = countryWithPlus + " " + \_number;  Toast.*makeText*(ForgetPassword.this, full\_number, Toast.*LENGTH\_SHORT*).show();  Query checkUser = FirebaseDatabase.*getInstance*().getReference("Users").orderByChild("mobileNumber").equalTo(full\_number);  checkUser.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()){  forgetMobileNo.setError(null);  Intent i0 = new Intent(ForgetPassword.this, Otp\_Screen2.class);  i0.putExtra("whatToDo","updateData");  i0.putExtra("phoneNo",full\_number);  startActivity(i0);  }else{  forgetMobileNo.setError("No such user");  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });  }  });    }   private void setValuestoViews(String whatToDo, String phoneNumberVal) {   if (whatToDo.equals("UserProfile")){  forgetMobileNo.setText(phoneNumberVal.substring(4,phoneNumberVal.length()));  }else if (whatToDo.equals("WelcomePage")){  forgetMobileNo.setText(phoneNumberVal);  }  }   private void listners(Activity forget) {  String code = forgetCountryPicker.getSelectedCountryCode();  countryWithPlus = forgetCountryPicker.getSelectedCountryCodeWithPlus();   } }  (fullProfileView.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.ImageButton; import android.widget.ImageView; import android.widget.TextView;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity;  import com.bumptech.glide.Glide; import com.example.bloodgenix.R; import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query; import com.google.firebase.database.ValueEventListener;  public class fullProfileView extends AppCompatActivity {   ImageView fullProfile;  TextView fullProfileNumber;  ImageButton back\_button;   String Img, Number;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_full\_profile\_view*);   Intent previousExtract = getIntent();  Number = previousExtract.getStringExtra("phone Number"); // Number = previousExtract.getStringExtra();   fullProfile = findViewById(R.id.*fullProfile*);  fullProfileNumber = findViewById(R.id.*fullProfileNumber*);  back\_button = findViewById(R.id.*back\_button*);   fetchImageValue(Number);   back\_button.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent toBack = new Intent(fullProfileView.this, User\_Profile\_View.class);  toBack.putExtra("mobile number",Number);  startActivity(toBack);  finish();  }  });  }   private void fetchImageValue(String number) {  String \_number = number.substring(4, number.length());  Query checkUser = FirebaseDatabase.*getInstance*().getReference("Users").orderByChild("mobileNumber").equalTo(number);  checkUser.addValueEventListener(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  String \_img = snapshot.child(\_number).child("profileImg").getValue(String.class);  Glide.*with*(fullProfileView.this).load(\_img).into(fullProfile);  fullProfileNumber.setText(number);  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });  } }  (OnBoardingScreen.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.os.Bundle; import android.text.Html; import android.view.View; import android.widget.Button; import android.widget.LinearLayout; import android.widget.TextView;  import androidx.appcompat.app.AppCompatActivity; import androidx.viewpager.widget.ViewPager;  import com.example.bloodgenix.Adapters.SliderAdapter; import com.example.bloodgenix.R;  public class OnBoardingScreen extends AppCompatActivity {   //Defining Layouts  ViewPager sliderView;  LinearLayout dots;  private TextView[] mDots;  private Button FinishButton;  private int mCurrentPage;   private SliderAdapter sliderAdapter;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);   setContentView(R.layout.*activity\_on\_boarding*);  sliderView =(ViewPager) findViewById(R.id.*sliderView*);  dots = (LinearLayout) findViewById(R.id.*dots*);  FinishButton =findViewById(R.id.*FinishButton*);   sliderAdapter = new SliderAdapter(this);  sliderView.setAdapter(sliderAdapter);  addDotsIndicator(0);   sliderView.addOnPageChangeListener(viewListener);  }   public void addDotsIndicator(int position){  mDots = new TextView[3];  dots.removeAllViews();  for(int i=0; i<mDots.length; i++){  mDots[i] = new TextView(this);  mDots[i].setText(Html.*fromHtml*("&#8226;"));  mDots[i].setTextSize(37);  mDots[i].setTextColor(getResources().getColor(R.color.*content\_2*));  dots.addView(mDots[i]);   }  if(mDots.length > 0){  mDots[position].setTextColor(getResources().getColor(R.color.*black*));  }  }  ViewPager.OnPageChangeListener viewListener = new ViewPager.OnPageChangeListener() {  @Override  public void onPageScrolled(int position, float positionOffset, int positionOffsetPixels) {   }   @Override  public void onPageSelected(int position) {  addDotsIndicator(position);  mCurrentPage=position;  if (position == 2){  dots.setEnabled(false);  dots.setVisibility(View.*INVISIBLE*);  FinishButton.setEnabled(true);  FinishButton.setVisibility(View.*VISIBLE*);   FinishButton.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent welcome = new Intent(OnBoardingScreen.this, Welcome.class);  startActivity(welcome);  }  });  }else{  dots.setEnabled(true);  dots.setVisibility(View.*VISIBLE*);  FinishButton.setEnabled(false);  FinishButton.setVisibility(View.*INVISIBLE*);  }  }   @Override  public void onPageScrollStateChanged(int state) {   }  }; }  (Otp\_Screen2.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.net.Uri; import android.os.Bundle; import android.view.View; import android.widget.Button; import android.widget.TextView; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity;  import com.chaos.view.PinView; import com.example.bloodgenix.LoadingDialog; import com.example.bloodgenix.Models.Users; import com.example.bloodgenix.R; import com.google.android.gms.tasks.OnCompleteListener; import com.google.android.gms.tasks.OnSuccessListener; import com.google.android.gms.tasks.Task; import com.google.firebase.FirebaseException; import com.google.firebase.FirebaseTooManyRequestsException; import com.google.firebase.auth.AuthResult; import com.google.firebase.auth.FirebaseAuth; import com.google.firebase.auth.FirebaseAuthInvalidCredentialsException; import com.google.firebase.auth.FirebaseUser; import com.google.firebase.auth.PhoneAuthCredential; import com.google.firebase.auth.PhoneAuthOptions; import com.google.firebase.auth.PhoneAuthProvider; import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.storage.FirebaseStorage; import com.google.firebase.storage.StorageReference; import com.google.firebase.storage.UploadTask;  import java.util.concurrent.TimeUnit;  public class Otp\_Screen2 extends AppCompatActivity {   public String Otp\_details[] = new String[15];  String code\_by\_system, Number\_entered\_by\_user;  String imageURI, pdfUri, whatToDo, forgetNumber;  TextView dispNumber;  PinView otpVal;  Button resendBtn, verify;   private String mVerificationId;  private PhoneAuthProvider.ForceResendingToken mResendToken;  private PhoneAuthProvider.OnVerificationStateChangedCallbacks mCallbacks;    //FireBase Initialisation of variables  FirebaseAuth auth;  FirebaseDatabase database;  FirebaseStorage storage;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_otp\_screen2*);   //Values intake from previous activity  Intent i3 = getIntent();  Otp\_details = i3.getStringArrayExtra("Otp\_value");  whatToDo = i3.getStringExtra("whatToDo");  forgetNumber = i3.getStringExtra("phoneNo");    //SETTING THE PHONE NUMBER  Number\_entered\_by\_user = forgetNumber.substring(4, forgetNumber.length());  //Toast.makeText(Otp\_Screen2.this, whatToDo, Toast.LENGTH\_SHORT).show();   //firebase values  auth = FirebaseAuth.*getInstance*();  database = FirebaseDatabase.*getInstance*("https://bloodgenix-bb937-default-rtdb.firebaseio.com/");  storage = FirebaseStorage.*getInstance*();   //Otp code  otpVal = findViewById(R.id.*otpVal*);  dispNumber = findViewById(R.id.*dispNumber*); // dispNumber.setText(forgetNumber);  dispNumber.setText(forgetNumber);   mCallbacks = new PhoneAuthProvider.OnVerificationStateChangedCallbacks() {   @Override  public void onVerificationCompleted(PhoneAuthCredential credential) {  String code = credential.getSmsCode();  if (code != null) {  otpVal.setText(code);  verifyPhoneNumberWithCode(mVerificationId, code);  }   }   @Override  public void onVerificationFailed(FirebaseException e) {  // This callback is invoked in an invalid request for verification is made,  // for instance if the the phone number format is not valid.    if (e instanceof FirebaseAuthInvalidCredentialsException) {  Toast.*makeText*(Otp\_Screen2.this, e.getMessage(), Toast.*LENGTH\_SHORT*).show();  // Invalid request  } else if (e instanceof FirebaseTooManyRequestsException) {  // The SMS quota for the project has been exceeded  Toast.*makeText*(Otp\_Screen2.this, e.getMessage(), Toast.*LENGTH\_SHORT*).show();  }   // Show a message and update the UI  }   @Override  public void onCodeSent(@NonNull String verificationId,  @NonNull PhoneAuthProvider.ForceResendingToken token) {  // The SMS verification code has been sent to the provided phone number, we  // now need to ask the user to enter the code and then construct a credential  // by combining the code with a verification ID.    // Save verification ID and resending token so we can use them later  mVerificationId = verificationId;  mResendToken = token;  }  };   startPhoneNumberVerification(Number\_entered\_by\_user);  resendBtn = findViewById(R.id.*resendBtn*);  resendBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  resendVerificationCode(Number\_entered\_by\_user, mResendToken);  }  });   verify = findViewById(R.id.*verify*);  verify.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  String code = otpVal.getText().toString();  verifyPhoneNumberWithCode(mVerificationId, code);  }  });   }   private void startPhoneNumberVerification(String phoneNumber) {  // [START start\_phone\_auth]  PhoneAuthOptions options =  PhoneAuthOptions.*newBuilder*(auth)  .setPhoneNumber("+91" + phoneNumber) // Phone number to verify  .setTimeout(60L, TimeUnit.*SECONDS*) // Timeout and unit  .setActivity(Otp\_Screen2.this) // Activity (for callback binding)  .setCallbacks(mCallbacks) // OnVerificationStateChangedCallbacks  .build();  PhoneAuthProvider.*verifyPhoneNumber*(options);  // [END start\_phone\_auth]  }   private void verifyPhoneNumberWithCode(String verificationId, String code) {  // [START verify\_with\_code]   PhoneAuthCredential credential = PhoneAuthProvider.*getCredential*(verificationId, code);  signInWithPhoneAuthCredential(credential);  // [END verify\_with\_code]  }   // [START resend\_verification]  private void resendVerificationCode(String phoneNumber, PhoneAuthProvider.ForceResendingToken token) {  PhoneAuthOptions options =  PhoneAuthOptions.*newBuilder*(auth)  .setPhoneNumber("+91" + phoneNumber) // Phone number to verify  .setTimeout(60L, TimeUnit.*SECONDS*) // Timeout and unit  .setActivity(Otp\_Screen2.this) // Activity (for callback binding)  .setCallbacks(mCallbacks) // OnVerificationStateChangedCallbacks  .setForceResendingToken(token) // ForceResendingToken from callbacks  .build();  PhoneAuthProvider.*verifyPhoneNumber*(options);  }  // [END resend\_verification]   // [START sign\_in\_with\_phone]  private void signInWithPhoneAuthCredential(PhoneAuthCredential credential) {  auth.signInWithCredential(credential)  .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {  @Override  public void onComplete(@NonNull Task<AuthResult> task) {  if (task.isSuccessful()) {  LoadingDialog dialog = new LoadingDialog(Otp\_Screen2.this);  dialog.startDialog();   if (whatToDo.equals("updateData")) {  updateUsersData(dialog);  } else {  storeDataset(task, dialog);  }  } else {  // Sign in failed, display a message and update the UI  Toast.*makeText*(Otp\_Screen2.this, task.getException().getMessage(), Toast.*LENGTH\_SHORT*).show();   if (task.getException() instanceof FirebaseAuthInvalidCredentialsException) {  // The verification code entered was invalid  Toast.*makeText*(Otp\_Screen2.this, task.getException().getMessage(), Toast.*LENGTH\_SHORT*).show();  }  }  }  });  }   private void updateUsersData(LoadingDialog dialog) {  Intent setPass = new Intent(Otp\_Screen2.this, SetPassword.class);  setPass.putExtra("phoneNo", forgetNumber);  dialog.dismissDialog();  startActivity(setPass);  finish();  }   private void storeDataset(Task<AuthResult> task, LoadingDialog dialog) {  FirebaseUser user = task.getResult().getUser();  DatabaseReference reference = database.getReference().child("Users").child(Number\_entered\_by\_user);  StorageReference storageReference\_image = storage.getReference().child("profilePic").child(Number\_entered\_by\_user);  StorageReference storageReference\_pdf = storage.getReference().child("proof").child(Number\_entered\_by\_user);  storageReference\_image.putFile(Uri.*parse*(Otp\_details[0])).addOnCompleteListener(new OnCompleteListener<UploadTask.TaskSnapshot>() {  @Override  public void onComplete(@NonNull Task<UploadTask.TaskSnapshot> task) {  if (task.isSuccessful()) {  storageReference\_pdf.putFile(Uri.*parse*(Otp\_details[7])).addOnCompleteListener(new OnCompleteListener<UploadTask.TaskSnapshot>() {  @Override  public void onComplete(@NonNull Task<UploadTask.TaskSnapshot> task) {  if (task.isSuccessful()) {  storageReference\_image.getDownloadUrl().addOnSuccessListener(new OnSuccessListener<Uri>() {  @Override  public void onSuccess(Uri uri) {  imageURI = uri.toString();  storageReference\_pdf.getDownloadUrl().addOnSuccessListener(new OnSuccessListener<Uri>() {  @Override  public void onSuccess(Uri uri) {  pdfUri = uri.toString();  Users users = new Users(auth.getUid(), imageURI, Otp\_details[1], Otp\_details[2], Otp\_details[3], Otp\_details[4], Otp\_details[5], Otp\_details[6], pdfUri, forgetNumber);   reference.setValue(users).addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {  if (task.isSuccessful()) {  dialog.dismissDialog();  Intent dashBoard = new Intent(Otp\_Screen2.this, DashBoard\_Screen.class);  dashBoard.putExtra("profile Values", forgetNumber);  startActivity(dashBoard);  finish();  } else {  Toast.*makeText*(Otp\_Screen2.this, "Error in creating users", Toast.*LENGTH\_SHORT*).show();  }  }  });  }  });  }  });  } else {  Toast.*makeText*(Otp\_Screen2.this, "pdf inserting error", Toast.*LENGTH\_SHORT*).show();  dialog.dismissDialog();  }  }  });  } else {  Toast.*makeText*(Otp\_Screen2.this, "image inserting error", Toast.*LENGTH\_SHORT*).show();  dialog.dismissDialog();  }  }  });  }  }  (Otp\_Screen1.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.Button; import android.widget.EditText; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity;  import com.example.bloodgenix.R; import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query; import com.google.firebase.database.ValueEventListener; import com.hbb20.CountryCodePicker;  public class Otp\_Screen\_1 extends AppCompatActivity {   public String details\_2 [] = new String[15];  Button getOtpBtn;  CountryCodePicker codePicker\_2;  EditText mobileNumb;  String phoneNumberPattern="^(\\+\\d{1,3}( )?)?((\\(\\d{1,3}\\))|\\d{1,3})[- .]?\\d{3,4}[- .]?\\d{4}$";  String countryWithPlus, \_full\_number;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_otp\_screen1*);   //Storing Values from details\_2  Intent i2 = getIntent();  details\_2= i2.getStringArrayExtra("details\_2");   //Country Code Picker Code  codePicker\_2 = findViewById(R.id.*codePicker\_2*);  codePicker\_2.setOnCountryChangeListener(new CountryCodePicker.OnCountryChangeListener() {  @Override  public void onCountrySelected() {  countryWithPlus = codePicker\_2.getSelectedCountryCodeWithPlus();  }  });   countryWithPlus = codePicker\_2.getSelectedCountryCodeWithPlus();   mobileNumb = findViewById(R.id.*mobileNumb*);   getOtpBtn = findViewById(R.id.*getOtpBtn*);  getOtpBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  if(mobileNumb.getText().toString().trim().isEmpty() == false){  if(mobileNumb.getText().length() == 10 && !mobileNumb.getText().toString().equals("1234567890")){  Query personCheck = FirebaseDatabase.*getInstance*().getReference("Users").orderByChild("mobileNumber").equalTo("+91 "+mobileNumb.getText().toString());  personCheck.addValueEventListener(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()){  Toast.*makeText*(Otp\_Screen\_1.this, "Already registered user", Toast.*LENGTH\_SHORT*).show();  }else{  Intent Otp2 = new Intent(Otp\_Screen\_1.this,Otp\_Screen2.class);  String \_full\_number = countryWithPlus+" "+mobileNumb.getText().toString();  Otp2.putExtra("Otp\_value",details\_2);  Otp2.putExtra("phoneNo", \_full\_number);  startActivity(Otp2);  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });   }else{  Toast.*makeText*(Otp\_Screen\_1.this, "Invalid number", Toast.*LENGTH\_SHORT*).show();  }   }else{  Toast.*makeText*(Otp\_Screen\_1.this, "Enter Mobile Number", Toast.*LENGTH\_SHORT*).show();  }   }  });   }  }  (ProfileView.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.EditText; import android.widget.ImageButton; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity;  import com.bumptech.glide.Glide; import com.example.bloodgenix.R; import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query; import com.google.firebase.database.ValueEventListener;  import de.hdodenhof.circleimageview.CircleImageView;  public class ProfileView extends AppCompatActivity {   CircleImageView profileImageView;  EditText personName, personBlood, personGender;  String phoneNumb;  ImageButton cancelBtnProfile, directMessage, block, report;  String chatData [] = new String[5];    @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_profile\_view*);   Intent p = getIntent();  phoneNumb = p.getStringExtra("mobile number");   personBlood = findViewById(R.id.*personBlood*);  personGender = findViewById(R.id.*personGender*);  personName = findViewById(R.id.*personName*);  profileImageView = findViewById(R.id.*profileImageView*);  cancelBtnProfile = findViewById(R.id.*cancelBtnProfile*);  directMessage = findViewById(R.id.*directMessage*);  block = findViewById(R.id.*bottom\_navigation*);  report = findViewById(R.id.*report*);   cancelBtnProfile.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent searchPage = new Intent(ProfileView.this,BloodDnonorSearch.class);  startActivity(searchPage);  }  });   String \_number = phoneNumb.substring(4, phoneNumb.length());  Query checkUser = FirebaseDatabase.*getInstance*().getReference("Users").orderByChild("mobileNumber").equalTo(phoneNumb);  checkUser.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()) {  String \_fullName = snapshot.child(\_number).child("fullName").getValue(String.class);  String \_emailId = snapshot.child(\_number).child("emailId").getValue(String.class);  String \_gender = snapshot.child(\_number).child("gender").getValue(String.class);  String \_dob = snapshot.child(\_number).child("d\_o\_b").getValue(String.class);  String \_img = snapshot.child(\_number).child("profileImg").getValue(String.class);  String \_auth = snapshot.child(\_number).child("uid").getValue(String.class);  personName.setText(\_fullName);  personGender.setText(\_gender);  chatData [0] = phoneNumb;  chatData [1] = \_fullName;  chatData [2] = \_img;  chatData [3] = \_auth;  Glide.*with*(ProfileView.this).load(\_img).into(profileImageView);  Query donorDetails = FirebaseDatabase.*getInstance*().getReference("DonationDetails").orderByChild("phoneNumber").equalTo(phoneNumb);  donorDetails.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot2) {  if (snapshot2.exists()) {  //TAKING VALUES FROM DONATION SIDE.  String \_bloodGroup = snapshot2.child(phoneNumb).child("blGroup").getValue(String.class);  personBlood.setText(\_bloodGroup);  } else {  //TAKING VALUES FROM RECIPIENT SIDE  Query recipientDetails = FirebaseDatabase.*getInstance*().getReference("RecipientDetails").orderByChild("phoneNumber").equalTo(phoneNumb);  recipientDetails.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot3) {  if (snapshot3.exists()) {   String \_bloodGroup = snapshot3.child(phoneNumb).child("bGroupRecipient").getValue(String.class);  personBlood.setText(\_bloodGroup);   } else {  personBlood.setText("Not applied");  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });   } else {  Toast.*makeText*(ProfileView.this, "no such user ", Toast.*LENGTH\_SHORT*).show();  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {  Toast.*makeText*(ProfileView.this, "Something issue in extracting", Toast.*LENGTH\_SHORT*).show();  }  });    directMessage.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent chatSection = new Intent(ProfileView.this,ChatActivity.class);  chatSection.putExtra("chatData",chatData);  startActivity(chatSection);  }  });  } }  (SetPassword.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.Button; import android.widget.ImageButton; import android.widget.Toast;  import androidx.appcompat.app.AppCompatActivity;  import com.example.bloodgenix.R; import com.google.android.material.textfield.TextInputEditText; import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase;  public class SetPassword extends AppCompatActivity {   TextInputEditText newPassword, newPasswordConfirm;  Button updatePassword;  ImageButton setPasswordBack;   Boolean chckpass;  String phoneNo;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_set\_password*);   Intent i9 = getIntent();  phoneNo = i9.getStringExtra("phoneNo");   newPassword = findViewById(R.id.*newPassword*);  newPasswordConfirm = findViewById(R.id.*newPasswordConfirm*);  updatePassword = findViewById(R.id.*updatePassword*);  setPasswordBack = findViewById(R.id.*setPasswordBack*);   setPasswordBack.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  startActivity(new Intent(getApplicationContext(), ForgetPassword.class));  finish();  }  });   updatePassword.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  if (passValidate(newPassword.getText().toString(), newPasswordConfirm.getText().toString())){  newPassword.setError(null);  newPasswordConfirm.setError(null);  DatabaseReference reference = FirebaseDatabase.*getInstance*().getReference("Users");  reference.child(phoneNo.substring(4,phoneNo.length())).child("password").setValue(newPassword.getText().toString());  Toast.*makeText*(SetPassword.this, "Successfully updated", Toast.*LENGTH\_SHORT*).show();  startActivity(new Intent(getApplicationContext(), Welcome.class));  finish();  }  }  });  }   private boolean passValidate(String newPass, String confirmNewPass) {   if (newPass.equals(confirmNewPass)){  if (newPass.equals("12345678") || newPass.equals("123456789")){  newPassword.setError("weak password");  chckpass = false;  }  else if(newPass.length() <=6 || confirmNewPass.length() <=6){  newPassword.setError("To small pass");  chckpass = false;  }  else{  chckpass = true;  }  }else{  newPasswordConfirm.setError("Not matches");  chckpass = false;  }  return chckpass;  } }  (Settings.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.ImageView; import android.widget.LinearLayout; import android.widget.Toast;  import androidx.appcompat.app.AppCompatActivity;  import com.example.bloodgenix.R; import com.example.bloodgenix.SessionManager;  import java.util.HashMap;  public class Settings extends AppCompatActivity {   private ImageView backBtn;  LinearLayout FAQLayout, helpLayout;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_settings*);   backBtn = findViewById(R.id.*backButtons*);  FAQLayout = findViewById(R.id.*FAQLayout*);  helpLayout = findViewById(R.id.*helpLayout*);  SessionManager sessionManager = new SessionManager(this,"userLoginSession");  HashMap<String, String> userDetails = sessionManager.getUserDetailsFromSession();   String phoneNumb = userDetails.get(SessionManager.*KEY\_PHONENUMBER*);   backBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent dashBoard = new Intent(Settings.this, DashBoard\_Screen.class);  dashBoard.putExtra("profile Values", "+91 "+phoneNumb);  startActivity(dashBoard);  }  });   FAQLayout.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Toast.*makeText*(Settings.this, "FAQ's clicked", Toast.*LENGTH\_SHORT*).show();  }  });   helpLayout.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Toast.*makeText*(Settings.this, "Help clicked", Toast.*LENGTH\_SHORT*).show();  }  });  } }  (Signup\_Screen\_1.java)  package com.example.bloodgenix.activities;  import android.content.Context; import android.content.Intent; import android.net.Uri; import android.os.Bundle; import android.view.View; import android.widget.ImageButton; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity;  import com.example.bloodgenix.R; import com.github.drjacky.imagepicker.ImagePicker; import com.google.android.gms.tasks.OnSuccessListener; import com.google.android.material.textfield.TextInputEditText; import com.google.firebase.storage.FirebaseStorage; import com.google.firebase.storage.StorageReference;  import de.hdodenhof.circleimageview.CircleImageView;  public class Signup\_Screen\_1 extends AppCompatActivity {   ImageButton nextButton\_1, profilePickerBtn;  CircleImageView profileImage, profileImage2;  TextInputEditText fullName, userName, emailId, password;  Uri profile\_uri;  Context context;   StorageReference storageReference;   public String details [] = new String[15];   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_signup\_screen1*);   profileImage = findViewById(R.id.*profileImage*);  profileImage2 = findViewById(R.id.*profileImage2*);  profilePickerBtn = findViewById(R.id.*profilePickerBtn*);  fullName = findViewById(R.id.*fullName*);  userName = findViewById(R.id.*userName*);  emailId = findViewById(R.id.*emailId*);  password = findViewById(R.id.*password*);    profilePickerBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  ImagePicker.*Companion*.with(Signup\_Screen\_1.this)  .crop() //Crop image(Optional), Check Customization for more option  .maxResultSize(1080,1080)  .start();  }  });   nextButton\_1 = findViewById(R.id.*nextButton\_1*);  nextButton\_1.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  if(fullName.getText().toString().isEmpty() &&  userName.getText().toString().isEmpty() &&  emailId.getText().toString().isEmpty() &&  password.getText().toString().isEmpty()){  Toast.*makeText*(Signup\_Screen\_1.this, "Many fields left empty", Toast.*LENGTH\_SHORT*).show();  }else{   if (profileImage.getDrawable() == null){   fetchDefault();  profileImage2.setImageURI(profile\_uri); // Toast.makeText(Signup\_Screen\_1.this, profile\_uri.toString(), Toast.LENGTH\_SHORT).show();  }  if (fullName.getText().toString().isEmpty()){  fullName.setError("Enter name");  }  if (userName.getText().toString().isEmpty()){  userName.setError("Enter user name");  }if (emailId.getText().toString().isEmpty()){  emailId.setError("Enter email");  }if (password.getText().toString().isEmpty()){  password.setError("Enter password");  }  try {  Intent signup\_2 = new Intent(Signup\_Screen\_1.this,Signup\_Screen\_2.class);   details[0]=profile\_uri.toString();  details[1]=(fullName.getText().toString());  details[2]=(userName.getText().toString());  details[3]=(emailId.getText().toString());  details[4]=(password.getText().toString());   signup\_2.putExtra("details\_1",details);  Toast.*makeText*(Signup\_Screen\_1.this, details[0], Toast.*LENGTH\_SHORT*).show();  startActivity(signup\_2);  }  catch (Exception e){  System.*out*.println(e.getMessage());  }  }  }  });  }   private void fetchDefault() {  FirebaseStorage storage = FirebaseStorage.*getInstance*();  StorageReference forestRef = storage.getReference().child("profilePic");  forestRef.child("male.jpg").getDownloadUrl().addOnSuccessListener(new OnSuccessListener<Uri>() {  @Override  public void onSuccess(@NonNull Uri uri) {  profile\_uri = uri;  Toast.*makeText*(Signup\_Screen\_1.this, ""+profile\_uri.toString(), Toast.*LENGTH\_SHORT*).show();  }  });  }   @Override  protected void onActivityResult(int requestCode, int resultCode, Intent data) {  super.onActivityResult(requestCode, resultCode, data);  profile\_uri = data.getData();  profileImage.setImageURI(profile\_uri);  } }  (Signup\_Screen\_2.java)  package com.example.bloodgenix.activities;  import android.content.Context; import android.content.Intent; import android.net.Uri; import android.os.Bundle; import android.view.View; import android.widget.AdapterView; import android.widget.ArrayAdapter; import android.widget.AutoCompleteTextView; import android.widget.DatePicker; import android.widget.ImageButton; import android.widget.TextView; import android.widget.Toast;  import androidx.appcompat.app.AppCompatActivity; import androidx.cardview.widget.CardView;  import com.example.bloodgenix.R;  public class Signup\_Screen\_2 extends AppCompatActivity {   private AutoCompleteTextView gender;  ArrayAdapter <String> arrayAdapter;  DatePicker datePicker;  ImageButton docSelection, nextButton\_2,docDeSelect;  TextView docText;  CardView docLayout;  public String details [] = new String[15];  String gender\_type;  Uri docUri;  int Requestcode =1;    @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_signup\_screen2*);  // Getting Profile 1 Values  Intent i = getIntent();  details = i.getStringArrayExtra("details\_1");   // for gender selection code  gender = findViewById(R.id.*Gender*);  String [] gender\_val= getResources().getStringArray(R.array.*gender*);  arrayAdapter= new ArrayAdapter<String>(this,R.layout.*dropdown\_item\_gender*,gender\_val);  gender.setAdapter(arrayAdapter);  gender.setOnItemClickListener(new AdapterView.OnItemClickListener() {  @Override  public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  gender\_type = parent.getItemAtPosition(position).toString();  }  });  Context context = getApplicationContext();  datePicker = findViewById(R.id.*Dob*);  // file Uploading code  docText = findViewById(R.id.*docText*);  docLayout = findViewById(R.id.*docLayout*);  docDeSelect = findViewById(R.id.*docDeSelect*);   docSelection = findViewById(R.id.*docSelection*);  docSelection.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent intent = new Intent();  intent.setType("application/pdf");  intent.setAction(intent.*ACTION\_GET\_CONTENT*);  startActivityForResult(intent,Requestcode);  }  });   docDeSelect.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  docLayout.setVisibility(View.*INVISIBLE*);  docSelection.setVisibility(View.*VISIBLE*);  docDeSelect.setVisibility(View.*INVISIBLE*);  }  });   nextButton\_2 = findViewById(R.id.*nextButton\_2*);  nextButton\_2.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  int day = datePicker.getDayOfMonth();  int month = datePicker.getMonth()+1;  int year = datePicker.getYear();  String dateOfBirth = day+"/"+month+"/"+year;   details[5] = dateOfBirth;  details[6] = gender\_type;  details[7] = String.*valueOf*(docUri);  Intent Otp\_1 = new Intent(Signup\_Screen\_2.this,Otp\_Screen\_1.class);  Otp\_1.putExtra("details\_2",details);  startActivity(Otp\_1);  }  });  }   @Override  protected void onActivityResult(int requestCode, int resultCode, Intent data) {  super.onActivityResult(requestCode, resultCode, data); // Toast.makeText(Signup\_Screen\_2.this, requestCode+" "+data.getData(), Toast.LENGTH\_SHORT).show();  if (requestCode ==1){  if (data == null){  return;  }  docLayout.setVisibility(View.*VISIBLE*);  docUri = data.getData();  docText.setText(data.getDataString().substring(data.getDataString().lastIndexOf("/")+1)+".pdf");  docSelection.setVisibility(View.*INVISIBLE*);  docDeSelect.setVisibility(View.*VISIBLE*);   }  else {  Toast.*makeText*(Signup\_Screen\_2.this, "Please try again ", Toast.*LENGTH\_SHORT*).show();  }  }  }  (User\_Profile\_View.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.net.Uri; import android.os.Bundle; import android.util.Log; import android.view.LayoutInflater; import android.view.MenuItem; import android.view.View; import android.widget.ArrayAdapter; import android.widget.AutoCompleteTextView; import android.widget.Button; import android.widget.DatePicker; import android.widget.ImageButton; import android.widget.LinearLayout; import android.widget.PopupMenu; import android.widget.TextView; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity; import androidx.swiperefreshlayout.widget.SwipeRefreshLayout;  import com.bumptech.glide.Glide; import com.example.bloodgenix.R; import com.github.drjacky.imagepicker.ImagePicker; import com.google.android.gms.tasks.OnCompleteListener; import com.google.android.gms.tasks.OnSuccessListener; import com.google.android.gms.tasks.Task; import com.google.android.material.bottomsheet.BottomSheetDialog; import com.google.android.material.textfield.TextInputEditText; import com.google.android.material.textfield.TextInputLayout; import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query; import com.google.firebase.database.ValueEventListener; import com.google.firebase.storage.FirebaseStorage; import com.google.firebase.storage.StorageReference; import com.google.firebase.storage.UploadTask;  import java.util.HashMap;  import de.hdodenhof.circleimageview.CircleImageView;  public class User\_Profile\_View extends AppCompatActivity implements PopupMenu.OnMenuItemClickListener, View.OnClickListener {   CircleImageView userProfileImage;  ImageButton backBtn, dotMenuBtn;  TextView userProfileName, ProfileUserName, toolbarName;  TextView userProfileBlood, userPersonBleed, userProfileLocation;  TextView userProfileContact, userProfileEmail, userProfileDOB;   //EDIT VIEWS DECLARATION  ImageButton BloodBtn, BleedBtn, ContactBtn;  ImageButton EmailBtn, DOBBtn, cancel\_button, locationBtn;  ImageButton editBtn, picUpdate;   LinearLayout bleedLayout;  String mobileNumber;  DatePicker Dob;  int donorFlag = 0, recipientFlag = 0;  boolean updatePic = false;  Uri profile\_uri;  Button deleteChoice;   //BOTTOM SHEET DECLARATION  BottomSheetDialog sheetDialog2;  TextInputEditText editView;  TextInputLayout textInputLayout, textInputLayout2;  AutoCompleteTextView autoView;    //FireBase declaration  FirebaseDatabase database;  DatabaseReference reference;  FirebaseStorage storage;   SwipeRefreshLayout refreshLayout;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_user\_profile\_view*);   viewInitialisation();  btnviewInitialisation();  Intent profile = getIntent();  mobileNumber = profile.getStringExtra("mobile number");  reference = FirebaseDatabase.*getInstance*().getReference();  bleedLayout = findViewById(R.id.*bleedLayout*);  deleteChoice = findViewById(R.id.*deleteChoice*);   storage = FirebaseStorage.*getInstance*();   refreshLayout = findViewById(R.id.*refreshLayout*);  refreshLayout.setOnRefreshListener(new SwipeRefreshLayout.OnRefreshListener() {  @Override  public void onRefresh() {  Intent i\_again = new Intent(getApplicationContext(), User\_Profile\_View.class);  i\_again.putExtra("mobile number", mobileNumber);   String TempNumber = mobileNumber.substring(4, mobileNumber.length());  StorageReference forestRef = storage.getReference().child("profilePic");  HashMap hashMapImg = new HashMap();  if (updatePic == true) {  try {  forestRef.child(TempNumber).delete().addOnSuccessListener(new OnSuccessListener<Void>() {  @Override  public void onSuccess(Void unused) {  forestRef.child(TempNumber).putFile(profile\_uri).addOnCompleteListener(new OnCompleteListener<UploadTask.TaskSnapshot>() {  @Override  public void onComplete(@NonNull Task<UploadTask.TaskSnapshot> task) {  if (task.isSuccessful()) {  forestRef.child(TempNumber).getDownloadUrl().addOnSuccessListener(new OnSuccessListener<Uri>() {  @Override  public void onSuccess(Uri uri) {  hashMapImg.put("profileImg", uri.toString());  FirebaseDatabase.*getInstance*().getReference().child("Users").child(TempNumber).updateChildren(hashMapImg).addOnSuccessListener(new OnSuccessListener() {  @Override  public void onSuccess(Object o) {  if (chckDonor() == 1){  hashMapImg.clear();  hashMapImg.put("personProfileImg", uri.toString());  FirebaseDatabase.*getInstance*().getReference().child("DonationDetails").child(mobileNumber).updateChildren(hashMapImg).addOnSuccessListener(new OnSuccessListener() {  @Override  public void onSuccess(Object o) {  updatePic = false;  startActivity(i\_again);  }  });  }else if (chckRecipient() == 1){  hashMapImg.clear();  hashMapImg.put("personProfile", uri.toString());  FirebaseDatabase.*getInstance*().getReference().child("RecipientDetails").child(mobileNumber).updateChildren(hashMapImg).addOnSuccessListener(new OnSuccessListener() {  @Override  public void onSuccess(Object o) {  updatePic = false;  startActivity(i\_again);  }  });  }else{  updatePic = false;  startActivity(i\_again);  }  }  });  }  });  } else {  Toast.*makeText*(User\_Profile\_View.this, "Not Updated", Toast.*LENGTH\_SHORT*).show();  }  }  });  }  });   } catch (Exception ex) {  Log.*i*("ERROR :", ex.getMessage());  }  }else{  startActivity(i\_again);  }  }  });   dotMenuBtn = findViewById(R.id.*dotMenuBtn*);  dotMenuBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  PopupMenu popupMenu = new PopupMenu(User\_Profile\_View.this, v);  popupMenu.setOnMenuItemClickListener(User\_Profile\_View.this);  popupMenu.inflate(R.menu.*dot\_menu*);  popupMenu.show();  }  });   deleteChoice.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  if (chckDonor() == 1) {  FirebaseDatabase.*getInstance*().getReference("DonationDetails").child(mobileNumber).removeValue().addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {  if (task.isSuccessful()) {  Toast.*makeText*(User\_Profile\_View.this, "Donation deleted", Toast.*LENGTH\_SHORT*).show();  }  }  });  } else if (chckRecipient() == 1) {  FirebaseDatabase.*getInstance*().getReference("RecipientDetails").child(mobileNumber).removeValue().addOnCompleteListener(new OnCompleteListener<Void>() {  @Override  public void onComplete(@NonNull Task<Void> task) {  if (task.isSuccessful()) {  Toast.*makeText*(User\_Profile\_View.this, "Recipient deleted", Toast.*LENGTH\_SHORT*).show();  }  }  });  }  }  });   backBtn = findViewById(R.id.*bacBtn*);  backBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent back = new Intent(User\_Profile\_View.this, DashBoard\_Screen.class);  back.putExtra("profile Values", mobileNumber);  startActivity(back);  finish();  }  });   userProfileImage.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent fullView = new Intent(User\_Profile\_View.this, fullProfileView.class);  fullView.putExtra("phone Number", mobileNumber);  startActivity(fullView);  }  });   String \_number = mobileNumber.substring(4, mobileNumber.length());  Query checkUser = FirebaseDatabase.*getInstance*().getReference("Users").orderByChild("mobileNumber").equalTo(mobileNumber);  checkUser.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()) {  String \_fullName = snapshot.child(\_number).child("fullName").getValue(String.class);  String \_userName = snapshot.child(\_number).child("userName").getValue(String.class);  String \_emailId = snapshot.child(\_number).child("emailId").getValue(String.class);  String \_gender = snapshot.child(\_number).child("gender").getValue(String.class);  String \_dob = snapshot.child(\_number).child("d\_o\_b").getValue(String.class);  String \_img = snapshot.child(\_number).child("profileImg").getValue(String.class);   userProfileName.setText(\_fullName);  ProfileUserName.setText(\_userName);  userProfileContact.setText(mobileNumber);  userProfileEmail.setText(\_emailId);  userProfileDOB.setText(\_dob);  Glide.*with*(User\_Profile\_View.this).load(\_img).into(userProfileImage);  if (chckDonor() == 0) {  if (chckRecipient() == 0) {  userProfileBlood.setText("Not applied");  userPersonBleed.setText("Not applied");  userProfileLocation.setText("Not known");  }  }  } else {  Toast.*makeText*(User\_Profile\_View.this, "no person found", Toast.*LENGTH\_SHORT*).show();  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {   }  });   buttonListener();  }   //BUTTON LISTENERS CREATED  private void buttonListener() {  BloodBtn.setOnClickListener(this);  BloodBtn.setOnClickListener(this);  BleedBtn.setOnClickListener(this);  ContactBtn.setOnClickListener(this);  EmailBtn.setOnClickListener(this);  locationBtn.setOnClickListener(this);  DOBBtn.setOnClickListener(this);  picUpdate.setOnClickListener(this);   }   //BOTTOM SHEET DIALOG CREATION FOR EDITTEXT VALUES  private BottomSheetDialog BottomSheet() {  BottomSheetDialog sheet = new BottomSheetDialog(User\_Profile\_View.this, R.style.*BottomSheetStyle*);  View view = LayoutInflater.*from*(User\_Profile\_View.this).inflate(R.layout.*edit\_bottom\_layout*, (LinearLayout) findViewById(R.id.*sheet2*));  sheet.setContentView(view);   //Initialising the views  editView = sheet.findViewById(R.id.*editView*);  editBtn = sheet.findViewById(R.id.*editBtn*);  cancel\_button = sheet.findViewById(R.id.*cancel\_button*);  textInputLayout = sheet.findViewById(R.id.*textInputLayout*);  textInputLayout2 = sheet.findViewById(R.id.*textInputLayout2*);  Dob = sheet.findViewById(R.id.*Dob*);  textInputLayout.setVisibility(View.*VISIBLE*);   sheet.show();  cancel\_button.setOnClickListener(v1 -> sheet.dismiss());  return sheet;  }   //BOTTOM SHEET DIALOG CREATION FOR DROPDOWN  private BottomSheetDialog BottomSheetDrop() {   BottomSheetDialog sheet = new BottomSheetDialog(User\_Profile\_View.this, R.style.*BottomSheetStyle*);  View view = LayoutInflater.*from*(User\_Profile\_View.this).inflate(R.layout.*edit\_bottom\_layout*, (LinearLayout) findViewById(R.id.*sheet2*));  sheet.setContentView(view);   //Initialising the views  editView = sheet.findViewById(R.id.*editView*);  editBtn = sheet.findViewById(R.id.*editBtn*);  cancel\_button = sheet.findViewById(R.id.*cancel\_button*);  autoView = sheet.findViewById(R.id.*autoView*);  textInputLayout = sheet.findViewById(R.id.*textInputLayout*);  textInputLayout2 = sheet.findViewById(R.id.*textInputLayout2*);  textInputLayout2.setVisibility(View.*VISIBLE*);   sheet.show();  cancel\_button.setOnClickListener(v1 -> sheet.dismiss());  return sheet;  }   //INITIALISING BUTTONS VIEWS  private void btnviewInitialisation() {  BloodBtn = findViewById(R.id.*BloodBtn*);  BleedBtn = findViewById(R.id.*BleedBtn*);  ContactBtn = findViewById(R.id.*ContactBtn*);  EmailBtn = findViewById(R.id.*EmailBtn*);  locationBtn = findViewById(R.id.*locationBtn*);  DOBBtn = findViewById(R.id.*DOBBtn*);  picUpdate = findViewById(R.id.*picUpdate*);  }   //ON CLICK DEFINITION  @Override  public void onClick(View v) {  ArrayAdapter<String> arrayAdapter;  String auto[];  switch (v.getId()) {   case R.id.*picUpdate*:  ImagePicker.*Companion*.with(User\_Profile\_View.this)  .crop() //Crop image(Optional), Check Customization for more option  .maxResultSize(1080, 1080)  .start();  break;   case R.id.*BloodBtn*:  sheetDialog2 = BottomSheetDrop();  textInputLayout2.setHint("Blood Group");  auto = getResources().getStringArray(R.array.*bloodgroup*);  arrayAdapter = new ArrayAdapter<String>(this, R.layout.*items\_list\_blood*, auto);  autoView.setAdapter(arrayAdapter);   editBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  HashMap hashMap = new HashMap();   if (chckDonor() == 1) {  hashMap.put("blGroup", autoView.getText().toString());  FirebaseDatabase.*getInstance*().getReference().child("DonationDetails").child(mobileNumber).updateChildren(hashMap).addOnSuccessListener(new OnSuccessListener() {  @Override  public void onSuccess(Object o) {  Toast.*makeText*(User\_Profile\_View.this, "Successfully edited", Toast.*LENGTH\_SHORT*).show();  textInputLayout2.setVisibility(View.*INVISIBLE*);  sheetDialog2.dismiss();  }  });  } else if (chckRecipient() == 1) {  hashMap.put("bGroupRecipient", autoView.getText().toString());  FirebaseDatabase.*getInstance*().getReference().child("RecipientDetails").child(mobileNumber).updateChildren(hashMap).addOnSuccessListener(new OnSuccessListener() {  @Override  public void onSuccess(Object o) {  Toast.*makeText*(User\_Profile\_View.this, "Successfully edited", Toast.*LENGTH\_SHORT*).show();  textInputLayout2.setVisibility(View.*INVISIBLE*);  sheetDialog2.dismiss();  }  });  }   }  });  break;  case R.id.*BleedBtn*:  sheetDialog2 = BottomSheetDrop();  textInputLayout2.setHint("Last Donation Period");  auto = getResources().getStringArray(R.array.*periodMonth*);  arrayAdapter = new ArrayAdapter<String>(this, R.layout.*donation\_period\_dropdown*, auto);  autoView.setAdapter(arrayAdapter);   editBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  HashMap hashMap = new HashMap();   if (chckDonor() == 1) {  hashMap.put("donateMonth", autoView.getText().toString());  FirebaseDatabase.*getInstance*().getReference().child("DonationDetails").child(mobileNumber).updateChildren(hashMap).addOnSuccessListener(new OnSuccessListener() {  @Override  public void onSuccess(Object o) {  Toast.*makeText*(User\_Profile\_View.this, "Successfully edited", Toast.*LENGTH\_SHORT*).show();  textInputLayout2.setVisibility(View.*INVISIBLE*);   sheetDialog2.dismiss();  }  });  } else if (chckRecipient() == 1) {  Toast.*makeText*(User\_Profile\_View.this, "You are applied for blood", Toast.*LENGTH\_SHORT*).show();  sheetDialog2.dismiss();  }  textInputLayout2.setEnabled(true);  }   });  break;  case R.id.*ContactBtn*:  Toast.*makeText*(User\_Profile\_View.this, "Cannot update contact", Toast.*LENGTH\_SHORT*).show(); // sheetDialog2 = BottomSheet(); // textInputLayout.setHint("Contact");  break;  case R.id.*locationBtn*:  sheetDialog2 = BottomSheet();  textInputLayout.setHint("Location");   editBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  HashMap hashMap = new HashMap();   if (chckDonor() == 1) {  hashMap.put("donorLocation", editView.getText().toString());  FirebaseDatabase.*getInstance*().getReference().child("DonationDetails").child(mobileNumber).updateChildren(hashMap).addOnSuccessListener(new OnSuccessListener() {  @Override  public void onSuccess(Object o) {  Toast.*makeText*(User\_Profile\_View.this, "Successfully edited", Toast.*LENGTH\_SHORT*).show();  textInputLayout.setVisibility(View.*INVISIBLE*);   sheetDialog2.dismiss();  }  });  } else if (chckRecipient() == 1) {  hashMap.put("locationRecipient", editView.getText().toString());  FirebaseDatabase.*getInstance*().getReference().child("RecipientDetails").child(mobileNumber).updateChildren(hashMap).addOnSuccessListener(new OnSuccessListener() {  @Override  public void onSuccess(Object o) {  Toast.*makeText*(User\_Profile\_View.this, "Successfully edited", Toast.*LENGTH\_SHORT*).show();  textInputLayout2.setVisibility(View.*INVISIBLE*);   sheetDialog2.dismiss();  }  });  }  }   });  break;  case R.id.*EmailBtn*:  sheetDialog2 = BottomSheet();  textInputLayout.setHint("Email");   editBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  HashMap hashMap = new HashMap();  hashMap.put("emailId", editView.getText().toString());  FirebaseDatabase.*getInstance*().getReference().child("Users").child(mobileNumber.substring(4, mobileNumber.length())).updateChildren(hashMap).addOnSuccessListener(new OnSuccessListener() {  @Override  public void onSuccess(Object o) {  Toast.*makeText*(User\_Profile\_View.this, "Successfully edited", Toast.*LENGTH\_SHORT*).show();  textInputLayout.setVisibility(View.*INVISIBLE*);   sheetDialog2.dismiss();  }  });  }  });  break;  case R.id.*DOBBtn*:  sheetDialog2 = BottomSheet();  textInputLayout.setVisibility(View.*INVISIBLE*);  Dob.setVisibility(View.*VISIBLE*);   editBtn.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  HashMap hashMap = new HashMap();  int day = Dob.getDayOfMonth();  int month = Dob.getMonth() + 1;  int year = Dob.getYear();  String dateOfBirth = day + "/" + month + "/" + year;  hashMap.put("d\_o\_b", dateOfBirth);  FirebaseDatabase.*getInstance*().getReference().child("Users").child(mobileNumber.substring(4, mobileNumber.length())).updateChildren(hashMap).addOnSuccessListener(new OnSuccessListener() {  @Override  public void onSuccess(Object o) {  Toast.*makeText*(User\_Profile\_View.this, "Successfully edited", Toast.*LENGTH\_SHORT*).show();  Dob.setVisibility(View.*INVISIBLE*);  sheetDialog2.dismiss();  }  });  }  });  break;  default:  break;  }  }    //FUNCTION TO CHECK LOGON PERSON IS RECIPIENT  private int chckRecipient() {  Query recipientDetails = FirebaseDatabase.*getInstance*().getReference("RecipientDetails").orderByChild("phoneNumber").equalTo(mobileNumber);  recipientDetails.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot3) {  if (snapshot3.exists()) {  bleedLayout.setVisibility(View.*INVISIBLE*);  String \_bloodGroup = snapshot3.child(mobileNumber).child("bGroupRecipient").getValue(String.class);  String \_location = snapshot3.child(mobileNumber).child("locationRecipient").getValue(String.class);  userProfileBlood.setText(\_bloodGroup);   userProfileLocation.setText(\_location);  recipientFlag = 1;  } else {  recipientFlag = 0;  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {  recipientFlag = 0;  }  });  return recipientFlag;  }   //FUNCTION TO CHECK LOGON PERSON IS DONOR  private int chckDonor() {   Query donorDetails = FirebaseDatabase.*getInstance*().getReference("DonationDetails").orderByChild("phoneNumber").equalTo(mobileNumber);  donorDetails.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot2) {  if (snapshot2.exists()) {  bleedLayout.setVisibility(View.*VISIBLE*);  String \_bloodGroup = snapshot2.child(mobileNumber).child("blGroup").getValue(String.class);  String \_bleed = snapshot2.child(mobileNumber).child("donateMonth").getValue(String.class);  String \_location = snapshot2.child(mobileNumber).child("donorLocation").getValue(String.class);  userProfileBlood.setText(\_bloodGroup);  userProfileLocation.setText(\_location);  userPersonBleed.setText(\_bleed);   if (\_bleed.length() <= 0) {  userPersonBleed.setText("Not Mention");  } else {  userPersonBleed.setText(\_bleed);  }  donorFlag = 1;  } else {  donorFlag = 0;  }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {  donorFlag = 0;  }  });  return donorFlag;  }   private void viewInitialisation() {  toolbarName = findViewById(R.id.*toolbarName*);  userProfileImage = findViewById(R.id.*userProfileImage*);  userProfileName = findViewById(R.id.*userProfileName*);  ProfileUserName = findViewById(R.id.*ProfileUserName*);  userProfileBlood = findViewById(R.id.*userProfileBlood*);  userPersonBleed = findViewById(R.id.*userPersonBleed*);  userProfileLocation = findViewById(R.id.*userProfileLocation*);  userProfileContact = findViewById(R.id.*userProfileContact*);  userProfileEmail = findViewById(R.id.*userProfileEmail*);  userProfileDOB = findViewById(R.id.*userProfileDOB*);  }   @Override  public boolean onMenuItemClick(MenuItem item) {  switch (item.getItemId()) {  case R.id.*forgotPassword*:  Intent forgotPage = new Intent(getApplicationContext(), ForgetPassword.class);  forgotPage.putExtra("whatToDo", "UserProfile");  forgotPage.putExtra("mobile Number", mobileNumber);  startActivity(forgotPage);  finish();  return true;   default:  return false;  }  }   @Override  protected void onActivityResult(int requestCode, int resultCode, Intent data) {  super.onActivityResult(requestCode, resultCode, data);  profile\_uri = data.getData();  userProfileImage.setImageURI(profile\_uri);  updatePic = true;    } }  (Welcome.java)  package com.example.bloodgenix.activities;  import android.content.Intent; import android.graphics.Bitmap; import android.os.Bundle; import android.text.TextUtils; import android.view.LayoutInflater; import android.view.View; import android.widget.Button; import android.widget.CheckBox; import android.widget.ImageButton; import android.widget.LinearLayout; import android.widget.Toast;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity;  import com.example.bloodgenix.LoadingDialog; import com.example.bloodgenix.R; import com.example.bloodgenix.SessionManager; import com.google.android.material.bottomsheet.BottomSheetDialog; import com.google.android.material.textfield.TextInputEditText; import com.google.android.material.textfield.TextInputLayout; import com.google.firebase.auth.FirebaseAuth; import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.Query; import com.google.firebase.database.ValueEventListener; import com.google.firebase.storage.StorageReference; import com.hbb20.CountryCodePicker;  import java.util.HashMap; import java.util.Objects;  public class Welcome extends AppCompatActivity {   Button loginView;  BottomSheetDialog sheetDialog;  Button signUpBtn, Forgot;  String profile\_details[] = new String[15];  Bitmap bitmap;    //Bottom Sheet Dialog  TextInputLayout passwordLayout;  LoadingDialog dialog;  CheckBox rememberBtn;  Button loginBtn;  ImageButton back\_btn;  TextInputEditText mobileNo, Password;  CountryCodePicker picker;  TextInputLayout mobLayout;  SessionManager sessionManager;  String countryWithPlus;  String phoneNumberPattern = "^(\\+\\d{1,3}( )?)?((\\(\\d{1,3}\\))|\\d{1,3})[- .]?\\d{3,4}[- .]?\\d{4}$";   //FireBase variable initialise  FirebaseAuth mAuth;  StorageReference storageReference;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_welcome*);   loginView = findViewById(R.id.*loginView*);  mAuth = FirebaseAuth.*getInstance*();   dialog = new LoadingDialog(Welcome.this);   loginView.setOnClickListener(v -> {   sheetDialog = new BottomSheetDialog(Welcome.this, R.style.*BottomSheetStyle*);  View view = LayoutInflater.*from*(Welcome.this).inflate(R.layout.*activity\_login\_screen*, (LinearLayout) findViewById(R.id.*sheet*));  sheetDialog.setContentView(view);   mobileNo = sheetDialog.findViewById(R.id.*mobileNo*);  Password = sheetDialog.findViewById(R.id.*Password*);  picker = sheetDialog.findViewById(R.id.*codePicker*);  back\_btn = sheetDialog.findViewById(R.id.*back\_btn*);  passwordLayout = sheetDialog.findViewById(R.id.*passwordLayout*);  rememberBtn = sheetDialog.findViewById(R.id.*rememberBtn*);  Forgot = sheetDialog.findViewById(R.id.*Forgot*);  sheetDialog.show();   sessionManager = new SessionManager(Welcome.this, SessionManager.*SESSION\_REMEMBERME*);  if (sessionManager.checkRememberMe()){  HashMap<String, String> rememberMeDetails = sessionManager.getRememberMeDetailsFromSession();  mobileNo.setText(rememberMeDetails.get(sessionManager.*KEY\_SESSION\_PHONENUMBER*));  Password.setText(rememberMeDetails.get(sessionManager.*KEY\_SESSION\_PASSWORD*));  }   listners(sheetDialog);  back\_btn.setOnClickListener(v12 -> {  sheetDialog.dismiss();  });   Forgot.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent forget = new Intent(sheetDialog.getContext(), ForgetPassword.class);  forget.putExtra("mobile Number" , mobileNo.getText().toString());  forget.putExtra("whatToDo","WelcomePage");  startActivity(forget);  }  });   loginBtn = sheetDialog.findViewById(R.id.*loginBtn*);  assert loginBtn != null;  loginBtn.setOnClickListener(v1 -> {    String \_number = Objects.*requireNonNull*(mobileNo.getText()).toString().trim();  String \_password = Objects.*requireNonNull*(Password.getText()).toString();  String full\_number = countryWithPlus + " " + \_number;   passwordLayout.setEndIconActivated(false);   if (TextUtils.*isEmpty*(\_number) && TextUtils.*isEmpty*(\_password)) {  Toast.*makeText*(sheetDialog.getContext(), " Enter valid data ", Toast.*LENGTH\_SHORT*).show();  mobileNo.setError("Empty field");  passwordLayout.setEndIconDrawable(0);  Password.setError("Empty field");   } else if (TextUtils.*isEmpty*(\_password)) {  passwordLayout.setEndIconDrawable(0);  Password.setError("Empty field");  } else if (full\_number.matches(phoneNumberPattern) == false) {  mobileNo.setError(" Invalid number ");   } else if (\_password.length() <= 6) {  passwordLayout.setEndIconDrawable(0);  Password.setError(" Small password ");  } else {  mobileNo.setError(null);  Password.setError(null);   dialog.startDialog();  if (rememberBtn.isChecked()){  sessionManager = new SessionManager(Welcome.this, SessionManager.*SESSION\_REMEMBERME*);  sessionManager.createRememberMeSession(\_number, \_password);  }   Query checkUser = FirebaseDatabase.*getInstance*().getReference("Users").orderByChild("mobileNumber").equalTo(full\_number);  checkUser.addListenerForSingleValueEvent(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot snapshot) {  if (snapshot.exists()) {  String system\_password = snapshot.child(\_number).child("password").getValue(String.class);   if (system\_password.equals(\_password)) {   String \_fullName = snapshot.child(\_number).child("fullName").getValue(String.class);  String \_userName = snapshot.child(\_number).child("userName").getValue(String.class);  String \_emailId = snapshot.child(\_number).child("emailId").getValue(String.class);  String \_gender = snapshot.child(\_number).child("gender").getValue(String.class);  String \_password = snapshot.child(\_number).child("password").getValue(String.class);  String \_dob = snapshot.child(\_number).child("d\_o\_b").getValue(String.class);  String \_img = snapshot.child(\_number).child("profileImg").getValue(String.class);  // profile\_details [0] = \_fullName; // profile\_details [1] = \_emailId; // profile\_details [2] = system\_password; // profile\_details [3] =\_dob; // profile\_details [4] =\_gender; // profile\_details [5] = \_number; // profile\_details [6] = \_img;   SessionManager sessionManager = new SessionManager(Welcome.this,SessionManager.*SESSION\_USERSESSION*);  sessionManager.createLoginSession(\_fullName, \_userName, \_emailId, \_number, \_gender, \_password);   dialog.dismissDialog();  Intent dashBoard = new Intent(sheetDialog.getContext(), DashBoard\_Screen.class);  dashBoard.putExtra("profile Values", full\_number);  startActivity(dashBoard);  } else {  dialog.dismissDialog();  Toast.*makeText*(Welcome.this, "Incorrect password", Toast.*LENGTH\_SHORT*).show();   }  } else {  dialog.dismissDialog();  Toast.*makeText*(Welcome.this, "User does not exists", Toast.*LENGTH\_SHORT*).show();   }  }   @Override  public void onCancelled(@NonNull DatabaseError error) {  dialog.dismissDialog();  Toast.*makeText*(sheetDialog.getContext(), error.getMessage(), Toast.*LENGTH\_SHORT*).show();  }  });   }  });   });  signUpBtn = findViewById(R.id.*signUpBtn*);  signUpBtn.setOnClickListener(v13 -> {  Intent signUp = new Intent(Welcome.this, Signup\_Screen\_1.class);  startActivity(signUp);  });   }   private void listners(BottomSheetDialog sheetDialog) {  String code = picker.getSelectedCountryCode();  countryWithPlus = picker.getSelectedCountryCodeWithPlus();   }  }  Adapters  (MessagesAdapter.java)  package com.example.bloodgenix.Adapters;  import static com.example.bloodgenix.activities.ChatActivity.*receiverImg*; import static com.example.bloodgenix.activities.ChatActivity.*senderImg*;  import android.content.Context; import android.view.LayoutInflater; import android.view.View; import android.view.ViewGroup; import android.widget.TextView;  import androidx.annotation.NonNull; import androidx.recyclerview.widget.RecyclerView;  import com.bumptech.glide.Glide; import com.example.bloodgenix.Models.Messages; import com.example.bloodgenix.R;  import java.util.ArrayList;  import de.hdodenhof.circleimageview.CircleImageView;  public class MessagesAdapter extends RecyclerView.Adapter {  Context context;  ArrayList<Messages> messagesArrayList;  int ITEM\_SEND=1;  int ITEM\_RECEIVE=2;  String senderID;   public MessagesAdapter(Context context, ArrayList<Messages> messagesArrayList, String senderID) {  this.context = context;  this.messagesArrayList = messagesArrayList;  this.senderID = senderID;  }   @NonNull  @Override  public RecyclerView.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  if (viewType == ITEM\_SEND){   View view = LayoutInflater.*from*(context).inflate(R.layout.*sender\_layout\_item*, parent, false);  return new SenderViewsHolder(view);   }  else{   View view = LayoutInflater.*from*(context).inflate(R.layout.*receiver\_layout\_item*, parent, false);  return new ReceiverViewsHolder(view);  }   }   @Override  public void onBindViewHolder(@NonNull RecyclerView.ViewHolder holder, int position) {  Messages messages= messagesArrayList.get(position);   if (holder.getClass() == SenderViewsHolder.class){    SenderViewsHolder viewsHolder = (SenderViewsHolder) holder;  viewsHolder.senderMessageTxt.setText(messages.getMessage());    Glide.*with*(context).load(*senderImg*).into(viewsHolder.senderProfileImg);  }else{  ReceiverViewsHolder viewsHolder = (ReceiverViewsHolder) holder;  viewsHolder.senderMessageTxt.setText(messages.getMessage());     Glide.*with*(context).load(*receiverImg*).into(viewsHolder.senderProfileImg);  }  }   @Override  public int getItemCount() {  return messagesArrayList.size();  }   @Override  public int getItemViewType(int position) {  Messages messages = messagesArrayList.get(position);  if (senderID.equals(messages.getSenderID()))  {   return ITEM\_SEND;  }else{   return ITEM\_RECEIVE;  }  }   class SenderViewsHolder extends RecyclerView.ViewHolder{  CircleImageView senderProfileImg;  TextView senderMessageTxt;  public SenderViewsHolder(@NonNull View itemView) {  super(itemView);   senderProfileImg = itemView.findViewById(R.id.*ProfileImg*);  senderMessageTxt = itemView.findViewById(R.id.*MessageTxt*);  }  }   class ReceiverViewsHolder extends RecyclerView.ViewHolder{  CircleImageView senderProfileImg;  TextView senderMessageTxt;  public ReceiverViewsHolder(@NonNull View itemView) {  super(itemView);  senderProfileImg = itemView.findViewById(R.id.*ProfileImg*);  senderMessageTxt = itemView.findViewById(R.id.*MessageTxt*);  }  } }  (NotificationAdapter.java)  package com.example.bloodgenix.Adapters;  import android.content.Context; import android.content.Intent; import android.view.LayoutInflater; import android.view.View; import android.view.ViewGroup; import android.widget.TextView; ` import androidx.annotation.NonNull; import androidx.recyclerview.widget.RecyclerView;  import com.bumptech.glide.Glide; import com.example.bloodgenix.Models.Messages; import com.example.bloodgenix.R; import com.example.bloodgenix.activities.ChatActivity;  import java.text.Format; import java.text.SimpleDateFormat; import java.util.ArrayList; import java.util.Date;  import de.hdodenhof.circleimageview.CircleImageView;  public class NotificationAdapter extends RecyclerView.Adapter<NotificationAdapter.notificationViewHolder>{   Context context;  ArrayList<Messages> messagesArrayList;  String passedData [] = new String[5];  public String senderImg, senderName;   public NotificationAdapter(Context context, ArrayList<Messages> messagesArrayList) {  this.context = context;  this.messagesArrayList = messagesArrayList;  }   @NonNull  @Override  public notificationViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  View view = LayoutInflater.*from*(context).inflate(R.layout.*notificationchat\_layout*, parent, false);  return new notificationViewHolder(view);  }   @Override  public void onBindViewHolder(@NonNull notificationViewHolder holder, int position) {  Messages messages = messagesArrayList.get(position);  holder.chatSenderName.setText(messages.getSenderName());  holder.chatSenderMessage.setText(messages.message);  String timeStampString = convertTime(messages.getTimeStamp());  holder.timeStamp.setText(timeStampString.substring(10,timeStampString.length()-3));  Glide.*with*(context).load(messages.getSenderImg()).into(holder.chatProfile);   holder.itemView.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent new\_Intent = new Intent(context, ChatActivity.class);  passedData [0] = messages.senderID.toString();  passedData [1] = messages.getSenderName();  passedData [2] = messages.getSenderImg();  new\_Intent.putExtra("chatData",passedData);  new\_Intent.putExtra("whatToDo","notificationChat");  context.startActivity(new\_Intent);  }  });  }     @Override  public int getItemCount() {  return messagesArrayList.size();  }   class notificationViewHolder extends RecyclerView.ViewHolder{   CircleImageView chatProfile;  TextView chatSenderName, chatSenderMessage;  TextView timeStamp;  public notificationViewHolder(@NonNull View itemView) {  super(itemView);  chatProfile = itemView.findViewById(R.id.*chatProfile*);  chatSenderName = itemView.findViewById(R.id.*chatSenderName*);  chatSenderMessage = itemView.findViewById(R.id.*chatSenderMessage*);  timeStamp = itemView.findViewById(R.id.*timeStamp*);  }  }  public String convertTime(long time){  Date date = new Date(time);  Format format = new SimpleDateFormat("yyyy MM dd HH:mm:ss");  return format.format(date);  } }  (SliderAdapter.java)  package com.example.bloodgenix.Adapters;  import android.content.Context; import android.view.LayoutInflater; import android.view.View; import android.view.ViewGroup; import android.widget.ImageView; import android.widget.TextView;  import androidx.annotation.NonNull; import androidx.constraintlayout.widget.ConstraintLayout; import androidx.viewpager.widget.PagerAdapter;  import com.example.bloodgenix.R;  public class SliderAdapter extends PagerAdapter {   Context context;  LayoutInflater layoutInflater;  ImageView sliderImage;  TextView sliderTitle, sliderDesc;    public SliderAdapter(Context context) {  this.context = context;  }   public int [] slider\_images={  R.drawable.*ic\_drawer\_1*, R.drawable.*ic\_drawer\_2*, R.drawable.*ic\_drawer\_3* };   public String [] title={  "Donate Blood",  "Search Blood Donor",  "Explore update near you"  };   public int[] desc={R.string.*blood\_donation*,  R.string.*Search\_blood\_donor*,  R.string.*Explore\_update* };   @Override  public int getCount() {  return title.length;  }   @Override  public boolean isViewFromObject( View view, Object object) {  return view == (ConstraintLayout)object;  }   @Override  public Object instantiateItem( ViewGroup container, int position) {   layoutInflater = (LayoutInflater) context.getSystemService(context.*LAYOUT\_INFLATER\_SERVICE*);  View view = layoutInflater.inflate(R.layout.*slide\_layout\_dashboard*, container,false);   sliderImage = (ImageView) view.findViewById(R.id.*sliderImage*);  sliderTitle = (TextView) view.findViewById(R.id.*sliderTitle*);  sliderDesc = (TextView) view.findViewById(R.id.*sliderDesc*);   sliderImage.setImageResource(slider\_images[position]);  sliderTitle.setText(title[position]);  sliderDesc.setText(desc[position]);  container.addView(view);   return view;  }   @Override  public void destroyItem(@NonNull ViewGroup container, int position, @NonNull Object object) {  container.removeView((ConstraintLayout) object);  } }  (myAdapter.java)  package com.example.bloodgenix;  import android.view.LayoutInflater; import android.view.View; import android.view.ViewGroup; import android.widget.TextView;  import androidx.annotation.NonNull; import androidx.recyclerview.widget.RecyclerView;  import com.bumptech.glide.Glide; import com.example.bloodgenix.Models.DonationDetails; import com.firebase.ui.database.FirebaseRecyclerAdapter; import com.firebase.ui.database.FirebaseRecyclerOptions;  import de.hdodenhof.circleimageview.CircleImageView;  public class myAdapter extends FirebaseRecyclerAdapter<DonationDetails, myAdapter.myviewholder> {   private OnItemClickListener mListener;  String SearchMobileNumber;   public interface OnItemClickListener {  String OnItemClick(int position, String number);  }   public void setOnItemClickListener(OnItemClickListener listener) {  mListener = listener;  }   public myAdapter(@NonNull FirebaseRecyclerOptions<DonationDetails> options) {  super(options);  }   @Override  protected void onBindViewHolder(@NonNull myviewholder holder, int position, @NonNull DonationDetails model) {  holder.personNameSearch.setText(model.personFullName);  holder.personLocationSearch.setText(model.donorLocation);  holder.bloodGroupVal.setText(model.blGroup);  holder.personNumberSearch.setText(model.phoneNumber);  Glide.*with*(holder.profileImageSearch.getContext()).load(model.personProfileImg).into(holder.profileImageSearch);  }   @NonNull  @Override  public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {   View view = LayoutInflater.*from*(parent.getContext()).inflate(R.layout.*list\_layout\_search\_page*, parent, false);  return new myviewholder(view, mListener);  }   public class myviewholder extends RecyclerView.ViewHolder {  CircleImageView profileImageSearch;  TextView personNameSearch, personLocationSearch, bloodGroupVal;  TextView personNumberSearch;   public myviewholder(@NonNull View itemView, OnItemClickListener listener) {  super(itemView);  profileImageSearch = itemView.findViewById(R.id.*profileImageSearch*);  personNameSearch = itemView.findViewById(R.id.*personNameSearch*);  personLocationSearch = itemView.findViewById(R.id.*personLocationSearch*);  bloodGroupVal = itemView.findViewById(R.id.*bloodGroupVal*);  personNumberSearch = itemView.findViewById(R.id.*personNumberSearch*);   itemView.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  if (listener != null) {  int position = getAdapterPosition();  if (position != RecyclerView.*NO\_POSITION*) {  listener.OnItemClick(position, personNumberSearch.getText().toString());  }  }  }  });  }  } }  Models  (Users.java)  package com.example.bloodgenix.Models;  public class Users {  String uid;  String profileImg, FullName, UserName ;  String EmailId, password,d\_o\_b, Gender;  String IdentityProof, MobileNumber;   public Users(){   }  public Users(String uid, String profileImg, String fullName, String userName, String emailId, String password, String d\_o\_b, String gender, String identityProof, String mobileNumber) {  this.uid = uid;  this.profileImg = profileImg;  FullName = fullName;  UserName = userName;  EmailId = emailId;  this.password = password;  this.d\_o\_b = d\_o\_b;  Gender = gender;  IdentityProof = identityProof;  MobileNumber = mobileNumber;  }   public String getUid() {  return uid;  }   public void setUid(String uid) {  this.uid = uid;  }   public String getProfileImg() {  return profileImg;  }   public void setProfileImg(String profileImg) {  this.profileImg = profileImg;  }   public String getFullName() {  return FullName;  }   public void setFullName(String fullName) {  FullName = fullName;  }   public String getUserName() {  return UserName;  }   public void setUserName(String userName) {  UserName = userName;  }   public String getEmailId() {  return EmailId;  }   public void setEmailId(String emailId) {  EmailId = emailId;  }   public String getPassword() {  return password;  }   public void setPassword(String password) {  this.password = password;  }   public String getD\_o\_b() {  return d\_o\_b;  }   public void setD\_o\_b(String d\_o\_b) {  this.d\_o\_b = d\_o\_b;  }   public String getGender() {  return Gender;  }   public void setGender(String gender) {  Gender = gender;  }   public String getIdentityProof() {  return IdentityProof;  }   public void setIdentityProof(String identityProof) {  IdentityProof = identityProof;  }   public String getMobileNumber() {  return MobileNumber;  }   public void setMobileNumber(String mobileNumber) {  MobileNumber = mobileNumber;  } }  (DonationDetails.java)  package com.example.bloodgenix.Models;  public class DonationDetails {   String uid;  public String phoneNumber;  public String blGroup;  String ruDiabetic, diabeticCount, otherDisease;  String ruDonated, donateMonth, weight;  public String donorLocation;  public String personFullName;  public String personProfileImg;   public DonationDetails(){   }   public DonationDetails(String uid, String phoneNumber, String blGroup, String ruDiabetic, String diabeticCount, String otherDisease, String ruDonated, String donateMonth, String weight, String donorLocation, String personFullName, String personProfileImg) {  this.uid = uid;  this.phoneNumber = phoneNumber;  this.blGroup = blGroup;  this.ruDiabetic = ruDiabetic;  this.diabeticCount = diabeticCount;  this.otherDisease = otherDisease;  this.ruDonated = ruDonated;  this.donateMonth = donateMonth;  this.weight = weight;  this.donorLocation = donorLocation;  this.personFullName = personFullName;  this.personProfileImg = personProfileImg;  }   public String getPersonProfileImg() {  return personProfileImg;  }   public void setPersonProfileImg(String personProfileImg) {  this.personProfileImg = personProfileImg;  }   public String getPersonFullName() {  return personFullName;  }   public void setPersonFullName(String personFullName) {  this.personFullName = personFullName;  }   public String getDonorLocation() {  return donorLocation;  }   public void setDonorLocation(String donorLocation) {  this.donorLocation = donorLocation;  }   public String getUid() {  return uid;  }   public void setUid(String uid) {  this.uid = uid;  }   public String getPhoneNumber() {  return phoneNumber;  }   public void setPhoneNumber(String phoneNumber) {  this.phoneNumber = phoneNumber;  }   public String getBlGroup() {  return blGroup;  }   public void setBlGroup(String blGroup) {  this.blGroup = blGroup;  }   public String getRuDiabetic() {  return ruDiabetic;  }   public void setRuDiabetic(String ruDiabetic) {  this.ruDiabetic = ruDiabetic;  }   public String getDiabeticCount() {  return diabeticCount;  }   public void setDiabeticCount(String diabeticCount) {  this.diabeticCount = diabeticCount;  }   public String getOtherDisease() {  return otherDisease;  }   public void setOtherDisease(String otherDisease) {  this.otherDisease = otherDisease;  }   public String getRuDonated() {  return ruDonated;  }   public void setRuDonated(String ruDonated) {  this.ruDonated = ruDonated;  }   public String getDonateMonth() {  return donateMonth;  }   public void setDonateMonth(String donateMonth) {  this.donateMonth = donateMonth;  }   public String getWeight() {  return weight;  }   public void setWeight(String weight) {  this.weight = weight;  } }  (RecipientDetails.java)  package com.example.bloodgenix.Models;  public class RecipientDetails {  String uid;  String bGroupRecipient;  String locationRecipient, otherRequirement;  String reasonBloodRequest,phoneNumber;  public String personFullName;  public String personProfile;   public String getPersonFullName() {  return personFullName;  }   public void setPersonFullName(String personFullName) {  this.personFullName = personFullName;  }   public String getPersonProfile() {  return personProfile;  }   public void setPersonProfile(String personProfile) {  this.personProfile = personProfile;  }   public RecipientDetails(){   }   public String getReasonBloodRequest() {  return reasonBloodRequest;  }   public void setReasonBloodRequest(String reasonBloodRequest) {  this.reasonBloodRequest = reasonBloodRequest;  }   public String getPhoneNumber() {  return phoneNumber;  }   public void setPhoneNumber(String phoneNumber) {  this.phoneNumber = phoneNumber;  }   public RecipientDetails(String uid, String bGroupRecipient, String locationRecipient, String otherRequirement, String reasonBloodRequest, String phoneNumber, String personFullName, String personProfile) {  this.uid = uid;  this.bGroupRecipient = bGroupRecipient;  this.locationRecipient = locationRecipient;  this.otherRequirement = otherRequirement;  this.reasonBloodRequest = reasonBloodRequest;  this.phoneNumber = phoneNumber;  this.personFullName = personFullName;  this.personProfile = personProfile;  }   public String getUid() {  return uid;  }   public void setUid(String uid) {  this.uid = uid;  }   public String getbGroupRecipient() {  return bGroupRecipient;  }   public void setbGroupRecipient(String bGroupRecipient) {  this.bGroupRecipient = bGroupRecipient;  }   public String getLocationRecipient() {  return locationRecipient;  }   public void setLocationRecipient(String locationRecipient) {  this.locationRecipient = locationRecipient;  }   public String getOtherRequirement() {  return otherRequirement;  }   public void setOtherRequirement(String otherRequirement) {  this.otherRequirement = otherRequirement;  } }  (SearchRecordDatabase.java)  package com.example.bloodgenix.Models;  public class SearchRecordDatabase {  String profileImg, personName;  String location, bloodGroupSearch;   public SearchRecordDatabase(){  }   public SearchRecordDatabase(String profileImg, String personName, String location, String bloodGroupSearch) {  this.profileImg = profileImg;  this.personName = personName;  this.location = location;  this.bloodGroupSearch = bloodGroupSearch;  }   public String getProfileImg() {  return profileImg;  }   public void setProfileImg(String profileImg) {  this.profileImg = profileImg;  }   public String getPersonName() {  return personName;  }   public void setPersonName(String personName) {  this.personName = personName;  }   public String getLocation() {  return location;  }   public void setLocation(String location) {  this.location = location;  }   public String getBloodGroupSearch() {  return bloodGroupSearch;  }   public void setBloodGroupSearch(String bloodGroupSearch) {  this.bloodGroupSearch = bloodGroupSearch;  } }  (PersonStatus.java)  package com.example.bloodgenix.Models;  public class PersonStatus {   String phoneNumber;  String status;   public PersonStatus(String phoneNumber, String status) {  this.phoneNumber = phoneNumber;  this.status = status;  }   public String getPhoneNumber() {  return phoneNumber;  }   public void setPhoneNumber(String phoneNumber) {  this.phoneNumber = phoneNumber;  }   public String getStatus() {  return status;  }   public void setStatus(String status) {  this.status = status;  } }  (BlockUser.java)  package com.example.bloodgenix.Models;  public class BlockUser {  String receiver;  String sender;   public BlockUser(){   }   public BlockUser(String receiver, String sender) {  this.receiver = receiver;  this.sender = sender;  }   public String getReceiver() {  return receiver;  }   public void setReceiver(String receiver) {  this.receiver = receiver;  }   public String getSender() {  return sender;  }   public void setSender(String sender) {  this.sender = sender;  } } |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |