1.Technology stack table:

| **Layer** | **Technology and Components** |
| --- | --- |
| **Mobile App** | Android Studio, Kotlin |
| **User Interface** | XML, Android XML layouts |
| **Architecture** | MVVM (Model-View-ViewModel) |
| **Networking** | Retrofit for RESTful API communication |
| **Database** | Room (for local database) |
| **Authentication** | Firebase Authentication |
| **Real-time Communication** | Firebase Realtime Database or Firestore |
| **Push Notifications** | Firebase Cloud Messaging (FCM) |
| **Dependency Injection** | Dagger 2 |
| **Asynchronous Programming** | Kotlin Coroutines |
| **Image Loading** | Glide or Picasso |
| **Logging** | Timber for logging |
| **Testing** | JUnit, Espresso (for UI testing) |
| **Version Control** | Git |

2.Characteristics Table:

| **Characteristic** | **Description** |
| --- | --- |
| **Application Type** | Chat Application |
| **Platform** | Android |
| **Programming Language** | Kotlin |
| **Development Environment** | Android Studio |
| **User Interface (UI)** | Intuitive and responsive UI with chat conversation threads, contact list, message input, and multimedia support |
| **Architecture Pattern** | MVVM (Model-View-ViewModel) |
| **Authentication** | Firebase Authentication |
| **Real-time Communication** | Firebase Realtime Database or Firestore |
| **Push Notifications** | Firebase Cloud Messaging (FCM) |
| **Database** | Room (Local database for caching and offline support) |
| **Networking** | Retrofit for RESTful API communication |
| **Dependency Injection** | Dagger 2 |
| **Asynchronous Programming** | Kotlin Coroutines |
| **Image Loading** | Glide or Picasso for loading and caching images |
| **Logging** | Timber for logging |
| **Testing** | JUnit for unit testing, Espresso for UI testing |
| **Security** | Secure data transmission (HTTPS), user data encryption |
| **Error Handling** | Graceful handling of errors, user-friendly error messages |
| **User Authentication** | Email/password authentication, social media login |
| **Multimedia Support** | Image, video, and file sharing capabilities |
| **Offline Support** | Caching, local database for offline access |
| **Accessibility** | Consideration for accessibility features (e.g., screen readers) |
| **Internationalization (i18n)** | Support for multiple languages and locales |
| **Scalability** | Design considerations for potential future growth and increased user base |
| **Analytics** | Integration with analytics tools for user behavior tracking |
| **Version Control** | Git |