
Madhur Ahlawat

madhur.ahlawat17@gmail.com

9958417372

Noida, Uttar Pradesh

Summary

Being a hardworking and highly proficient Android application developer with a real passion for mobile application development, I have a great knowledge of the application life cycle right from modeling to develop. I also have incredible attention to detail and perseverance in getting the job done well.

Education

Bachelor of Technology, Computer Science & Engineering

INVERTIS UNIVERSITY • Bareilly, Uttar Pradesh

10/2016

Experience

Junior Android Developer

Clixlogix Technologies • Noida, Uttar Pradesh

11/2019 - 08/2020

At Clixlogix Technologies, I started my career in Android Application Development with Java as the programming language. I worked on four projects individually. I made myself capable of making attractive UIs. I learned about various Android components like foreground and background services, Activity and fragment life-cycle, maintaining the state of the app, using worker threads for fluent user experience. I made myself good at Room and LiveData and Databinding and worked on MVVM and RxJava based projects.

Android Developer

Antino Labs • Noida, Uttar Pradesh

07/2020 - Present

At Antino Labs, I worked for their Mumbai-based client "Multiliving Technologies". I worked on the real estate-based Android project.

I worked on Two Android applications build for tenants and the staff. We can assume the whole project as a huge hotel management system. As I built separate apps for tenants and the staff. It includes making the communication between both parties easy and hassle-free. A tenant can request the services, He or she can order groceries, rate the services, and can chat with the staff.

Skills

- Java SE
 - Kotlin
 - RxJava
 - Root Cause Analysis
 - SOAP
 - Design Patterns
 - UI Design
 - JUnit 5
 - Espresso
 - MVVM
 - Android SDK
 - MySQL
 - Test-driven development
 - OOP
 - Multithreading
 - Coroutines
 - Dagger-Hilt
-

Languages

- English
 - Hindi
-

Apps made live on PlayStore

- <https://play.google.com/store/apps/details?id=com.photostat.app&hl=en>
- <https://play.google.com/store/apps/details?id=com.funngage.funngage>
- <https://play.google.com/store/apps/details?id=com.usa.mlc>
- <https://play.google.com/store/apps/details?id=musicPlug.com>
- <https://play.google.com/store/apps/details?id=com.cap.main&hl=en>
- <https://play.google.com/store/apps/details?id=com.impossibletraining.impossibletraining&hl=en>
- <https://play.google.com/store/apps/details?id=com.cap.main>
- <https://play.google.com/store/apps/details?id=com.doma.consumer>
- <https://play.google.com/store/apps/details?id=com.doma.owner>
- <https://play.google.com/store/apps/details?id=com.app.lal10>

Nonlive apps:

<https://drive.google.com/drive/folders/1gbSrofE8hR10jefmuqKtmrDkFOiylHV7?usp=sharing>

Concepts covered

Java SE

- OOP concepts
- Exception handling
- Multithreaded programming
- Generics
- Collections framework
-

Database

- DDL & DML queries
- Joins
- Views
- Subqueries
- Triggers
- Stored procedures

Android application development

UI:

- LinearLayout, RelativeLayout, ConstraintLayout
- RecyclerView
- GridLayout
- NavigationDrawer
- AppBarLayout
- TabLayout
- FrameLayout
- Coordinator Layout
- BottomSheetDialog
- Jetpack Compose
- Notifications (Heads up, Lock screen, App icon badge and Expandable notifications)

Services:

- Background Service
- Foreground Service
- Intent Service

Broadcasts and broadcast receivers:

- System generated broadcast intents
- User-generated broadcast intents
- Creating and registering broadcast receivers

Activity:

- Activity launch mode
- Excluding an activity from back-stack history
- Activity lifecycle
- Navigation graph (Jetpack compose)
- Deep Linking

Fragments:

- Passing data b/w activity and fragment through Interface and ViewModel class
- Fragment lifecycle
- Sending events back to activity with the callback interface

SharedPreferences:

- Reading, writing & listening data changes in SharedPreferences

Job Scheduling:

- JobScheduler
- WorkManager

Networking:

- Retrofit
- WebSockets (Made a chatroom app based on WebSockets)

Database:

- SQLite, Room with LiveData

Firebase:

- Firebase realtime database
- FCM

APIs:

- Facebook, LinkedIn, Twitter, Google Maps & Google APIs, VOIP calling APIs like Sinch, Twilio SDK
- FFmpeg

Architecture:

- MVVM

Tools

- Windows, Linux & Mac
- Android Studio

- Git
- Postman