1️⃣ Mobile App Developer (Flutter + Native Android)

💼 **Responsibilities:**

- Develop the **Flutter-based MDM Agent** app.
- Implement **MethodChannels** to integrate **Kotlin/Java for deep system access**.
- Use **Firebase Cloud Messaging (FCM)** for remote commands.
- Handle **app installation/uninstallation, screen mirroring, and device control**.
- Ensure the app runs in the **background (Foreground Service in Kotlin)**.

🛠️ **Skills Required:**

✅ **Flutter & Dart** (for UI and remote command handling)
✅ **Kotlin/Java (Android Native)** (for MDM features)
✅ **Android APIs:**

- **DevicePolicyManager** (for device control: lock, wipe, reset)
- **MediaProjection API** (for screen sharing)
- **AccessibilityService API** (for remote interaction)
- **PackageManager API** (for app install/uninstall)
  - ✅ **Firebase (FCM, Firestore, Authentication)**
  - ✅ **WebRTC (for live screen streaming - optional)**
  - ✅ **Background Services (Foreground Service in Kotlin)**

📌 **Key Deliverables:**

- **Flutter MDM Agent App** (Communicates with backend via Firebase/WebSockets)
- **MethodChannel Integration** for **Kotlin-based MDM features**
- **Device Monitoring & Control** (Remote app management, lock, wipe)
- **Screen Sharing via WebRTC (optional)**
- **Push Notification Handling for Remote Commands**

📁 **Example Flutter Project Structure:**

```
/mdm_agent_app
├── lib/
│   ├── main.dart
│   ├── services/
│   │   ├── firebase_service.dart  # Handles push commands
│   │   ├── native_mdm_service.dart  # Calls native Kotlin code
│   ├── screens/
│   │   ├── dashboard.dart
│   │   ├── device_info.dart
│   │   ├── screen_share.dart
├── android/
│   ├── MainActivity.kt  # Handles MethodChannel for native features
│   ├── services/
│   │   ├── MdmService.kt  # Lock/Wipe/Install Apps
```

```
|   |   ├── ScreenCaptureService.kt  # Live screen sharing
```

📌 How They Work Together

1. **Mobile App Developer** builds the **Flutter MDM Agent**, which:

   - Registers with the **backend** via API.
   - Receives commands via **Firebase Cloud Messaging (FCM)** or WebSockets.
   - Sends **device status updates** to the backend.
2. **Web Developer** builds the **Admin Dashboard & Backend**, which:

   - Stores **device info** in a database.
   - Allows **sending remote commands** (install apps, lock phone, wipe data).
   - Displays **real-time device data** on a web UI.