Mugo Plumbing Solutions – Mobile App Architecture

# 1. App Stack Overview

| Layer | Tech Stack |

|--------------------|-----------------------------------------------|

| Frontend (App) | React Native (Expo) + Tailwind-RN |

| Backend/API | FastAPI or Node.js + Express |

| Database | MongoDB Atlas |

| Auth | Firebase Auth or JWT |

| File Storage | Cloudinary or Firebase Storage |

| Realtime Updates | Socket.IO or Firebase Cloud Messaging |

| Push Notifications | Expo Push Notifications or FCM |

| Payments | Paynow, Flutterwave, or Stripe |

| Deployment | Expo EAS Build, Play Store, App Store |

# 2. App Features by Role

Client Features:

- Book a plumber

- View booking status

- In-app chat with plumber

- Leave ratings and reviews

- View invoices / pay online

Plumber Features:

- Receive & accept jobs

- Mark jobs as complete

- Upload job photos or invoices

- View earning history

Admin (Web/Admin App):

- Approve plumber applications

- Track active jobs

- Review earnings & analytics

# 3. Authentication & Authorization

- Firebase Auth or JWT-based login

- Roles: client, plumber, admin

- Role-based access control and UI rendering

# 4. Backend API Endpoints

| Endpoint | Function |

|-------------------------|-----------------------------------|

| POST /auth/signup | Register client/plumber |

| POST /auth/login | Login with email/password |

| GET /plumbers | List available plumbers |

| POST /bookings | Client books a service |

| GET /bookings/:id | Booking status |

| POST /plumber/complete | Mark job complete (plumber) |

| POST /upload | Upload job images or documents |

# 5. Messaging Options

- In-app chat using Socket.IO or Firebase Realtime DB

- WhatsApp notifications via Twilio (optional)

# 6. Notifications

- Use Expo Push Notifications or Firebase Cloud Messaging (FCM)

- Notify plumbers, clients, and admins of job events

# 7. Payments

- Integrate Paynow, Flutterwave, or Stripe

- Start with cash-on-delivery if needed

# 8. Developer Tools

| Purpose | Tool |

|------------------|-----------------------------|

| State Management | Zustand, Redux, or Context |

| Navigation | React Navigation |

| Forms | React Hook Form or Formik |

| Validation | Yup |

| API Calls | Axios or Fetch |

# 9. App Deployment

- Build using Expo EAS

- Deploy to Play Store and App Store

- Monitor using Sentry or Firebase Crashlytics

# 10. Optional Enhancements

| Feature | Tool/Service |

|------------------------|--------------------------|

| Maps & location | Google Maps API |

| Plumber verification | File uploads + Admin |

| Analytics | Firebase / PostHog |

| App walkthrough | Onboarding screens |

# Architecture Diagram (text format)

[React Native App (Expo)]

|

HTTPS

|

[FastAPI / Express Backend]

|

REST API / WebSocket

|

[MongoDB Atlas] [Cloudinary] [Twilio/FCM]

| | |

User Data Media Uploads Messaging & Push