Merchant Complaint Management Application

---

1. Overview

Purpose
To provide merchants a simple and reliable application to log, track and resolve complaints or service requests related to devices, payments, advertisements, or any other issues.

Scope

- Mobile App (Android & iOS) for merchants
- Web-based Admin Panel for support team
- RESTful API backend

---

2. Stakeholders

Role	Responsibilities
Merchant	Register/login, raise complaints, upload evidence, track status
Support Agent	View, assign, update complaints, resolve issues
Admin	Manage categories, SLAs, users, and reports

---

3. Functional Requirements

3.1 Merchant App

Authentication & Profile

- Login via mobile number + OTP.
- **First-time login:** Merchant must provide mandatory details:
    - Merchant Name
    - Merchant ID (if available)
    - Contact Number
    - Business Name / Address (optional)
- After submission, details saved on backend and used to pre-fill forms for future logins.

- Profile view & edit feature to update these details later.

Complaint Submission

- Form fields:
    - Merchant Name (pre-filled from profile)
    - Merchant ID (pre-filled from profile)
    - Contact Number (pre-filled from profile)
    - Select Device/Order ID (dropdown or QR scan)
    - Complaint Category (dropdown: Device Issue, Payment Issue, Ad Management Issue, Billing, Other)
    - Short Description (text area)
    - Attach Photos/Videos (multiple files)
    - Priority Level (Normal/Urgent)
- Submit button generates Ticket ID and confirmation message.

Complaint Tracking

- “My Complaints” screen listing:
    - Ticket ID
    - Date Raised
    - Category
    - Status (Open/In-Progress/Resolved/Closed)
    - SLA countdown indicator
- Detail view with history, updates, and chat with support.

Notifications

- SMS and Push Notifications for:
    - Ticket creation
    - Status change
    - Resolution
- In-app notification center.

Feedback

- Prompt for rating (1–5) and comments after resolution.

3.2 Admin / Support Panel

Login

- Role-based authentication.

Ticket Management

- Dashboard:
    - Open Tickets
    - In-progress Tickets
    - SLA Breaches
    - Filter by category, status, priority
- Ticket Detail View:
    - Merchant details
    - Issue details
    - Attachments
    - Status update (Open/In-progress/Resolved/Closed)
    - Add internal notes
    - Assign to agent

Configuration Management

- Manage Complaint Categories
- Manage Priority Levels
- Manage SLAs (response/resolution time per category)
- Manage Merchant Profiles

Reports & Analytics

- Ticket Volume by category
- SLA compliance reports
- Agent performance
- Feedback scores

3.3 Backend / API

- REST/GraphQL API for mobile app and admin panel.
- JWT-based authentication.
- Data model:
    - Merchants (created on first login)
    - Devices/Orders
    - Complaints
    - Status Updates
    - Attachments
- Integration with SMS/Email gateway for notifications.

---

4. Non-Functional Requirements

Area	Requirement
Security	Data encryption in transit (HTTPS); Role-based access
Performance	<2s response time for main API calls
Scalability	Handle 10,000+ merchants simultaneously
Availability	99.5% uptime
Data Storage	Cloud-based database with daily backup
Audit Logs	All ticket updates logged

---

5. User Interface Guidelines

- Clean, minimalistic UI.
- Mobile-first design.
- Dropdowns for categories to avoid typing errors.
- Large “Submit Complaint” button.
- Use icons for attachment upload and status.

---

6. Assumptions

- No existing master database of merchants; details collected at first login and stored on backend.
- Merchant profile info validated via OTP to ensure authenticity.
- SMS/Email service provider available for notifications.
- Support team already has defined SLAs.

---

7. Future Enhancements

- WhatsApp integration for ticket updates.
- AI-based issue categorization.
- Multi-language support.
- Offline complaint submission with later sync.

This shows:

- Merchant uses the **mobile app** to log in and raise complaints.
- App talks to the **Backend/API Server** for authentication, profile saving, ticket creation.
- Backend notifies **Support Team/Admin Panel** with the new ticket.
- Support updates status → backend → app → merchant sees real-time updates.