# **📄 1. Bug Reporting Template (In-App)**

## **🔧 Bug Report Form Fields**

| **Field** | **Description** |
| --- | --- |
| **Bug Title** | Short, descriptive title of the issue |
| **Description** | Detailed explanation of the bug |
| **Steps to Reproduce** | List of steps to replicate the issue |
| **Expected Behavior** | What the user expected to happen |
| **Actual Behavior** | What actually happened |
| **Severity** | Low / Medium / High / Critical |
| **Device Info** | Auto-filled or selected (Device model, OS version) |
| **Screenshot** | Optional file upload (image or screen recording) |
| **Timestamp** | Auto-generated |
| **User ID (if logged in)** | Auto-attached securely |

## **📨 Submission Options**

* **Backend Endpoint**: POST to /api/report-bug
* **Third-Party Tools** (optional): Sentry Issue Tracker, Firebase Crashlytics Feedback SDK
* **Fallback**: support@defeah.app

## **🗂 Backend Metadata Format**

{

"bug\_id": "auto-uuid",

"title": "App crashes on Study Guide screen",

"description": "Happens every time I open the Study Guide after saving a note.",

"severity": "High",

"device": "Pixel 6, Android 13",

"steps": ["Open app", "Navigate to Study Guide", "Tap on saved guide"],

"expected": "The guide opens normally.",

"actual": "App crashes.",

"user\_id": "uid\_923X",

"timestamp": "2025-07-04T14:30:45Z"

}

# **📄 2. Error Logging Plan**

## **🔍 What to Log**

### **✅ LLM-Related**

* Prompt failure
* Timeout or response delay
* Flagged content by moderation filter
* Retry attempts + model version used

### **✅ Backend/API**

* HTTP errors (4xx, 5xx)
* DB failures (reads/writes/timeouts)
* Auth issues (expired tokens, permission errors)

### **✅ Frontend**

* App crashes
* Unhandled Flutter exceptions
* Firebase sync failures

### **✅ In-App Events**

* Failed study guide generation
* Failed daily verse fetch
* Sharing attempt errors
* Bug report form submission errors

## **🧾 Log Format**

| **Field** | **Description** |
| --- | --- |
| **event\_type** | e.g., llm\_generation\_error, db\_write\_failure |
| **user\_id** | if available |
| **timestamp** | ISO timestamp |
| **severity** | INFO, WARN, ERROR, CRITICAL |
| **snapshot** | Small JSON payload of relevant data (prompt, error code, etc.) |

## **☁️ Storage Options**

* **Firebase Crashlytics**: Frontend crash logging + user traces
* **Supabase Logs**: Server-side error capture
* **Sentry**: Consolidated event logging (LLM + API + mobile)

## **🔐 Security**

* Strip or hash PII (e.g., email, full names)
* Access logs only via RBAC-protected dashboard
* Log rotation & retention policies (30–90 days)

# **📄 3. Release Notes Template**

### **📦 Format**

\*\*Version:\*\* vX.X.X

\*\*Release Date:\*\* YYYY-MM-DD

### ✨ New Features

- Feature A

- Feature B

### 🔧 Improvements

- UI enhancements

- Faster API response for study guides

### 🐞 Bug Fixes

- Fixed crash on Note screen

- Corrected verse rendering issue on dark mode

### ⚠️ Known Issues

- Hindi reflection text wraps poorly on small screens

### 🛠 Developer Notes

- Switched backend to Supabase Edge for better latency

### **✅ Example: v1.0 Release Notes (Aug 14, 2025)**

\*\*Version:\*\* v1.0.0

\*\*Release Date:\*\* 2025-08-14

### ✨ New Features

- AI-powered Bible study guide generator

- Topic-based or Scripture-based queries

- Saved notes & favorites system

- Basic authentication and onboarding

### 🔧 Improvements

- Added loading state animations

- Optimized guide caching logic

### 🐞 Bug Fixes

- Crash on first install fixed

- Timeout issue on large Bible topics resolved

### ⚠️ Known Issues

- Daily verse not yet localized (coming in v1.2)

### 🛠 Developer Notes

- Prompt templates tested with ESV and ASV flows

# **📄 4. User Feedback Log Format**

## **🧾 Schema**

| **Field** | **Description** |
| --- | --- |
| **feedback\_type** | Feature Request / Bug / Praise / Complaint |
| **message** | Full feedback content |
| **screen** | Context (e.g., "StudyGuide", "DailyVerse") |
| **user\_id** | optional, if logged in |
| **timestamp** | ISO format |
| **priority** | Low / Medium / High |
| **source** | In-App / Email / WhatsApp / Review Site |

## **🏷 Tagging & Grouping**

* Auto-tag by:  
  + Feature (Daily Verse, Study Guide, Notes, etc.)
  + Language (EN/HI/ML)
  + Sentiment (Positive/Negative/Neutral via NLP)

## **🎯 Prioritization Strategy**

* Critical = Bug affecting >10% of users or involving crash
* High = Feature request with >3 upvotes / repeated support requests
* Medium = Minor friction point
* Low = Cosmetic, non-blocking issues

## **🛠 Tools**

* Feedback can be stored in Firestore
* Use Airtable or Notion as triage boards
* Tag feedback for product planning sprints