## **Video Game Stats App - MVP Plan**

### **1. Overview**

A secure Android application that allows users to select video games, link in-game usernames, and view:

* Individual game statistics
* Aggregated stats by genre (FPS, MOBA, etc.)
* Overall player performance across all added games

This MVP focuses on core functionality, secure data handling, and a clear development roadmap.

### **2. Core Features**

1. **Game Selection & Search**: Browse or search a list of supported games, grouped by genre.
2. **Username Management**: Add and manage usernames for each selected game.
3. **Stats Retrieval**: Fetch and display statistics for each game via external APIs.
4. **Genre Aggregation**: Calculate and show aggregated stats per genre based on added games.
5. **Global Aggregation**: Present overall performance metrics across all games.

### **3. Technical Stack**

* **Language & Framework**: Kotlin +
* **Architecture**: MVVM (?)
* **Networking**:
* **Local Storage**: Room for cached stats
* **API Integrations**:
  + RAWG Video Games Database API for game catalogue and genres
  + Game-specific APIs (e.g., Overwatch, LoL) via modular repository pattern
* **Dependency Injection**: Hilt

### **4. Security & Threat Modeling (Remove?)**

* **Threat Model (STRIDE)**: Identify Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege
* **Data at Rest**: Encrypt sensitive data using AndroidX Security library
* **Transport Security**: Enforce HTTPS endpoints and pin certificates
* **Input Validation**: Sanitize and validate all user inputs before network calls
* **Authentication**: Local authentication via BiometricPrompt or PIN
* **Code Quality & Review**: Use Detekt for static analysis, OWASP Dependency-Check for libraries

### **5. UI Flow (Screens)**

1. **Home**: Genre dashboard with counts of tracked games
2. **Game List**: Titles per genre; option to add username
3. **Stats Detail**: Individual game stats view
4. **Genre Stats**: Aggregated view per genre
5. **Global Stats / Profile**: Overall performance metrics

### **6. Project Setup & CI/CD (Remove?)**

* **Version Control**: GitHub repo with feature-branch workflow
* **CI Pipeline**: GitHub Actions for:
  + Kotlin linting (ktlint), static analysis (Detekt)
  + Unit tests (JUnit)
  + Dependency scanning
* **Release**:

### **7. Deliverables (Remove?)**

* Secure SDLC documentation (plan, threat model)
* Working Android APK (MVP)
* Code repository with CI/CD pipeline
* Code review report and OWASP checklist
* Project presentation slides