1. Requirement Snapshot

MVP mobile app with a **swipe-to-play feed** of 25 mini-games.
Core: **Scroll View → Play Game → Back → Coin system → Ads + IAP**.
Game stats (likes, plays, winners) are **faked & auto-growing** in MVP.

---

1. App Framework

Screens

- **Scroll View**
    - Swipe vertically (TikTok-style) to browse games.
    - Each card shows:
        - Game preview (shrunken 75% view, tinted yellow).
        - Play Now button.
        - Coin entry: *Play with X coins → Win Y coins*.
        - Stats (Likes, Plays, Players, Winners, Coins Spent).
    - Scrolling should be a smooth, full-page snap.
- **Game View**
    - Loads selected games (full screen).
    - Controls: swipe, tap, drag (per game).
    - Top-right: Like button.
    - Top-left: Back button (returns to Scroll View).

    - Game ends → results → update coin balance.

- **Coin Shop**
    - HUD: coin balance always visible at top.
    - Shop options:
        - Remove Ads (one-time purchase).
        - Buy 100 coins (default, configurable).

---

2. Games

- Pick 25 from PlayTok game list (HTML5).
- Each game lives in its own folder (server).
- Served dynamically via config JSON (URL-based).
- For MVP → **no multiplayer, no real backend leaderboards**.

---

3. Game Stats (MVP → Fake Data)

- Stats shown: Likes, Plays, Players, Winners, Coins Spent.
- Likes: stored client-side, persists locally.
- Others: faked using base values + growth function.

Growth Formula:

```
current = base * (1 + rate) ^ hoursSinceLaunch
```

- Rate default: 10%/hour (configurable).
- Launch timestamp stored in config.

---

4. Coin Economy

- Configurable per game:
    - Coins to play (X).
    - Coins rewarded if win (Y).
- Survival-only games: always deduct X coins, no Y reward.
- If user has <X coins → block Play button, show Coin Shop CTA.

---

5. Ads

- Integrate **AdMob** (banners + interstitials; rewarded optional).
- Configurable:
    - Show ad after N scrolls.
    - Show ad after M games played.
- Remove ads if user purchases IAP.

---

6. Backend

- Games hosted on server, loaded by URL.

Configuration file (JSON or CSV) served from backend:
GameID

  - GameURL
  - CoinsToPlay
  - Winnable? (true/false)
  - CoinsRewarded
  - BaseStats (plays, likes, players, winners, coinsSpent)
  - GrowthRate
  - AdFrequency (scrolls, plays)
  - InitialCoins
  - IAP_CoinPack (default 100)

7. Other App Aspects

- **Firebase Notifications integrated for push.**
- **Basic error handling:**
  - **If the game URL fails, show a fallback screen with a retry button.**
  - **If no coins are left, auto-redirect to Coin Shop.**

---

8. MVP Deliverables

- Working Android app (Scroll View + Game View + Coin Shop).
- 25 HTML5 mini-games integrated via config.
- Fake stats with growth logic.
- AdMob integrated with config-driven triggers.

- Firebase push notifications.
- IAP (remove ads + coin packs).

2. Detailed explanations for required headlines

1. **App framework**
    a. The “scroll view” and “game view” are 2 main screens we need.
    b. Scroll view:
        i. When player opens the App, one of the games shown on the screen randomly with “Play Now” button, and player can swipe or click to play
        ii. When player clicks the Play button, he will then be moving to game view
        iii. From the game view, the player can come back by pressing “back button” on the screen, or by pressing device back key.
        iv. In the scroll view, the user only will swipe up or down.
        v. When scrolled, the effect has to be a perfect scroll view like we see in the TikTok or instagram videos.
        vi. Just beneath the Play button 2 data will be displayed
            1. Play with: X coins, Win Y coins
        vii. Every Game will have following **game stats** displayed on the scroll view
            1. Likes
            2. Plays
            3. Players
            4. No of Winners
            5. Coins spent
        viii. Game Stats:
            1. We need to store the likes data from the players from the game view and display here.
            2. Rest of the data has to be stored in the back end and displayed here respectively
            3. IMPORTANT FOR MVP:
                a. We need to fake the data in the MVP as we will not have enough ratings to show for the game.
                b. We need data for every stat above and that data grows every hour by a small %
                c. This needs to be part of the configurations system
    c. Game view:
        i. After pressing the play button in the scroll view, the player comes to the specific game view.

ii. Here in the game view swipe control is considered the game control to play the game (Example fruit ninja).
iii. The scroll view of the game is nothing, only the game shirinked to 75%, has a yellow tint, shows the game name, and has the Play Now button on it.
iv. Player will be able to like the game on the game screen on right top of the game screen
v. The player will go back to the scroll view from the left top button.

d. Other app aspects:

i. Coins Shop:
   1. We need to display the coins balance on the top HUD.
   2. IAP from google play to do following:
      a. Remove ads
      b. Buy 100 coins

2. **Games list**: (pick any 25 games from this list) PlayTok Games list

3. Backend implementation

a. We need all games hosted on the server and are served to players on demand.
b. We need server configuration for "Configuration System" which can be applied in the live games
c. We need to implement each game as an individual folder and Admin can control any new games or deleting old games or updating games code.

4. Coin Economy

a. The coin data must be read from Configuration system
b. Each game requires the player to spend x coins. In some games where there is a winner, they win y coins. For games without any win condition like survival game, they dont give any coins

5. Ads

a. Bring the trigger data from the Configuration system for when to play Ads.
b. We need to integrate **AdMob** in the system for mobile Ads

6. **Android App"**
    a. In addition to above scope, please integrate Fire base notifications and analytics.

7. **Configuration System:**
    a. A JSON or a CSV format where all games playable URLs will be feeded by Admin. Data structure is like below
        i. S No
        ii. Game URL
        iii. Coins to Play
        iv. Winnable?
            1. If Yes, How many coins
    b. A Config for How many scrolls of games do we show a single Admob video Ad (think of Instagram as an example)
    c. How many initial coins for users on installation
    d. For IAP - whether we give 100 coins or different numbers.

Firebase Analytics

Event: Played_game:
Game_ID, Game_Name, Coins used, coins won, Time PLayed, Score gained, swipes before playing

Event:Coins bought
How many coins bought, user ID