

Mobile Application Agent Analysis

Game Economy & Payment Integration Strategy

1. Project Overview

This document outlines the logical flow and technical steps required to implement an automated payment and progression agent for a Car Racing Mobile Application.

Current Case Study:

- **Starting Capital:** \$100
- **Action 1:** Account creation (\$50 deducted). Remaining: \$50.
- **Action 2:** Car purchase (\$50 deducted). Remaining: \$0.
- **Problem:** User needs to enter the next level but has insufficient funds.

2. Step-by-Step Implementation Analysis

Step 1: Resource Monitoring Agent

The agent must constantly track the user's "Wallet Balance" variable. When the balance hits \$0, the agent disables the "Start Next Level" button and changes its state to "Locked/Requires Payment".

Step 2: Triggering the Payment Gateway

Once the user attempts to click the locked level, the agent triggers a 'Purchase' pop-up. It will offer various top-up packages (e.g., \$10, \$50, \$100) to be processed via an online credit/debit card.

Step 3: Secure Online Transaction

The agent facilitates a secure handshake with a Payment Service Provider (PSP). The user enters their card details. The agent ensures that the transaction is encrypted and verified through 3D Secure (OTP) authentication.

Step 4: Balance Synchronization

Upon receiving a "Success" signal from the bank, the agent instantly updates the local and server-side databases. The user's balance is incremented by the purchased amount.

Step 5: Level Unlock Logic

With the new balance available, the agent runs a final check. If the balance is > \$0 (or the required entry fee), the "Next Level" becomes accessible, and the user is permitted to enter the gameplay loop.

3. Technical Recommendations

- **Security:** Use API keys and SSL encryption for all financial communications.
- **User Feedback:** Show a clear "Processing..." state during the card payment to prevent double-clicking.
- **Database:** Ensure the wallet balance is stored on a secure server to prevent hacking of local game files.

