**Software Requirements Specification (SRS)**

**Project Name:** Pi Game – Phase 2 Expansion  
**Version:** 1.0  
**Prepared by:** IT Business Analyst Team

### **Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Changes** | **Version** |
| IT BA Team | 18/04/2025 | Initial Draft | 1.0 |

## 1. Introduction

### **1.1 Purpose**

This document defines the software requirements for the Ton Tycoon Phase 2 expansion. It focuses on new modules for in-app purchases using Telegram Stars and wallet-based interaction via the TON blockchain.

### **1.2 Document Conventions**

* Functional requirements: labeled as **FR-XXX**
* Non-functional requirements: labeled as **NFR-XXX**
* Priorities follow the 1–5 scale (1: Critical, 5: Future)

### **1.3 Project Scope**

This SRS covers:

* In-App Purchase module (Ticket Shop)
* Wallet Integration (TON Check-in)
* Transaction history logging
* Error handling and notifications

### **1.4 References**

* BRD: Ton Tycoon Phase 2 (v1.0)
* Telegram Payment API Docs
* TON Wallet API Guide
* FRS Document

## 2. Overall Description

### **2.1 Product Perspective**

This expansion is a follow-up phase of the Ton Tycoon game. It integrates with Telegram’s in-app purchase platform and external TON blockchain for wallet connectivity. It also introduces backend logging mechanisms and interactive UI components for purchase and wallet activities.

### **2.2 Product Features**

* Display and sell in-game items using Telegram Stars
* Connect and verify TON wallet
* Perform onchain check-in actions
* Show transaction history
* Notify users on transaction result or system error

### **2.3 User Classes and Characteristics**

|  |  |
| --- | --- |
| User Class | Characteristics |
| Player | Casual or crypto-aware users; requires intuitive UI |
| Admin | Oversees config and reports; needs data accuracy |
| QA/Developer | Requires traceability and testability of all flows |

### **2.4 Operating Environment**

* Telegram App on mobile (Android/iOS)
* WebView-based UI on mobile
* Supported APIs: Telegram Stars API, TON Wallet API
* Hosting: Cloud-based, HTTPS protocol enforced

### **2.5 Assumptions and Dependencies**

* Players use up-to-date mobile apps
* TON Wallets are created and managed via Tonkeeper or compatible services
* Payment processing relies on Telegram infrastructure

## 3. Functional Requirements

### **3.1 Ticket Shop Module**

**Description and Priority:** Critical; enables core monetization.

#### Functional Requirements

* **FR-001**: Display dynamic package listing in Ticket Shop.
* **FR-002**: Initiate Telegram Star payment upon item selection.
* **FR-003**: Confirm payment and update user’s ticket balance.
* **FR-004**: Record purchase transactions (package, amount, timestamp).
* **FR-005**: Notify user of transaction outcome (success or failure).

### **3.2 Wallet Integration Module**

**Description and Priority:** Critical for onchain features and future ecosystem expansion.

#### Functional Requirements

* **FR-006**: Prompt and guide users to connect their TON Wallet.
* **FR-007**: Retrieve and display wallet address and balance.
* **FR-008**: Check wallet balance eligibility before check-in.
* **FR-009**: Execute daily check-in as an onchain transaction.
* **FR-010**: Display error messages on API failure or disconnect.

## 4. Nonfunctional Requirements

### **4.1 Performance Requirements**

* **NFR-001**: Wallet connection response time under 3 seconds
* **NFR-002**: Payment confirmation response under 2 seconds

### **4.2 Safety Requirements**

* Data validation for all user actions before processing
* Confirmation dialogs before committing transactions

### **4.3 Security Requirements**

* **NFR-003**: HTTPS for all external API communication
* **NFR-004**: Token-based authentication for wallet actions

### **4.4 Software Quality Attributes**

* **Usability**: Mobile-friendly design and user flow
* **Availability**: 99.9% uptime target
* **Reliability**: Redundant fallback in case of third-party service failure
* **Maintainability**: Modular design for easy future extension

## 5. Other Requirements

* Transactions must be exportable for auditing
* Multilingual UI support (English & Vietnamese)
* System time must be synced with NTP server for transaction accuracy

## Appendix A: Glossary

* **Telegram Stars**: Telegram’s in-app currency used for purchasing digital goods
* **TON Wallet**: Blockchain wallet operating on The Open Network
* **Check-in**: Daily participation action recorded on blockchain

## Appendix B: Analysis Models

(To be provided by UX and architecture team)

## Appendix C: Issues List

|  |  |  |
| --- | --- | --- |
| ID | Description | Status |
| TBD-001 | Define wallet error handling UI | Open |
| TBD-002 | Confirm translation strings with localization team | Open |