# 

**B. TECH (CSE) - V SEM**

**UE23CS341A - Software Engineering**

**PROJECT DOCUMENTATION ON**

**Software Test Plan**

**SEC: C**

**Team Members:**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **SRN** | **Name** |
| 1 | PES1UG23CS154 | C Kaustubh |
| 2 | PES1UG23CS166 | Chinmay Shivanand Muragod |
| 3 | PES1UG23CS167 | Chirag K M |
| 4 | PES1UG23CS173 | Darshith M S |

**AUGUST – DECEMBER 2025**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**RR CAMPUS (MAIN CAMPUS)**

**BENGALURU – 560085, KARNATAKA**

**Software Test Plan (STP) - ATM System (Sample)**

**Project:** Digital Asset and Cryptocurrency Portfolio Tracker  
**Version:** 1.0  
**Authors:** C Kaustubh, Chinmay Shivanand Muragod, Chirag K M, Darshith M S  
**Date:** 10-09-2025  
**Status:**  Draft

**1. Introduction**

**Purpose:** This document defines the test plan for the Digital Asset and Cryptocurrency Portfolio Tracker System v1.0. It outlines objectives, scope, strategy, resources, schedule, and responsibilities for testing.

**Scope:** Testing covers crypto portfolio management features such as user authentication with 2FA, portfolio creation and management, real-time market data integration, transaction tracking, performance analytics, exchange API integration, NFT management, tax reporting, and mobile applications. Core blockchain protocols and third-party exchange internal systems are excluded.

**References:** Digital Asset Portfolio Tracker SRS v1.0, Design Specifications v1.0, Cryptocurrency Exchange API Documentation, GDPR Compliance Guidelines, PCI-DSS Security Standards.  
  
**Definitions:** API (Application Programming Interface), DeFi (Decentralized Finance), NFT (Non-Fungible Token), 2FA (Two-Factor Authentication), KYC (Know Your Customer), GDPR (General Data Protection Regulation), P&L (Profit and Loss), OHLCV (Open, High, Low, Close, Volume), SRS (Software Requirements Specification), RTM (Requirements Traceability Matrix).

**2. Test Items**

* User Authentication and Security module
* Portfolio Management module
* Real-time Market Data Integration module
* Performance Analytics module
* Alerts and Notifications module
* Exchange Integration module
* Advanced Analytics module
* NFT Management module
* Tax Reporting and Compliance module
* Data Management and Backup module
* Mobile Applications (iOS/Android)
* Web Dashboard Interface
* React Frontend Application Components
* Rust Backend Services and API Layer
* API Gateway and Integration Services

**3. Features to be Tested**

Features mapped to SRS requirement IDs:  
**Authentication & Security:**

-CRYPTO-F-001: Email verification during registration

-CRYPTO-F-002: 2FA enforcement using TOTP

-CRYPTO-F-003: Automatic session timeout after 30 minutes

**Portfolio Management:**

-CRYPTO-F-004: Multiple portfolio creation with custom names

-CRYPTO-F-005: Manual cryptocurrency holdings addition

-CRYPTO-F-006: Automatic transaction import from exchanges

**Performance Analytics:**

-CRYPTO-F-010: Multi-currency portfolio value calculation

-CRYPTO-F-011: Realized/unrealized gains/losses display

-CRYPTO-F-012: Portfolio allocation analysis with rebalancing suggestions

**Alerts & Notifications:**

-CRYPTO-F-013: Configurable price alerts for cryptocurrencies

-CRYPTO-F-014: Email portfolio performance summaries

-CRYPTO-F-015: Mobile push notifications for significant changes

**Exchange Integration:**

-CRYPTO-F-016: Read-only API integration with 10+ major exchanges

**Advanced Analytics:**

-CRYPTO-F-017: Portfolio volatility, Sharpe ratio, and maximum drawdown metrics

**NFT Management:**

-CRYPTO-F-018: Automatic NFT holdings detection from connected wallets

**Tax Reporting:**

-CRYPTO-F-019: Multi-jurisdiction tax report generation

**Data Management:**

-CRYPTO-F-020: Automatic daily encrypted backups with recovery

**Non-Functional Requirements:**

-CRYPTO-NF-001: Portfolio dashboard load time ≤ 3 seconds

-CRYPTO-NF-002: System uptime ≥ 99.5% monthly

-CRYPTO-NF-003: AES-256 encryption at rest, TLS 1.3 in transit

-CRYPTO-NF-004: Support for 1000+ concurrent users

-CRYPTO-NF-005: Mobile app resource usage limits

**Security Requirements:**

-CRYPTO-SR-001: Secure API communications with OAuth 2.0

-CRYPTO-SR-002: Rate limiting and DDoS protection

-CRYPTO-SR-003: Session expiry and re-authentication requirements

-CRYPTO-SR-004: Comprehensive audit logging

-CRYPTO-SR-005: Input validation and sanitization

**4. Features Not to be Tested**

- Core blockchain protocol implementations (assumed stable and tested by blockchain providers)

- Third-party cryptocurrency exchange internal systems (vendor responsibility)

- External market data provider APIs (assumed reliable and tested by providers)

- Hardware wallet firmware (vendor responsibility)

- Core banking systems for fiat currency transactions

- Third-party payment processing internals (Stripe, etc.)

- Cloud infrastructure provider services (AWS/Azure internals)

**5. Test Approach / Strategy**

**Levels:**  
- Unit tests (React components, Rust functions and modules)  
- Integration tests (React-Rust API communication, database interactions)  
- System tests (end-to-end cryptocurrency portfolio management)  
- User Acceptance tests (UAT with real users)

- Security tests (penetration testing, vulnerability assessment

**Types:**  
 - Functional testing (core portfolio features)

* Regression testing (after each release)
* Performance testing (load, stress, volume testing)
* Usability testing (web and mobile UI/UX)
* Compatibility testing (browsers, mobile devices)
* API testing (exchange integrations, market data feeds)
* Component testing (React UI components and state management)
* Backend service testing (Rust API endpoints and business logic)
* Security testing (authentication, data protection)

**Entry Criteria:**

* Stable build delivered with all planned features
* Test environment configured with exchange API sandboxes
* Test data prepared (dummy portfolios, transactions, market data)
* All test cases reviewed and approved

**Exit Criteria:**

* 100% of planned test cases executed
* 0 critical defects open
* All high-priority functional requirements verified
* Performance benchmarks met
* Security requirements validated
* 95% code coverage achieved
* All acceptance criteria satisfied

**5.1 Security Validation**

* Validate 2FA implementation (TOTP, SMS)
* API key encryption and secure storage verification
* TLS 1.3 encryption validation for all communications
* OAuth 2.0 implementation testing for exchange APIs
* Input sanitization and XSS prevention testing
* SQL injection prevention verification
* Rate limiting and DDoS protection testing
* Session management and timeout validation
* GDPR compliance verification (data privacy, deletion)
* Penetration testing of authentication and API endpoints
* Audit log completeness and security validation

**6. Test Environment**

**Hardware:**

* Web servers (load balancer, application servers)
* Database servers (primary and backup)
* Mobile devices (iOS/Android for app testing)
* Hardware security keys for 2FA testing

**Software:**

* Crypto Portfolio Tracker App v1.0
* Exchange API sandboxes (Binance, Coinbase, Kraken test environments)
* Market data API test endpoints (Coin Gecko, CoinMarketCap)
* Mobile applications (iOS/Android builds)
* Node.js runtime environment (for React development)
* Rust toolchain and Cargo package manager
* React frontend application v1.0
* Rust backend services v1.0
* Database systems (PostgreSQL/MySQL)

**Tools:**

* Selenium WebDriver (web UI automation)
* Appium (mobile app automation)
* Postman/Newman (API testing)
* JMeter (performance and load testing)
* OWASP ZAP (security testing)
* Jira (defect tracking and test management)
* Jenkins (CI/CD pipeline testing)
* Docker (containerized test environments)
* Jest and React Testing Library (React component testing)
* Cargo test framework (Rust unit testing)
* Cypress or Playwright (React end-to-end testing)

**Test Data:**

* Dummy user accounts with various portfolio configurations
* Sample cryptocurrency transaction data
* Mock exchange API responses
* Test wallet addresses and keys
* Sample NFT collections data
* Historical market data for testing

**7. Test Schedule**

Milestones:  
 - Test case design and review: 01-Oct-2025

* Test environment setup and configuration: 03-Oct-2025
* Exchange API sandbox integration: 04-Oct-2025
* Unit testing completion: 08-Oct-2025
* Integration testing start: 09-Oct-2025
* System testing execution start: 12-Oct-2025
* Performance and security testing: 15-Oct-2025
* Mobile app testing: 17-Oct-2025
* User Acceptance Testing (UAT): 20-Oct-2025 to 25-Oct-2025
* Test completion and sign-off: 27-Oct-2025

**8. Test Deliverables**

* Test Plan (this document)
* Test Cases (functional, integration, system, security)
* Automated Test Scripts (web and mobile)
* API Test Collections (Postman)
* Performance Test Scripts (JMeter)
* Security Test Reports (OWASP ZAP, penetration testing)
* Test Data Sets and Configuration
* Test Execution Logs and Reports
* Defect Reports and Resolution Status
* React Component Test Suites
* Rust Unit and Integration Test Results
* Frontend-Backend Communication Test Reports
* Requirements Traceability Matrix (RTM)
* Test Summary Report with metrics
* User Acceptance Test Results
* Mobile App Compatibility Reports

**9. Roles and Responsibilities**

|  |  |  |
| --- | --- | --- |
| Role | Name | Responsibility |
| QA Lead | Chinmay Shivanand Muragod | Prepare overall QA strategy, guide team, final approval on QA deliverables |
| Test Plan & Coordination | Darshith M S | Prepare test plan, coordinate execution, manage test team, stakeholder communication |
| Test Engineer - Backend | C Kaustubh | API testing, database testing, integration testing, performance testing |
| Test Engineer - Frontend | Chinmay Shivanand Muragod | Web UI testing, mobile app testing, usability testing, cross-browser compatibility |
| Security Test Engineer | Darshith M S | Security testing, penetration testing, compliance validation, audit logging verification |
| Full Stack Developer | Chirag K M | Test environment maintenance, CI/CD pipeline support, deployment assistance |
| Technical Lead | C Kaustubh | Support defect fixes, provide technical guidance, environment setup assistance |
| Test Data & Reporting Engineer | Chirag K M | Test data creation, automation result tracking, dashboard/report generation |
| Product Owner | Course Coordinator | Approve test results, acceptance criteria validation, final sign-off |

**10. Risks and Mitigation**

|  |  |
| --- | --- |
| Risk | Mitigation |
| Delay in stable build delivery | Request early smoke builds from dev team |
| Exchange API rate limiting during testing | Implement API call throttling, use multiple test accounts, coordinate with exchange providers |
| Third-party market data API downtime | Maintain backup data sources, implement mock data services for testing |
| Mobile device compatibility issues | Maintain device lab with various iOS/Android versions, use cloud testing platforms |
| Security vulnerabilities discovered late | Implement security testing throughout development cycle, regular security reviews |
| Performance degradation under load | Continuous performance monitoring, early load testing, scalability planning |
| Test data privacy and compliance issues | Use synthetic test data, implement data masking, ensure GDPR compliance |
| Complex multi-exchange integration failures | Phased integration approach, comprehensive API documentation, fallback mechanisms |
| React-Rust API communication failures or latency issues | Implement comprehensive API contract testing, use mock services for isolated testing, monitor response times |

**11. Assumptions & Dependencies**

**Assumptions:**

* Exchange API sandboxes will remain stable and accessible throughout testing
* Market data APIs will provide consistent test data
* Test users will have access to required mobile devices and browsers
* Network connectivity will be stable for real-time data testing
* Third-party services (email, SMS) will be available for 2FA testing

**Dependencies:**

* Exchange API credentials and sandbox access from providers
* Test cryptocurrency wallets and addresses setup
* Mobile app store approval for beta testing (if required)
* Security testing tools licenses and access
* Performance testing environment provisioning
* Test data preparation and database seeding
* Node.js and npm/yarn package managers for React development environment
* Rust compiler and Cargo for backend service compilation and testing
* SSL certificates and security configurations

**12. Suspension & Resumption Criteria**

**Suspend testing if:**

* Critical exchange APIs unavailable for >4 hours
* Build is unstable and blocks >50% of test cases
* Security vulnerabilities discovered that require immediate attention
* Test environment becomes completely inaccessible
* Major blocking defects prevent core functionality testing

**Resume testing if:**

* Blocking defects are resolved and verified
* Exchange API connectivity is restored
* Test environment is stabilized and validated
* Security issues are addressed and confirmed
* New stable build is available with fixes

**13. Test Case Management & Traceability**

RTM ensures mapping of SRS requirements to test cases.  
**Authentication Examples:**

* CRYPTO-F-001 (Email verification) → TC-Auth-01, TC-Auth-02, TC-Auth-03
* CRYPTO-F-002 (2FA enforcement) → TC-Auth-04, TC-Auth-05, TC-Auth-06
* CRYPTO-F-003 (Session timeout) → TC-Auth-07, TC-Auth-08

**Portfolio Management Examples:**

* CRYPTO-F-004 (Multiple portfolios) → TC-Portfolio-01, TC-Portfolio-02
* CRYPTO-F-005 (Manual holdings) → TC-Portfolio-03, TC-Portfolio-04
* CRYPTO-F-006 (Exchange import) → TC-Portfolio-05, TC-Portfolio-06

**Market Data Examples:**

* CRYPTO-F-007 (Real-time prices) → TC-Market-01, TC-Market-02
* CRYPTO-F-008 (Historical charts) → TC-Market-03, TC-Market-04
* CRYPTO-F-009 (Percentage changes) → TC-Market-05, TC-Market-06

**Performance Examples:**

* CRYPTO-NF-001 (Dashboard load time) → TC-Perf-01, TC-Perf-02
* CRYPTO-NF-004 (Concurrent users) → TC-Scale-01, TC-Scale-02

**14. Test Metrics & Reporting**

**Metrics collected:**

* Test execution progress (% test cases executed)
* Test pass/fail rates by module
* Defect discovery rate and severity distribution
* API response time and reliability metrics
* Code coverage percentage
* Requirement coverage percentage
* Defect aging and resolution time
* Security vulnerability count and risk assessment
* Performance benchmarks (response time, throughput)
* Mobile app crash rates and performance metrics

**Reports:**

* Daily test execution status dashboard
* Weekly test progress reports with metrics
* Defect summary reports by severity and module
* API performance and reliability reports
* Security testing summary with risk assessment
* Mobile compatibility matrix reports
* Final Test Summary Report with recommendations
* Requirements coverage analysis report

**15. Approvals**

|  |  |  |
| --- | --- | --- |
| Role | Name | Signature / Date |
| QA Lead | Chinmay Shivanand Muragod |  |
| Test Team Lead | Chirag K M |  |
| Security Test Lead | Darshith M S |  |
| Dev Lead | C Kaustubh |  |
| Product Owner | Course Coordinator |  |

***Thank You***