

OCH Cyber Talent Engine

*Developer's Specification
Document (DSD)*

Prepared on 13th October

Table of Contents

| | |
|---|-----------|
| OCH Cyber Talent Engine Developer's Specification Document | 1 |
| 1. Document Overview..... | 7 |
| 1.1 Purpose of the Document | 7 |
| 1.2 Product Vision (High-Level) | 7 |
| 1.3 Core Philosophy | 8 |
| 1.3.1 "Mentees do the work." | 8 |
| 1.3.2 "We guide the transformation." | 8 |
| 1.4 Scope of the System..... | 8 |
| 1.5 Definitions & Terminology..... | 9 |
| 2. Product Architecture (High-Level) | 10 |
| 2.1 System Overview..... | 10 |
| 2.2 Component Map..... | 10 |
| 2.2.1 Frontend..... | 10 |
| 2.2.2 Backend Services..... | 10 |
| 2.3 Data Flow Architecture | 11 |
| Example Data Flow: | 11 |
| 2.4 Module Dependencies | 11 |
| 3. System Roles | 12 |
| 3.1 Mentee (Primary User)..... | 12 |
| 3.2 Mentor..... | 12 |
| 3.3 Program Director..... | 12 |
| 3.4 Employer..... | 13 |
| 3.5 Admin..... | 13 |
| 4. System Architecture..... | 13 |
| 4.1 High-Level Diagram (Textual)..... | 13 |
| 4.2 Technology Assumptions (Flexible)..... | 14 |
| 4.3 Integration Layer | 14 |
| 5. Functional Requirements | 15 |
| 5.1 User Onboarding & Identity..... | 15 |
| 5.1.1 Account Creation | 15 |
| 5.1.2 Profile Completion | 15 |
| 5.2 Profiler Engine (Future-You Projection)..... | 16 |

| | |
|---|----|
| 5.2.1 Profiler Introduction | 16 |
| 5.2.2 Current Self-Assessment | 16 |
| 5.2.3 Future-You Projection..... | 16 |
| 5.2.4 Track Recommendation | 17 |
| 5.2.5 Profiler Output..... | 18 |
| 5.3 Subscription Engine — \$3 / \$7 Tiers..... | 18 |
| 5.3.1 Subscription State Management..... | 18 |
| 5.3.2 \$3 Starter — Enhanced Access (First 6 Months) | 18 |
| 5.3.3 \$3 Starter — Normal Mode..... | 19 |
| 5.3.4 \$7 Premium..... | 19 |
| 5.3.5 Upgrade/Downgrade..... | 19 |
| 5.4 Payment Engine | 20 |
| 5.4.1 Payment Methods..... | 20 |
| 5.4.2 Payment Events | 20 |
| 5.4.3 Entitlement Management..... | 20 |
| 5.4.4 Free Tier (Non-paying users) | 21 |
| 5.4.5 Webhooks..... | 21 |
| 5.5 Coaching OS (Habits, Goals, Actions, Reflection) | 21 |
| 5.5.1 Habits Engine | 21 |
| 5.5.2 Goals Engine..... | 22 |
| 5.5.3 Reflection Engine | 22 |
| 5.5.4 AI Coach | 22 |
| 5.6 Curriculum Engine | 23 |
| 5.6.1 Curriculum Structure..... | 23 |
| 5.6.2 Track View | 23 |
| 5.6.3 Module Access | 23 |
| 5.7 Missions Engine (MXP)..... | 23 |
| 5.7.1 Mission Types | 23 |
| 5.7.2 Mission Submission..... | 24 |
| 5.7.3 AI Feedback..... | 24 |
| 5.7.4 Mentor Review (\$7 only) | 24 |
| 5.8 Portfolio Engine | 25 |
| 5.8.1 Portfolio Structure | 25 |

| | |
|---|----|
| 5.8.2 Portfolio Visibility | 25 |
| 5.9 TalentScope Analytics Engine..... | 25 |
| 5.9.1 TalentScope Inputs | 25 |
| 5.9.2 TalentScope Outputs | 26 |
| 5.9.3 Mentor Influence Index..... | 26 |
| 5.9.4 Career Readiness Report..... | 26 |
| 10. Non-Functional Requirements (NFRs) | 27 |
| 10.1 Performance Requirements | 27 |
| 10.1.1 Response Times | 27 |
| 10.1.2 Page Load Performance | 27 |
| 10.1.3 Scalability..... | 27 |
| 10.2 Availability Requirements..... | 28 |
| 10.3 Reliability Requirements..... | 28 |
| 10.4 Data Integrity Requirements | 28 |
| 10.5 Usability Requirements | 29 |
| 10.6 Accessibility Requirements | 29 |
| 10.7 Localization Requirements..... | 29 |
| 10.8 Logging Requirements..... | 30 |
| 10.9 Monitoring Requirements | 30 |
| 10.10 Backup & Disaster Recovery | 30 |
| 11. Security Requirements | 31 |
| 11.1 Authentication | 31 |
| 11.2 Authorization..... | 31 |
| 11.3 Data Security | 32 |
| 11.4 File Upload Security | 32 |
| 11.5 AI Security..... | 32 |
| 11.6 Mentor/Employer Data Access | 32 |
| 11.7 Activity Audit Trail..... | 33 |
| 11.8 Rate Limiting..... | 33 |
| 11.9 Session Security | 33 |
| 11.10 Payment Security | 33 |
| 12. Compliance & Governance | 34 |
| 12.1 GDPR & POPIA | 34 |

| | |
|--|----|
| 12.2 User Consent & Privacy | 34 |
| 12.3 Data Minimization | 34 |
| 12.4 Data Retention..... | 35 |
| 12.5 Copyright & IP..... | 35 |
| 12.6 Community Governance..... | 35 |
| 12.7 Audit & Oversight..... | 35 |
| 13. Data Model & ERD (Entity Relationship Design) | 36 |
| 13.1 Core User & Identity Entities | 36 |
| 13.1.1 <code>users</code>..... | 36 |
| 13.1.2 <code>user_profiles</code> | 36 |
| 13.1.3 <code>roles</code> & <code>user_roles</code> (optional if multi-role is needed)..... | 37 |
| 13.2 Profiler & Track Mapping Entities..... | 37 |
| 13.2.1 <code>profiler_sessions</code>..... | 37 |
| 13.2.2 <code>profiler_answers</code> | 37 |
| 13.2.3 <code>future_you_profiles</code>..... | 37 |
| 13.2.4 <code>tracks</code> | 38 |
| 13.2.5 <code>user_tracks</code> | 38 |
| 13.3 Subscription & Payment Entities | 38 |
| 13.3.1 <code>plans</code>..... | 38 |
| 13.3.2 <code>subscriptions</code> | 39 |
| 13.3.3 <code>payment_providers</code>..... | 39 |
| 13.3.4 <code>payment_transactions</code>..... | 39 |
| 13.3.5 <code>entitlements</code>..... | 40 |
| 13.4 Coaching OS Entities | 40 |
| 13.4.1 <code>habits</code> | 40 |
| 13.4.2 <code>habit_logs</code>..... | 40 |
| 13.4.3 <code>goals</code>..... | 41 |
| 13.4.4 <code>reflections</code> | 41 |
| 13.5 AI Coach Entities | 42 |
| 13.5.1 <code>ai_coach_sessions</code>..... | 42 |
| 13.5.2 <code>ai_coach_messages</code> | 42 |
| 13.6 Curriculum & Missions Entities..... | 42 |
| 13.6.1 <code>curriculum_modules</code> | 42 |

| | |
|--|----|
| 13.6.2 lessons | 43 |
| 13.6.3 missions | 43 |
| 13.6.4 mission_submissions | 43 |
| 13.6.5 mission_files | 44 |
| 13.6.6 mentor_reviews | 44 |
| 13.7 Mentorship & Program Entities | 44 |
| 13.7.1 mentors | 44 |
| 13.7.2 mentee_mentor_assignments | 44 |
| 13.7.3 mentor_sessions | 45 |
| 13.7.4 session_attendance | 45 |
| 13.8 Community Entities | 45 |
| 13.8.1 communities | 45 |
| 13.8.2 community_memberships | 46 |
| 13.8.3 community_posts | 46 |
| 13.8.4 community_reactions | 46 |
| 13.8.5 leaderboard_snapshots | 47 |
| 13.9 Portfolio & Marketplace Entities | 47 |
| 13.9.1 portfolio_items | 47 |
| 13.9.2 employers | 47 |
| 13.9.3 marketplace_profiles | 48 |
| 13.9.4 employer_interest_logs | 48 |
| 13.10 TalentScope Entities | 48 |
| 13.10.1 ts_skill_signals | 48 |
| 13.10.2 ts_behavior_signals | 49 |
| 13.10.3 ts_readiness_snapshots | 49 |
| 13.10.4 ts_mentor_influence | 49 |
| 13.10.5 ts_ecosystem_metrics | 50 |
| 14. Payment Integration Logic | 50 |
| 14.1 Subscription Creation Flow | 50 |
| 14.2 Renewal Flow | 50 |
| 14.3 Upgrade \$3 → \$7 Flow | 51 |
| 14.4 Downgrade \$7 → \$3 Flow | 51 |
| 14.5 Refund & Reconciliation | 51 |

1. Document Overview

1.1 Purpose of the Document

The purpose of this Developer Specification Document (DSD v7) is to define, in full operational and technical detail, the functional requirements, non-functional requirements, system architecture, data models, modules, workflows, and interaction patterns required to build the **OCH Cyber Talent Engine** — a comprehensive, AI-driven mentorship, coaching, and talent-development ecosystem designed to produce Africa's elite cyber workforce.

This document serves multiple audiences:

- **Product leadership** (to validate vision & scope)
- **Engineering teams** (backend, frontend, mobile, DevOps)
- **Design/UI/UX teams** (interface flows & structural interactions)
- **Data science/AI teams** (TalentScope, scoring models, AI Coach logic)
- **QA teams** (testing requirements & acceptance criteria)
- **Founders/investors** (alignment with business model & product value)

It is intended to be detailed enough to begin full-scale engineering.

1.2 Product Vision (High-Level)

The OCH Cyber Talent Engine is an **AI-powered cyber transformation OS** that guides Africans from curiosity → competence → mastery → employability. It is not an LMS; it is a **mentorship and coaching engine** that:

1. Profiles a learner's **current state**
2. Projects their **Future-You** cyber persona
3. Generates a personalized **skills, habits & mission journey**
4. Drives daily actions, practice, and reflection
5. Builds a portfolio that employers trust
6. Creates a community of transformation
7. Uses analytics (TalentScope) to quantify readiness
8. Enables mentors & program directors to shape elite talent
9. Provides employers access to job-ready professionals

Unlike traditional cyber programs, OCH focuses on:

- Identity formation
- Habit discipline
- Skills mastery through missions

- Mentorship & behavioral development
- Real-time analytics of talent readiness
- A seamless ecosystem from “beginner” to “hireable”

1.3 Core Philosophy

OCH is built on two philosophical principles:

1.3.1 “Mentees do the work.”

- They learn
- They practice
- They self-reflect
- They build habits
- They submit missions
- They grow their portfolio

1.3.2 “We guide the transformation.”

OCH provides:

- Structure
- Mentorship
- Journey architecture
- AI coaching
- Deep analytics
- Career readiness visibility
- A community of excellence

This creates transformation at scale.

1.4 Scope of the System

The Cyber Talent Engine consists of the following major subsystems:

1. **OCH Profiler (Identity + Future-You Projection)**
2. **Coaching OS (Habits + Goals + Actions + Reflection)**
3. **AI Coach (Guidance, nudges, learning plans)**
4. **Mentorship OS (Program directors, mentors, cohorts)**
5. **Curriculum Engine (Modules, tracks, specializations)**
6. **Missions/MXP (Missions Execution Platform)**
7. **Portfolio Engine**
8. **Community Engine**
9. **Marketplace (Employer → Talent matching)**
10. **TalentScope Analytics Engine**

11. **Events & Calendar Engine**
12. **Integrations Layer (LMS, labs, Zoom, payment gateway)**
13. **Subscription Engine (\$3/\$7 with 6-month enhanced access)**
14. **Payment Engine (Billing, entitlements, renewals)**
15. **Admin Console**

Every module interacts with others through the **Core User Graph** and **Talent Data Fabric**, detailed later.

1.5 Definitions & Terminology

Some key definitions:

| Term | Meaning |
|-------------------------|---|
| User/Mentee | Someone undergoing training |
| Mentor | Human coach guiding users |
| Program Director | Senior mentor overseeing tracks |
| Track | Cyber specialization recommended by profiler |
| Mission | A practical challenge that produces skill evidence |
| Portfolio Item | A user's completed mission or project |
| TalentScope | The analytics engine measuring readiness |
| Future-You | AI-generated projection of ideal cyber identity |
| Coaching OS | Habit, goals, actions, reflections subsystem |
| Entitlement | Feature access determined by subscription level |
| Enhanced Access | 6-month full access to AI & auto features in \$3 tier |
| Marketplace | Where employers view and hire talent |

2. Product Architecture (High-Level)

2.1 System Overview

The OCH Cyber Talent Engine is a **modular, API-driven SaaS platform** built on:

- Microservices or modular monolith
- AI-powered scoring & guidance
- Data-driven behavioral analytics
- Multi-role access (mentee, mentor, employer, admin)
- Multi-tenant, regionally scalable deployment

It operates through interconnected subsystems:

Profiler → Track Mapping → Coaching OS → Missions → Portfolio → TalentScope → Marketplace

With:

- AI Coach acting as the “auto mentor”
- Mentorship OS guiding human mentors
- Payment Engine enforcing entitlements
- Community Engine ensuring engagement
- Curriculum Engine structuring progression
- Integrations Layer enabling learning tools

2.2 Component Map

2.2.1 Frontend

- Web App (React/Vue)
- Mobile App (React Native)
- Mentor Dashboard
- Program Director Dashboard
- Employer Dashboard
- Admin Console

2.2.2 Backend Services

| Service | Responsibilities |
|------------------|--|
| User Service | Identity, profile, access roles |
| Profiler Service | Assessments, Future-You, initial mapping |

| | |
|-----------------------------|---|
| Coaching OS Service | Habits, goals, reflections |
| Mission Service | Mission definitions, submissions, scoring |
| Mentorship Service | Groups, sessions, mentoring assignments |
| TalentScope Service | Analytics, readiness scoring |
| Subscription Service | \$3/\$7 tiers + entitlements |
| Payment Service | Billing, renewals, webhooks |
| Community Service | Groups, chats, leaderboards |
| Portfolio Service | Uploads, public profiles |
| Marketplace Service | Employer browsing & filters |
| Integration Service | Zoom, LMS, labs, calendars |
| Notification Service | Emails, push, in-app nudges |
| File Storage Service | Portfolio, submissions, recordings |

2.3 Data Flow Architecture

Example Data Flow:

User Submits Mission → AI → Mentor (if \$7) → Portfolio → TalentScope

1. User uploads mission →
2. Mission Service receives file →
3. AI Service analyzes →
4. If \$7 tier, Mentor OS receives scoring request →
5. Final score stored in Mission DB →
6. TalentScope updates readiness, skill heatmaps →
7. Portfolio updates evidence →
8. Marketplace updates profile ranking

This chain ensures **real-time skill visibility**.

2.4 Module Dependencies

Each module relies on others:

- Profiler → Track Engine
- Track Engine → Curriculum Engine
- Curriculum Engine → Missions

- Missions → Portfolio + TalentScope
- Coaching OS → TalentScope
- Community → Track Engine
- Marketplace → Portfolio + TalentScope
- Payment Engine → Subscription & all entitlements

This creates a unified experience.

3. System Roles

3.1 Mentee (Primary User)

Abilities:

- Complete profiler
- Receive Future-You projection
- Access Coaching OS
- Work on missions
- Join community
- Build portfolio
- Track progress in TalentScope
- Upgrade/downgrade subscription

Tier restrictions apply per subscription logic.

3.2 Mentor

Abilities:

- Run group mentorship sessions
- Review mission submissions (Professional tier)
- Score capstone projects
- Leave written or audio feedback
- Tag technical competencies
- Raise flags for struggling mentees
- Track mentee performance via TalentScope Mentor View

3.3 Program Director

Abilities:

- Approve tracks
- Modify scoring breakdown per track
- Assign mentors

- Override mentee placements
- Access track-level analytics
- Manage mentor reviews
- Publish or adjust missions

3.4 Employer

Abilities:

- Browse talent marketplace
- Filter by skill, readiness, portfolio depth
- Access mentee profiles based on subscription visibility rules
- Contact Professional-tier mentees
- Post roles or assignments

Starter-tier visibility is limited.

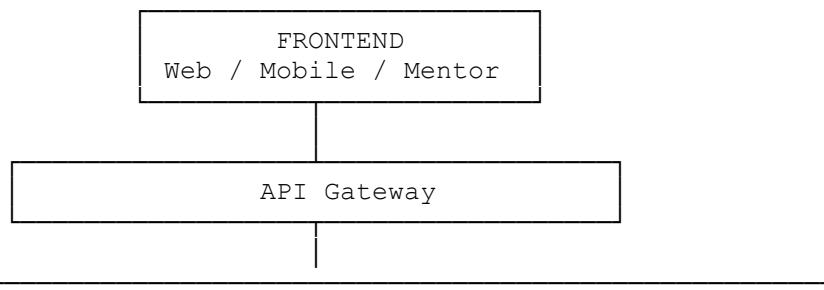
3.5 Admin

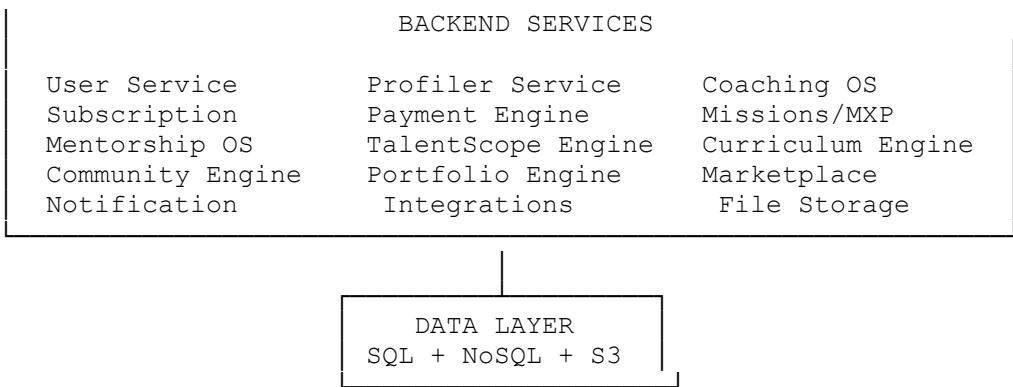
Abilities:

- Manage users across all roles
- Adjust system settings
- Configure subscription rules
- Audit logs
- Manage payment settings
- Oversee community
- Adjust curriculum or missions
- Manage integrations

4. System Architecture

4.1 High-Level Diagram (Textual)





4.2 Technology Assumptions (Flexible)

- Backend: Node.js / Django / Go
- Frontend: React + React Native
- AI Layer: Python microservices + LangChain
- Storage: S3 / GCS for files
- Database: PostgreSQL + Redis
- Analytics: Clickhouse or BigQuery
- Notifications: Firebase + Email provider
- Payments: Stripe / Paystack / Flutterwave

4.3 Integration Layer

The platform SHALL support:

1. **Zoom/Google Meet** for mentorship sessions
2. **LMS integrations** (Coursera, Antsyphon, etc.)
3. **Practical Labs** (TryHackMe, HackTheBox, custom labs)
4. **Calendar Sync**
5. **Payment Gateways (Stripe/Paystack)**

5. Functional Requirements

The Functional Requirements are grouped by module.

Each FR block includes:

- Description
- Scope
- Edge cases
- Entitlement logic
- Inter-service dependencies
- Data requirements

5.1 User Onboarding & Identity

5.1.1 Account Creation

FR-USER-1

The system SHALL allow users to register using:

- Email + password
- Google SSO
- Apple ID

FR-USER-2

The system SHALL require verification via email OTP.

FR-USER-3

The system SHALL assign a default role = “Mentee” on creation.

FR-USER-4

The system SHALL auto-trigger the **Profiler Onboarding** upon first login.

5.1.2 Profile Completion

FR-USER-5

Profile fields SHALL include:

- Full name
- Country
- Time zone
- Preferred learning style
- Career goals

- Cyber exposure level
- Optional: LinkedIn profile

FR-USER-6

Allow user to skip non-critical fields but prompt completion later.

FR-USER-7

Profile SHALL feed into TalentScope baseline calculations.

5.2 Profiler Engine (Future-You Projection)

5.2.1 Profiler Introduction

FR-PROF-1

Upon onboarding, the system SHALL require the mentee to complete a **two-phase profiler**:

1. **Current Self-Assessment** (skills, behaviors, identity)
2. **Future-You Time Machine Exercise**

5.2.2 Current Self-Assessment

FR-PROF-2

The system SHALL ask structured questions in four categories:

1. Skill exposure
2. Behavioral traits
3. Learning preferences
4. Career aspirations

FR-PROF-3

Answers SHALL be used to compute:

- Current Identity Pattern
- Behavioral Readiness Index
- Learning Velocity Estimate
- Skill Baseline Score

These form the TalentScope “Day Zero” metrics.

5.2.3 Future-You Projection

FR-PROF-4

The system SHALL ask the user to:

- Describe desired future state
- Choose values
- Identify motivators
- Visualize ideal professional identity
- Select long-term impact goals

FR-PROF-5

An AI model SHALL generate:

- A named Future-You Persona
- A persona description
- Projected skills
- Projected impact
- Career identity archetype
- A recommended track

5.2.4 Track Recommendation

FR-PROF-6

The system SHALL map the user to a recommended track based on:

- Persona
- Behavioral data
- Skill baseline
- Values
- Career motivation

FR-PROF-7

Tracks include:

- Cyber Defense
- Cloud Security
- GRC & Governance
- SOC Analyst
- Threat Intel & DFIR
- Offensive Security
- Identity & Access Security
- Secure Architecture
- DevSecOps
- Cyber Leadership (advanced)

FR-PROF-8

Track SHALL NOT be permanently locked.

However, the system SHALL lock the mentee's **learning experience** around this selection unless overridden manually.

FR-PROF-9

A mentee may switch tracks **once per month**, but the system SHALL issue a warning about loss of progress alignment.

5.2.5 Profiler Output

FR-PROF-10

Profiler output SHALL produce:

- Track recommendation
- Future-You Persona
- Identity Alignment Summary
- Learning strategy
- First 30-day action plan
- Mission difficulty level suggestion
- Habit recommendations
- Estimated readiness window
- TalentScope baseline

5.3 Subscription Engine — \$3 / \$7 Tiers

5.3.1 Subscription State Management

FR-SUB-1

The system SHALL enforce entitlements at the feature level.

FR-SUB-2

The system SHALL store:

- Subscription tier
- Start date
- Renewal date
- Payment method
- Enhanced Access flag (true/false)

5.3.2 \$3 Starter — Enhanced Access (First 6 Months)

The system SHALL:

- Unlock full AI features
- Unlock full mission catalog (AI-only missions)

- Unlock unlimited portfolio
- Unlock community
- Unlock curriculum visibility
- Unlock TalentScope preview mode
- Prevent access to mentors, capstones, employer contact

FR-SUB-3

Enhanced Access SHALL expire exactly 180 days after first subscription.

5.3.3 \$3 Starter — Normal Mode

After 6 months:

- Limited AI
- Limited missions
- Limited portfolio (5 items)
- Basic TalentScope
- Limited community
- No employer contact
- No advanced analytics

5.3.4 \$7 Premium

Professional tier SHALL include:

- Mentorship
- Group sessions + recordings
- Mission mentor reviews
- Capstones
- Full AI
- Full TalentScope analytics
- Lab integrations
- Portfolio unlimited
- Full curriculum
- Marketplace contact enabled
- Readiness breakdown
- CV scoring
- Mentor influence index

5.3.5 Upgrade/Downgrade

FR-SUB-4

Upgrading SHALL take effect instantly.

FR-SUB-5

Downgrading SHALL take effect after the billing cycle ends.

5.4 Payment Engine

5.4.1 Payment Methods

FR-PAY-1

Platform SHALL support:

- Visa/Mastercard
- Mobile Money (M-Pesa, Orange, Airtel Money)
- Paystack/Flutterwave
- Stripe (for US/EU)

5.4.2 Payment Events

The system SHALL handle:

- Subscription creation
- Renewal
- Failure
- Grace period
- Cancellation
- Refund triggers
- Re-activation

FR-PAY-2

All payment events SHALL sync automatically into Subscription Service.

5.4.3 Entitlement Management

FR-PAY-3

Upon successful payment:

- Entitlements SHALL activate instantly
- Enhanced Access flag SHALL be applied to \$3 users only for the first 6 months

FR-PAY-4

If a payment fails:

- Grace period = 5 days
- After 5 days → auto-downgrade to Free Tier (read-only mode)

5.4.4 Free Tier (Non-paying users)

FR-PAY-5

Free Tier users SHALL get:

- Read-only curriculum
- Limited AI Coach (1/day)
- Limited community
- No missions
- No portfolio
- No TalentScope
- No marketplace

This incentivizes first upgrade.

5.4.5 Webhooks

FR-PAY-6

The system SHALL listen to payment gateway webhooks for:

- charge.succeeded
- charge.failed
- subscription.updated
- subscription.deleted

5.5 Coaching OS (Habits, Goals, Actions, Reflection)

5.5.1 Habits Engine

FR-HAB-1

System SHALL provide three core habits that cannot be deleted:

- Learn
- Practice
- Reflect

FR-HAB-2

Users SHALL define custom habits.

FR-HAB-3

Habit logs SHALL be used by TalentScope.

FR-HAB-4

Habit streaks SHALL influence readiness scoring.

5.5.2 Goals Engine

FR-GOAL-1

Users SHALL set:

- Monthly goals
- Weekly goals
- Daily actions

FR-GOAL-2

Professional-tier mentees SHALL receive mentor feedback on goals.

5.5.3 Reflection Engine

FR-REF-1

System SHALL provide daily reflection prompts.

FR-REF-2

AI SHALL perform sentiment analysis and behavior categorization.

FR-REF-3

Reflections SHALL generate learning insights in TalentScope.

5.5.4 AI Coach

FR-AI-1

AI Coach SHALL:

- Generate learning plans
- Recommend missions
- Optimize habits
- Provide identity alignment nudges
- Offer reflection interpretations
- Provide motivation statements

Entitlement affects prompt limits.

5.6 Curriculum Engine

5.6.1 Curriculum Structure

FR-CUR-1

Curriculum SHALL be hierarchical:

Track → Modules → Lessons → Missions

5.6.2 Track View

FR-CUR-2

Users SHALL see:

- Core modules
- Optional modules
- Progress status
- Mentor notes (\$7 only)
- Recommended order

5.6.3 Module Access

FR-CUR-3

\$3 Enhanced Access users → full visibility

\$3 Normal Access → limited visibility

\$7 Professional → full access

5.7 Missions Engine (MXP)

5.7.1 Mission Types

FR-MIS-1

Mission types SHALL include:

- Beginner
- Intermediate

- Advanced
- Capstone
- Practical labs
- Scenario-based missions (ransomware, phishing, DFIR)

5.7.2 Mission Submission

FR-MIS-2

Users SHALL upload:

- Files
- Screenshots
- Notebook links
- GitHub links
- Video walk-throughs

5.7.3 AI Feedback

FR-MIS-3

AI SHALL:

- Score fundamental correctness
- Identify gaps
- Suggest improvements
- Tag competencies

5.7.4 Mentor Review (\$7 only)

FR-MIS-4

Mentors SHALL:

- Provide deeper analysis
- Issue a pass/fail
- Add comments
- Tag skills
- Recommend next missions

5.8 Portfolio Engine

5.8.1 Portfolio Structure

FR-PF-1

Portfolio items SHALL consist of:

- Title
- Summary
- Evidence files
- Skills tagged
- Mission link

5.8.2 Portfolio Visibility

FR-PF-2

Starter users:

- Unlimited portfolio for first 6 months
- 5 items after
- Visible in marketplace (limited)

Professional users:

- Unlimited portfolio
- Full visibility

5.9 TalentScope Analytics Engine

5.9.1 TalentScope Inputs

FR-TS-1

TalentScope SHALL ingest:

- Habit logs
- Reflection sentiment
- Mission scores
- Curriculum progress
- AI coach interactions
- Mentor evaluations
- Community engagement
- Portfolio depth
- Identity alignment

- Skill coverage
- Lab data (Professional tier)

5.9.2 TalentScope Outputs

FR-TS-2

Outputs include:

- Core readiness score
- Skill heatmap
- Behavior/study patterns
- Learning velocity
- Estimated readiness window
- Strengths & weaknesses
- Career pathway insight
- Track-specific benchmarks

Starter-tier users see only “Preview Mode.”

5.9.3 Mentor Influence Index

FR-TS-3

For \$7 tier:

- Measures impact of mentor interactions
- Captures submission → feedback → performance correlation

5.9.4 Career Readiness Report

FR-TS-4

Professional tier gets:

- Full readiness breakdown
- Job fit score
- Hiring timeline prediction
- Missing skills
- Improvement plan

Starter sees teaser metrics only.

10. Non-Functional Requirements (NFRs)

Non-functional requirements define how the system must behave, regardless of the functional modules.

They ensure the platform is **fast, reliable, secure, scalable, and maintainable**.

10.1 Performance Requirements

10.1.1 Response Times

NFR-PERF-1

All standard user-facing API calls SHALL respond within < **500ms** under normal load.

NFR-PERF-2

High-load operations (AI scoring, analytics, mission uploads) SHALL respond within:

- Real-time response: < 2 seconds
- Background async queue: < 15 seconds

NFR-PERF-3

Admin dashboard analytics queries SHALL execute within < **2 seconds** on indexed datasets.

10.1.2 Page Load Performance

NFR-PERF-4

All pages SHALL load the first meaningful content within < **3 seconds** on a standard 3G mobile network.

NFR-PERF-5

Static assets SHALL be served through CDN caching.

10.1.3 Scalability

NFR-PERF-6

The system SHALL scale horizontally to support:

- 100,000+ mentees
- 5,000+ mentors
- 10,000+ concurrent users
- 1M+ monthly mission uploads

NFR-PERF-7

Microservices SHALL scale independently based on load (missions, AI, TalentScope especially).

10.2 Availability Requirements

NFR-AVAIL-1

The platform SHALL maintain **99.5% monthly uptime** minimum.

NFR-AVAIL-2

Critical services (authentication, missions, AI) SHALL maintain **99.9% uptime**.

NFR-AVAIL-3

Scheduled maintenance SHALL be communicated 24 hours in advance.

10.3 Reliability Requirements

NFR-REL-1

Mission submissions SHALL NEVER be lost.

Uploads MUST be atomic: either fully stored or rolled back.

NFR-REL-2

Community posts SHALL be stored durably and recoverable after failures.

NFR-REL-3

Payment events SHALL be idempotent to avoid double-charging.

10.4 Data Integrity Requirements

NFR-DATA-1

All mutable fields SHALL be versioned for audit (mentorship notes, mission reviews).

NFR-DATA-2

TalentScope SHALL maintain immutable “Skill Evidence Logs.”

NFR-DATA-3

Portfolio items SHALL maintain provenance of origin (mission ID, mentor ID, timestamps).

10.5 Usability Requirements

NFR-USE-1

The platform SHALL follow mobile-first UI design.

NFR-USE-2

All critical actions SHALL include tooltips or contextual hints.

NFR-USE-3

The onboarding flow SHALL not exceed 5 minutes.

NFR-USE-4

Beginner users SHALL always see prompts guiding their next action (“What to do next”).

10.6 Accessibility Requirements

NFR-ACC-1

The system SHALL comply with **WCAG 2.1 AA** for:

- Color contrast
- Text alternatives
- Keyboard navigation

NFR-ACC-2

Screen reader support SHALL be enabled for major UI components.

10.7 Localization Requirements

NFR-LOC-1

System SHALL support language localization for:

- English (default)
- French (future)
- Swahili (future)

NFR-LOC-2

Date formats SHALL follow user locale.

10.8 Logging Requirements

NFR-LOG-1

System SHALL log all:

- Login attempts
- Subscription changes
- Mentor actions
- Mission scoring
- AI interactions
- Payment events
- Admin changes
- Portfolio updates

NFR-LOG-2

Logs SHALL be immutable and stored for 12 months minimum.

10.9 Monitoring Requirements

NFR-MON-1

Platform SHALL be monitored via:

- API latency
- Error rates
- AI service failures
- Mission upload failures
- Payment failures
- CPU, memory usage
- Queue backlogs

NFR-MON-2

Critical alerts MUST trigger:

- Email
- Slack/Teams integration
- PagerDuty/SMS (optional)

10.10 Backup & Disaster Recovery

NFR-DR-1

Daily backups SHALL be stored in geo-redundant storage.

NFR-DR-2

Recovery Point Objective (RPO): 12 hours.

NFR-DR-3

Recovery Time Objective (RTO): 1 hour.

11. Security Requirements

Security is a core requirement of a cybersecurity talent platform. These requirements ensure trust, privacy, and resilience.

11.1 Authentication

SEC-AUTH-1

System SHALL use secure password hashing (bcrypt or Argon2).

SEC-AUTH-2

System SHALL support MFA (optional for mentees, required for admins).

SEC-AUTH-3

JWT with rotation SHALL be used for session management.

11.2 Authorization

SEC-AUTHZ-1

RBAC (Role-Based Access Control) SHALL govern all access.

Roles:

- Mentee
- Mentor
- Program Director
- Employer
- Admin

SEC-AUTHZ-2

Cross-role data access SHALL be prohibited unless explicitly permitted.

11.3 Data Security

SEC-DATA-1

All data in transit SHALL use TLS 1.2+.

SEC-DATA-2

All stored data SHALL use AES-256 encryption at rest.

SEC-DATA-3

PII MUST be protected with field-level encryption (email, phone).

11.4 File Upload Security

SEC-FILE-1

All mission uploads SHALL pass antivirus scanning.

SEC-FILE-2

Executable file types SHALL be blocked unless explicitly allowed.

SEC-FILE-3

Maximum file size SHALL be controlled (configurable, default 50MB).

11.5 AI Security

SEC-AI-1

AI prompts SHALL be sanitized to prevent injection attacks.

SEC-AI-2

AI outputs SHALL not expose internal system data.

11.6 Mentor/Employer Data Access

SEC-ROLES-1

Mentors SHALL only access mentees assigned to them.

SEC-ROLES-2

Employers SHALL only view talent permitted via Marketplace entitlements.

11.7 Activity Audit Trail

SEC-AUDIT-1

System MUST track:

- Who changed what
- When
- Before/after values
- Device/IP metadata

SEC-AUDIT-2

Admins SHALL view audit logs through a secure console.

11.8 Rate Limiting

SEC-RATE-1

API requests SHALL be rate-limited:

- Auth endpoints: 5/min
- AI endpoints: configurable based on tier
- Upload endpoints: 10/min

Prevents abuse & DDoS.

11.9 Session Security

SEC-SESSION-1

Session tokens SHALL auto-expire after:

- 24 hours of inactivity for mentees
- 12 hours for mentors
- 2 hours for admins

11.10 Payment Security

SEC-PAY-1

Platform SHALL NOT store raw credit card data.

SEC-PAY-2

All payment processing SHALL occur via PCI-DSS compliant gateway.

SEC-PAY-3

Webhooks MUST be validated with signature verification.

12. Compliance & Governance

12.1 GDPR & POPIA

Because users come from multiple jurisdictions:

GOV-PRIV-1

The platform SHALL comply with:

- GDPR
- Botswana Data Protection Act
- Kenya Data Protection Act
- Any applicable local cyber regulations

12.2 User Consent & Privacy

GOV-PRIV-2

Users SHALL explicitly consent to:

- Data collection
- Behavioral analytics
- AI-based guidance
- Community visibility
- Marketplace visibility

GOV-PRIV-3

Consent MUST be withdrawable.

12.3 Data Minimization

GOV-PRIV-4

Only data directly required for operations SHALL be collected.

12.4 Data Retention

GOV-PRIV-5

Personal data SHALL be deletable upon request (“Right to be forgotten”).

GOV-PRIV-6

Mentorship & mission records SHALL retain anonymized metrics even after deletion for ecosystem analytics.

12.5 Copyright & IP

GOV-IP-1

Missions, content, curriculum, and training materials remain property of OCH.

GOV-IP-2

Portfolio items belong to the mentee.

12.6 Community Governance

GOV-COMM-1

Community guidelines SHALL govern:

- Harassment
- Cheating
- Plagiarism
- Offensive content
- Abuse of AI tools

GOV-COMM-2

Admins SHALL have the ability to remove content or suspend accounts.

12.7 Audit & Oversight

GOV-AUDIT-1

Annual system audits SHALL be mandatory.

GOV-AUDIT-2

Security testing SHALL occur quarterly (Pen test or automated scans).

13. Data Model & ERD (Entity Relationship Design)

This section outlines the **core entities**, their relationships, and key fields.

It is not a full column-level schema but a **developer-ready conceptual + logical data model**.

Assumption: Primary DB is **PostgreSQL** with some analytic/OLAP layer (e.g., Clickhouse/BigQuery) for TalentScope.

13.1 Core User & Identity Entities

13.1.1 `users`

Represents any person in the system.

Key Fields:

- `id` (PK, UUID)
- `email` (unique)
- `password_hash` (nullable for SSO)
- `first_name`
- `last_name`
- `role` (enum: mentee, mentor, director, employer, admin)
- `timezone`
- `country`
- `status` (active, suspended, deleted)
- `created_at`
- `updated_at`

13.1.2 `user_profiles`

Extended profile metadata.

Fields:

- `user_id` (FK → `users.id`, PK)
- `headline`
- `bio`
- `linkedin_url`
- `profile_photo_url`
- `experience_level` (beginner, junior, mid, senior)
- `preferred_learning_style`
- `career_goal` (text)
- `tags` (JSONB)

13.1.3 `roles` & `user_roles` (optional if multi-role is needed)

If users need multiple roles, use `user_roles`.

13.2 Profiler & Track Mapping Entities

13.2.1 `profiler_sessions`

Stores each execution of the profiler.

Fields:

- `id`
- `user_id` FK
- `status` (`in_progress`, `completed`)
- `current_score_snapshot` (JSONB)
- `created_at`
- `completed_at`

13.2.2 `profiler_answers`

Stores answers to profiler questions.

Fields:

- `id`
- `profiler_session_id` FK
- `question_id`
- `answer_raw`
- `answer_normalized`
- `created_at`

13.2.3 `future_you_profiles`

AI-generated future persona.

Fields:

- `id`
- `user_id` FK
- `persona_name`
- `description` (text)

- `projected_skills` (JSONB)
- `projected_roles` (JSONB)
- `identity_archetype` (enum/string)
- `recommended_track_id` FK → `tracks.id`
- `generated_at`

13.2.4 `tracks`

Defines the main cyber tracks.

Fields:

- `id`
- `code` (e.g., “SOC_DEFENSE”)
- `name`
- `description`
- `level` (entry, intermediate, advanced)
- `is_active`

13.2.5 `user_tracks`

Mapping between users and tracks.

Fields:

- `id`
- `user_id`
- `track_id`
- `assigned_by` (system, mentor, director)
- `assigned_at`
- `current_status` (active, switched, completed)

13.3 Subscription & Payment Entities

13.3.1 `plans`

Configurable plans (e.g., Starter, Professional).

Fields:

- `id`
- `code` (STARTER_3USD, PRO_7USD)
- `name`
- `description`
- `monthly_price`
- `is_active`

13.3.2 `subscriptions`

User's active subscription state.

Fields:

- `id`
- `user_id`
- `plan_id` FK
- `status` (active, past_due, canceled, trial, expired)
- `start_date`
- `end_date` (nullable for auto-renew)
- `renewal_date`
- `enhanced_access_until` (datetime, null if not applicable)
- `created_at`
- `updated_at`

13.3.3 `payment_providers`

Which processor used.

Fields:

- `id`
- `name` (Stripe, Paystack, Flutterwave)
- `config` (JSONB)

13.3.4 `payment_transactions`

Logs all payment attempts.

Fields:

- `id`
- `user_id`
- `subscription_id`
- `provider_id`
- `provider_tx_id`
- `amount`
- `currency`
- `status` (success, failed, pending, refunded)
- `error_message` (nullable)

- `raw_payload` (JSONB)
- `created_at`

13.3.5 `entitlements`

Derived effective access.

Fields:

- `id`
- `user_id`
- `plan_id`
- `is_active`
- `entitlements_json` (JSONB – list of feature flags)
- `effective_from`
- `effective_to`

Note: Backend can store entitlements as feature keys like `"ai_coach_full": true, "mentor_access": false`, etc.

13.4 Coaching OS Entities

13.4.1 `habits`

Core and custom habits.

Fields:

- `id`
- `user_id`
- `name` (Learn, Practice, Reflect / custom)
- `type` (core, custom)
- `frequency` (daily, weekly)
- `is_active`
- `created_at`

13.4.2 `habit_logs`

Daily logs per habit.

Fields:

- id
- habit_id
- user_id
- status (completed, skipped)
- log_date
- metadata (JSONB)

13.4.3 goals

Goals with timeframe.

Fields:

- id
- user_id
- type (monthly, weekly)
- title
- description
- status (active, completed, abandoned)
- start_date
- end_date
- created_at

13.4.4 reflections

Daily/weekly reflections.

Fields:

- id
- user_id
- content (text)
- sentiment_score
- emotion_tags (JSONB)
- ai_insight_summary (text)
- created_at

13.5 AI Coach Entities

13.5.1 `ai_coach_sessions`

Tracks conversation sessions.

Fields:

- `id`
- `user_id`
- `session_type` (onboarding, weekly_plan, mission_help, reflection)
- `prompt_count`
- `metadata` (JSONB)
- `created_at`

13.5.2 `ai_coach_messages`

Messages in a conversation.

Fields:

- `id`
- `session_id`
- `role` (user, assistant, system)
- `content`
- `created_at`

13.6 Curriculum & Missions Entities

13.6.1 `curriculum_modules`

Modules under each track.

Fields:

- `id`
- `track_id`
- `title`
- `description`
- `order_index`
- `level` (beginner, intermediate, advanced)
- `is_required`

13.6.2 lessons

Optional if you break modules down further.

13.6.3 missions

Core mission definitions.

Fields:

- `id`
- `track_id` (nullable if cross-track)
- `module_id` (nullable)
- `title`
- `description`
- `difficulty` (1–5)
- `mission_type` (beginner, intermediate, advanced, capstone)
- `requires_mentor_review` (bool)
- `requires_lab_integration` (bool)
- `estimated_duration_min`
- `skills_tags` (JSONB)
- `is_active`

13.6.4 mission_submissions

Per-user submissions.

Fields:

- `id`
- `mission_id`
- `user_id`
- `status` (draft, submitted, ai_reviewed, mentor_reviewed, revision_requested, finalized)
- `ai_score`
- `mentor_score`
- `overall_status` (pass, fail, pending)
- `submission_data` (JSONB or file refs)
- `submitted_at`
- `ai_reviewed_at`
- `mentor_reviewed_at`

13.6.5 mission_files

File references.

Fields:

- id
- submission_id
- file_url
- file_type
- created_at

13.6.6 mentor_reviews

Explicit mentor notes per submission.

Fields:

- id
- submission_id
- mentor_id
- comments (text)
- score_breakdown (JSONB)
- recommended_next_steps (text)
- created_at

13.7 Mentorship & Program Entities

13.7.1 mentors

Mentor profiles.

Fields:

- user_id (PK, FK)
- bio
- expertise_tags (JSONB)
- availability (JSONB schedule)

13.7.2 mentee_mentor_assignments

Links mentees to mentors.

Fields:

- id

- `mentee_id`
- `mentor_id`
- `track_id`
- `assigned_by` (director/system)
- `assigned_at`

13.7.3 `mentor_sessions`

Group mentorship events.

Fields:

- `id`
- `mentor_id`
- `title`
- `description`
- `track_id`
- `session_type` (group, workshop)
- `meeting_link`
- `scheduled_for`
- `recording_url` (after)
- `transcript_url` (after)

13.7.4 `session_attendance`

Tracks who attended.

Fields:

- `id`
- `session_id`
- `user_id`
- `joined_at`
- `left_at`

13.8 Community Entities

13.8.1 `communities`

Community circles (track-based, general, thematic).

Fields:

- `id`
- `name`
- `description`

- `type` (open, track_READONLY, track_PRIVATE)
- `track_id` (nullable)
- `is_active`

13.8.2 `community_memberships`

Which users are in which circles.

Fields:

- `id`
- `community_id`
- `user_id`
- `role` (member, moderator)
- `joined_at`

13.8.3 `community_posts`

Posts in community.

Fields:

- `id`
- `community_id`
- `user_id`
- `content` (text/markdown)
- `attachments` (JSONB)
- `created_at`

13.8.4 `community_reactions`

Likes/upvotes.

Fields:

- `id`
- `post_id`
- `user_id`
- `reaction_type`

13.8.5 `leaderboard_snapshots`

Leaderboards captured periodically.

Fields:

- `id`
- `scope` (global, track, community)
- `data` (JSONB of ranked users + metrics)
- `snapshot_date`

13.9 Portfolio & Marketplace Entities

13.9.1 `portfolio_items`

Showcase work.

Fields:

- `id`
- `user_id`
- `title`
- `summary`
- `mission_id` (nullable)
- `evidence_files` (JSONB array)
- `skills_tags` (JSONB)
- `visibility_level` (private, marketplace_preview, full)
- `created_at`

13.9.2 `employers`

Employer-specific profile data.

Fields:

- `user_id` (PK, FK)
- `company_name`
- `company_website`
- `sector`
- `country`
- `description`

13.9.3 `marketplace_profiles`

How mentees appear in the marketplace.

Fields:

- `id`
- `user_id`
- `current_tier` (starter, professional)
- `readiness_score` (denormalized from TalentScope)
- `profile_status` (foundation_mode, emerging_talent, job_ready)
- `is_contact_enabled` (true/false)
- `last_updated_at`

13.9.4 `employer_interest_logs`

When employers view, favorite, or shortlist.

Fields:

- `id`
- `employer_id`
- `mentee_id`
- `interest_type` (view, save, shortlist, contact_request)
- `created_at`

13.10 TalentScope Entities

These represent the analytics backbone.

13.10.1 `ts_skill_signals`

Skill signals derived from missions, labs, etc.

Fields:

- `id`
- `user_id`
- `source` (mission, lab, mentor, ai)
- `skill_code`
- `signal_strength` (0–1)
- `recorded_at`
- `metadata` (JSONB)

13.10.2 ts_behavior_signals

Habit/behavioral signals.

Fields:

- `id`
- `user_id`
- `signal_type` (`habit_streak`, `consistency`, `reflection_frequency`)
- `value` (`numeric`)
- `recorded_at`

13.10.3 ts_readiness_snapshots

Periodic full readiness snapshots.

Fields:

- `id`
- `user_id`
- `overall_score`
- `skill_score`
- `behavior_score`
- `identity_alignment_score`
- `career_readiness_stage` (enum: `exploring`, `building`, `emerging`, `ready`)
- `snapshot_date`
- `breakdown` (`JSONB`)

13.10.4 ts_mentor_influence

Impact of mentors.

Fields:

- `id`
- `mentee_id`
- `mentor_id`
- `time_window_start`
- `time_window_end`
- `delta_readiness_score`
- `delta_skill_score`
- `metadata`

13.10.5 `ts_ecosystem_metrics`

Aggregated metrics for Africa Cyber Talent Reports.

Fields:

- `id`
- `time_window` (month, quarter, year)
- `region`
- `metric_code`
- `metric_value`
- `metadata`

14. Payment Integration Logic

This section clarifies **how the payment engine connects to entitlements and subscriptions**.

14.1 Subscription Creation Flow

1. User completes Profiler & chooses track
2. User is presented with **\$3 Starter** and **\$7 Professional**
3. User selects plan and payment method
4. Payment gateway checkout initiated
5. On success:
 - o `subscriptions` created/updated
 - o `entitlements` generated
 - o If \$3 and first-time subscription → `enhanced_access_until = now + 180 days`
6. User is redirected to dashboard with appropriate access.

14.2 Renewal Flow

- Payment provider triggers webhook → `payment_transactions` entry
- If success:
 - o Extend `renewal_date`
 - o Maintain entitlements
- If failure:
 - o Set `status = past_due`
 - o Start 5-day grace period
 - o Notify user

- After 5 days of no success → set `status = canceled`, downgrade entitlements to Free Tier.

14.3 Upgrade \$3 → \$7 Flow

- User chooses “Upgrade to Professional” from anywhere (TalentScope, missions, marketplace prompts).
- System generates upgrade checkout with proration (optional).
- On success:
 - `subscription.plan_id` updated to Professional
 - `status = active`
 - `entitlements` updated with Professional feature flags
 - Mentorship & full TalentScope access is granted instantly.

14.4 Downgrade \$7 → \$3 Flow

- User requests downgrade
- System schedules downgrade at period end
- At `renewal_date`:
 - `plan_id` changes to Starter
 - `enhanced_access_until` logic applies only if user is within first 6 months; otherwise, no enhanced access.
 - `entitlements` recalculated
 - TalentScope, Mentor access, marketplace contact adjust accordingly.

14.5 Refund & Reconciliation

- Admins can mark transactions as “refunded”
- Refunded subscriptions MAY be canceled immediately or downgraded to Free Tier
- All adjustments MUST be logged in `payment_transactions`.