SkyConnect
Impacting Life

ASSEMBLE INSURANCE TANZANIA LTD

Software Requirements Specification

SAS Infrastructure & Security Module

Version: 1.0

https://www.assemble.co.tz/

Table of Contents

1 Document Information

Field	Value
Project Name	SAS Infrastructure & Security Module
Version	1.0
Date Created	2025-10-17
Date Modified	2025-10-17
Document Type	Software Requirements Specification
Module Code	INFRASTRUCTURE
Prepared By	Sky Software Engineering Team
Status	Draft

2 Project Overview

2.1 Description

The Infrastructure & Security module provides the foundational technical infrastructure, security framework, integrations, and DevOps capabilities that support all SAS modules. It includes authentication & authorization, data security & encryption, audit logging, API management, integration framework, data warehouse & ETL, backup & disaster recovery, monitoring & alerting, performance optimization, and DevOps automation. This module ensures the system is secure, scalable, reliable, and compliant with data protection regulations while providing seamless integrations with external systems and comprehensive operational visibility.

2.2 Objectives

- Implement robust authentication and authorization (SSO, MFA, RBAC)

- Ensure data security through encryption at rest and in transit

- Maintain comprehensive audit trails for compliance

- Provide RESTful API framework with rate limiting and versioning

- Enable seamless integrations with external systems (Sage ERP, payment gateways, TIRA, etc.)

- Build data warehouse with ETL pipelines for analytics

- Implement automated backup and disaster recovery

- Provide real-time monitoring, alerting, and performance metrics

- Optimize system performance and scalability

- Automate DevOps processes (CI/CD, deployment, infrastructure as code)

2.3 Scope

2.3.1 In Scope

- Authentication & Authorization (SSO via Keycloak, MFA, RBAC, session management)

- Data Security (encryption at rest/transit, data masking, secrets management)

- Audit Logging (comprehensive audit trails, tamper-proof logs, 7-year retention)

- API Management (RESTful API, rate limiting, versioning, API gateway)

- External Integrations (Sage ERP, payment gateways, SMS, email, TIRA portal)

- Data Warehouse & ETL (PostgreSQL data warehouse, ETL pipelines, data modeling)

- Backup & Disaster Recovery (automated backups, point-in-time recovery, DR plan)

- Monitoring & Alerting (application monitoring, infrastructure monitoring, alerting)

- Performance Optimization (caching with Redis, database optimization, CDN)

- DevOps & Automation (CI/CD pipelines, infrastructure as code, automated testing)

- Security Compliance (GDPR compliance, penetration testing, vulnerability scanning)

- System Administration (user management, system configuration, health checks)

2.3.2 Out Of Scope

- Physical infrastructure provisioning - handled by cloud provider

- Network infrastructure beyond application level - handled by IT operations

- End-user device management - handled by IT support

2.4 Technology Stack

2.4.1 Backend

Django REST Framework, PostgreSQL, Redis, Celery

2.4.2 Authentication

Keycloak (SSO, OIDC, SAML), Django authentication

2.4.3 Security

Let's Encrypt (SSL/TLS), Vault (secrets), bcrypt (password hashing)

2.4.4 Api Gateway

Kong or Traefik

2.4.5 Monitoring

Prometheus, Grafana, ELK Stack (Elasticsearch, Logstash, Kibana)

2.4.6 Ci Cd

GitHub Actions, Docker, Kubernetes

2.4.7 Infrastructure

AWS/Azure (cloud hosting), Terraform (IaC)

2.4.8 Backup

pg_dump, AWS S3/Azure Blob Storage

2.4.9 Data Warehouse

PostgreSQL with Timescale DB extension

2.5 Key Stakeholders

- IT Operations Team (infrastructure management, monitoring)
- Security Team (security policies, penetration testing)
- DevOps Team (CI/CD, deployments, automation)
- Compliance Team (audit logs, data protection)
- Integration Team (external system integrations)
- Database Administrators (data warehouse, ETL, performance)
- All Module Users (benefit from secure, reliable infrastructure)

3 User Requirements

3.1 Authentication Authorization

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-AUTH-SSO	authenticate using Single Sign-On (SSO)	access SAS with my corporate credentials	Must Have	SSO via Keycloak supporting OIDC and SAML. Integration with Active Directory/ LDAP. Support multiple identity providers.
FT-INFRA-AUTH-MFA	enable multi-factor authentication (MFA)	add extra security to my account	Must Have	MFA options: TOTP (Google Authenticator, Authy), SMS OTP, email OTP. Mandatory MFA for admin roles, optional for users.
FT-INFRA-AUTH-RBAC	manage user permissions via role-based access control	ensure users only access authorized features	Must Have	Roles: Super Admin, Admin, Manager, User, Viewer. Permissions at module and feature level. Permission inheritance from roles.
FT-INFRA-AUTH-SESSION	manage user sessions securely		Must Have	Session timeout (configurable,

Feature Code	I Want To	So That I Can	Priority	Notes
		prevent unauthorized access		default 30 min inactivity), concurrent session limits, session revocation, remember me option with extended timeout.
FT-INFRA-AUTH-PASSWORD	enforce strong password policies	ensure account security	Must Have	Password requirements: min 12 chars, uppercase, lowercase, number, special char. Password history (prevent reuse of last 5), password expiry (90 days), password reset flow.

3.2 Data Security

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-SEC-ENCRYPTION-REST	encrypt sensitive data at rest	protect data stored in database	Must Have	AES-256 encryption for sensitive fields (passwords, PII, financial data). Database-level encryption (PostgreSQL encryption). Key rotation policy.
			Must Have	TLS 1.3 for all communications.

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-SEC-ENCRYPTION-TRANSIT	encrypt data in transit	protect data during transmission		SSL certificates from Let's Encrypt. HTTPS enforcement, redirect HTTP to HTTPS. Certificate auto-renewal.
FT-INFRA-SEC-DATA-MASKING	mask sensitive data in non-production environments	protect real data during development and testing	Must Have	Automated data masking for dev/test environments. Preserve data format and referential integrity. Mask PII, financial data, health information.
FT-INFRA-SEC-SECRETS	manage secrets and credentials securely	avoid hardcoded credentials in code	Must Have	HashiCorp Vault for secrets management. Store API keys, database credentials, encryption keys. Secret rotation, access control, audit logging.
FT-INFRA-SEC-GDPR	comply with GDPR data protection requirements	protect member privacy	Must Have	Right to access (data export), right to erasure (data deletion), consent management, data retention policies, privacy by design.

3.3 Audit Logging

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-AUDIT-COMPREHENSIVE	log all critical system activities	maintain audit trail for compliance	Must Have	Log: user authentication, authorization changes, data modifications (CRUD), financial transactions, claims adjudication, policy changes. Include: who, what, when, where (IP), before/after values.
FT-INFRA-AUDIT-TAMPERPROOF	ensure audit logs are tamper-proof	maintain integrity for compliance	Must Have	Append-only log storage, cryptographic hashing of log entries, separate audit database with restricted access. No delete/modify permissions.
FT-INFRA-AUDIT-RETENTION	retain audit logs for required period	meet regulatory retention requirements	Must Have	7-year retention for all audit logs. Automated archival to cold storage after 1 year. Searchable archive, compliance with TIRA requirements.
FT-INFRA-AUDIT-SEARCH		investigate incidents and	Must Have	Full-text search on audit logs,

Feature Code	I Want To	So That I Can	Priority	Notes
	search and analyze audit logs	generate compliance reports		filter by user/ action/date/ module, export audit reports, audit trail visualization, anomaly detection.

3.4 Api Management

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-API-REST	expose RESTful APIs for all modules	enable integrations with external systems	Must Have	RESTful API design following best practices, JSON request/ response, standard HTTP methods (GET, POST, PUT, PATCH, DELETE), HATEOAS principles.
FT-INFRA-API-VERSIONING	version APIs to manage changes	maintain backward compatibility	Must Have	API versioning via URL path (/ api/v1/, /api/ v2/), deprecation policy (min 6 months notice), version documentation, migration guides.
FT-INFRA-API-RATE-LIMITING	rate limit API requests		Must Have	Rate limits: 100 req/min per user, 1000 req/

Feature Code	I Want To	So That I Can	Priority	Notes
		prevent abuse and ensure fair usage		min per API key. Configurable limits per client, rate limit headers in response, throttling with 429 status.
FT-INFRA-API-GATEWAY	manage APIs through API gateway	centralize authentication, routing, and monitoring	Must Have	Kong or Traefik as API gateway. API authentication (API keys, OAuth2), request/ response transformation, API analytics, load balancing.
FT-INFRA-API-DOCUMENTATION	provide interactive API documentation	enable developers to consume APIs easily	Must Have	OpenAPI/ Swagger specification, interactive API docs (Swagger UI), code examples in multiple languages, sandbox environment for testing.

3.5 Integrations

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-INT-SAGE-ERP	integrate with Sage ERP		Must Have	Sage ERP API integration for AR/AP sync,

Feature Code	I Want To	So That I Can	Priority	Notes
		sync financial data bidirectionally		invoice push, payment sync. Real-time and batch sync options, error handling and retry logic, reconciliation reports.
FT-INFRA-INT-PAYMENT-GATEWAYS	integrate with payment gateways	process online payments	Must Have	DPO Group, Flutterwave integration. Support: card payments, mobile money, bank transfers. Webhook handling for payment status, PCI DSS compliance.
FT-INFRA-INT-SMS	integrate with SMS gateway	send SMS notifications	Must Have	SMS provider: Twilio, Africa's Talking. SMS types: OTP, alerts, reminders. Delivery status tracking, SMS templates, cost tracking.
FT-INFRA-INT-EMAIL	integrate with email service	send email notifications	Must Have	Email provider: SendGrid, AWS SES. Email types: transactional, marketing. Email templates, delivery tracking, bounce handling,

Feature Code	I Want To	So That I Can	Priority	Notes
				unsubscribe management.
FT-INFRA-INT-TIRA	integrate with TIRA regulatory portal	submit regulatory returns electronically	Must Have	TIRA portal API integration (if available) or file-based submission. Generate returns in prescribed format, electronic submission, submission confirmation tracking.
FT-INFRA-INT-WHATSAPP	integrate with WhatsApp Business API	communicate via WhatsApp	Must Have	WhatsApp Business API integration, message templates approval, chatbot integration (Dialogflow/ Rasa), delivery status, media support.

3.6 Data Warehouse

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-DWH-SCHEMA	design data warehouse schema	support analytics and reporting	Must Have	Star schema design with fact and dimension tables. Fact tables: claims, premiums, policies, payments.

Feature Code	I Want To	So That I Can	Priority	Notes
				Dimension tables: time, member, product, provider, geography.
FT-INFRA-DWH-ETL	build ETL pipelines to populate data warehouse	keep analytics data current	Must Have	ETL tool: Apache Airflow or custom Python scripts. Extract from operational DB, transform (clean, aggregate, denormalize), load to DWH. Scheduled runs (hourly/daily).
FT-INFRA-DWH-MATERIALIZED-VIEWS	create materialized views for performance	speed up complex analytical queries	Must Have	Materialized views for common aggregations (monthly claims, member counts, revenue). Refresh strategy (incremental, full), query optimization, indexing.
FT-INFRA-DWH-DATA-QUALITY	ensure data quality in warehouse	trust analytics and reports	Must Have	Data validation rules, anomaly detection, data profiling, data quality dashboards. Alert on data quality issues, automated data

Feature Code	I Want To	So That I Can	Priority	Notes
				cleansing where possible.

3.7 Backup Disaster Recovery

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-BACKUP-AUTOMATED	automate database backups	protect against data loss	Must Have	Automated daily full backups, hourly incremental backups. Backup to AWS S3/ Azure Blob with versioning. Retention: daily for 30 days, weekly for 3 months, monthly for 7 years.
FT-INFRA-BACKUP-PITR	support point-in-time recovery	recover to any point in time	Must Have	PostgreSQL WAL archiving for PITR. Recovery to any point within retention window. Automated restore testing monthly.
FT-INFRA-DR-PLAN	maintain disaster recovery plan	recover from catastrophic failures	Must Have	DR plan with RTO (4 hours) and RPO (1 hour). Failover procedures, DR testing quarterly, secondary region/AZ

Feature Code	I Want To	So That I Can	Priority	Notes
				setup, runbook documentation.
FT-INFRA-BACKUP-RESTORE	restore from backups	recover data when needed	Must Have	Restore procedures for full and partial recovery. Restore to production or staging. Restore verification, restore time SLA (< 4 hours).

3.8 Monitoring Alerting

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-MON-APPLICATION	monitor application health and performance	detect and resolve issues proactively	Must Have	Prometheus for metrics collection, Grafana for visualization. Metrics: response time, error rate, throughput, queue depth. Application logs centralized in ELK.
FT-INFRA-MON-INFRASTRUCTURE	monitor infrastructure health	ensure system availability	Must Have	Monitor: CPU, memory, disk, network usage. Database monitoring (connections, query performance, locks). Redis monitoring

Feature Code	I Want To	So That I Can	Priority	Notes
				(memory, hit rate).
FT-INFRA-ALERT-CONFIG	configure alerting rules	get notified of critical issues	Must Have	Alerting via Prometheus Alertmanager. Alert channels: email, SMS, Slack, PagerDuty. Alert severity levels, escalation policies, on-call rotations.
FT-INFRA-MON-UPTIME	monitor system uptime	track availability SLA	Must Have	Uptime monitoring with UptimeRobot or Pingdom. Target: 99.9% uptime. Uptime dashboards, incident tracking, downtime reports.
FT-INFRA-MON-LOGS	centralize and analyze logs	troubleshoot issues and detect anomalies	Must Have	ELK Stack (Elasticsearch, Logstash, Kibana) for log management. Structured logging (JSON), log retention (30 days hot, 1 year warm), log search and visualization.

3.9 Performance

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-PERF-CACHING	implement caching strategy	improve response times	Must Have	Redis for caching frequently accessed data (lookups, session data, API responses). Cache invalidation strategy, cache hit rate monitoring, TTL configuration.
FT-INFRA-PERF-DB-OPTIMIZATION	optimize database performance	ensure fast query execution	Must Have	Database indexing strategy, query optimization, connection pooling (PgBouncer), query monitoring (pg_stat_statements), regular VACUUM and ANALYZE.
FT-INFRA-PERF-CDN	use CDN for static assets	improve page load times	Must Have	CloudFront or Azure CDN for static assets (images, CSS, JS). Asset optimization (minification, compression), cache headers, CDN purging.
FT-INFRA-PERF-LOAD-BALANCING	implement load balancing	distribute traffic and ensure high availability	Must Have	Application load balancer (AWS ALB, Azure Load Balancer). Health checks, session persistence, SSL termination at load balancer, auto-scaling based on load.

3.10 Devops

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-DEVOPS-CI-CD	automate CI/CD pipelines	deploy code changes quickly and reliably	Must Have	GitHub Actions for CI/CD. Pipeline stages: build, test, security scan, deploy. Automated testing (unit, integration), deployment to staging then production, rollback capability.
FT-INFRA-DEVOPS-CONTAINERS	containerize applications	ensure consistency across environments	Must Have	Docker for containerization, Docker Compose for local development. Multi-stage builds for optimization, image scanning for vulnerabilities, container registry (ECR, ACR).
FT-INFRA-DEVOPS-ORCHESTRATION	orchestrate containers with Kubernetes	manage deployments at scale	Should Have	Kubernetes for container orchestration (EKS, AKS). Deployments, services, ingress. Auto-scaling (HPA), rolling updates, health

https://www.assemble.co.tz/

Feature Code	I Want To	So That I Can	Priority	Notes
				checks, secrets management.
FT-INFRA-DEVOPS-IAC	manage infrastructure as code	version and automate infrastructure provisioning	Must Have	Terraform for infrastructure as code. Version control for IaC, automated provisioning, state management, infrastructure documentation.
FT-INFRA-DEVOPS-ENV-MANAGEMENT	manage multiple environments	separate dev, staging, and production	Must Have	Separate environments: development, staging, production. Environment parity, configuration management per environment, promotion workflow (dev → staging → prod).

3.11 Security Compliance

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-SEC-PENTEST	conduct regular penetration testing	identify and fix security vulnerabilities	Must Have	Annual penetration testing by external security firm. Vulnerability remediation plan, retest after fixes,

Feature Code	I Want To	So That I Can	Priority	Notes
				penetration test reports.
FT-INFRA-SEC-VULN-SCAN	scan for vulnerabilities continuously	detect security issues early	Must Have	Automated vulnerability scanning (OWASP ZAP, Snyk). Scan: application code, dependencies, containers, infrastructure. Critical vulnerability alerts, remediation tracking.
FT-INFRA-SEC-COMPLIANCE-SCAN	scan for compliance violations	ensure adherence to security policies	Must Have	Compliance scanning for PCI DSS, GDPR, HIPAA (if applicable). Policy enforcement, compliance dashboards, non-compliance alerts.
FT-INFRA-SEC-INCIDENT-RESPONSE	have incident response plan	respond effectively to security incidents	Must Have	Incident response plan with roles and procedures. Incident detection, containment, eradication, recovery. Post-incident review, lessons learned.

3.12 System Admin

Feature Code	I Want To	So That I Can	Priority	Notes
FT-INFRA-ADMIN-USER-MGMT	manage users and roles	control system access	Must Have	User CRUD operations, role assignment, permission management. Bulk user operations, user deactivation, user activity monitoring.
FT-INFRA-ADMIN-CONFIG	configure system settings	customize system behavior	Must Have	System configuration UI for: email settings, SMS settings, timeout settings, password policies, feature flags. Configuration validation, audit log of changes.
FT-INFRA-ADMIN-HEALTH	monitor system health	ensure system is running properly	Must Have	Health check endpoints, system status dashboard. Check: database connectivity, Redis connectivity, external API availability, disk space, memory usage.
FT-INFRA-ADMIN-MAINTENANCE	perform system maintenance	keep system optimized	Must Have	Maintenance mode toggle, database maintenance

Feature Code	I Want To	So That I Can	Priority	Notes
		Page 35 of 87		(VACUUM, ANALYZE), cache clearing, log rotation, cleanup of old data.

4 Detailed Feature Requirements

4.1 Ft Infra Auth Sso

4.1.1 Priority

Must Have

4.1.2 User Story

As a user, I want to authenticate using Single Sign-On (SSO) with my corporate credentials so that I can access SAS without managing separate passwords

4.1.3 Preconditions

Keycloak SSO configured, identity provider integrated, user account exists in identity provider

4.1.4 Postconditions

User authenticated successfully, session created, user redirected to dashboard

4.1.5 Test Cases

Id	Description	Weight
SSO-TC-001	Verify Keycloak SSO integration configured	High
SSO-TC-002	Verify OIDC authentication flow	High
SSO-TC-003	Verify SAML authentication flow	High
SSO-TC-004	Verify Active Directory/LDAP integration	High
SSO-TC-005	Verify multiple identity provider support	Medium
SSO-TC-006	Verify SSO logout functionality	High
SSO-TC-007	Verify single logout (SLO) across applications	Medium

4.2 Ft Infra Auth Mfa

4.2.1 Priority

Must Have

4.2.2 User Story

As a user, I want to enable multi-factor authentication (MFA) so that I can add extra security to my account

4.2.3 Preconditions

User account exists, MFA not yet enabled

4.2.4 Postconditions

MFA enabled, backup codes generated, MFA required on next login

4.2.5 Test Cases

Id	Description	Weight
MFA-TC-001	Verify TOTP MFA setup (Google Authenticator, Authy)	High
MFA-TC-002	Verify SMS OTP MFA	High
MFA-TC-003	Verify email OTP MFA	High
MFA-TC-004	Verify backup codes generation and usage	High
MFA-TC-005	Verify MFA mandatory for admin roles	High
MFA-TC-006	Verify MFA optional for regular users	Medium
MFA-TC-007	Verify MFA recovery process	High

4.3 Ft Infra Auth Rbac

4.3.1 Priority

Must Have

4.3.2 User Story

As a system administrator, I want to manage user permissions via role-based access control so that I can ensure users only access authorized features

4.3.3 Preconditions

Roles and permissions defined, user account exists

4.3.4 Postconditions

User assigned to role, permissions applied, access controlled

4.3.5 Test Cases

Id	Description	Weight
RBAC-TC-001	Verify Super Admin role with full access	High
RBAC-TC-002	Verify Admin role permissions	High
RBAC-TC-003	Verify Manager role permissions	High
RBAC-TC-004	Verify User role permissions	High
RBAC-TC-005	Verify Viewer role (read-only) permissions	High
RBAC-TC-006	Verify module-level permissions	High
RBAC-TC-007	Verify feature-level permissions	High
RBAC-TC-008	Verify permission inheritance from roles	Medium

4.4 Ft Infra Auth Session

4.4.1 Priority

Must Have

4.4.2 User Story

As a security officer, I want to manage user sessions securely so that I can prevent unauthorized access

4.4.3 Preconditions

User authenticated

4.4.4 Postconditions

Session created with timeout, session tracked, session revocable

4.4.5 Test Cases

Id	Description	Weight
SESSION-TC-001	Verify session timeout after inactivity (default 30 min)	High
SESSION-TC-002	Verify configurable session timeout	Medium
SESSION-TC-003	Verify concurrent session limits enforced	High
SESSION-TC-004	Verify session revocation capability	High
SESSION-TC-005	Verify 'remember me' extended session	Medium
SESSION-TC-006	Verify session hijacking protection	High

4.5 Ft Infra Auth Password

4.5.1 Priority

Must Have

4.5.2 User Story

As a security officer, I want to enforce strong password policies so that I can ensure account security

4.5.3 Preconditions

User creating/changing password

4.5.4 Postconditions

Password meets policy requirements, password hashed and stored securely

4.5.5 Test Cases

Id	Description	Weight
PASSWORD-TC-001	Verify minimum 12 characters requirement	High
PASSWORD-TC-002	Verify uppercase, lowercase, number, special char requirements	High
PASSWORD-TC-003	Verify password history (prevent reuse of last 5)	High
PASSWORD-TC-004	Verify password expiry after 90 days	High
PASSWORD-TC-005	Verify password reset flow	High
PASSWORD-TC-006	Verify bcrypt hashing for password storage	High

4.6 Ft Infra Sec Encryption Rest

4.6.1 Priority

Must Have

4.6.2 User Story

As a security officer, I want to encrypt sensitive data at rest so that I can protect data stored in database

4.6.3 Preconditions

Encryption keys configured, sensitive fields identified

4.6.4 Postconditions

Sensitive data encrypted in database, encryption keys rotated regularly

4.6.5 Test Cases

Id	Description	Weight
ENCRYPTION-REST-TC-001	Verify AES-256 encryption for passwords	High
ENCRYPTION-REST-TC-002	Verify AES-256 encryption for PII fields	High
ENCRYPTION-REST-TC-003	Verify AES-256 encryption for financial data	High
ENCRYPTION-REST-TC-004	Verify PostgreSQL database-level encryption	High
ENCRYPTION-REST-TC-005	Verify encryption key rotation policy	High
ENCRYPTION-REST-TC-006	Verify encrypted backup files	High

4.7 Ft Infra Sec Encryption Transit

4.7.1 Priority

Must Have

4.7.2 User Story

As a security officer, I want to encrypt data in transit so that I can protect data during transmission

4.7.3 Preconditions

SSL certificates configured, TLS enabled

4.7.4 Postconditions

All communications encrypted with TLS 1.3, HTTP redirected to HTTPS

4.7.5 Test Cases

Id	Description	Weight
ENCRYPTION-TRANSIT-TC-001	Verify TLS 1.3 enabled for all endpoints	High
ENCRYPTION-TRANSIT-TC-002	Verify SSL certificates from Let's Encrypt	High
ENCRYPTION-TRANSIT-TC-003	Verify HTTPS enforcement (HTTP to HTTPS redirect)	High
ENCRYPTION-TRANSIT-TC-004	Verify certificate auto-renewal	High
ENCRYPTION-TRANSIT-TC-005	Verify strong cipher suites configured	High
ENCRYPTION-TRANSIT-TC-006	Verify HSTS headers configured	Medium

4.8 Ft Infra Sec Data Masking

4.8.1 Priority

Must Have

4.8.2 User Story

As a developer, I want to mask sensitive data in non-production environments so that I can protect real data during development and testing

4.8.3 Preconditions

Non-production environment, sensitive data identified

4.8.4 Postconditions

Sensitive data masked, data format preserved, referential integrity maintained

4.8.5 Test Cases

Id	Description	Weight
DATA-MASKING-TC-001	Verify automated masking for dev environment	High
DATA-MASKING-TC-002	Verify automated masking for test environment	High
DATA-MASKING-TC-003	Verify PII data masked (names, emails, phone)	High
DATA-MASKING-TC-004	Verify financial data masked	High
DATA-MASKING-TC-005	Verify health information masked	High
DATA-MASKING-TC-006	Verify data format preserved after masking	High
DATA-MASKING-TC-007	Verify referential integrity maintained	High

4.9 Ft Infra Sec Secrets

4.9.1 Priority

Must Have

4.9.2 User Story

As a DevOps engineer, I want to manage secrets and credentials securely so that I can avoid hardcoded credentials in code

4.9.3 Preconditions

HashiCorp Vault configured, secrets defined

4.9.4 Postconditions

Secrets stored in Vault, accessed via API, audit logged

4.9.5 Test Cases

Id	Description	Weight
SECRETS-TC-001	Verify HashiCorp Vault integration	High
SECRETS-TC-002	Verify API keys stored in Vault	High
SECRETS-TC-003	Verify database credentials stored in Vault	High
SECRETS-TC-004	Verify encryption keys stored in Vault	High
SECRETS-TC-005	Verify secret rotation capability	High
SECRETS-TC-006	Verify access control to secrets	High
SECRETS-TC-007	Verify audit logging for secret access	High

4.10 Ft Infra Sec Gdpr

4.10.1 Priority

Must Have

4.10.2 User Story

As a compliance officer, I want to comply with GDPR data protection requirements so that I can protect member privacy

4.10.3 Preconditions

GDPR requirements understood, privacy policies defined

4.10.4 Postconditions

GDPR compliance controls implemented, member privacy protected

4.10.5 Test Cases

Id	Description	Weight
GDPR-TC-001	Verify right to access (data export)	High
GDPR-TC-002	Verify right to erasure (data deletion)	High
GDPR-TC-003	Verify consent management	High
GDPR-TC-004	Verify data retention policies enforced	High
GDPR-TC-005	Verify privacy by design principles	Medium
GDPR-TC-006	Verify data breach notification procedures	High

4.11 Ft Infra Audit Comprehensive

4.11.1 Priority

Must Have

4.11.2 User Story

As a compliance officer, I want to log all critical system activities so that I can maintain audit trail for compliance

4.11.3 Preconditions

Audit logging configured, activities identified

4.11.4 Postconditions

All critical activities logged with complete context

4.11.5 Test Cases

Id	Description	Weight
AUDIT-COMPREHENSIVE-TC-001	Verify user authentication logged	High
AUDIT-COMPREHENSIVE-TC-002	Verify authorization changes logged	High
AUDIT-COMPREHENSIVE-TC-003	Verify data modifications (CRUD) logged	High
AUDIT-COMPREHENSIVE-TC-004	Verify financial transactions logged	High
AUDIT-COMPREHENSIVE-TC-005	Verify claims adjudication logged	High
AUDIT-COMPREHENSIVE-TC-006	Verify policy changes logged	High
AUDIT-COMPREHENSIVE-TC-007	Verify audit log includes: who, what, when, where (IP), before/after	High

4.12 Ft Infra Audit Tamperproof

4.12.1 Priority

Must Have

4.12.2 User Story

As a compliance officer, I want to ensure audit logs are tamper-proof so that I can maintain integrity for compliance

4.12.3 Preconditions

Audit logging configured with tamper-proof mechanisms

4.12.4 Postconditions

Audit logs immutable, cryptographically hashed

4.12.5 Test Cases

Id	Description	Weight
TAMPERPROOF-TC-001	Verify append-only log storage	High
TAMPERPROOF-TC-002	Verify cryptographic hashing of log entries	High
TAMPERPROOF-TC-003	Verify separate audit database with restricted access	High
TAMPERPROOF-TC-004	Verify no delete permissions on audit logs	High
TAMPERPROOF-TC-005	Verify no modify permissions on audit logs	High
TAMPERPROOF-TC-006	Verify tamper detection mechanisms	High

4.13 Ft Infra Audit Retention

4.13.1 Priority

Must Have

4.13.2 User Story

As a compliance officer, I want to retain audit logs for required period so that I can meet regulatory retention requirements

4.13.3 Preconditions

Retention policy defined, archival configured

4.13.4 Postconditions

Audit logs retained for 7 years, archived appropriately

4.13.5 Test Cases

Id	Description	Weight
RETENTION-TC-001	Verify 7-year retention for all audit logs	High
RETENTION-TC-002	Verify automated archival to cold storage after 1 year	High
RETENTION-TC-003	Verify searchable archive	High
RETENTION-TC-004	Verify TIRA compliance for retention	High
RETENTION-TC-005	Verify secure deletion after retention period	Medium

4.14 Ft Infra Audit Search

4.14.1 Priority

Must Have

4.14.2 User Story

As an auditor, I want to search and analyze audit logs so that I can investigate incidents and generate compliance reports

4.14.3 Preconditions

Audit logs available, search interface configured

4.14.4 Postconditions

Audit logs searchable, reports generated

4.14.5 Test Cases

Id	Description	Weight
AUDIT-SEARCH-TC-001	Verify full-text search on audit logs	High
AUDIT-SEARCH-TC-002	Verify filter by user	High
AUDIT-SEARCH-TC-003	Verify filter by action	High
AUDIT-SEARCH-TC-004	Verify filter by date range	High
AUDIT-SEARCH-TC-005	Verify filter by module	High
AUDIT-SEARCH-TC-006	Verify export audit reports (CSV, PDF)	High
AUDIT-SEARCH-TC-007	Verify audit trail visualization	Medium
AUDIT-SEARCH-TC-008	Verify anomaly detection on audit logs	Medium

4.15 Ft Infra Api Rest

4.15.1 Priority

Must Have

4.15.2 User Story

As an integration developer, I want to expose RESTful APIs for all modules so that I can enable integrations with external systems

4.15.3 Preconditions

API framework configured, endpoints defined

4.15.4 Postconditions

RESTful APIs exposed, documentation available

4.15.5 Test Cases

Id	Description	Weight
API-REST-TC-001	Verify RESTful API design following best practices	High
API-REST-TC-002	Verify JSON request/response format	High
API-REST-TC-003	Verify standard HTTP methods (GET, POST, PUT, PATCH, DELETE)	High
API-REST-TC-004	Verify HATEOAS principles	Medium
API-REST-TC-005	Verify proper HTTP status codes	High
API-REST-TC-006	Verify error handling and error responses	High

4.16 Ft Infra Api Versioning

4.16.1 Priority

Must Have

4.16.2 User Story

As an API consumer, I want versioned APIs so that I can maintain backward compatibility

4.16.3 Preconditions

API versioning strategy defined

4.16.4 Postconditions

Multiple API versions supported, deprecated versions documented

4.16.5 Test Cases

Id	Description	Weight
API-VERSIONING-TC-001	Verify API versioning via URL path (/api/v1/, /api/v2/)	High
API-VERSIONING-TC-002	Verify deprecation policy (min 6 months notice)	High
API-VERSIONING-TC-003	Verify version documentation	High
API-VERSIONING-TC-004	Verify migration guides between versions	Medium
API-VERSIONING-TC-005	Verify sunset headers for deprecated versions	Medium

4.17 Ft Infra Api Rate Limiting

4.17.1 Priority

Must Have

4.17.2 User Story

As a system administrator, I want to rate limit API requests so that I can prevent abuse and ensure fair usage

4.17.3 Preconditions

Rate limiting configured

4.17.4 Postconditions

API requests rate limited, abuse prevented

4.17.5 Test Cases

Id	Description	Weight
RATE-LIMITING-TC-001		High

Id	Description	Weight
	Verify 100 req/min per user limit	
RATE-LIMITING-TC-002	Verify 1000 req/min per API key limit	High
RATE-LIMITING-TC-003	Verify configurable limits per client	Medium
RATE-LIMITING-TC-004	Verify rate limit headers in response (X-RateLimit-*)	High
RATE-LIMITING-TC-005	Verify 429 status code when rate limit exceeded	High
RATE-LIMITING-TC-006	Verify Retry-After header in 429 response	Medium

4.18 Ft Infra Api Gateway

4.18.1 Priority

Must Have

4.18.2 User Story

As a DevOps engineer, I want to manage APIs through API gateway so that I can centralize authentication, routing, and monitoring

4.18.3 Preconditions

API gateway (Kong or Traefik) configured

4.18.4 Postconditions

All APIs routed through gateway, centralized management

4.18.5 Test Cases

Id	Description	Weight
API-GATEWAY-TC-001	Verify Kong or Traefik as API gateway	High
API-GATEWAY-TC-002	Verify API authentication (API keys, OAuth2)	High
API-GATEWAY-TC-003	Verify request/response transformation	Medium
API-GATEWAY-TC-004	Verify API analytics and metrics	High
API-GATEWAY-TC-005	Verify load balancing across backends	High
API-GATEWAY-TC-006	Verify circuit breaker pattern	Medium

4.19 Ft Infra Api Documentation

4.19.1 Priority

Must Have

4.19.2 User Story

As a developer, I want interactive API documentation so that I can easily consume APIs

4.19.3 Preconditions

OpenAPI specification generated, Swagger UI configured

4.19.4 Postconditions

Interactive API documentation available, developers enabled

4.19.5 Test Cases

Id	Description	Weight
API-DOCUMENTATION-TC-001	Verify OpenAPI/Swagger specification	High
API-DOCUMENTATION-TC-002	Verify interactive API docs (Swagger UI)	High
API-DOCUMENTATION-TC-003	Verify code examples in multiple languages	Medium
API-DOCUMENTATION-TC-004	Verify sandbox environment for testing	High
API-DOCUMENTATION-TC-005	Verify auto-generation of docs from code	Medium

4.20 Ft Infra Int Sage Erp

4.20.1 Priority

Must Have

4.20.2 User Story

As a finance officer, I want to integrate with Sage ERP so that I can sync financial data bidirectionally

4.20.3 Preconditions

Sage ERP API credentials configured, integration enabled

4.20.4 Postconditions

Financial data synced between SAS and Sage ERP

4.20.5 Test Cases

Id	Description	Weight
SAGE-ERP-TC-001	Verify Sage ERP API integration	High

Id	Description	Weight
SAGE-ERP-TC-002	Verify AR/AP sync	High
SAGE-ERP-TC-003	Verify invoice push to Sage	High
SAGE-ERP-TC-004	Verify payment sync from Sage	High
SAGE-ERP-TC-005	Verify real-time sync option	High
SAGE-ERP-TC-006	Verify batch sync option	High
SAGE-ERP-TC-007	Verify error handling and retry logic	High
SAGE-ERP-TC-008	Verify reconciliation reports	High

4.21 Ft Infra Int Payment Gateways

4.21.1 Priority

Must Have

4.21.2 User Story

As a member, I want to make online payments so that I can pay premiums conveniently

4.21.3 Preconditions

Payment gateway credentials configured, integration enabled

4.21.4 Postconditions

Online payments processed, payment status updated

4.21.5 Test Cases

Id	Description	Weight
PAYMENT-GATEWAY-TC-001	Verify DPO Group integration	High
PAYMENT-GATEWAY-TC-002	Verify Flutterwave integration	High

Id	Description	Weight
PAYMENT-GATEWAY-TC-003	Verify card payments support	High
PAYMENT-GATEWAY-TC-004	Verify mobile money support	High
PAYMENT-GATEWAY-TC-005	Verify bank transfer support	High
PAYMENT-GATEWAY-TC-006	Verify webhook handling for payment status	High
PAYMENT-GATEWAY-TC-007	Verify PCI DSS compliance	High

4.22 Ft Infra Int Sms

4.22.1 Priority

Must Have

4.22.2 User Story

As a system administrator, I want to integrate with SMS gateway so that I can send SMS notifications

4.22.3 Preconditions

SMS gateway credentials configured, integration enabled

4.22.4 Postconditions

SMS notifications sent, delivery status tracked

4.22.5 Test Cases

Id	Description	Weight
SMS-TC-001	Verify Twilio or Africa's Talking integration	High
SMS-TC-002	Verify OTP SMS delivery	High

Id	Description	Weight
SMS-TC-003	Verify alert SMS delivery	High
SMS-TC-004	Verify reminder SMS delivery	High
SMS-TC-005	Verify delivery status tracking	High
SMS-TC-006	Verify SMS templates	Medium
SMS-TC-007	Verify SMS cost tracking	Medium

4.23 Ft Infra Int Email

4.23.1 Priority

Must Have

4.23.2 User Story

As a system administrator, I want to integrate with email service so that I can send email notifications

4.23.3 Preconditions

Email service credentials configured, integration enabled

4.23.4 Postconditions

Email notifications sent, delivery tracked

4.23.5 Test Cases

Id	Description	Weight
EMAIL-TC-001	Verify SendGrid or AWS SES integration	High
EMAIL-TC-002	Verify transactional email delivery	High
EMAIL-TC-003	Verify marketing email delivery	High
EMAIL-TC-004	Verify email templates	High
EMAIL-TC-005	Verify delivery tracking	High

Id	Description	Weight
EMAIL-TC-006	Verify bounce handling	High
EMAIL-TC-007	Verify unsubscribe management	High

4.24 Ft Infra Int Tira

4.24.1 Priority

Must Have

4.24.2 User Story

As a compliance officer, I want to integrate with TIRA regulatory portal so that I can submit regulatory returns electronically

4.24.3 Preconditions

TIRA portal credentials configured, return data prepared

4.24.4 Postconditions

Regulatory returns submitted to TIRA, confirmation received

4.24.5 Test Cases

Id	Description	Weight
TIRA-TC-001	Verify TIRA portal API integration (if available)	High
TIRA-TC-002	Verify file-based submission support	High
TIRA-TC-003	Verify return generation in prescribed format	High
TIRA-TC-004	Verify electronic submission	High
TIRA-TC-005	Verify submission confirmation tracking	High
TIRA-TC-006	Verify audit trail of submissions	High

4.25 Ft Infra Int Whatsapp

4.25.1 Priority

Must Have

4.25.2 User Story

As a customer service manager, I want to integrate with WhatsApp Business API so that I can communicate via WhatsApp

4.25.3 Preconditions

WhatsApp Business API configured, templates approved

4.25.4 Postconditions

WhatsApp messages sent, delivery status tracked

4.25.5 Test Cases

Id	Description	Weight
WHATSAPP-TC-001	Verify WhatsApp Business API integration	High
WHATSAPP-TC-002	Verify message templates approval process	High
WHATSAPP-TC-003	Verify chatbot integration (Dialogflow/Rasa)	High
WHATSAPP-TC-004	Verify delivery status tracking	High
WHATSAPP-TC-005	Verify media support (images, documents)	Medium
WHATSAPP-TC-006	Verify two-way messaging	High

4.26 Ft Infra Dwh Schema

4.26.1 Priority

Must Have

4.26.2 User Story

As a data analyst, I want a well-designed data warehouse schema so that I can support analytics and reporting

4.26.3 Preconditions

Data warehouse database provisioned, schema design completed

4.26.4 Postconditions

Star schema implemented with fact and dimension tables

4.26.5 Test Cases

Id	Description	Weight
DWH-SCHEMA-TC-001	Verify star schema design	High
DWH-SCHEMA-TC-002	Verify fact table: claims	High
DWH-SCHEMA-TC-003	Verify fact table: premiums	High
DWH-SCHEMA-TC-004	Verify fact table: policies	High
DWH-SCHEMA-TC-005	Verify fact table: payments	High
DWH-SCHEMA-TC-006	Verify dimension tables (time, member, product, provider, geography)	High
DWH-SCHEMA-TC-007	Verify foreign key relationships	High

4.27 Ft Infra Dwh Etl

4.27.1 Priority

Must Have

4.27.2 User Story

As a data engineer, I want ETL pipelines to populate data warehouse so that I can keep analytics data current

4.27.3 Preconditions

ETL tool configured, source and target systems connected

4.27.4 Postconditions

Data extracted, transformed, loaded to DWH on schedule

4.27.5 Test Cases

Id	Description	Weight
DWH-ETL-TC-001	Verify Apache Airflow or custom Python ETL	High
DWH-ETL-TC-002	Verify extract from operational DB	High
DWH-ETL-TC-003	Verify transform (clean, aggregate, denormalize)	High
DWH-ETL-TC-004	Verify load to DWH	High
DWH-ETL-TC-005	Verify scheduled runs (hourly/daily)	High
DWH-ETL-TC-006	Verify error handling and retry logic	High
DWH-ETL-TC-007	Verify ETL monitoring and alerting	High

4.28 Ft Infra Dwh Materialized Views

4.28.1 Priority

Must Have

4.28.2 User Story

As a data analyst, I want materialized views for performance so that I can speed up complex analytical queries

4.28.3 Preconditions

Common analytical queries identified, materialized views defined

4.28.4 Postconditions

Materialized views created, queries optimized

4.28.5 Test Cases

Id	Description	Weight
MATERIALIZED-VIEWS-TC-001	Verify materialized views for monthly claims	High
MATERIALIZED-VIEWS-TC-002	Verify materialized views for member counts	High
MATERIALIZED-VIEWS-TC-003	Verify materialized views for revenue	High
MATERIALIZED-VIEWS-TC-004	Verify incremental refresh strategy	High
MATERIALIZED-VIEWS-TC-005	Verify full refresh strategy	Medium
MATERIALIZED-VIEWS-TC-006	Verify query optimization and indexing	High

4.29 Ft Infra Dwh Data Quality

4.29.1 Priority

Must Have

4.29.2 User Story

As a data analyst, I want to ensure data quality in warehouse so that I can trust analytics and reports

4.29.3 Preconditions

Data quality rules defined, monitoring configured

4.29.4 Postconditions

Data quality monitored, issues detected and alerted

4.29.5 Test Cases

Id	Description	Weight
DATA-QUALITY-TC-001	Verify data validation rules	High
DATA-QUALITY-TC-002	Verify anomaly detection	High
DATA-QUALITY-TC-003	Verify data profiling	Medium
DATA-QUALITY-TC-004	Verify data quality dashboards	High
DATA-QUALITY-TC-005	Verify alerts on data quality issues	High
DATA-QUALITY-TC-006	Verify automated data cleansing	Medium

4.30 Ft Infra Backup Automated

4.30.1 Priority

Must Have

4.30.2 User Story

As a database administrator, I want automated database backups so that I can protect against data loss

4.30.3 Preconditions

Backup system configured, backup schedule defined

4.30.4 Postconditions

Database backed up automatically, backups stored securely

4.30.5 Test Cases

Id	Description	Weight
BACKUP-AUTOMATED-TC-001	Verify automated daily full backups	High
BACKUP-AUTOMATED-TC-002	Verify automated hourly incremental backups	High
BACKUP-AUTOMATED-TC-003	Verify backup to AWS S3/ Azure Blob	High
BACKUP-AUTOMATED-TC-004	Verify backup versioning	High
BACKUP-AUTOMATED-TC-005	Verify retention: daily for 30 days	High
BACKUP-AUTOMATED-TC-006	Verify retention: weekly for 3 months	High
BACKUP-AUTOMATED-TC-007	Verify retention: monthly for 7 years	High

4.31 Ft Infra Backup Pitr

4.31.1 Priority

Must Have

4.31.2 User Story

As a database administrator, I want point-in-time recovery so that I can recover to any point in time

4.31.3 Preconditions

PostgreSQL WAL archiving configured

4.31.4 Postconditions

Point-in-time recovery capability available

4.31.5 Test Cases

Id	Description	Weight
PITR-TC-001	Verify PostgreSQL WAL archiving enabled	High
PITR-TC-002	Verify recovery to any point within retention window	High
PITR-TC-003	Verify automated restore testing monthly	High
PITR-TC-004	Verify WAL backup to remote storage	High
PITR-TC-005	Verify PITR documentation and procedures	Medium

4.32 Ft Infra Dr Plan

4.32.1 Priority

Must Have

4.32.2 User Story

As a CTO, I want a disaster recovery plan so that I can recover from catastrophic failures

4.32.3 Preconditions

DR plan documented, DR procedures defined

4.32.4 Postconditions

DR plan in place, tested regularly

4.32.5 Test Cases

Id	Description	Weight
DR-PLAN-TC-001	Verify DR plan with RTO (4 hours)	High
DR-PLAN-TC-002	Verify DR plan with RPO (1 hour)	High
DR-PLAN-TC-003	Verify failover procedures documented	High
DR-PLAN-TC-004	Verify DR testing quarterly	High
DR-PLAN-TC-005	Verify secondary region/AZ setup	High
DR-PLAN-TC-006	Verify runbook documentation	High

4.33 Ft Infra Backup Restore

4.33.1 Priority

Must Have

4.33.2 User Story

As a database administrator, I want to restore from backups so that I can recover data when needed

4.33.3 Preconditions

Backups available, restore procedures defined

4.33.4 Postconditions

Data restored successfully

4.33.5 Test Cases

Id	Description	Weight
BACKUP-RESTORE-TC-001	Verify restore procedures for full recovery	High
BACKUP-RESTORE-TC-002		High

Id	Description	Weight
	Verify restore procedures for partial recovery	
BACKUP-RESTORE-TC-003	Verify restore to production	High
BACKUP-RESTORE-TC-004	Verify restore to staging	High
BACKUP-RESTORE-TC-005	Verify restore verification	High
BACKUP-RESTORE-TC-006	Verify restore time SLA (< 4 hours)	High

4.34 Ft Infra Mon Application

4.34.1 Priority

Must Have

4.34.2 User Story

As a DevOps engineer, I want to monitor application health and performance so that I can detect and resolve issues proactively

4.34.3 Preconditions

Prometheus and Grafana configured, metrics collected

4.34.4 Postconditions

Application metrics monitored, dashboards available

4.34.5 Test Cases

Id	Description	Weight
MON-APPLICATION-TC-001	Verify Prometheus for metrics collection	High
MON-APPLICATION-TC-002	Verify Grafana for visualization	High

Id	Description	Weight
MON-APPLICATION-TC-003	Verify response time metrics	High
MON-APPLICATION-TC-004	Verify error rate metrics	High
MON-APPLICATION-TC-005	Verify throughput metrics	High
MON-APPLICATION-TC-006	Verify queue depth metrics	High
MON-APPLICATION-TC-007	Verify application logs centralized in ELK	High

4.35 Ft Infra Mon Infrastructure

4.35.1 Priority

Must Have

4.35.2 User Story

As a DevOps engineer, I want to monitor infrastructure health so that I can ensure system availability

4.35.3 Preconditions

Infrastructure monitoring configured

4.35.4 Postconditions

Infrastructure metrics monitored, issues detected

4.35.5 Test Cases

Id	Description	Weight
MON-INFRASTRUCTURE-TC-001	Verify CPU usage monitoring	High
MON-INFRASTRUCTURE-TC-002	Verify memory usage monitoring	High
MON-INFRASTRUCTURE-TC-003	Verify disk usage monitoring	High

Id	Description	Weight
MON-INFRASTRUCTURE-TC-004	Verify network usage monitoring	High
MON-INFRASTRUCTURE-TC-005	Verify database connections monitoring	High
MON-INFRASTRUCTURE-TC-006	Verify database query performance monitoring	High
MON-INFRASTRUCTURE-TC-007	Verify Redis memory and hit rate monitoring	High

4.36 Ft Infra Alert Config

4.36.1 Priority

Must Have

4.36.2 User Story

As a DevOps engineer, I want to configure alerting rules so that I can get notified of critical issues

4.36.3 Preconditions

Alerting system configured, rules defined

4.36.4 Postconditions

Alerts triggered on critical issues, team notified

4.36.5 Test Cases

Id	Description	Weight
ALERT-CONFIG-TC-001	Verify Prometheus Alertmanager configured	High
ALERT-CONFIG-TC-002	Verify email alert channel	High
ALERT-CONFIG-TC-003	Verify SMS alert channel	High
ALERT-CONFIG-TC-004	Verify Slack alert channel	High

Id	Description	Weight
ALERT-CONFIG-TC-005	Verify PagerDuty integration	Medium
ALERT-CONFIG-TC-006	Verify alert severity levels	High
ALERT-CONFIG-TC-007	Verify escalation policies	High
ALERT-CONFIG-TC-008	Verify on-call rotations	Medium

4.37 Ft Infra Mon Uptime

4.37.1 Priority

Must Have

4.37.2 User Story

As a CTO, I want to monitor system uptime so that I can track availability SLA

4.37.3 Preconditions

Uptime monitoring configured, SLA target defined

4.37.4 Postconditions

Uptime monitored, SLA tracked

4.37.5 Test Cases

Id	Description	Weight
MON-UPTIME-TC-001	Verify uptime monitoring with UptimeRobot or Pingdom	High
MON-UPTIME-TC-002	Verify 99.9% uptime target	High
MON-UPTIME-TC-003	Verify uptime dashboards	High
MON-UPTIME-TC-004	Verify incident tracking	High
MON-UPTIME-TC-005	Verify downtime reports	High

4.38 Ft Infra Mon Logs

4.38.1 Priority

Must Have

4.38.2 User Story

As a DevOps engineer, I want to centralize and analyze logs so that I can troubleshoot issues and detect anomalies

4.38.3 Preconditions

ELK Stack configured, logs collected

4.38.4 Postconditions

Logs centralized, searchable, and analyzed

4.38.5 Test Cases

Id	Description	Weight
MON-LOGS-TC-001	Verify ELK Stack (Elasticsearch, Logstash, Kibana)	High
MON-LOGS-TC-002	Verify structured logging (JSON)	High
MON-LOGS-TC-003	Verify log retention: 30 days hot	High
MON-LOGS-TC-004	Verify log retention: 1 year warm	High
MON-LOGS-TC-005	Verify log search functionality	High
MON-LOGS-TC-006	Verify log visualization in Kibana	High
MON-LOGS-TC-007	Verify log anomaly detection	Medium

4.39 Ft Infra Perf Caching

4.39.1 Priority

Must Have

4.39.2 User Story

As a developer, I want to implement caching strategy so that I can improve response times

4.39.3 Preconditions

Redis configured, caching strategy defined

4.39.4 Postconditions

Frequently accessed data cached, response times improved

4.39.5 Test Cases

Id	Description	Weight
PERF-CACHING-TC-001	Verify Redis for caching	High
PERF-CACHING-TC-002	Verify lookups cached	High
PERF-CACHING-TC-003	Verify session data cached	High
PERF-CACHING-TC-004	Verify API responses cached	High
PERF-CACHING-TC-005	Verify cache invalidation strategy	High
PERF-CACHING-TC-006	Verify cache hit rate monitoring	High
PERF-CACHING-TC-007	Verify TTL configuration	Medium

4.40 Ft Infra Perf Db Optimization

4.40.1 Priority

Must Have

4.40.2 User Story

As a database administrator, I want to optimize database performance so that I can ensure fast query execution

4.40.3 Preconditions

Database optimization strategy defined

4.40.4 Postconditions

Database performance optimized, queries fast

4.40.5 Test Cases

Id	Description	Weight
PERF-DB-TC-001	Verify database indexing strategy	High
PERF-DB-TC-002	Verify query optimization	High
PERF-DB-TC-003	Verify connection pooling (PgBouncer)	High
PERF-DB-TC-004	Verify query monitoring (pg_stat_statements)	High
PERF-DB-TC-005	Verify regular VACUUM operations	High
PERF-DB-TC-006	Verify regular ANALYZE operations	High

4.41 Ft Infra Perf Cdn

4.41.1 Priority

Must Have

4.41.2 User Story

As a user, I want fast page load times so that I can have a smooth experience

4.41.3 Preconditions

CDN configured, static assets identified

4.41.4 Postconditions

Static assets served via CDN, page load times improved

4.41.5 Test Cases

Id	Description	Weight
PERF-CDN-TC-001	Verify CloudFront or Azure CDN configured	High
PERF-CDN-TC-002	Verify static assets (images, CSS, JS) served via CDN	High
PERF-CDN-TC-003	Verify asset optimization (minification, compression)	High
PERF-CDN-TC-004	Verify cache headers configured	High
PERF-CDN-TC-005	Verify CDN purging capability	Medium

4.42 Ft Infra Perf Load Balancing

4.42.1 Priority

Must Have

4.42.2 User Story

As a DevOps engineer, I want load balancing so that I can distribute traffic and ensure high availability

4.42.3 Preconditions

Load balancer configured, multiple backend instances

4.42.4 Postconditions

Traffic distributed, high availability ensured

4.42.5 Test Cases

Id	Description	Weight
PERF-LB-TC-001	Verify AWS ALB or Azure Load Balancer configured	High
PERF-LB-TC-002	Verify health checks configured	High
PERF-LB-TC-003	Verify session persistence	High
PERF-LB-TC-004	Verify SSL termination at load balancer	High
PERF-LB-TC-005	Verify auto-scaling based on load	High
PERF-LB-TC-006	Verify traffic distribution algorithms	Medium

4.43 Ft Infra Devops Ci Cd

4.43.1 Priority

Must Have

4.43.2 User Story

As a developer, I want automated CI/CD pipelines so that I can deploy code changes quickly and reliably

4.43.3 Preconditions

CI/CD tool configured, pipeline defined

4.43.4 Postconditions

Code changes deployed automatically through pipeline

4.43.5 Test Cases

Id	Description	Weight
DEVOPS-CICD-TC-001	Verify GitHub Actions for CI/CD	High
DEVOPS-CICD-TC-002	Verify pipeline stage: build	High
DEVOPS-CICD-TC-003	Verify pipeline stage: test	High
DEVOPS-CICD-TC-004	Verify pipeline stage: security scan	High
DEVOPS-CICD-TC-005	Verify pipeline stage: deploy	High
DEVOPS-CICD-TC-006	Verify automated testing (unit, integration)	High
DEVOPS-CICD-TC-007	Verify deployment to staging then production	High
DEVOPS-CICD-TC-008	Verify rollback capability	High

4.44 Ft Infra Devops Containers

4.44.1 Priority

Must Have

4.44.2 User Story

As a developer, I want containerized applications so that I can ensure consistency across environments

4.44.3 Preconditions

Docker configured, Dockerfile created

4.44.4 Postconditions

Application containerized, images built and pushed

4.44.5 Test Cases

Id	Description	Weight
DEVOPS-CONTAINERS-TC-001	Verify Docker for containerization	High
DEVOPS-CONTAINERS-TC-002	Verify Docker Compose for local development	High
DEVOPS-CONTAINERS-TC-003	Verify multi-stage builds for optimization	High
DEVOPS-CONTAINERS-TC-004	Verify image scanning for vulnerabilities	High
DEVOPS-CONTAINERS-TC-005	Verify container registry (ECR, ACR)	High
DEVOPS-CONTAINERS-TC-006	Verify image tagging strategy	Medium

4.45 Ft Infra Devops Orchestration

4.45.1 Priority

Should Have

4.45.2 User Story

As a DevOps engineer, I want to orchestrate containers with Kubernetes so that I can manage deployments at scale

4.45.3 Preconditions

Kubernetes cluster provisioned, manifests created

4.45.4 Postconditions

Containers orchestrated, deployments managed

4.45.5 Test Cases

Id	Description	Weight
DEVOPS-K8S-TC-001	Verify Kubernetes cluster (EKS, AKS)	High
DEVOPS-K8S-TC-002	Verify deployments configured	High
DEVOPS-K8S-TC-003	Verify services configured	High
DEVOPS-K8S-TC-004	Verify ingress configured	High
DEVOPS-K8S-TC-005	Verify auto-scaling (HPA)	High
DEVOPS-K8S-TC-006	Verify rolling updates	High
DEVOPS-K8S-TC-007	Verify health checks (liveness, readiness)	High
DEVOPS-K8S-TC-008	Verify secrets management in Kubernetes	High

4.46 Ft Infra Devops Iac

4.46.1 Priority

Must Have

4.46.2 User Story

As a DevOps engineer, I want to manage infrastructure as code so that I can version and automate infrastructure provisioning

4.46.3 Preconditions

Terraform installed, infrastructure defined

4.46.4 Postconditions

Infrastructure provisioned via Terraform, versioned in Git

4.46.5 Test Cases

Id	Description	Weight
DEVOPS-IAC-TC-001	Verify Terraform for infrastructure as code	High
DEVOPS-IAC-TC-002	Verify version control for IaC in Git	High
DEVOPS-IAC-TC-003	Verify automated provisioning	High
DEVOPS-IAC-TC-004	Verify state management (remote state)	High
DEVOPS-IAC-TC-005	Verify infrastructure documentation	Medium

4.47 Ft Infra Devops Env Management

4.47.1 Priority

Must Have

4.47.2 User Story

As a developer, I want multiple environments so that I can separate dev, staging, and production

4.47.3 Preconditions

Multiple environments provisioned

4.47.4 Postconditions

Environments isolated, promotion workflow established

4.47.5 Test Cases

Id	Description	Weight
DEVOPS-ENV-TC-001	Verify development environment	High
DEVOPS-ENV-TC-002	Verify staging environment	High
DEVOPS-ENV-TC-003		High

https://www.assemble.co.tz/

Id	Description	Weight
	Verify production environment	
DEVOPS-ENV-TC-004	Verify environment parity	High
DEVOPS-ENV-TC-005	Verify configuration management per environment	High
DEVOPS-ENV-TC-006	Verify promotion workflow (dev → staging → prod)	High

4.48 Ft Infra Sec Pentest

4.48.1 Priority

Must Have

4.48.2 User Story

As a security officer, I want regular penetration testing so that I can identify and fix security vulnerabilities

4.48.3 Preconditions

Penetration testing schedule defined, security firm engaged

4.48.4 Postconditions

Vulnerabilities identified, remediation plan created

4.48.5 Test Cases

Id	Description	Weight
PENTEST-TC-001	Verify annual penetration testing by external firm	High
PENTEST-TC-002	Verify vulnerability remediation plan	High
PENTEST-TC-003	Verify retest after fixes	High

https://www.assemble.co.tz/

Id	Description	Weight
PENTEST-TC-004	Verify penetration test reports	High
PENTEST-TC-005	Verify executive summary for management	Medium

4.49 Ft Infra Sec Vuln Scan

4.49.1 Priority

Must Have

4.49.2 User Story

As a security officer, I want continuous vulnerability scanning so that I can detect security issues early

4.49.3 Preconditions

Vulnerability scanning tools configured

4.49.4 Postconditions

Vulnerabilities detected, alerts triggered

4.49.5 Test Cases

Id	Description	Weight
VULN-SCAN-TC-001	Verify automated vulnerability scanning (OWASP ZAP, Snyk)	High
VULN-SCAN-TC-002	Verify application code scanning	High
VULN-SCAN-TC-003	Verify dependency scanning	High
VULN-SCAN-TC-004	Verify container scanning	High
VULN-SCAN-TC-005	Verify infrastructure scanning	High
VULN-SCAN-TC-006		High

Id	Description	Weight
	Verify critical vulnerability alerts	
VULN-SCAN-TC-007	Verify remediation tracking	High

4.50 Ft Infra Sec Compliance Scan

4.50.1 Priority

Must Have

4.50.2 User Story

As a compliance officer, I want to scan for compliance violations so that I can ensure adherence to security policies

4.50.3 Preconditions

Compliance scanning tools configured, policies defined

4.50.4 Postconditions

Compliance violations detected, reports generated

4.50.5 Test Cases

Id	Description	Weight
COMPLIANCE-SCAN-TC-001	Verify PCI DSS compliance scanning	High
COMPLIANCE-SCAN-TC-002	Verify GDPR compliance scanning	High
COMPLIANCE-SCAN-TC-003	Verify HIPAA compliance scanning (if applicable)	Medium
COMPLIANCE-SCAN-TC-004	Verify policy enforcement	High
COMPLIANCE-SCAN-TC-005	Verify compliance dashboards	High

Id	Description	Weight
COMPLIANCE-SCAN-TC-006	Verify non-compliance alerts	High

4.51 Ft Infra Sec Incident Response

4.51.1 Priority

Must Have

4.51.2 User Story

As a security officer, I want an incident response plan so that I can respond effectively to security incidents

4.51.3 Preconditions

Incident response plan documented, team trained

4.51.4 Postconditions

Incidents responded to effectively, lessons learned documented

4.51.5 Test Cases

Id	Description	Weight
INCIDENT-RESPONSE-TC-001	Verify incident response plan with roles and procedures	High
INCIDENT-RESPONSE-TC-002	Verify incident detection mechanisms	High
INCIDENT-RESPONSE-TC-003	Verify incident containment procedures	High
INCIDENT-RESPONSE-TC-004	Verify incident eradication procedures	High
INCIDENT-RESPONSE-TC-005	Verify incident recovery procedures	High

Id	Description	Weight
INCIDENT-RESPONSE-TC-006	Verify post-incident review	High
INCIDENT-RESPONSE-TC-007	Verify lessons learned documentation	Medium

4.52 Ft Infra Admin User Mgmt

4.52.1 Priority

Must Have

4.52.2 User Story

As a system administrator, I want to manage users and roles so that I can control system access

4.52.3 Preconditions

User management interface available

4.52.4 Postconditions

Users managed, access controlled

4.52.5 Test Cases

Id	Description	Weight
ADMIN-USER-MGMT-TC-001	Verify user CRUD operations	High
ADMIN-USER-MGMT-TC-002	Verify role assignment	High
ADMIN-USER-MGMT-TC-003	Verify permission management	High
ADMIN-USER-MGMT-TC-004	Verify bulk user operations	Medium
ADMIN-USER-MGMT-TC-005	Verify user deactivation	High
ADMIN-USER-MGMT-TC-006	Verify user activity monitoring	Medium

4.53 Ft Infra Admin Config

4.53.1 Priority

Must Have

4.53.2 User Story

As a system administrator, I want to configure system settings so that I can customize system behavior

4.53.3 Preconditions

System configuration interface available

4.53.4 Postconditions

System configured, settings applied

4.53.5 Test Cases

Id	Description	Weight
ADMIN-CONFIG-TC-001	Verify email settings configuration UI	High
ADMIN-CONFIG-TC-002	Verify SMS settings configuration UI	High
ADMIN-CONFIG-TC-003	Verify timeout settings configuration	High
ADMIN-CONFIG-TC-004	Verify password policies configuration	High
ADMIN-CONFIG-TC-005	Verify feature flags configuration	Medium
ADMIN-CONFIG-TC-006	Verify configuration validation	High
ADMIN-CONFIG-TC-007	Verify audit log of configuration changes	High

4.54 Ft Infra Admin Health

4.54.1 Priority

Must Have

4.54.2 User Story

As a system administrator, I want to monitor system health so that I can ensure system is running properly

4.54.3 Preconditions

Health check endpoints configured

4.54.4 Postconditions

System health monitored, status visible

4.54.5 Test Cases

Id	Description	Weight
ADMIN-HEALTH-TC-001	Verify health check endpoints	High
ADMIN-HEALTH-TC-002	Verify system status dashboard	High
ADMIN-HEALTH-TC-003	Verify database connectivity check	High
ADMIN-HEALTH-TC-004	Verify Redis connectivity check	High
ADMIN-HEALTH-TC-005	Verify external API availability check	High
ADMIN-HEALTH-TC-006	Verify disk space monitoring	High
ADMIN-HEALTH-TC-007	Verify memory usage monitoring	High

4.55 Ft Infra Admin Maintenance

4.55.1 Priority

Must Have

4.55.2 User Story

As a system administrator, I want to perform system maintenance so that I can keep system optimized

4.55.3 Preconditions

Maintenance procedures defined

4.55.4 Postconditions

System maintained, performance optimized

4.55.5 Test Cases

Id	Description	Weight
ADMIN-MAINTENANCE-TC-001	Verify maintenance mode toggle	High
ADMIN-MAINTENANCE-TC-002	Verify database maintenance (VACUUM, ANALYZE)	High
ADMIN-MAINTENANCE-TC-003	Verify cache clearing	High
ADMIN-MAINTENANCE-TC-004	Verify log rotation	High
ADMIN-MAINTENANCE-TC-005	Verify cleanup of old data	High
ADMIN-MAINTENANCE-TC-006	Verify maintenance scheduling	Medium

https://www.assemble.co.tz/