Software Requirements Specification (SRS)

AI-Driven Patient-Centric Healthcare Platform

Version: 1.0
Date: January 2025
Document Type: Software Requirements Specification

Table of Contents

---

1. Introduction

1.1 Purpose

This document specifies the requirements for an AI-driven, patient-centric healthcare platform that addresses fragmented healthcare delivery in the USA. The platform empowers patients with complete control over their medical records while providing intelligent treatment recommendations and cost transparency.

1.2 Scope

The platform encompasses:

- **Patient Data Management**: Centralized, patient-controlled medical records
- **Multi-Agent AI System**: CrewAI-orchestrated intelligent healthcare agents
- **Knowledge Graph Integration**: Semantic medical knowledge representation
- **Treatment Optimization**: AI-driven treatment planning and cost analysis
- **Provider Coordination**: Seamless care coordination across specialists

1.3 Platform Vision

"Of the Patient, To the Patient, By the Patient"

Build a democratic healthcare platform that:

- Centralizes all patient medical records under patient control
- Democratizes access to best-in-class medical expertise
- Provides transparent treatment costs and options
- Optimizes treatment plans through AI-driven analysis
- Coordinates care across multiple specialists seamlessly

---

2. System Overview

2.1 Core Problem Statement

- **EHR Fragmentation**: Patient medical records scattered across multiple providers
- **Information Silos**: Doctors lack access to complete patient history
- **Treatment Inconsistency**: Varied treatment approaches without coordination
- **Cost Opacity**: Patients cannot access transparent pricing
- **Patient Disempowerment**: Limited control over medical data

2.2 Solution Architecture

The platform consists of five core layers:

1. **Patient Interface Layer**: Mobile/web applications with voice interface
2. **AI Agent Orchestration Layer**: CrewAI multi-agent coordination
3. **Knowledge Management Layer**: Medical knowledge graphs and ontologies
4. **Data Integration Layer**: EHR systems and external medical data sources
5. **Infrastructure Layer**: Cloud-native, scalable computing resources

---

3. Functional Requirements

3.1 Patient Record Management System

3.1.1 Unified Medical Record Repository

REQ-001: The system SHALL implement a knowledge graph-based medical record repository using Neo4j/Amazon Neptune.

REQ-002: The system SHALL integrate medical ontologies including:

- SNOMED CT for clinical terminology
- ICD-11 for disease classification
- LOINC for laboratory data
- RxNorm for medication terminology

REQ-003: The system SHALL implement GraphRAG (Retrieval-Augmented Generation) for contextual medical information retrieval with sub-2 second response times.

REQ-004: The system SHALL implement PathRAG for treatment pathway discovery using knowledge graph reasoning.

REQ-005: The system SHALL provide FHIR R4 compliant data standardization with ontological mapping.

3.1.2 Patient Data Ownership Controls

REQ-006: The system SHALL implement ontology-driven permission management allowing patients to define access rights using medical domain ontologies.

REQ-007: The system SHALL provide semantic data portability with preserved ontological relationships.

REQ-008: The system SHALL implement AI-powered privacy controls using machine learning models for intelligent data anonymization.

REQ-009: The system SHALL maintain blockchain-based immutable audit trails with cryptographic knowledge graph validation.

3.2 Multi-Agent AI System

3.2.1 CrewAI Agent Orchestration

REQ-010: The system SHALL implement a Chief Medical AI Coordinator as a meta-agent orchestrating all medical agents.

REQ-011: The system SHALL implement specialized medical agent teams:

- **Diagnostic Team**: Radiology AI, Pathology AI, Clinical Assessment AI
- **Treatment Team**: Pharmacology AI, Surgery Planning AI, Therapy Optimization AI
- **Coordination Team**: Scheduling AI, Resource Management AI, Communication AI
- **Research Team**: Literature Mining AI, Clinical Trial Matching AI, Evidence Synthesis AI

3.2.2 Diagnostic Analysis Agent

REQ-012: The system SHALL implement computer vision ML models:

- CNN architectures (ResNet-50, EfficientNet, Vision Transformers)
- Medical image segmentation (U-Net, Mask R-CNN)
- Multi-modal learning fusion of imaging, lab results, and clinical notes

REQ-013: The system SHALL implement medical NLP capabilities:

- BioBERT/ClinicalBERT for medical text understanding
- Named Entity Recognition using SNOMED CT ontologies
- Relation extraction for symptoms, conditions, and treatments

REQ-014: The system SHALL achieve >95% diagnostic accuracy across medical specialties.

3.2.3 Treatment Planning Agent

REQ-015: The system SHALL implement reinforcement learning models:

- Multi-armed bandit algorithms for treatment optimization
- Causal machine learning for treatment effect estimation
- Graph neural networks for treatment pathway prediction

REQ-016: The system SHALL implement PathRAG for treatment planning:

- Treatment pathway discovery using knowledge graph paths
- Clinical guideline integration via RAG systems
- Personalized medicine with genetic ontology integration

REQ-017: The system SHALL achieve 20% improvement in treatment efficacy predictions.

3.2.4 Cost Analysis Agent

REQ-018: The system SHALL implement predictive ML models:

- LSTM/GRU networks for healthcare cost forecasting
- Reinforcement learning for insurance negotiation strategies
- Anomaly detection for fraudulent billing

REQ-019: The system SHALL implement economic knowledge graphs modeling relationships between procedures, costs, and outcomes.

3.2.5 Care Coordination Agent

REQ-020: The system SHALL implement workflow optimization:

- Process mining for optimal care pathway discovery
- Temporal knowledge graphs for time-dependent patient care

- Multi-objective optimization balancing cost, quality, and preferences

REQ-021: The system SHALL implement MCP (Model Context Protocol) for real-time communication between care team members.

3.2.6 Patient Advocacy Agent

REQ-022: The system SHALL implement preference learning:

- Multi-criteria decision analysis for patient preference modeling
- Fairness-aware ML ensuring equitable treatment recommendations
- Explainable AI generating human-interpretable explanations

REQ-023: The system SHALL implement medical ethics ontology encoding principles of autonomy, beneficence, and justice.

3.2.7 Research Intelligence Agent

REQ-024: The system SHALL implement scientific GraphRAG for real-time medical research integration.

REQ-025: The system SHALL implement automated evidence synthesis using NLP and knowledge graphs.

3.3 Specialist Matching and Consultation System

3.3.1 Expert Network Integration

REQ-026: The system SHALL implement doctor expertise ontology with semantic representation of physician specializations.

REQ-027: The system SHALL use graph neural networks for optimal provider-patient matching.

REQ-028: The system SHALL provide virtual consultation platform with real-time AI assistance.

3.3.2 Quality Scoring System

REQ-029: The system SHALL implement ML models for provider performance prediction.

REQ-030: The system SHALL perform patient feedback sentiment analysis using NLP.

3.4 Transparent Pricing and Cost Management

3.4.1 Dynamic Pricing Engine

REQ-031: The system SHALL implement ensemble ML models for real-time cost estimation.

REQ-032: The system SHALL use reinforcement learning for optimal insurance negotiation strategies.

REQ-033: The system SHALL provide alternative treatment cost comparison using economic knowledge graphs.

---

4. Non-Functional Requirements

4.1 Performance Requirements

NFR-001: The system SHALL provide sub-2 second response times for AI recommendations.

NFR-002: The system SHALL maintain 99.9% uptime SLA.

NFR-003: The system SHALL support 1M+ concurrent users.

NFR-004: The system SHALL provide real-time data synchronization across healthcare providers.

4.2 Scalability Requirements

NFR-005: The system SHALL implement microservices-based architecture.

NFR-006: The system SHALL provide auto-scaling capabilities.

NFR-007: The system SHALL support multi-region deployment.

NFR-008: The system SHALL integrate CDN for global access.

4.3 Reliability Requirements

NFR-009: The system SHALL implement multi-AZ deployment with automatic failover.

NFR-010: The system SHALL provide backup and disaster recovery with RTO < 1 hour.

NFR-011: The system SHALL implement circuit breakers for external service dependencies.

4.4 Usability Requirements

NFR-012: The system SHALL provide intuitive dashboard design for patients.

NFR-013: The system SHALL comply with WCAG 2.1 accessibility standards.

NFR-014: The system SHALL support multiple languages and cultural contexts.

---

5. AI/ML System Requirements

5.1 Knowledge Graph and Ontology Infrastructure

5.1.1 Multi-Scale Knowledge Representation

AI-001: The system SHALL implement knowledge representation at multiple scales:

- **Molecular Level**: Protein interactions, genetic variants, drug mechanisms

- **Cellular Level**: Cell types, biomarkers, disease pathways
- **Organ Level**: Anatomical structures, physiological processes
- **Patient Level**: Demographics, medical history, treatment responses
- **Population Level**: Epidemiological patterns, public health data

5.1.2 Ontology Integration

AI-002: The system SHALL integrate multiple medical ontologies:

- UMLS (Unified Medical Language System) as central ontology hub
- Gene Ontology (GO) for genetic annotations
- Human Phenotype Ontology (HPO) for phenotype descriptions
- ChEBI for chemical and drug ontologies

AI-003: The system SHALL implement automated cross-ontology mapping and reconciliation.

5.1.3 Dynamic Knowledge Graph Updates

AI-004: The system SHALL implement real-time literature integration using NLP pipelines.

AI-005: The system SHALL extract new relationships from patient data using graph neural networks.

AI-006: The system SHALL implement federated knowledge learning across healthcare institutions.

5.2 Advanced RAG Implementation

5.2.1 Multi-Modal RAG Systems

AI-007: The system SHALL implement text-image-graph RAG combining:

- Medical literature retrieval
- Medical image retrieval for comparative analysis
- Knowledge graph substructure retrieval

5.2.2 Hierarchical RAG Architecture

AI-008: The system SHALL implement coarse-to-fine retrieval with initial broad retrieval followed by fine-grained selection.

AI-009: The system SHALL implement context-aware retrieval adjusting based on patient context and query intent.

5.2.3 PathRAG for Clinical Decision Support

AI-010: The system SHALL implement evidence-based pathway retrieval finding optimal treatment paths.

AI-011: The system SHALL implement outcome-guided path selection with best predicted outcomes.

AI-012: The system SHALL implement causal chain discovery identifying causal sequences in medical knowledge graphs.

5.3 Model Context Protocol (MCP) Implementation

5.3.1 Medical Context Standardization

AI-013: The system SHALL implement standardized encoding of:

- Patient clinical status
- Ongoing treatments and temporal relationships
- Healthcare provider capabilities and constraints
- Hospital resources and environmental factors

5.3.2 Context Sharing Protocol

AI-014: The system SHALL implement secure, encrypted context transfer between systems.

AI-015: The system SHALL implement context versioning tracking changes over time.

AI-016: The system SHALL implement efficient context compression for real-time sharing.

5.3.3 Real-Time Context Updates

AI-017: The system SHALL implement event-driven context updates based on clinical events.

AI-018: The system SHALL implement conflict detection for inconsistencies from different sources.

AI-019: The system SHALL implement context reconciliation merging updates from multiple providers.

5.4 Federated Learning and Privacy-Preserving ML

5.4.1 Federated Learning Framework

AI-020: The system SHALL implement healthcare-specific federated learning using NVIDIA FLARE.

AI-021: The system SHALL implement cross-institutional learning without data sharing.

AI-022: The system SHALL implement continual learning preventing catastrophic forgetting.

5.4.2 Privacy-Preserving Technologies

AI-023: The system SHALL implement differential privacy using Opacus/TensorFlow Privacy.

AI-024: The system SHALL implement homomorphic encryption for privacy-preserving computations.

AI-025: The system SHALL ensure zero patient data leakage in federated systems.

5.5 Agent Learning and Adaptation

5.5.1 Multi-Agent Reinforcement Learning

AI-026: The system SHALL implement agents learning to collaborate more effectively over time.

AI-027: The system SHALL implement curriculum learning with progressive complexity increase.

AI-028: The system SHALL implement meta-learning for rapid adaptation to new medical domains.

5.5.2 Agent Performance Optimization

AI-029: The system SHALL achieve >90% successful task completion rate for agent collaboration.

AI-030: The system SHALL achieve >95% clinically appropriate recommendations.

AI-031: The system SHALL maintain <1% unresolved conflicts between agents.

---

6. Integration Requirements

6.1 Healthcare System Integration

6.1.1 EHR System Integration

INT-001: The system SHALL integrate with major EHR systems:

- Epic (FHIR API)
- Cerner (FHIR API)
- Allscripts (HL7)
- Meditech (custom adapters)

INT-002: The system SHALL implement HL7 FHIR R4 compliance for all healthcare data exchanges.

INT-003: The system SHALL use Redox Engine for standardized EHR connectivity.

6.1.2 External Medical Data Sources

INT-004: The system SHALL integrate with medical knowledge sources:

- PubMed API for research literature
- Clinical Trials API for trial matching
- RxNorm/SNOMED for drug databases
- Medical guidelines from NICE/AMA

6.1.3 Provider Network Integration

INT-005: The system SHALL integrate with insurance APIs using X12 EDI standards.

INT-006: The system SHALL integrate with hospital networks using FHIR APIs.

INT-007: The system SHALL integrate with pharmacy APIs using NCPDP standards.

6.2 API Gateway and Service Integration

6.2.1 API Management

INT-008: The system SHALL implement API gateway with Kong/AWS API Gateway.

INT-009: The system SHALL implement rate limiting and throttling for API protection.

INT-010: The system SHALL implement GraphQL API layer for efficient data querying.

6.2.2 Data Integration Hub

INT-011: The system SHALL implement ETL/ELT processes using Apache NiFi/Talend.

INT-012: The system SHALL implement real-time streaming using Apache Kafka.

INT-013: The system SHALL implement data quality validation using Great Expectations.

---

7. Security and Compliance Requirements

7.1 Security Architecture

7.1.1 Zero-Trust Security

SEC-001: The system SHALL implement zero-trust security architecture.

SEC-002: The system SHALL provide end-to-end encryption for all data transactions using AES-256.

SEC-003: The system SHALL implement multi-factor authentication for all users.

SEC-004: The system SHALL implement role-based access control (RBAC) with fine-grained permissions.

7.1.2 Data Protection

SEC-005: The system SHALL implement homomorphic encryption for privacy-preserving AI computations.

SEC-006: The system SHALL use HashiCorp Vault for secrets management.

SEC-007: The system SHALL implement blockchain-based audit trails for data access tracking.

7.2 Compliance Requirements

7.2.1 Healthcare Compliance

COMP-001: The system SHALL comply with HIPAA regulations for US operations.

COMP-002: The system SHALL comply with GDPR for international users.

COMP-003: The system SHALL achieve SOC 2 Type II certification.

COMP-004: The system SHALL comply with FDA regulations for AI diagnostic tools.

7.2.2 AI/ML Compliance

COMP-005: The system SHALL implement AI bias detection and mitigation using IBM AI Fairness 360.

COMP-006: The system SHALL provide explainable AI for all medical recommendations.

COMP-007: The system SHALL implement model versioning and reproducibility for regulatory compliance.

---

8. Technology Stack Requirements

8.1 AI/ML Technology Stack

8.1.1 Core ML/AI Frameworks

TECH-001: The system SHALL use the following AI/ML frameworks:

- **PyTorch Lightning**: Research-oriented medical AI development
- **TensorFlow Extended (TFX)**: Production ML pipelines
- **Hugging Face Transformers**: Medical NLP and multi-modal models
- **MONAI**: Medical imaging AI framework for 3D data

8.1.2 Multi-Agent AI Frameworks

TECH-002: The system SHALL implement multi-agent systems using:

- **CrewAI**: Core multi-agent orchestration framework
- **AutoGen**: Microsoft's multi-agent conversation framework
- **LangGraph**: Graph-based agent workflow management

8.1.3 Knowledge Graph and Ontology Tools

TECH-003: The system SHALL use graph databases:

- **Neo4j**: Primary knowledge graph storage with APOC plugins
- **Amazon Neptune**: Cloud-native graph database for scalability
- **TigerGraph**: High-performance graph analytics

TECH-004: The system SHALL use ontology management tools:

- **Protégé**: Ontology development and editing
- **Apache Jena**: Java framework for semantic web applications
- **RDFLib**: Python library for RDF graph manipulation

8.1.4 RAG and Vector Databases

TECH-005: The system SHALL use vector databases:

- **Pinecone**: Managed vector database for medical embeddings
- **Weaviate**: Open-source vector database with semantic search
- **Milvus**: Scalable vector database for production

TECH-006: The system SHALL implement GraphRAG using:

- **Microsoft GraphRAG**: Advanced graph-based retrieval
- **LangChain**: RAG applications with graph support
- **LlamaIndex**: Data framework for LLM applications

8.2 Infrastructure and Platform Technologies

8.2.1 Cloud Infrastructure

TECH-007: The system SHALL deploy on cloud platforms:

- **AWS**: Primary cloud provider with health-specific services
- **Azure**: Secondary provider for redundancy
- **Google Cloud**: Tertiary provider for specialized AI services

8.2.2 Container Orchestration

TECH-008: The system SHALL use container technologies:

- **Kubernetes**: Container orchestration
- **Docker**: Containerization
- **Istio**: Service mesh for microservices
- **Helm**: Package management

8.2.3 Data Storage and Processing

TECH-009: The system SHALL use databases:

- **PostgreSQL**: Structured medical data
- **MongoDB**: Unstructured data (imaging, notes)
- **ClickHouse**: Analytics and cost data

- **Redis**: Caching and real-time data

TECH-010: The system SHALL use data processing:

- **Apache Spark**: Large-scale data processing
- **Apache Kafka**: Real-time streaming
- **Snowflake**: Data warehousing
- **AWS HealthLake**: FHIR-compliant storage

8.3 Security and Monitoring Technologies

8.3.1 Security Tools

TECH-011: The system SHALL implement security using:

- **HashiCorp Vault**: Secrets management
- **AWS KMS**: Key management
- **Okta**: Identity and access management
- **Sophos**: Endpoint protection

8.3.2 Monitoring and Observability

TECH-012: The system SHALL implement monitoring using:

- **Prometheus**: Metrics collection
- **Grafana**: Dashboards and visualization
- **Jaeger**: Distributed tracing
- **ELK Stack**: Logging and analysis

---

9. Implementation Phases

9.1 Phase 1: Foundation & Knowledge Infrastructure (Months 1-6)

9.1.1 Core Platform Development

PHASE1-001: Implement basic CrewAI multi-agent framework setup.

PHASE1-002: Integrate medical ontologies (SNOMED CT, ICD-11, LOINC).

PHASE1-003: Construct initial knowledge graph using Neo4j with 100K+ medical concepts.

PHASE1-004: Implement MCP (Model Context Protocol) for healthcare context sharing.

PHASE1-005: Establish basic FHIR integration with pilot EHR systems.

PHASE1-006: Set up foundational ML pipeline with MLflow and Kubeflow.

9.1.2 AI/ML Deliverables

PHASE1-007: Deploy simple diagnostic AI agent using pre-trained medical imaging models.

PHASE1-008: Implement basic medical NLP for clinical note processing.

PHASE1-009: Develop prototype GraphRAG system for medical question answering.

PHASE1-010: Establish federated learning infrastructure for privacy-preserving training.

9.2 Phase 2: Advanced AI Agent Development (Months 7-12)

9.2.1 Multi-Agent AI Enhancement

PHASE2-001: Implement full CrewAI agent orchestration with specialized medical agents.

PHASE2-002: Deploy advanced diagnostic agents with multi-modal AI (vision + language).

PHASE2-003: Implement treatment planning agents using reinforcement learning.

PHASE2-004: Deploy PathRAG implementation for treatment pathway discovery.

PHASE2-005: Implement cost analysis agents with predictive ML models.

9.2.2 Knowledge Graph Expansion

PHASE2-006: Implement dynamic knowledge graph updates from medical literature.

PHASE2-007: Create patient-specific knowledge subgraphs for personalized medicine.

PHASE2-008: Establish cross-institutional knowledge federation using privacy-preserving techniques.

PHASE2-009: Implement temporal reasoning in knowledge graphs for treatment sequences.

PHASE2-010: Deploy causal inference networks for evidence-based medicine.

9.2.3 ML/AI Features

PHASE2-011: Deploy multi-modal medical AI combining imaging, genomics, and clinical data.

PHASE2-012: Implement explainable AI for treatment recommendations.

PHASE2-013: Deploy bias detection and mitigation systems for fair healthcare AI.

PHASE2-014: Implement continuous learning from patient outcomes.

PHASE2-015: Achieve real-time AI inference with sub-2 second response times.

9.3 Phase 3: Intelligent Integration & Optimization (Months 13-18)

9.3.1 Advanced RAG and Knowledge Systems

PHASE3-001: Deploy comprehensive GraphRAG with multi-hop reasoning.

PHASE3-002: Implement hierarchical PathRAG for complex treatment planning.

PHASE3-003: Integrate scientific literature with automated knowledge extraction.

PHASE3-004: Deploy cross-modal RAG combining text, images, and molecular data.

PHASE3-005: Implement personalized knowledge graphs adapted to individual patients.

9.3.2 AI Performance Optimization

PHASE3-006: Implement model quantization and optimization for edge deployment.

PHASE3-007: Deploy federated learning across multiple hospitals.

PHASE3-008: Implement multi-agent reinforcement learning for collaborative decision-making.

PHASE3-009: Deploy advanced meta-learning for rapid adaptation to new medical domains.

PHASE3-010: Implement automated hyperparameter optimization for medical AI models.

9.3.3 Context-Aware Systems

PHASE3-011: Deploy advanced MCP implementation with real-time context synchronization.

PHASE3-012: Implement context-aware AI recommendations based on patient, provider, and environmental factors.

PHASE3-013: Deploy temporal context modeling for longitudinal patient care.

PHASE3-014: Implement privacy-preserving context sharing using homomorphic encryption.

9.4 Phase 4: Market-Ready AI Platform (Months 19-24)

9.4.1 Production-Grade AI Systems

PHASE4-001: Deploy scalable multi-agent deployment supporting millions of patients.

PHASE4-002: Implement real-time knowledge graph querying with millisecond response times.

PHASE4-003: Deploy advanced federated learning across 100+ healthcare institutions.

PHASE4-004: Implement AI safety and reliability systems for critical healthcare decisions.

PHASE4-005: Achieve regulatory-compliant AI meeting FDA and international standards.

9.4.2 Advanced AI Features

PHASE4-006: Deploy predictive healthcare analytics using population-level data.

PHASE4-007: Implement AI-driven clinical trial recruitment and patient matching.

PHASE4-008: Deploy automated medical coding and billing optimization.

PHASE4-009: Implement real-time adverse event detection using ML monitoring.

PHASE4-010: Deploy precision medicine recommendations based on genetic and molecular data.

9.4.3 Global Knowledge Integration

PHASE4-011: Deploy multi-language medical knowledge graphs for international deployment.

PHASE4-012: Implement cross-cultural medical ontologies for diverse patient populations.

PHASE4-013: Integrate global clinical guidelines from WHO, NICE, and other authorities.

PHASE4-014: Deploy international drug and device databases with regulatory compliance tracking.

PHASE4-015: Implement real-time global health surveillance using federated AI systems.

---

10. Success Metrics and KPIs

10.1 AI Model Performance Metrics

10.1.1 Diagnostic Accuracy

KPI-001: Diagnostic accuracy >95% sensitivity and specificity across medical specialties.

KPI-002: Treatment recommendation accuracy >90% agreement with expert physicians.

KPI-003: Knowledge graph completeness >99% coverage of major medical concepts.

KPI-004: RAG system performance <500ms query response time with >85% relevance.

10.1.2 Multi-Agent Coordination

KPI-005: Multi-agent coordination efficiency <2% conflict rate between agent recommendations.

KPI-006: Agent collaboration score >90% successful task completion rate.

KPI-007: Decision quality >95% clinically appropriate recommendations.

KPI-008: Conflict resolution <1% unresolved conflicts between agents.

10.2 ML Model Quality Metrics

10.2.1 Model Performance

KPI-009: Model drift detection <5% performance degradation before retraining.

KPI-010: Bias metrics <10% performance disparity across demographic groups.

KPI-011: Explainability scores >80% physician understanding of AI recommendations.

KPI-012: Federated learning convergence <50 rounds for model convergence.

10.2.2 Privacy and Security

KPI-013: Privacy preservation zero patient data leakage in federated systems.

KPI-014: Security incidents <0.1% of transactions flagged for security review.

KPI-015: Compliance score 100% adherence to HIPAA/GDPR requirements.

10.3 Knowledge Graph and Ontology Metrics

10.3.1 Knowledge Quality

KPI-016: Knowledge graph accuracy >98% factual correctness of medical relationships.

KPI-017: Ontology coverage >95% mapping coverage to standard medical vocabularies.

KPI-018: Update latency <24 hours for critical medical knowledge updates.

KPI-019: Query performance <100ms for complex multi-hop knowledge graph queries.

10.3.2 Knowledge Provenance

KPI-020: Knowledge provenance 100% traceability of knowledge sources.

KPI-021: Knowledge freshness >90% of knowledge updated within 30 days of publication.

10.4 PathRAG and Treatment Planning Metrics

10.4.1 Treatment Planning Quality

KPI-022: Pathway accuracy >85% alignment with evidence-based guidelines.

KPI-023: Personalization score >80% patient-specific pathway modifications.

KPI-024: Outcome prediction >75% accuracy in predicting treatment outcomes.

KPI-025: Cost optimization >15% reduction in treatment costs while maintaining quality.

10.4.2 Care Coordination

KPI-026: Timeline optimization >25% reduction in time to diagnosis and treatment.

KPI-027: Care coordination efficiency >90% successful multi-provider coordination.

10.5 Patient and Clinical Outcomes

10.5.1 Patient Satisfaction

KPI-028: Patient satisfaction score >4.5/5.0 for platform usability.

KPI-029: Patient empowerment score >4.0/5.0 for control over medical data.

KPI-030: Cost transparency score >4.5/5.0 for pricing clarity.

10.5.2 Clinical Outcomes

KPI-031: Diagnostic error reduction >30% compared to baseline.

KPI-032: Treatment success rates >20% improvement in relevant outcomes.

KPI-033: Hospital readmission reduction >15% for platform users.

KPI-034: Time to treatment >25% reduction in diagnosis-to-treatment time.

10.6 Platform Performance Metrics

10.6.1 System Performance

KPI-035: System uptime >99.9% availability.

KPI-036: Response time <2 seconds for AI recommendations.

KPI-037: Concurrent users support for >1M users.

KPI-038: Data synchronization real-time across providers.

10.6.2 Adoption and Usage

KPI-039: Platform adoption >500K active users by end of Year 2.

KPI-040: Provider integration >100 healthcare institutions by end of Year 2.

KPI-041: Data volume >10M patient records integrated.

KPI-042: AI recommendation usage >80% of recommendations reviewed by clinicians.

Conclusion

This Software Requirements Specification provides a comprehensive framework for building an AI-driven, patient-centric healthcare platform that leverages cutting-edge technologies including multi-agent AI systems, knowledge graphs, advanced RAG implementations, and privacy-preserving machine learning. The platform addresses critical healthcare challenges while empowering patients and improving clinical outcomes through intelligent automation and coordination.