
Software Requirements Specification for AI-powered Knowledge Organizer (AIKO)

Version 1.2 approved

Prepared by:

Devansh Parapalli, Kaustubh Warade,
Aditya Deshmukh, and Yashasvi Thool

Government College of Engineering, Nagpur

July 31, 2024

Table of Contents

- 1. INTRODUCTION 1
 - 1.1. Purpose 1
 - 1.2. Scope 1
 - 1.3. Product Overview 2
 - 1.4. Definitions 6
- 2. REFERENCES 7
- 3. REQUIREMENTS 7
 - 3.1. Functions 7
 - 3.2. Performance Requirements 10
 - 3.3. Usability Requirements 11
 - 3.4. Interface Requirements 11
 - 3.5. Logical Database Requirements 12
 - 3.6. Design Constraints 12
 - 3.7. Software System Attributes 12
 - 3.8. Supporting Information 13
- 4. VERIFICATION 14
 - 4.1. Unit Testing 14
 - 4.2. Integration Testing 14
 - 4.3. User Acceptance Testing (UAT) 14
 - 4.4. Cross-Platform Compatibility Testing 14
 - 4.5. End-to-End (E2E) Testing 14
 - 4.6. Performance Testing 14
 - 4.7. Accessibility Testing 15
 - 4.8. Data Integrity Testing 15
 - 4.9. Regression Testing 15
- APPENDICES 16
 - Acronyms and Abbreviations 16

Revision History

Name	Date	Change Description	Version
Initial Version	July 28, 2024	Initial release of the Software Requirements Specification document	1.0
Version 1.1	July 29, 2024	Updated scope, functions, and performance requirements	1.1
Version 1.2	July 30, 2024	Finalized Software Requirements Specification document for approval	1.2
Version 1.2 approved	July 31, 2024	Approved Software Requirement Specification document	1.2 approved

1. INTRODUCTION

1.1. Purpose

The purpose of this Software Requirements Specification (SRS) is to delineate the comprehensive functional and non-functional requirements for AIKO (AI-Powered Knowledge Organizer). AIKO is designed to address the multifaceted challenges in knowledge management and retrieval precipitated by the exponential proliferation of digital information across heterogeneous platforms. By leveraging state-of-the-art artificial intelligence and advanced data storage techniques, AIKO aims to provide an innovative solution for efficient information organization and access, thereby mitigating the cognitive load associated with managing vast quantities of disparate data.

1.2. Scope

1.2.1. Functional Scope

AIKO will encompass the following core functionalities:

1. Integration of diverse information sources into a cohesive, centralized knowledge repository
2. Multi-modal information processing and analysis, including but not limited to textual, audio, visual, and structured/unstructured datasets
3. Implementation of advanced search and retrieval mechanisms leveraging natural language processing (NLP) and semantic understanding
4. Provision of a cross-platform, user-centric interface optimized for accessibility and intuitive interaction
5. Robust data security and user privacy protection through state-of-the-art encryption and access control mechanisms

1.2.2. Application and Benefits

AIKO is designed for deployment in various contexts, including:

- Personal knowledge management ecosystems
- Organizational knowledge bases and corporate intranets
- Academic and research environments requiring sophisticated information retrieval

Key benefits include:

1. Substantial enhancement in productivity through streamlined information retrieval processes
2. Facilitation of data-driven decision-making via improved access to relevant knowledge assets
3. Significant reduction in cognitive load associated with managing and navigating large volumes of information
4. Promotion of knowledge discovery and cross-pollination of ideas through advanced content linking and suggestion algorithms

Objectives:

1. Develop a unified data layer capable of ingesting and harmonizing heterogeneous data sources
2. Engineer a high-performance, scalable search engine incorporating advanced NLP and machine learning algorithms for semantic analysis and context-aware information retrieval
3. Implement a multi-layered security architecture ensuring data integrity, confidentiality, and compliance with relevant data protection regulations
4. Design and develop an intuitive, responsive user interface leveraging modern web technologies and adhering to established usability heuristics
5. Create a robust API ecosystem to facilitate seamless integration with existing productivity tools and third-party applications

1.2.3. Alignment with Higher-Level Specifications

AIKO's design and functionality are in consonance with contemporary knowledge management systems and AI-powered information retrieval tools. The system adheres to industry standards for data interoperability, security protocols, and user interface design principles.

1.3. Product Overview

1.3.1. Product Perspective

AIKO is conceptualized as a standalone, yet highly integrable product designed to interface seamlessly with existing information ecosystems and platforms. The system architecture comprises the following key components:

1. System Interfaces:

- RESTful APIs for data ingestion, retrieval, and system administration
- Webhook system for real-time event notifications and integration with external services

2. User Interfaces:

- Responsive web application built on modern frontend frameworks (e.g., Svelte, Vue.js)
- Application Programming Interface (API) for power users and system administrators

3. Hardware Interfaces:

- Standard compatibility with web browsers and mobile devices
- Optional support for specialized hardware such as e-ink devices or smart displays

4. Software Interfaces:

- Integration modules for popular productivity suites (e.g., Microsoft Office, Google Workspace)
- Connectors for custom formats and data sources through extensible plugins

5. Communications Interfaces:

- HTTPS for secure web traffic
- WebSocket protocol for real-time, bidirectional communication
- SSL/TLS encryption for all data in transit

6. Memory Constraints:

- Utilization of scalable, cloud-based storage solutions (e.g., Amazon S3, Google Cloud Storage)
- Implementation of efficient data compression and deduplication techniques

7. Operations:

- Automated monitoring and alerting systems
- Continuous integration and deployment (CI/CD) pipeline for seamless updates

8. Site Adaptation Requirements:

- Configurable self-hosted deployment option for organizations with specific data sovereignty requirements
- Customizable theming and branding capabilities
- Localization and internationalization support

1.3.2. Product Functions

AIKO's core functions encompass:

1. Intelligent Knowledge Integration:

- Automated ingestion and categorization of information from diverse sources
- Entity recognition and relationship mapping across different content types
- Continuous learning and knowledge base enrichment through user interactions

2. Multimodal Information Processing:

- Natural language processing for textual content analysis
- Computer vision algorithms for image and video content understanding
- Audio processing and speech-to-text conversion for audio content

3. Advanced Search and Retrieval:

- Semantic search capabilities with natural language query understanding
- Context-aware result ranking and personalized recommendations
- Faceted search and filtering options for precise information discovery

4. Cross-Platform Accessibility:

- Real-time synchronization of data across devices and platforms
- Progressive Web App (PWA) implementation for seamless mobile experience

5. Personalization and Adaptive Learning:

- User behavior analysis for tailored content suggestions
- Customizable dashboards and information feeds
- Collaborative filtering for knowledge sharing within organizations

1.3.3. User Characteristics

The intended user base for AIKO encompasses:

1. Knowledge Workers:

- Proficiency: Advanced digital literacy
- Usage Pattern: Heavy daily use for information management and retrieval
- Key Requirements: Efficiency, accuracy, and integration with existing work-flows

2. Researchers and Academics:

- Proficiency: High technical competence in specific domains
- Usage Pattern: Intensive use for literature review and knowledge synthesis
- Key Requirements: Comprehensive search capabilities, citation management, and collaboration features

3. Corporate Professionals:

- Proficiency: Varying levels of technical expertise
- Usage Pattern: Regular use for decision support and information sharing
- Key Requirements: User-friendly interface, robust security, and integration with enterprise systems

4. Students:

- Proficiency: Basic to intermediate digital skills
- Usage Pattern: Periodic intensive use for study and project work
- Key Requirements: Intuitive interface, multi-format content support, and collaborative features

5. Data Scientists and Analysts:

- Proficiency: Advanced technical skills in data manipulation and analysis
- Usage Pattern: Regular use for data exploration and knowledge extraction
- Key Requirements: API access, support for large datasets, and integration with analysis tools

1.3.4. Limitations

1. Data Source Constraints:

- System effectiveness is contingent upon the quality and availability of source data
- Certain proprietary data formats may have limited support

2. AI Model Limitations:

- Processing capabilities are bounded by the current state of AI and ML technologies
- Potential for bias in AI-driven recommendations based on training data

3. Connectivity Requirements:

- Full functionality is dependent on internet connectivity, with limited offline capabilities

4. Scalability Considerations:

- Performance may degrade with extremely large datasets or high concurrent user loads

5. Regulatory Compliance:

- System deployment and data handling may be subject to region-specific regulatory requirements

1.4. Definitions

- NLP: Natural Language Processing
- ML: Machine Learning
- OAuth: Open Authorization
- SAML: Security Assertion Markup Language
- GDPR: General Data Protection Regulation
- CI/CD: Continuous Integration and Continuous Deployment
- PWA: Progressive Web Application
- SSL/TLS: Secure Sockets Layer/Transport Layer Security

2. REFERENCES

- [1] John Brooke. 1995. SUS: A quick and dirty usability scale. *Usability Eval. Ind.* 189, (1995), .
- [2] Jakob Nielsen, Kelly Gordan, Kate Morgan, and Feifei Liu. 2024. 10 Usability Heuristics for User Interface Design. Retrieved July 28, 2024 from <https://www.nngroup.com/articles/ten-usability-heuristics/>
- [3] Devansh Parapalli, Kausutbh Warade, Aditya Deshmukh, and Yashasvi Thool. 2024. Project Proposal for AIKO.
- [4] Devansh Parapalli, Kausutbh Warade, Aditya Deshmukh, and Yashasvi Thool. 2024. Project Synopsis for AIKO.
- [5] Joint Technical Committee ISO/IEC JTC 1 and Information Technology, Subcommittee SC 7 - Systems and software engineering. 2018. ISO/IEC/IEEE International Standard - Systems and software engineering – Life cycle processes – Requirements engineering. *ISO/IEC/IEEE 29148:2018(E)* 0, (2018), 1–104. <https://doi.org/10.1109/IEEESTD.2018.8559686>

3. REQUIREMENTS

3.1. Functions

3.1.1. Data Ingestion

- Input Acceptance:
 - System shall support ingestion from multiple sources including web crawling, file uploads, and API integrations
 - Valid input formats shall include, but not be limited to: PDF, DOCX, TXT, CSV, JSON, XML, MP3, MP4, JPG, PNG
- Input Validation:
 - System shall perform format validation on all incoming data
 - Content type detection and validation shall be implemented to ensure data integrity
- Processing and Categorization:
 - Incoming information shall be automatically processed and categorized using ML algorithms

-
- Metadata extraction shall be performed to facilitate efficient indexing and retrieval
 - System shall support custom categorization rules defined by users.

3.1.2. Information Processing

- Content Analysis:
 - NLP algorithms shall be employed for textual content analysis, including:
 - Named Entity Recognition (NER)
 - Sentiment Analysis
 - Topic Modeling
 - Computer Vision algorithms as well as Vision and Auditory LLMs shall be used for image and video analysis
 - Audio processing shall include speech-to-text conversion and audio feature extraction
- Metadata Generation:
 - System shall automatically generate relevant metadata for all processed content
 - Metadata shall include, but not be limited to: creation date, last modified date, author, keywords, file size, and content summary
 - Metadata shall be generated utilizing open source models.
- Knowledge Base Integration:
 - New information shall be cross-referenced with existing knowledge base
 - System shall identify and establish relationships between related pieces of information
 - Continuous learning mechanisms shall be implemented to improve categorization and relationship mapping over time. Implementation of Feedback Systems such as Reinforcement Learning using Human Feedback (RLHF) shall be done.

3.1.3. Search and Retrieval

- Query Processing:
 - System shall support natural language queries
 - Advanced boolean search operators shall be supported, both as part of the NL queries and as part of the query API.

-
- Query expansion and synonym matching shall be implemented to improve search results
 - Results Ranking:
 - Search results shall be ranked based on relevance, recency, and user context. User's last few actions shall be saved and used to provide the context.
 - Personalized ranking algorithms shall be employed to tailor results to individual user preferences.
 - System shall support customizable ranking criteria for different users (user-customizable)
 - Result Presentation:
 - Search results shall be presented in a clear, scannable format
 - Preview functionality shall be available for different content types
 - Faceted navigation shall be provided to allow users to filter and refine search results, filtering shall be done client sided once the initial broad response is known.

3.1.4. User Management

- Authentication:
 - System shall support multiple authentication methods, including:
 - Username/password
 - Single Sign-On (SSO) via SAML or OAuth 2.0
 - Multi-factor authentication (MFA)
- Authorization:
 - Role-based access control (RBAC) shall be implemented
 - Granular permissions shall be configurable for different content types and actions
 - User activity logging shall be implemented for audit purposes
- Personalization:
 - User profiles shall store individual preferences and settings
 - Personalized dashboards shall be customizable by users
 - System shall learn from user behavior to improve personalized recommendations

3.1.5. Synchronization

- Data Consistency:
 - Real-time synchronization shall be implemented across all connected devices
 - Conflict resolution mechanisms shall be in place to handle simultaneous updates
- Update Propagation:
 - Changes shall be propagated to all relevant parts of the system in real-time
 - Caching mechanisms shall be implemented to optimize performance and reduce latency
 - Notifications shall be sent to relevant users upon important updates or changes

3.2. Performance Requirements

1. Search Response Time:
 - 95th percentile of search queries shall return results in < 4 seconds
 - 99th percentile of search queries shall return results in < 8 seconds
2. Data Ingestion Rate:
 - System shall be capable of processing up to 100 MB of data per minute
 - Batch ingestion jobs shall support processing of up to 1 GB of data per hour
3. Synchronization Delay:
 - Updates shall be synchronized across devices within 15 seconds under normal network conditions
4. System Availability:
 - System shall maintain 99.9% uptime, excluding scheduled maintenance
 - Scheduled maintenance shall not exceed 16 hours per month
5. Scalability:
 - System shall support up to 100 concurrent users without degradation in performance
 - Database shall be capable of storing and efficiently querying up to 1 TB of data
6. API Performance:
 - API endpoints shall start to respond to requests within 1s for 95% of calls
 - API shall support up to 100 requests per second

3.3. Usability Requirements

1. User Interface:

- Interface shall adhere to established usability heuristics as defined by Nielsen Norman Group
- System shall achieve a System Usability Scale (SUS) score of at least 80

2. Learnability:

- New users shall be able to perform basic search and retrieval tasks within 5 minutes of introduction to the system
- Interactive tutorials shall be provided for all major features

3. Efficiency:

- Expert users shall be able to access any desired information within 3 clicks or interactions
- Keyboard shortcuts shall be provided for all common actions

4. Error Handling:

- Error messages shall be clear, concise, and actionable
- System shall provide suggestions for error recovery where applicable

5. Accessibility:

- Web interface shall comply with WCAG 2.1 Level AA standards
- System shall be compatible with common screen readers and assistive technologies

3.4. Interface Requirements

3.4.1. User Interfaces

- Web Application:
 - Responsive design supporting desktop, tablet, and mobile browsers
 - Support for latest versions of Chrome, Firefox, Safari, and Edge
 - Minimum supported screen resolution: 320px width
- Browser Extension:
 - Support for Chrome, Firefox, and Edge browsers
 - Ability to save web content with one-click action

3.4.2. API Interfaces

- RESTful API:
 - OpenAPI (Swagger) specification for all endpoints
 - JSON format for request and response payloads
 - Rate limiting and throttling mechanisms
- WebSocket API:
 - Support for real-time data updates and notifications
 - Implement heartbeat mechanism for connection health monitoring

3.4.3. Data Formats

- Supported Input Formats:
 - Documents: PDF, DOCX, TXT, RTF, ODT
 - Spreadsheets: XLSX, CSV, ODS
 - Presentations: PPTX, ODP
 - Images: JPG, PNG, GIF, SVG
 - Audio: MP3, WAV, OGG
 - Video: MP4, AVI, MOV
 - Data: JSON, XML, YAML
 - Plain Text: Any format convertible to plaintext.

3.5. Logical Database Requirements

- Implement a scalable, distributed database system
- Support for both structured and unstructured data storage
- Efficient indexing mechanisms for fast retrieval
- Robust backup and recovery mechanisms
- Real-time synchronization across multiple instances

3.6. Design Constraints

- Utilize open-source technologies where feasible
- Ensure compliance with data protection regulations (e.g., GDPR)
- Implement a modular architecture to facilitate extensibility
- Support self-hosted deployments for organizations with specific requirements
- Optimize for performance and scalability to handle large volumes of data
- Design for cross-platform compatibility (web, iOS, Android)

3.7. Software System Attributes

- Reliability
 - Implement fault-tolerant design with graceful degradation
 - Provide comprehensive error handling and logging mechanisms
 - Ensure data integrity through robust validation and consistency checks
- Availability
 - Aim for 99.9% uptime (excluding scheduled maintenance)
 - Implement load balancing and failover mechanisms
 - Design for 24/7 operation with minimal downtime
- Security
 - Implement end-to-end encryption for data in transit and at rest
 - Support multi-factor authentication
 - Conduct regular security audits and penetration testing
 - Implement OAuth 2.0 and SAML for authentication and authorization
 - Ensure compliance with relevant data protection regulations
- Maintainability
 - Develop a well-documented codebase
 - Implement automated testing and continuous integration practices
 - Design a modular architecture for easy updates and feature additions
 - Provide comprehensive API documentation
- Portability
 - Ensure cross-platform compatibility (web, iOS, Android)
 - Support containerized deployment for easy migration and scaling
 - Design for cloud-agnostic deployment to avoid vendor lock-in

3.8. Supporting Information

- User interface mockups and wire-frames
- API documentation
- Data flow diagrams
- Use case scenarios
- Entity-Relationship Diagrams (ERDs) for the database schema

4. VERIFICATION

The verification process for AIKO will ensure that the system meets the specified requirements. The following verification activities will be conducted:

4.1. Unit Testing

- Develop and execute unit tests for key components and modules
- Use appropriate testing frameworks (e.g., Vitest for JavaScript, pytest for Python)
- Aim for at least 70% code coverage across critical units

4.2. Integration Testing

- Perform integration tests for main system interfaces
- Test data flow between different modules
- Verify correct functioning of primary API endpoints

4.3. User Acceptance Testing (UAT)

- Develop a UAT plan covering main user stories and use cases
- Conduct UAT with a small group of potential users (e.g., batch-mates, professors)
- Collect and analyze user feedback on UI/UX elements

4.4. Cross-Platform Compatibility Testing

- Test the web application on major browsers (Chrome, Firefox, Safari)
- Verify responsiveness on desktop and mobile devices
- Ensure consistent performance across supported platforms

4.5. End-to-End (E2E) Testing

- Develop E2E test scenarios for major user workflows
- Verify the complete user journey from data ingestion to retrieval
- Test basic error handling throughout the system

4.6. Performance Testing

- Execute basic performance tests to ensure system responsiveness
- Verify that the system meets key performance requirements
- Test system behavior under moderate load conditions

4.7. Accessibility Testing

- Conduct basic accessibility checks (e.g., color contrast, keyboard navigation)
- Test compatibility with at least one common screen reader

4.8. Data Integrity Testing

- Verify data consistency across storage layers
- Test basic data synchronization processes
- Ensure data integrity is maintained during typical operations

4.9. Regression Testing

- Implement basic automated regression tests for critical functionalities
- Run regression tests after significant updates or bug fixes

APPENDICES

Acronyms and Abbreviations

- AI: Artificial Intelligence
- API: Application Programming Interface
- GDPR: General Data Protection Regulation
- ML: Machine Learning
- NLP: Natural Language Processing
- OAuth: Open Authorization
- SAML: Security Assertion Markup Language
- UI: User Interface
- UX: User Experience
- WCAG: Web Content Accessibility Guidelines
- XAI: Explainable AI

Approval:

Dr. D. J. Chaudhari

Project Guide

Assistant Professor, CSE Department

Sector-27, MIHAN Rehabilitation Colony

Khapri, Nagpur

441108

Date: July 31, 2024