



Namal University Mianwali
Department of Computer Science
CSC-225 – Software Engineering

**Software Requirements Specification
for
AI-Based Task Manager & Smart Scheduler**

Version 1.0
Approved

Prepared by:

Mishaal Aqdas (NUM-BSCS-2024-35)
Khadija tul Kubra (NUM-BSCS-2024-30)
Muhammad Imran (NUM-BSCS-2024-50)

Submitted to:
Mam Asiya Batool

Date: 28 December 2025

Contents

1 Revision History	3
Revision History	3
2 Introduction	4
2.1 Purpose	4
2.2 Document Conventions	4
2.3 Intended Audience and Reading Suggestions	4
2.4 Project Scope	4
2.4.1 Key Features	4
2.5 References	5
3 Overall Description	6
3.1 Product Perspective	6
3.2 Product Functions	6
3.3 User Classes and Characteristics	6
3.3.1 Primary Users - Registered Users	6
3.3.2 Secondary Users - System Administrator	7
3.4 Operating Environment	7
3.5 Design and Implementation Constraints	7
3.6 User Documentation	7
3.7 Assumptions and Dependencies	8
3.7.1 Assumptions	8
3.7.2 Dependencies	8
4 Specific Requirements	9
4.1 Functional Requirements	9
4.1.1 User Authentication	9
4.1.2 Task Management	9
4.1.3 AI-Based Task Rescheduling	10
4.1.4 Smart Suggestions	10
4.1.5 Schedule Views	11
4.1.6 Reminder and Notification System	11
4.1.7 Search Functionality	12
4.1.8 Progress Tracking	13
4.1.9 Data Backup and Restore	13
4.1.10 User Role Management	14
4.2 External Interface Requirements	15
4.2.1 User Interfaces	15
4.2.2 Hardware Interfaces	15
4.2.3 Software Interfaces	15
4.2.4 Communications Interfaces	16
4.3 Performance Requirements	17
4.4 Design Constraints	17
4.4.1 Standards Compliance	17
4.4.2 Hardware Limitations	18

4.4.3	Technology Constraints	18
4.4.4	Regulatory Constraints	18
4.5	Software System Attributes	19
4.5.1	Security	19
4.5.2	Maintainability	19
4.5.3	Reliability	20
4.5.4	Availability	20
4.5.5	Portability	20
4.6	Other Requirements	21
4.6.1	Database Requirements	21
4.6.2	Operations Requirements	21
4.6.3	Site Adaptation Requirements	22
4.6.4	Documentation Requirements	22
4.6.5	Training Requirements	22
4.7	Non-Functional Requirements	23
4.7.1	NFR-1: Performance Requirements	23
4.7.2	NFR-2: Security Requirements	23
4.7.3	NFR-3: Usability Requirements	23
4.7.4	NFR-4: Reliability Requirements	23
4.7.5	NFR-5: Scalability Requirements	24
4.7.6	NFR-6: Maintainability Requirements	24
4.7.7	NFR-7: Compatibility Requirements	24
4.7.8	NFR-8: Availability Requirements	24
4.7.9	NFR-9: Data Integrity Requirements	24
4.7.10	NFR-10: Privacy Requirements	25
4.7.11	NFR-11: AI Accuracy Requirements	25
4.7.12	NFR-12: Logging and Monitoring Requirements	25
4.7.13	NFR-13: Backup and Recovery Requirements	25
4.7.14	NFR-14: Response and Error Handling	25
5	Use Case Diagram	26
5.1	Diagram	26
5.2	Link	26
6	Context Diagram	27
6.1	Diagram	27
6.2	Link	27
6.3	User to AI-Based Task Manager	27
6.4	AI-Based Task Manager to User	27
6.5	AI-Based Task Manager to AI	28
6.6	AI to AI-Based Task Manager	28
6.7	Admin to AI-Based Task Manager	28
6.8	AI-Based Task Manager to Admin	28
7	Appendices	30
7.1	Appendix A: Glossary	30
7.2	Appendix B: Analysis Models	30
7.3	Appendix C: Issues List	31

1 Revision History

Version	Date	Description	Author
1.0	28 Dec 2025	Initial SRS Document	Project Team

2 Introduction

2.1 Purpose

This Software Requirements Specification (SRS) document provides a complete description of all functions and specifications of the AI-Based Task Manager & Smart Scheduler application. This document is intended for developers, project managers, testers, users, and stakeholders involved in the development and deployment of the system.

2.2 Document Conventions

- Priority levels are classified as: Critical, High, Medium, and Low
- Use cases are identified with unique IDs (UC-01, UC-02, etc.)
- Functional requirements are labeled as FR-X
- Non-functional requirements are labeled as NFR-X
- Technical terms are explained in the Glossary section

2.3 Intended Audience and Reading Suggestions

This document is intended for:

- **Developers:** Should focus on Sections 3 (Specific Requirements) and 4 (System Features)
- **Project Managers:** Should read all sections for complete project understanding
- **Testers:** Should focus on Sections 3, 4, and 5 for test case development
- **End Users:** Should read Sections 1, 2, and 4 for system understanding
- **Documentation Writers:** Should read all sections

2.4 Project Scope

The AI-Based Task Manager & Smart Scheduler is a mobile application designed to help users efficiently organize and manage their daily tasks and schedules. The system addresses the common problem of task management by providing a digital platform that stores tasks, sets reminders, and uses AI to provide intelligent scheduling suggestions.

2.4.1 Key Features

- User authentication and account management
- Task creation, editing, deletion, and completion tracking
- AI-powered automatic task rescheduling
- Smart suggestions based on user behavior

- Multiple schedule views (daily, weekly, monthly)
- Reminders/notifications
- Progress tracking and productivity statistics
- Data backup and restore capabilities

2.5 References

- IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications
- Project Proposal Document - AI-Based Task Manager & Smart Scheduler
- Waterfall Software Development Model Documentation

3 Overall Description

3.1 Product Perspective

The AI-Based Task Manager & Smart Scheduler is a new, self-contained product that operates as a standalone mobile application. The system consists of the following major components:

- **User Interface:** Mobile-friendly interface for task management
- **Database System:** Stores user data, tasks, and schedules (Firebase/SQLite)
- **AI Scheduling Module:** Provides intelligent task rescheduling and suggestions
- **Notification Service:** Manages reminders and alerts
- **Authentication System:** Handles user login and security
- **Cloud Backup Service:** Enables data backup and restoration

3.2 Product Functions

The major functions of the system include:

- User registration and authentication
- Task management (create, read, update, delete)
- Automatic AI-based task rescheduling
- Smart productivity suggestions
- Multiple calendar views
- Reminder and notification system
- Task search functionality
- Progress tracking and statistics
- Data backup and restore

3.3 User Classes and Characteristics

3.3.1 Primary Users - Registered Users

- Students managing academic schedules
- Professionals handling work tasks
- Individuals with multiple daily responsibilities
- Technical expertise: Basic to intermediate
- Frequency of use: Daily, multiple times per day

3.3.2 Secondary Users - System Administrator

- Manages system maintenance and updates
- Handles user account issues
- Monitors system performance
- Technical expertise: Advanced
- Frequency of use: As needed for maintenance

3.4 Operating Environment

- **Client Side:** Mobile devices (Android/iOS) with modern web browsers
- **Server Side:** Cloud-based server infrastructure
- **Database:** Firebase or SQLite
- **Network:** Internet connectivity required for synchronization and cloud features
- **Operating Systems:** Compatible with Android 8.0+ and iOS 12.0+

3.5 Design and Implementation Constraints

- Must comply with data protection and privacy regulations
- System response time must be under 3 seconds for most operations
- Must work on devices with limited storage and processing power
- AI module requires sufficient computational resources
- Must maintain data consistency across multiple devices
- Limited by third-party API capabilities for notifications

3.6 User Documentation

The following user documentation will be provided:

- User Manual with step-by-step instructions
- Quick Start Guide for new users
- Video tutorials for key features
- In-app help and tooltips
- FAQ section for common issues
- Administrator Guide for system maintenance

3.7 Assumptions and Dependencies

3.7.1 Assumptions

- Users have access to a smartphone or tablet device
- Users have basic technical literacy
- Internet connectivity is generally available
- Users will provide valid email addresses for registration

3.7.2 Dependencies

- Availability of Firebase/SQLite database services
- Reliability of cloud storage providers for backups
- Operating system notification APIs
- Third-party authentication services (if implemented)
- Internet service provider reliability

4 Specific Requirements

4.1 Functional Requirements

4.1.1 User Authentication

Introduction

The system shall provide secure user authentication mechanisms to allow users to register, log in, and log out safely.

Inputs

- Email address
- Username
- Password
- Login credentials

Processing

- Validate email format and password strength
- Prevent duplicate email registrations
- Authenticate users with valid credentials
- Handle invalid login attempts
- Enable secure logout
- Provide password recovery functionality

Outputs

- Successful login/logout confirmation
- Error messages for invalid authentication attempts

4.1.2 Task Management

Introduction

The system shall allow users to manage tasks efficiently.

Inputs

- Task title
- Task description
- Deadline
- Priority level

Processing

- Create new tasks
- Edit task details
- Delete tasks with confirmation
- Mark tasks as completed
- Validate task data
- Prevent past deadline entries

Outputs

- Task saved, updated, deleted, or completed confirmation

4.1.3 AI-Based Task Rescheduling

Introduction

The system shall automatically reschedule overdue tasks using AI algorithms.

Inputs

- Task deadlines
- User availability
- Task priority

Processing

- Detect overdue tasks
- Analyze available time slots
- Assign new deadlines
- Notify users of changes

Outputs

- Updated task schedule
- Rescheduling notification

4.1.4 Smart Suggestions

Introduction

The system shall provide AI-powered productivity suggestions.

Inputs

- User task history
- Work duration
- Free time slots

Processing

- Analyze user behavior
- Suggest breaks or pending tasks
- Learn from user responses

Outputs

- Productivity suggestions
- User notifications

4.1.5 Schedule Views

Introduction

The system shall provide multiple schedule views.

Inputs

- Task data
- Selected time range

Processing

- Display daily, weekly, monthly views
- Filter tasks by date
- Navigate between dates

Outputs

- Visual schedule display

4.1.6 Reminder and Notification System

Introduction

The system shall notify users about upcoming tasks.

Inputs

- Reminder time
- Task deadline

Processing

- Schedule notifications
- Send reminders
- Manage permissions

Outputs

- Reminder notifications

4.1.7 Search Functionality

Introduction

The system shall provide a task search and filtering mechanism that allows users to retrieve tasks by applying filters and entering search keywords.

Inputs

- Selected filter criteria (Task Name, Category, Creation Date, Due Date)
- Search keywords entered by the user

Processing

- User clicks the Search option
- System displays available filter options
- User applies one or more filters
- User enters search keywords
- System scans task records based on selected filters and keywords
- System matches relevant tasks from the database
- System generates a filtered result set

Outputs

- List of tasks matching the selected filters and entered keywords

4.1.8 Progress Tracking

Introduction

The system shall track productivity statistics to help users monitor task completion and performance trends.

Inputs

- Completed task records
- Pending task records

Processing

- System retrieves completed and pending task data from the database
- System calculates total number of tasks
- System calculates the number of completed tasks
- System computes task completion percentage
- System analyzes productivity data for selected time periods (daily, weekly, monthly)
- System generates productivity statistics
- System prepares graphical data representations

Outputs

- Completion percentage
- Weekly and monthly productivity charts
- Productivity summary report

4.1.9 Data Backup and Restore

Introduction

The system shall support data backup and restoration.

Inputs

- Task data
- Cloud storage credentials

Processing

- Encrypt data
- Backup and restore data

Outputs

- Backup confirmation
- Restored data

4.1.10 User Role Management

Introduction

The system shall manage user roles.

Inputs

- User role information

Processing

- Assign roles
- Restrict access

Outputs

- Role-based access control

4.2 External Interface Requirements

4.2.1 User Interfaces

- The system shall provide a mobile-friendly, responsive user interface
- All screens shall follow consistent design patterns and branding
- The interface shall support both portrait and landscape orientations
- Touch gestures (swipe, tap, long-press) shall be supported
- The interface shall be accessible with support for screen readers
- The system shall display clear error messages and feedback to users
- Navigation shall be intuitive with a maximum of 3 clicks to reach any feature
- The UI shall use appropriate color schemes for different priority levels
- Loading indicators shall be displayed for operations taking more than 1 second

4.2.2 Hardware Interfaces

- The system shall interface with device notification systems (Android/iOS)
- The system shall access device storage for local caching of task data
- The system shall support device camera for potential photo attachments to tasks
- The system shall utilize device sensors for activity detection in future versions
- The system shall support biometric hardware (fingerprint/face recognition) for authentication
- The system shall be compatible with various screen sizes and resolutions
- Minimum hardware requirements: 2GB RAM, 100MB storage space

4.2.3 Software Interfaces

- Database Interface: Firebase Realtime Database or Cloud Firestore for data storage and retrieval
- Alternative Database: SQLite for offline local storage
- Cloud Storage Interface: Firebase Storage or AWS S3 for backup and restore functionality
- Operating System APIs: Android API Level 26+ (Android 8.0) and iOS 12.0+
- Authentication APIs: Firebase Authentication for user management
- Notification APIs: Firebase Cloud Messaging (FCM) and Apple Push Notification Service (APNS)

- Third-party Calendar APIs: Google Calendar API, Apple Calendar (EventKit) for integration
- AI/ML Services: TensorFlow Lite or custom AI models for smart suggestions

4.2.4 Communications Interfaces

- HTTPS/TLS 1.2+ protocol for all client-server communications
- RESTful API architecture for backend services
- JSON format for data exchange between client and server
- WebSocket connections for real-time synchronization of tasks (optional)
- Push notification protocols: FCM for Android, APNS for iOS
- OAuth 2.0 for third-party authentication (Google, Facebook)
- Maximum API response time: 3 seconds under normal conditions
- Support for offline mode with data synchronization upon reconnection

4.3 Performance Requirements

1. The system shall load the dashboard within 2-3 seconds under normal network conditions (3G or better).
2. Task creation, editing, and deletion operations shall complete within 1 second.
3. AI suggestions shall be generated and displayed within 5 seconds.
4. The system shall support a minimum of 10,000 concurrent users without performance degradation.
5. Search functionality shall return results within 2 seconds for databases containing up to 10,000 tasks per user.
6. The system shall handle up to 1,000 tasks per user without noticeable slowdown.
7. Notification delivery shall occur within 10 seconds of the scheduled reminder time.
8. Database queries shall be optimized to execute in less than 500ms.
9. The application size shall not exceed 50MB for initial download.
10. Memory usage shall not exceed 150MB during peak operation.
11. Battery consumption shall be optimized to use less than 5% per hour of active use.
12. Data synchronization shall complete within 10 seconds for up to 500 tasks.

4.4 Design Constraints

4.4.1 Standards Compliance

- The system shall comply with IEEE 830-1998 standard for Software Requirements Specification
- The system shall follow ISO/IEC 25010 software quality standards
- The system shall adhere to W3C Web Content Accessibility Guidelines (WCAG) 2.1 Level AA
- The system shall implement OWASP Mobile Application Security standards
- The system shall comply with OAuth 2.0 standard for authentication
- The system shall follow REST architectural constraints for API design
- The system shall adhere to Material Design guidelines (Android) and Human Interface Guidelines (iOS)

4.4.2 Hardware Limitations

- The system must operate on devices with minimum 2GB RAM
- The system must function on devices with at least 100MB available storage
- The system must support screen sizes from 4.5 inches to 13 inches
- The system must work on devices with ARM or x86 processor architectures
- The system must be compatible with devices having various screen densities (ldpi to xxxhdpi)
- Battery optimization shall be implemented to avoid excessive drain on older devices
- The system shall gracefully handle devices with limited network bandwidth (2G/3G)

4.4.3 Technology Constraints

- The system shall be developed using React Native or Flutter for cross-platform compatibility
- Backend services shall be implemented using Node.js or Python (Django/Flask)
- Database shall be Firebase (Firestore/Realtime Database) or MongoDB
- AI/ML components shall use TensorFlow Lite or PyTorch Mobile
- Version control shall use Git with GitHub or GitLab
- CI/CD pipeline shall be implemented using GitHub Actions or Jenkins

4.4.4 Regulatory Constraints

- The system shall comply with GDPR (General Data Protection Regulation)
- The system shall comply with CCPA (California Consumer Privacy Act)
- The system shall comply with COPPA for users under 13 years of age
- The system shall implement proper data encryption as per industry standards
- The system shall provide mechanisms for data export and deletion (right to be forgotten)
- Privacy policy and terms of service must be clearly presented and accepted by users

4.5 Software System Attributes

4.5.1 Security

- SEC-1: User passwords shall be hashed using bcrypt or Argon2 with salt
- SEC-2: Session management shall use secure, randomly generated tokens
- SEC-3: All API endpoints shall require authentication except login and signup
- SEC-4: The system shall implement input validation to prevent SQL injection and XSS attacks
- SEC-5: Sensitive data shall be encrypted at rest using AES-256 encryption
- SEC-6: All network communications shall use TLS 1.2 or higher
- SEC-7: The system shall implement rate limiting to prevent brute force attacks (max 5 failed attempts)
- SEC-8: Session tokens shall expire after 30 minutes of inactivity
- SEC-9: Multi-factor authentication (MFA) shall be available as an optional security feature
- SEC-10: Admin functions shall require additional authentication and authorization
- SEC-11: Security logs shall be maintained for all critical operations
- SEC-12: The system shall comply with OWASP Mobile Top 10 security guidelines

4.5.2 Maintainability

- MAINT-1: Code shall follow consistent naming conventions and coding standards
- MAINT-2: All functions and classes shall include inline documentation
- MAINT-3: The system shall use modular architecture with clear separation of concerns
- MAINT-4: API documentation shall be maintained using Swagger/OpenAPI specification
- MAINT-5: The system shall include comprehensive unit tests with minimum 80% code coverage
- MAINT-6: Integration tests shall be implemented for critical user workflows
- MAINT-7: Version control shall be used with meaningful commit messages
- MAINT-8: Code reviews shall be mandatory before merging to main branch
- MAINT-9: Dependency management shall use package managers (npm, pip) with locked versions
- MAINT-10: The system shall support hot-fixes without requiring full redeployment
- MAINT-11: Database migrations shall be version-controlled and reversible
- MAINT-12: System shall include logging and monitoring capabilities for debugging

4.5.3 Reliability

- REL-1: The system shall maintain 99.5% uptime availability
- REL-2: Mean Time Between Failures (MTBF) shall be at least 720 hours
- REL-3: Mean Time To Recovery (MTTR) shall not exceed 1 hour
- REL-4: The system shall implement automatic failover mechanisms
- REL-5: Data backups shall be performed automatically every 24 hours
- REL-6: The system shall gracefully handle and recover from crashes
- REL-7: Notification system shall have 99% delivery reliability

4.5.4 Availability

- AVAIL-1: The system shall be accessible 24/7 with scheduled maintenance windows
- AVAIL-2: Planned maintenance shall be scheduled during low-usage periods
- AVAIL-3: Users shall be notified 48 hours in advance of scheduled maintenance
- AVAIL-4: Critical services (reminders, notifications) shall remain available during non-critical maintenance
- AVAIL-5: The system shall implement redundancy for critical components

4.5.5 Portability

- PORT-1: The system shall be deployable on multiple cloud platforms (AWS, Google Cloud, Azure)
- PORT-2: The mobile application shall support both Android and iOS platforms
- PORT-3: Data export functionality shall support JSON, CSV, and PDF formats
- PORT-4: The system shall allow migration between different database systems with minimal code changes

4.6 Other Requirements

4.6.1 Database Requirements

- DB-1: The database shall support ACID properties for transactional integrity
- DB-2: Database schema shall include tables/collections for: Users, Tasks, Reminders, Settings, Logs
- DB-3: User table shall store: user_id, email, username, password_hash, created_at, updated_at
- DB-4: Task table shall store: task_id, user_id, title, description, deadline, priority, status, created_at, updated_at
- DB-5: The database shall implement proper indexing on frequently queried fields (user_id, deadline, status)
- DB-6: Foreign key constraints shall be implemented to maintain referential integrity
- DB-7: Database shall support full-text search capabilities for task search functionality
- DB-8: Soft delete mechanism shall be implemented for tasks (with is_deleted flag)
- DB-9: Database backup retention period shall be minimum 30 days
- DB-10: The system shall implement database connection pooling for optimal performance

4.6.2 Operations Requirements

- OPS-1: System administrators shall have access to monitoring dashboards
- OPS-2: Real-time alerts shall be sent for system errors and performance degradation
- OPS-3: The system shall generate daily operational reports including user activity, system health, and errors
- OPS-4: Log files shall be rotated weekly and archived for 90 days
- OPS-5: The system shall support graceful shutdown and startup procedures
- OPS-6: Deployment process shall be automated using CI/CD pipelines
- OPS-7: Rollback procedures shall be available for failed deployments
- OPS-8: System health checks shall run every 5 minutes
- OPS-9: Performance metrics (CPU, memory, network) shall be monitored continuously
- OPS-10: The system shall support A/B testing for new features

4.6.3 Site Adaptation Requirements

- SITE-1: The system shall support multiple languages (English as default, with framework for additional languages)
- SITE-2: Date and time formats shall adapt to user's locale settings
- SITE-3: The system shall support multiple time zones for international users
- SITE-4: Currency formats shall adapt based on user location (for potential premium features)
- SITE-5: The system shall comply with local data protection regulations in regions where deployed
- SITE-6: Content shall be adaptable for right-to-left (RTL) languages
- SITE-7: The system shall support region-specific holiday calendars

4.6.4 Documentation Requirements

- DOC-1: User manual shall be provided in digital format
- DOC-2: API documentation shall be comprehensive and up-to-date
- DOC-3: Installation and deployment guides shall be provided for administrators
- DOC-4: In-app help and tutorials shall be available for first-time users
- DOC-5: FAQ section shall address common user questions
- DOC-6: Technical documentation shall include system architecture diagrams

4.6.5 Training Requirements

- TRAIN-1: The system shall be intuitive enough that users can complete basic tasks without formal training
- TRAIN-2: Tutorial videos shall be available for advanced features
- TRAIN-3: Interactive onboarding flow shall guide new users through key features
- TRAIN-4: Administrator training materials shall be provided for system management

4.7 Non-Functional Requirements

4.7.1 NFR-1: Performance Requirements

- NFR-1.1: The system shall load the dashboard within 2-3 seconds under normal network conditions
- NFR-1.2: Task creation, editing, and deletion shall occur without noticeable delay (less than 1 second)
- NFR-1.3: AI suggestions shall be generated in real-time or within 5 seconds
- NFR-1.4: The system shall handle multiple concurrent users without performance degradation
- NFR-1.5: Search results shall be displayed within 2 seconds of query input

4.7.2 NFR-2: Security Requirements

- NFR-2.1: User passwords must be encrypted using industry-standard hashing algorithms
- NFR-2.2: Only authenticated users can access protected modules
- NFR-2.3: Admin features must be accessible only to authorized administrators
- NFR-2.4: The system shall protect against SQL injection and XSS attacks
- NFR-2.5: Session tokens shall expire after a defined period of inactivity
- NFR-2.6: Data transmission shall use HTTPS encryption

4.7.3 NFR-3: Usability Requirements

- NFR-3.1: The user interface shall be intuitive and user-friendly
- NFR-3.2: New users shall understand basic operations without formal training
- NFR-3.3: Important actions shall be easily accessible within 3 clicks
- NFR-3.4: The system shall provide clear feedback messages for all operations
- NFR-3.5: Error messages shall be descriptive and suggest corrective actions
- NFR-3.6: The interface shall follow consistent design patterns throughout

4.7.4 NFR-4: Reliability Requirements

- NFR-4.1: The system shall be available 24/7 with 99.5% uptime
- NFR-4.2: User data shall not be lost during system crashes or unexpected shutdowns
- NFR-4.3: Automatic backups shall be performed daily
- NFR-4.4: The notification system shall deliver reminders with 99% reliability
- NFR-4.5: System failures shall not affect data integrity

4.7.5 NFR-5: Scalability Requirements

- NFR-5.1: The system shall support up to 10,000 concurrent users initially
- NFR-5.2: The database shall efficiently handle millions of task records
- NFR-5.3: The AI engine shall scale to handle growing user activity
- NFR-5.4: System architecture shall support horizontal scaling

4.7.6 NFR-6: Maintainability Requirements

- NFR-6.1: The system shall be easy to update and modify
- NFR-6.2: Code shall be well-documented with inline comments
- NFR-6.3: System maintenance shall not affect active users
- NFR-6.4: Updates shall not cause data loss or require migration
- NFR-6.5: Bug fixes shall be deployable within 24 hours

4.7.7 NFR-7: Compatibility Requirements

- NFR-7.1: The application shall work on Android 8.0+ devices
- NFR-7.2: The application shall work on iOS 12.0+ devices
- NFR-7.3: The system shall be compatible with modern web browsers
- NFR-7.4: The interface shall be responsive across different screen sizes
- NFR-7.5: The system shall support integration with third-party calendar services

4.7.8 NFR-8: Availability Requirements

- NFR-8.1: The system shall maintain high uptime with minimal downtime
- NFR-8.2: Scheduled maintenance shall be announced 48 hours in advance
- NFR-8.3: Critical services (reminders, notifications) shall remain available during maintenance
- NFR-8.4: System recovery time shall be less than 1 hour in case of failure

4.7.9 NFR-9: Data Integrity Requirements

- NFR-9.1: Task data shall remain accurate and consistent across all views
- NFR-9.2: Duplicate task records shall not be created
- NFR-9.3: All task updates shall be properly saved and reflected system-wide
- NFR-9.4: Database transactions shall follow ACID properties

4.7.10 NFR-10: Privacy Requirements

- NFR-10.1: User personal data shall be kept confidential
- NFR-10.2: The system shall comply with GDPR and data protection regulations
- NFR-10.3: Users shall have control over their personal information
- NFR-10.4: User data shall not be shared with third parties without consent
- NFR-10.5: Users shall be able to export or delete their data

4.7.11 NFR-11: AI Accuracy Requirements

- NFR-11.1: AI suggestions shall be relevant with 80% accuracy
- NFR-11.2: The system shall continuously improve recommendations based on usage
- NFR-11.3: Incorrect suggestions shall be minimized through feedback learning
- NFR-11.4: AI rescheduling shall consider user preferences and patterns

4.7.12 NFR-12: Logging and Monitoring Requirements

- NFR-12.1: System activities and errors shall be logged for troubleshooting
- NFR-12.2: Admins shall have access to real-time system performance metrics
- NFR-12.3: Logs shall help identify security issues and system failures
- NFR-12.4: Log retention period shall be at least 90 days

4.7.13 NFR-13: Backup and Recovery Requirements

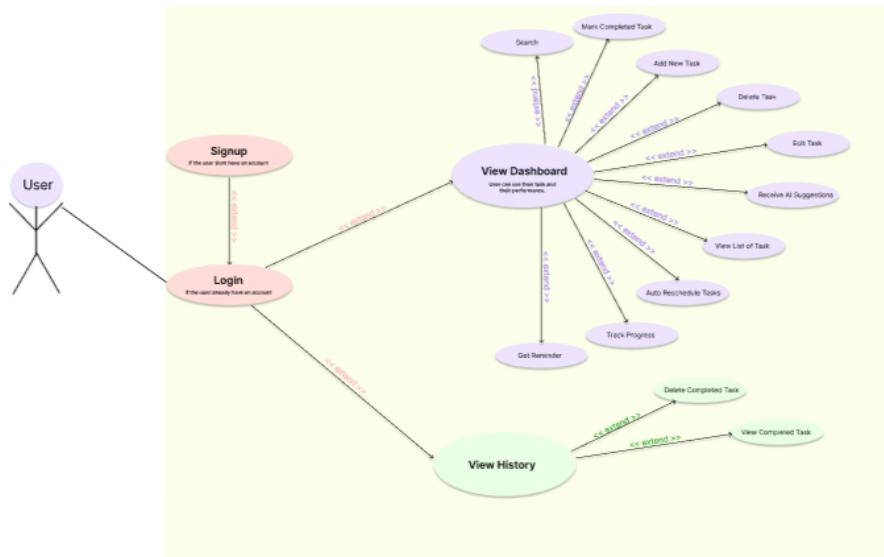
- NFR-13.1: Database backups shall be performed automatically every 24 hours
- NFR-13.2: The system shall support quick recovery in case of failure
- NFR-13.3: Backup data shall be stored securely in multiple locations
- NFR-13.4: Recovery time objective (RTO) shall be less than 2 hours
- NFR-13.5: Recovery point objective (RPO) shall be less than 24 hours

4.7.14 NFR-14: Response and Error Handling

- NFR-14.1: The system shall display clear error messages when issues occur
- NFR-14.2: Errors shall not cause complete system crashes
- NFR-14.3: Users shall be guided on resolving common issues
- NFR-14.4: Error messages shall not expose sensitive system information
- NFR-14.5: The system shall gracefully handle network disconnections

5 Use Case Diagram

5.1 Diagram

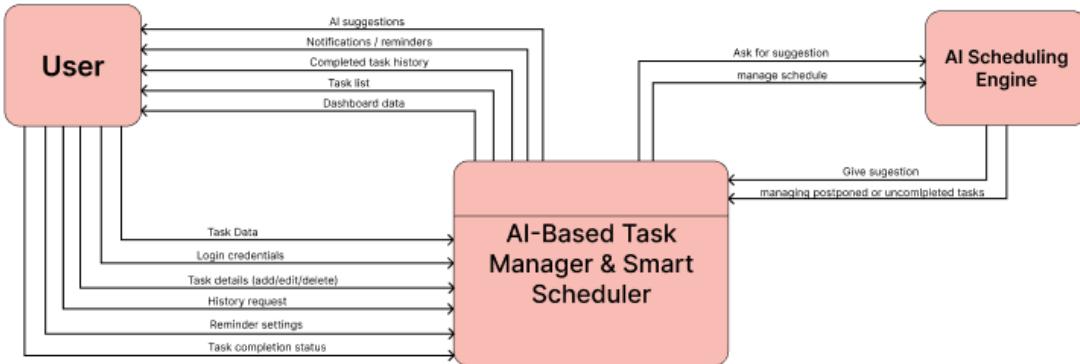


5.2 Link

Use Case Diagram Link

6 Context Diagram

6.1 Diagram



6.2 Link

Context Diagram Link

6.3 User to AI-Based Task Manager

Source: User

Destination: AI-Based Task Manager

Description: This data flow contains all information submitted by the user to interact with the system.

Data Elements:

- Registration data (username, email, password)
- Login credentials
- Task details (title, description, deadline, priority)
- Task update and deletion requests
- Reminder and notification settings
- Search keywords
- Progress/statistics requests
- Backup and restore requests

6.4 AI-Based Task Manager to User

Source: AI-Based Task Manager

Destination: User

Description: All responses, information, and notifications provided by the system to the user.

Data Elements:

- Task lists and schedule views
- Dashboard information

- Confirmation messages
- Reminder notifications
- AI productivity suggestions
- Progress and productivity reports
- Backup and restore confirmations

6.5 AI-Based Task Manager to AI

Source: AI-Based Task Manager

Destination: AI Module

Description: Task and schedule data sent to the AI module for intelligent analysis and optimization.

Data Elements:

- Pending and overdue tasks
- Task priorities
- User availability and free time slots
- Workload statistics
- Task history summaries

6.6 AI to AI-Based Task Manager

Source: AI Module

Destination: AI-Based Task Manager

Description: Intelligent outputs generated by the AI module.

Data Elements:

- Suggested time slots
- Auto-rescheduling decisions
- Productivity improvement suggestions
- Break reminders
- Optimized task sequences

6.7 Admin to AI-Based Task Manager

Source: Admin

Destination: AI-Based Task Manager

Description: Commands and configuration data used by administrators to manage the system.

Data Elements:

- Admin login credentials
- User account management commands
- System configuration settings
- Maintenance commands

6.8 AI-Based Task Manager to Admin

Source: AI-Based Task Manager

Destination: Admin

Description: Monitoring and reporting data provided to administrators.

Data Elements:

- User records
- System usage statistics
- Error and audit logs
- Backup status reports
- Maintenance reports

7 Appendices

7.1 Appendix A: Glossary

AI (Artificial Intelligence): Machine learning algorithms used for task rescheduling and smart suggestions

Authentication: The process of verifying user identity through credentials

Cloud Storage: Remote data storage accessed via the internet

Dashboard: The main user interface showing task overview and statistics

Deadline: The date and time by which a task must be completed

Firebase: A cloud-based platform for mobile app development

GDPR: General Data Protection Regulation for data privacy

Priority: The importance level assigned to a task (High, Medium, Low)

Rescheduling: The process of assigning a new deadline to a task

Session Token: A unique identifier for authenticated user sessions

SQLite: A lightweight relational database system

Stakeholder: Any person or entity with interest in the project

Use Case: A description of how a user interacts with the system

7.2 Appendix B: Analysis Models

The system interacts with the following external entities:

- Users (primary and administrative)
- Database servers
- Cloud storage services
- Notification services
- Third-party authentication providers

7.3 Appendix C: Issues List

Issue ID	Description	Status
I-01	Decision on biometric authentication support	Open
I-02	Integration with Google/Facebook login	Open
I-03	Voice command task creation feasibility	Under Review
I-04	Photo/document attachment to tasks	Open
I-05	Auto-logout timeout duration	Open
I-06	Task change history tracking requirement	Open
I-07	Deleted tasks trash vs permanent deletion	Under Review
I-08	Color-coding system for task priorities	Open
I-09	Recurring tasks support	Open
I-10	Mood tracking for AI suggestions	Future Version