

Comprehensive System Requirements Document

AI-Powered Test Case Generation and Management System

System Overview

This system represents a comprehensive web-based platform designed to revolutionize software testing through the integration of generative AI models. The platform serves as both a research tool and a practical solution for automated test case generation and management. Built on a modern technology stack utilizing React with Bootstrap for the frontend and Spring Boot for the backend, the system provides an intuitive user experience while maintaining robust backend processing capabilities.

The core functionality revolves around accepting Software Requirement Specification (SRS) documents in PDF and Word formats, processing them through various AI models, and generating comprehensive test cases organized into manageable modules. Users can interact with the system through a clean, responsive interface that guides them from initial project creation through detailed test case management and execution tracking.

The system architecture supports multiple AI model integrations including Microsoft Copilot, Google Gemini, ChatGPT, DeepSeek, Grok, and Claude, allowing for comparative analysis and research into the effectiveness of different AI approaches to test case generation. Each generated test case includes detailed attributes such as unique identifiers, descriptions, test steps, expected results, and status tracking capabilities.

The platform emphasizes user experience through its structured navigation flow, starting with an informative home screen that showcases project details and provides links to research repositories and papers. Users progress through a comprehensive authentication system to access personalized dashboards where they can manage multiple projects, each containing organized modules of test cases with real-time status tracking and commenting capabilities.

Export functionality supports multiple formats including JSON, PDF, Word documents, CSV, and Excel, ensuring compatibility with existing testing workflows and tools. The system maintains comprehensive audit trails and provides statistical insights into test case execution, helping teams track project progress and identify testing bottlenecks.

Security and performance are integral to the system design, with JWT-based authentication, role-based access control, and optimized database queries ensuring both data protection and responsive user interactions. The modular architecture supports scalability while maintaining code quality through comprehensive testing strategies and continuous integration practices.

1. Research Objectives & Questions

Primary Objective

Investigate the capability of generative AI models to automatically generate test cases and test management plans from Software Requirement Specification (SRS) documents and build a web system based on the research outcome.

Research Questions

- How accurately can generative AI models interpret an SRS document to produce valid test cases and a test management plan?
- What are the differences in efficiency (time and effort) between AI-generated and human-written test cases?
- How do AI-generated test cases compare to human-written ones in terms of test coverage and bug detection?
- What are the key challenges and limitations of using generative AI for test case generation?
- How can the findings inform the development of an automated testing system?

2. Technology Stack

Frontend

- Framework:** React.js
- UI Library:** Bootstrap for responsive design and components
- Additional Libraries:**
 - React Router for navigation
 - Axios for API calls
 - React Bootstrap for integrated components

Backend

- Framework:** Spring Boot (Java)
- Security:** Spring Security for authentication
- Data Access:** Spring Data JPA
- File Processing:** Apache POI for Word documents, PDF libraries for PDF processing

Database

- Primary Database:** MySQL
- ORM:** JPA/Hibernate

AI Model Integration

1. Microsoft Copilot
2. Google Gemini
3. ChatGPT (OpenAI API)
4. DeepSeek
5. Grok
6. Claude

3. System Architecture & Flow

High-Level System Flow

User Registration/Login → Dashboard → Create Project → Upload SRS →
Select AI Model → AI Processing → Generate Test Cases →
Module Organization → Test Case Management → Export Results

Detailed Application Flow

Phase 1: User Onboarding

1. **Home Screen** → User learns about system
2. **Registration/Login** → User creates account or logs in
3. **Dashboard Access** → User gains access to personal workspace

Phase 2: Project Creation

1. **Create New Project** → User initiates new testing project
2. **File Upload** → User uploads SRS documents (PDF/Word)
3. **AI Model Selection** → User chooses preferred AI model
4. **Custom Instructions** → User adds specific requirements
5. **AI Processing** → System processes SRS and generates test cases

Phase 3: Test Case Management

1. **Module Organization** → Test cases are grouped into modules
2. **Test Case Review** → User reviews generated test cases
3. **Status Management** → User updates test case status (Pending/Pass/Fail)
4. **Comments & Notes** → User adds observations and comments
5. **Export Options** → User exports results in various formats

4. Detailed Screen Requirements

4.1 Home Screen

Purpose: Landing page with project introduction and navigation

Components:

- **Header Navigation:** Home, Contact, Login, About, Dashboard (if logged in)
- **Hero Section:** Project title and description
- **Features Section:** Key system capabilities
- **Research Links:**
 - Link to GitHub repository
 - Link to research paper (PDF download)
- **Call-to-Action:** "Get Started" button leading to registration/login
- **Footer:** Contact information and additional links

Bootstrap Components: Navbar, Jumbotron, Cards, Buttons

4.2 Contact Screen

Purpose: Allow users to contact the development team

Components:

- **Contact Form:**
 - Name (required)
 - Email (required, validated)
 - Subject (dropdown with predefined options)
 - Message (textarea, required)
 - Submit button
- **Contact Information:**
 - Email addresses

- Institution details
- **Form Validation:** Client-side and server-side validation
- **Success Message:** Confirmation after successful submission

Bootstrap Components: Forms, Input Groups, Alerts

4.3 About Screen

Purpose: Information about the project and team

Components:

- **Project Background:** Research objectives and methodology
- **Team Section:**
 - **Abdul Basit:** AI model training, data analysis, model comparison
 - **Mah Noor:** System development and implementation
- **Technology Stack:** Overview of tools and frameworks used
- **Research Timeline:** Project milestones and deliverables

Bootstrap Components: Cards, Grid System, Badges

4.4 Login/Authentication System

Purpose: Secure user access and session management

Components:

- **Login Form:**
 - Username/Email field (required)
 - Password field (required, masked)
 - "Remember Me" checkbox
 - Login button
- **Additional Options:**
 - "Sign Up" link for new users
 - "Forgot Password" link
 - Social login options (optional)
- **Registration Form:**
 - Full Name (required)
 - Email (required, unique)
 - Username (required, unique)
 - Password (required, strength validation)
 - Confirm Password (required, match validation)
 - Terms & Conditions checkbox
- **Password Recovery:**
 - Email input for reset link
 - Token-based password reset

Security Features:

- JWT token-based authentication
- Password encryption (BCrypt)
- Session timeout handling
- CSRF protection

Bootstrap Components: Forms, Modals, Alerts, Input Groups

4.5 Dashboard Screen (Post-Login)

Purpose: Central hub for project management

Components:

- **Welcome Message:** Personalized greeting with user name
- **Quick Stats:**
 - Total projects count
 - Total test cases generated
 - Recent activity summary
- **Project List:**
 - Project name
 - Creation date
 - Last modified date
 - Test case count
 - Status overview (Pending/Pass/Fail counts)
 - Action buttons (View, Edit, Delete)
- **Create New Project Button:** Prominent call-to-action
- **Recent Activity Feed:** Last 10 actions across all projects
- **Profile Management:** Link to user profile settings

Bootstrap Components: Cards, Tables, Badges, Buttons, Progress Bars

4.6 Create New Project Screen

Purpose: Initialize new testing project with AI-generated test cases

Components:

- **Project Information:**
 - Project Name (required)
 - Project Description (optional)
 - Tags/Categories (optional)
- **File Upload Section:**
 - Drag & drop area for SRS files
 - File browser button
 - **Supported Formats:** PDF (.pdf), Word (.docx, .doc)
 - **File Size Limit:** Maximum 10MB per file
 - **File Validation:** Format checking, virus scanning
 - Upload progress indicator
- **AI Configuration:**
 - **AI Model Selection** (dropdown):
 - Microsoft Copilot
 - Google Gemini
 - ChatGPT
 - DeepSeek
 - Grok
 - Claude
 - **Custom Instructions** (textarea):
 - Additional requirements
 - Specific testing focus areas
 - Compliance standards (ISO/IEC/IEEE)
- **Processing Options:**
 - Test case priority levels
 - Module organization preferences
- **Action Buttons:**
 - "Generate Test Cases" (primary action)
 - "Save as Draft"
 - "Cancel"

AI Processing Flow:

1. File upload and validation
2. Document parsing and text extraction
3. Content preprocessing and formatting
4. AI model API call with SRS content and instructions
5. Response parsing and JSON formatting
6. Test case organization into modules
7. Database storage and user notification

Bootstrap Components: Forms, File Input, Progress Bars, Modals, Spinners

4.7 Project Management Screen

Purpose: Overview and navigation for individual projects

Components:

- **Project Header:**
 - Project name and description
 - Creation and last modified dates
 - AI model used for generation
- **Statistics Dashboard:**
 - **Total Test Cases:** Overall count
 - **Status Distribution:**
 - Pending (yellow badge)
 - Passed (green badge)
 - Failed (red badge)
 - **Completion Progress:** Visual progress bar
- **Module List:**
 - Module name
 - Test case count per module
 - Module description
 - Status summary per module
 - "View Module" button/link
- **Project Actions:**
 - Export entire project
 - Regenerate test cases
 - Edit project settings
 - Delete project
- **Quick Filters:**
 - Filter by status
 - Filter by module
 - Search test cases

Bootstrap Components: Cards, Progress Bars, Badges, Tables, Buttons

4.8 Module/Test Case Management Screen

Purpose: Detailed test case management within modules

Components:

- **Module Header:**
 - Module name and description

- Test case count
 - Module-specific statistics
- **Test Case Table:**
 - **Test Case Attributes:**
 - **Test ID:** Unique identifier (auto-generated)
 - **Test Case Title:** Descriptive name
 - **Description:** Detailed test case description
 - **Preconditions:** Setup requirements
 - **Test Steps:** Step-by-step instructions
 - **Expected Results:** Expected outcomes
 - **Actual Results:** User-input field
 - **Status:** Dropdown (Pending/Pass/Fail)
 - **Priority:** High/Medium/Low
 - **Comments:** User notes and observations
 - **Created Date:** Auto-populated
 - **Last Modified:** Auto-updated
 - **Assigned To:** Team member assignment
- **Test Case Actions:**
 - Edit test case details
 - Update status
 - Add/edit comments
 - Delete test case
 - Duplicate test case
- **Bulk Operations:**
 - Bulk status updates
 - Bulk export
 - Bulk delete
- **Filtering & Search:**
 - Status filter
 - Priority filter
 - Text search across all fields
 - Date range filter

Bootstrap Components: Tables, Forms, Modals, Dropdowns, Badges

5. Export Functionality

Supported Export Formats

- **JSON:** Machine-readable format for API integration
- **PDF:** Formatted report with project summary and test cases
- **Word Document:** Editable format for documentation
- **CSV:** Spreadsheet format for data analysis
- **Excel:** Advanced spreadsheet with multiple sheets

Export Options

- **Full Project Export:** All modules and test cases
- **Module-Specific Export:** Individual module data
- **Filtered Export:** Based on status, priority, or custom filters
- **Custom Template Export:** User-defined format templates

6. Database Schema Requirements

Core Tables

- **users:** User authentication and profile data
- **projects:** Project metadata and configuration
- **modules:** Project module organization
- **test_cases:** Detailed test case information
- **comments:** Test case comments and notes
- **ai_models:** Available AI model configurations
- **export_templates:** Custom export format templates
- **audit_log:** Change tracking and history

Relationships

- Users → Projects (One-to-Many)
- Projects → Modules (One-to-Many)
- Modules → Test Cases (One-to-Many)
- Test Cases → Comments (One-to-Many)

7. API Endpoints Structure

Authentication Endpoints

- POST /api/auth/login
- POST /api/auth/register
- POST /api/auth/logout
- POST /api/auth/forgot-password
- POST /api/auth/reset-password

Project Management Endpoints

- GET /api/projects - List user projects
- POST /api/projects - Create new project
- GET /api/projects/{id} - Get project details
- PUT /api/projects/{id} - Update project
- DELETE /api/projects/{id} - Delete project

File Upload Endpoints

- POST /api/upload/srs - Upload SRS document
- GET /api/upload/status/{uploadId} - Check upload status

AI Integration Endpoints

- POST /api/ai/generate - Generate test cases
- GET /api/ai/models - List available AI models
- GET /api/ai/status/{jobId} - Check generation status

Test Case Management Endpoints

- GET /api/projects/{id}/modules - List project modules
- GET /api/modules/{id}/testcases - List module test cases
- PUT /api/testcases/{id} - Update test case
- POST /api/testcases/{id}/comments - Add comment

Export Endpoints

- GET /api/export/project/{id} - Export project
- GET /api/export/module/{id} - Export module
- POST /api/export/custom - Custom export

8. Security Requirements

Authentication & Authorization

- JWT-based authentication
- Role-based access control
- Session timeout management
- Password strength validation
- Account lockout after failed attempts

Data Security

- HTTPS enforcement
- Input validation and sanitization
- SQL injection prevention
- XSS protection
- File upload security scanning
- Data encryption at rest

Privacy & Compliance

- User data privacy protection
- GDPR compliance considerations
- Data retention policies
- Audit trail maintenance

9. Performance Requirements

Response Time Targets

- Page load time: < 3 seconds
- API response time: < 500ms
- File upload processing: < 30 seconds for 10MB files
- AI model response: < 2 minutes for typical SRS documents

Scalability Requirements

- Support for 100+ concurrent users
 - Handle 1000+ projects per user
 - Process 10,000+ test cases per project
 - Database optimization for large datasets
-

10. Integration Requirements

AI Model Integration

- RESTful API integration for all supported models
- Error handling and fallback mechanisms
- Rate limiting and quota management
- Response format standardization

Third-Party Services

- Email service for notifications
 - File storage service for document handling
 - Analytics service for usage tracking
 - Backup service for data protection
-

11. Testing & Quality Assurance

Testing Strategy

- Unit testing for all components
- Integration testing for API endpoints
- End-to-end testing for user workflows
- Performance testing for scalability
- Security testing for vulnerabilities

Quality Metrics

- Code coverage > 80%
 - API response time < 500ms
 - UI responsiveness across devices
 - Accessibility compliance (WCAG 2.1)
-

12. Deployment & DevOps

Environment Setup

- Development environment
- Staging environment
- Production environment
- CI/CD pipeline configuration

Monitoring & Logging

- Application performance monitoring
 - Error tracking and alerting
 - User activity logging
 - System health monitoring
-

13. Team Responsibilities

Abdul Basit

- AI model training and optimization
- Data analysis and model comparison
- Research methodology implementation
- Performance evaluation metrics

Mah Noor

- Full-stack system development
 - Database design and implementation
 - API development and integration
 - Frontend UI/UX implementation
 - System deployment and maintenance
-

14. Project Timeline & Milestones

Phase 1: Foundation (Weeks 1-4)

- System architecture design
- Database schema implementation
- Basic authentication system
- Core API endpoints

Phase 2: Core Features (Weeks 5-8)

- File upload functionality
- AI model integration
- Test case generation
- Basic project management

Phase 3: Advanced Features (Weeks 9-12)

- Test case management
- Export functionality
- User interface polish
- Performance optimization

Phase 4: Testing & Deployment (Weeks 13-16)

- Comprehensive testing
- Bug fixes and improvements
- Production deployment
- Documentation and training

This comprehensive requirements document provides a complete roadmap for developing the AI-powered test case generation and management system, ensuring all aspects are clearly defined and actionable.