

Software Requirements Specification (SRS)

Spartex AI Homepage - Corporate Website Application

Version: 1.0

Date: October 06, 2025

Project: Spartex AI Corporate Website

Technology Stack: Angular 19

Document Status: Draft for Review

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document provides a comprehensive description of the functional and non-functional requirements for the Spartex AI corporate website homepage. The document is intended for:

- **Business Stakeholders:** To validate that the solution aligns with business objectives and marketing goals
- **Development Team:** To understand the complete scope of features, user interfaces, and technical requirements
- **Quality Assurance Team:** To develop comprehensive test cases and validation criteria
- **Project Managers:** To plan resources, timelines, and deliverables
- **UI/UX Designers:** To create wireframes and visual designs consistent with requirements

The primary purpose of this system is to establish a professional online presence for Spartex AI, showcase the company's AI-as-a-Service offerings, demonstrate industry expertise, and generate qualified leads through an intuitive and engaging user experience.

1.2 Scope

Product Name: Spartex AI Corporate Website

Product Tagline: "AI as a Service – Empowering Your Business with Intelligent Solutions"

In-Scope:

The Spartex AI Homepage application is a single-page application (SPA) built using Angular 19 that will:

- Present comprehensive information about Spartex AI's services, process, and value proposition

- Showcase four core service offerings: Multi-agent AI Solutions, AI Chatbots & Virtual Assistants, LLM Integration, and Custom AI Product Development
- Highlight six target industry verticals: Retail, Healthcare, Finance, Logistics, Professional Services, and Legal
- Display three case studies demonstrating proven success and ROI metrics
- Provide detailed company background including founding story, vision, and mission
- Facilitate user engagement through a contact form and multiple communication channels
- Enable user authentication through Login and Sign Up functionality
- Deliver a responsive, mobile-first design optimized for all device types
- Implement SEO best practices for improved search engine visibility
- Ensure accessibility compliance with WCAG 2.1 Level AA standards

Out-of-Scope:

The following elements are explicitly excluded from the current phase:

- Backend API development and database implementation
- User account management system and dashboard
- E-commerce functionality or payment processing
- Content Management System (CMS) integration
- Multi-language support and internationalization
- Real-time chat functionality
- Blog or news content management
- Client portal or authenticated user areas
- Integration with third-party CRM systems
- Advanced analytics and reporting dashboards

1.3 Definitions, Acronyms, Abbreviations

Term	Definition
AI	Artificial Intelligence - Computer systems able to perform tasks that typically require human intelligence
API	Application Programming Interface - A set of protocols for building and integrating application software
CTA	Call-to-Action - Design element encouraging user interaction (e.g., "Get Started" button)
LLM	Large Language Model - AI models trained on vast amounts of text data for natural language processing
POC	Proof of Concept - A demonstration to verify that certain concepts have the potential for real-world application

Term	Definition
R&D	Research and Development - Creative work undertaken to increase knowledge and devise new applications
ROI	Return on Investment - A performance measure used to evaluate the efficiency of an investment
SEO	Search Engine Optimization - Process of improving website visibility in search engine results
SMB	Small and Medium Business - Companies with employee counts or revenue below certain thresholds
SPA	Single Page Application - Web application that loads a single HTML page and dynamically updates content
SRS	Software Requirements Specification - A comprehensive description of intended purpose and environment for software
UI	User Interface - The space where interactions between humans and machines occur
UX	User Experience - The overall experience of a person using a product, especially in terms of ease and satisfaction
WCAG	Web Content Accessibility Guidelines - International standards for web accessibility

1.4 References

- Angular 19 Official Documentation: <https://angular.dev>
- Material Design Guidelines: <https://material.io/design>
- WCAG 2.1 Accessibility Standards: <https://www.w3.org/WAI/WCAG21/quickref/>
- W3C HTML5 Specification: <https://www.w3.org/TR/html52/>
- TypeScript Language Specification: <https://www.typescriptlang.org/docs/>
- Spartex AI Brand Guidelines (Internal Document - To Be Provided)
- Spartex AI Marketing Strategy Document (Internal Document - To Be Provided)

1.5 Overview

This SRS document is organized into seven major sections:

Section 2 - Overall Description: Provides context for the system including product perspective, core functions, user characteristics, operating environment, and key assumptions.

Section 3 - System Features: Details all functional requirements organized by major features and user interactions, including inputs, processing logic, and expected outputs.

Section 4 - External Interface Requirements: Specifies user interface designs, hardware requirements, software dependencies, and communication protocols.

Section 5 - Non-Functional Requirements: Defines quality attributes including performance metrics, reliability standards, security requirements, scalability expectations, and usability criteria.

Section 6 - Other Requirements: Addresses compliance obligations, data management strategies, and operational considerations.

Section 7 - Appendices: Provides supplementary information including glossary terms and reference materials.

2. Overall Description

2.1 Product Perspective

The Spartex AI Homepage is a standalone, self-contained single-page web application that serves as the primary digital storefront and lead generation tool for Spartex AI. The system operates independently as a marketing and information platform, though it is designed with future integration capabilities in mind.

System Context:

- **Position in Ecosystem:** The homepage acts as the initial touchpoint in the customer acquisition funnel, bridging the gap between digital marketing campaigns and qualified sales opportunities.
- **Relationship to Other Systems:** While currently standalone, the architecture supports future integration with:
 - CRM systems for lead management
 - Email marketing platforms for nurture campaigns
 - Analytics platforms for conversion tracking
 - Backend authentication services for user account management
- **Design Philosophy:** Built on Angular 19 framework utilizing component-based architecture, the system emphasizes modularity, reusability, and maintainability. Each section of the homepage is implemented as a discrete Angular component, enabling independent development, testing, and future enhancement.
- **Responsive Design:** The application follows a mobile-first approach, ensuring optimal user experience across all device categories (smartphones, tablets, laptops, desktops, and large displays).
- **Browser Compatibility:** Designed to function seamlessly across modern web browsers including Chrome, Firefox, Safari, Edge, and Opera (latest two versions).

2.2 Product Functions

The Spartex AI Homepage provides the following core functions:

F1: Brand Presentation & Navigation

- Display company branding with logo and microchip icon
- Provide intuitive navigation menu for quick access to all sections

- Enable smooth scrolling navigation within the single-page interface
- Present authentication options (Login/Sign Up) for future user account access

F2: Value Proposition Communication

- Showcase the four-step business understanding process (R&D, Prototypes/POCs, Understanding, Solution Design)
- Highlight competitive advantages and unique approach
- Present clear calls-to-action for user engagement ("Get Started" and "Learn More")

F3: Service Portfolio Display

- Present four core service offerings with detailed descriptions:
 - Multi-agent AI Solutions
 - AI Chatbots & Virtual Assistants
 - LLM Integration
 - Custom AI Product Development
- Enable users to access detailed information about each service

F4: Industry Expertise Demonstration

- Showcase expertise across six industry verticals
- Provide industry-specific use cases and applications
- Demonstrate relevance to target audience segments

F5: Social Proof & Credibility

- Display three detailed case studies with quantifiable results
- Present ROI metrics and success indicators
- Build trust through demonstrated track record

F6: Company Background Information

- Share startup founding story and timeline
- Communicate company vision regarding AI democratization
- Articulate mission statement focused on SMB empowerment
- Display team imagery to humanize the brand

F7: Multi-Channel Communication

- Provide structured contact form with validation
- Display multiple contact methods (email, phone, physical address)
- Show business hours for customer reference
- Enable direct message submission

F8: Information Architecture

- Present comprehensive site navigation through footer links
- Organize content into logical categories (Company, Services, Industries, Contact)
- Provide social media integration points
- Display copyright and legal information

2.3 User Classes & Characteristics

UC1: Business Decision Makers (Primary User Class)

- **Description:** C-level executives, business owners, and department heads at small and medium businesses seeking AI solutions
- **Characteristics:**
 - Age range: 35-60 years
 - High business acumen, moderate technical literacy
 - Time-constrained, seeking quick information gathering
 - Value-focused with emphasis on ROI and business outcomes
- **Frequency of Use:** First-time visitors primarily, occasional return visits for information verification
- **Technical Expertise:** Basic to intermediate computer skills, comfortable with standard web browsing
- **Goals:**
 - Quickly understand service offerings and value proposition
 - Assess company credibility and expertise
 - Determine relevance to their specific industry and use case
 - Initiate contact with sales team

UC2: Technical Evaluators (Secondary User Class)

- **Description:** IT managers, CTOs, technical directors responsible for evaluating AI solution providers
- **Characteristics:**
 - Age range: 28-55 years
 - High technical literacy and AI/ML awareness

- Detail-oriented, seeking technical depth and implementation approach
- Focused on integration capabilities and technical feasibility
- **Frequency of Use:** Multiple visits during evaluation phase
- **Technical Expertise:** Advanced technical knowledge, familiar with AI/ML concepts
- **Goals:**
 - Evaluate technical approach and methodology
 - Understand integration capabilities and technical stack
 - Review case studies for technical implementation details
 - Assess company technical credibility

UC3: Operational Managers (Tertiary User Class)

- **Description:** Operations managers, department heads, and process owners exploring AI-driven efficiency improvements
- **Characteristics:**
 - Age range: 30-50 years
 - Moderate technical literacy, process-improvement focused
 - Seeking practical applications and implementation timelines
 - Budget-conscious with focus on operational ROI
- **Frequency of Use:** Occasional visits during problem-solving or initiative planning
- **Technical Expertise:** Basic to intermediate technical understanding
- **Goals:**
 - Identify specific solutions for operational challenges
 - Understand implementation process and timeline
 - Review industry-specific applications
 - Assess compatibility with existing workflows

UC4: Job Seekers & Potential Partners (Peripheral User Class)

- **Description:** Professionals seeking career opportunities or partnership opportunities with Spartex AI
- **Characteristics:**
 - Varied age range and backgrounds
 - Interest in company culture, values, and opportunities
 - Looking for company background and team information

- **Frequency of Use:** Limited, targeted visits
- **Technical Expertise:** Varies widely
- **Goals:**
 - Learn about company culture and mission
 - Access career or partnership contact information
 - Understand company positioning and growth trajectory

2.4 Operating Environment

Client-Side Environment:

- **Framework:** Angular 19.x (latest stable version)
- **Language:** TypeScript 5.x
- **Build Tool:** Angular CLI with Vite bundler
- **Package Manager:** npm or yarn

Supported Browsers:

- Google Chrome (latest 2 versions) - Primary target
- Mozilla Firefox (latest 2 versions)
- Apple Safari (latest 2 versions)
- Microsoft Edge (latest 2 versions)
- Opera (latest version)

Device Categories:

- Mobile phones: 320px - 767px width
- Tablets: 768px - 1023px width
- Laptops/Desktops: 1024px - 1919px width
- Large displays: 1920px and above

Network Requirements:

- Minimum bandwidth: 2 Mbps for optimal experience
- Application optimized for 3G/4G mobile networks
- Graceful degradation for slower connections

Screen Resolutions:

- Minimum supported: 320px × 568px (iPhone SE)
- Optimal: 1920px × 1080px (Full HD)
- Maximum tested: 3840px × 2160px (4K)

Hosting Environment:

- Static file hosting compatible (AWS S3, Netlify, Vercel, Azure Static Web Apps)
- CDN delivery for global performance optimization
- HTTPS protocol mandatory for production deployment

2.5 Design & Implementation Constraints

Technical Constraints:

- **Framework Dependency:** Application must be built exclusively using Angular 19 framework and follow Angular best practices and style guide
- **Component Architecture:** Must utilize standalone components (Angular 19 feature) for improved modularity
- **State Management:** Application state should be managed through Angular services with dependency injection
- **Routing:** Must implement Angular Router for navigation even within single-page structure to enable deep linking
- **Build Size:** Initial bundle size should not exceed 500KB (gzipped) for optimal load performance
- **Browser APIs:** Must utilize only standardized Web APIs supported across all target browsers

Design Constraints:

- **Responsive Design:** Must implement mobile-first responsive design approach with breakpoints at 768px, 1024px, and 1440px
- **Accessibility:** Must comply with WCAG 2.1 Level AA standards for accessibility
- **Brand Guidelines:** All visual elements must conform to Spartex AI brand guidelines (colors, typography, spacing)
- **Loading Performance:** First Contentful Paint must occur within 1.5 seconds on 4G connection
- **Visual Consistency:** Must maintain consistent visual language across all sections with bordered/boxed design elements

Business Constraints:

- **Timeline:** Project must be completed within defined sprint cycles (specific timeline to be provided by project management)
- **Budget:** Development must utilize open-source libraries and frameworks to minimize licensing costs
- **Content Management:** Content is hardcoded in initial phase; must be structured for future CMS integration
- **Scalability Planning:** Architecture must accommodate future addition of authenticated user areas, blog, and client portal

Regulatory Constraints:

- **Data Privacy:** Must implement cookie consent mechanisms for GDPR compliance
- **Accessibility:** Must meet ADA (Americans with Disabilities Act) compliance requirements
- **Data Collection:** Contact form submissions must include appropriate privacy notice and consent mechanisms
- **Copyright:** All content, images, and branding must respect intellectual property rights

Integration Constraints:

- **Future Authentication:** UI must accommodate integration with OAuth 2.0 / OpenID Connect authentication services
- **Analytics:** Must include integration points for Google Analytics or similar analytics platforms
- **Form Submission:** Contact form must be designed to integrate with backend API endpoint (endpoint specification to be provided)
- **Third-Party Scripts:** Must support integration of marketing pixels and tracking scripts without performance degradation

2.6 Assumptions & Dependencies

Assumptions:

1. **Content Availability:** All marketing copy, service descriptions, case study details, and company information will be provided by the Spartex AI marketing team in final form before development completion
2. **Visual Assets:** High-quality images, icons, and logo files will be delivered in web-optimized formats (SVG for icons/logos, WebP/JPEG for photographs)
3. **Browser Compatibility:** Users will access the application using modern, updated browsers; no support required for Internet Explorer
4. **Internet Connectivity:** Users have stable internet connection; offline functionality is not required

5. **Contact Information Accuracy:** Phone number (4142750857), email (contact.support@spartexai.com), and physical address information provided is accurate and current
6. **Form Backend:** Backend API endpoint for contact form submission will be developed separately and specification provided
7. **Authentication Services:** Login and Sign Up functionality will connect to separately developed authentication service
8. **Mobile Usage:** Significant portion of traffic (estimated 40-50%) will access site via mobile devices
9. **SEO Requirements:** Basic SEO implementation sufficient for initial launch; advanced SEO optimization will be iterative
10. **Performance Targets:** Application will be hosted on modern cloud infrastructure with CDN support

Dependencies:

External Dependencies:

1. **Angular Framework:** Angular 19.x framework and associated Angular libraries
2. **Node.js Ecosystem:** Node.js runtime environment for build tools and development server
3. **NPM Packages:**
 - RxJS for reactive programming
 - Angular Router for navigation
 - Angular Forms for form validation
 - Third-party UI component libraries (if approved)
4. **Web Fonts:** Google Fonts or similar font delivery service for typography
5. **Icon Libraries:** Font Awesome, Material Icons, or similar for UI icons
6. **Browser Standards:** HTML5, CSS3, ECMAScript 2022+ support in target browsers

Internal Dependencies:

1. **Content Delivery:** Marketing team must deliver finalized content copy and messaging
2. **Design Assets:** Design team must provide:
 - Brand style guide and design system
 - Icon set in SVG format
 - Case study images and team photographs
 - Logo variants for different backgrounds
3. **Backend Services:** Development team must provide:
 - Contact form submission API endpoint specification

- Authentication API endpoint specification
- CORS configuration for API access

4. **Infrastructure:** DevOps team must provide:

- Hosting environment configuration
- CDN setup and configuration
- SSL certificate provisioning
- Domain name configuration

5. **Testing Resources:** QA team requires:

- Access to staging environment
- Device testing lab or BrowserStack access
- Test data for form submission validation

Risk Dependencies:

1. **Third-Party Service Availability:** Application assumes availability of CDN, font services, and icon libraries
 2. **Browser Updates:** Application assumes stable APIs; breaking changes in browser updates could require code modifications
 3. **Framework Updates:** Angular framework updates may require code refactoring; version will be locked for production
 4. **Content Changes:** Late-stage content modifications may impact development timeline
 5. **Design Changes:** Significant design revisions may impact implementation schedule and technical approach
-

3. System Features

3.1 Functional Requirements

FR-1: HEADER SECTION (Top Navigation Bar)

Priority: High

Description: The header provides persistent navigation and branding across all page sections.

FR-1.1: Logo Display

- **Requirement:** System shall display the Spartex AI logo consisting of a microchip icon and "Spartex AI" text within a bordered box design

- **Input:** None (static display)
- **Processing:** Logo component renders SVG or image asset with appropriate styling
- **Output:** Visible logo in top-left corner of viewport
- **Acceptance Criteria:**
 - Logo is visible on all viewport sizes
 - Logo maintains aspect ratio and clarity at all resolutions
 - Logo serves as clickable link to scroll to top of page
 - Logo uses brand-approved colors and dimensions

FR-1.2: Navigation Menu

- **Requirement:** System shall provide a horizontal navigation menu with four links: Home, Services, Research, and Contact
- **Input:** User click or tap on navigation link
- **Processing:** Application scrolls to corresponding page section using smooth scroll behavior
- **Output:** Page scrolls to target section with appropriate offset for fixed header
- **Acceptance Criteria:**
 - All four navigation links are visible and clearly labeled
 - Navigation links trigger smooth scroll animation
 - Active section is highlighted in navigation menu
 - Navigation menu collapses to hamburger menu on mobile devices ($< 768\text{px}$)
 - Expanded mobile menu displays links vertically with appropriate spacing
 - Mobile menu closes automatically after link selection

FR-1.3: Authentication Buttons

- **Requirement:** System shall display Login (outlined) and Sign Up (filled) buttons on the right side of header
- **Input:** User click on Login or Sign Up button
- **Processing:** Application navigates to respective authentication page or modal
- **Output:** Authentication interface displays (implementation to be defined in future phase)
- **Acceptance Criteria:**
 - Both buttons are visible on desktop and tablet viewports ($\geq 768\text{px}$)
 - Login button has outlined style (border with transparent background)

- Sign Up button has filled style (solid background with brand color)
- Buttons stack vertically or hide on small mobile devices (< 768px)
- Hover states provide visual feedback
- Buttons are keyboard accessible

FR-1.4: Header Behavior

- **Requirement:** System shall maintain header visibility during page scroll
- **Input:** User scrolls page content
- **Processing:** Header remains fixed at top of viewport
- **Output:** Header visible at all scroll positions
- **Acceptance Criteria:**
 - Header uses fixed positioning and remains visible during scroll
 - Header includes subtle shadow or border to distinguish from content
 - Header maintains consistent height across all viewports
 - Z-index ensures header appears above all other content
 - Header transitions smoothly when scrolling begins

FR-2: HERO SECTION

Priority: High

Description: The hero section communicates the core value proposition and company process.

FR-2.1: Brand Headline Display

- **Requirement:** System shall prominently display "Spartex AI" in a large bordered box with tagline "AI as a Service – Empowering Your Business with Intelligent Solutions"
- **Input:** None (static display)
- **Processing:** Hero component renders headline with appropriate typography hierarchy
- **Output:** Visually prominent headline area
- **Acceptance Criteria:**
 - Headline uses largest font size on page (4xl or larger)
 - Bordered box design consistent with brand guidelines
 - Tagline positioned directly below main headline with adequate spacing
 - Text remains readable across all viewport sizes
 - Headline is semantically marked as H1 for SEO

FR-2.2: Secondary Headline

- **Requirement:** System shall display "What Our Approach & Why Choose Us" as secondary message
- **Input:** None (static display)
- **Processing:** Secondary headline renders with medium emphasis typography
- **Output:** Visible secondary headline
- **Acceptance Criteria:**
 - Uses H2 semantic markup
 - Visually distinct from main headline through size/weight
 - Adequate spacing from main headline and process section

FR-2.3: Process Steps Display

- **Requirement:** System shall present four-step business understanding process in bordered boxes with numbered badges
- **Input:** None (static display)
- **Processing:** Process component renders four steps with consistent styling
- **Output:** Visible process steps in grid or linear layout
- **Steps:**
 1. Step 1: R&D for Your Business - Circle badge "1" with description
 2. Step 2: Prototypes and POCs - Circle badge "2" with description
 3. Step 3: Understand Better Way - Circle badge "3" with description
 4. Step 4: Design Solution - Circle badge "4" with description
- **Acceptance Criteria:**
 - Section titled "Our Process to Understand Your Business"
 - Four steps display in logical order (1-4)
 - Each step includes circular numbered badge
 - Each step includes bordered box with descriptive text
 - Layout adapts responsively (grid on desktop, stack on mobile)
 - Visual flow indicators connect steps (optional: arrows or connectors)
 - Consistent spacing between all steps

FR-2.4: Call-to-Action Buttons

- **Requirement:** System shall provide two CTAs: "Get Started" (primary) and "Learn More" (secondary)

- **Input:** User click on CTA button
- **Processing:**
 - "Get Started": Scrolls to Contact section
 - "Learn More": Scrolls to Services section
- **Output:** Page navigates to respective section
- **Acceptance Criteria:**
 - "Get Started" uses filled button style with high contrast
 - "Learn More" uses outlined button style
 - Buttons positioned side-by-side on desktop, stacked on mobile
 - Hover states provide clear visual feedback
 - Buttons are keyboard and screen reader accessible
 - Sufficient size for touch targets (minimum 44x44px)

FR-3: SERVICES SECTION

Priority: High

Description: The services section showcases the four core AI service offerings.

FR-3.1: Section Header

- **Requirement:** System shall display section title "Our Services" with subtitle "Specialized AI solutions designed for small and medium businesses"
- **Input:** None (static display)
- **Processing:** Section header component renders with appropriate hierarchy
- **Output:** Visible section header
- **Acceptance Criteria:**
 - Main title uses H2 semantic markup
 - Subtitle provides context for service offerings
 - Centered alignment with adequate spacing
 - Visually distinguishes section from hero area above

FR-3.2: Service Cards Grid

- **Requirement:** System shall display four service cards in responsive grid layout
- **Input:** User hover/focus on card or click "Learn More" link
- **Processing:**

- **Hover:** Card displays elevation/highlight effect
- **Click:** Navigate to detailed service page (future implementation)
- **Output:** Interactive service cards with visual feedback
- **Services:**
 1. Multi-agent AI Solutions (Network/nodes icon)
 2. AI Chatbots & Virtual Assistants (Chat bubbles icon)
 3. LLM Integration (Brain icon)
 4. Custom AI Product Development (Code icon)
- **Acceptance Criteria:**
 - Four cards display in 2x2 grid on desktop ($\geq 1024\text{px}$)
 - Cards display in 2x2 or 1x4 layout on tablet (768-1023px)
 - Cards stack vertically on mobile ($< 768\text{px}$)
 - Each card includes: icon in bordered box, title, description in dashed border box, "Learn More" link with arrow
 - Consistent card height within each row
 - Cards have hover state with subtle animation (lift, shadow, or border highlight)
 - Icons are consistently sized and styled
 - "Learn More" links include directional arrow icon
 - Cards maintain readability and visual hierarchy

FR-3.3: Service Card Interactions

- **Requirement:** System shall provide interactive feedback on service cards
- **Input:** User interaction (hover, focus, click)
- **Processing:** Card applies interactive state styling
- **Output:** Visual feedback confirming interaction
- **Acceptance Criteria:**
 - Smooth transition animations (200-300ms)
 - Touch-friendly for mobile devices
 - Keyboard navigation support (tab order)
 - Focus indicators visible for accessibility
 - Click target encompasses entire card area

FR-4: INDUSTRIES SECTION

Priority: Medium

Description: The industries section demonstrates domain expertise across six vertical markets.

FR-4.1: Section Header

- **Requirement:** System shall display "Industries We Serve - Small & Medium Businesses" with subtitle "Specialized AI solutions for various industry sectors"
- **Input:** None (static display)
- **Processing:** Section header renders with appropriate styling
- **Output:** Visible section header
- **Acceptance Criteria:**
 - Main title uses H2 semantic markup
 - Subtitle provides additional context
 - Centered alignment with consistent spacing

FR-4.2: Industry Cards Grid

- **Requirement:** System shall display six industry cards in responsive grid layout
- **Input:** User views industry cards
- **Processing:** Cards render with industry-specific icons and descriptions
- **Output:** Grid of industry cards
- **Industries:**
 1. Retail - Shopping cart icon - "Inventory management, customer service, and sales forecasting"
 2. Healthcare - Heartbeat/medical icon - "Patient scheduling, diagnostic support, and administrative automation"
 3. Finance - Bank/university building icon - "Fraud detection, customer service, and financial forecasting"
 4. Logistics - Truck icon - "Route optimization, demand forecasting, and supply chain management"
 5. Professional Services - Briefcase icon - "Client management, document processing, and workflow automation"
 6. Attorneys and Lawyers - Gavel/legal icon - "Document analysis, case research, and client communication"
- **Acceptance Criteria:**
 - Six cards display in 3x2 grid on desktop ($\geq 1024\text{px}$)
 - Cards display in 2x3 grid on tablet (768-1023px)

- Cards stack vertically on mobile (< 768px)
- Each card includes: industry icon in bordered box, industry title, description text
- Consistent card dimensions and spacing
- Icons are thematically appropriate and recognizable
- Cards have subtle hover effect for interactivity
- Text remains readable across all viewport sizes

FR-5: CASE STUDIES SECTION

Priority: Medium

Description: The case studies section provides social proof through detailed success stories.

FR-5.1: Section Header

- **Requirement:** System shall display "Case Studies & Success Stories" with subtitle "Discover how our AI solutions have transformed small and medium businesses"
- **Input:** None (static display)
- **Processing:** Section header renders with standard formatting
- **Output:** Visible section header
- **Acceptance Criteria:**
 - Main title uses H2 semantic markup
 - Subtitle provides context and encouragement
 - Centered alignment with adequate spacing

FR-5.2: Case Study Cards

- **Requirement:** System shall display three case study cards with images, results, and call-to-action links
- **Input:** User clicks "Read Case Study" link
- **Processing:** Navigate to detailed case study page (future implementation)
- **Output:** Interactive case study cards
- **Case Studies:**
 1. **Retail Chain Optimization**
 - Image: Store icon or retail environment
 - Results: 35% cost reduction, 28% satisfaction improvement
 2. **Manufacturing Efficiency**
 - Image: Handshake icon or manufacturing setting

- Results: 42% output increase

3. Healthcare Provider Transformation

- Image: Hospital/patient icon or healthcare setting
- Results: 60% reduction in no-shows

- **Acceptance Criteria:**

- Three cards display in row on desktop ($\geq 1024\text{px}$)
- Cards stack vertically on tablet and mobile ($< 1024\text{px}$)
- Each card includes: image placeholder area with icon, case study title, results in dashed border box, "Read Case Study" link with arrow
- Results highlight quantifiable metrics prominently
- Images are optimized for web delivery (WebP format preferred)
- Cards maintain consistent height on same row
- "Read Case Study" links include directional arrow
- Hover states provide visual feedback

FR-5.3: Results Display

- **Requirement:** System shall prominently display quantifiable results within each case study card
- **Input:** None (static display)
- **Processing:** Results render with emphasis on numeric values
- **Output:** Visible metrics with visual hierarchy
- **Acceptance Criteria:**
 - Numeric values (35%, 42%, 60%) are visually emphasized
 - Results are contained within dashed border boxes
 - Metrics are easily scannable
 - Color coding or icons enhance readability (optional)

FR-6: ABOUT SECTION

Priority: Medium

Description: The about section builds trust through company background and mission.

FR-6.1: Two-Column Layout

- **Requirement:** System shall present about content in two-column layout with text on left and team image on right

- **Input:** None (static display)
- **Processing:** Layout adapts responsively across viewport sizes
- **Output:** Balanced two-column presentation
- **Acceptance Criteria:**
 - Two columns on desktop ($\geq 1024\text{px}$)
 - Stacked layout on tablet and mobile ($< 1024\text{px}$)
 - Columns have equal or proportional width
 - Adequate spacing between columns
 - Text column maintains comfortable reading line length

FR-6.2: About Text Content

- **Requirement:** System shall display "About Us" heading with three subsections: Our Startup Story, Vision, and Mission
- **Input:** None (static display)
- **Processing:** Content renders with appropriate typography and spacing
- **Output:** Structured about text
- **Content Structure:**
 - Main heading: "About Us"
 - **Our Startup Story:** Background paragraph about founding in 2023
 - **Vision & Mission:** Two paragraphs
 - Vision: Democratizing AI for small and medium businesses
 - Mission: Empowering SMBs with accessible, effective AI solutions
- **Acceptance Criteria:**
 - "About Us" uses H2 semantic markup
 - Subsection headings use H3 markup
 - Startup story in dashed border box
 - Vision and mission in separate dashed border boxes
 - Comfortable reading typography (line height, spacing)
 - Content builds narrative arc from founding to future vision

FR-6.3: Team Image Display

- **Requirement:** System shall display team photograph showing collaborative workspace environment

- **Input:** None (static display)
- **Processing:** Image renders responsively with appropriate aspect ratio
- **Output:** Visible team photograph
- **Acceptance Criteria:**
 - Image contained in bordered box consistent with design system
 - Image is optimized for web (WebP format preferred)
 - Responsive image sizing (srcset for different resolutions)
 - Alt text provided for accessibility
 - Image maintains aspect ratio without distortion
 - Image loading is optimized (lazy loading)

FR-7: CONTACT SECTION

Priority: High

Description: The contact section facilitates lead generation through multiple communication channels.

FR-7.1: Two-Column Layout

- **Requirement:** System shall present contact information in two-column layout with contact details on left and contact form on right
- **Input:** None (static display)
- **Processing:** Layout adapts responsively across viewport sizes
- **Output:** Balanced two-column contact presentation
- **Acceptance Criteria:**
 - Two columns on desktop ($\geq 1024\text{px}$)
 - Stacked layout on tablet and mobile ($< 1024\text{px}$)
 - Form column appears first on mobile for priority access
 - Adequate spacing between columns
 - Equal or proportional column widths

FR-7.2: Contact Information Display

- **Requirement:** System shall display "Contact Us" heading with introductory text and four contact methods
- **Input:** User clicks on contact information (email, phone)
- **Processing:** System initiates appropriate action (email client, phone dialer)
- **Output:** Contact information with actionable links

- **Contact Methods:**
 1. **Address:** Map marker icon + "Address" label + street address
 2. **Phone:** Phone icon + "Phone" label + 4142750857
 3. **Email:** Envelope icon + "Email" label + contact.support@spartexai.com
 4. **Business Hours:** Clock icon + "Business Hours" label + operating hours
- **Acceptance Criteria:**
 - "Contact Us" uses H2 semantic markup
 - Introductory paragraph in dashed border box
 - Each contact method includes icon in bordered box, label, and information
 - Phone number formatted as clickable tel: link
 - Email address formatted as clickable mailto: link
 - Icons consistently sized and aligned
 - Contact information easily scannable
 - Address could link to map application (optional)

FR-7.3: Contact Form

- **Requirement:** System shall provide contact form with four input fields and submit button
- **Input:** User enters information and clicks "Send Message"
- **Processing:**
 - Client-side validation of all fields
 - Form data submitted to backend API endpoint
 - Success/error message displayed to user
- **Output:** Form submission confirmation or error message
- **Form Fields:**
 1. **Name:** Text input, required, minimum 2 characters
 2. **Email:** Email input, required, valid email format
 3. **Subject:** Text input, required, minimum 5 characters
 4. **Message:** Textarea, required, minimum 10 characters
- **Acceptance Criteria:**
 - Form contained in bordered box with title "Send Us a Message"
 - All fields have visible labels
 - All fields show border styling consistent with design system

- Real-time validation feedback (inline error messages)
- Required field indicators (asterisk or label)
- Submit button spans full width of form
- Submit button uses filled style with high contrast
- Submit button shows loading state during submission
- Success message displays after successful submission
- Error message displays if submission fails
- Form resets after successful submission
- Form accessible via keyboard navigation
- Proper ARIA labels for screen readers
- Name field: maxlength 100 characters
- Subject field: maxlength 200 characters
- Message field: maxlength 2000 characters

FR-7.4: Form Validation

- **Requirement:** System shall validate all form inputs before submission
- **Input:** User interaction with form fields
- **Processing:** Angular reactive forms validation
- **Output:** Validation feedback messages
- **Validation Rules:**
 - **Name:** Required, 2-100 characters, letters and spaces only
 - **Email:** Required, valid email format (RFC 5322)
 - **Subject:** Required, 5-200 characters
 - **Message:** Required, 10-2000 characters
- **Acceptance Criteria:**
 - Validation occurs on blur and on submit
 - Error messages display below respective fields
 - Error messages are specific and actionable
 - Fields with errors highlighted with red border
 - Submit button disabled until form is valid
 - Error messages clear when user corrects input

- Validation prevents form submission with invalid data

FR-8: FOOTER SECTION

Priority: Medium

Description: The footer provides comprehensive site navigation and company information.

FR-8.1: Four-Column Grid Layout

- **Requirement:** System shall organize footer content into four columns: Company, Services, Industries, and Contact Us
- **Input:** None (static display)
- **Processing:** Layout adapts responsively across viewport sizes
- **Output:** Organized footer navigation
- **Acceptance Criteria:**
 - Four columns on desktop ($\geq 1024\text{px}$)
 - Two columns on tablet (768-1023px)
 - Single column stack on mobile ($< 768\text{px}$)
 - Consistent spacing between columns
 - Column headings are underlined for emphasis
 - Adequate padding within footer section

FR-8.2: Footer Navigation Links

- **Requirement:** System shall provide comprehensive navigation links organized by category
- **Input:** User clicks footer link
- **Processing:** Navigate to corresponding page section or external page
- **Output:** Navigation to target destination
- **Link Categories:**
 - **Company:** About Us, Our Team, Careers, News
 - **Services:** Multi-agent AI Solutions, AI Chatbots, LLM Integration, Custom Development
 - **Industries:** Retail, Healthcare, Finance, Logistics, Professional Services, Legal
 - **Contact Us:** Email address, Phone number, Location
- **Acceptance Criteria:**
 - All links are properly formatted and functional
 - Links to page sections use smooth scroll navigation

- External links open in new tab (if applicable)
- Links have hover state styling
- Links are keyboard accessible
- Column headings (H3 or H4) with underline decoration
- Links styled consistently with subtle hover effect
- Adequate spacing between links for touch targets

FR-8.3: Social Media Integration

- **Requirement:** System shall display social media icons for Twitter, LinkedIn, GitHub, and YouTube
- **Input:** User clicks social media icon
- **Processing:** Open social media profile in new browser tab
- **Output:** Navigation to Spartex AI social media profile
- **Acceptance Criteria:**
 - Four social icons: Twitter, LinkedIn, GitHub, YouTube
 - Icons displayed below Contact Us column
 - Icons styled as circular buttons
 - Icons have hover state (color change or lift effect)
 - Icons link to actual social media profiles (URLs to be provided)
 - Links open in new tab with rel="noopener noreferrer"
 - Icons sized consistently (minimum 40x40px for touch)
 - Icons accessible with proper ARIA labels
 - Icons visually balanced and aligned

FR-8.4: Copyright Bar

- **Requirement:** System shall display copyright notice "© 2023 Spartex AI. All rights reserved."
- **Input:** None (static display)
- **Processing:** Copyright text renders at bottom of footer
- **Output:** Visible copyright notice
- **Acceptance Criteria:**
 - Copyright text centered horizontally
 - Text uses smaller font size than footer links
 - Adequate padding above and below text

- Text color provides sufficient contrast for readability
- Year updates automatically (2025 currently, but note says 2023 - clarify with stakeholders)
- Separated from footer columns by horizontal line or spacing

FR-9: RESPONSIVE BEHAVIOR

Priority: High

Description: The system adapts layout and functionality across all device categories.

FR-9.1: Mobile Optimization (< 768px)

- **Requirement:** System shall optimize all sections for mobile viewport
- **Input:** Application loaded on mobile device
- **Processing:** Responsive CSS applies mobile-specific layouts
- **Output:** Mobile-optimized interface
- **Acceptance Criteria:**
 - Header: Hamburger menu for navigation, logo scales appropriately
 - Hero: Single column layout, CTAs stack vertically
 - Services: Cards stack vertically with full width
 - Industries: Cards stack vertically with full width
 - Case Studies: Cards stack vertically with full width
 - About: Text and image stack vertically, image appears after text
 - Contact: Form appears first, contact info follows
 - Footer: Single column stack
 - Touch targets minimum 44x44px
 - Font sizes scale appropriately for readability
 - No horizontal scrolling required
 - Images scale proportionally

FR-9.2: Tablet Optimization (768px - 1023px)

- **Requirement:** System shall optimize layout for tablet viewports
- **Input:** Application loaded on tablet device
- **Processing:** Responsive CSS applies tablet-specific layouts
- **Output:** Tablet-optimized interface

- **Acceptance Criteria:**
 - Header: Full navigation menu visible
 - Services: 2x2 grid layout
 - Industries: 2x3 grid layout
 - Case Studies: May display 2 per row or stack
 - About: Two columns maintained or stacked based on content
 - Contact: Two columns maintained or stacked
 - Footer: 2x2 column grid
 - Comfortable touch targets
 - Appropriate spacing and padding

FR-9.3: Desktop Optimization ($\geq 1024\text{px}$)

- **Requirement:** System shall provide full-featured desktop experience
- **Input:** Application loaded on desktop/laptop
- **Processing:** Responsive CSS applies desktop layouts
- **Output:** Desktop-optimized interface
- **Acceptance Criteria:**
 - All sections display in intended multi-column layouts
 - Maximum content width maintained for readability (e.g., 1400px)
 - Content centered with appropriate margins
 - Hover effects fully functional
 - Smooth transitions and animations
 - Optimal typography scaling

FR-9.4: Large Display Optimization ($\geq 1920\text{px}$)

- **Requirement:** System shall maintain design integrity on large displays
- **Input:** Application loaded on large display
- **Processing:** Content scales within maximum width constraints
- **Output:** Appropriately scaled interface
- **Acceptance Criteria:**
 - Content container max-width prevents over-stretching
 - Images maintain quality at larger sizes

- White space used effectively
- Typography remains comfortable to read
- No layout breaking or distortion

FR-10: NAVIGATION & ROUTING

Priority: High

Description: The system provides smooth navigation between page sections.

FR-10.1: Section Navigation

- **Requirement:** System shall enable smooth scrolling navigation to any page section
- **Input:** User clicks navigation link or CTA button
- **Processing:** Angular Router or scroll service navigates to target section
- **Output:** Smooth scroll animation to target section
- **Acceptance Criteria:**
 - Smooth scroll animation (500-800ms duration)
 - Appropriate scroll offset for fixed header
 - Active section highlighted in navigation menu
 - Browser back/forward buttons work correctly
 - Deep linking supported via URL fragments
 - Scroll position maintained on page refresh

FR-10.2: Mobile Menu Navigation

- **Requirement:** System shall provide hamburger menu navigation on mobile devices
- **Input:** User taps hamburger icon
- **Processing:** Mobile menu overlay displays/hides
- **Output:** Mobile navigation menu
- **Acceptance Criteria:**
 - Hamburger icon displays on mobile (< 768px)
 - Menu animates open/closed smoothly
 - Menu overlay covers viewport
 - Menu includes all navigation links
 - Menu closes on link selection
 - Menu closes on outside tap/click

- Menu accessible via keyboard
- Body scroll disabled when menu open

FR-10.3: Active Section Tracking

- **Requirement:** System shall highlight active section in navigation menu during scroll
- **Input:** User scrolls page
- **Processing:** Intersection Observer tracks visible sections
- **Output:** Active navigation link highlighted
- **Acceptance Criteria:**
 - Active section detected based on viewport position
 - Corresponding navigation link highlighted
 - Smooth transition between active states
 - Works correctly during manual scroll and programmatic navigation

3.2 Inputs, Processing, Outputs

Input Specifications

User Inputs:

1. Navigation Clicks

- Type: Mouse click or touch tap
- Source: Navigation menu, footer links, CTA buttons
- Format: DOM event
- Validation: None required
- Frequency: Multiple per session

2. Contact Form Data

- **Name Field:**
 - Type: Text string
 - Format: UTF-8 characters
 - Validation: 2-100 characters, letters and spaces only
 - Required: Yes
- **Email Field:**
 - Type: Email string

- Format: Valid email (RFC 5322)
- Validation: Email format validation
- Required: Yes
- **Subject Field:**
 - Type: Text string
 - Format: UTF-8 characters
 - Validation: 5-200 characters
 - Required: Yes
- **Message Field:**
 - Type: Text string
 - Format: UTF-8 characters
 - Validation: 10-2000 characters
 - Required: Yes

3. **Authentication Actions**

- Type: Button clicks
- Source: Login/Sign Up buttons
- Format: DOM event
- Validation: None (navigation action)
- Frequency: Once per session typically

4. **Scroll Events**

- Type: Scroll position data
- Source: User scroll actions
- Format: Viewport position coordinates
- Validation: None required
- Frequency: Continuous during scroll

5. **Window Resize Events**

- Type: Viewport dimension changes
- Source: Browser window resize or device orientation change
- Format: Window dimensions
- Validation: None required
- Frequency: Occasional

Processing Requirements

Client-Side Processing:

1. Form Validation Processing

- Algorithm: Angular Reactive Forms validators
- Processing Time: < 50ms per field
- Dependencies: Angular Forms module
- Error Handling: Display validation messages
- Output: Validation state (valid/invalid)

2. Scroll Navigation Processing

- Algorithm: Smooth scroll to calculated offset position
- Processing Time: 500-800ms animation duration
- Dependencies: Angular Router or custom scroll service
- Error Handling: Fallback to instant scroll if animation fails
- Output: Updated viewport position

3. Section Tracking Processing

- Algorithm: Intersection Observer API
- Processing Time: Real-time during scroll
- Dependencies: Intersection Observer browser API
- Error Handling: Fallback to scroll position calculation
- Output: Active section identifier

4. Responsive Layout Processing

- Algorithm: CSS media query evaluation
- Processing Time: Instantaneous
- Dependencies: CSS engine
- Error Handling: Progressive enhancement approach
- Output: Applied CSS rules

5. Form Submission Processing

- Algorithm: HTTP POST request to backend API
- Processing Time: Dependent on network (typically 500-2000ms)
- Dependencies: Angular HttpClient, backend API

- Error Handling: Retry logic, timeout handling, error display
- Output: Success/error response

Data Transformation:

1. Form Data Serialization

- Input: Form field values
- Process: Create JSON object
- Output: JSON payload for API submission
- Format:

```
{ "name": "string", "email": "string", "subject": "string", "message": "string", "timestamp": "ISO8601" }
```

2. Email Format Validation

- Input: Email string
- Process: Regex pattern matching
- Output: Boolean valid/invalid
- Pattern: RFC 5322 compliant regex

3. URL Fragment Processing

- Input: URL hash fragment
- Process: Extract section identifier
- Output: Target section element reference

Output Specifications

Visual Outputs:

1. Rendered Page Sections

- Format: HTML DOM elements
- Update Frequency: On load and state changes
- Presentation: Styled via CSS according to design system
- Responsiveness: Adapts to viewport dimensions

2. Navigation Feedback

- Format: Visual state changes (highlighting, animations)
- Update Frequency: On user interaction
- Presentation: CSS transitions and animations
- Duration: 200-300ms for most transitions

3. Form Validation Messages

- Format: Text strings in error styling
- Update Frequency: On field blur and form submit
- Presentation: Below respective form fields in red text
- Content: Specific, actionable error descriptions

4. Loading States

- Format: Spinners, skeleton screens, or progress indicators
- Update Frequency: During asynchronous operations
- Presentation: Overlay or inline depending on context
- Duration: Until operation completes

Data Outputs:

1. Form Submission Data

- Format: JSON
- Destination: Backend API endpoint
- Protocol: HTTPS POST request
- Headers: Content-Type: application/json
- Sample:

json

```
{  
  "name": "John Doe",  
  "email": "john@example.com",  
  "subject": "Inquiry about AI Solutions",  
  "message": "I would like to learn more about your services...",  
  "timestamp": "2025-10-06T14:30:00Z",  
  "source": "homepage_contact_form"  
}
```

2. Analytics Events

- Format: Event tracking calls
- Destination: Analytics platform (e.g., Google Analytics)
- Events Tracked:
 - Page load
 - Section views

- Button clicks
- Form submissions
- Link clicks
- Error occurrences

3. **Browser Storage**

- Format: LocalStorage key-value pairs (if needed for user preferences)
- Data: Theme preferences, consent flags
- Persistence: Until cleared by user
- Size Limit: Minimal (< 5KB)

User Feedback Outputs:

1. **Success Messages**

- Type: Toast notification or inline message
- Duration: 3-5 seconds for toasts
- Style: Green background with success icon
- Position: Top-right corner or inline with form
- Content: "Thank you! Your message has been sent successfully. We'll get back to you within 24 hours."

2. **Error Messages**

- Type: Toast notification or inline message
- Duration: Until dismissed or 10 seconds
- Style: Red background with error icon
- Position: Top-right corner or inline with form
- Content: Specific error description and suggested action

3. **Validation Feedback**

- Type: Inline text below form fields
- Style: Red text with error icon
- Content: Field-specific validation errors
- Examples:
 - "Name must be at least 2 characters"
 - "Please enter a valid email address"
 - "Subject is required"

- "Message must be at least 10 characters"

3.3 Error Handling & Exceptions

Error Categories

Category 1: User Input Errors

E-1.1: Form Validation Errors

- **Scenario:** User submits form with invalid or missing data
- **Detection:** Client-side Angular validators
- **Handling:**
 - Prevent form submission
 - Display field-specific error messages
 - Highlight invalid fields with red border
 - Focus first invalid field
 - Maintain user-entered data (do not clear form)
- **User Message:** Field-specific messages (e.g., "Email address is required" or "Message must be at least 10 characters")
- **Recovery:** User corrects invalid fields and resubmits
- **Logging:** Not required for client-side validation

E-1.2: Invalid Email Format

- **Scenario:** User enters malformed email address
- **Detection:** Email validation regex
- **Handling:** Display inline error message, prevent submission
- **User Message:** "Please enter a valid email address (e.g., name@example.com)"
- **Recovery:** User corrects email format
- **Logging:** Not required

E-1.3: Special Characters in Name Field

- **Scenario:** User enters numbers or special characters in name
- **Detection:** Name field validation pattern
- **Handling:** Display inline error message

- **User Message:** "Name should contain only letters and spaces"
- **Recovery:** User removes invalid characters
- **Logging:** Not required

Category 2: Network & Communication Errors

E-2.1: Form Submission API Failure

- **Scenario:** Backend API endpoint unreachable or returns error
- **Detection:** HTTP error response or network timeout
- **Handling:**
 - Display user-friendly error message
 - Maintain form data (do not clear)
 - Provide retry option
 - Log error details to console
- **User Message:** "We're sorry, but we couldn't send your message at this time. Please try again or contact us directly at contact.support@spartexai.com"
- **Recovery:** User clicks retry button or uses alternative contact method
- **Logging:** Console error log with timestamp, endpoint, status code

E-2.2: Network Timeout

- **Scenario:** Form submission exceeds timeout threshold (30 seconds)
- **Detection:** HTTP request timeout
- **Handling:**
 - Cancel request
 - Display timeout error message
 - Maintain form data
 - Provide retry option
- **User Message:** "The request is taking longer than expected. Please check your internet connection and try again."
- **Recovery:** User checks connection and retries
- **Logging:** Console warning with timestamp and endpoint

E-2.3: Slow Network Loading

- **Scenario:** Initial page load on slow connection
- **Detection:** Page load time monitoring
- **Handling:**
 - Display loading skeleton or progress indicator
 - Lazy load images and non-critical resources
 - Prioritize above-the-fold content
- **User Message:** Loading indicators (no error message unless timeout)
- **Recovery:** Automatic as resources load
- **Logging:** Performance metrics to analytics

Category 3: Browser & Compatibility Errors

E-3.1: Unsupported Browser

- **Scenario:** User accesses site from outdated browser (e.g., IE11)
- **Detection:** User agent detection or feature detection
- **Handling:**
 - Display browser upgrade notice
 - Provide degraded but functional experience if possible
 - Show links to modern browser downloads
- **User Message:** "For the best experience, please use a modern browser like Chrome, Firefox, Safari, or Edge."
- **Recovery:** User upgrades browser
- **Logging:** Analytics event for unsupported browser detection

E-3.2: JavaScript Disabled

- **Scenario:** User has JavaScript disabled in browser
- **Detection:** Noscript tag
- **Handling:**
 - Display message in noscript element
 - Provide static contact information
- **User Message:** "This website requires JavaScript to function properly. Please enable JavaScript in your browser settings."
- **Recovery:** User enables JavaScript

- **Logging:** Not possible (JavaScript disabled)

E-3.3: Browser Storage Quota Exceeded

- **Scenario:** LocalStorage quota exceeded (if used)
- **Detection:** QuotaExceededError exception
- **Handling:**
 - Gracefully degrade to session-only storage
 - Log warning to console
 - Continue operation without storage
- **User Message:** None (silent handling)
- **Recovery:** Automatic fallback
- **Logging:** Console warning

Category 4: Resource Loading Errors

E-4.1: Image Load Failure

- **Scenario:** Image resource fails to load (404, network error)
- **Detection:** Image onerror event
- **Handling:**
 - Display placeholder image or icon
 - Retry load once after delay
 - Fallback to alt text
- **User Message:** None (visual fallback only)
- **Recovery:** Automatic retry, then graceful degradation
- **Logging:** Console warning with image URL

E-4.2: Font Loading Failure

- **Scenario:** Web font fails to load
- **Detection:** Font loading API or timeout
- **Handling:**
 - Fallback to system fonts
 - Maintain layout integrity
- **User Message:** None (automatic fallback)

- **Recovery:** Automatic system font substitution
- **Logging:** Console warning

E-4.3: Icon Library Loading Failure

- **Scenario:** Icon font or SVG sprite fails to load
- **Detection:** Resource loading error
- **Handling:**
 - Display text labels as fallback
 - Use Unicode symbols where appropriate
- **User Message:** None (visual fallback)
- **Recovery:** Automatic fallback to text/unicode
- **Logging:** Console warning

Category 5: Navigation & State Errors

E-5.1: Invalid URL Fragment

- **Scenario:** User navigates to non-existent section via URL hash
- **Detection:** Route guard or scroll service validation
- **Handling:**
 - Scroll to top of page
 - Log warning
- **User Message:** None (silent handling with default scroll position)
- **Recovery:** Automatic scroll to safe position
- **Logging:** Console warning with invalid fragment

E-5.2: Scroll Position Restoration Failure

- **Scenario:** Browser fails to restore scroll position
- **Detection:** Scroll restoration error
- **Handling:**
 - Default to top of page
 - Allow user to navigate manually
- **User Message:** None
- **Recovery:** Manual navigation

- **Logging:** Console warning

Category 6: External Service Errors

E-6.1: Social Media Link Failure

- **Scenario:** Social media profile URL is incorrect or profile deleted
- **Detection:** User reports or monitoring
- **Handling:**
 - Link opens but shows 404 on destination
 - Update link configuration
- **User Message:** None from application (external site handles)
- **Recovery:** Update configuration with correct URL
- **Logging:** None (external issue)

E-6.2: Analytics Service Unavailable

- **Scenario:** Analytics script fails to load or initialize
- **Detection:** Script loading error or API unavailability
- **Handling:**
 - Silently continue without analytics
 - Do not block user experience
- **User Message:** None
- **Recovery:** Automatic (application continues normally)
- **Logging:** Console warning

Error Logging Strategy

Console Logging:

- All errors logged to browser console with severity level
- Format: [SPARTEX-AI] [ERROR/WARNING/INFO] [Component] Message
- Include timestamp, error type, and context

Analytics Logging:

- Critical errors sent to analytics platform
- Include error type, user agent, viewport size

- Track error frequency for monitoring

User-Facing Error Display:

- Use Toast notifications for temporary errors
- Use inline messages for form validation
- Use modal dialogs for critical errors requiring user action
- Always provide actionable next steps

Error Recovery Procedures

Automatic Recovery:

- Network request retry with exponential backoff (1s, 2s, 4s)
- Image load retry after 3 seconds
- Automatic fallback to alternative resources

User-Initiated Recovery:

- "Try Again" buttons for failed operations
- "Refresh Page" option for critical failures
- Alternative contact methods for form submission failures

Graceful Degradation:

- Progressive enhancement approach ensures basic functionality
 - Core content accessible even with JavaScript errors
 - Static contact information always visible
-

4. External Interface Requirements

4.1 User Interfaces (UI/UX)

UI Design Principles

Design System:

- **Visual Style:** Modern, clean, professional with technology-forward aesthetic
- **Key Element:** Bordered boxes and dashed containers as primary design motif
- **Color Scheme:**

- Primary: Technology blue (to be defined in brand guidelines)
- Secondary: Complementary accent color
- Neutral: Grays for text and backgrounds
- Semantic: Green for success, red for errors, yellow for warnings
- **Typography:**
 - Headings: Sans-serif font, bold weight
 - Body: Sans-serif font, regular weight
 - Size scale: Responsive type scale (mobile: 14-32px, desktop: 16-48px)
 - Line height: 1.5 for body text, 1.2 for headings
- **Spacing System:** 8px base unit (8, 16, 24, 32, 48, 64px)
- **Border Radius:** Consistent 8px for cards and buttons
- **Shadows:** Subtle elevation system for depth (0-4 levels)

Layout Specifications:

Header (Fixed)

- Height: 80px on desktop, 64px on mobile
- Background: White with subtle shadow or border
- Z-index: 1000
- Content max-width: 1400px centered

Section Spacing

- Vertical padding: 80px on desktop, 48px on mobile
- Maximum content width: 1400px
- Horizontal padding: 24px (mobile), 48px (tablet), 64px (desktop)

Grid System

- 12-column grid for layout flexibility
- Gutter width: 24px on desktop, 16px on mobile
- Service cards: 4-column grid (3 columns on tablet, 1 on mobile)
- Industry cards: 6-column grid (3 per row on desktop, 2 on tablet, 1 on mobile)

Component Specifications:

Buttons

- Primary (filled): Solid background, white text, 48px height
- Secondary (outlined): Border 2px, transparent background, 48px height
- Minimum width: 120px
- Padding: 16px horizontal
- Border radius: 8px
- Hover: Slight elevation or color darkening
- Active: Pressed state with reduced elevation
- Disabled: 50% opacity, no pointer events

Form Inputs

- Height: 48px for single-line inputs
- Border: 1px solid neutral gray
- Border radius: 8px
- Padding: 12px 16px
- Focus state: Border color changes to primary, 2px outline
- Error state: Red border, error icon, error message below
- Placeholder: Light gray text
- Font size: 16px (prevents mobile zoom on focus)

Cards

- Background: White
- Border: 1px solid light gray or dashed for specific contexts
- Border radius: 12px
- Padding: 24px
- Hover: Subtle lift effect (translateY(-4px)) with shadow increase
- Transition: 200ms ease-in-out

Icons

- Size: 24px for inline icons, 48px for feature icons, 64px for section icons
- Style: Line icons or solid as per brand guidelines
- Color: Primary color or contextual color

- Contained icons: In bordered 64px squares with 16px padding

Accessibility Requirements

WCAG 2.1 Level AA Compliance:

Visual Accessibility

- Color contrast ratio minimum 4.5:1 for normal text
- Color contrast ratio minimum 3:1 for large text (18pt+)
- Information not conveyed by color alone
- Focus indicators visible and high contrast (2px outline)
- Text resizable up to 200% without loss of functionality

Keyboard Accessibility

- All interactive elements accessible via keyboard
- Logical tab order throughout page
- Skip navigation link for screen readers
- Keyboard shortcuts documented (if implemented)
- No keyboard traps
- Focus never hidden or lost

Screen Reader Support

- Semantic HTML5 elements (header, nav, main, section, footer)
- Proper heading hierarchy (H1 → H2 → H3)
- ARIA labels for icons and interactive elements
- ARIA live regions for dynamic content updates
- Alt text for all images (decorative images: alt="")
- Form labels properly associated with inputs
- Error messages associated with form fields via aria-describedby

Motor Accessibility

- Touch targets minimum 44x44px
- Adequate spacing between interactive elements (minimum 8px)
- No hover-only functionality

- Generous click/tap areas

Cognitive Accessibility

- Consistent navigation and layout
- Clear, simple language
- Instructions provided for complex interactions
- Error messages clear and actionable
- Sufficient time for form completion (no timeouts)
- Clear visual hierarchy and information grouping

Responsive Breakpoints

Mobile Small (320px - 479px)

- Single column layout
- Stacked navigation
- Font sizes: Base 14px
- Padding: 16px
- Hamburger menu

Mobile Large (480px - 767px)

- Single column layout
- Enhanced spacing
- Font sizes: Base 15px
- Padding: 20px

Tablet (768px - 1023px)

- Two-column layouts where appropriate
- Visible navigation menu
- Font sizes: Base 16px
- Padding: 32px
- Service cards: 2x2 grid

Desktop Small (1024px - 1439px)

- Multi-column layouts

- Full navigation menu
- Font sizes: Base 16px
- Padding: 48px
- Service cards: 2x2 or 1x4 grid
- Case studies: May be 2 + 1 layout

Desktop Large (1440px+)

- Maximum content width: 1400px
- Optimal spacing
- Font sizes: Base 16px-18px
- Padding: 64px
- All grids at full capacity

Animation & Interaction Guidelines

Animation Principles

- Duration: 200-300ms for micro-interactions, 500-800ms for page transitions
- Easing: ease-in-out for most animations
- Respect prefers-reduced-motion media query
- Animations enhance, never block interaction

Hover Effects

- Buttons: Background color change, slight elevation
- Cards: Lift effect (4px), shadow increase
- Links: Color change, underline appearance
- Icons: Color change, subtle scale increase (1.05x)

Loading States

- Buttons: Spinner animation, disabled state
- Forms: Overlay with spinner
- Page sections: Skeleton screens or fade-in
- Images: Progressive loading with blur-up effect

Scroll Animations

- Smooth scroll: 500-800ms ease-in-out
- Parallax effects: Subtle, performant
- Fade-in on scroll: For section reveals (optional)
- Navigation highlighting: Instant feedback

Micro-interactions

- Form field focus: Border color change, outline appearance
- Input validation: Checkmark or error icon animation
- Success messages: Slide-in and fade-out
- Menu transitions: Smooth open/close animations

4.2 Hardware Interfaces

Client Device Requirements:

Minimum Specifications

- Processor: Any modern CPU (2015 or newer)
- RAM: 2GB minimum
- Display: 320px minimum width
- Input: Touch screen, mouse, or keyboard
- Network: 2 Mbps minimum bandwidth

Recommended Specifications

- Processor: Dual-core 2.0 GHz or better
- RAM: 4GB or more
- Display: 1920x1080 or higher resolution
- Input: Mouse and keyboard for optimal experience
- Network: 10 Mbps or faster

Device Categories

- Smartphones: iOS 14+ (iPhone 8+), Android 9+ (mid-range and above)
- Tablets: iPad 5th generation+, Android tablets with 9+ OS
- Laptops: Any model from 2015 or newer
- Desktops: Any modern configuration

- Large displays: 4K monitors supported

Input Devices

- Touch screens: Full touch gesture support (tap, swipe, pinch-zoom)
- Mouse: Hover states, click interactions, scroll wheel
- Keyboard: Full keyboard navigation support
- Trackpad: All standard trackpad gestures

Display Requirements

- Color depth: 24-bit minimum
- Resolution: 320px width minimum
- Aspect ratio: Support for standard ratios (16:9, 16:10, 4:3, 21:9)
- Orientation: Portrait and landscape support

No Special Hardware Required

- No camera access needed
- No microphone access needed
- No GPS location required (geolocation disabled)
- No accelerometer required
- No NFC or Bluetooth required

4.3 Software Interfaces

Frontend Framework & Libraries:

Core Framework

- **Angular 19.x**
 - Purpose: Primary application framework
 - Interface: TypeScript/JavaScript API
 - Version: Latest stable (19.x)
 - Components: Standalone components architecture
 - Modules: Router, Forms, HttpClient, Common

Programming Language

- **TypeScript 5.x**

- Purpose: Primary development language
- Compilation target: ES2022
- Strict mode enabled

UI Component Libraries (Optional)

- **Angular Material** or **PrimeNG** (if approved)
 - Purpose: Pre-built UI components
 - Usage: Form controls, dialogs, tooltips
 - Customization: Themed to match brand guidelines

Icon Libraries

- **Font Awesome** or **Material Icons** or **Lucide Icons**
 - Purpose: Icons for UI elements
 - Format: SVG or icon font
 - Usage: Navigation, features, social media

Animation Libraries (Optional)

- **Angular Animations** (built-in)
 - Purpose: Page transitions, micro-interactions
 - Usage: Smooth UI animations

Form Validation

- **Angular Reactive Forms**
 - Purpose: Form state management and validation
 - Built-in validators and custom validators

HTTP Communication

- **Angular HttpClient**
 - Purpose: API communication for form submission
 - Interface: RESTful API calls
 - Format: JSON payloads

Routing

- **Angular Router**

- **Purpose:** Section navigation with URL fragments
- **Features:** Deep linking, scroll position management

State Management (If Needed)

- **Angular Services** with RxJS
 - **Purpose:** Application state management
 - **Pattern:** Observable-based reactive state

Build & Development Tools:

Build System

- **Angular CLI** with **Vite**
 - **Purpose:** Build, development server, code generation
 - **Commands:** ng serve, ng build, ng test
 - **Output:** Optimized production bundles

Package Manager

- **npm** or **yarn**
 - **Purpose:** Dependency management
 - **Package registry:** npmjs.com

CSS Preprocessing (Optional)

- **SCSS/SASS** (if needed)
 - **Purpose:** Enhanced CSS with variables, mixins
 - **Compilation:** Built into Angular CLI

Linting & Formatting

- **ESLint** for TypeScript/JavaScript
- **Prettier** for code formatting
- **Stylelint** for CSS (optional)

Testing Frameworks

- **Jasmine** or **Jest** for unit testing
- **Karma** or **Jest** as test runner
- **Protractor** or **Cypress** for E2E testing (optional in initial phase)

Backend API Interface:

Contact Form API

- **Protocol:** HTTPS
- **Method:** POST
- **Endpoint:** `https://api.spartexai.com/v1/contact` (URL to be confirmed)
- **Authentication:** API key in header (if required)
- **Request Format:**

json

```
{  
  "name": "string",  
  "email": "string",  
  "subject": "string",  
  "message": "string",  
  "timestamp": "ISO8601 datetime",  
  "source": "homepage_contact_form"  
}
```

- **Response Format (Success):**

json

```
{  
  "status": "success",  
  "message": "Form submitted successfully",  
  "ticket_id": "string"  
}
```

- **Response Format (Error):**

json

```
{  
  "status": "error",  
  "message": "Error description",  
  "errors": {  
    "field_name": ["Error message"]  
  }  
}
```

- **Status Codes:**

- 200: Success
- 400: Bad request (validation errors)
- 429: Rate limit exceeded
- 500: Server error
- 503: Service unavailable

Authentication API (Future)

- **Protocol:** HTTPS
- **Authentication:** OAuth 2.0 / OpenID Connect
- **Endpoints:**
 - Login: `/auth/login`
 - Register: `/auth/register`
 - Token refresh: `/auth/refresh`
- **Interface:** JWT tokens in Authorization header

Third-Party Services:

Google Fonts (or similar)

- **Purpose:** Web typography
- **Interface:** CSS link in HTML head
- **Fallback:** System fonts if unavailable

Google Analytics (or similar)

- **Purpose:** Usage analytics and tracking
- **Interface:** JavaScript tracking snippet
- **Data:** Page views, events, user interactions

CDN Services

- **Purpose:** Static asset delivery
- **Providers:** AWS CloudFront, Cloudflare, or similar
- **Interface:** HTTP/HTTPS requests
- **Content:** Images, fonts, compiled JavaScript/CSS

Social Media Platforms

- **Purpose:** Social media links
- **Platforms:** Twitter, LinkedIn, GitHub, YouTube
- **Interface:** Standard HTTPS links to profile pages

Browser APIs:

Web APIs Used

- **Intersection Observer API:** Section visibility tracking
- **Fetch API / XMLHttpRequest:** HTTP communication
- **History API:** URL fragment management
- **LocalStorage API:** User preferences (optional)
- **Console API:** Error logging
- **Window/Document APIs:** DOM manipulation, event handling
- **FormData API:** Form handling

Web APIs Not Required

- Geolocation API (explicitly disabled)
- Camera/Media APIs
- WebRTC
- Web Audio API
- File System Access API
- Payment Request API

4.4 Communication Interfaces

Network Protocols:

HTTPS (Primary)

- **Usage:** All client-server communication
- **Version:** TLS 1.2 or higher
- **Port:** 443
- **Purpose:** Secure data transmission
- **Certificates:** Valid SSL/TLS certificate required

HTTP/2

- **Usage:** Optimized resource loading
- **Benefits:** Multiplexing, header compression
- **Fallback:** HTTP/1.1 if not supported

WebSocket (Future Consideration)

- **Usage:** Real-time features (if implemented in future)
- **Protocol:** WSS (WebSocket Secure)
- **Purpose:** Live chat, notifications

DNS

- **Usage:** Domain name resolution
- **Domain:** spartexai.com (or actual domain)
- **Records:** A/AAAA records for website, MX records for email

Data Formats:

JSON

- **Usage:** API requests and responses
- **Content-Type:** application/json
- **Encoding:** UTF-8
- **Structure:** Well-formed, validated JSON

HTML5

- **Usage:** Page markup and structure
- **Version:** HTML5
- **Encoding:** UTF-8
- **Validation:** W3C compliant

CSS3

- **Usage:** Styling and layout
- **Version:** CSS3
- **Encoding:** UTF-8
- **Validation:** W3C compliant

JavaScript/TypeScript

- **Usage:** Application logic
- **Version:** ECMAScript 2022+
- **Encoding:** UTF-8
- **Transpilation:** TypeScript to JavaScript

SVG

- **Usage:** Icons and logos
- **Format:** XML-based vector graphics
- **Optimization:** Minified and compressed

Image Formats

- **WebP:** Primary format for photographs (with fallback)
- **JPEG:** Fallback for photographs
- **PNG:** Logos, screenshots (when transparency needed)
- **SVG:** Icons, logos, illustrations

API Communication Specifications:

Request Headers

```
POST /v1/contact HTTP/1.1
Host: api.spartexai.com
Content-Type: application/json
Accept: application/json
Authorization: Bearer {api_key} (if required)
User-Agent: SpartexAI-Homepage/1.0
Origin: https://www.spartexai.com
```

Response Headers

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: <https://www.spartexai.com>

Cache-Control: no-cache

X-Request-ID: unique-request-identifier

CORS Configuration

- **Allowed Origins:** <https://www.spartexai.com>
- **Allowed Methods:** GET, POST, OPTIONS
- **Allowed Headers:** Content-Type, Authorization
- **Credentials:** Include if authentication required
- **Max Age:** 3600 seconds

Error Handling

- **Network Errors:** Retry with exponential backoff
- **Timeout:** 30 seconds maximum
- **Rate Limiting:** Respect 429 status codes
- **Server Errors:** Display user-friendly messages

Data Transmission Security:

Encryption

- All data transmitted over HTTPS
- TLS 1.2 or 1.3 encryption
- Strong cipher suites only

Data Validation

- Client-side validation before transmission
- Server-side validation for security
- Input sanitization to prevent XSS

Privacy

- No sensitive data stored in browser storage
- Form data transmitted securely

- Compliance with GDPR, CCPA guidelines
- Cookie consent mechanism

Performance Optimization:

Compression

- Gzip or Brotli compression for text assets
- Image optimization and compression
- Minification of JavaScript and CSS

Caching

- Browser caching for static assets
- Cache-Control headers properly configured
- Service Worker for offline capabilities (future)

CDN Delivery

- Static assets served via CDN
- Geographic distribution for low latency
- Edge caching for improved performance

Request Optimization

- HTTP/2 multiplexing
- Resource bundling and code splitting
- Lazy loading for images and non-critical resources
- Preloading critical resources

5. Non-Functional Requirements

5.1 Performance

Page Load Performance:

Initial Page Load

- **First Contentful Paint (FCP):** ≤ 1.5 seconds on 4G connection
- **Largest Contentful Paint (LCP):** ≤ 2.5 seconds on 4G connection

- **Time to Interactive (TTI):** ≤ 3.5 seconds on 4G connection
- **First Input Delay (FID):** ≤ 100 milliseconds
- **Cumulative Layout Shift (CLS):** ≤ 0.1
- **Speed Index:** ≤ 3.0 seconds

Resource Metrics

- **Initial Bundle Size:** ≤ 500 KB (gzipped)
- **Total Page Weight:** ≤ 2 MB (including images)
- **JavaScript Bundle:** ≤ 250 KB (gzipped)
- **CSS Bundle:** ≤ 50 KB (gzipped)
- **Image Assets:** Optimized, WebP format preferred

Runtime Performance:

Interaction Response

- **Button Clicks:** Response within 100ms
- **Form Input:** Keystroke response within 50ms
- **Navigation:** Section scroll completes within 800ms
- **Form Submission:** Feedback within 200ms (loading state)
- **Hover Effects:** Transition within 200ms

Rendering Performance

- **Frame Rate:** Maintain 60 FPS during animations and scrolling
- **Scroll Performance:** No jank during scroll (16.7ms per frame)
- **Animation Performance:** GPU-accelerated where possible
- **Re-paint Time:** ≤ 16 ms for layout changes

API Performance

- **Contact Form Submission:** Response within 2 seconds (95th percentile)
- **API Timeout:** 30 seconds maximum
- **Retry Logic:** Maximum 3 retries with exponential backoff

Performance Benchmarks by Device:

Desktop (High-End)

- Page Load: < 1 second
- Interaction: < 50ms
- Smooth 60 FPS animations

Desktop (Low-End)

- Page Load: < 2 seconds
- Interaction: < 100ms
- Minimum 30 FPS animations

Mobile (High-End - iPhone 12+, Samsung S20+)

- Page Load: < 2 seconds on 4G
- Interaction: < 100ms
- 60 FPS animations

Mobile (Mid-Range)

- Page Load: < 3 seconds on 4G
- Interaction: < 150ms
- 30 FPS animations acceptable

Mobile (Low-End)

- Page Load: < 5 seconds on 3G
- Interaction: < 200ms
- Reduced animations, core functionality maintained

Optimization Requirements:

Code Splitting

- Lazy load non-critical components
- Route-based code splitting
- Vendor code separation

Image Optimization

- Responsive images with srcset
- WebP format with JPEG/PNG fallback
- Lazy loading for below-fold images

- Blur-up loading technique

Asset Optimization

- Tree-shaking for unused code
- Minification of JavaScript and CSS
- Compression (Gzip/Brotli)
- CDN delivery for static assets

Caching Strategy

- Browser caching for static assets (1 year)
- Service worker caching (future phase)
- API response caching where appropriate

Performance Monitoring:

- Google Lighthouse score: ≥ 90 for Performance
- Real User Monitoring (RUM) implementation
- Core Web Vitals tracking
- Performance budget alerts

5.2 Reliability

Availability:

Uptime Targets

- **Production Uptime:** 99.9% monthly uptime (≈ 43 minutes downtime/month)
- **Maintenance Windows:** Scheduled during low-traffic periods
- **Planned Downtime:** Advance notice required
- **Incident Response Time:** < 15 minutes for critical issues

Error Rates

- **Client-Side Errors:** $< 0.1\%$ of page loads
- **Failed API Requests:** $< 1\%$ of total requests
- **Failed Form Submissions:** $< 0.5\%$ of attempts

Fault Tolerance:

Graceful Degradation

- Core content accessible even if JavaScript fails
- Progressive enhancement approach
- Fallback for unsupported features
- Static contact information always visible

Error Recovery

- Automatic retry for failed network requests (3 attempts)
- Form data preservation during errors
- Clear error messages with recovery instructions
- Alternative contact methods provided

Redundancy

- CDN with multiple edge locations
- Multiple DNS servers
- Load balancing across multiple servers (if applicable)

Data Integrity:

Form Submission

- Client-side validation prevents bad data
- Data integrity checks before submission
- Confirmation of successful submission
- No data loss during transmission errors

State Management

- Consistent application state
- Proper state synchronization
- No stale data displays

Browser Compatibility:

- Tested across all supported browsers
- Feature detection for unsupported APIs
- Polyfills for missing features (where feasible)

Monitoring & Alerting:

- Real-time error tracking (e.g., Sentry)
- Performance monitoring dashboards
- Automated alerts for critical failures
- User feedback collection mechanism

Backup & Recovery:

- Static site architecture enables quick redeployment
- Version control for all code (Git)
- Rollback capability for failed deployments
- Configuration backups

Testing Requirements:

- Comprehensive unit test coverage ($\geq 80\%$)
- Integration testing for critical paths
- End-to-end testing for user workflows
- Cross-browser testing
- Performance regression testing

5.3 Security

Data Security:

Data Transmission

- **Encryption:** All data transmitted over HTTPS with TLS 1.2+
- **Certificate:** Valid SSL/TLS certificate from trusted CA
- **HSTS:** HTTP Strict Transport Security enabled
- **Cipher Suites:** Strong ciphers only, weak ciphers disabled

Form Data Protection

- **Validation:** Comprehensive input validation
- **Sanitization:** All user inputs sanitized before processing
- **XSS Prevention:** Content Security Policy (CSP) headers
- **CSRF Protection:** CSRF tokens for form submissions (if using cookies)

- **Rate Limiting:** API rate limiting to prevent abuse

Sensitive Information

- **No Storage:** No passwords or payment information collected
- **PII Handling:** Personal information transmitted securely
- **Data Minimization:** Only collect necessary information
- **Secure Disposal:** Clear form data after submission

Authentication & Authorization (Future):

Authentication

- OAuth 2.0 / OpenID Connect for user authentication
- Secure session management
- Password requirements: Minimum 8 characters, complexity rules
- Multi-factor authentication support (future)

Authorization

- Role-based access control (when authenticated areas added)
- Principle of least privilege
- Secure token storage

Session Management

- Secure session tokens
- Session timeout after inactivity
- Proper session invalidation on logout

Application Security:

Code Security

- **Dependency Scanning:** Regular security audits of npm packages
- **Vulnerability Patching:** Timely updates for security vulnerabilities
- **Code Reviews:** Security-focused code reviews
- **Static Analysis:** Automated security scanning tools

XSS Prevention

- Angular's built-in XSS protection
- Content Security Policy (CSP) headers
- Input sanitization
- Output encoding

CSRF Prevention

- CSRF tokens for state-changing operations
- SameSite cookie attribute
- Origin validation

Clickjacking Prevention

- X-Frame-Options header: DENY or SAMEORIGIN
- Frame-ancestors CSP directive

Content Security Policy

```
Content-Security-Policy:  
default-src 'self';  
script-src 'self' 'unsafe-inline' 'unsafe-eval' https://cdnjs.cloudflare.com;  
style-src 'self' 'unsafe-inline' https://fonts.googleapis.com;  
font-src 'self' https://fonts.gstatic.com;  
img-src 'self' data: https;;  
connect-src 'self' https://api.spartexai.com;
```

Privacy & Compliance:

GDPR Compliance

- Cookie consent banner
- Privacy policy link in footer
- User data rights information
- Data processing transparency

CCPA Compliance

- "Do Not Sell My Personal Information" link (if applicable)
- Privacy policy disclosure

Cookie Management

- Cookie consent mechanism
- Essential cookies only without consent
- Analytics cookies require opt-in
- Cookie policy documentation

Data Privacy

- Minimal data collection
- Clear privacy policy
- No third-party data sharing without consent
- User control over data

Security Headers:

Recommended Headers

```
Strict-Transport-Security: max-age=31536000; includeSubDomains
X-Content-Type-Options: nosniff
X-Frame-Options: DENY
X-XSS-Protection: 1; mode=block
Referrer-Policy: strict-origin-when-cross-origin
Permissions-Policy: geolocation=(), microphone=(), camera=()
```

Security Monitoring:

- Security event logging
- Intrusion detection monitoring
- Vulnerability scanning
- Penetration testing (periodic)

Incident Response:

- Security incident response plan
- Escalation procedures
- User notification process
- Breach disclosure procedures

5.4 Scalability

Traffic Scalability:

Expected Load

- **Initial Launch:** 1,000-5,000 daily visitors
- **6 Months:** 10,000-20,000 daily visitors
- **1 Year:** 50,000+ daily visitors
- **Peak Traffic:** 3x average during campaigns

Concurrent Users

- **Normal Load:** 50-200 concurrent users
- **Peak Load:** 500-1,000 concurrent users
- **Stress Test Target:** 2,000 concurrent users

Form Submissions

- **Daily Submissions:** 50-200 forms
- **Peak Hour:** 20-50 submissions
- **API Rate Limit:** 100 requests/minute per IP

Architectural Scalability:

Frontend Scalability

- Static site architecture enables easy horizontal scaling
- CDN distribution for global reach
- Edge caching reduces origin load
- Serverless deployment options (AWS S3, Netlify, Vercel)

Backend API Scalability (Future)

- RESTful API design for stateless operations
- Horizontal scaling capability
- Load balancing across multiple instances
- Database connection pooling

Content Delivery

- Multi-region CDN deployment
- Asset caching at edge locations

- Image optimization and responsive serving
- Lazy loading for non-critical resources

Database Scalability (Future)

- Indexed queries for performance
- Connection pooling
- Read replicas for high read loads
- Caching layer (Redis) for frequently accessed data

Growth Accommodation:

Content Expansion

- Architecture supports adding new sections
- Component-based design enables modular growth
- Scalable navigation structure
- CMS integration ready (future)

Feature Addition

- Blog/news section capability
- Client portal integration points
- E-commerce functionality potential
- Multi-language support structure

User Base Growth

- Authentication system integration ready
- User profile management capability
- Dashboard framework prepared
- Analytics infrastructure scalable

Performance Under Load:

Load Testing Requirements

- Simulate 2x expected peak load
- Test sustained load over 1 hour
- Test spike scenarios (sudden traffic increase)

- Test concurrent form submissions

Performance Degradation

- Graceful degradation under extreme load
- Priority given to core functionality
- Non-essential features may be throttled
- Clear user communication during high load

Resource Optimization:

- Code splitting reduces initial load
- Lazy loading defers non-critical resources
- Asset optimization reduces bandwidth
- Efficient algorithms for client-side processing

Monitoring & Auto-Scaling:

- Real-time performance monitoring
- Auto-scaling triggers (if applicable to hosting)
- Alert thresholds for capacity planning
- Usage analytics for capacity forecasting

5.5 Usability

Learnability:

First-Time User Experience

- **Time to Understand Layout:** < 30 seconds
- **Time to Navigate to Section:** < 10 seconds
- **Time to Complete Contact Form:** < 3 minutes
- **Instructions Required:** Minimal to none

User Interface Clarity

- Clear, self-explanatory navigation labels
- Intuitive button placement and styling
- Obvious form field purposes
- Visual hierarchy guides user attention

Onboarding

- No tutorials required for basic navigation
- Contextual help for complex interactions (if any)
- Clear calls-to-action throughout

Efficiency:

Task Completion Time

- **Find Service Information:** < 1 minute
- **View Case Study:** < 2 minutes
- **Submit Contact Form:** < 3 minutes
- **Navigate Between Sections:** < 5 seconds

Interaction Efficiency

- Minimal clicks to reach any section
- Keyboard shortcuts for power users (optional)
- Quick access to contact information
- Persistent navigation always available

Form Efficiency

- Auto-fill support for form fields
- Tab order logical and efficient
- Clear required field indicators
- Inline validation prevents errors

User Satisfaction:

Aesthetic Design

- Modern, professional visual design
- Consistent with brand identity
- Pleasant color scheme and typography
- Balanced white space and content density

Emotional Response

- Trustworthy and credible appearance
- Professional yet approachable tone
- Engaging without being overwhelming
- Confidence-inspiring design elements

User Confidence

- Clear feedback for all actions
- Progress indicators where needed
- Confirmation messages for submissions
- Transparent error handling

Error Prevention & Recovery:

Error Prevention

- Input validation prevents common errors
- Clear instructions for complex fields
- Confirmation for destructive actions
- Auto-save for long forms (if applicable)

Error Recovery

- Clear, specific error messages
- Actionable guidance for resolution
- Form data preserved during errors
- Easy correction of mistakes

Help & Support:

Self-Service Help

- Clear contact information prominently displayed
- FAQ section (if implemented in future)
- Tooltips for complex elements (if needed)
- Accessible support documentation

Contact Methods

- Multiple contact options provided

- Contact form for detailed inquiries
- Phone and email for direct contact
- Business hours clearly stated

Accessibility as Usability:

- WCAG 2.1 AA compliance ensures broad usability
- Keyboard navigation for power users
- Screen reader support for visually impaired
- High contrast for improved readability

Mobile Usability:

- Touch-optimized interface
- Finger-friendly tap targets
- Swipe gestures where appropriate
- Readable text without zooming
- Forms optimized for mobile input

Cross-Browser Consistency:

- Consistent experience across browsers
- Feature parity where possible
- Graceful degradation for unsupported features
- Clear messaging for browser limitations

Performance as Usability:

- Fast page loads reduce frustration
- Responsive interactions feel native
- No perceptible lag in interactions
- Loading indicators for clarity

Usability Testing Requirements:

- User testing with target audience
- A/B testing for critical elements
- Heatmap analysis for interaction patterns

- User feedback collection mechanism
 - Usability metrics tracking (task success rate, time on task, error rate)
-

6. Other Requirements

6.1 Compliance

Web Standards Compliance:

HTML5 Compliance

- Valid HTML5 markup
- Semantic HTML elements used appropriately
- W3C validation passing
- Proper document structure

CSS3 Compliance

- Valid CSS3 stylesheets
- W3C CSS validation passing
- Progressive enhancement approach
- Vendor prefix handling

JavaScript/ECMAScript Compliance

- ES2022+ standards followed
- Strict mode enabled
- TypeScript strict mode
- No deprecated APIs used

Accessibility Compliance:

WCAG 2.1 Level AA

- Perceivable: Content presented in multiple ways
- Operable: All functionality keyboard accessible
- Understandable: Clear, consistent, predictable
- Robust: Compatible with assistive technologies

ADA Compliance

- Americans with Disabilities Act requirements
- Equal access to information and functionality
- Accommodations for various disabilities
- Regular accessibility audits

Section 508 Compliance

- U.S. federal accessibility standards
- Required for potential government clients
- Electronic and information technology accessibility

Privacy & Data Protection:

GDPR (General Data Protection Regulation)

- **Lawful Basis:** Consent for data collection
- **Transparency:** Clear privacy policy
- **User Rights:** Access, rectification, erasure, portability
- **Data Minimization:** Collect only necessary data
- **Security:** Appropriate technical measures
- **Cookie Consent:** Explicit consent for non-essential cookies
- **Privacy by Design:** Privacy considered in development

CCPA (California Consumer Privacy Act)

- **Disclosure:** What data is collected and why
- **Opt-Out:** Right to opt-out of data selling
- **Deletion:** Right to request data deletion
- **Non-Discrimination:** No discrimination for exercising rights

ePrivacy Directive (Cookie Law)

- Cookie consent banner
- Essential cookies explanation
- Analytics/marketing cookies require consent
- Cookie policy available

Other Data Protection Laws

- PIPEDA (Canada)
- LGPD (Brazil)
- Data Protection Act (UK)
- Compliance varies by target markets

Security Standards:

OWASP Top 10 Compliance

- Protection against common vulnerabilities:
 1. Injection attacks
 2. Broken authentication
 3. Sensitive data exposure
 4. XML external entities (XXE)
 5. Broken access control
 6. Security misconfiguration
 7. Cross-site scripting (XSS)
 8. Insecure deserialization
 9. Using components with known vulnerabilities
 10. Insufficient logging & monitoring

PCI DSS (If Applicable in Future)

- Payment Card Industry Data Security Standard
- Required if handling payment card information
- Not applicable to initial homepage implementation

Industry-Specific Compliance:

HIPAA (If Targeting Healthcare)

- Health Insurance Portability and Accountability Act
- Required for healthcare data handling
- Privacy and security rules
- Not applicable unless handling PHI

SOC 2 (Future Consideration)

- Service Organization Control
- Trust principles: Security, availability, processing integrity
- Demonstrates commitment to data security
- Beneficial for enterprise clients

Legal Compliance:

Copyright & Intellectual Property

- All content properly licensed
- No copyright infringement
- Attribution where required
- Terms of use available

CAN-SPAM Act

- If email marketing implemented
- Unsubscribe mechanism required
- Clear sender identification
- Honest subject lines

Terms of Service

- Clear terms of website use
- Liability limitations
- User obligations
- Dispute resolution

Privacy Policy

- Comprehensive privacy policy
- Easy to access (footer link)
- Plain language explanation
- Updated as practices change

SEO & Marketing Compliance:

Search Engine Guidelines

- Google Webmaster Guidelines compliance
- No black-hat SEO techniques
- Honest, accurate meta descriptions
- No cloaking or deceptive practices

Advertising Standards

- Truthful advertising
- No misleading claims
- Substantiated statements
- Clear disclosures

Geographic Compliance:

Localization Requirements

- Compliance with local laws in target markets
- Currency, date, time formats as appropriate
- Local contact requirements (if applicable)
- Language requirements (future)

Documentation Requirements:

Compliance Documentation

- Privacy policy document
- Cookie policy document
- Terms of service document
- Accessibility statement
- Security policy
- Incident response plan

Audit Trail

- Change logs for compliance-related updates
- Security incident logs
- Privacy-related request handling logs
- Regular compliance audits

6.2 Data Migration, Backup, Recovery

Initial Data Loading:

Content Migration

- **Source:** Marketing team deliverables
- **Format:** Text documents, spreadsheets, images
- **Process:**
 - Content review and approval
 - Asset optimization (images, videos)
 - Hardcoding into Angular components
 - Version control via Git
- **Validation:** Content accuracy review by stakeholders

Asset Migration

- **Images:** Optimize and convert to WebP/JPEG
- **Icons:** Convert to SVG format
- **Logos:** Multiple formats and sizes
- **Fonts:** License verification and loading
- **Process:** Upload to CDN or include in build

No Database Migration

- Static site with no database initially
- No legacy data to migrate
- Future CMS integration will require migration plan

Backup Strategy:

Code Backup

- **Version Control:** Git repository (GitHub, GitLab, or Bitbucket)
- **Frequency:** Continuous (every commit)
- **Branches:**
 - Main/Production branch
 - Development branch

- Feature branches
- **Retention:** Indefinite (full history)
- **Remote Backups:** Multiple remote repositories
- **Access Control:** Role-based repository access

Build Artifacts

- Production builds stored in artifact repository
- Tagged releases for each deployment
- Retention: Last 10 production builds minimum
- Quick rollback capability

Configuration Files

- Environment configurations in version control
- Secrets stored in secure vault (not in repository)
- Infrastructure as Code (IaC) for hosting configuration
- Backup of deployment scripts

Asset Backup

- **Images:** Stored in version control (optimized versions)
- **Original Assets:** Separate backup repository
- **CDN Assets:** Replicated across CDN nodes
- **Frequency:** With each content update
- **Storage:** Cloud storage (S3, Azure Blob, etc.)

Content Backup

- All page content in source control
- Content versioning through Git commits
- Ability to restore previous versions
- Change history fully tracked

Recovery Procedures:

Code Recovery

- **Scenario:** Corrupted or lost code

- **Procedure:**
 1. Identify last known good commit
 2. Clone repository from remote
 3. Checkout specific commit or branch
 4. Rebuild and test locally
 5. Redeploy if necessary
- **RTO (Recovery Time Objective):** < 15 minutes
- **RPO (Recovery Point Objective):** Last commit (typically < 1 day)

Deployment Rollback

- **Scenario:** Faulty production deployment
- **Procedure:**
 1. Identify issues in production
 2. Trigger rollback to previous build
 3. Restore previous deployment artifacts
 4. Verify functionality
 5. Investigate and fix root cause
- **RTO:** < 5 minutes for rollback
- **Automated:** Rollback scripts ready

Asset Recovery

- **Scenario:** Lost or corrupted assets
- **Procedure:**
 1. Identify missing/corrupted assets
 2. Retrieve from Git repository or backup storage
 3. Re-optimize if necessary
 4. Re-upload to CDN
 5. Clear CDN cache
 6. Verify asset loading
- **RTO:** < 30 minutes
- **RPO:** Last asset update

Configuration Recovery

- **Scenario:** Lost configuration files
- **Procedure:**
 1. Retrieve from version control
 2. Restore environment variables from secure vault
 3. Validate configuration
 4. Redeploy with correct configuration
- **RTO:** < 15 minutes

Complete Site Recovery

- **Scenario:** Total site loss (hosting provider failure)
- **Procedure:**
 1. Provision new hosting environment
 2. Clone repository from remote Git
 3. Restore configuration and secrets
 4. Run production build
 5. Deploy to new hosting
 6. Update DNS records
 7. Restore CDN configuration
 8. Verify full functionality
- **RTO:** < 2 hours
- **RPO:** Last commit (typically < 1 day)

Business Continuity:

Disaster Recovery Plan

- Documented recovery procedures
- Designated recovery team members
- Communication plan for stakeholders
- Regular DR testing (quarterly)
- Alternative hosting providers identified

High Availability

- CDN provides geographic redundancy

- Multiple DNS servers
- Static site architecture minimizes dependencies
- No single point of failure in content delivery

Data Loss Prevention

- Multiple Git remotes for redundancy
- Automated backups of repositories
- Regular backup testing
- Commit discipline (frequent, meaningful commits)

Monitoring & Alerts

- Uptime monitoring with alerts
- Deployment success/failure notifications
- Git repository access monitoring
- CDN performance monitoring

Future Considerations:

CMS Integration Backup

- Database backup strategy when CMS added
- Content versioning system
- Automated daily backups
- Point-in-time recovery capability

User-Generated Content

- Backup strategy for form submissions (via backend)
- User account data backup (when authentication added)
- Regular backup schedule
- GDPR-compliant data retention

Form Submission Data

- Backend API handles persistence
- Database backups managed by backend team
- No form data stored in frontend

- Submission confirmations provided to users
-

7. Appendices

7.1 Glossary

Technical Terms:

Angular A TypeScript-based open-source web application framework developed by Google for building single-page applications.

API (Application Programming Interface) A set of protocols and tools for building software applications, specifying how software components should interact.

ARIA (Accessible Rich Internet Applications) A set of attributes that define ways to make web content and applications more accessible to people with disabilities.

Bundling The process of combining multiple files (JavaScript, CSS) into fewer files to optimize loading performance.

CDN (Content Delivery Network) A geographically distributed network of proxy servers and data centers that delivers web content to users based on their location.

CORS (Cross-Origin Resource Sharing) A mechanism that allows restricted resources on a web page to be requested from another domain outside the domain from which the first resource was served.

CSS (Cascading Style Sheets) A style sheet language used for describing the presentation of a document written in HTML.

DOM (Document Object Model) A programming interface for HTML and XML documents that represents the page structure as a tree of objects.

Gzip A file compression format and software application used to compress and decompress files for faster transmission.

HTTPS (Hypertext Transfer Protocol Secure) An extension of HTTP that uses encryption for secure communication over a computer network.

JSON (JavaScript Object Notation) A lightweight data-interchange format that is easy for humans to read and write and easy for machines to parse and generate.

Lazy Loading A design pattern that defers initialization of an object until it is needed, commonly used to improve performance.

Responsive Design An approach to web design that makes web pages render well on a variety of devices and window sizes.

REST (Representational State Transfer) An architectural style for designing networked applications using stateless communication and standard HTTP methods.

RxJS (Reactive Extensions for JavaScript) A library for reactive programming using observables, making it easier to compose asynchronous or callback-based code.

SEO (Search Engine Optimization) The practice of increasing the quantity and quality of traffic to a website through organic search engine results.

SPA (Single Page Application) A web application that interacts with users by dynamically rewriting the current page rather than loading entire new pages.

SSL/TLS (Secure Sockets Layer/Transport Layer Security) Cryptographic protocols designed to provide secure communication over a computer network.

TypeScript A strongly typed programming language that builds on JavaScript, adding static type definitions.

UI/UX (User Interface/User Experience) UI refers to the visual elements users interact with; UX refers to the overall experience and satisfaction when using a product.

Viewport The visible area of a web page within the browser window.

WebP A modern image format that provides superior compression for images on the web, supporting both lossy and lossless compression.

Business Terms:

B2B (Business-to-Business) Commercial transactions between businesses rather than between a business and individual consumers.

Chatbot An AI-powered software application that conducts conversations with users via text or voice interfaces.

CTA (Call-to-Action) A prompt that encourages users to take a specific action, typically represented as a button or link.

KPI (Key Performance Indicator) A measurable value that demonstrates how effectively a company is achieving key business objectives.

Lead Generation The process of attracting and converting prospects into potential customers who have expressed interest in products or services.

LLM (Large Language Model) An AI model trained on vast amounts of text data to understand and generate human-like text.

Multi-agent System A system composed of multiple intelligent agents that interact or work together to solve problems beyond individual capabilities.

POC (Proof of Concept) A demonstration to verify that certain concepts or theories have practical potential.

R&D (Research and Development) Creative work undertaken to increase knowledge and devise new applications of available knowledge.

ROI (Return on Investment) A performance measure used to evaluate the efficiency of an investment, calculated as $(\text{gain} - \text{cost}) / \text{cost}$.

SMB/SME (Small and Medium Business/Enterprise) Businesses with employee counts or revenue below certain thresholds, typically fewer than 500 employees.

Value Proposition A statement that explains what benefit a product or service provides, how it solves customer problems, and why it's better than alternatives.

Project-Specific Terms:

Bordered Box Design A visual design element used throughout the Spartex AI website featuring content enclosed in rectangular borders for structure and emphasis.

Case Study Card A component displaying success stories with quantifiable results, client information, and a link to detailed information.

Hero Section The prominent area at the top of a webpage, typically containing headline messages and primary calls-to-action.

Industry Card A component showcasing Spartex AI's expertise in specific industry verticals with icons and descriptions.

Process Steps The four-step methodology (R&D, Prototypes/POCs, Understanding, Solution Design) that Spartex AI uses to serve clients.

Service Card A component displaying one of Spartex AI's core service offerings with icon, description, and call-to-action.

Smooth Scroll An animated scrolling effect that gradually moves the viewport to a target section rather than jumping instantly.

Section Navigation The ability to navigate between different areas of the single-page application using the navigation menu or URL fragments.

7.2 Revision History

Version	Date	Author	Description
1.0	October 06, 2025	Business Analyst	Initial SRS document creation for Spartex AI Homepage

7.3 Approval Signatures

Document Approval:

This Software Requirements Specification has been reviewed and approved by:

Business Stakeholders:

- Marketing Director: _____ Date: _____
- Product Owner: _____ Date: _____

Technical Leadership:

- Technical Lead: _____ Date: _____
- Solution Architect: _____ Date: _____

Development Team:

- Frontend Lead: _____ Date: _____
- QA Lead: _____ Date: _____

Project Management:

- Project Manager: _____ Date: _____
-

7.4 Assumptions Log

1. **Content Delivery:** All final content, copy, and marketing materials will be delivered by the marketing team within agreed timelines.
2. **Design Assets:** High-quality design assets (logos, icons, images) will be provided in web-optimized formats.
3. **Backend API:** Backend development team will provide API specifications and endpoint URLs for contact form submission.

4. **Brand Guidelines:** Complete brand guidelines including colors, typography, and visual identity will be available.
 5. **Browser Support:** Target audience uses modern browsers; no Internet Explorer support required.
 6. **Hosting Infrastructure:** DevOps team will provide appropriate hosting infrastructure with CDN support.
 7. **Third-Party Services:** All required third-party services (fonts, analytics, icons) are approved and accessible.
 8. **Authentication Future:** Login/Sign Up functionality will integrate with separately developed authentication service.
 9. **Mobile Traffic:** Significant portion (40-50%) of traffic will come from mobile devices.
 10. **Performance Targets:** Hosting environment will support performance targets outlined in this document.
-

7.5 Dependencies Log

External Dependencies:

1. **Marketing Team:** Content delivery, case study details, company information
2. **Design Team:** Visual design specifications, brand guidelines, asset creation
3. **Backend Team:** API endpoint specifications, authentication service
4. **DevOps Team:** Hosting configuration, SSL certificates, DNS setup
5. **Legal Team:** Privacy policy, terms of service, compliance review
6. **Third-Party Services:** CDN provider, font service, analytics platform

Technical Dependencies:

1. **Angular 19:** Framework must be stable and released
2. **Node.js:** Required for build tools and development
3. **NPM Packages:** External libraries must be available and maintained
4. **Browser Standards:** Target browsers must support required APIs
5. **CDN Service:** Must be provisioned and configured
6. **SSL Certificate:** Must be obtained and installed

Timeline Dependencies:

1. Content must be finalized before development completion
2. Design assets must be available during implementation
3. API specifications needed before form integration testing

4. Hosting infrastructure required before deployment testing
 5. QA testing dependent on feature completion
-

7.6 Risk Register

Risk Identification and Mitigation:

R1: Content Delivery Delays

- **Probability:** Medium
- **Impact:** High (delays development completion)
- **Mitigation:** Use placeholder content, parallel development tracks, early content submission deadlines
- **Contingency:** Phased deployment with content updates post-launch

R2: Third-Party Service Unavailability

- **Probability:** Low
- **Impact:** Medium (affects functionality)
- **Mitigation:** Graceful degradation, alternative providers identified, fallback strategies
- **Contingency:** Local asset hosting, alternative services

R3: Browser Compatibility Issues

- **Probability:** Medium
- **Impact:** Medium (affects user experience)
- **Mitigation:** Comprehensive testing, feature detection, polyfills
- **Contingency:** Progressive enhancement, browser-specific fixes

R4: Performance Not Meeting Targets

- **Probability:** Medium
- **Impact:** High (affects user satisfaction)
- **Mitigation:** Performance monitoring, optimization best practices, early testing
- **Contingency:** Additional optimization sprint, infrastructure upgrades

R5: Backend API Delays

- **Probability:** Medium
- **Impact:** High (blocks form functionality)

- **Mitigation:** Early API specification, mock API for testing, parallel development
- **Contingency:** Temporary email-based contact, phased feature deployment

R6: Accessibility Compliance Failure

- **Probability:** Low
- **Impact:** High (legal risk, user exclusion)
- **Mitigation:** Accessibility-first development, automated testing, manual audits
- **Contingency:** Remediation sprint, accessibility specialist consultation

R7: Security Vulnerabilities

- **Probability:** Medium
- **Impact:** Critical (data breach, reputation damage)
- **Mitigation:** Security best practices, dependency scanning, penetration testing
- **Contingency:** Immediate patching, incident response plan

R8: Scope Creep

- **Probability:** High
- **Impact:** Medium (timeline, budget)
- **Mitigation:** Clear requirements, change control process, stakeholder communication
- **Contingency:** Prioritization, phased releases

R9: Angular Framework Issues

- **Probability:** Low
- **Impact:** High (technical blocker)
- **Mitigation:** Use stable Angular version, monitor releases, test early
- **Contingency:** Framework version rollback, alternative approach

R10: Hosting/Infrastructure Issues

- **Probability:** Low
 - **Impact:** High (service unavailability)
 - **Mitigation:** Reliable hosting provider, monitoring, redundancy
 - **Contingency:** Alternative hosting provider, disaster recovery plan
-

7.7 Success Criteria

Project Success Metrics:

Functional Success:

- All functional requirements implemented as specified
- All user stories and acceptance criteria met
- Zero critical bugs at launch
- All sections display correctly across supported browsers and devices
- Contact form successfully submits data to backend API
- Navigation functions smoothly across all sections

Performance Success:

- Lighthouse Performance score ≥ 90
- First Contentful Paint ≤ 1.5 seconds
- Largest Contentful Paint ≤ 2.5 seconds
- Time to Interactive ≤ 3.5 seconds
- All performance targets met on target devices

Quality Success:

- Code review approval from technical lead
- Unit test coverage $\geq 80\%$
- Zero accessibility violations (WCAG 2.1 AA)
- Zero high-severity security vulnerabilities
- Browser compatibility validated across all target browsers

User Experience Success:

- Average task completion time within targets
- User satisfaction score $\geq 4/5$ in testing
- Zero navigation confusion in user testing
- Mobile usability rating $\geq 90/100$

Business Success:

- Project delivered on time and within budget

- Stakeholder approval obtained
- Marketing team satisfied with content presentation
- Lead generation capability functional
- Brand identity effectively communicated

Technical Success:

- Clean, maintainable code structure
 - Comprehensive documentation provided
 - Deployment pipeline functional
 - Monitoring and analytics implemented
 - Scalability requirements met
-

Document Summary

This Software Requirements Specification (SRS) provides a comprehensive blueprint for the development of the Spartex AI Homepage, a single-page Angular 19 application designed to establish professional online presence, showcase AI-as-a-Service offerings, and generate qualified leads.

Key Deliverables:

- Responsive single-page application with 8 major sections
- Four service offerings with detailed presentations
- Six industry vertical showcases
- Three compelling case studies with quantifiable results
- Fully functional contact form with validation
- WCAG 2.1 AA compliant accessible interface
- High-performance, secure, and scalable architecture

Technology Stack:

- Frontend: Angular 19 with TypeScript
- Styling: CSS3/SCSS with responsive design
- Build: Angular CLI with Vite bundler
- Hosting: Static hosting with CDN delivery

Critical Success Factors:

- Timely content delivery from marketing team
- Performance targets achieved (Lighthouse ≥ 90)
- Accessibility compliance (WCAG 2.1 AA)
- Security best practices implemented
- Cross-browser compatibility ensured
- Mobile-first responsive design executed

This document serves as the single source of truth for all development, testing, and acceptance activities related to the Spartex AI Homepage project.

End of Software Requirements Specification Document

For questions, clarifications, or change requests, please contact the Project Manager or Business Analyst.