

Phase 1

Accessible PDF Viewer with AI-Powered Content Analysis

Jan 2025

Group F

Ilesanmi Asaolu

Isaac Izehi

Bingchen Yang

1. Software Requirements Specification

1.1 Introduction

1.1.1 Purpose

To provide a detailed description of the Accessible PDF View with AI-Powered Content Analysis. It covers the features, functionalities, and constraints of the product.

1.1.2 Scope

This section covers requirements for developing an accessible PDF viewer which assists visually impaired users with integration from AI-powered content analysis. Key features include:

- Ensuring complex content such as tables, mathematical formulas, and graphical figures is made understandable.
- Providing inclusive access to documents for:
 - Academic institutions.
 - Government agencies.
 - Businesses prioritizing accessibility compliance.
- Operating offline to ensure:
 - User privacy.
 - Performance reliability.

1.2. Overall Description

1.2.1 Product Perspective

The application will be an plugin/widget/extension embedded in the web browsers. With AI-driven content analysis and text-to-speech functionalities, the application is meant to enhance the accessibility of PDF documents or web pages by:

- Prioritizing accessibility.
- Utilizing AI models for content processing and explanation.

1.2.2 Product Functions

- Embedding and integrating the PDF viewer into web pages.
- Analyzing PDF content offline using AI tools to extract and structure content (text, tables, figures, and formulas).
- Providing clear, descriptive explanations of structured content, including complex elements like charts, graphs, and equations.
- Supporting text-to-speech functionalities for real-time audio output.
- Ensuring compatibility with screen readers and other assistive technologies.

- Supporting multiple languages for inclusivity.
- Allowing customization based on user preferences, such as language and content verbosity.

1.2.3 User Classes and Characteristics

- Primary Users:
 - Blind or visually impaired individuals who rely on assistive technologies for comprehension on documents and web pages.
 - Depend on keyboard navigation or voice commands for interaction.
- Secondary Users:
 - Accessibility-focused web developers and institutions who want to embed the PDF viewer into their webpages.
 - Content creators who need to ensure their PDF documents comply with accessibility criteria.

1.2.4 Operating Environment

- Web Browsers: The PDF viewer will be compatible with modern web browsers.
- Devices: The viewer will be optimized responsively for use on desktop computers, laptops, tablets, and mobile devices.
- Assistive Technologies: The product will support popular screen readers (JAWS, NVDA, VoiceOver) and text-to-speech tools to ensure accessibility.
- Operating Systems: The system will focus only on function on Windows.

1.2.5 Constraints

- Offline Processing: work offline to ensure data privacy.
- Assistive Technology Compatibility: compatible with major screen readers and VoiceOver.
- Multi-Language Support: multiple languages to fulfill user diversity.
- Efficient Processing: handle limited file sizes for efficient processing without significant delays.

1.2.6 Assumptions and Dependencies

Assumptions:

- Users will have access to modern web browsers (e.g., Chrome, Firefox, Edge, Safari) that support HTML5 and JavaScript.
- Users will have the necessary hardware (desktop, laptop, tablet, or mobile device) to access the PDF viewer.
- Users will use screen readers and other assistive technologies compatible with the PDF viewer.

- The AI tools and models used for content analysis will be available and functional for offline processing.
- Input PDFs will comply with standard formats for parsing.

Dependencies:

- The performance of the AI content analysis depends on the efficiency and accuracy of the AI tools and models.
- The integration of the PDF viewer into web pages depends on the compatibility and stability of the hosting environment.
- The quality and functionality of the text-to-speech output depend on the compatibility with the user's assistive technologies.
- The system's ability to support multiple languages relies on the availability of accurate translation and localization tools.

1.3. Specific Requirements

1.3.1 External Interface Requirements

User Interfaces:

- PDF Viewer Interface: The viewer shall provide controls for navigation. It shall be responsive and adaptable to various screen sizes.
- Accessibility Interface: The system shall support screen readers and VoiceOver. It shall provide text-to-speech options for real-time reading of content. Users will interact through standard input devices (keyboard, mouse) or voice commands.

Hardware Interfaces:

- The system shall be compatible with standard web browsers on desktops, tablets, and mobile devices.
- No special hardware requirements are necessary beyond a device capable of running modern web browsers.

Software Interfaces:

- integrate seamlessly into any webpage.
- interface with AI tools for offline content analysis.
- use standard web technologies for the user interface.
- interface with text-to-speech APIs for real-time audio output.
- integrate with screen readers and modern web browsers.

1.3.2 Functional Requirements

- **PDF Viewer Integration:**
 - Embed a responsive PDF viewer into web pages.
 - Provide controls for navigation, such as next page, previous page, and zoom.
- **AI-Based Content Analysis:**
 - Analyze PDF content offline using AI tools to extract and structure content (text, tables, figures, formulas).
 - Generate descriptive explanations for complex elements like charts, graphs, and equations.
- **Text-to-Speech Functionality:**
 - Provide text-to-speech options for real-time reading of content.
 - Support customization of speech output based on user preferences (e.g., language, speed).
- **Accessibility Support:**
 - Ensure compatibility with popular screen readers (JAWS, NVDA, VoiceOver).
 - Provide keyboard navigation and voice command support.
- **Multi-Language Support:**
 - Support multiple languages to cater to diverse user groups.
 - Allow users to select their preferred language for content and text-to-speech output.
- **User Customization:**
 - Allow customization based on user preferences, such as language selection and content verbosity.
 - Save user preferences for future sessions.
- **Content Processing:**
 - Extract and process PDF content accurately.
 - Convert extracted content into structured data.
 - Generate descriptive explanations for complex content.

1.3.3 Performance Requirements

- **Response Time:** Process and analyze a PDF within 10 seconds for files up to 10 MB.
- **Accuracy:**
 - The AI tools shall achieve a minimum accuracy of 95% in extracting and structuring content from PDFs.
 - The text-to-speech functionality shall provide clear and accurate audio output without noticeable delays.

1.3.4 Software System Attributes

- **Reliability:**
 - The system shall have an uptime of 99.9% to ensure continuous availability.
 - The system shall operate without crashes for standard-compliant PDFs.
- **Security:**
 - The system shall ensure secure offline processing of sensitive PDF data.
- **Maintainability:**
 - The system shall be designed using modular architecture to facilitate easy maintenance and updates.
 - The codebase shall follow standard coding practices to ensure readability and maintainability.
- **Scalability:**
 - The AI content analysis module shall be designed to handle increased data volume without performance degradation.
- **Usability:**
 - The user interface shall be intuitive and user-friendly to ensure ease of use.
 - The system shall provide helpful error messages and guidance to assist users in resolving issues.
- **Performance:**
 - The system shall meet the specified performance requirements for response time, accuracy, and efficiency.

2. Use Case Diagrams & Descriptions

Use Case 1: PDF Content Accessibility

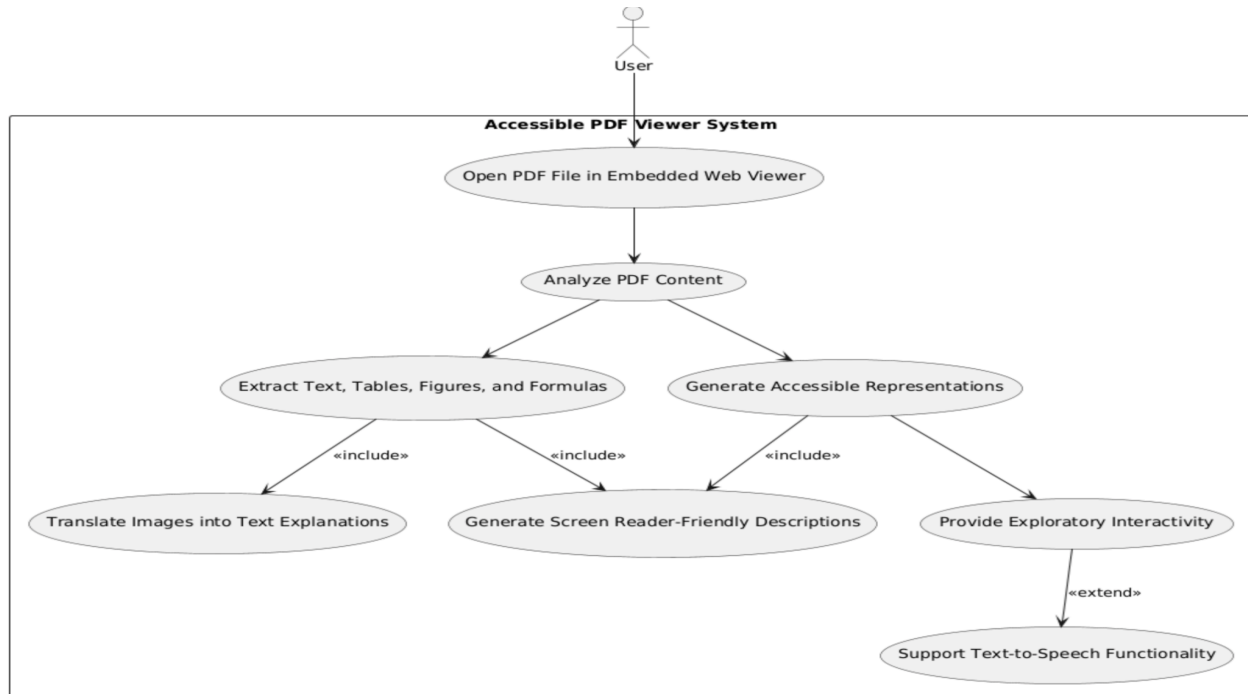


Diagram Explanation:

1. Actors:

- The User interacts with the Accessible PDF Viewer system.

2. System Boundary:

- Contains all functionality provided by the system

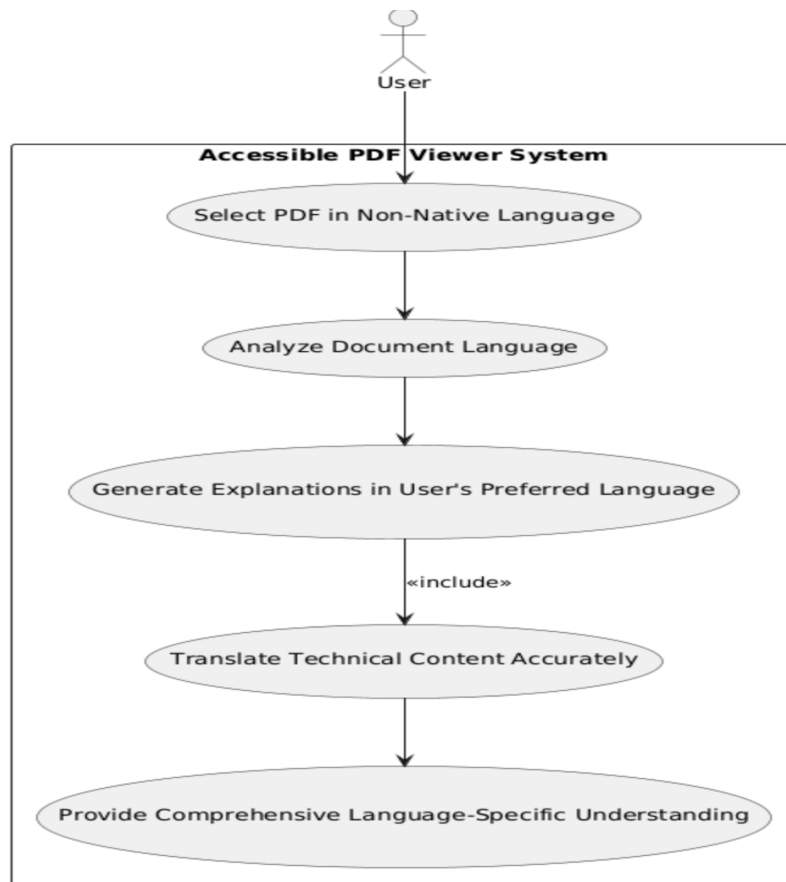
3. Use Cases:

- Each requirement is broken down into separate use cases
- Use cases may be linked to each other (eg "Analyze PDF content" includes content extraction and presentation generation).

4. Relationships:

- **Contains:** Represents required subtasks (eg "Generate descriptions suitable for screen readers")
- **Enhanced:** represents an optional task or condition (eg, text-to-speech enhances interactivity)
-

Use Case 2: Multi-Language Content Explanation



Actors:

- The user is the primary actor interacting with the system

System Boundary:

- The Accessible PDF Viewer System rectangle contains all system functionality for this use case

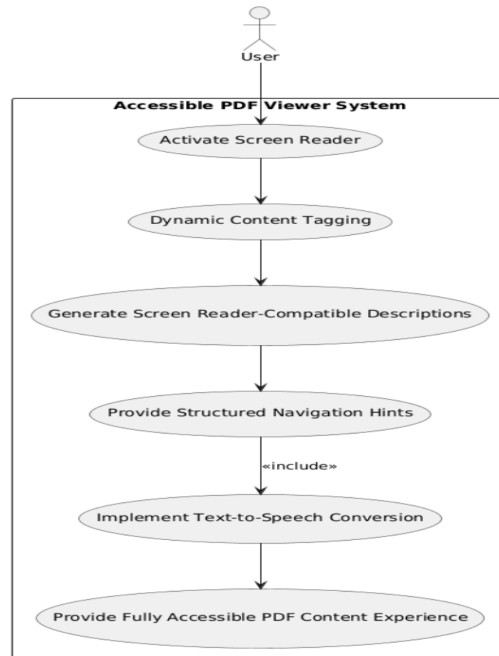
Use Cases:

- **Select PDF in Non-Native Language:** A launch action initiated by the user.
- **Analyze Document Language:** The system determines the document language.
- **Generate Description in User's Preferred Language:** The system generates a content description for the user's preferred language.
- **Correctly translate technical content:** This ensures that complex terms, formulas, and tables are translated correctly
- **Provide comprehensive language-specific understanding:** The end goal is for users to fully understand the content.

Relationships:

- **Content:** Translating technical content is a necessary step in preparing your instructions

Use Case 3: Advanced Accessibility Features



Actors:

- The user initiates the enhanced accessibility features by activating a screen reader.

System Boundary:

- The rectangular Accessible PDF Viewer system contains all use cases related to the advanced accessibility features

Use Cases:

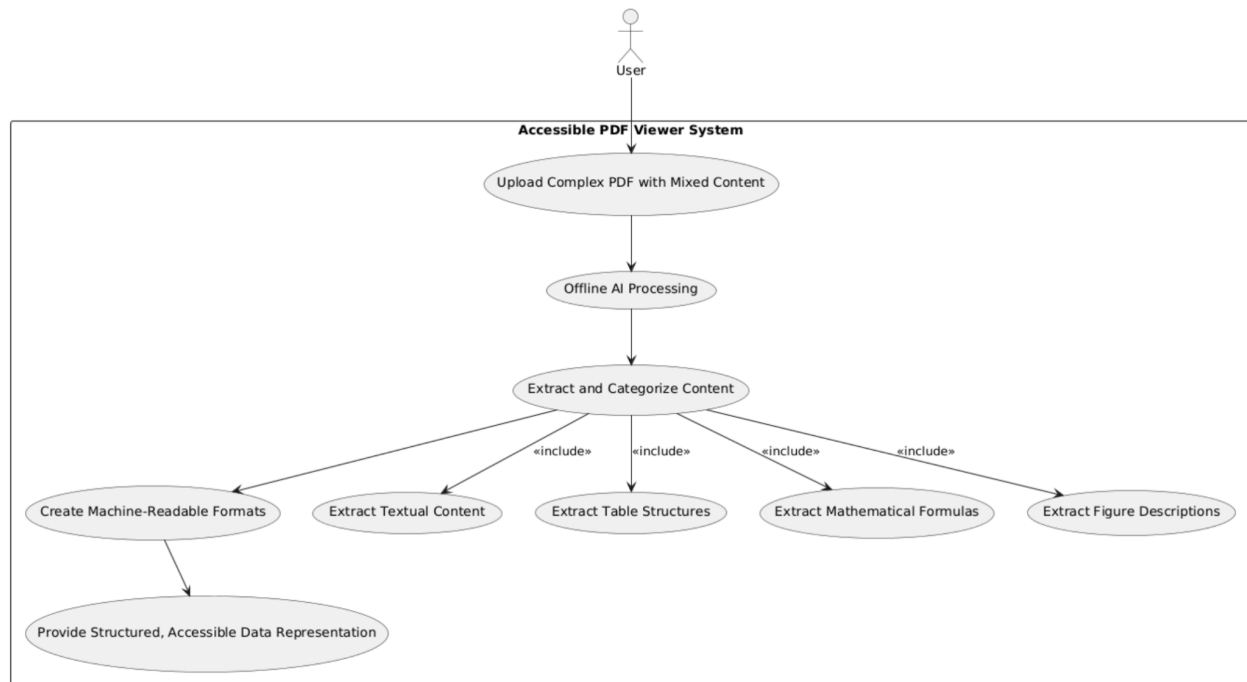
- **Enable Screen Reader:** The entry point to initiate the accessibility features
- **Dynamic Content Tagging:** Automatically and dynamically tags content for accessibility
- **Generates Screen Reader Compatible Descriptions:** Converts visual content into screen reader compatible descriptions
- **Provides notes on structured navigation:** Allows users to navigate the document effectively

- **Implements text-to-speech conversion:** Adds speech conversion of text content
- **Provides a fully accessible PDF content experience:** The end goal is to ensure a seamless and accessible experience for users

Relationships:

- **Content:** Text-to-speech conversion is a required feature of structured navigation
- **Order:** Each step flows naturally into the next and represents a process of creating a fully accessible experience

Use Case 4: AI-Powered Content Extraction



Actors:

- User initiates the process by uploading a complex PDF file.

System Boundary:

- Rectangular, accessible PDF viewer the system includes AI-powered content extraction

Use Cases:

- **Upload of a complex PDF file with mixed content:** User action to provide input
- **Offline AI Processing:** The system processes the PDF files offline for privacy and efficiency reasons
- **Extract and classify content:** Split content into specific types (text, tables, formulas, figures)
- **Create Machine-Readable Format:** Convert classified content into a structured, accessible format
- **Provides a structured and accessible representation of data:** It is the final output that is ready for user interaction

Sub-Use Cases (Under Content Extraction):

- **Extract Textual Content:** Extracts readable text from the PDF files.
- **Extract Table Structures:** Identifies and formats tabular data.
- **Extract Mathematical Formulas:** Processes equations and formulas for accessibility.
- **Extract Figure Descriptions:** Converts visual figures into descriptive text.

Relationships:

- **Includes:** Sub-use cases (e.g., "Textual Content Extraction") are integral part of content extraction and classification

3. High-Level Architecture Design

3.1 Logical Diagram:

3.1.1 User Interface

- Web Page (Embedded PDF Viewer)

3.1.2 Application Layer

- PDF Parser
- AI Analysis Module
- Content Explanation Engine
- Text-to-Speech Engine

3.1.3 Data Layer

- Structured Data Storage
- User Preferences

3.2 Physical Diagram:

3.2.1 Client Device

- Web Browser
- Assistive Technologies
- AI Tools
- Database

3.3 Description and Explanation

The client device hosts the user interface within a web browser, which includes the embedded PDF Viewer. Users interact with the viewer to load and navigate PDF documents. The viewer sends the PDF content to the AI Content Analysis Module, which processes and analyzes the content offline within the client device. The Content Explanation Engine then converts the structured data into accessible descriptions.

The Text-to-Speech Engine generates audible output that is played back on the client device. The client device's screen readers and other assistive technologies ensure the content is accessible.

User preferences and configuration settings are managed within the application, ensuring a fully offline operation.

3.4 Logical Architecture Diagram

3.4.1 Client Layer

- User Interface: The PDF Viewer embedded in a webpage, providing basic navigation controls, responsive design, and user interaction.
- Accessibility Tools: Integrated screen reader compatibility and text-to-speech functionality.

3.4.2 Application Layer

- PDF Parser: Processes the PDF content to extract text, tables, figures, and formulas.
- AI Analysis Module: Uses offline AI tools to analyze the extracted content and generate structured data.
- Content Explanation Engine: Converts structured data into accessible descriptions and explanations in multiple languages.
- Text-to-Speech Engine: Reads the extracted and explained content aloud in real-time.

3.4.3 Data Layer

- Structured Data Storage: Temporary storage for the structured data generated from PDF content.
- Configuration and Preferences: Stores user preferences and configuration settings for the viewer and accessibility features

3.5 Physical Architecture Diagram

3.5.1 Client Device

- Web Browser: Runs the embedded PDF Viewer and interacts with the user.
- Assistive Technologies: Screen readers (JAWS, NVDA, VoiceOver) and text-to-speech systems installed on the client device.
- AI Tools: Contains the AI models and processing tools used for offline analysis of PDF content.
- Local: Stores structured data and user preferences.

4. Individual Contributions:

Ilesanmi Asaolu:

- Research on requirement specification
- Draft on requirement specification
- Revise on use case diagram
- Revise on high level architecture design
- Finalize requirement specification

Isaac Izehi:

- Research on use case diagram
- Draft on use case diagram
- Revise on high level architecture design
- Finalize use case diagram

Bingchen Yang:

- Draft on high level architecture design
- Revise on requirement specification
- Finalize high level architecture design
- Finalize requirement specification
- Draft phase templates for all sections
- Coordinate progress