**Software Requirements Specification (SRS)**

**Project Title:** PDF to Flipbook Generator

**Table of Contents**

1. **Introduction  
   1.1 Purpose  
   1.2 Scope  
   1.3 Definitions, Acronyms, and Abbreviations  
   1.4 References  
   1.5 Overview**
2. **System Overview**
3. **Stack  
   3.1 Frontend  
   3.2 Backend**
4. **Functional Requirements  
   4.1 PDF Upload  
   4.2 Flipbook Conversion  
   4.3 Customization  
   4.4 Analytics  
   4.5 Mobile Responsiveness**
5. **Non-Functional Requirements  
   5.1 Performance  
   5.2 Scalability  
   5.3 Security  
   5.4 Usability  
   5.5 Compatibility**
6. **Future Enhancements**
7. **Appendix**
   * **Example user flows**
   * **Wireframe sketches of the UI**
8. **Wireframe**
9. **Use case**

**1. Introduction**

1.1 **Purpose**  
The purpose of this document is to define the functional, non-functional, and technical requirements for the development of a PDF to Flipbook Generator. This tool will allow users to upload PDF files and convert them into interactive, web-based flipbooks.

1.2 **Scope**  
The PDF to Flipbook Generator will be a web-based application aimed at users who wish to create engaging digital content from static PDF files. The flipbooks will feature interactive elements like page-turn animations and multimedia support. The tool will support export formats for embedding in websites or sharing via links.

1.3 **Definitions, Acronyms, and Abbreviations**

1. HTML: HyperText Markup Language
2. CSS: Cascading Style Sheets
3. JavaScript: A high-level, interpreted programming language
4. UI: User Interface
5. UX: User Experience
6. API: Application Programming Interface
7. SRS: Software Requirements Specification

1.4 **References**

* Adobe PDF Specifications
* Web Content Accessibility Guideline

1.5 **Overview**

This document outlines the requirements for the PDF to Flipbook Generator, including user needs, system features, and design constraints.

**2. System Overview**

The system will consist of the following components:

* **Frontend:** A user-friendly interface for uploading PDFs, previewing flipbooks, and customizing settings.
* **Backend:** Core logic for PDF processing, rendering flipbooks, and managing user data.
* **Database:** Storage for user-generated flipbooks, settings, and metadata.

**3. Stack**

**Frontend:**

* HTML
* CSS
* JavaScript

**Backend:**

* Database (e.g., SQL, Node.js,Express).

**4. Functional Requirements**

4.1**PDF Upload**

* Support PDF upload up to 100 MB.
* Validate uploaded files for compatibility.

4.2**Flipbook Conversion**

* Convert uploaded PDFs into flipbooks with page-turn animations.
* Maintain the layout, fonts, and embedded links from the original PDF.

4.3**Customization**

* Allow users to configure the following options:
  + Background color or image.
  + Navigation controls (e.g., arrows, thumbnails).

4.4**Analytics**

* Provide analytics for flipbook views, interactions.

4.5**Mobile Responsiveness**

* Ensure that the generated flipbooks are responsive and functional on various devices and screen sizes.

**5. Non-Functional Requirements**

5.1 **Performance**

* The system should convert PDFs to flipbooks within 30 seconds for files under 50 pages.

5.2 **Scalability**

* The system should support up to 1,000 concurrent users.

5.3 **Security**

* Use HTTPS for all data transmission.
* Encrypt user credentials and sensitive data.

5.4 **Usability**

* Ensure the UI is intuitive and accessible per WCAG 2.1 standards.

5.5 **Compatibility**

* Support major browsers (Chrome, Firefox, Safari, Edge).
* Support PDF versions 1.4 and higher.

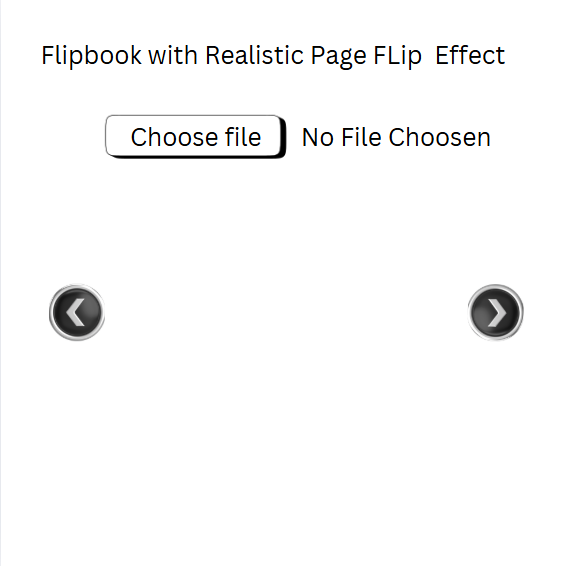
**6. Future Enhancements**

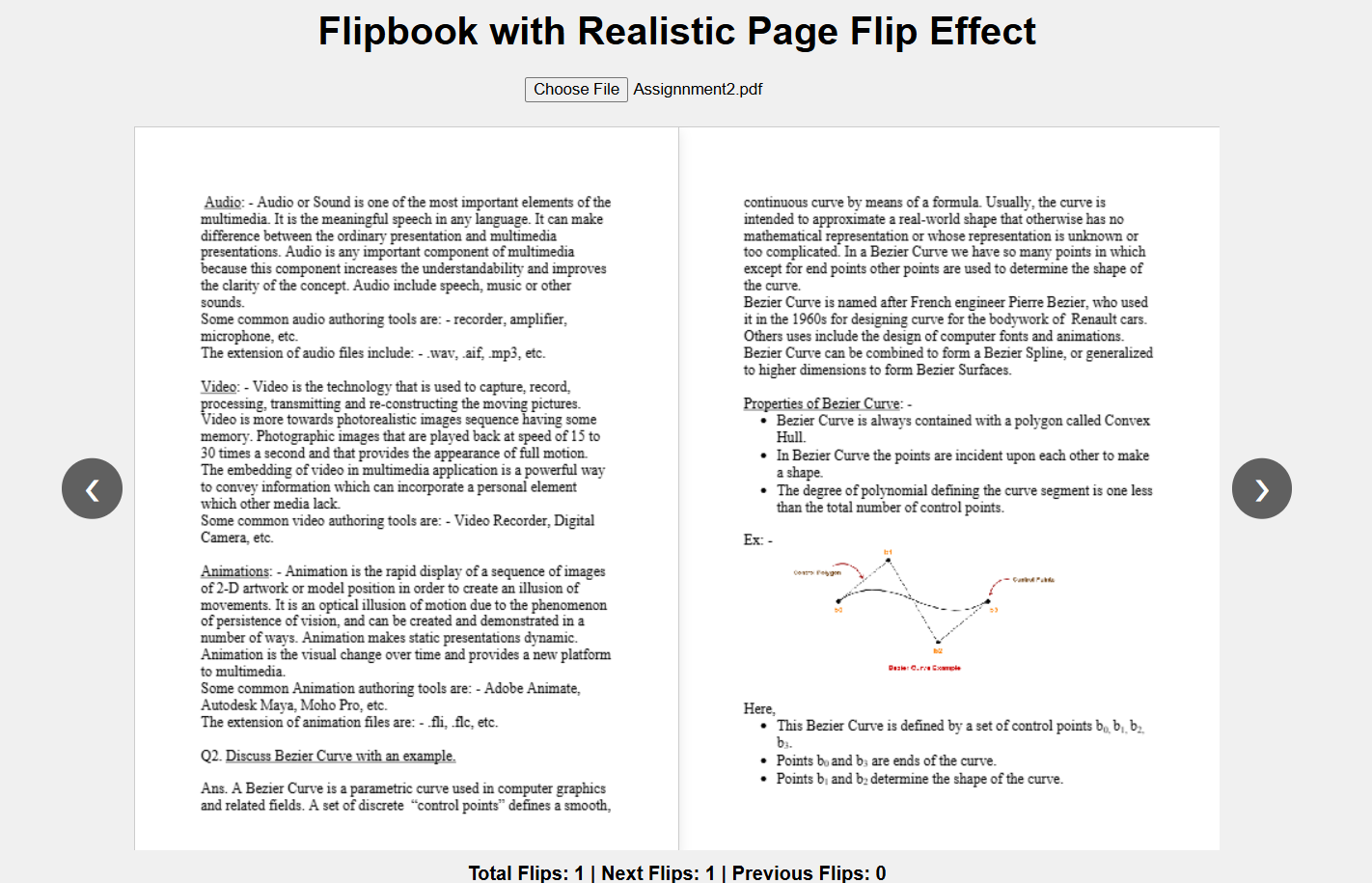
* Add support for collaborative editing of flipbooks.
* Enable integration with CMS platforms like WordPress.
* Provide additional templates for flipbook customization.

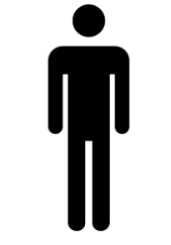
1. **Appendix**

* Example user flows
* Wireframe sketches of the UI

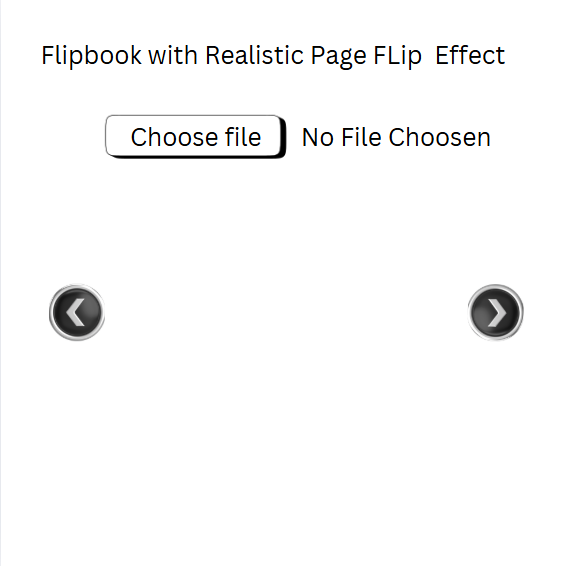
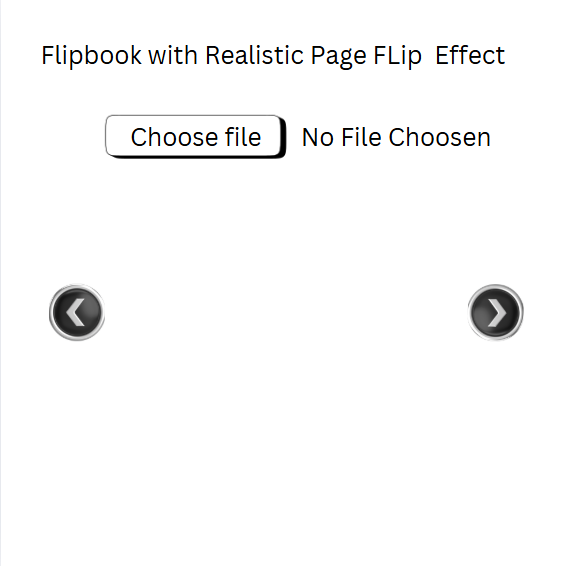
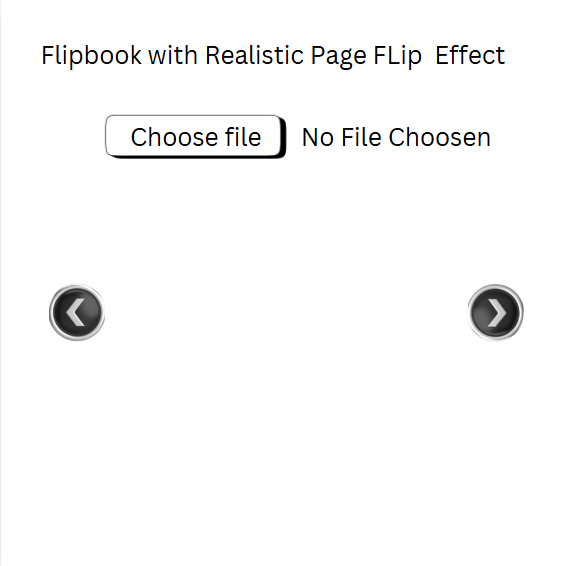
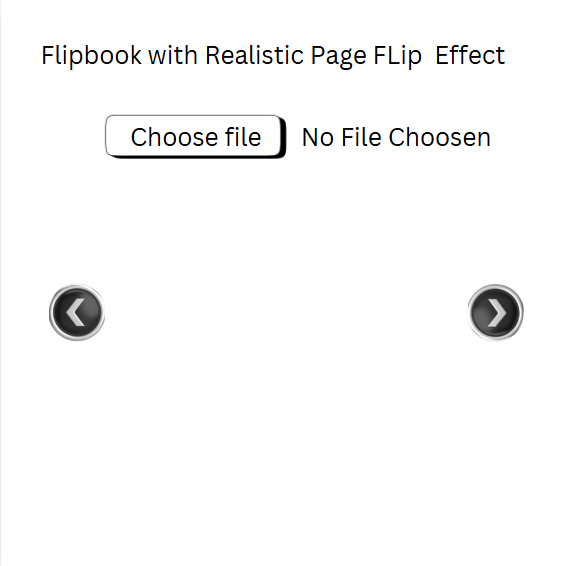
**WIREFRAME**

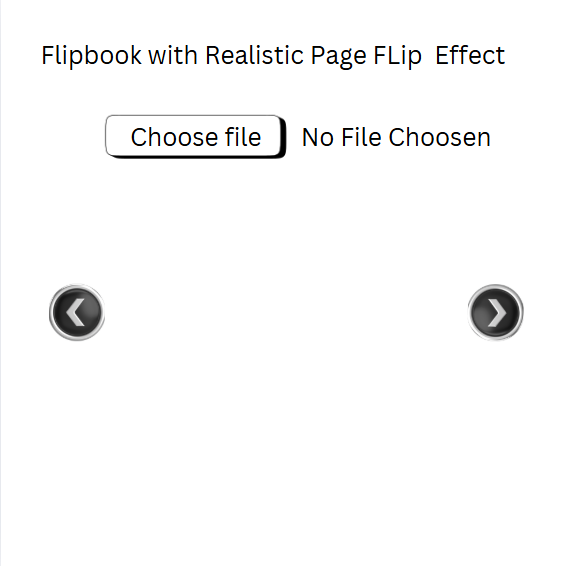




**Use CaseDiagram**

FLIPBOOK SYSTEM





SYSTEM