DyslexAI

Software Requirements Specification

Project Members:

Saba Begum

Mahbubur Khan

Aaron Perel

Ishmael Gonzales

Revised History:

Date	Description	Author	Comments
05/29/2024	Version 1.0	Saba Begum Mahbubur Khan Aaron Perel Ishmael Gonzales	First Draft
06/01/2024	Version 1.1	Saba Begum Mahbubur Khan Aaron Perel Ishmael Gonzales	Revised Draft

Document Approval:

The following software requirements specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
Ishmael Gonzalez	Ishmael Gonzales	Documentation Lead	5/29/2024
Ishmael Gonzales	Ishmael Gonzales	Documentation Lead	6/1/2024

Table of Contents:

1. Introduction

This section gives a scope description and overview of everything included in this SRS document. Also, the purpose for this document is described and a list of abbreviations and definitions is provided.

1.1 Purpose

This document aims to accurately and explicitly describe the requirements for the successful development of the DyslexAI web application. It will clearly define all features expected in the completed application and include the application's different interfaces. The SRS provides a comprehensive list of application expectations and highlights any existing constraints, serving as a guide for our group and stakeholders in defining the scope and deliverables of the project.

1.2 Scope

The DyslexAI web application consists of a user-friendly web interface, a robust backend system, a comprehensive database for managing user data and preferences, and various APIs for seamless communication between the frontend and backend. The application is designed to provide a dyslexia-friendly reading and learning experience, focusing on accessibility and personalization.

The DyslexAI web application includes features such as user authentication, customizable reading interfaces, e-book search and import, AI-powered recommendations, and specialized modes for users with visual or hearing impairments. The application will run on users' computers and mobile devices, providing a consistent experience across different platforms.

The user authentication functionality, managed by Firebase Auth, allows users to securely log in, sign up, reset their passwords or usernames, and access personalized features. This ensures a secure and individualized user experience. The customizable reading interface enables users to adjust font type, size, background color, and spacing to enhance readability. If customization fails, the application will restore default settings to maintain usability.

The e-book search and import functionality allows users to search for and import e-books from an online library. The system supports various e-book formats and returns search results or error messages for unsupported files. AI-powered book recommendations analyze user reading history and preferences to suggest books that match their interests. If the AI recommendation system encounters any issues, the application will display default recommendations to ensure a continuous user experience.

Accessibility modes include Deaf Mode, which converts selected text into images using DALL-E for visual representation, and Blind Mode, which offers text-to-speech functionality, reading aloud interface text to assist navigation. Error handling mechanisms inform users of any issues with these functionalities.

The database, managed by Firebase, will store user credentials, preferences, reading history, and imported e-books. It will maintain accurate information and support real-time updates to ensure data consistency. The database is expected to be available upwards of 99% of the time and should return information quickly to support a responsive user experience.

The APIs, including OpenAI's API and Response Voice API, will handle AI-related requests and audio interactions. These APIs will facilitate the transfer of information to and from the database, ensuring reliable data management and user interaction. The application will be hosted on Vercel, which provides automatic deployment and ensures the most current production build is always live.

Our goal is to have a fully functional DyslexAI web application that includes all mentioned features and functionalities. We aim to create an inclusive, user-friendly platform that enhances the reading experience for users with dyslexia. We are not responsible for hardware purchasing or modification, providing any test hardware, or managing server installations.

1.3 Definition, acronyms, and abbreviations

Term	**Definition**
Adaptive Text Display (ATD)	A technology that changes the display of text to enhance readability for users with dyslexia, such as altering font size, spacing, color contrast, and background.
Dyslexia Font	A type of font designed to mitigate some common reading errors caused by dyslexia, such as letter mirroring and swapping. These fonts often feature heavily weighted bottoms to show direction.
API Endpoint	An API endpoint is a specific URL where an API interacts with a web service to perform tasks like retrieving or changing data.
Application Programming Interface (API)	Application Programming Interface is a set of rules that allows different software applications to communicate and share data with each other.
Firebase	Firebase provides detailed documentation and cross-flatform app development SDKs, to help you build and ship apps for IOS, Android, the Web, Flutter, Unity, and C++.

User Interface (UI)	User Interface defines the way humans interact with the Information systems.
Artificial Intelligence (AI)	Technology that enables computers and digital devices to learn, read, write, create and analyze.
Large Language Model (LLM)	Large Language Models (LLMs) are machine learning models that can comprehend and generate human language text. They work by analyzing massive data sets of language.
User Authentication	User authentication helps to identify verified users and allows them to access accounts and networks securely.
Deaf Mode	An accessibility feature that converts selected text into images using DALL-E to aid users who are deaf or hard of hearing.
Blind Mode	It is an accessibility feature that provides text-to-speech functionality, reading aloud interface text and assisting navigation for users who are blind or visually impaired.
DALL-E	DALL-E is a technology introduced by Open AI and it is a neural network-based picture-generating system. DALL-E is a technology that helps users create new images with their imagination only by using graphics prompts.
OPENAI API	An application programming interface provided by OpenAI that allows the integration of AI functionalities, such as generating text or images, into the application.
Response Voice API	An API that handles audio interactions, converting text to speech to assist users in Blind Mode.
Vercel	A web-based hosting platform designed for hosting web applications, providing features such as automatic deployment and version control.

Kubernetes	An open-source system for automating deployment, scaling, and management of containerized applications.
Amazon Web Services (AWS)	AWS is a cloud platform by Amazon that offers a wide range of services to help businesses and developers build, manage, and scale applications online. It includes services for computing power, storage, databases, machine learning, and more. AWS allows you to run applications without managing physical servers, making it easier to handle varying workloads and grow your applications efficiently.
Error Handling	Error handling is the process of responding to and recovering from error conditions in your program.

Accessibility	The design and implementation of features that ensure the application is usable by people with various disabilities, including visual and hearing impairments.
Data Flow Diagram (DFD)	Shows that flow of information for any process, program, or system.

1.4 References

Wood, Emily, and Rebecca Woodard. “Design for Neurodiverse Learners: ATD.” *Main*, 7 May 2021, www.td.org/magazines/td-magazine/design-for-neurodiverse-learners.

“Home.” *Dyslexie Font*, 24 May 2024, dyslexiefont.com/en/.

“API Endpoints.” *Smartbear.Com*, smartbear.com/learn/performance-monitoring/api-endpoints/. Accessed 29 May 2024.

Google, Google, firebase.google.com/. Accessed 30 May 2024.

"User Interface (UI) Design." GeeksforGeeks, GeeksforGeeks, 13 May 2024, www.geeksforgeeks.org/user-interface-ui/.

"What Is Artificial Intelligence (AI)?" IBM, 25 Aug. 2023, www.ibm.com/topics/artificial-intelligence.

What Is a Large Language Model (LLM)? | Cloudflare, www.cloudflare.com/learning/ai/what-is-large-language-model/. Accessed 30 May 2024.

"What Is User Authentication, and Why Is It Important?" *NordVPN*, 30 May 2024, nordvpn.com/blog/what-is-user-authentication/.

"What Is Dall-E?" GeeksforGeeks, GeeksforGeeks, 30 Apr. 2024, www.geeksforgeeks.org/what-is-dall-e/.

Introduction - Openai API, platform.openai.com/docs/introduction. Accessed 30 May 2024.

"Build and Deploy the Best Web Experiences with the Frontend Cloud." *Vercel*, vercel.com/. Accessed 30 May 2024.

"Production-Grade Container Orchestration." Kubernetes, kubernetes.io/. Accessed 30 May 2024.

"This Page Requires JavaScript." *Swift.Org*, docs.swift.org/swift-book/documentation/the-swift-programming-language/errorhandling/. Accessed 30 May 2024.

1.5 Overview

The introduction was the first of four sections, which have been broken down into subsections. Its primary goal has been to introduce the software requirement document, outline a general overview of the project, and provide readers with the knowledge necessary to interpret the document.

The second section is titled "General Functions". This section briefly describes the application, including the user interface and functions of the application. It also covers characteristics about the user of the application, assumptions and dependencies that impact requirements, and general design constraints.

Requirements make up the third section. This is the longest section of the document. Interface requirements, which are broken down even further, are described first. Functional requirements, which are broken down by their programming layer, follow. This subsection covers all functionality that is expected in the application. This is followed by non-functional requirements, which must be system-wide, design constraints, and logical database requirements.

Fourth, and finally, we have the “Analysis Models” section. This section includes our data flow diagrams, illustrating how information moves throughout our applications.

2. General Factor

2.1 Product perspective

The Dyslexia AI system is a comprehensive web-based application with three main components: the User Interface (UI), the Backend Server, and the API. The UI is designed for individuals with dyslexia, featuring a customizable Reading Interface that allows adjustments to font type, size, color, and spacing to enhance readability. It also includes specialized modes like 'Deaf Mode' and 'Blind Mode' for visual and audio adaptations. The E-book Library enables users to browse, select, and read dyslexia-friendly e-books with personalized settings. The Administration Application manages user accounts, profile settings, and the e-book library, ensuring content is updated and categories are well-organized. The API connects the UI and administration backend to the central database, facilitating data retrieval and updates regarding user preferences, book selections, and reading histories. It also integrates with external services like OpenAI to support advanced features such as text-to-image conversion and book recommendations through natural language processing.

2.2 Product Functions

The Dyslexia AI application offers several functional and non-functional features to enhance the reading experience for users with dyslexia. Functional features are defined by whether the feature is essential to the system's functioning. In contrast, a non-functional feature will be determined by features that are non-essential to its functioning.

The Dyslexia AI system is designed with multiple functional features to provide a comprehensive and accessible reading experience. Key components include User Authentication, E-Book Search and Import, Deaf Mode, Blind Mode, Customizable Reading Interface, Book Recommendations, User Preferences Management, Profile Management, and LLM and AI Integration. User Authentication uses Firebase Auth to securely authenticate users via a multimodal login/sign-up portal. E-Book Search and Import allows users to search for e-books from an online library or import their own, supporting various formats like PDF and ePub. Deaf Mode converts selected text into images for deaf users, while Blind Mode offers text-to-speech functionality to assist blind users. The Customizable Reading Interface lets users adjust settings such as font type, size, color, and spacing for enhanced readability. Book Recommendations use AI to analyze reading history and suggest books, providing personalized suggestions. User Preferences Management allows users to save and retrieve their reading settings, ensuring a tailored experience. Profile Management enables users to update their personal details and preferences. Finally, LLM and AI Integration utilize OpenAI's API for AI-related requests, such as generating images and suggesting e-books, enhancing the application's functionality. These features collectively offer a robust, user-friendly platform tailored to the needs of individuals with dyslexia, ensuring an accessible and personalized reading experience.

The non-functional features of the Dyslexia AI application include performance, updating database information, response time, server error handling, uptime and availability, portability, and scalability. The application is designed to ensure optimal performance, with pages loading and responding to user actions within 2 seconds, overseeing simultaneous user interactions efficiently without significant delays. It ensures that updates to database information, such as user profiles, preferences, reading history, and e-book data, are processed within 2 seconds of the user's request to support data consistency. The system guarantees a maximum response time of 3 seconds for all non-AI API calls, ensuring a seamless user experience. It includes mechanisms for encountering and managing server errors and request errors, employing print statements, error catching, and handling to maintain smooth operations.

The application ensures 24/7 uptime and availability, running on a web server hosted within a Linux environment, utilizing React.js, Node.js, and NPM for development. It is designed for scalability to accommodate more users and increased data volume, leveraging technologies like Kubernetes, Firebase, and Vercel. These features collectively ensure that the Dyslexia AI application is robust, efficient, and capable of handling extensive user interactions and data management effectively.

2.3 Users characteristics

The Dyslexia AI system is designed for three main user groups: individuals with dyslexia, educators, and administrators. Individuals with dyslexia, likely ranging in age from young children to adults, will primarily use the reading interface. This group may have varying levels of computer proficiency, and the system is designed to be user-friendly with customizable text settings to enhance readability. Features such as text-to-speech and content filtering allow users to engage with the content more effectively, regardless of their reading level or technical expertise.

Educators, including teachers and therapists, will use the system to track progress, manage content, and assist in the customization of reading experiences for their students or clients. They are expected to have moderate to advanced technical skills to efficiently navigate and utilize the educational and administrative features of the platform. This enables them to provide targeted support, monitor engagement, and adapt resources to meet the diverse needs of learners with dyslexia.

Lastly, administrators will handle the backend management of the system, including user account management and system settings. This group is expected to possess advanced technical knowledge to oversee the smooth operation of the Dyslexia AI application, perform updates, and ensure data security. Administrators are also responsible for generating reports, ensuring compliance with data protection regulations, and maintaining the overall integrity of the system.

This critical role supports the sustainability and reliability of the service, ensuring that it remains an effective tool for reading enhancement.

2.4 General Constraints

A general constraint that affects the usage of dyslexAI is the users access to an internet connection. While most eBook in general do not require internet to read, DyslexAI does require an internet connection because of three reasons, which the usage of AI (Artificial Intelligence), API, and the Usage of DyslexAI's preference settings. While DyslexAI gives you the option to download your eBook with your desired preferences, you will need an internet connection beforehand to set those preferences. Once you download the eBook with your desire preferences, you can't not modify the preference, unless you reconnect to the internet and download another version of it. Also, the download version does not come with the AI and API features.

2.5 Assumption and dependencies

The Dyslexia AI system operates under several assumptions and dependencies crucial for its functionality. It assumes a stable and continuous internet connection to access cloud services for storing and retrieving user data, e-book content, and system settings. Another key assumption is that users, particularly those with dyslexia, will have basic digital literacy skills to navigate the application's user interface. The system is also dependent on the reliability of external APIs, like OpenAI for text-to-image processing and text-to-speech services, which are integral for the 'Deaf Mode' and 'Blind Mode' functionalities. Additionally, it assumes that educators and administrators will regularly use the platform to update content and manage user settings, which is vital for maintaining the system's effectiveness and relevance. Lastly, the system's performance is contingent upon the proper functioning of user devices, particularly their compatibility with the latest web technologies used in the Dyslexia AI application.

3. Requirement Section

3.1 External Interface Requirements

External interfaces will be the driving forces in our web application. All inputs will be the user via a web browser and their computer hardware.

3.1.2 Hardware Interfacing

The web application will be displayed visually on the user's computer. All inputs will be manual by the user and will stem from mouse interaction or keyboard interaction. This will encompass features such as audibly guiding users to where buttons are on the screen, text input for e-book queries, search queries, and filling in user information.

Graphical displays of e-books, or modified text sizes, font and more for users with Dyslexia or other disabilities will accommodate the users screen size dynamically and will accommodate all screen sizes including mobile phone browsers.

3.1.3 Software Interfacing

We will be using Google's Firebase cloud service to manage all database storage and interactions as well as authentication. Firebase provides real-time database storage; this allows us to increase responsiveness and data concurrency when using user reading history and preferences. All requests will be made via JSON to keep requests easily readable and consistent.

OpenAI's API will be used in our application to manage all AI or LLM related requests. The API returns an object holding the response data as well as status codes and more for error handling. When a user turns on deaf mode the chosen text will be sent to DALL-E to be translated into an image. All book suggestions will be a response from the API after the user's reading history is used to create a prompt for GPT.

The ResponseVoice API will be used to manage all audio interactions while in blind mode. When blind mode is enabled, the API will send all text from the UI upon interaction with it. If a user hovers over any object containing text, the text will be announced. If it is a long text, it will first be announced what type of text it is.

To host our application, we will be using Vercel* a web-based hosting platform built for hosting web applications. Vercel will oversee automatic deployment of the main branch of the GitHub repository to keep the most current production build always live. If issues arise with a production build, we will be able to revert to a previous deployment on the platform ensuring that the platform stays online.

3.1.4 Communications Interfaces

The Dyslexia AI system will be hosted on a cloud-based server, requiring a constant internet connection to ensure it operates smoothly and accurately. This server primarily stores and retrieves essential information such as user preferences, e-book content, and account details. It also handles real-time data synchronization to ensure that updates, such as changes in user settings or new e-book additions, are reflected immediately across all devices. This setup supports the seamless interaction of the Dyslexia AI application with various external APIs, like OpenAI for enhanced features including text-to-speech and image generation in 'Deaf Mode' and 'Blind Mode'.

3.2 Functional Requirements

FR 3.2.1	User Authentication
Description	Using Firebase Auth, authenticate users to access the application via a multimodal login / sign-up portal. Using the keyboard for text-based input.
Inputs	Username, Password, Text from keyboard
Outputs	Access Token generated by Firebase
Error Handling	Error message displayed on the screen or in the user's preferred mode for invalid credentials or unmet requirements

Dependencies	NFR 3.4.2, FR 3.2.7
Priority	High

FR 3.2.2	E-Book Search and Import
Description	Users can search for e-books from an online library or import their own e-books into the application. The system must support various e-book formats and ensure they are accessible within the application.
Inputs	Search query, E-book file (e.g., PDF, ePub)
Outputs	Formatted list of all search results that can be imported.
Error Handling	Display error messages for failed searches or unsupported file formats. Offer user guidance on how to find supported formats.
Dependencies	FR 3.2.1, FR 3.2.5
Priority	High

FR 3.2.3	Deaf Mode
Description	Converts selected text into images to aid deaf users in understanding content through visual representations. Utilizes DALL-E to generate relevant images based on the text which will be provided line by line to generate images using lines of text as prompts.
Inputs	Text selected by the user, speed of image generation i.e., user selects the speed at which text is sent as an input
Outputs	Generated images standing for the text used as an input.
Error Handling	Error messages for API request errors and server errors such as a 500 error, Detailed error messages if image cannot be generated
Dependencies	FR 3.2.1, FR 3.2.5
Priority	High

FR 3.2.4	Blind Mode
Description	Provides text-to-speech functionality to aid blind users. Users will have all text, including buttons and titles, read to them. Upon load an explanation will be read to the user and a slight ticking sound that speeds up upon nearing an object / interface is used to guide users.
Inputs	Mouse position, text to be read to the user

Outputs	Audio output of all text, Audio output for guiding the user’s mouse
Error Handling	Audio based error display if a task cannot be completed, or nothing is interacted with
Dependencies	FR 3.2.1, FR 3.2.9, FR 3.2.7
Priority	High

FR 3.2.5	Customizable Reading Interface
Description	Users can adjust reading settings to improve readability. Options include changing font type, font size, background color, line spacing, and paragraph spacing.
Inputs	User Preferences that define font type, font size, background color and more.
Outputs	Customized UI and UX based on user preferences
Error Handling	Restore default settings if customization fails
Dependencies	FR 3.2.1, FR 3.2.4, FR 3.2.2, NFR 3.3.2
priority	Medium

FR 3.2.6	Book Recommendations
Description	Uses AI to analyze user reading history and recommend books. The system provides personalized recommendations for enhanced user experience.
Inputs	User reading history, User preferences
Outputs	Formatted list of recommended books
Error Handling	Display default recommendations if AI recommendations fail. Notify users of any issues with the recommendation system.
Dependencies	FR 3.2.1, FR 3.2.5, FR 3.2.2, NFR 3.3.2
priority	Medium

FR 3.2.7	User Preferences Management
Description	Allows users to save and retrieve their reading preferences, such as font size, background color, and preferred reading modes. Ensures a personalized experience for each session.
Inputs	User preferences
Outputs	Stored preferences, display of all preferences and current settings

Error Handling	Display error messages for failed storage or retrieval of preferences. Provide default settings if user-specific preferences cannot be loaded.
Dependencies	FR 3.2.1, FR 3.2.5, NFR 3.3.2
priority	Low

FR 3.2.8	Profile Management
Description	Users can manage their profile information, including updating their name, email, password, and reading preferences. Ensures user data is current and correct and user has ability to change their information
Inputs	Profile data such as username, password, email, name
Outputs	Updated or current user information
Error Handling	Display error messages for failed updates or invalid input data. Offer user guidance on profile updates.
Dependencies	FR 3.2.1, FR 3.2.4, FR 3.2.7
priority	Low

FR 3.2.9	LLM and AI Integration
Description	The use of OpenAI's API to manage all AI related requests. This includes querying DALL-E for images based on the users imported text, GPT for finding and suggesting e-books to the user, GPT for finding e-books via the new GPT-4o ability to search the web to find documents
Inputs	User selected text as prompts for image generation, text as prompts upon searching for e-books for import, text as prompts for e-book suggestions from users read history.
Outputs	Generated images, generated text to be parsed for book recommendations, generated text to for automatic book suggestions.
Error Handling	Dynamic modal to display all AI request errors
Dependencies	FR 3.2.1, FR 3.2.6, FR 3.2.3
Priority	High

3.3 Non-Functional Requirements

NFR 3.3.1	Performance

Description	The application should ensure best performance, with pages loading and responding to user actions within 2 seconds. The system should oversee simultaneous user interactions efficiently without significant delays.
Dependencies	FR 3.2.9, FR 3.2.6, FR 3.2.5, FR 3.2.2, FR 3.2.1, NFR 3.3.2, NFR, 3.3.3
Sub Requirements	Use of state management, memorization, callbacks, proper unit testing.

NFR 3.3.2	Updating Database Information
Description	The system must ensure that updates to database information (such as user profiles, preferences, reading history, account deletion or creation, and e-book data) are processed < 2 seconds of the user's request. Changes should be reflected at once to support data consistency.
Dependencies	FR 3.2.8, FR 3.2.7, FR 3.2.6, FR 3.2.5, FR 3.2.2, FR 3.2.1,
Sub Requirements	Firebase Firestore to support real time data concurrency

NFR 3.3.3	Response Time
Description	The system must have a response time of a maximum of 3 seconds for all non-AI API calls to ensure a seamless user experience. This includes fetching user data, book information, and preference settings.
Dependencies	FR 3.2.9, FR 3.2.6, FR 3.2.5, FR 3.2.2, NFR 3.3.2, NFR, 3.3.3
Sub Requirements	Unit Testing, metrics and statistics from Firebase to oversee backend performance

NFR 3.3.4	Server Errors
Description	Encountering server errors and request errors.
Dependencies	NFR 3.3.3, FR 3.2.9, FR 3.2.2, FR 3.2.5
Sub Requirements	Print statements, error catching and handling.

NFR 3.3.5	Uptime and availability
Description	Server will be supported and live 24/7
Dependencies	N/A
Sub Requirements	N/A

NFR 3.3.6	Portability

Description	Application should be able to run on a web server hosted within a Linux environment.
Dependencies	N/A
Sub Requirements	React.js, Node.js, Linux, NPM

NFR 3.3.7	Scalability
Description	It should be designed to scale to accommodate more users and the volume of data used.
Dependencies	NFR 3.3.2, FR 3.2.7
Sub Requirements	Kubernetes*, Firebase, Vercel*.

Items marked with * are subject to change.

4. Design Constraints

The application will need to be able to run on any hardware. This means that the application should work on Windows 7 operating systems and above and Mac OS 10.7 and above. Since the application is entirely web based it will be able to run on any major web browser including Chrome, Safari, Firefox, and Edge. Using solely web-based browsers the application should be easily used by anyone with internet access. There will be no minimum hardware requirements besides the ability to run a web browser. Cross platform compatibility will also be present; the application can run on both mobile browsers and desktop browsers, allowing users to use either type of platform to view the web application.

Because we are targeting a demographic of those with disabilities, we need to keep in mind the accessibility of certain features such as audio queues. These queues will need to be played at max volume which must then be changed by the user themselves or else some may not hear certain audio ques. Also, we will have to comply with the ADA act that provides certain rules to developers that allow websites to be accessible to those with hearing disabilities.

Our web application will be running on Vercels servers which run on the AWS Edge Network. There aren't any minimum requirements for this as Vercel adapts services according to the load of your app upon the server. Our Database and all backend functions are external and do not require us to host them ourselves. Maintenance and performance checks will be regularly carried out using the Vercel dashboard and CLI to keep metrics up to date. An Admin user will be created that has access to the CLI and all other external resources so the team can access them when need be.

5. Logical Database Requirements

In this section, we will specify and discuss the logical requirements for the data that will be utilized by DyslexAI database. DyslexAI will utilize the firebase platform as its database host to complement the front-end development by using Firebase JavaScript SDK.

5.1 Type of Information.

For DyslexAI to work efficiently, DyslexAI will obtain and utilize various types of data, for the purpose of personalizing the reading experience for comfortability. We achieve this by using the following type of information in our database:

Input data provided by the user (for user preferences)

- Numeric data: for storing and changing desirable text size, text spacing, paragraph spacing, line light,
- Slider Bar data: based on numerical data to change text size.
- Textual Data: purpose for storing selected font style, bolding, italic, underlining of text, and credential info. And for changing text colors.

5.2 Frequency of use

To improve users, experience in terms of conveniency and efficiency, DyslexAI will be storing important data that will be frequently accessed by the web application or the users. This will serve the purpose of editing profile details and personalization of settings to fit the user preference. Specific data that will be stored throughout out database because of frequence usages consist of:

- User Credentials
  - Signing in, editing Profile information
- User preference settings
  - Setting text size, font style, background color, etc
- Page bookmark
  - Page bookmark for continues reading
- Default settings
  - Resetting preference

5.3 Accessing capability

To make sure our web application is safe and secure, we limited the user's capabilities to access certain data. The information the users do have access to is Credential information and preference setting. We allowed access capability to these two specific types of data for the main purpose of reading, modifying and deleting data. In terms of accessing Credential information, users will have the ability to read, modify and delete the following data:

- Username
- Full name
- Password

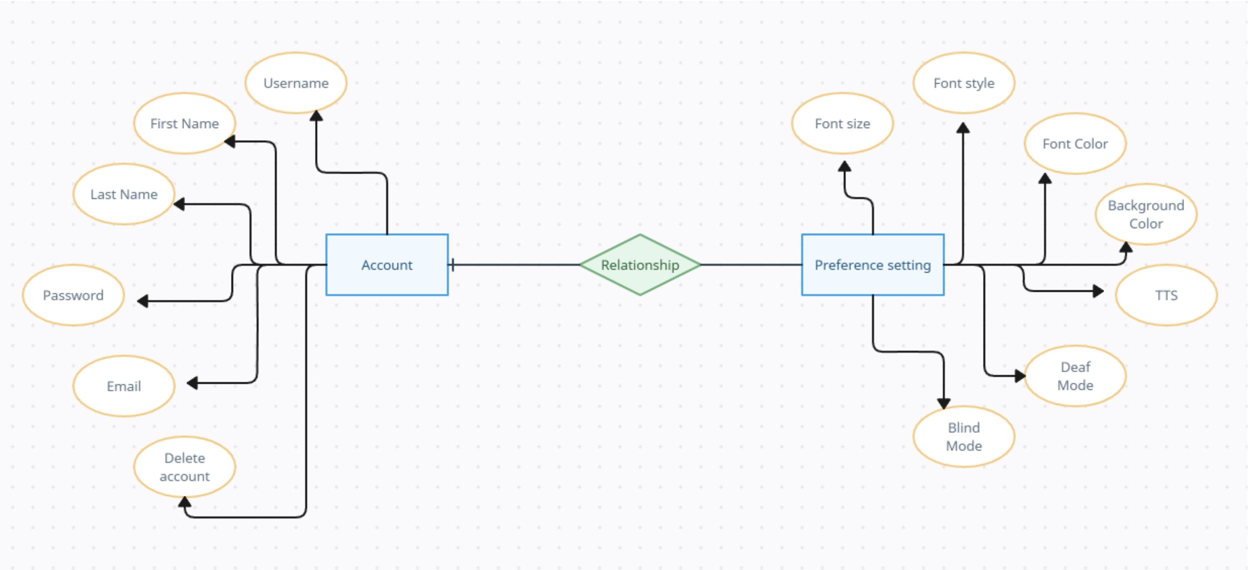
- Email
- Delete account

Preferences setting comes with more restrictions which limits the user accessibility more than credential section. In terms of accessing preferences setting, users can only read and modify data to a limited range. This consist of:

- Font size
- Font style
- Line spacing
- Background color

5.4 Data entity and their relationship

Entity Relationship Diagram



5.5 Integrity constraints

In this sub section, we will discuss and define rules to set constraints on the types of data that will be use throughout this web application. To improve the efficiency of DyslexAI by ensuring the acquired data is accurate, consistent, and valid with the constraint's rules.

Constraint for credential data:

- No empty text field for first name

- No empty text field for last name

- No empty text field for email

- No empty text field for password

- Unique values for all text fields

-NIST guild line for creating password

Constraints for preference setting data:

- Fonts size value must be between 1 – 100

- Font style options are Arial, Verdana, or Open Dyslexic

- Line spacing can be between 1.5 - 2 times the font size

- Line length can only be between 60 – 70 characters long

- Only using bold text for emphasis

- Background color can only be a solid color

5.6 Data retention requirements

DyslexAI policy for data retention periods.

Credential

Upon your request to delete your account, DyslexAI will give you 30 days to make a final decision before deleting your account parentally.

Preference setting

For consistency, DyslexAI will retain your preference setting until a modification is made to a feature, then that specific feature value will be deleted and replaced with the new value that was provided by the user. DyslexAI will only delete your preference setting if the user choice you reset the preferences features with the default setting.

6. Other Requirements

Additional requirements:

FR 3.4.8	Offline download API
Description	The ability to download pdf books with all your saved preference for offline reading.
Output	Readable pdf book
Priority	LOW

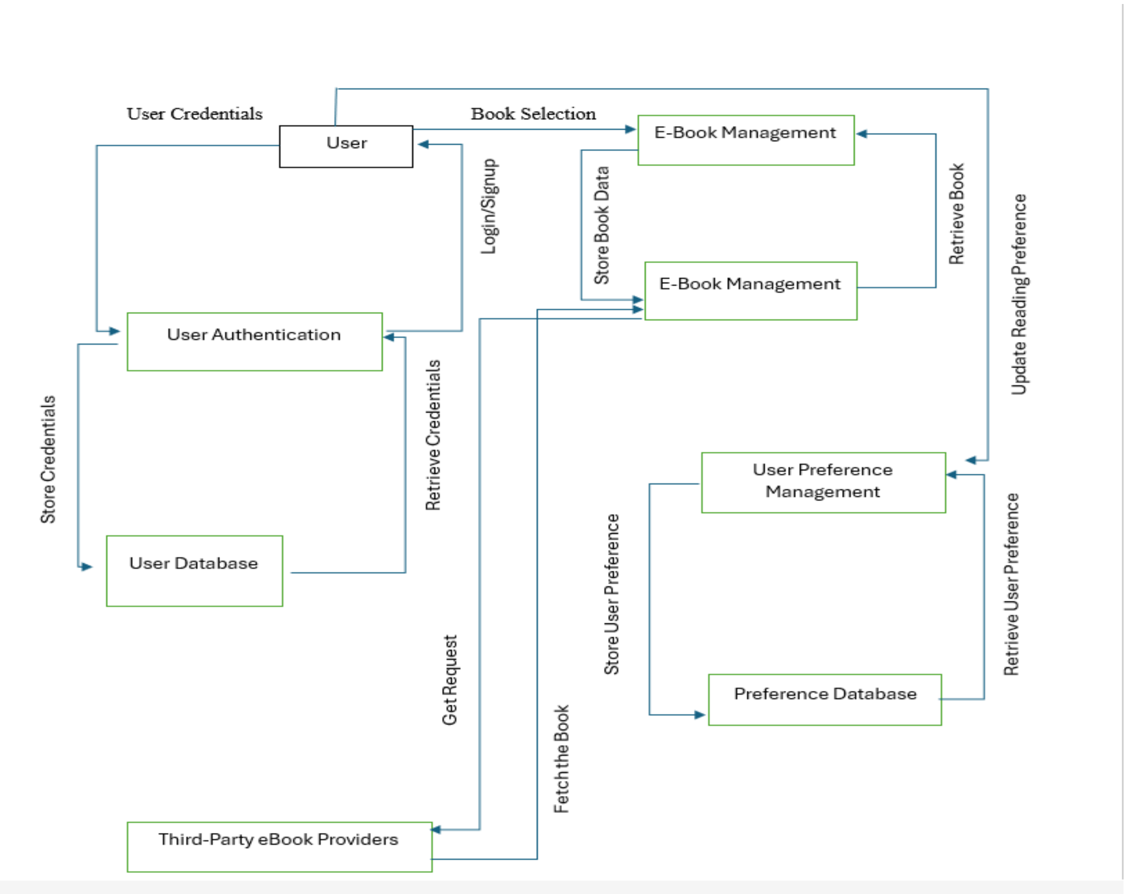
FR 3.4.9	Tailwind CSS v2.0
Description	CCS Framework that utilized the Utility-first approach for the purpose of UI web design

Requirement	Microsoft edge or newer for window users
Priority	High

7. Analysis Models

The Data flow Diagram represents the data flow within DyslexiaAI website application.

7.1 Data Flow Diagrams (DFD)



8. Appendices

8.1 Requirements Traceability Matrix

FR (ID)	NFR (ID)
3.2.1	3.3.1
3.2.2	3.3.2
3.2.3	3.3.3

3.2.4	3.3.4
3.2.5	3.3.5
3.2.6	3.3.6
3.2.7	3.3.7
3.2.8	
3.2.9	