WebToSpeech Project Requirements

Names: John Hurley, Chandler Boone, Danny Lee, Brady Byrd
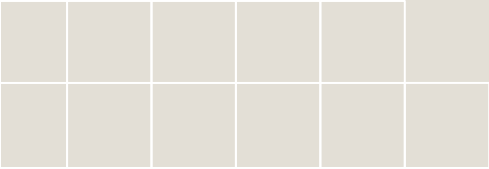
Contents

Abstract Summary

WebToSpeech is a web application created with the purpose to make a simplified and organized reading experience for a wide audience of users. Users will have the ability to upload and then organize their documents. Once they have at least 1 document, the user will be able to open that document and use any of the services provided features, such as a text-to-speech reader or any of the annotation tools.

Tools/Tech

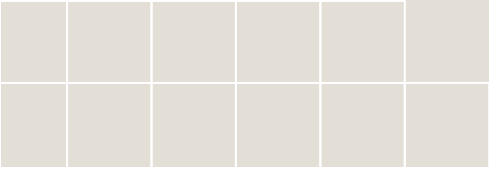| Category	Tools
Front end languages	Typescript, Javascript, HTML, and CSS
Web frameworks	React and Bootstrap
Libraries	Talkify (JS)
Server	Vercel
Database	Supabase(Postgres SQL)

Webpage Overview

The WebToSpeech website will consist of 4 pages

1. Sign-In
2. Dashboard
3. Document
4. Profile

The Sign-In page will direct to the dashboard after signing in.

The Dashboard page will contain links to both the documents and profile pages.
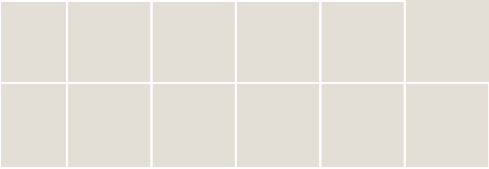
Sign-in Page

First page seen by the user

Google Sign-In button prompt

Sign-ins used for storing user collections, uploaded documents, and annotation data

Other pages can't be accessed unless signed in, redirected to sign-in page if not

Sign-in Page Requirements

Sign-in and validation handled through Google Sign-in

No account creation without a Google account

Account data stored using Supabase on a Vercel server

Profiles are created for new users in the database

Directs users to the dashboard

Load content for existing users, initialize content for new users

Dashboard Page

Toolbar

Profile: Profile Picture in top left, can "view profile" or "sign out"

Search: Searchbar in the middle, can search documents by name

Filter: Sort documents by date accessed, date added, author, or genre

Sidebar

Add collection: Create a collection by providing a name, set it as public or private

Collections: List of created collections, clicking one will populate the document selection with that collection's documents

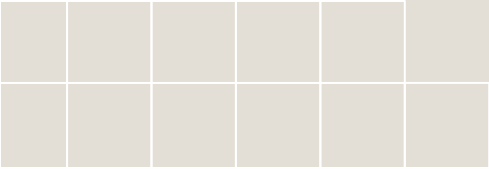Upload button: Will trigger the document selection's upload pop-up

Document selection

Display: Documents will be visible in a grid, showing their title, author, and cover photo.

Add document

Uploading: First button in the grid, shows upload pop-up. The user can either select or drag-and-drop, one file of a supported type

Dashboard Page Requirements

Profile picture and Filters show pop-up options

Searchbar limited to 255 chars, will only show documents with names containing searched terms

User must press enter or click search to trigger search

Upload file select will only support one file at a time for supported file types.

Document Page

Start/Pause and speed

A button in the bottom left will be used to start and stop the reader

Next to the start and stop button will be a slider used to control the speed of the reader.

Voice, bookmark, and font size

At the bottom of the page will be a font button used to change the size and style of the document's font.

Next to the font button will be a bookmark button that will bookmark the page the user is currently on

Next to the bookmark button will be the voice button, which allows the user to change the TTS voice.

Annotations

Highlighting the text will open a menu With annotation options

These options will include a highlighter and a read from selected area button

Document Format Requirements

Acceptable file types: TXT, PDF, EPUB, and docx.

Must be smaller than 50MB (Supabase global file size limit)

File names must contain only:

Alphanumeric: A–Z, a–z, 0–9

Punctuation: _(underscore), -(hyphen),

.(period), '(apostrophe), ,(comma)

Special characters: !, *, &, $, @, =, ', :, +, ?, (, and).
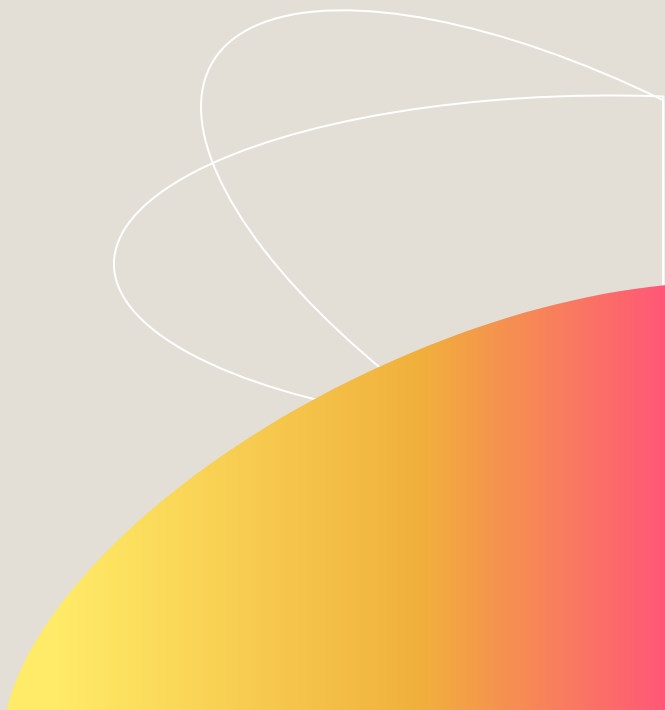
White space:

Profile Page

Profile Information

Displays Google profile picture, username, documents uploaded stat, and public collections

Profiles private by default, public visibility set manually

Settings wheel in top right of page for visibility settings

Public collections listed in a table of collection name, number of documents, and a view collection button

Profile Requirements

Information loaded from Google account

"Files Uploaded" stat loaded from Supabase database

Private profiles only visible to account owner

Private profiles will display an error message to other users and direct back to the dashboard

Only populate profile collections list with public collections
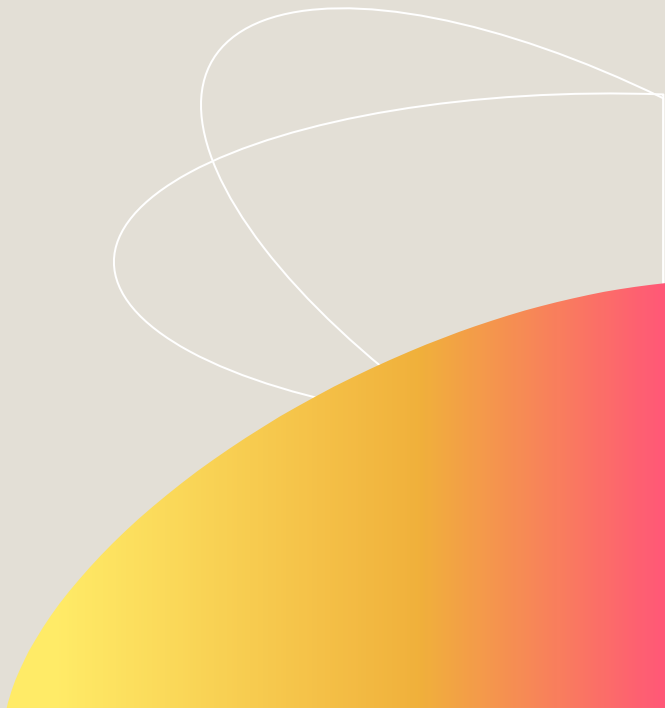
Public collections not modifiable by other users

Risky Project Requirements

TTS Integration

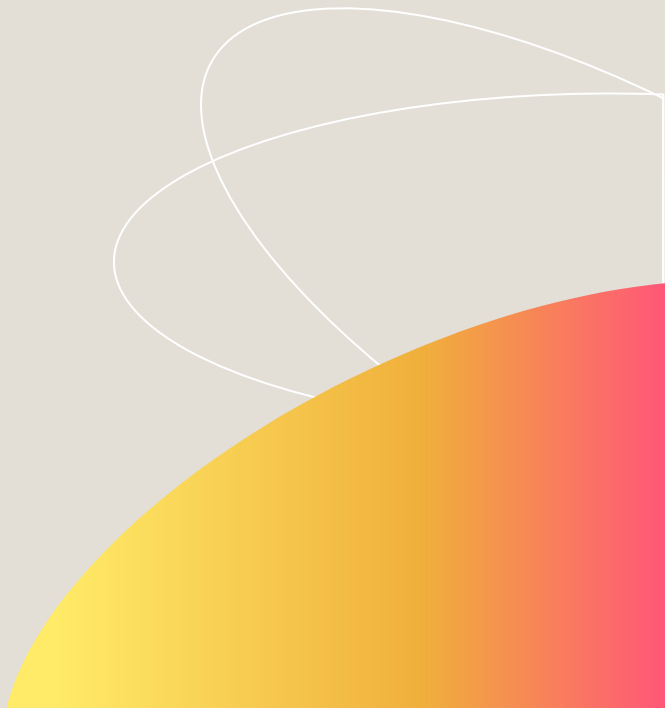**Annotations and Bookmarks**

Public Collections

TTS Integration Risks

Text needs to be read in real time to work immediately with any document the user uploads

Need to be able to correctly extract text information from multiple formats

Want to give the user multiple controls for the voice, while avoiding token restrictions on API services
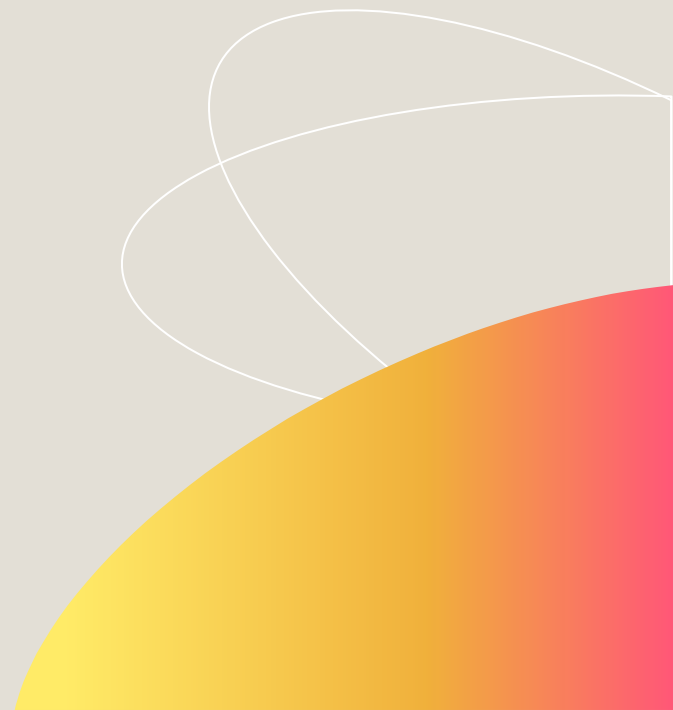
Annotation and Bookmark Risks

Highlight button will appear from selecting text, needs to be compatible on different browsers and devices without interfering with OS/Browser UI

Annotations will need to be persistent, highlights will need to be stored and loaded for the document

Bookmarks will also need to be persistent, keeping track of the page for the bookmark and moving the document to that position.

Somewhat consistent database usage to update and load both annotations and bookmarks.
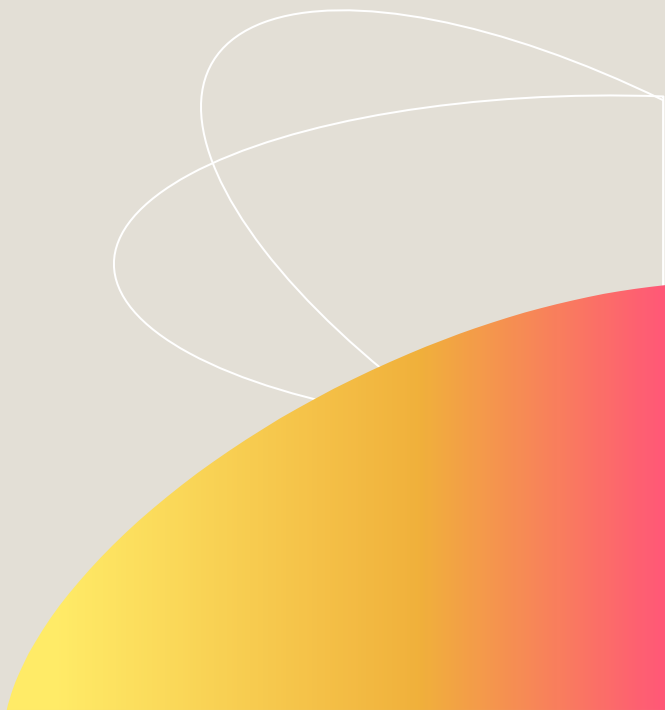
Public Collection Risks

Collections will be visible to others if the user profile and collection are both set to public

Documents will need to be visible without being able to modify the collection

Annotations and bookmark data need to be separated from the document itself, being per user.

Security concern leads to collections being private by default

That's all...

Any Questions?