

PRINCETON UNIVERSITY

**MannaBook:
Connecting Manna Christian Fellowship**

Author:
Doris LI

Advisor:
Dr. Robert DONDERO

SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
BACHELOR OF ARTS
DEPARTMENT OF COMPUTER SCIENCE
PRINCETON UNIVERSITY



May 2022

I hereby declare that I am the sole author of this thesis.

I authorize Princeton University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Doris Li

I further authorize Princeton University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Doris Li

Acknowledgements

To Dr. Robert Dondero, thank you for your guidance and support throughout the year. Although there were plenty of setbacks in the beginning and end, I am grateful for your unending patience and advice that allowed this project to propel forward. Thank you for always believing in me and challenging me to take on more ambitious content - I have learned so much. I am blessed to have received so much wisdom from you as an advisor.

To Dr. Xiaoyan Li, thank you for your time to read my thesis. I am and have been appreciative of your support since my sophomore year; I have learned countless things under your teaching, many of which have been a great influence in creating this thesis project. Thank you again for your patience and willingness to help me.

To Geoff Sackett, thank you for your unceasing encouragement. As executive director of Manna Christian Fellowship, you have been an immense help to me in designing the website and giving me insight into a user's perspective. I appreciate your constant feedback throughout the year, which has aided tremendously in this project's development.

To my friends, thank you for four memorable years at Princeton. To Six14, thank you for the laughs, cries, late night rehearsals, early morning rehearsals, and for showing me the joy of dancing. My senior year would not have been the same without this company, and the memories made with each one of you are unforgettable. To my brothers and sisters in Manna Christian Fellowship, I am so grateful that we could walk together in Christ and uplift one another in love. You were the inspiration for this project, and I am so thankful for this wonderful community. To my roommates, neighbors, and friends in Spelman 66 and 65, Grace, Monique, Joanna, Christine, Emily, Rachel, Sandy, and Irene, you have seen the craziest of me, and I will miss our obnoxious laughing, late night "studying", horror movie marathons, and verbatimz. To Yedoh, thank you for being the most responsive test user of my project, for being the best discipleship group co-leader, and for being my most honest friend. Tracie. To my thesis fairy Katie, thank you for supplying me with delicious food and a lifetime supply of Hot Cheetos - your support has brought me joy amidst many trials. To Spelman 51 boys, thank you for tolerating my late night karaoke sessions and being the source of many, many laughs.

To my family—my parents, Jasmine, and Derek—thank you for loving me and providing for me so sacrificially. Your encouragement and patience has shaped me into who I am today, and I am eternally grateful for that. I love you.

To God, thank You for everything. You are my strength and hope, and without You, I am nothing. I dedicate my thesis to You; may You be pleased with this work.

Abstract

Manna is an evangelical, inter-denominational Christian Fellowship in Princeton University that has approximately 70 students. Since there are many students involved in the organization, there is a great need for a formal and centralized platform to host information outside of sending weekly emails and Group Me reminders. This paper describes the design, implementation, and evaluation of MannaBook, a website created to house general information about Manna—from its leadership to events—and a way to connect its members to each other through adding a database of those involved in the fellowship. In this application, users can create a profile, upload a photo and any file of their choice, and share bios that can then be read by others. In addition, users will be able to listen to and filter through recorded messages from Manna’s Large Group, a weekly gathering. As an extension of the website, an experimental study was conducted to find the best semantic text matching algorithm to suggest connections between the messages and bible passages from the gospel books. When fully implemented, this feature would aim to enhance a user’s learning and engagement when listening to a recorded message. Unlike the current in-use Manna website, MannaBook provides a way for students to introduce themselves to and virtually meet others in the fellowship. Through user evaluation, I demonstrate that MannaBook successfully centralizes Manna’s information, unifies its members, and offers a new way to engage in recorded messages with bible readings.

Contents

1	Introduction	9
1.1	Motivation and Goal	9
1.2	An Overview	10
2	Related Work	13
2.1	Manna’s Current Website	13
2.2	Document Distance	13
2.3	TF-IDF	15
2.4	Word-2-Vec	16
3	Functionality	18
3.1	Homepage/Landing Page	18
3.1.1	Easily Accessible Links for Newcomers	18
3.1.2	Featured Events	18
3.1.3	Recent Sermons	19
3.1.4	Board of Trustees	20
3.2	About Us Tab	20
3.3	Leadership Tab	21
3.4	Resources Tab	21
3.4.1	Events	22
3.4.2	Sermons	22
3.4.3	Profiles	24
3.5	My Profile Page	26
3.6	Admin Interface	28
3.7	Additional Features	30
4	Implementation	31

4.1	System Architecture	31
4.2	Server and Database	32
4.3	File Storage	32
4.3.1	Amazon Simple Storage Service	33
4.3.2	Spotify Anchor	34
4.4	Front-End	35
4.4.1	Login and Logout	35
4.4.2	Contact Us and Report Profile	36
4.4.3	Profiles and Sermons	37
4.4.4	Sermon to Bible Text Matching	39
5	Evaluation	42
5.1	Methods	42
5.1.1	Experiment Design	42
5.1.2	Collecting Data	43
5.1.3	Metrics	45
5.1.4	Results	45
5.1.5	Sermon to Bible Verse Text Matching	48
6	Limitations and Future Work	49
6.1	Known Limitations	49
6.2	Future Work	50
7	Conclusion	51

List of Figures

1	Homepage of the Previous Website	11
2	Homepage of MannaBook	11
3	Words "King" and "Queen" are Closely Related	16
4	Visual Representations of CBOW and Skip-Gram [3]	17
5	Featured Events on the Homepage	19
6	Recent Sermons on the Homepage	19
7	Board of Trustees on the Homepage	20
8	Navigation Bar Dropdowns	21
9	Hyperlinked Emails in Contact Information	21
10	Interactive Google Calendar in Events Page	22
11	Button to Change Views in Sermons Page	23
12	Grid View of Sermons	23
13	List View of Sermons	24
14	Filtering Tags for All Sermons	24
15	The User Interface of a Sermon Page	25
16	The User Interface of Profiles	25
17	The User Interface of a Profile Page	26
18	The User Interface of My Profile	27
19	Blank Avatar Default for a Profile	28
20	Adding and Editing a Website Link for a Profile	29
21	Admin View of Top Bar in Sermon Page	29
22	Admin View of a Sermon Card and Table Entry	30
23	Sketch of the User Interface for Interactive Sermon Listening	30
24	A Diagram of MannaBook's System Architecture	31
25	A Screenshot of Amazon S3 Bucket Organization	35
26	Login and Logout on the Navigation Bar	36

27 The Contact Us and Report Profile Forms 37

28 An Example of a Survey Question for Sermon to Bible Text Matching 41

List of Tables

- 1 Description of the entries in the Profiles table 33
- 2 Description of the entries in the Sermon table 34

1 Introduction

1.1 Motivation and Goal

As a member of Manna Christian Fellowship since my freshman year at Princeton, I have noticed that the way information is distributed to its members is rather inefficient. The organization boasts a variety of events, including Large Group, a weekly gathering that consists of worship, prayer, and a message by one of our staff members, bible studies, prayer meetings, and more; however, the current way that these are presented and advertised to its members is, in simple terms, old-fashioned. To elaborate, the main form of communication to members is a weekly email that is only fully and regularly read by fewer than one-third of its recipients, and only recently was there a group chat created to send reminders.

To better understand the sentiment around this, I surveyed a few students in Manna and asked whether they were content with the current form of communication and what more they might want. The most common response was that there is no centralized form of information; students would have to dig through emails to find event times, locations, which is inefficient and time consuming. Also, due to the many emails we receive on a daily basis from other student groups and listservs, it is easy to miss emails from Manna containing important information.

Another common problem arises when we think about newcomers who wish to attend Manna's events or learn more about the fellowship. One must be on the email listserv and have access to the group chat to get involved, but how will a new member be able to ask for access without already knowing one of the staff or someone in the fellowship? For an organization that houses over 70 students and is steadily growing in size, it is crucial that Manna has a more efficient and coordinated way to reach students both in the group and outside.

In addition to reaching those who are not already a part of the fellowship, the increasing number of students joining the organization means that there needs to be an easier way to get to know others. As a student who has experienced Manna's community both in-person and virtually, I have noticed that the transitions from both in-person to virtual and back to in-person gatherings has made it

extremely difficult for people to meet each other organically. While being back on campus has certainly made things easier, there are people who have acclimated to the habit of staying indoors or have "lost socializing skills" over this past year.

Furthermore, the Christian fellowship actually has a current website, though, alarmingly, it is under-maintained and unused or unknown by a wide majority of students. When asked about the website, many stated that they had not known about it simply because there was no publicity for it, while others confessed that they saw no need to use it because events were not regularly updated and some of the information seemed outdated. A few even thought that the user interface was unattractive (Figure 1). When asked about the page containing recordings of past Large Group sermons, students generally stated that they rarely listened to them even if they missed Large Group that week, admitting that it was not engaging to listen to a recording on their own.

With these concerns in mind, I saw this as opportunity to create a new website for Manna that addressed the complaints that those surveyed raised. The goal of this project is to develop a centralized way to house up-to-date information about Manna and its events and also to include a way for students to connect with each other through a part of the website. A secondary goal that is explored in this paper is to better engage students as they listen to sermon recordings through using semantic text matching algorithms to suggest bible passages that relate to different parts of the sermon's transcript. This portion would be interactive and enriching, as the user will be able to control what parts of the sermon transcript will be analyzed.

1.2 An Overview

MannaBook is the new website for Princeton's Manna Christian Fellowship, with its title created by combining Manna, the name of the organization, with the popular social media platform "Facebook". Along with providing information about the fellowship, MannaBook offers a new way to meet others virtually by encouraging students to make profiles for themselves and share a bio or story, such as testimonies of how they came to faith. Every created profile is then displayed

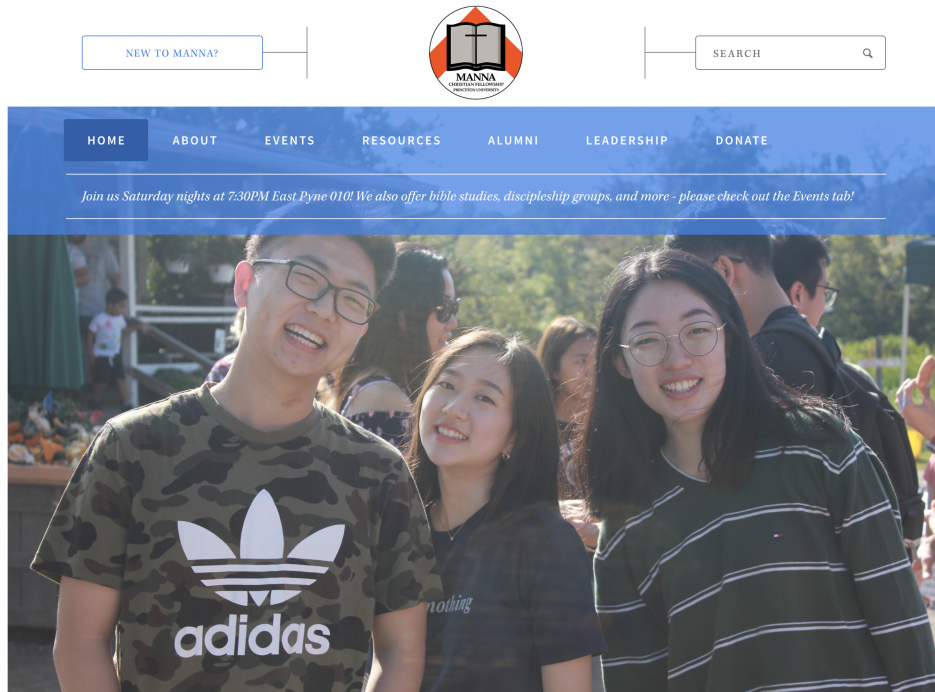
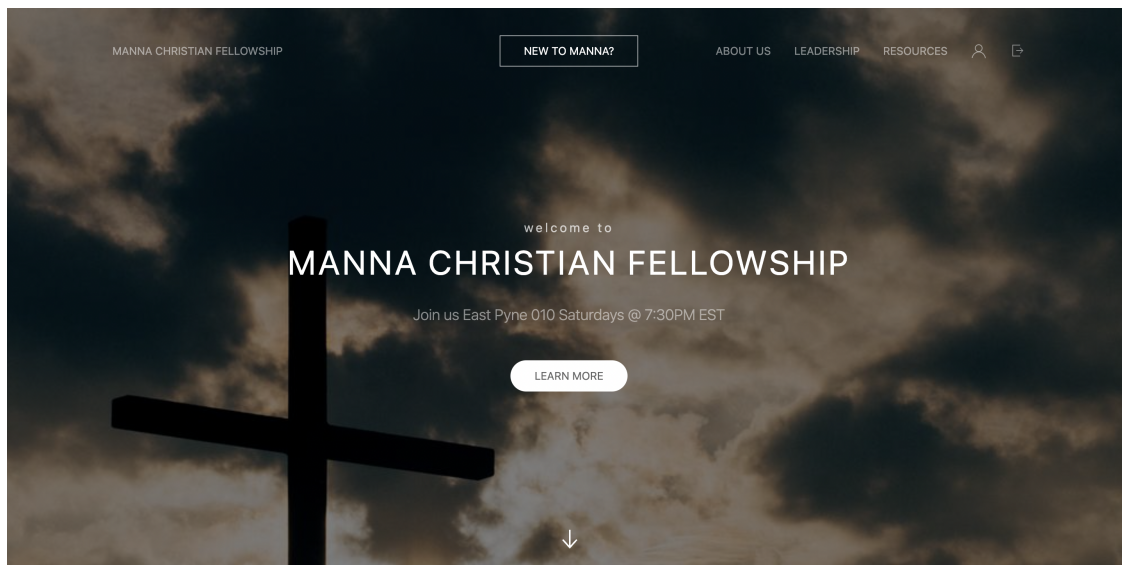


Figure 1: Homepage of the Previous Website

on a page that users can scroll, click, and read through. Figure 2 shows the user interface of MannaBook’s landing page.



WHAT DO WE DO?

[more details about events](#)

Figure 2: Homepage of MannaBook

The main features of this web application are: 1) users can view static information about Manna Christian Fellowship, including its mission, vision, core values, and leadership, 2) users can listen to recorded messages from past large group gatherings and filter based on speaker, series, and tags, 3) users can create a profile for themselves and upload a picture, file, and website links, and 4) users can view other people's profiles and click on their links and files. A feature that will be implemented after the submission of this paper will be the ability to choose a section of a sermon transcript and see bible verses in the gospels that are most similar to it.

There is an admin interface as well, which includes all the above functionalities but also allows the admin to delete any user profile and edit and upload new sermon recordings. Naturally, admin privileges are restricted to staff. Additionally, MannaBook is publicly available, but the section to view and create profiles is protected by Princeton's CAS Authentication. Thus, it is required for someone to have a Princeton login and password in order to access to the profiles of people in Manna. Along the same lines, admin will also have to authenticate in order to gain admin access.

From user evaluation, MannaBook accomplishes its goals, as user feedback has been all positive—they prefer this platform over the old Manna website, they were very likely to use MannaBook to keep track of events and updates, and they found the profiles to be helpful for getting to know new people.

2 Related Work

2.1 Manna's Current Website

There is currently a website in use for Manna Christian Fellowship that houses information about the organization [7]. Specifically, it describes Manna's mission, vision, core values, FAQ's, history, and leadership. Notably, events are discussed but only in minor detail; times and locations of each are omitted, and the events page only lists four events when in reality, there are over six official Manna events. Under the "Resources" tab, users can listen to previous Large Group recordings, with the ability to download the audio file and view other sermons by speaker and series.

Upon surveying other students about the website, I found that there are a few limitations to the current site, namely that the information is not fully updated and, as mentioned previously, the events page does not contain specific enough information. Even Large Group messages are updated slowly, which leads students to believe that the website is unreliable as a source to listen to recent sermons. Simply, there is no real incentive for students to use it, despite the website containing important information about the fellowship itself that many had not known about (e.g., history and core values). The current website is good at displaying information, but apart from that, there is no functionality that entices its users to return to the website after an initial look around.

MannaBook, on the other hand, is a website that improves upon the problems found in the existing one, while bringing more to the table. For instance, the information on MannaBook is up-to-date, events are explained in much greater detail, and it contains a more comprehensive list of events. An added section to display profiles of staff and students allows users to engage with the website and check back regularly, either to update personal information or to read about someone else.

2.2 Document Distance

As an extension to MannaBook, I am calculating similarity scores between sermons with bible passages. In the future, this will be included in a function that will supplement and enhance sermon recordings. Document similarity, as the name suggests, is a method to determine how similar two

documents are to each other [8]. In the context of this project, I calculate the document similarity between a part of a sermon recording (from Manna's Large Group) and sections of the gospels of Matthew, Luke, and Mark from the Bible to suggest additional bible readings to users who listen to the message recordings. The first method of doing so is by finding document distance, which consists of vectorizing the two text files and then calculating the cosine angle between them.

The main approach to this method is to calculate word frequency distribution of each document, allowing each text file to be represented as a vector that maps its words to the frequency count, or how often the word appears in the file [5]. The cosine angle is then calculated between the two vectors using the following equation [4]:

$$\cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|}$$

Here, A and B are the vectorized documents, and the resulting angle is the angular distance, or also document distance, which is a number between 0 and 90 degrees. A distance of 0 degrees indicates that the documents are exactly the same, and one with 90 degrees implies that they are extremely different. An example is best used to illustrate this concept. Let us begin with two sentences that we will treat as document D_1 and document D_2 :

D_1 : "Manna is a cool Christian Fellowship."

D_2 : "A lot of people in Manna are Christian."

The similar words from the two documents are: "Manna a Christian." If we take the dot product

between the two vectors, we get something like:

$$\begin{aligned}
 A \cdot B &= \text{Manna} \cdot \text{Manna} + \text{is} \cdot \text{lot} + \text{a} \cdot \text{a} + \text{cool} \cdot \text{of} + \text{Christian} \cdot \text{Christian} \\
 &\quad + \text{Fellowship} \cdot \text{people} + \text{in} \cdot 0 + \text{are} \cdot 0 \\
 &= 1 + 0 + 1 + 0 + 1 + 0 + 0 + 0 \\
 &= 3
 \end{aligned}$$

Using this, we can then calculate the cosine similarity and thus, make a judgement on the closeness of the two documents.

2.3 TF-IDF

Term frequency-inverse document frequency, or Tf-idf, is part of another method of calculating document similarity. This method uses how many times a word appears in a document and the inverse document frequency of the word across many documents [11]. In summary, Tf-idf assigns a weight to every word in a document that is increased if it has a large frequency (or abundance), but is offset by how many other documents contain that word as well. Thus, words that appear in every document many times, such as "the", will rank low because they are not particularly important to the text. On the other hand, if the word, for example, "bread" appears in one document many times but not many times in others, it is probably important to the first document and is assigned a greater weight.

The Tf-idf for each word in a document is calculated by multiplying the term frequency of that word in the text with the inverse document frequency [11]. The term frequency is simply the raw count of the appearance of that word in the document, where the inverse document frequency is found by taking the total number of documents, dividing it by the number of documents that contain that word, and taking the logarithm. If the word is relevant to the document, then the Tf-idf score will be high, and if the word is very common and appears in many documents, then the score will be closer to 0.

In more mathematical terms, the Tf-idf score for the word w in document d in a set of documents D (and number of documents N) is [11]:

$$tfidf(w, d, D) = tf(w, d) \cdot idf(w, D), \text{ where}$$

$$tf(w, d) = \log(1 + freq(w, d))$$

$$idf(w, D) = \log\left(\frac{N}{count(d \in D : t \in d)}\right)$$

2.4 Word-2-Vec

Word2Vec is a process that uses word embeddings to calculate text similarity. Word embedding is a technique used to transform words into vectors that try to capture characteristics of that word [12]. This includes semantic meanings, definitions, other words that are similar, and more. With this, Word2Vec is able to group together words of similar meaning and can create good estimates on a word's meaning based on their occurrences in the text. As an example, words like "King" and "Queen" are indeed similar, and their word embeddings reflect that numerically. In fact, the 2-dimensional embedding vector of "king" subtracted by the 2-dimensional vector for "man" and added with the 2-dimensional vector of "woman" gives a result that is close to the 2-dimensional embedding vector of the word "queen" [12], as seen in Figure 3:

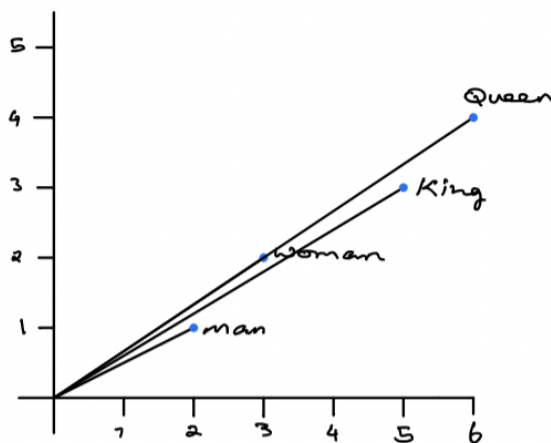


Figure 3: Words "King" and "Queen" are Closely Related

An architecture that Word2Vec relies on is the Continuous Bag of Words, or CBOW. This model tries to predict a word from a list of context words. As a simple example, take the phrase "Have a blessed day." If the model had context words ["have", "blessed", "day"], then it would try to predict the target word, "a". The left half of Figure 4 visualizes this.

Skip-gram is another popular architectural model that Word2Vec utilizes. This model predicts the probability that a word is present given an input word being present. Intuitively, Skip-gram is the opposite of CBOW, in that Skip-gram tries to take in an input word and predict its context words. The right half of Figure 4 visualizes this.

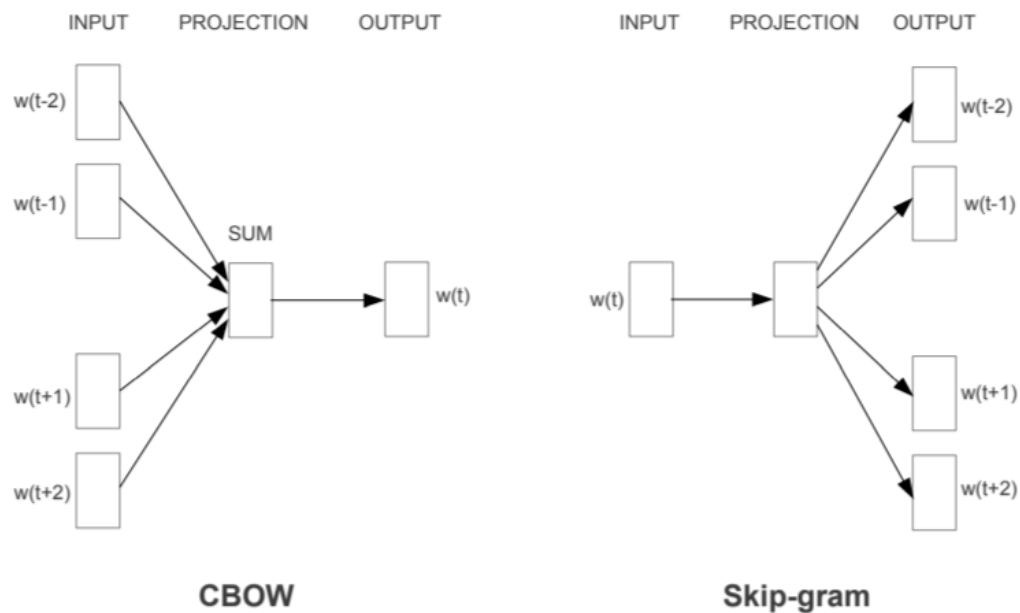


Figure 4: Visual Representations of CBOW and Skip-Gram [3]

3 Functionality

MannaBook is a website that strives to accommodate both people who are newcomers and veteran to Manna Christian Fellowship and to foster an environment that it is easier to get to know one another. To best represent this goal and understand what features should be included in the website, I surveyed students in Manna who had been active for less than a semester (still considered to be new members) and for over two years (considered to be veteran). I asked questions about the current in-use website and how connected they felt to others in the organization. From my observations and on the basis of survey results, I designed MannaBook to be organized in the manner outlined below.

3.1 Homepage/Landing Page

3.1.1 Easily Accessible Links for Newcomers

Figure 2 is a screenshot of how the website looks immediately upon loading. The top center of the navigation bar contains a square button that says "New to Manna?", which leads to the about page. This button was a feature that was well liked in the previous website (which can be seen on the top left of Figure 1) because it was an easy way for newcomers to see and click to learn about Manna. Additionally, there is another button in the middle of the landing screen that says "Learn More," which is also intended for newcomers and simply auto-scrolls the page down to the featured events.

3.1.2 Featured Events


Six of Manna's most prominent events are featured just below the initial welcome message, each of which contains a link to the official events page and a brief description (see Figure 5). Clicking on the names of each event and their respective icons will lead to the full description of the event, found in the events page. This feature was added due to popular user input of wanting to see upcoming events on the homepage.

WHAT DO WE DO?

[more details about events](#)


Large Group

One of our main events, complete with worship, teaching, and fellowship. Come learn about the book of John!




Prayer Meetings

Three times a week in order to build unity and foster fellowship while bringing our concerns before God.




Bible Study

Split by gender, these studies focus on different topics concerning the Princeton community and students.




Discipleship Groups

Grow spiritually, learn about the basics of Christianity, and work together to love and serve others on campus.



Clarifying Christianity

A weekly dinner discussion between students of various faith/non-faith backgrounds.



Unofficial Gatherings

Hang out with us at Happy Hour, play basketball with the Manna Ballers, or meet us over meals with Manna Meals!


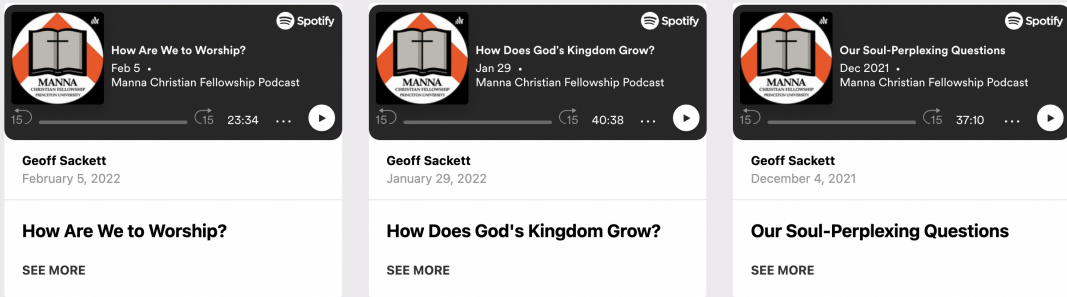


Figure 5: Featured Events on the Homepage

3.1.3 Recent Sermons

Scrolling past the featured events leads to a slider with the five most recent Large Group sermon recordings (see Figure 6). Clicking on the sermon title and "SEE MORE" will lead to a page consisting of more information about the sermon, and clicking on the speaker will lead to a page containing all the sermons by said speaker (see section 3.4.2). Users are able to play the sermon audio directly from the slider by clicking on the play button.

Listen to Recent Sermons



The slider displays three sermon recordings by Geoff Sackett from the Manna Christian Fellowship Podcast:

- How Are We to Worship?** (Feb 5, 2022, 23:34)
- How Does God's Kingdom Grow?** (Jan 29, 2022, 40:38)
- Our Soul-Perplexing Questions** (Dec 2021, 37:10)

Figure 6: Recent Sermons on the Homepage

3.1.4 Board of Trustees

An overwhelming majority of students in Manna, newcomers and veterans alike, have no idea who runs Manna behind-the-scenes. Geoff Sackett, the executive director of Manna, specifically requested that information about Manna's Board of Trustees be placed on the homepage so that users who visit the website can know who to contact with questions about the fellowship and can easily read about the leadership. A slider is located at the bottom of the homepage that contains pictures and brief contact information of the members in the Board of Trustees (see Figure 7). Clicking on someone's name, picture, or "MORE INFO" leads to a full description of them, found in the Board of Trustees page.

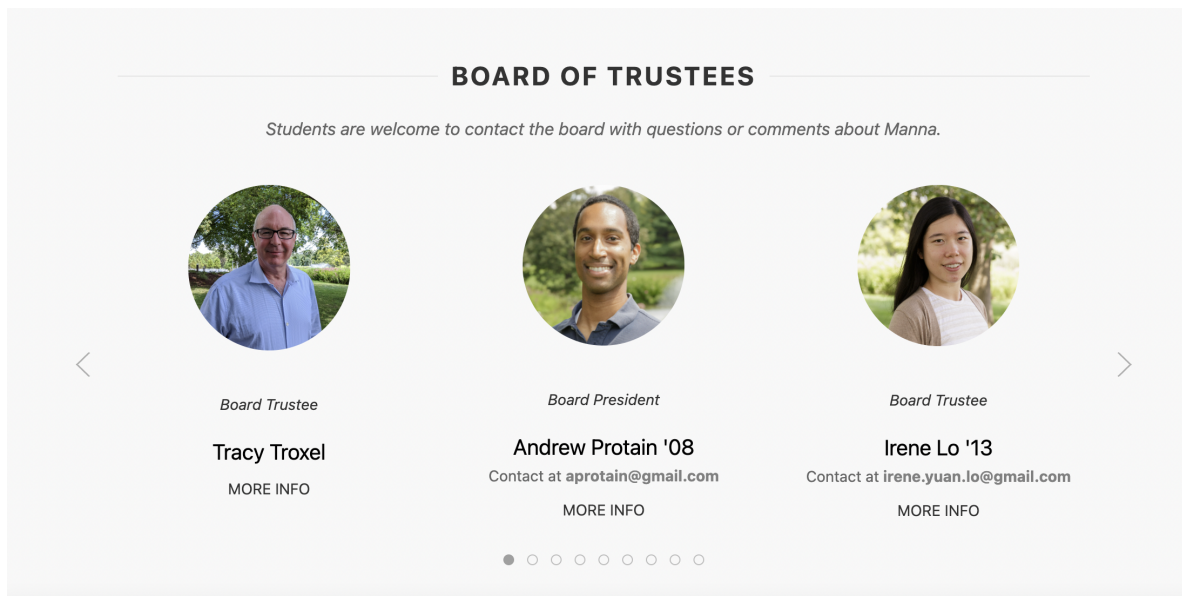


Figure 7: Board of Trustees on the Homepage

3.2 About Us Tab

The "About Us" tab on the top right of the navigation bar will lead to a page that explains what Manna Christian Fellowship is and provide shortcut links to learn about Manna's mission, vision, core values, history, FAQs, and alumni. Each of these links are static pages hosted on this website. Each of those pages are also linked in the nav-bar dropdown when one hovers their mouse over "About Us", as seen in the leftmost image in Figure 8.

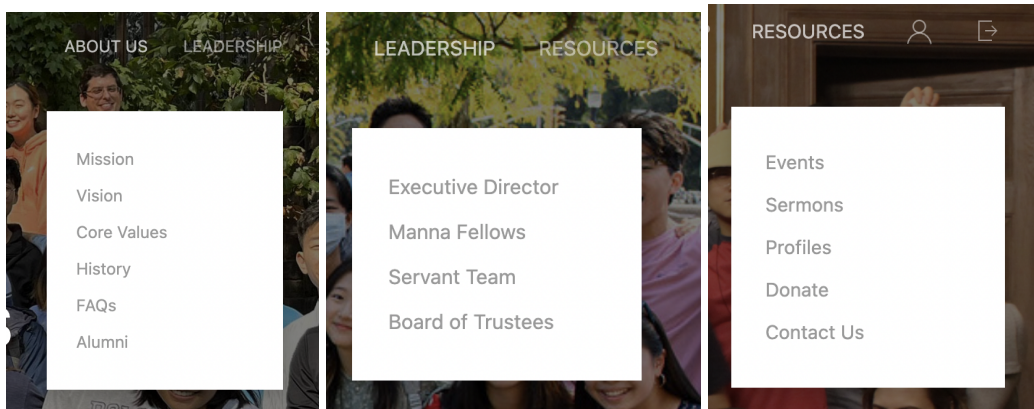


Figure 8: Navigation Bar Dropdowns

3.3 Leadership Tab

The "Leadership" tab of the website contains shortcut links to pages introducing and describing Manna's executive director, the Manna fellows, Servant Team, and the Board of Trustees. The information is also found on the nav-bar dropdown, as seen in the middle image in Figure 8. Email contact information for each of the leaders in Manna is hyperlinked for convenience and ease of access to both newcomers and veterans of Manna alike (see Figure 9).

Email: geoff.sackett@gmail.com

Figure 9: Hyperlinked Emails in Contact Information

3.4 Resources Tab

The "Resources" tab will lead to a page consisting of shortcut links to events page, sermons page, profiles page, and pages for users to donate to the fellowship and to contact staff directly through the website. Figure 8's rightmost image shows the links that can be accessed by hovering over the Resources tab.

3.4.1 Events

The events page hosts a variety of information about Manna's programming, most notably containing an interactive embedded Google Calendar of upcoming and past events (see Figure 10). Clicking each entry in the calendar will provide more information about the event, such as time and location, and users have the option to add the calendar to their own as well. Below the calendar, there is detailed information on Large Group, Bible Study Groups, Discipleship Groups, Prayer Meetings, Clarifying Christianity gatherings, Happy Hour, Manna Meals, and Manna Ballers.



Figure 10: Interactive Google Calendar in Events Page

3.4.2 Sermons

The sermons page consists of past Large Group recordings, ordered by date of the recording with the newest being on top. There are several features in this page:

- **Searching:** The search bar at the top of the page allows a user to search through the titles of sermons, speaker, series, and bible verse. The search updates with each keystroke.
- **Change Display:** Figure 11 gives the option to view the sermons in a card format (see Figure 12) or a list or table format (see Figure 13). The latter conserves space, and clicking on the "Listen" button on the right will open a modal with the recording and an option to play it.



Figure 11: Button to Change Views in Sermons Page

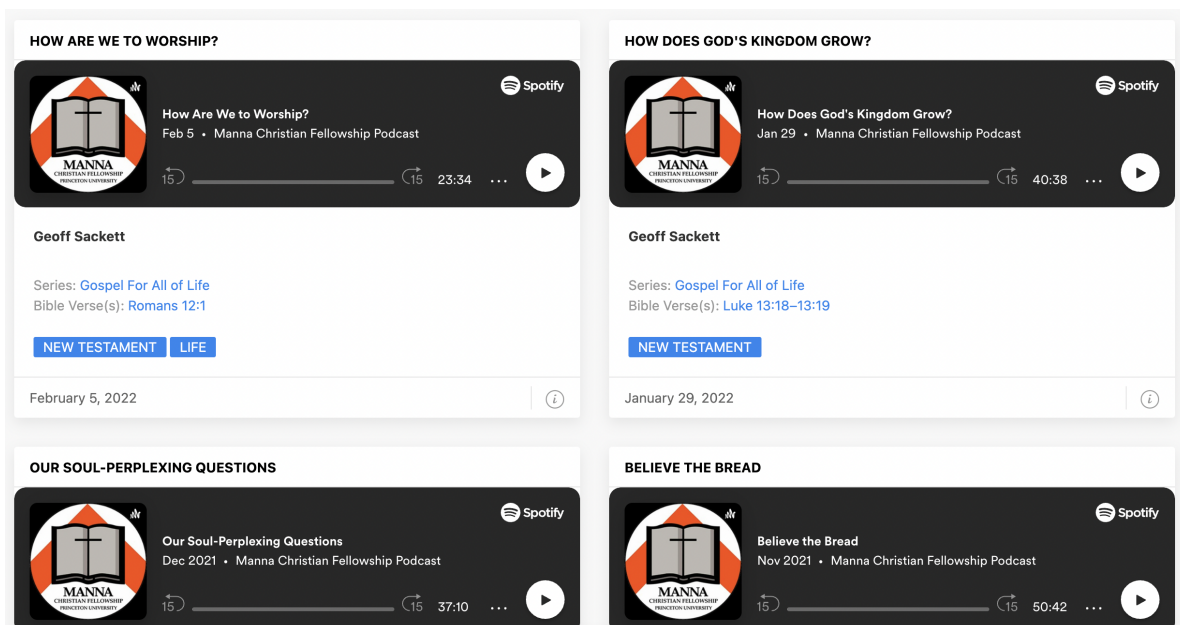


Figure 12: Grid View of Sermons

- **Filtering by Tags:** Beneath the search bar, there are buttons that filter all sermons by tags: All, Gospel, Life, New Testament, and Old Testament (see Figure 14). Each sermon card contains blue tags that correspond to these filters (see Figure 12).
- **View by Speaker:** Clicking on the speaker name in each sermon card or table entry will lead to a page containing all sermons by that specific speaker. Users are still able to search, change display, and filter through sermons in this page.

DATE	SPEAKER + TITLE	DESCRIPTION	
February 5, 2022	<i>Geoff Sackett</i> HOW ARE WE TO WORSHIP?	Series: Gospel For All of Life Bible Verse(s): Romans 12:1	<input type="button" value="LISTEN"/>
January 29, 2022	<i>Geoff Sackett</i> HOW DOES GOD'S KINGDOM GROW?	Series: Gospel For All of Life Bible Verse(s): Luke 13:18-13:19	<input type="button" value="LISTEN"/>
December 4, 2021	<i>Geoff Sackett</i> OUR SOUL-PERPLEXING QUESTIONS	Series: The Gospel of John Bible Verse(s): John 7:25-7:52	<input type="button" value="LISTEN"/>
November 6, 2021	<i>John Sherk</i> BELIEVE THE BREAD	Series: The Gospel of John Bible Verse(s): John 6:25-6:35	<input type="button" value="LISTEN"/>
October 30, 2021	<i>Geoff Sackett</i> HEALING OF THE INVALID	Series: The Gospel of John Bible Verse(s): John 5:1-5:15	<input type="button" value="LISTEN"/>

Figure 13: List View of Sermons

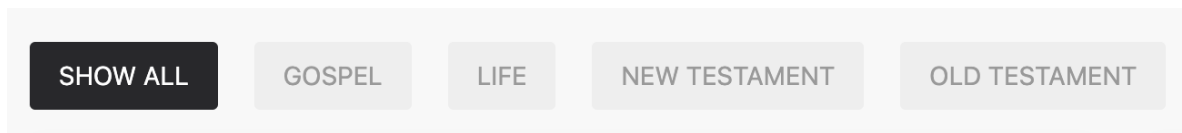


Figure 14: Filtering Tags for All Sermons

- **View by Series:** Clicking on the series name in each sermon card or table entry will lead to a page containing all sermons in that specific series. Users are still able to search, change display, and filter through sermons in this page.
- **View Message Details:** Clicking on the title of the sermon or on the information icon on the bottom right of a sermon card in grid view (see Figure 12) will lead to a page displaying all the information about that particular sermon. Figure 15 shows the user interface and layout of this page. An additional feature will be added to this page, which is described in detail in section 3.7.

3.4.3 Profiles

Created to help those in Manna virtually meet each other, the profiles page consists of all the profiles that users have made. Each profile, at minimum, contains a photo, their name, and Princeton graduation year, if applicable (see Figure 16). There are a few features on this page:

- **Searching:** The search bar at the top of the page allows a user to search through the names of

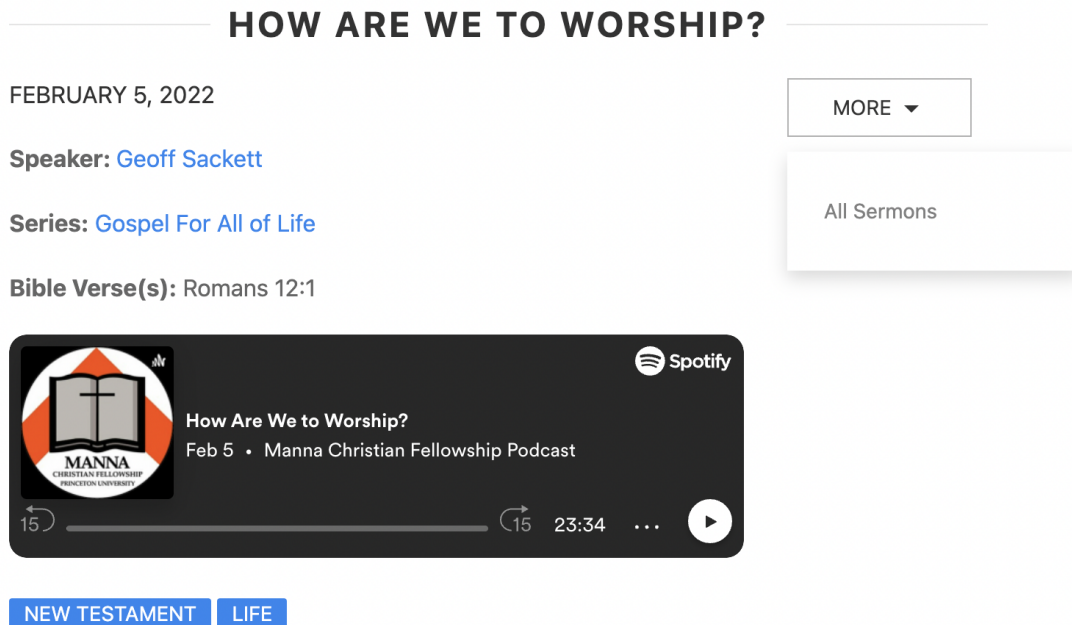


Figure 15: The User Interface of a Sermon Page

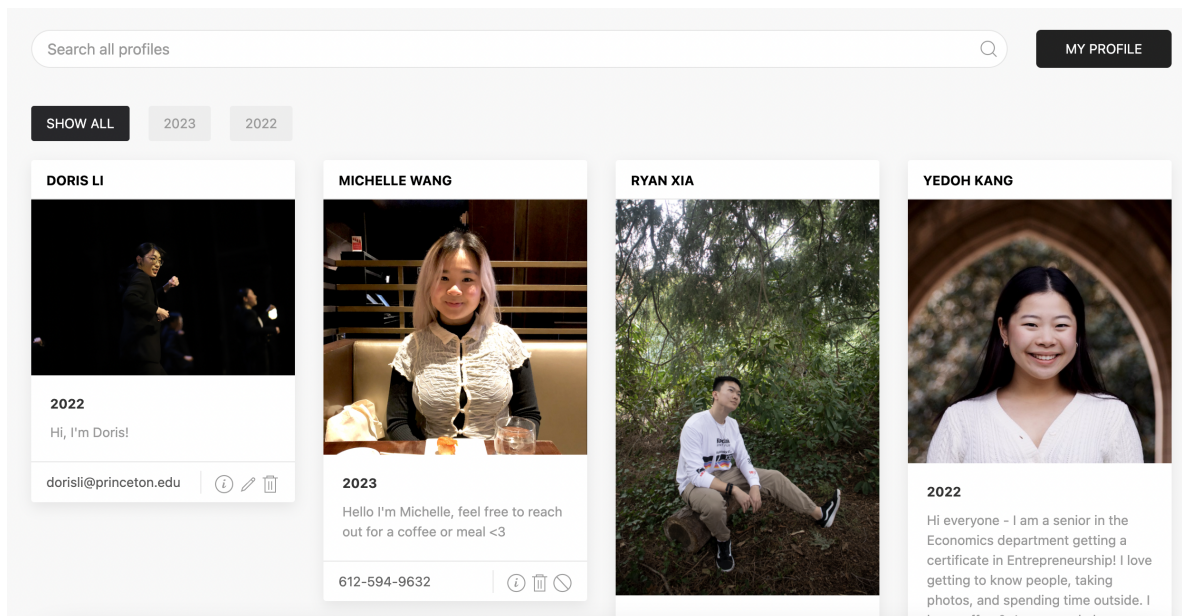


Figure 16: The User Interface of Profiles

the profiles, year of graduation, and by the text written in the card (the bio/testimony). The search updates with each keystroke.

- **Filtering by Year:** Underneath the search bar are buttons that filter profiles by graduation year (see Figure 16). Clicking them will display the profiles that match the selected year.

- **Viewing Other Profiles:** Clicking on someone's name, their picture, or the information icon on the bottom right of the card will lead to a page with more information about them. Figure 17 shows the user interface of viewing someone's profile. Links will be displayed, and clicking them will open a new tab with the attached website, document, photo, or video.



Figure 17: The User Interface of a Profile Page

- **Reporting a Profile:** Clicking on the circle with a slash icon on the bottom right of a profile (see Figure 16) or clicking on the "Report profile" in the "More" dropdown in a profile page (see Figure 17) will allow a user to report that specific profile. Upon submitting a form asking the user to explain why they would like to report a profile, administration will be notified. This feature was added in response to user concerns regarding fake or offensive profiles.
- **Going to Your Profile Page:** Clicking on "My profile" from the profiles page (see Figure 16) will allow a user to create and edit a personal profile. If you already have a profile, clicking on your name or your photo will also bring you to your profile page. See section 3.5 for more information.

3.5 My Profile Page

Clicking on the person icon in the right of the navigation bar will allow a user to create and edit a personal profile. The first time a user navigates to this page, they will be prompted to fill out a form

that asks for name, contact information, year of Princeton graduation, and a brief introduction or testimony. Upon submission, the user is led to their profile page (see Figure 18) and can then edit all this information and upload a profile photo.

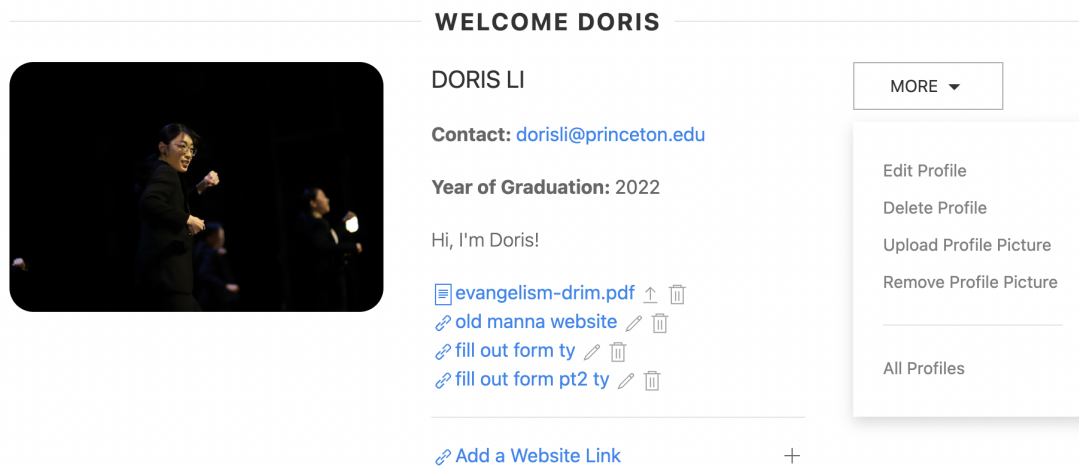


Figure 18: The User Interface of My Profile

A user can do the following on their profile page:

- **Edit Their Profile:** Clicking on the "Edit profile" link under the "More" dropdown (see Figure 18) or clicking on the pencil icon on their profile card from the "Profiles" page (see Figure 19) will lead a user to a form that, upon submission, will reflect any changes made.
- **Delete Their Profile:** Clicking on the "Delete Profile" link under the "More" dropdown (see Figure 18) or the trash can icon on their profile card (see Figure 19) will allow a user to delete their profile entirely.
- **Upload or Remove a Profile Picture:** Hovering over the profile picture in "My Profile" will give the option to upload or remove a photo, as will clicking on the respective links under the "More" dropdown. Every profile will by default, come with a photo of an empty avatar (see Figure 19). Uploading a profile picture will replace this photo, and removing a profile picture will replace the previous photo with the blank avatar.
- **Uploading and Deleting a File:** Upon creation of a profile, users will have the option to upload one supporting file. This could be a video, pdf file, word document, or more. Once

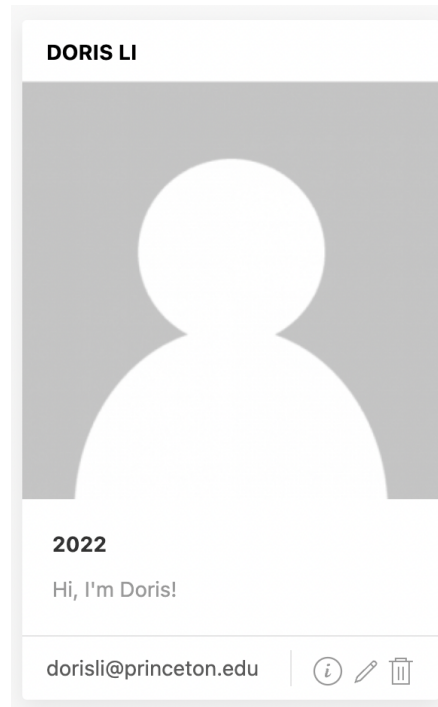


Figure 19: Blank Avatar Default for a Profile

uploaded, users can re-upload a new file by clicking the upload icon to to the right of the file name or removing it by clicking on the trash icon next to it (see Figure 18).

- **Adding, Editing, and Removing Website Links:** Upon creation of a profile, users will have the option to add multiple website links. Clicking on the "Add a Website Link" button will reveal a form that takes a link and title as inputs. Clicking on the pencil icon on the right of the name will open a modal in which users can edit and save the new information (see Figure 20), and clicking on the trash icon to the right will remove the link from the profile.

3.6 Admin Interface

There are certain roles and functions that only assigned admin of the website can perform, which are related to maintenance and security. Admin roles are granted exclusively to adult and already graduated staff.

- **Uploading a New Sermon:** Since Large Group recordings are produced every week, it is vital that a staff is able to upload them regularly. On the admin interface, there is a button next to

[Add a Website Link](#)

Website Link (req. format: https://...)

Title of Page*
(this will be the name displayed on your profile)

** If you have multiple links, please do not use the same Title as others or else previous links will be overwritten.*

SUBMIT

Edit Link Information

Website Link

Title of Page*
(this will be the name displayed on your profile)

** Please do not use the same Title as other links.*

CANCEL **SAVE**

Figure 20: Adding and Editing a Website Link for a Profile

the search bar that allows an admin to upload a new sermon (see Figure 21). This leads to a form that asks for date of recording, title, speaker, recording link, series, bible verses, and a short description.

Search all sermons

GRID LIST **+ NEW SERMON**

Figure 21: Admin View of Top Bar in Sermon Page

- **Editing Sermon Details:** Admins are able to edit sermon details after submission. Clicking on the pencil icon on the bottom right of a sermon card or right of a table entry in list view (see Figure 22) will lead an admin to a form that edits can be submitted on.
- **Deleting a Sermon:** Admins are able to delete any sermon entry by clicking on the trash icon on the bottom right of a sermon card or the trash icon on the right of a table entry (see Figure 22).
- **Deleting a Profile:** Admins are able to delete any profile that is created by clicking on the trash icon on the bottom right of a profile card (see Figure 16) or by clicking on the "Delete Profile" option under the "More" dropdown (see Figure 17). This feature was added for security and a way to address user concern about fake or offensive profiles.

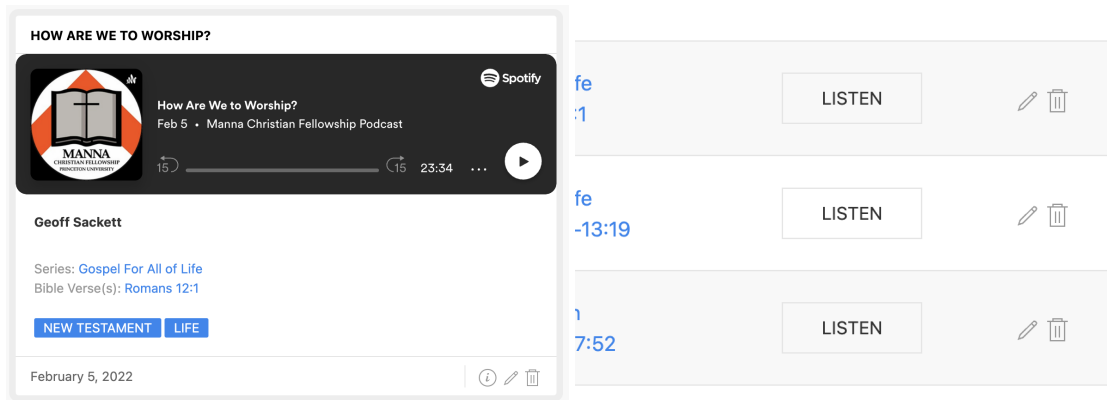


Figure 22: Admin View of a Sermon Card and Table Entry

3.7 Additional Features

Prior to the end of this semester, an additional feature will be added to enhance the sermons page. In the sermon page, seen in Figure 15, there will be an option to view the entire transcript of the sermon and interact with it. A user will be able to select a portion of the sermon with no size restrictions and view a suggested bible passage chosen from the gospels of Luke, Mark, and Matthew that is related to that selected text. This feature aims to make listening to sermons interactive and encourage more of MannaBook's users to use the website to review sermon recordings. A rough sketch of what the user interface will be is in Figure 23.

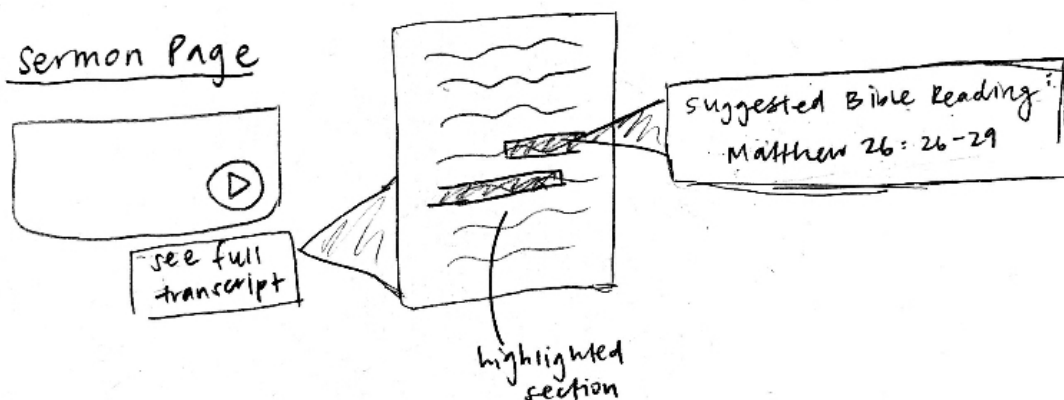


Figure 23: Sketch of the User Interface for Interactive Sermon Listening

4 Implementation

MannaBook is a three-tier application that uses Princeton’s CAS Authentication and allows users to create, edit, and delete a profile with personal information, upload photos and videos, and store website links. Admin users can upload, edit, and delete sermon details from the website, along with deleting any created profile. Users listening to sermons will be able to select a part of the transcript and see a recommended bible passage alongside it, which is will be automated and predicted by a machine learning algorithm.

4.1 System Architecture

The structure of the website is founded upon three main layers, as depicted in Figure 24. The first layer is the front-end user interface in the form of a website. The second layer is the server processing tier in which the front-end, or client-side tier, makes GET requests when the user wants to view data from the database and POST requests to the server when the user edits or adds information stored in the database. The final layer is the database and storage tier, where the database stores relational information about users, such as their profiles, information about sermons, and uploads and grabs files to and from Amazon S3.

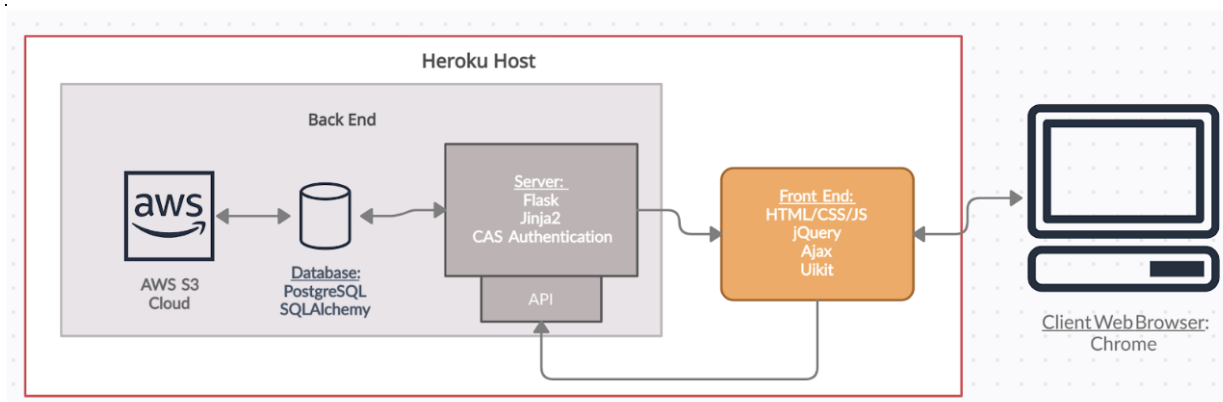


Figure 24: A Diagram of MannaBook’s System Architecture

MannaBook is deployed and hosted on HerokuApp, at <https://mannabook.herokuapp.com/>. Heroku is free and compatible with the database, PostgreSQL, making it an attractive option to

use. It is a cloud platform that contains its own Postgres database service that hosts MannaBook database tables, the Sermon and Profiles tables. To set up the system, I followed Sahil Diwan's blog [6], outlines how to create a virtual environment, setting up Postgres locally, creating a Flask application, and how to deploy to Heroku on a Mac computer.

4.2 Server and Database

Python Flask was used for creating MannaBook on the back-end server, and the client server interacts with the back-end through a series of GET and POST requests. Most of the time, these requests are handled by simple HTML, but some requests, like passing files to the server is handled by AJAX. In the back-end, Jinja templating is used to pass variables in Python to HTML files.

In order to interact with the database itself, I use SQLAlchemy, a Flask extension, in Python. This has been very useful for storing objects like dictionaries, and there are various written functions to communicate with the database, such as adding new entries into the database tables and updating certain entries with new ones.

The database is hosted on PostgreSQL and consists of two tables: Profiles and Sermons. The Profiles table keeps track of each user based off their Princeton netid, which is retrieved when logging in, and the Sermons table contains information on all the sermons based off their vidid, a counter that increments with each sermon entry. Table 1 outlines the information stored in the Profiles table, and Table 2 outlines the information stored in the Sermons table.

4.3 File Storage

Each profile has the option to upload a profile picture and additional supporting file, which are stored in Amazon S3. Sermons are uploaded to Anchor, an extension of Spotify made specifically for podcasts. The database directly interacts with Amazon S3 by uploading files through the the Amazon S3 API, where sermons are manually uploaded and stored onto Anchor.

Key	Description
netid	This is the primary key for this table. The netid is unique for all users and the login username for CAS Authentication.
firstname	The user's first name.
lastname	The user's last name.
contact	The user's contact information.
year	The user's year of graduation from Princeton, if applicable.
profpic	The Amazon S3 link to the user's profile picture.
tmony	The user's self written bio / testimony.
cont_form	The user's form of contact information. It can either be email, phone, or other.
files	The Amazon S3 link to the user's uploaded files on their profile, if applicable.
websites	A Python dictionary that maps each website link with its given name from each user's profile.

Table 1: Description of the entries in the Profiles table

4.3.1 Amazon Simple Storage Service

MannaBook uses Amazon Simple Storage Service, known as Amazon S3, to store PDF documents, images, video files, and others. Amazon S3 is a service offered by Amazon and provides file storage through a web interface [1]. For the website, I used the free tier, which allows up to 5 TB of data for a single account and used their Python API for uploading to buckets, naming, and fetching file URLs.

In Amazon S3, buckets contain objects that contain files that are uploaded. For profiles, there are two separate buckets: one for storing profile pictures (only images) and one for storing supporting files (pdf files, videos, images, etc). Each bucket consists of objects, which are named after the user's netid. Within each bucket, files uploaded by that user are present. Figure 25 shows how the objects in a bucket are organized. An example of how a photo named "profpic.png" that a user with netid "dorisli" uploaded would show up in Amazon S3: bucket "mannabook" → object "dorisli" → item "profpic.png". The link to the picture or file can be found in the item description.

Key	Description
vidid	This is the primary key for this table. The vidid is a counter and unique to all sermon entries.
date	The date of the sermon recording, in the format 'MM-DD-YYYY'.
title	The sermon's title.
speaker	The sermon's speaker.
description	A brief description of the sermon's contents.
link	The link to the Spotify entry of the recording.
linkid	The unique ID of the recording from Spotify.
fulldate	The date of the sermon recording, fully written out (e.g., December 25th, 2022).
series	The series that the sermon belongs to, if applicable.
verses	The bible verses that the sermon recording is based off.
tags	An array of tags that define the sermon (can consist of life, gospel, new testament, and/or old testament).

Table 2: Description of the entries in the Sermon table

In order for the server to upload a file and retrieve the S3 link, a function was written that accesses the S3 bucket using an access key ID and secret access key ID. Both provided by Amazon as unique account information. The function uses Boto3, the Python Software Development Kit for Amazon Web Services, to upload the file in the requisite object (directory) in the bucket, create a new object if not already existent, and return a URL for the uploaded file.

4.3.2 Spotify Anchor

Anchor is a platform by Spotify, a music streaming service, that hosts podcasts of any length for free [2]. I chose to upload sermon recordings on Anchor because of its easy-to-use interface and free unlimited episodes. It also allows the sermons to be grouped up in an official public podcast, which makes it easy for users to find sermons outside of MannaBook. Although Anchor is not directly interacting with the database, it offers an embedded feature that allows sermons to be integrated

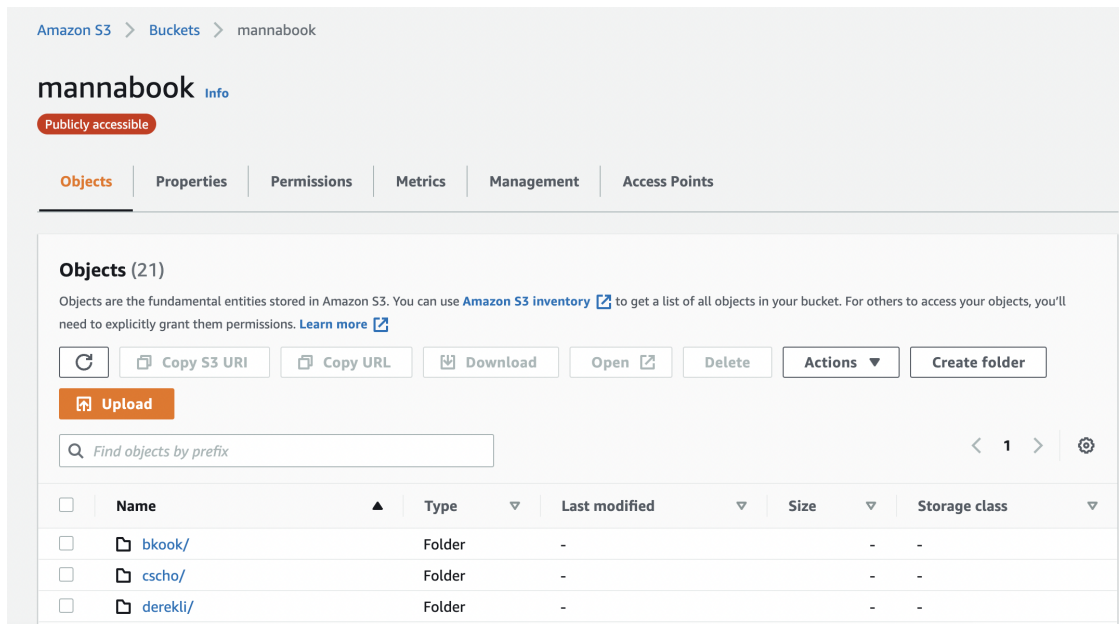


Figure 25: A Screenshot of Amazon S3 Bucket Organization

seamlessly on the front-end (see section 4.4.3).

4.4 Front-End

The front-end of the website is built using UIKit, a lightweight and modular framework written in HTML and CSS to create interfaces [10]. It provides components and classes that HTML can inherit; UIKit has been used heavily in the creation of the navigation bars, forms, buttons, modals, icons, and cards. UIKit is able to provide a clean and aesthetic user interface and is extremely easy to work with. To support functionalities found the elements that UIKit provides, such as selecting dates or uploading files in form submissions, JavaScript—in particular JQuery and AJAX—were used.

4.4.1 Login and Logout

Since MannaBook is an application made for a Princeton organization with Princeton students, the website is partially protected by Princeton CAS, or Central Authentication Service. To integrate it into the website, a Python file that initializes a `CASClient` object, written by Alex Halderman, Scott Karlin, Brian Kernighan, and Bob Dondero, is used. I use the `CASClient` object to authenticate

users (which prompts them to login through an external CAS page if not already logged in) and for the logout functionality. The `authenticate()` function is also able to return the user's netid, which is how the user is uniquely identified and how the web application knows whether the user is logged in or not.

The top right of the navigation bar shows the option to login or logout at any time while viewing the website. A local boolean variable `logged_in` checks whether that user is logged in (i.e., if the website is able to locate the netid of the user from CAS) and refreshes every time a user navigates to a new page. The `logged_in` variable is passed to the HTML of the navbar; if a user is logged in, the icon will change to a logout one, and clicking it will log that user out, and if a user is logged out, the icon will appear to be a login symbol, and clicking it will allow the user to login. Figure 26 depicts the navbar changes.

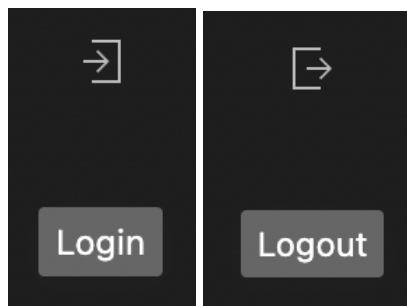


Figure 26: Login and Logout on the Navigation Bar

Since the only page that requires login is "Profiles," if a user is not logged in and visits that tab (i.e., if variable `logged_in` is `false`), a page prompting the user to log in first will appear.

Admin roles are assigned using an array of netids and allowing those specific netids to access the admin interface. Since the web application needs a username (netid) to distinguish between a normal user and an admin user, admins must be logged in to gain their additional status and roles.

4.4.2 Contact Us and Report Profile

The 'Contact Us' page under the 'Resources' tab contains the ability to send an email to Manna via a form submission. This is done by using the `smtpplib` package in Python, which defines an SMTP client session object that can be used to send mail [9]. The contact us form asks for a subject

line and email body text, but `smtplib` is able to keep the sender anonymous by logging into the Manna email account and sending an email to itself with the subject header and body text that was submitted in the form. The email is sent with the `sendmail` function.

The same mechanics are used in the report profile function, in which a user fills out a form that asks for a subject title and email body. The `smtplib` package is then used again to send Manna an email from itself. Figure 27 shows these forms.

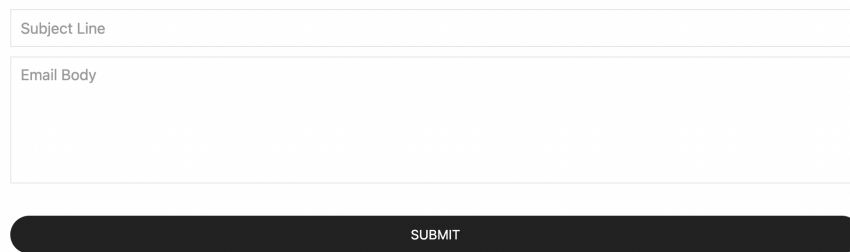
The image shows a web form with two input fields. The top field is labeled 'Subject Line' and the bottom field is labeled 'Email Body'. Below the fields is a dark, rounded rectangular button with the word 'SUBMIT' in white capital letters.

Figure 27: The Contact Us and Report Profile Forms

4.4.3 Profiles and Sermons

Viewing profiles and sermons requires GET requests to the database, which retrieve the information through Python functions that I created using SQLAlchemy. In particular, viewing all profiles invokes the function `fetchallprofiles()` that returns all the profiles stored in the Profiles table in an array. This array is then passed to the HTML using Jinja. The profile pictures are displayed using an `img` element containing the link to where it is stored in AWS S3.

Similarly, sermons are displayed using the function `fetchallsermons()` that returns all sermons stored in the Sermons table in an array. Like the profiles, this array is passed to the HTML using Jinja. On the front-end, the sermons are displayed using an `iframe` element that contains a Spotify embedded URL. In order to view sermons based off series or speaker, different Python SQLAlchemy functions are invoked to query Sermons that match a given speaker or series.

There are more functions and features shared between both profiles and sermons that are implemented in similar ways:

- **Viewing one Profile or Sermon:** Viewing a profile or sermon is when a user clicks on

someone's profile (not their own) or clicks on a sermon title or information icon that brings them to a separate page containing more information about the person or sermon. This is done through the use of the `request.args` functionality from the Flask API. The netid of the profile or vidid of the sermon is passed from the front-end to the server using `request.args` (i.e., placing the username and vidid into the url of the page after the "?"). For example, when viewing a profile of someone with netid "dorisli", the end of the url (after `.com`) to the page would be `studentprofile?username=dorisli`, with `username` storing the target profile's netid. To display information on the page, the database tables are queried by netid and vidid to return the row that matches the requested one.

- **Creating a new Profile or Sermon:** Creating a new profile or sermon is done with the use of a form and POST request to the server upon submission of the form. Python functions `addprofile()` and `addsermon()` are invoked and simply add a new entry to their respective tables by using SQLAlchemy's `session.add()`.
- **Editing a Profile or Sermon:** Editing a profile or sermon is done through the use of a form and a POST request to the server upon submission of the form. Python functions `updateprofile()` and `updatesermon()` are invoked and work by updating each column in the selected profile or sermon.
- **Deleting a new Profile or Sermon:** Deleting a profile or sermon is done through invoking Python functions `deleteprofile()` and `deletesermon()`. They delete the row in the table that matches the netid or vidid of the selected profile or sermon by using SQLAlchemy's `session.delete()`.
- **Searching:** The search functionality in the profiles and sermons pages are implemented primarily through JavaScript and UIKit's `util` API. The script enables search through UIKit's `data-tags` and `data-uk-filter-control`; the filter in the latter component searches through the words assigned to `data-tags`. The search function updates its results with every keystroke by specifying a parameter in the function.

- **Uploading Files:** Uploading a profile picture or file is done through a form and upon submission, the filename and data are parsed with JQuery. The form data is then sent to the server through an AJAX POST request and retrieved through Flask's API function `request.files.getlist()`. It is then uploaded to Amazon S3, as outlined in section 4.3.1.
- **Removing a file:** Removing a file that has been uploaded is done by erasing the appropriate column entry in the database table. To elaborate, removing a profile picture invokes Python function `updatepicture()`, which replaces the old Amazon S3 link with the link to the blank avatar photo (see section 3.5), and removing an uploaded file from a profile invokes function `updatetmonyfiles()`, which replaces the Amazon S3 link with an empty string.
- **Website Links:** Adding a website link is done through a form on the "My Profile" page that, upon submission, sends the name of the website and the link to the server as a POST request. This then invokes Python function `updatetmonywebsites()`, which adds a new key value pair to the existing dictionary from the `websites` column of the selected profile. Editing the website information is also done through a form and POST request, and invokes the same python function. The old key is saved and deleted from the dictionary, while the new key is added. Removing a website link from a profile invokes the same Python function but it simply deletes the key value pair from the dictionary. The key is passed to the server side through Flask's API function `request.args.get`.

4.4.4 Sermon to Bible Text Matching

The premise of the sermon-to-bible-text matching feature is explained in section 3.7, and there was a need to make the feature as accurate and helpful as possible. A survey was sent to determine which of the three algorithms in section 2 is the best for suggesting bible passages. The survey consisted of eight excerpts from different sermons and for each excerpt, 3 bible passages that each algorithm suggested. The first option would in each question be the passage that the document distance algorithm suggested, the second from TF-IDF, and the third from Word-2-Vec. Prior to this

survey, however, many experiments were conducted to finalize different aspects of the sermon to bible text matching feature:

- **Semantic Text Matching Algorithms:** There are many semantic text matching algorithms, but the three that were chosen were different in implementation and relatively easy to understand. I also had previous experience using a bag-of-words from a machine learning course, so I decided to also use algorithms that I was familiar with.
- **Granularity of the Sermon:** Although the granularity of the sermon is ultimately determined by the user, there were many experiments done to determine the best granularity. Sermons were tested as a full transcript, split in half, and then split into 10 minute and 1 minute intervals. When the granularity became smaller than the 10 minute mark, the algorithms consistently suggested different bible passages for different sections of the same sermon. In other words, when the sermons were split in two halves, the algorithms suggested the same bible passage for both halves; but with smaller sections, there was more variety.
- **Books of the Bible:** The books of the gospel were chosen as opposed to the other books for a few reasons: 1) comparing against all of the bible is too large, so choosing the gospels was a way to limit the scope of the matching algorithm, 2) most of the sermons already relate to the gospels, so matching sermons with parts of the gospel made sense, and 3) the gospel of John was omitted because it is stylistically and content-wise the most different from the other three.
- **Granularity of the Bible Verses:** The size of the bible was also experimented with. Initially, the bible was split into books, then chapters, and then into sections smaller than chapter. The granularity that yielded the best (most interesting and differentiating) results was the third option—splitting up the chapters by sections. Thus, in the final implementation of this feature, suggested bible readings will be in section lengths (around 3-4 verses).

The sermon excerpts found in the survey were taken from three different transcripts and were chosen because those excerpts, when run through the three algorithms, outputted the most different results. A list of about 20 sermon excerpts were tested, and I picked 8 of the most compelling and

interesting ones in terms of bible passages suggested. Many of the other sermons gave results all in the same book of the bible, and the 8 chosen were the most different from each other as well in terms of results. An example of a survey is shown below in Figure 28.

Read the following excerpt: "Two people can study God's word and dive into just responding theological words, but one to master the word and the other for the word to master him. Two people can pray with fervency, but one because they feel like a good Christian and the other because they know their soul needs that community with Him."

Which bible passage below seems the most SIMILAR to the excerpt above? *

"As for what was sown on rocky ground, this is the one who hears the word and immediately receives it with joy, 21 yet he has no root in himself, but endures for a while, and when tribulation or persecution arises on account of the word, immediately he falls away.[b] 22 As for what was sown among thorns, this is the one who hears the word, but the cares of the world and the deceitfulness of riches choke the word, and it proves unfruitful. 23 As for what was sown on good soil, this is the one who hears the word and understands it. He indeed bears fruit and yields, in one case a hundredfold, in another sixty, and in another thirty.'" - Matthew 13:20-23

"And these are the ones sown on rocky ground: the ones who, when they hear the word, immediately receive it with joy. 17 And they have no root in themselves, but endure for a while; then, when tribulation or persecution arises on account of the word, immediately they fall away.[a] 18 And others are the ones sown among thorns. They are those who hear the word, 19 but the cares of the world and the deceitfulness of riches and the desires for other things enter in and choke the word, and it proves unfruitful. 20 But those that were sown on the good soil are the ones who hear the word and accept it and bear fruit, thirtyfold and sixtyfold and a hundredfold.'" - Mark 4:16-20

"And he answered them, "Go and tell John what you have seen and heard: the blind receive their sight, the lame walk, lepers[e] are cleansed, and the deaf hear, the dead are raised up, the poor have good news preached to them. 23 And blessed is the one who is not offended by me.'" - Luke 7:22-23

I don't know.

Figure 28: An Example of a Survey Question for Sermon to Bible Text Matching

The results of the survey indicated that all three matching algorithms were effective, as there was no overwhelming majority vote for one of them. However, the algorithm that came out with the most votes was the second algorithm, or the TF-IDF. A close second was the first algorithm, the document distance algorithm, and the last place algorithm was the Word-2-Vec one.

5 Evaluation

5.1 Methods

The goal of this evaluation was to test the effectiveness of MannaBook as a platform to better house information about Manna and its events, connect new and veteran students, and engage students in sermon listening. Some thoughts that I kept in mind were: would a newcomer to Manna be able to get plugged into the community and events easily? Would using the profiles section encourage people to meet others in the organization? Will seeing suggested bible readings to accompany sermon listening be useful to the user?

5.1.1 Experiment Design

I carried out the evaluation in two different ways: in-person interviews and through a survey on Google Forms sent out through email. I decided on this method because in-person feedback is the most genuine, and as the creator of the application, I find it valuable to be able to see how the user navigates through the website and to identify any frustrations or likes. However, this method is time consuming, and I could only conduct five in-person interviews considering the one week I had to perform these interviews. Many people in Manna also expressed their desire to evaluate the website, so I decided that sending out an additional survey would be a good way to extend my reach to all the students in the fellowship and to get some more feedback.

The five people whom I interviewed were chosen deliberately, and the qualities below were the metrics I used to choose those people:

- **Level of Leadership:** I interviewed Geoff Sackett, the executive director of Manna and who holds the highest position. Other interviewees included students involved in Servant Team, a group of student leaders, Projects Team, a group of students who serve under those in Servant Team, and students not involved in leadership.
- **Years of Involvement:** I interviewed students who had been active in Manna for different

years; naturally these would be people of different grade levels who had joined Manna at different times.

- **Activeness in the Fellowship:** To separate this criteria from the first one, this metric was only considered among people who are not a part of leadership. Of the students who are not in leadership, one was very active, and one would come to events around once a month.

As implied earlier, the target audience for the evaluation was strictly people who had some connection to Manna, regardless of involvement level. Thus, the survey, was sent out to the Manna Unofficial Listserv, which is the listserv that non-staff members use to contact the students in the organization at large. There are hundreds of students on the listserv, some of which are alumni.

5.1.2 Collecting Data

For in person interviews, I gave the people testing my website a list of general things to do along with asking some questions at the end. I intentionally left some of the tasks vague to gauge how user-friendly the website was and to recreate how someone new would navigate through the pages. The following is a summarized list of tasks and questions asked:

- Read more about what Manna Christian Fellowship is
- Learn about their events and details
- Read about who manages and leads the organization
- Make a profile for yourself
- Search through and read other profiles
- Search through and listen to sermons
- If you were a newcomer, how easy would it be to navigate the website?
- How do you think making and reading profiles will affect relationships between you and others in the fellowship?
- How useful did you find <feature>?

- Would you visit the website again?

For the survey, I wanted to gauge how familiar students in the fellowship were with the current website and see how people usually met others, how they received event details, and how much information in general they knew about the organization (such as its history, core values, etc) and its leadership. Due to this, I asked questions both before letting the users look at the actual website and after they used it to evaluate whether the website made a difference in sentiment across a variety of categories. Below is a list of questions that were asked before giving users the website link:

- Do you use the current website? Why or why not?
- How do you get Manna event updates?
- Briefly describe two of Manna's events.
- Do you know Manna's history, core values, vision, etc.?
- Do you know who serves in leadership? Name a few of them.
- Do you feel connected to your peers in the fellowship?
- How likely are you to listen to a recording of Large Group you missed?

The questions that I asked after the users used the website were, in part, purposely written in a way to help users navigate through all the pages. Additionally, they were also aimed at gathering usefulness of the features in the website. Some of these questions were:

- How much did you learn about Manna (in general, events, and leadership)?
- Is the Manna website (specifically the calendar) a more convenient place to get information about events?
- How likely would you use the website to contact staff?
- If you saw a profile from someone you had never talked to, how likely would you be to talk to them after reading their profile?
- How likely are you to listen to sermons by the same speaker and/or series?
- How likely are you now to listen to Large Group recordings that you missed?

- Would you visit the website again?

5.1.3 Metrics

The metrics for success that I evaluated MannaBook with were found in the user feedback from both in-person interviews and the online survey. During the interviews, since I was looking for ease of use and practicality, I took note of body language and also whether they knew which buttons to press when given vague instructions. Their answers to questions asked after were also used for success metrics. Since it is more difficult to measure sentiment and raw reactions through a screen, I decided that the use of the likert-scale was best for gauging how much a user thought a particular feature was useful or how likely they were to use it in the future. By also comparing user responses prior to using the platform and after, I was able to determine how much the user benefited from using MannaBook.

5.1.4 Results

In total, I interviewed 4 students and 1 staff member (the executive director Geoff) in-person and got 10 additional responses from the online survey. Since students and staff play different roles in the organization and would use the website differently, Geoff's evaluation will be discussed separately from the rest. Beginning with the group of students, their responses to the questions listed in the beginning of section 5.1.2 can be summarized in the following sentiments:

- Finding static information was relatively easy and simple; however, some of them found it initially difficult to find how to view all profiles, since it is under the "Resources" tab. They suggested making it its own tab, since the Profiles page is of high importance to the overall website.
- The students who were newer to Manna thought that the website was a great help to getting to know about events and leadership. They had all stated that they learned something new and some even learned about a new event entirely.

- Three of the five students said that they would use the website and its embedded calendar to check events instead of Manna's weekly emails.
- The newer students who were not as involved in Manna believed that using the profiles feature would greatly help them meet new people and thought that it would equally be good as a way to introduce themselves to others.
- The more veteran and involved students stated that although they already felt connected to most students in Manna, the profiles section would still help them get to know people who don't come to events as often and to meet newcomers as well.
- Overall, all of the students believed that the profiles section would be useful, regardless of how much a person decided to share on their profile. Simply a name on a profile would be enough to start a conversation with the other person.
- Everyone said that they would visit the website again, either to check profiles, to listen to sermons, or read about Manna.
- Everyone thought that the user interface looked nicer than the old website and that it was also one of the incentives to visit the website more often.

Geoff, likewise, also thought that the website would be helpful to connect students together and encourage conversations. As a part of administration, however, his main desire for the website would be for it to accurately portray Manna's information, to make a welcoming environment for newcomers, and to centralize and make leadership contact information easily accessible to users. He was generally impressed with the user interface and was satisfied with the information posted and the way it was presented. He liked how the Manna Board of Trustees was featured on the front page and very visible to users, and also liked how the emails were hyperlinked for easy access. One change he would have wanted to see was to have upcoming events on the homepage so that it would be easily seen by users, especially newcomers. Overall, he was impressed with the new website and has hope that it will help to unite students of all grades and levels of involvement.

Of the participants in the survey, only one of the ten was a newcomer and one other was an

inactive member. Regardless, many of their responses to the survey were similar to those of the more active and veteran students:

- As expected, the veteran members did not learn much about Manna's events, but the newer and inactive members both indicated that they had learned something new, marking a 4 and 5 on a 1-to-5 scale, with a 5 being that everything they had read was new.
- Regardless of involvement level, the average response of how much they had learned about Manna's history, core values, and vision was 4 out of 5, with 5 being everything they read was new information.
- All of the surveyed participants indicated that the Google calendar was a more convenient way of learning of and viewing event details. Similarly, over half of the students indicated that it would be likely that they would use the calendar over reading weekly emails, choosing 4 on a scale of 1-to-5 (5 being exclusively using the calendar to get event details).
- Nine of the ten students indicated that they learned new information about Manna's Board of Trustees and other leadership, and all indicated that their contact information was more easily accessible. These were goals that Geoff had wanted.
- Regarding the profiles, the newer members weren't as comfortable sharing personal information, but the veteran members seemed more open to doing so. Despite this, the newcomers all made a profile, just with some omitted information (mainly omitted testimonies).
- All of the students surveyed indicated that they had learned something new about someone else through the profiles section and genuinely believed that the section would be useful for getting to know others in Manna.
- On a scale of 1-to-5, all participants indicated a number above 3 when asked how likely they were to talk to someone new after reading about them on the profiles page, with 5 being very likely.

5.1.5 Sermon to Bible Verse Text Matching

Since the sermon to bible verse text matching hasn't been implemented yet, there is currently no evaluation available. However, once coded into the website and if time permits, the evaluation for this part would be centered around the accuracy of the suggestions as well as how useful it is for each user. To gauge interest in the feature, I had asked in the survey: "If there was a way to see recommend bible verses most similar to certain sections of each recorded sermon, would it be helpful?" 8 of the students responded with yes, and 2 answered that it would be interesting but they were unsure about the usefulness. With this, it was reasonable to suspect that users would be interested in this feature.

6 Limitations and Future Work

6.1 Known Limitations

One limitation to MannaBook is that it is partially protected by Princeton's CAS Authentication. While CAS Authentication definitely holds its benefits in the application, it blocks anyone who does not have a working Princeton login from accessing the profiles page. If the website were limited to strictly current Princeton students and staff, this would not be an issue, but a secondary goal of the Profiles page is to be able to connect students with alumni of Manna. Unfortunately, alumni are not granted access to their Princeton logins after graduation. Likewise, if a current student creates a profile and then graduates, their profile will remain on the website while that student eventually loses access to it. In short, they would have no way of editing or deleting their profile.

Another limitation regarding the profiles section is that the website does not have a way to check the real identity of a user. Currently, the report profile option is the only method of defense against fake or offensive profiles, but it requires administration intervention and other students to catch the fake profile first. Since a Princeton netid is linked to a student's private information, if MannaBook were able to retrieve information like names, it would potentially make the Profiles section more secure and genuine.

Additionally, when uploading a profile picture, not all image file extensions are accepted. As of now, the extensions `.HEIC` and `.CR2` are unable to be uploaded, as they are not supported by the browser and upload button used. `.HEIC` files are Apple's version of a High-Efficiency Image File and are photos taken by an iPhone, and a `.CR2` file is a Canon Raw Version 2 image file, meaning that it was taken on and transferred from a Cannon digital camera. A user must first convert those image types into a more standard image extension type, such as `.jpg`, `.jpeg`, or `.png`, before uploading.

Lastly, all static information (i.e., not profile and sermon information) is currently housed in HTML files. This means that any changes to the static information, such as updates to leadership, must be changed within the file and then re-deployed to Heroku. This makes it difficult for future

maintenance, since either only I can update information or I would have to teach someone how to navigate through the GitHub files and deploy to Heroku, which might involve providing my personal Heroku account login information.

6.2 Future Work

To combat the first limitation of restricting non-Princeton users and alumni, the login function could be changed to a Google login that uses OAuth API to retrieve information. Although this method also fails to verify user identity, this would allow for alumni to participate in making profiles and increase connections for current students. To combat the last limitation, all static information could be moved to a Postgres database instead of stored in HTML files. This way, admins would be able to update information on the website front-end itself rather than having to change the back-end files. Due to this, the admin interface would have to be much more complex and would most likely require a new interface separate from the website.

As explained in many of the sections prior to this one, a future feature that will be implemented is the ability for users to see recommended bible passages that correspond to a certain part of a sermon. As how Figure 23 and section 3.7 outline, this feature would allow users to interact with the sermons themselves instead of just simply listening to them passively. The purpose of this feature is to encourage more users to: 1) listen to Large Group messages that they missed, 2) periodically review past sermons, and 3) enhance sermon listening with bible readings.

7 Conclusion

Overall, MannaBook was a success, as an overwhelming majority of students were more likely to use it over the old website. They also felt like the Profiles section of the application was a great way to meet others and also introduce themselves to the Manna community. The user interface is both aesthetically pleasing and easy to use, and while simple, there are still many features that a user can explore. An these features entice users to return back to the website even after an initial look-through.

While this project was created to help Manna Christian Fellowship grow closer as an organization of students, I was able to learn many new skills from the development of this website. Although I had taken a course on Python web-development before, this was the first time I had created and maintained a Postgres database and deployed an application to HerokuApp on my own. As much as the back-end development was new to me, the front-end design was also just as challenging. Learning to implement functions in JavaScript (JQuery and AJAX) involved many hours of trial and error and debugging, all of which will be immensely useful experience for future projects. This application is also heavily user-based, so learning how to take constructive criticism and realistically balance others' suggestions with practicality and functionality were skills that I sharpened along the way.

While there are aspects of the user interface to be improved and features to be implemented, it is very likely that MannaBook will replace the current website as the official Manna website. From my evaluation interview with Geoff, the executive director, it seems that he is very interested in pushing the project forward, which includes rerouting the current URL, <https://manna.princeton.edu/>, to MannaBook. A platform for both newcomers and active students, MannaBook is a step forward in connecting more of the fellowship together.

References

- [1] “Amazon s3.” [Online]. Available: <https://aws.amazon.com/s3/>
- [2] “Anchor by spotify.” [Online]. Available: <https://anchor.fm/features>
- [3] “Calculating document similarities using bert, word2vec, and other models.” [Online]. Available: <https://towardsdatascience.com/calculating-document-similarities-using-bert-and-other-models-b2c1a29c9630>
- [4] “Cosine similarity: How does it measure the similarity, maths behind and usage in python.” [Online]. Available: <https://towardsdatascience.com/cosine-similarity-how-does-it-measure-the-similarity-maths-behind-and-usage-in-python-50ad30aad7db>
- [5] “Document distance problem definition.” [Online]. Available: http://6.006.scripts.mit.edu/~6.006/spring08/wiki/index.php?title=Document_Distance_Problem_Definition
- [6] “Making a flask app using a postgresql database and deploying to heroku.” [Online]. Available: <http://blog.sahildiwan.com/posts/flask-and-postgresql-app-deployed-on-heroku/>
- [7] “Manna christian fellowship.” [Online]. Available: <https://manna.princeton.edu/>
- [8] “Measuring the document similarity in python.” [Online]. Available: <https://www.geeksforgeeks.org/measuring-the-document-similarity-in-python/>
- [9] “smtplib — smtp protocol client.” [Online]. Available: <https://docs.python.org/3/library/smtplib.html>
- [10] “Uikit.” [Online]. Available: <https://getuikit.com/docs/introduction>
- [11] “Understanding tf-idf: A simple introduction.” [Online]. Available: <https://monkeylearn.com/blog/what-is-tf-idf/>
- [12] “Word2vec explained.” [Online]. Available: <https://towardsdatascience.com/word2vec-explained-49c52b4ccb71>