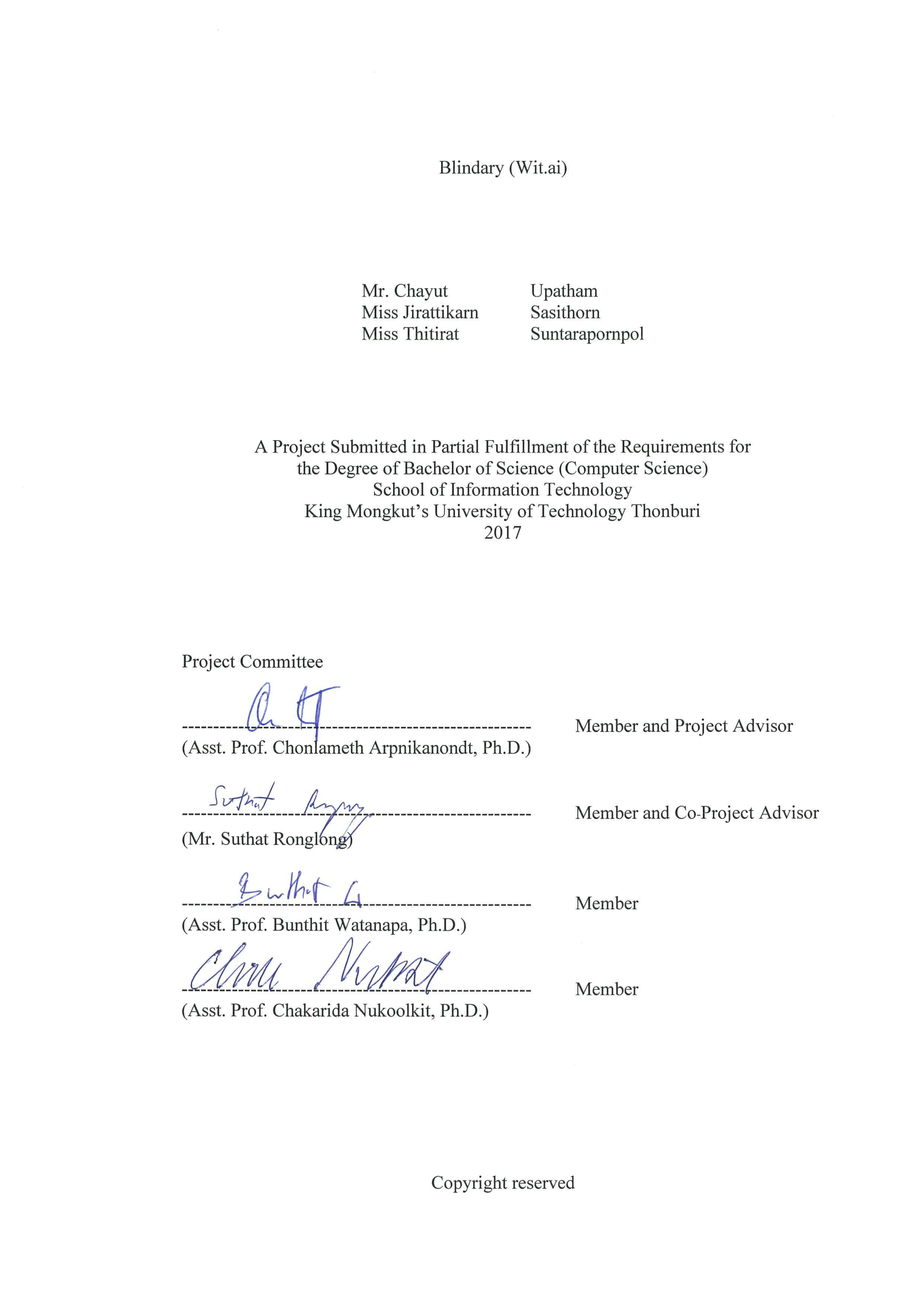
BLINDARY (WIT.AI)

MISTER CHAYUT UPATHAM

MISS JIRATTIKARN SASITHORN

MISS THITIRAT SUNTARAPORNPOL

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR  
THE BACHELOR OF SCIENCE IN COMPUTER SCIENCE  
SCHOOL OF INFORMATION TECHNOLOGY   
KING MONGKUT’S UNIVERSITY OF TECHNOLOGY THONBURI  
2017



Project title Blindary (Wit.ai)

Project Credit 3

Candidates Mister Chayut Upatham

Miss Jirattikarn Sasithorn

Miss Thitirat Suntarapornpol

Advisor Asst. Prof. Chonlameth Arpnikanondt, Ph. D

Program Bachelor of Science

Field of Study Computer Science (English Program)

Faculty School of Information Technology

Academic Year 2017

**Abstract**

Before the Internet became so common, reading a book was the best way to gain knowledge and information. However, it has always been hard for the blind to find braille books to read because they are very expensive and most are in English. However nowadays, people can access content on the Internet easily by using a smartphone or laptop. So, we decided to do the Blindary project. The aim was to provide more alternatives for the blind to access books, specifically for people in Thailand. This project was divided into two groups which were audiobook production and audiobook playback production. Our group was audiobook playback production. Using the audiobooks provided by the other group, we made a website for the blind to listen to the audiobooks. In addition, our website was designed to be easily accessible for the blind. Therefore, the blind can use many devices to access the website. Also, we used natural language processing(NLP) so that blind users can interact with the website. We asked a group of blind people several questions about problems that they had faced while surfing the Internet and their expectations of an audiobook website. Finally, we delivered an effective and suitable audiobook website for the blind.

Keywords: Blindary/ Wit.ai API / blind/ NLP/ Youtube IFrame API/

Google Book API/ audiobook/ online library

**ACKNOWLEDGEMENT**

We would like to use this opportunity to express our deepest thanks to Asst. Prof. Chonlameth Arpnikanondt, Ph.D., our project advisor and to Mr. Suthat Ronglong, our project co-advisor. Also, we would like to thank Mr. Pramok Promdach for his recommendations about the problems that the blind face. We would like to thank Mr. David Hargrave Cook and ELCS for carefully proofreading our project documents to make them as refined and complete as they are now.

**CONTENTS**

**PAGE**

FRONT COVER i

INSIDE COVER AND APPROVAL PAGE ii

ENGLISH ABSTRACT iii

ACKNOWLEDGEMENT iv

CONTENTS v

LIST OF TABLES vii

LIST OF FIGURES viii

CHAPTER 1 INTRODUCTION

1.1: Background 1

1.2: Objectives 1

1.3: Scope 1

1.4: Expected Benefits 2

CHAPTER 2 FEASIBILITY STUDY

2.1: Introduction 3

2.2: Problem Statement 3

2.3: Related Research and Project 4

2.3.1: Read for the Blind 4

2.3.2: TAB2Read 5

2.3.3: Smart AudioBook Player 5

2.4: Requirement Specifications 6

2.5: Implementation Techniques 6

2.5.1: Software 6

2.5.2: Computer Languages 6

2.6: Deliverables 6

2.7: Implementation Plan 7

2.7.1: Task Description 7

2.7.2: Gantt Chart 9

2.8: Project Success and Failure 10

2.8.1: Task 10

2.8.1.1 Playback 10

2.8.1.2 Production 10

2.8.1.3 Information that we have share together 10

2.8.2: Program Success Rate 12

2.8.2.1 Success 12

2.8.2.2 Challenged 12

2.8.2.3 Failure 12

CHAPTER 3 SYSTEM ANALYSIS AND DESIGN

3.1: Introduction 13

3.2: Existing System Analysis 13

3.3: User Requirement Analysis 14

3.4: System Design 14

**CONTENTS (Cont.)**

**PAGE**

3.4.1: Sequential Diagram 15

3.4.1.1: Feedback Phase of the Application 15

3.4.2: Use-Case Diagram 16

3.4.3: Entity Relationship Diagram (ER Diagram) 17

3.4.4: Data Dictionary of System 19

3.4.5: User Interface 22

CHAPTER 4 SYSTEM FUNCTIONALNITY

4.1: Introduction 28

4.2: Main Functions 28

4.2.1: Registration Page 29

4.2.2: Log in Page 30

4.2.3: Library Page 30

4.2.4: History Page 31

4.2.5: Search Page 31

4.2.6: Audiobook Player Page 32

4.2.7: Audiobook Request Page 33

4.2.8: Administrator Log In Page 33

4.2.9: Administrator Page 33

4.2.10: Approval Page 34

4.3: Test Plans and Test Results 34

CHAPTER 5 SUMMARY AND SUGGESTION

5.1: Introduction 38

5.2: Project Summary 38

5.3: Problem Encountered and Solution 38

5.3.1: Wit.ai 38

5.3.2: DAISY Digital Talk Book 38

5.3.3: Audiobook Player 38

5.3.4: Registration Page 38

5.4: Suggestion for Further Development 39

5.4.1: Audiobook Player with Wit.ai 39

5.4.2: Natural Language Processing 39

5.4.3: User Interface 39

REFERENCES 40

APPENDIX

A: USER MANUAL 42

**LIST OF TABLES**

**TABLE**  **PAGE**

2.1: Task Description 7

3.1: Existing System Analysis 13

3.2: BLIND Table 19

3.3: BLIND\_LOG Table 19

3.4: BLIND\_BOOKMARK Table 20

3.5: BLIND\_FAVORITE Table 20

3.6: BLIND\_REQUEST Table 20

3.7: BLIND\_RATING Table 20

3.8: BLIND\_FEEDBACK Table 21

3.9: CHAPTER Table 21

3.10: PARAGRAPH Table 21

4.1: User and Administrator Main Function 28

4.2: Test Plans and Test Results 34

**LIST OF FIGURES**

**FIGURES**  **PAGE**

2.1: Read for the Blind Application 4

2.2: TAB2Read Application 5

2.3: Smart AudioBook Player Application 5

2.4: Gantt Chart 9

2.5: Percentage of Tasks 11

2.6: Information that the Two Groups Share 11

2.7: Percentages of Success, Failure and Challenge 12

3.1: The System Design of Audiobook Production and Audiobook Playback 14

3.2: Feedback Phase Sequential Diagram 15

3.3: The Use-Case Diagram of Blindary from the Playback Side 16

3.4: The ER Diagram of Blindary 17

3.5: The ER Diagram of the Playback Side 18

3.6: Home Page 22

3.7: Log In Page 22

3.8: Registration Page 23

3.9: Instruction Page 23

3.10: Contact Us Page 24

3.11: Profile Page 24

3.12: Library Page 25

3.13: Bookmark Page 25

3.14: Favorite Page 26

3.15: History Page 26

3.16: Search Page 27

3.17: Audiobook Player Page 27

4.1: Registration Page 29

4.2: Registration Response Message 29

4.3: Log In Page 30

4.4: Library Page 30

4.5: History Page 31

4.6: Search Page 31

4.7: Audiobook Player Page 32

4.8: Audiobook Request Page 33

4.9: Administrator Page 33

4.10: Approval Page 34

A.1: Registration Page 42

A.2: Log In Page 42

A.3: Profile Page 43

A.4: Search Page 43

A.5: Library Page 44

A.6: Chapter Page 44

A.7: Audiobook Player Page 45

**CHAPTER 1**

**INTRODUCTION**

**1.1 Background**

Nowadays, the number of people who can access the Internet is increasing. They are not only people who can see like us but also people with disabilities, particularly blind people. The blind can use both computers and smartphones as do people who can see. An application that many blind people are using currently is Read for the Blind. It helps blind people access books and short stories. However, the current version of Read for the Blind cannot meet all the demands of users. Also, there are some other problems with the application. The blind need to have a smartphone to use the application. Android smartphones cannot adjust the speed of the speech, but iOS smartphones can. Also, some of the blind do not have a smartphone, so they cannot use this application. After we learned about the problems, we decided to create a new application on a web platform that can be used by all devices. This web application is going to help blind people access audiobooks and short stories. We have separated the project into two parts. One part is audiobook production, and the other part is audiobook playback. Both parts are going to be developed by separate teams. Our group will focus on the audiobook playback. Meanwhile, the other group will focus on audiobook production.

**1.2 Objectives**

After we observed the difficulties of accessing books for the blind, we understood why they cannot read books like other people. Mostly, the ways to find and access books are difficult. So, we decided to make a new web application that is easy for the blind to use. We will make a web application so that the blind can access the books that they want by using a UI specifically designed for the blind. Then blind people can request books that they want and find books that they want to listen to in our application by category. Moreover, we are going to put the content in a grid pattern. For example, the buttons will have the same size and same gaps on each page. Also, we will use colors in the UI that the color blind can see easily. We will use AI to help users interact with the application.

**1.3 Scope**

* Build AI that is able to interact with the blind
* Make an audiobook player
* Request new books for the blind
* Create a user friendly interface

**1.4 Expected Benefits**

* Improve the chances of the blind being able to access content
* Meet the needs of blind users

**CHAPTER 2**

**FEASIBILITY STUDY**

**2.1 Introduction**

In Thailand, there are about 600,000 blind people. In other words, about 1% of the Thai population is blind. Most of the blind have less opportunities to learn new things because they cannot see. So, it is hard for them to read books unlike other people, and it is harder for them to learn unlike people who can see. Although there are some applications for smartphones that can help the blind access books, not all blind people have smartphones. For these reasons, we want to help blind people by creating a website that helps them access many kinds of books. The blind can access the website by many devices that can connect to the Internet, such as smartphones, tablets and computers. Also, this website will help the blind to learn new things and gain more knowledge. They will easily access the website and listen to the audiobooks that they want because we will design the website especially for the blind.

**2.2 Problem Statement**

Blind people cannot access books like other people do. Most applications and websites are not designed for blind people, so they have to spend time to become familiar with them before they can use them. Most applications cannot meet the needs of blind people, and some applications are not designed for color blind people, who cannot see colors in applications correctly. Moreover, the designs of applications have to be specifically made for the blind. Because of this, we will use Usable Accessibility Design to create our web application. This means the buttons in our application will be the same size and aligned in grids. In addition, we will use AI that interacts with the blind when they are searching for books and using control commands on the playback page.

**2.3 Related Research and Project**

**2.3.1 Read for the Blind**

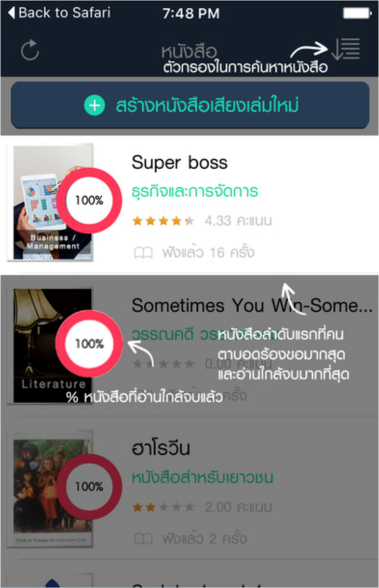


Figure 2.1: Read for the Blind Application

Read for the Blind is an application from the Thailand Association of the Blind. The system is divided into two parts, which are the volunteers part and the blind part. Volunteers can help blind people by reading a book, recording their voice, or uploading a recording. To use the blind part, the blind need to log in first, so they must get an ID from the Thailand Association of the Blind. After that, they can choose which books they want to listen to. Also, they can choose which reading voice they like to listen to.

**2.3.2 TAB2Read**

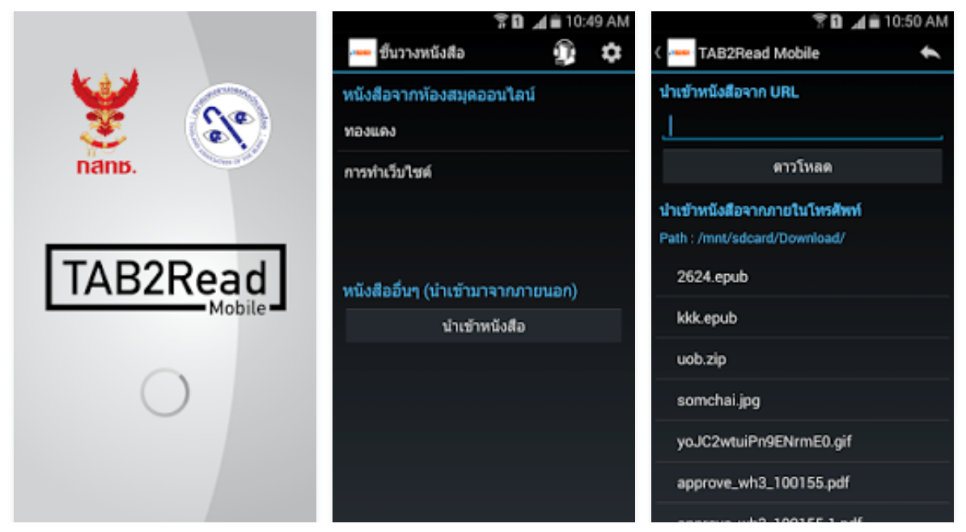


Figure 2.2: TAB2Read Application

The TAB2Read application is also provided by the Thailand Association of the Blind, but it is only for blind people. This application is a DAISY audiobook format reader. It can read books or text to the blind. To use this application, users import books to their device. Users can control the speed and volume of audiobooks by themselves.

**2.3.3 Smart AudioBook Player**

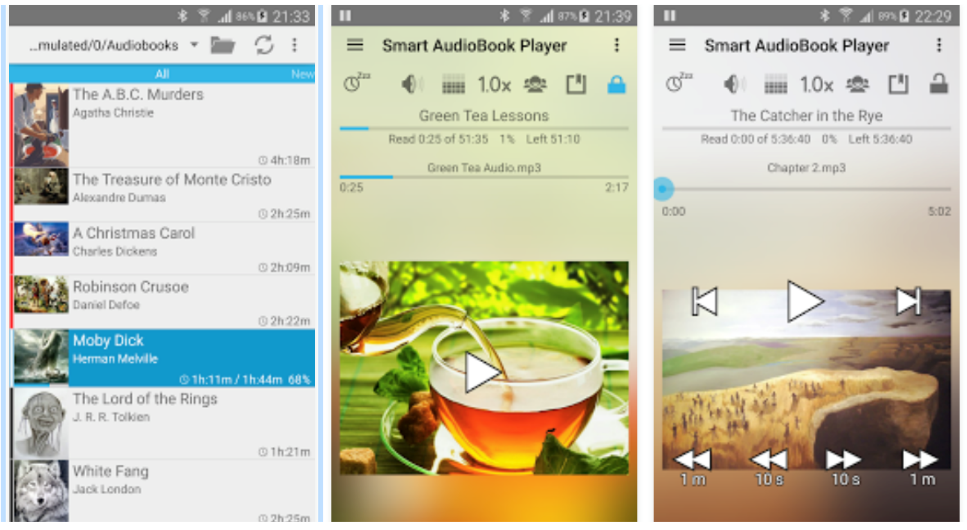


Figure 2.3: Smart AudioBook Player Application

The Smart AudioBook Player application was made by Alex Kravchenko. It is designed especially for playing audiobooks. However, this application is not designed for use by blind people, so it can be hard for blind people to use it.

**2.4 Requirement Specifications**

* Can log in by using username
* Can connect their device to the Internet
* Can allow users to control speed of playback
* Can recommend books
* Can receive feedback from users
* Can allow users to ask for new books by typing or saying a book’s name to search for it

**2.5 Implementation Techniques**

**2.5.1 Software**

* Brackets
* Sublime
* Cyberduck
* Photoshop

**2.5.2 Computer Languages**

* PHP
* HTML
* JavaScript

**2.6 Deliverables**

* Documentation
* Source code of the application
* Poster

**2.7 Implementation Plan**

**2.7.1 Task Description**

Table 2.1 shows the duration and description of each task and how each task is related to other tasks.

Table 2.1: Task Description

|  |  |  |  |
| --- | --- | --- | --- |
| Task No. | Description | Duration(days) | Dependencies |
| T1 | Brainstorm to select a topic | 1 | - |
| T2 | Discuss the project | 1 | T1 |
| T3 | Find the details that are necessary for the project | 3 | T2 |
| T4 | Submit draft topic | 1 | T3 |
| T5 | Research technical details | 5 | T4 |
| T6 | Submit draft-proposal | 2 | T5 |
| T7 | Find more information about users | 10 | T6 |
| T8 | Discuss the scope | 2 | T7 |
| T9 | Identify the problem | 1 | T7 |
| T10 | Submit chapters 1-2 | 1 | T8, T9 |
| T11 | Prepare for feasibility presentation | 14 | T10 |
| T12 | Feasibility Presentation | 1 | T11 |
| T13 | Discuss the problem and poster | 10 | T12 |
| T14 | Improve document and make the poster | 10 | T12 |
| T15 | Submit chapters 1-2 and poster | 1 | T13, 14 |
| T16 | Discuss the problem with real users | 3 | T15 |
| T17 | Submit chapter 3 | 1 | T16 |
| T18 | Design Presentation | 1 | T17 |
| T19 | Edit chapters 1-3 | 5 | T18 |
| T20 | Submit chapters 2-3 and poster | 1 | T19 |

Table 2.1: Task Description (cont.)

|  |  |  |  |
| --- | --- | --- | --- |
| Task No. | Description | Duration(days) | Dependencies |
| T21 | Develop an alpha version | 80 | T20 |
| T22 | Test and debug the application | 20 | T21 |
| T23 | Demo Alpha software | 1 | T22 |
| T24 | D-Day for Beta software | 1 | T23 |

**2.7.2 Gantt Chart**

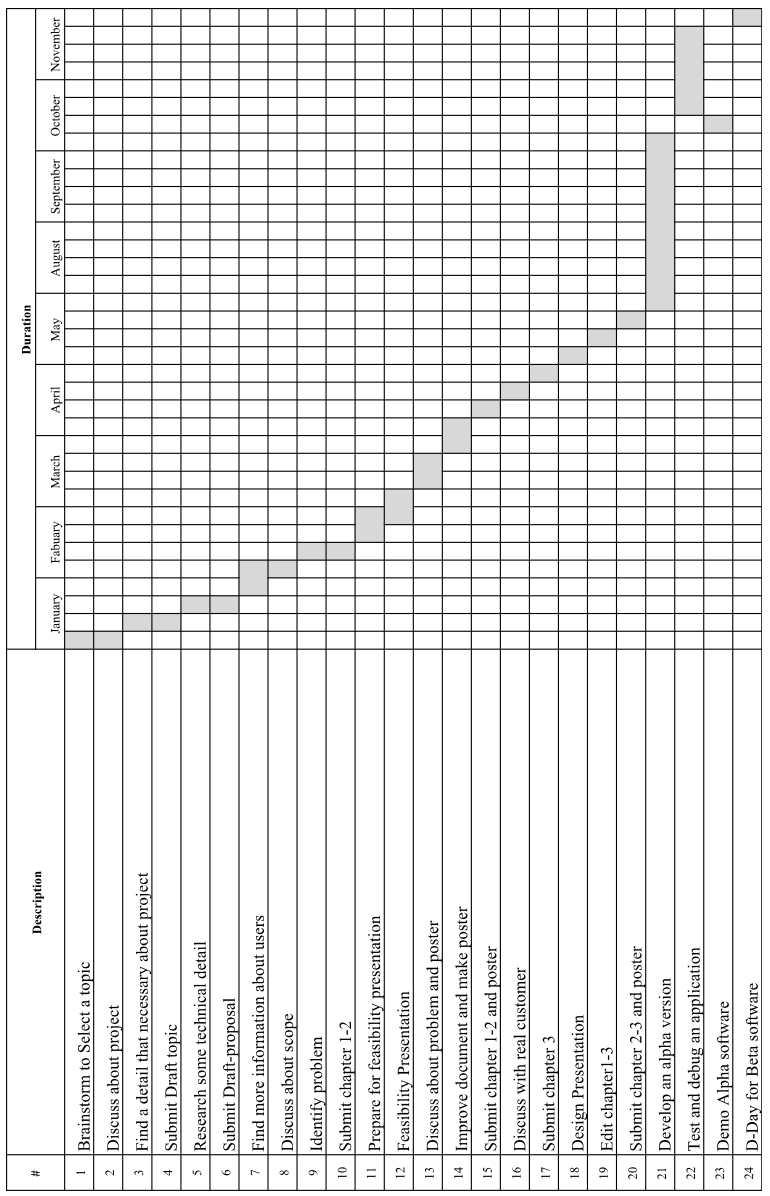


Figure 2.4: Gantt Chart

Figure 2.4 shows the schedule of project. The schedule plan helps us to work more efficiently by showing when we should finish each task.

**2.8 Project Success and Failure**

This part shows the main tasks of the two groups and shows the information that the two groups have to share together.

**2.8.1 Task**

**2.8.1.1 Playback**

* Design the user interface for the blind
* Test the user interface for accessibility
* Design the database for the blind
* Make audio able to play on the website
* Make it so users can control the speed of the audio
* Use NLP to help users to search for audiobooks
* Connect with the production part of the project
* Test and fix bugs

**2.8.1.2 Production**

* Design the user interface
* Design and maintain the database
* Use Google Books APIs to search for and retrieve book information
* Develop a voice recording system
* Provide an audio editor tool on the website
* Upload and manage each user’s voice
* Use Youtube to store the audio
* Connect with the playback part of the project
* Get book suggestions and feedback from the blind
* Testing and debugging

**2.8.1.3 Information that we have share together**

* Audio recordings of audiobooks
* Feedback of the blind
* Book suggestions from the blind

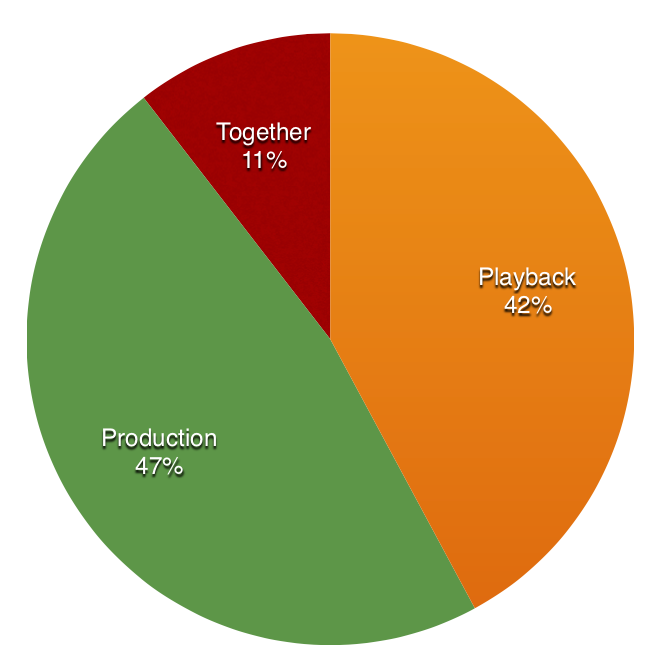


Figure 2.5: Percentages of Tasks

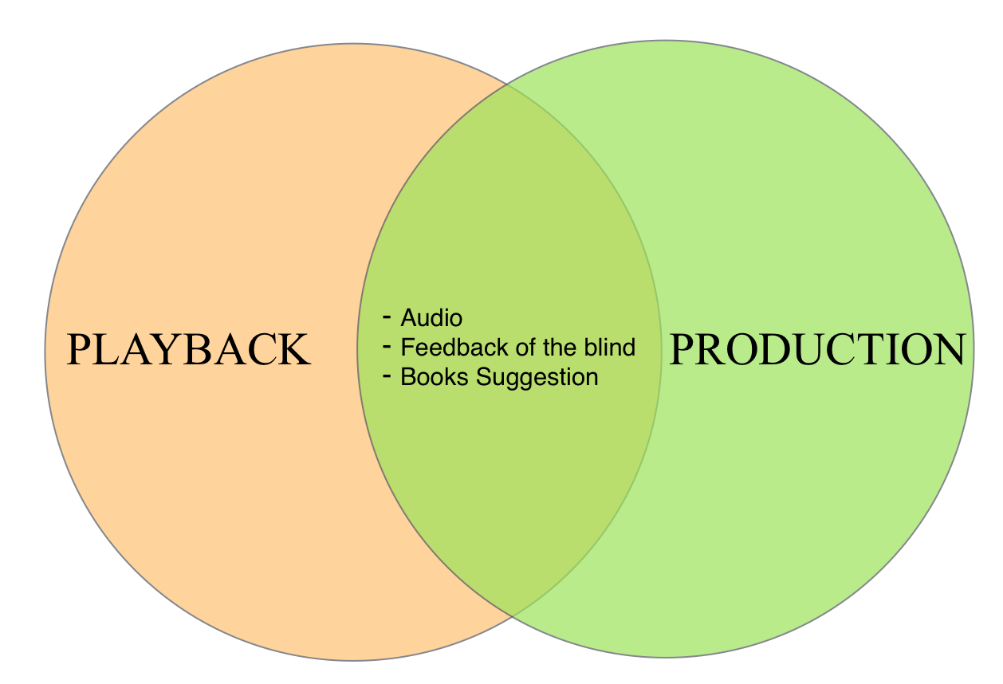


Figure 2.6: Information that the Two Groups Share

Figure 2.5 shows the proportions of tasks in Blindary by percentage. The information that the two groups share is shown in figure 2.6**2.8.2 Program Success Rate**

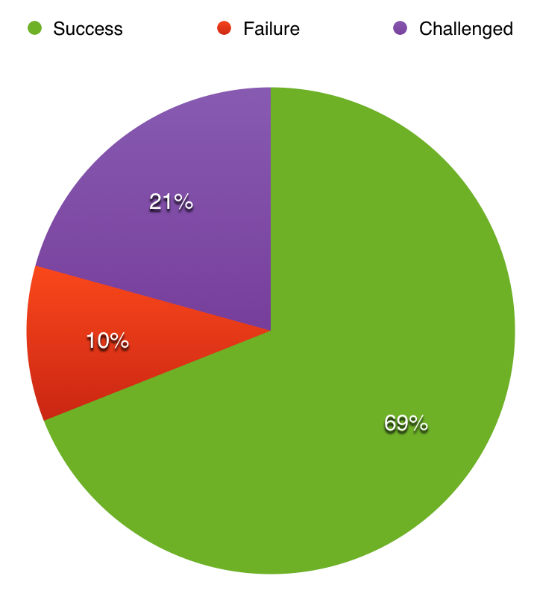


Figure 2.7: Percentages of Success, Failure and Challenge

Figure 2.7 shows the success percentage, failure percentage, and challenged percentage of the project. We have listed some possible successful accomplishments and problems that can occur during our project below.

**2.8.2.1 Success**

* The project is finished on time
* The project meets all requirements
* The blind can use the website with easy access
* The website can use the data from the other group

**2.8.2.2 Challenged**

* It might take a longer time than we expected to complete the project
* Some problems with technical details might occur
* The website has to be easy for the blind to use
* The website has to be able to combine the functions from both two groups

**2.8.2.3 Failure**

* The project may not have been well planned
* Not enough communication may occur between the two groups

If the other group does not finish on time, we plan to make a prototype of the audiobook and use it in the Blindary website instead.

**CHAPTER 3**

**SYSTEM ANALYSIS AND DESIGN**

**3.1 Introduction**

This chapter describes the design of the project. System analysis and design is one of the most important tasks in a project because a better design can help to reduce problems that might happen in the future. Especially, this chapter is about the appearance of the website. First, it covers how to make the website easy for the blind to use. Second, it covers the design of the database which is made to work efficiently with the website. This chapter also includes the use-case diagram, flow chart, and more.

**3.2 Existing System Analysis**

There are some applications and websites that were made for the blind, but most are mobile applications. Table 3.1 shows the comparison of the Blindary website to the existing mobile applications.

Table 3.1: Existing System Analysis

|  |  |  |
| --- | --- | --- |
| Application Name | Existing Mobile Applications | Blindary Website |
| Read for the Blind | Can be used only on smartphones | Can be used both on computers and smartphones |
| Read for the Blind, Tab2Read | Search for books by typing | Search by typing or voice command |
| Read for the Blind, Tab2Read | Cannot comment on audiobooks | Can send feedback to volunteers |
| Read for the Blind, Tab2Read | Cannot request books directly | Can request books directly |

**3.3 User Requirement Analysis**

This section describes how the users interact with the system. Users of this web application can do the following.

**User**

* Create an account
* Adjust playback speed of an audiobook
* Control audiobook playback to move position backward and forward
* Comment on and review an audiobook
* Give feedback to volunteers
* Search for books by typing or voice command
* Select books from recommended lists
* Select books from categories
* Request new books that are not yet available
* Change user interface to color blindness mode

**Administrator**

* Check user the disability ID card that users must use to register
* Add new users

**3.4 System Design**

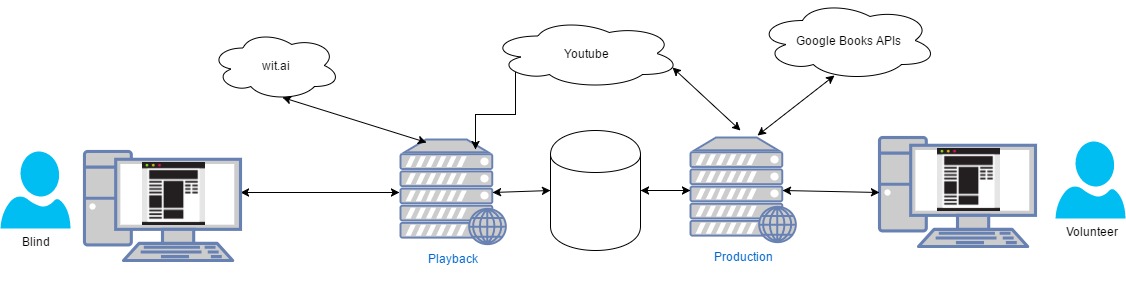


Figure 3.1: The System Design of Audiobook Production and Audiobook Playback

Blindary is a website that combines two services together, which are audiobook production and audiobook playback. Both audiobook production and audiobook playback use the same database to store data, but audiobook production and audiobook playback have different target users. Audiobook production provides audiobooks for blind users while audiobook playback provides services for the blind users. So, audiobook production and audiobook playback were separated to make each website easy to maintain. In audiobook playback, the blind users access the website via the Internet. When a user searches for audiobook to listen to, the system gets a list of books for the user. After the user selects the book, the system gets the link stored in the database and plays the audio from a specified Youtube link.

Blind users have two ways to search for audiobooks in our website, which are by typing or speaking. Audiobook playback uses Wit Speech API from Wit.ai to capture the speech of the user and transform it into text. Then Wit.ai is used to get keywords from the text. After that, the website responds to the input. The audioplay playback also uses Youtube IFrame Player API to make the audiobook player for the blind hide the Youtube link and video.

**3.4.1 Sequential Diagram**

**3.4.1.1 Feedback Phase of the Application**

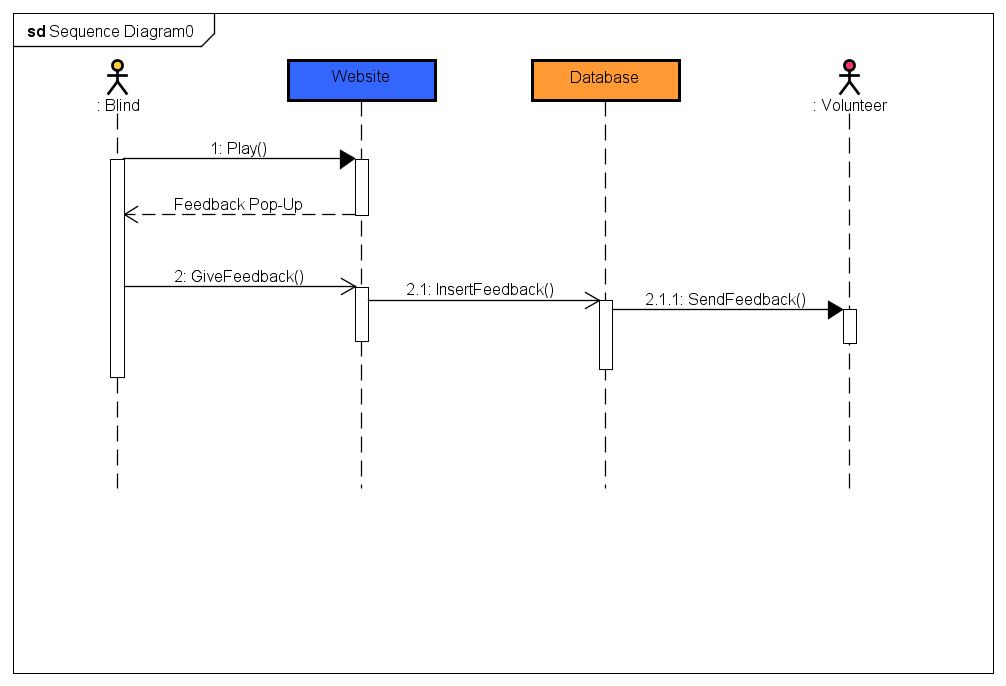


Figure 3.2: Feedback Phase Sequential Diagram

Figure 3.2 shows how a blind user gives feedback to volunteers about an audiobook that they have read. After the blind user has listened to the audiobook, he/she receives a form from the website for giving feedback to volunteers. However, giving feedback is optional for users. The user can either give feedback or ignore the form. If they give feedback, the feedback details are stored in the database, and the audiobook production website lets volunteers know if they got feedback by sending a notification to volunteers.

**3.4.2 Use-Case Diagram**

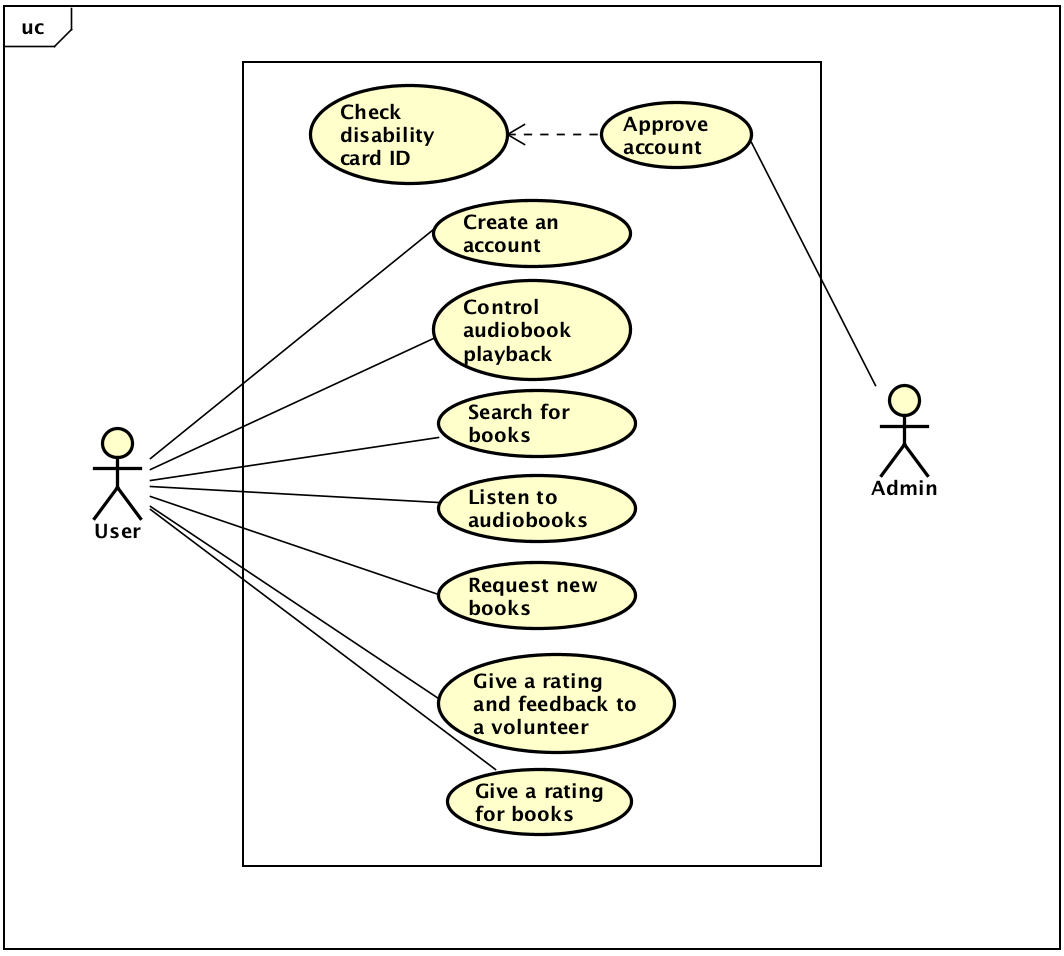


Figure 3.3: The Use-Case Diagram of Blindary from the Playback Side

Figure 3.3 shows the ways how blind users and administrators can interact with the website. First, users have to register by using a disability ID card and filling in some personal information to create an account. The administrator checks the blind person’s personal information and gives the user ability to access to the website. Users can control an audiobook by themselves. They can search for a book by typing or speaking keywords at the search page. They can request books not already on the website or give a rating and feedback to volunteers, such as if they want any improvements or if it is good enough. Moreover, users can give a rating to each book that they have listened to.

**3.4.3 Entity Relationship Diagram (ER diagram)**

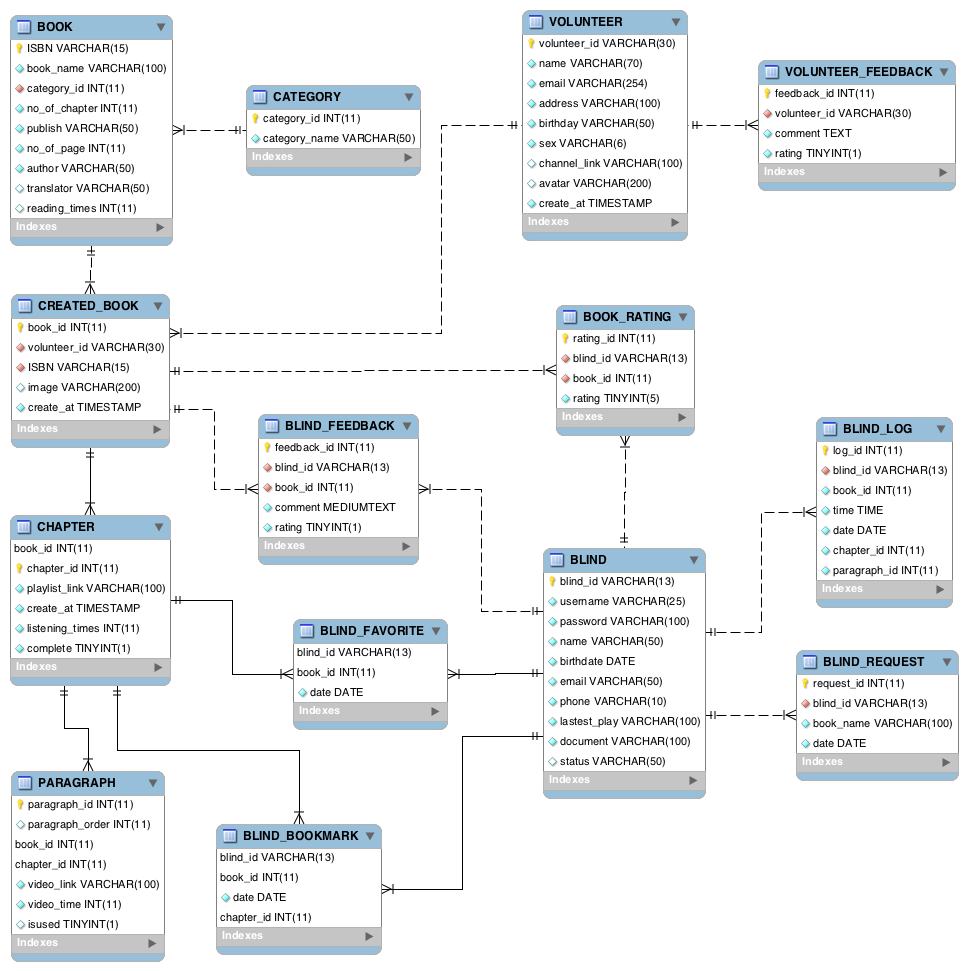


Figure 3.4: The ER Diagram of Blindary

Figure 3.4 shows the database of Blindary for both the audiobook playback and audiobook production. However, our group focused on the playback tables that are shown in figure 3.6 below.

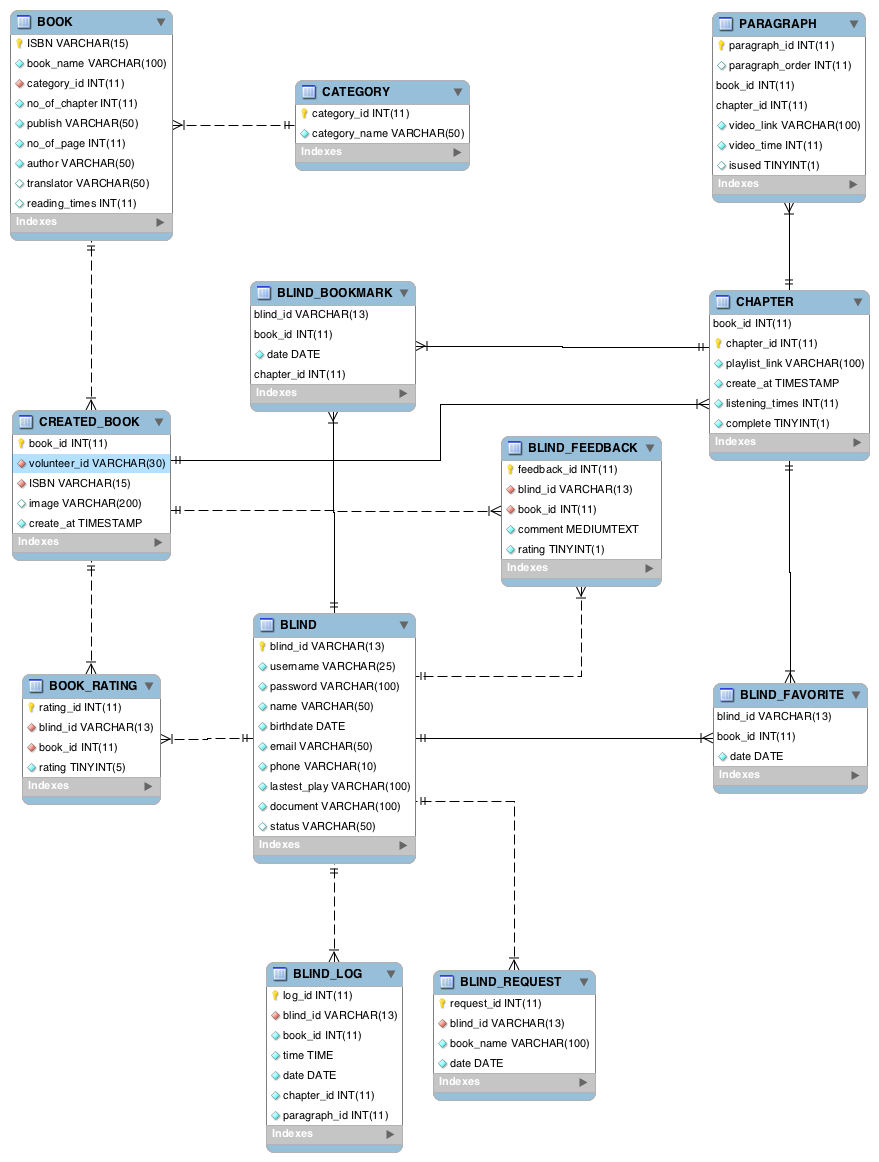


Figure 3.5: The ER Diagram of the Playback Side

Figure 3.5 shows the tables of Blindary on the playback side. All user information is stored in the BLIND table. This contains each username and password for logging into the system. What each user reads are stored in the BLIND\_LOG table. This table is used to bookmark a chapter that a user wants and is used to mark books as favorites for later listening. It stores requests for books that are not in the system. Moreover, the blind who listen to an audiobook can give feedback to volunteers who produce the audiobooks.

**3.4.4 Data Dictionary of System**

A data dictionary is a description of the data in database and covers each table. It can help people to understand more about what data is kept in each table.

Table 3.2: BLIND Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field name | Description | Data type | Data length | Key |
| blind\_id | Disability ID | varchar | 13 | PK |
| username | Username for loging into the system | varchar | 25 |  |
| password | password | varchar | 100 |  |
| name | Name of user | varchar | 50 |  |
| birthdate | User’s birthdate | date |  |  |
| email | User’s email | varchar | 50 |  |
| phone | User’s phone number | varchar | 10 |  |
| lastest\_play | Link to the last audio that a user listened to before closing the website | varchar | 100 |  |
| document | Directory of each user’s identification document | varchar | 100 |  |
| status | Status of a user  e.g. approved, not approved, pending | varchar | 50 |  |

Table 3.3: BLIND\_LOG Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field name | Description | Data type | Data length | Key |
| log\_id | Log ID | integer | 11 | PK |
| blind\_id | Disability ID | varchar | 13 | FK |
| book\_id | ISBN or ID of an audiobook | varchar | 11 |  |
| date | Date that a user listened to an audiobook | date |  |  |
| chapter\_id | Chapter that a user listened to | int |  |  |
| paragraph\_id | Paragraph that a user listened to | int |  |  |

Table 3.4: BLIND\_BOOKMARK Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field name | Description | Data type | Data length | Key |
| book\_id | ISBN or ID of an audiobook | integer | 11 | PK |
| blind\_id | Disability ID | varchar | 13 | FK |
| chpter\_id | Chapter that a user listened to | varchar | 10 |  |
| date | Date that a user bookmarked an audiobook | timestamp |  |  |

Table 3.5: BLIND\_FAVORITE Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field name | Description | Data type | Data length | Key |
| book\_id | ISBN or ID of an audiobook | integer | 11 | PK |
| blind\_id | Disability ID | varchar | 13 | FK |
| date | Date that a user favorited an audiobook | timestamp |  |  |

Table 3.6: BLIND\_REQUEST Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field name | Description | Data type | Data length | Key |
| request\_id | Request ID | integer | 11 | PK |
| blind\_id | Disability ID | varchar | 13 | FK |
| book\_name | Name of a requested book | varchar | 100 |  |
| date | Date requested | date |  |  |

Table 3.7: BLIND\_RATING Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field name | Description | Data type | Data length | Key |
| rating\_id | Rating ID | integer | 11 | PK |
| blind\_id | Disability ID | varchar | 13 | FK |
| book\_id | ISBN or ID of an audiobook | integer | 11 | FK |
| rating | Rating of a book | tinyint | 5 |  |

Table 3.8: BLIND\_FEEDBACK Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field name | Description | Data type | Data length | Key |
| feedback\_id | Feedback ID | integer | 11 | PK |
| blind\_id | Disability ID | varchar | 13 | FK |
| book\_id | ISBN or ID of an audiobook | integer | 11 | FK |
| comment | Comment to a volunteer | midiumtext |  |  |
| rating | Rating that a user gave to an audiobook | tinyint | 5 |  |

Table 3.9: CHAPTER Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field name | Description | Data type | Data length | Key |
| book\_id | ISBN or ID of an audiobook | integer | 11 | PK |
| chapter\_id | ID of a chapter | integer | 11 | FK |
| playlist\_link | Link of a chapter playlist | varchar | 100 | FK |
| create\_at | Time that a chapter was created | timestamp |  |  |
| listeining\_times | Time(s) that it was listened to | integer | 11 |  |
| complete | Status of the chapter (completed or not complete) | tinyint | 1 |  |

Table 3.10: PARAGRAPH Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field name | Description | Data type | Data length | Key |
| paragraph\_id | ID of a paragraph | integer | 11 | PK |
| paragraph\_order | Second ID of a paragraph | integer | 11 | FK |
| book\_id | ISBN or ID of audiobook | integer | 11 | FK |
| chapter\_id | Chapter ID that a paragraph came from | integer | 11 |  |
| video\_link | Link to the paragraph | varchar | 100 |  |
| vidoe\_time | Time of this video | integer | 11 |  |
| isused | Status of the paragraph | tinyint | 1 |  |

**3.4.5 User Interface**

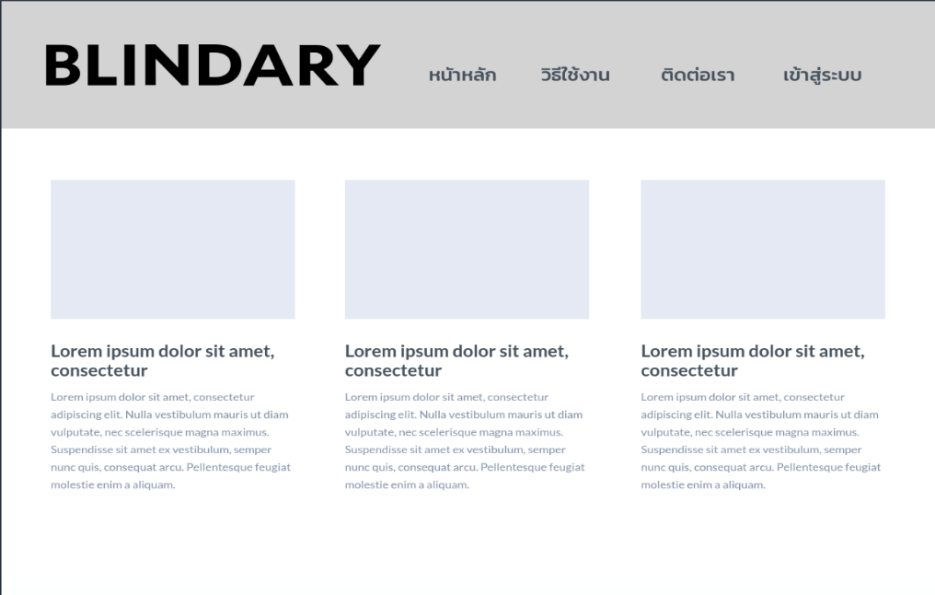


Figure 3.6: Home Page

Figure 3.6 shows the first page seen when a user opens the website. It contains news and announcements from the website and related information for the blind. Users can log into the website by clicking on “เข้าสู่ระบบ” button.



Figure 3.7: Log In Page

Figure 3.7 shows the log in page. If a user forgets his/her password, the user can click on “ลืมรหัสผ่าน” button to request to a new password or on Contact Us for Help. If person is not a member yet, the user can register by clicking on “สมัครสมาชิก” button.

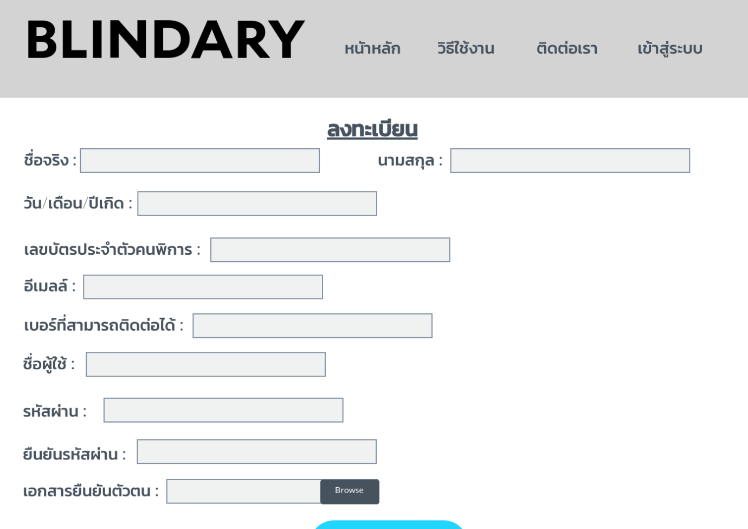


Figure 3.8: Registration Page

Figure 3.8 shows the registration page. At this step, the blind may ask another person to register for them if they cannot register by themselves. The user has to fill in the form and upload documents that are necessary to identify himself/herself.

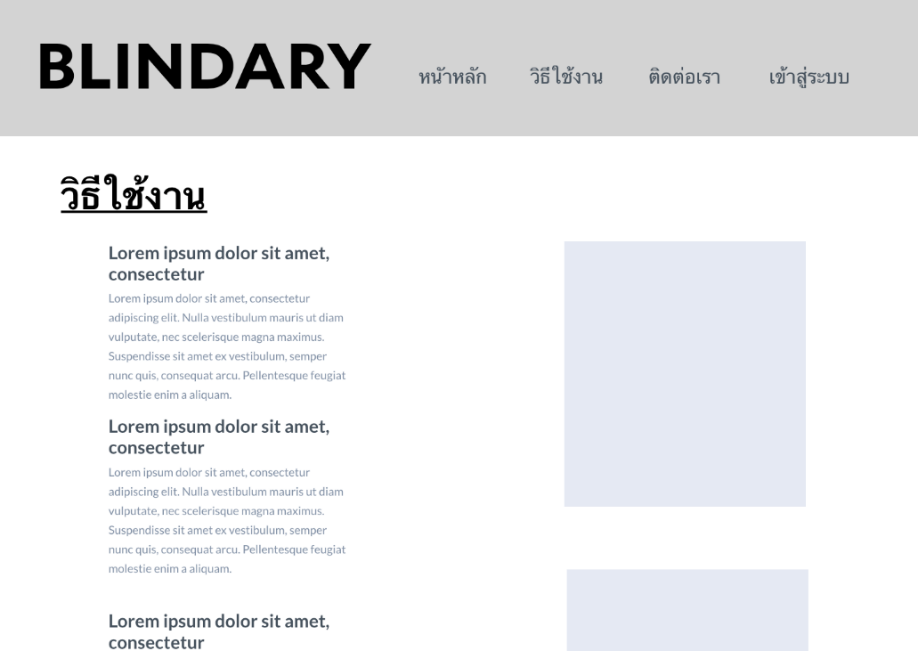


Figure 3.9: Instruction Page

Figure 3.9 shows the instruction page. For people who might not be familiar with using a website, it provides instructions of how to use the website. Clicking “วิธีใช้งาน” will make it easier to use the website.

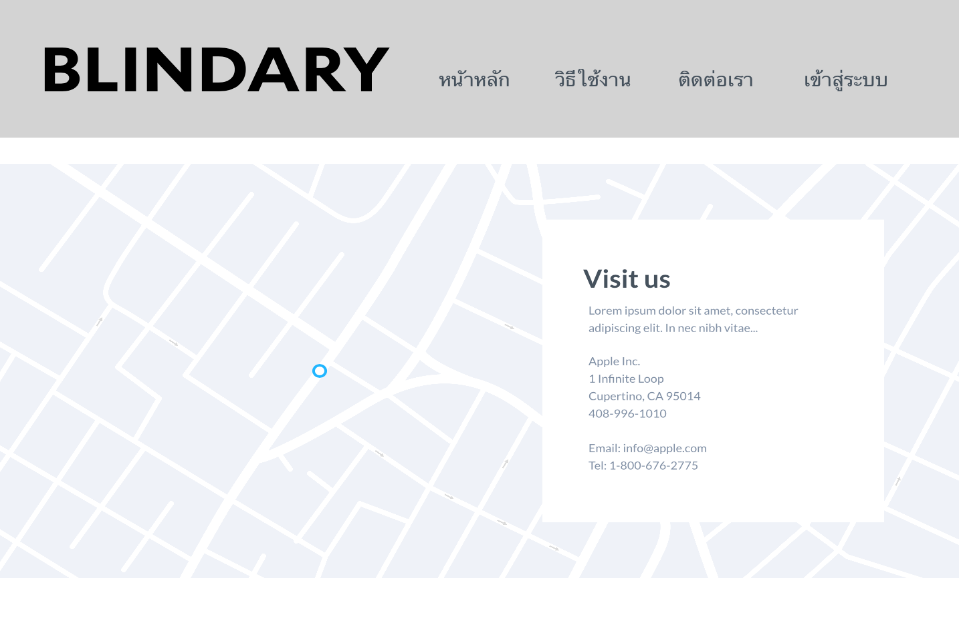


Figure 3.10: Contact Us Page

Figure 3.10 shows the contact us page. Users can contact the administrator if any problems occur or if the user wants to know further information.

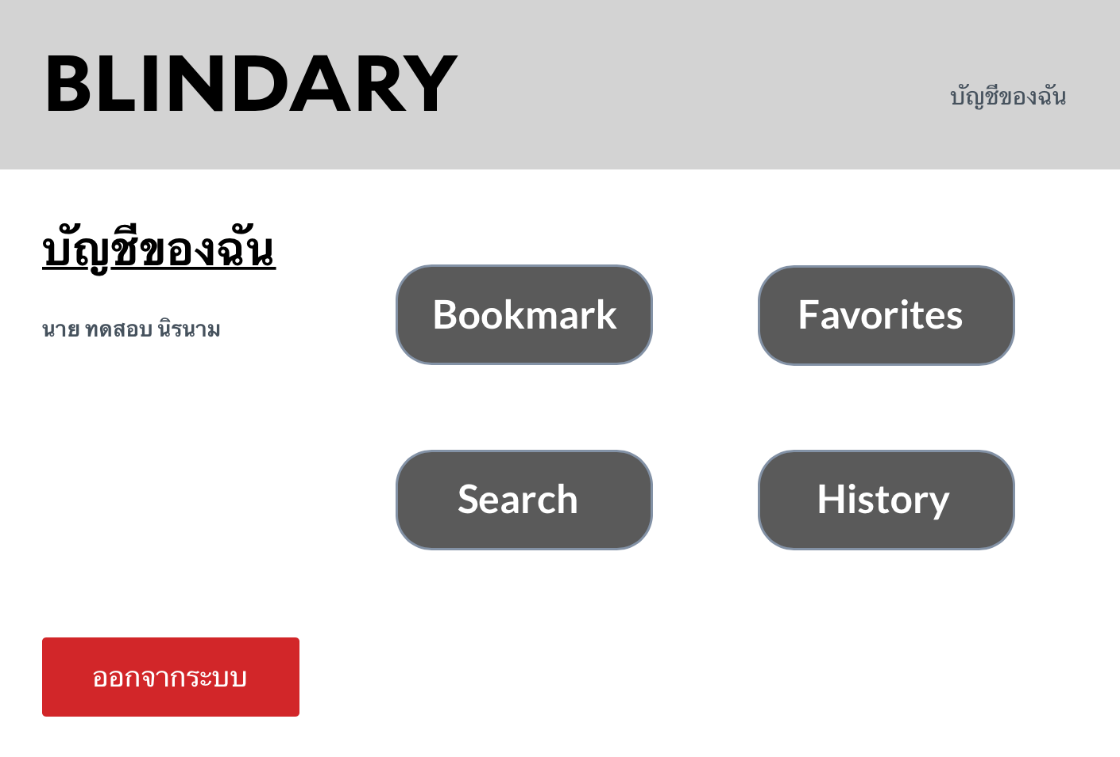


Figure 3.11: Profile Page

Figure 3.11 shows the profile page of a user. After users log into the website, it takes users to this page. It contains the name of the user, and five buttons which are bookmark, favorites, search, history and log out.

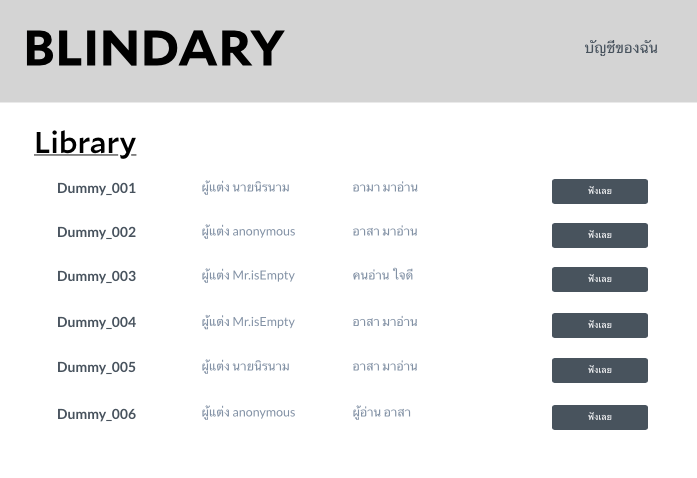


Figure 3.12: Library Page

Figure 3.12 shows the list of audiobooks that a user can find by using the search function or choosing the view all audiobooks function.



Figure 3.13: Bookmark Page

Figure 3.13 shows the bookmark page where users can bookmark audiobooks that they want to listen to later.

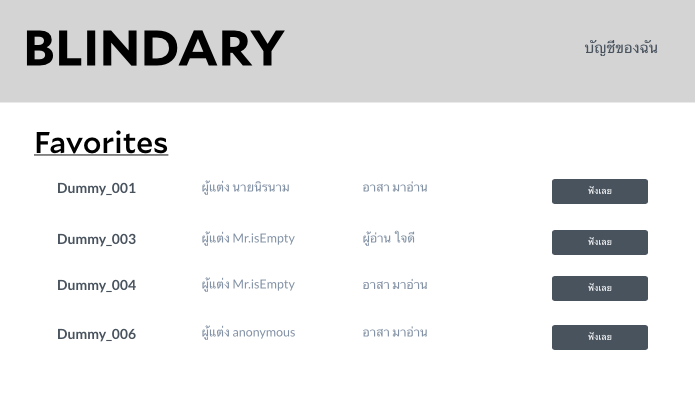


Figure 3.14: Favorite page

Figure 3.14 shows the favorites page. Users can add audiobooks that they like to the favorites list. By using favorites, users can listen to audiobooks that they like more easily.

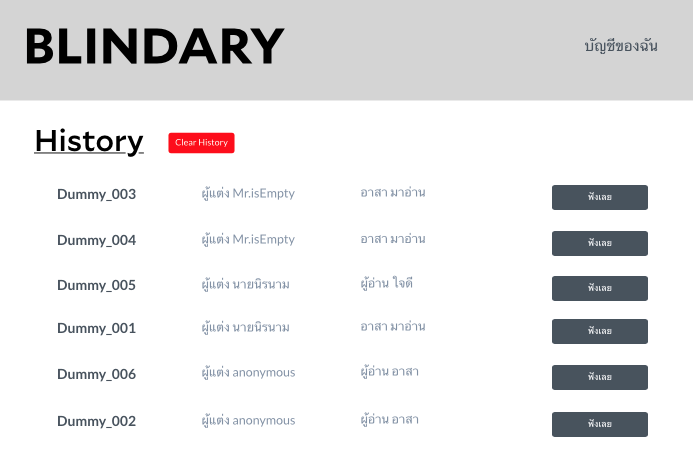


Figure 3.15: History Page

Figure 3.15 shows the history page. The website keeps a list of audiobooks that a user has listened to. Also, users can delete the history list by clicking the Clear History button.

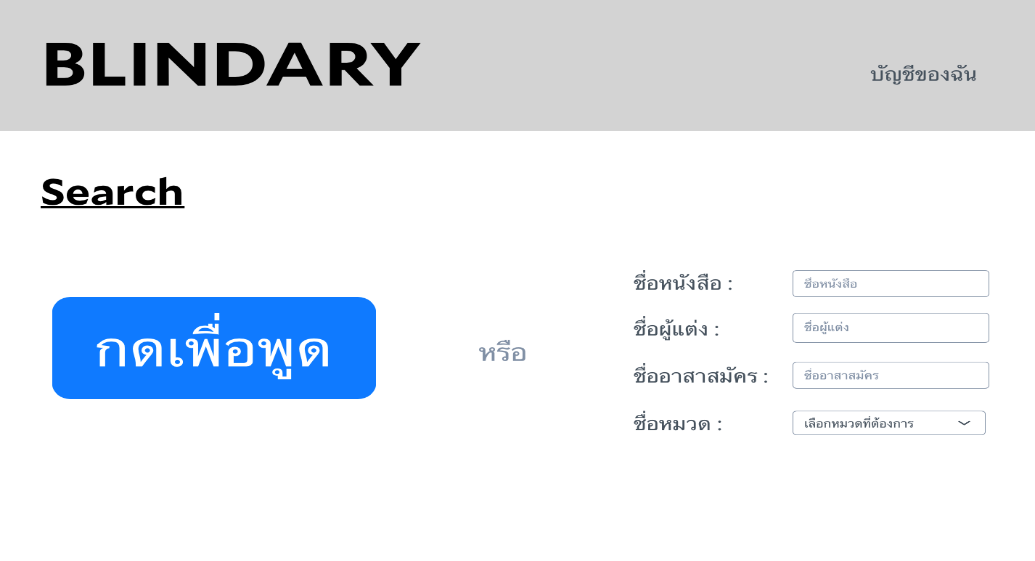


Figure 3.16: Search Page

Figure 3.16 shows the search page. Users can choose how they want to search for audiobooks. Users can tell the bot to help them or they can type in the details of an audiobook.



Figure 3.17: Audiobook Player Page

Figure 3.17 shows the audiobook player page. After users click the ฟังเลย button, it takes users to this page. On this page, users can select audiobooks by pressing the buttons. Also, they can set the playback speed of audiobooks to be slower or faster than normal. The player control buttons are arranged in a pattern that has the same size of buttons and the same amount of space between buttons. So, blind users can use the website more easily. Moreover, on devices that have a touch screen like smartphones and tablets, users can drag their finger around and listen to a voice over.

**CHAPTER 4**

**SYSTEM FUNCTIONALITY**

**4.1 Introduction**

This chapter describes the main functions and test plan of the system. First, the main functions section describes how the user interacts with the website. Next, the test plan of the project section consists of the description of each test case and its results.

**4.2 Main Functions**

Table 4.1: User and Administrator Main Functions

|  |  |
| --- | --- |
| User Type | Functions |
| User | Register for an account on the Blindary website  Log into the Blindary website  View all audiobooks  View the history of audiobooks previously listened to  Play an audiobook  Select audiobook chapters  Select audiobook paragraphs  Bookmark an audiobook  Favorite an audiobook  Give a rating for an audiobook  Give feedback to volunteers  Search for an audiobook by using voice command  Search for an audiobook by typing  Adjust the audiobook playback speed  Request a new book  Share an audiobook to Facebook  Share an audiobook to Twitter |
| Administrator | Log in as an administrator of the Blindary website  Approve a user account  View user information and documents |

**4.2.1 Registration Page**

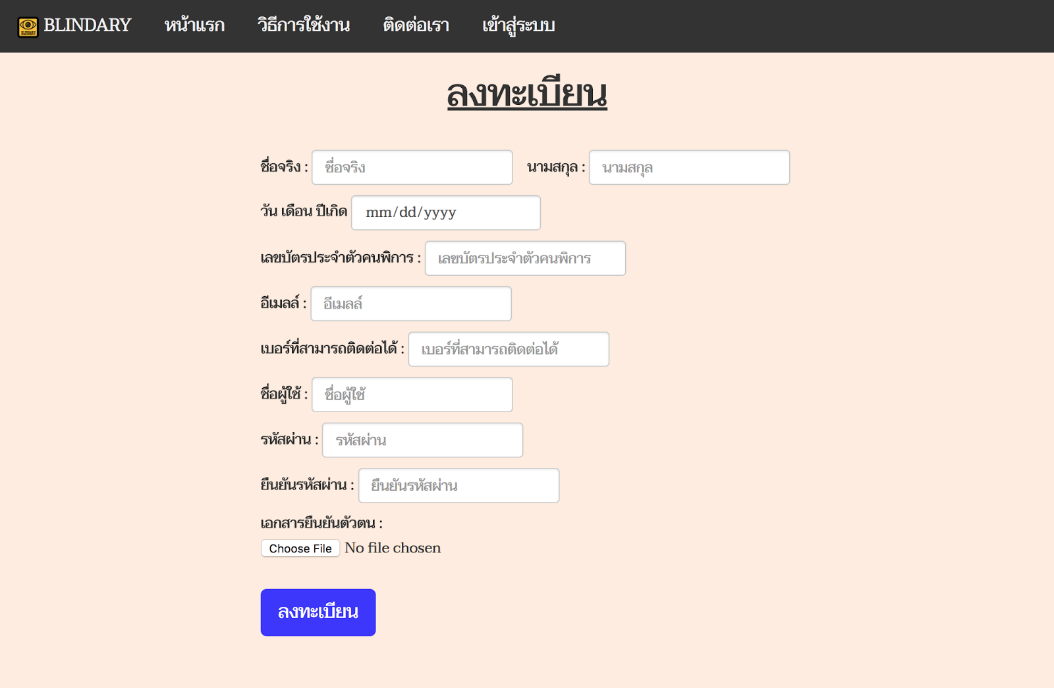


Figure 4.1: Registration Page

Figure 4.1 shows the registration page where users have to fill all textfields to register and then upload a file to the website to prove that the user is blind, such as an image of their disability ID card. After the user clicks the “ลงทะเบียน” button, the website gives them a reply message such as shown below in figure 4.2.

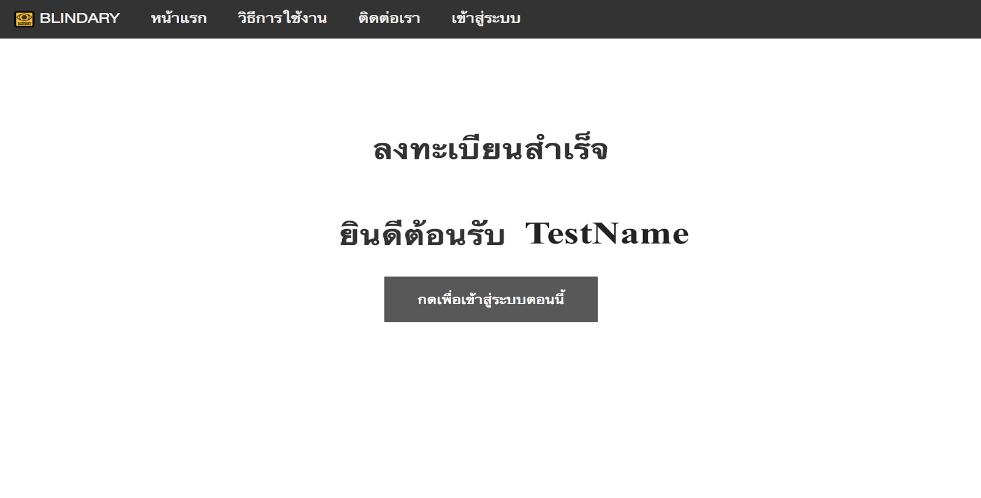


Figure 4.2: Registration Response Message

Figure 4.2 shows the response message after registration.

**4.2.2 Log In Page**



Figure 4.3: Log In Page

Figure 4.3 shows the log in page where the system requires input of a correct username and password to log into the website.

**4.2.3 Library Page**

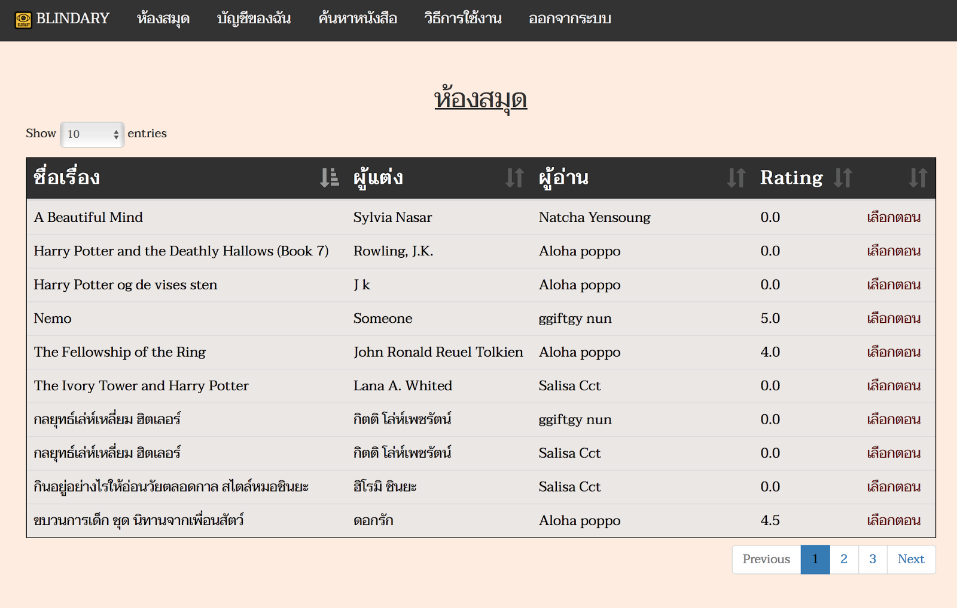


Figure 4.4: Library Page

Figure 4.4 shows the library page where users can view a list of audiobooks. The user can limit the number of audiobooks shown on each page, and the user can sort the audiobooks by clicking the table header.

**4.2.4 History Page**



Figure 4.5: History Page

Figure 4.5 shows the history page. After login, the website takes users to the “บัญชีของฉัน” page. The user has to click the “ประวัติการฟัง” button to visit the history page. On the history page, it shows the last 20 items that a user has listened to. Certain details are shown which are the book’s title, book’s author, and the date and time that the user listened to it. Moreover, the user can click the “ฟังเลย” button to listen to a book.

**4.2.5 Search Page**

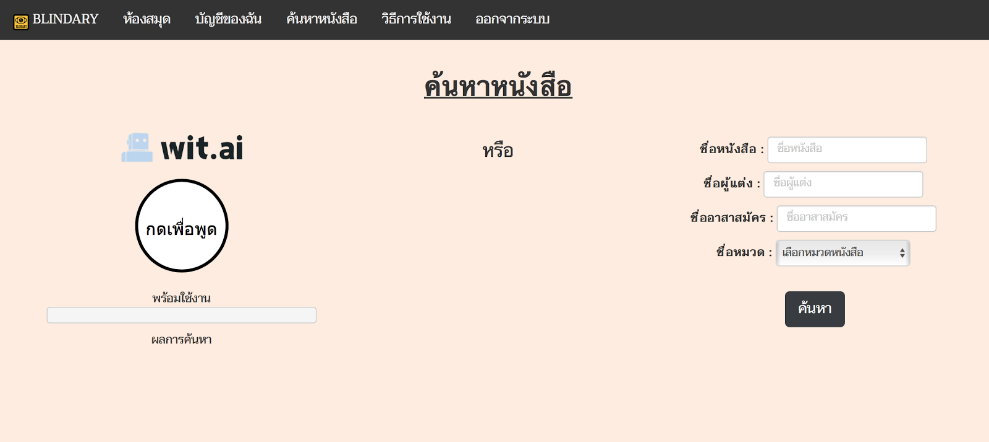


Figure 4.6: Search Page

Figure 4.6 shows the search page where the user can search by typing or speaking. For speaking, the user has to click the “กดเพื่อพูด” button and then speak the desired words. When the user has finished speaking, the user has to click the “หยุด” button. After this has been done, the system searches using keywords in the speech.

**4.2.6 Audiobook Player Page**

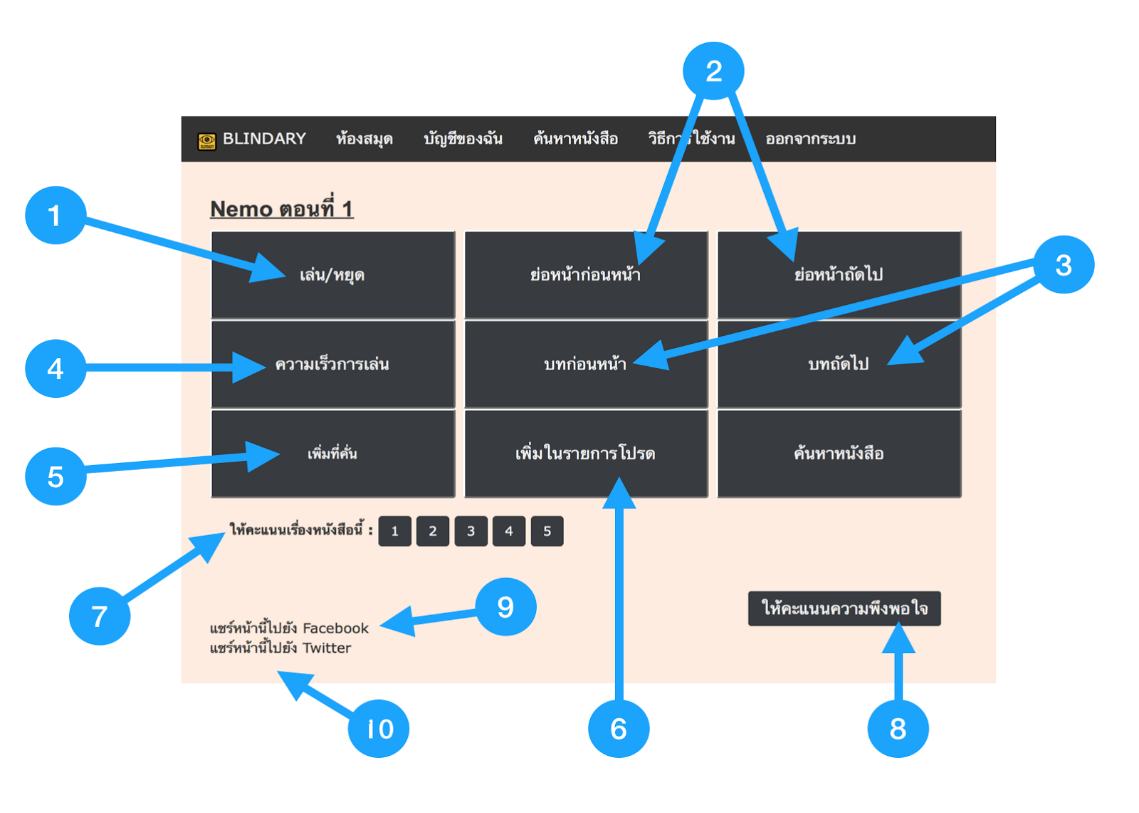


Figure 4.7: Audiobook Player Page

Figure 4.7 shows the audiobook player page. After the users clicks the “ฟังเลย” button on any page, the website takes the user to the audiobook player page.

There are 10 main functions on this page:

1. Play or pause audio
2. Go to previous or go to next paragraph
3. Go to previous chapter or next chapter
4. Adjust the audiobook playback speed
5. Add or remove a chapter bookmark
6. Add or remove an audiobook favorite
7. Give a rating for the current audiobook
8. Give feedback to volunteer
9. Share the current chapter to Facebook

10. Share the current chapter to Twitter

**4.2.7 Audiobook Request Page**



Figure 4.8: Audiobook Request Page

Figure 4.8 shows the audiobook request page. After login, the website takes users to the “บัญชีของฉัน” page. The user has to click the “ส่งคำขอเพิ่ม” button to visit the audiobook request page. On the audiobook request page, the user can enter an ISBN number or book title to request an audiobook that is not in the website. After the user fills in an ISBN number or book title, the user has to click the “ส่งคำขอ” button.

**4.2.8 Administrator Log In Page**

The administrator can log into the website by using the same login page as the user(Figure 4.3), but after login, the website takes the administrator to the administrator page.

**4.2.9 Administrator Page**



Figure 4.9: Administrator Page

Figure 4.9 shows the administrator page. On this page, administrator can search for users by using the search text field. The administrator can view more details of each user by clicking the “ข้อมูลเพิ่มเติม” button.

**4.2.10 Approval Page**

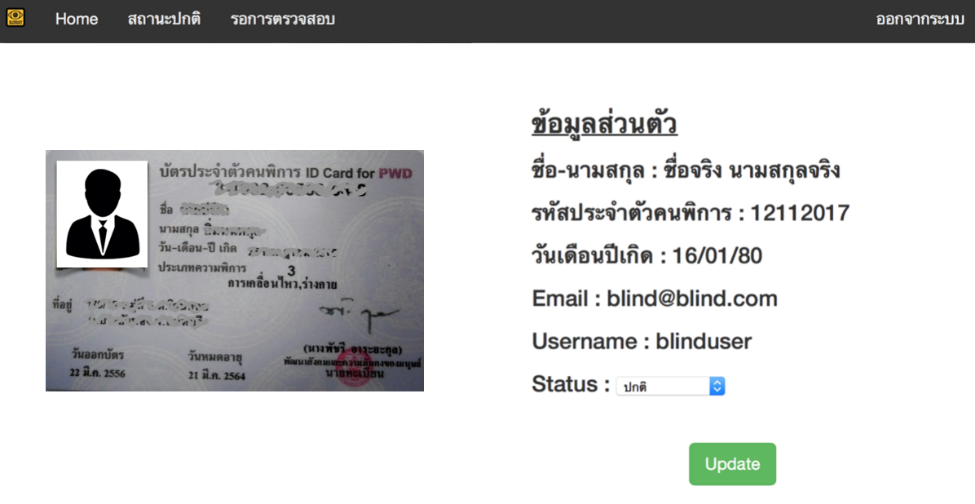


Figure 4.10: Approval Page

Figure 4.10 shows the approval page where the administrator checks a person’s details before approving the person to use the website.

**4.3 Test Plan and Test Results**

Table 4.2: Test Plan and Test Results

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Function | Test Case | Test Flow | Expected Output | Actual Output | Result |
| Login | User enters both a correct username and password | User entered both a username and password correctly | Takes user to member page | Took user to member page | Passed |
| User leaves username field blank | User left the username field blank | Prompts user to enter username | Prompted user to enter username | Passed |
| User enters an incorrect username or password | User entered an incorrect username or password | Prompts user to try again | Prompted user to try again | Passed |
| Register | User enters an available disability number | User entered an available disability number | Shows user that user can use this number | Showed user that user can use this number | Passed |

Table 4.2: Test Plan and Test Results (cont.)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Function | Test Case | Test Flow | Expected Output | Actual Output | Result |
| Register | User enters a disability number that was already used | User entered a disability number that was already used | Shows user disability number was already used | Showed user disability number was already used | Passed |
| User enters an available email | User entered the email correctly | Shows user can use the email | Showed user can use the email | Passed |
| User enters an email that was already used | User entered an email that was already used | Shows the email was already used | Showed the email was already used | Passed |
| User enters an available username | User entered the username correctly | Shows user can use this username | Showed user can use the username | Passed |
| User enters a username that was already used | User entered a username that was already used | Shows the username was already used | Showed the username was already used | Passed |
| Bookmark | User clicks add to bookmark button | User clicked add this chapter to bookmarks on player page | Change text in button to be “ลบที่คั่น” | Changed text in button to be “ลบที่คั่น” | Passed |
| User clicks remove from bookmark button | User clicked remove this chapter from bookmarks on player page | Change text in button to be “เพิ่มที่คั่น” | Changed text in button to be “เพิ่มที่คั่น” | Passed |
| Favorite | User clicks add to favorites | User clicked add this book to favorites | Change text in button to be “เพิ่มในรายการโปรด” | Changed text in button to be “เพิ่มในรายการโปรด” | Passed |

Table 4.2: Test Plan and Test Results (cont.)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Function | Test Case | Test Flow | Expected Output | Actual Output | Result |
| Favorite | User clicks to remove from favorites | User clicked to remove this book from favorites | Change text in button to be “ลบจากรายการโปรด” | Changed text in button to be “ลบจากรายการโปรด” | Passed |
| Rate audiobook | User gives a rating to audiobook | User gave a rating to audiobook | Show text as “ให้คะแนนหนังสือแล้ว” | Showed text as “ให้คะแนนหนังสือแล้ว” | Passed |
| Search | User searches available audiobooks by typing | User entered text on search form | Lists the audiobook that matches the search the result | Listed audiobook that matched the search the result | Passed |
| User searches for unavailable audiobooks by typing | User entered text on the search form | Nothing happens | Nothing happened | Passed |
| User searches for an available audiobook by using voice command | User clicked “กดเพื่อพูด” and spoke a voice command | Lists the audiobook that matches the search value | Listed the audiobook that matched the search value | Passed |
| User searches for an unavailable audiobook by using a voice command | User clicked “กดเพื่อพูด” and spoke a voice command | Nothing happens | Nothing happened | Passed |
| Book Request | User makes an ISBN request | User completed the ISBN field correctly | Shows user that the request was sent | Showed user that the request was sent | Passed |

Table 4.2: Test Plans and Test Results (cont.)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Function | Test Case | Test Flow | Expected Output | Actual Output | Result |
| Book Request | User makes a request with an ISBN that was already requested | User completed an ISBN that was already requested | Shows user that this book is in process | Showed user that this book is in process | Passed |
| User requests an invalid ISBN | User entered ISBN that does not match with Google Book API | Shows user that the website does not find this ISBN | Showed user that the website did not find this ISBN | Passed |
| User makes a request by using a book title | User entered the book title field correctly | Shows user that the request was sent | Showed user that the request was sent | Passed |
| User requests a book title that was already requested | User entered a book title that was already requested | Shows user that this book is in process | Showed user that this book is in process | Passed |

**CHAPTER 5**

**SUMMARY AND SUGGESTIONS**

**5.1 Introduction**

This chapter summarizes all details of the project. First, the project summary describes the project briefly. Second, the problems that were encountered and their solutions are covered. Lastly, suggestions for further development of the Blindary project are given.

**5.2 Project Summary**

Blindary is a website that combines two services together which are audiobook production and audiobook playback. For blind users, the audiobook playback service published audiobooks that were created by the audiobook production service. The website was designed to support easy access. Using the search function, users have two ways to search for audiobooks in the website, which are by typing or speaking. The audiobook playback service uses Wit.ai API to capture the speech of the user and also makes the system understand the meaning of the user’s speech.

**5.3 Problems Encountered and Solutions**

**5.3.1 Wit.ai API**

Wit.ai API supports the Thai language, but we had to train the NLP API to understand a given sentence by using many examples because Wit.ai API is not designed especially for use with the Thai language. So, we sent the search page to many people to get more patterns of sentences to train the NLP API.

**5.3.2 DAISY Digital Talk Book**

First, we tried to use DAISY as a player, but we found that DAISY API contained a lot of problems. After that, we found Youtube IFrame API which made creating the player easier than if we had used DAISY.

**5.3.3 Audiobook Player**

The audiobook player has to show the book title with the current chapter. The audiobook player page contains many functions and each function has to work properly whenever it is used. When we used PHP, the website seemed to be slow. So, we used JavaScript to call each function because we wanted the player to work without reloading the page, and we wanted it to update the chapter number without any mistakes.

**5.3.4 Registration Page**

If users use Safari as their browser, users can type alphabet characters in the birth date field on the registration page. This should be impossible. So, we had to add a new JavaScript function to validate this field before users can submit the form.

**5.4 Suggestions for Further Development**

**5.4.1 Audiobook Player with Wit.ai**

Developers should use Wit.ai in the audiobook player to help the user control the player more easily. Then the user could control it by using voice command.

**5.4.2 Natural Language Processing**

Developers could find another natural language processing service that supports the Thai language better and with more accuracy than Wit.ai

**5.4.3 User Interface**

Developers should run Lighthouse in Chrome DevTools when there is some change in the user interface. Developers could use the audits for the accessibility function and get a score that is generated by Lighthouse of more than 90. Each audit has a reference document that explains each response message and how to fix the user interface.

# REFERENCES

1. Google, **YouTube Player API Reference for iframe Embeds** [Online], Available: https://developers.google[.](https://developers.google.com/youtube/iframe_api_reference)com[/](https://developers.google.com/youtube/iframe_api_reference)youtube[/](https://developers.google.com/youtube/iframe_api_reference)iframe\_api\_reference [2017, September 14].
2. Jekyll, **How–to: Use Skip Navigation links** [Online], Available: https://developers.google[.](https://developers.google.com/youtube/iframe_api_reference)com[/](https://developers.google.com/youtube/iframe_api_reference)youtube[/](https://developers.google.com/youtube/iframe_api_reference)iframe\_api\_reference [2017, Auguest 25].
3. Mayo, M., **Natural Language Processing Key Terms, Explained** [Online], Available: https://www.kdnuggets.com/2017/02/natural-language-processing-key-terms-explained.html [2017, Auguest 28].
4. Sierra, J. S., Roca de Togores, J. S., “Designing Mobile Apps for Visually Impaired and Blind Users”, The Fifth International Conference on Advances in Computer-Human Interactions, n/a, pp. 47-52, 2012.
5. The DAISY Consortium, **DAISY Digital Talking Book**[Online], Available: https://www.kdnuggets.com/2017/02/natural-language-processing-key-terms-explained.html [2017, Febuary 8].

**APPENDIX**

**APPENDIX A**

**USER MANUAL**

**A.1 Registration Page**

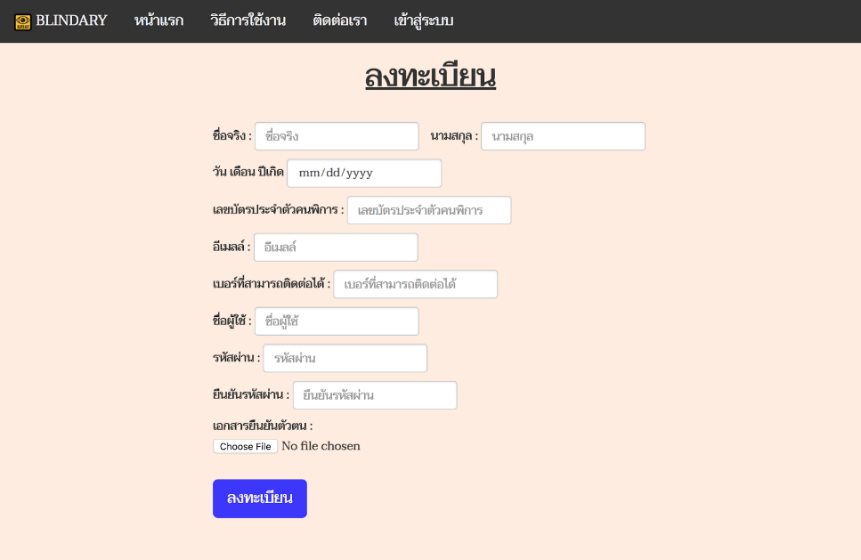


Figure A.1: Registration Page

Figure A.1 shows the registration page which allows the user to register for a Blindary account by filling in all the required information and uploading their identification document. Then the user can click the “ลงทะเบียน” button to register.

**A.2 Log In Page**



Figure A.2: Log In Page

Figure A.2 shows the log in page where the user enters his/her username and password. Then the user clicks the “เข้าสู่ระบบ” button to log into the Blindary website.

**A.3 Profile Page**

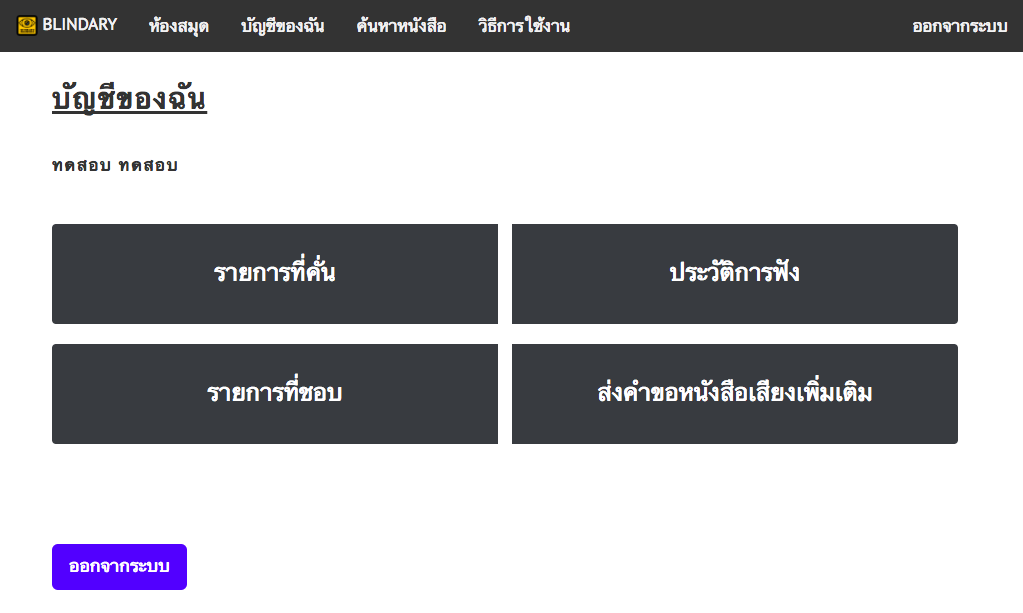


Figure A.3: Profile Page

Figure A.3 shows the profile page. After the user logs into the website, this page appears. On this page, there are four buttons that can take the user to other pages that the user might want to enter. These pages are the “รายการที่คั่น” page, “ประวัติการฟัง” page, “รายการที่ชอบ” page, and “ส่งคำขอหนังสือเพิ่มเติม” page.

**A.4 Search Page**

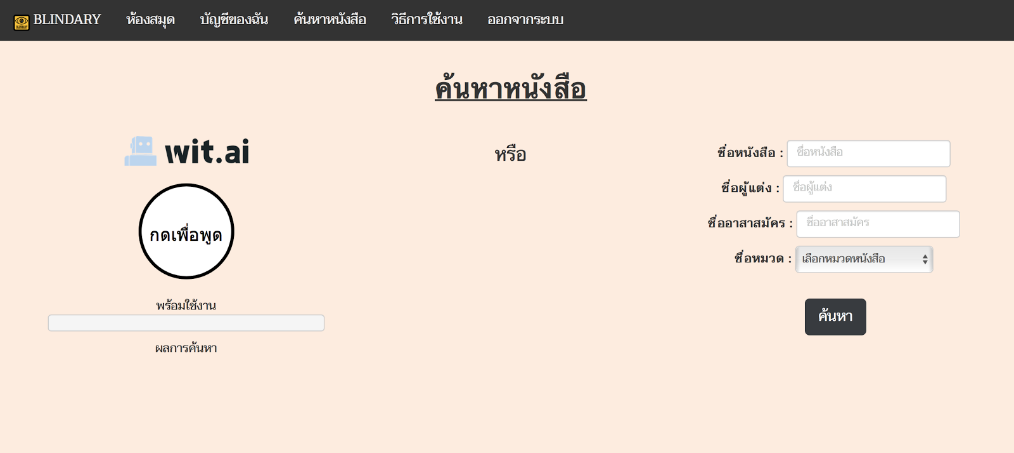


Figure A.4: Search Page

Figure A.4 shows the search page where the user can search for audiobooks in the Blindary website. On this page, the user can press the “กดเพื่อพูด” button to search for an audiobook by using voice or completeing search fields to search manually. **A.5 Library Page**

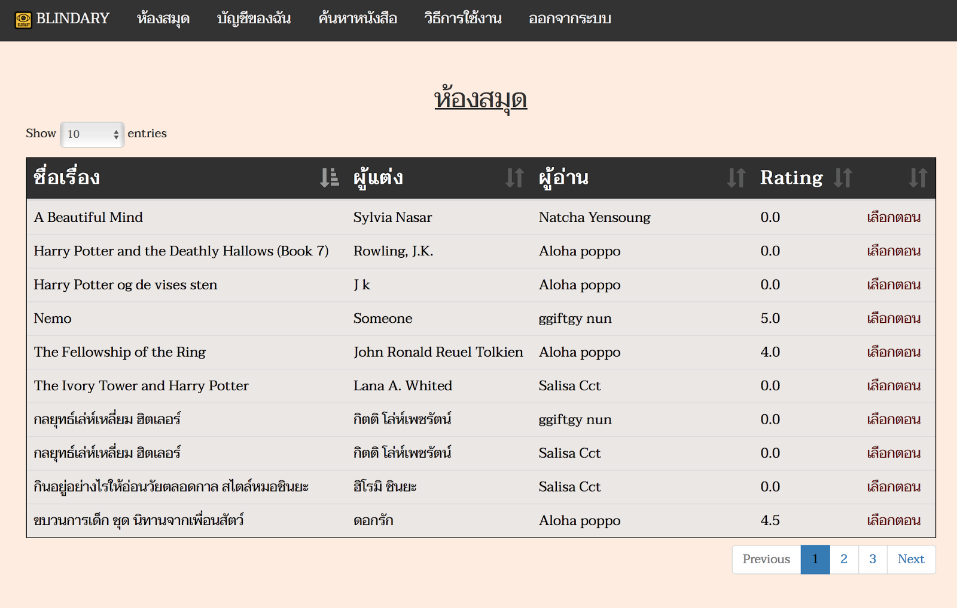


Figure A.5: Library Page

Figure A.5 shows the library page which displays all audiobooks in the Blindary website, and the user can clicks the “เลือกตอน” button to select chapters of a book.

**A.6 Chapter Page**



Figure A.6: Chapter Page

Figure A.6 shows the chapter page which displays all chapters in each audiobook. The user can select chapters by clicking the “ฟังเลย” button.

**A.7 Audiobook Player Page**



Figure A.7: Audiobook Player Page

Figure A.7 shows the audiobook player page where the user can listen to audiobooks. The users can play or pause an audiobook by clicking the “เล่น/หยุด” button. The user can go back to the previous paragraph by clicking the “ย่อหน้าก่อนหน้า” button or go to the next paragraph by clicking the “ย่อหน้าถัดไป” button. The user can adjust the audio playback speed by clicking the “ความเร็วการเล่น” button. The user can choose one of the four speeds which are 0.75x, 1.0x, 1.5x and 2x. The user can go back to the previous chapter by clicking the “บทก่อนหน้า” button or go to next chapter by clicking the “บทถัดไป” button. The user can add the current chapter to bookmarks by clicking the “เพิ่มที่คั่น” button. The user can add the audiobook to favorites by clicking the “เพิ่มในรายการโปรด” button. The user can go to the search page by clicking the “ค้นหาหนังสือ” button. Moreover, the user can rate the current audiobook by clicking a rating number button. The user can give feedback to the volunteer who created the current audiobook by clicking the “ให้คะแนนความพึงพอใจ” button. The user can also share the current audiobook by clicking the “แชร์หน้านี้ไปยัง Facebook” button or the “แชร์หน้านี้ไปยัง Twitter” button.

