**<Music Player>**

**System Design Specification (SDS)**

**<Version 9>**

**<5/25>**

<Project team members>

**张哲睿 2230026218**

**牟尹 2230026242**

**蔡杰超 2230026003**

**单佳影 2230026230**

**丁书萱 2230026232**

**Computer Science and Technology Program**

**United International College**

1. Document Change Log

|  |  |  |  |
| --- | --- | --- | --- |
| ***Change Date*** | ***Changed By*** | ***Version*** | ***Change Description*** |
| *23/04/2024* | *Cai Jiechao* | *1.0* | *Identification of topics* |
| *28/04/2024* | *Ding shuxuan* | 2.0 | *The UI interface* |
| *29/04/2024* | *Cai Jiechao* | 3.0 | Crawl data |
| *30/04/2024* | *Shan jiaying* | 4.0 | Load the data into UI interface |
| *05/05/2024* | *Zhang Zherui* | 5.0 | Implement relevant basic functions(play, pause, favorite...) |
| *07/05/2024* | *Cai Jiechao* | 6.0 | Add word cloud function |
| *07/05/2024* | *Ding shuxuan* | 7.0 | Optimize UI interface |
| *11/05/2024* | *Mu Yin* | 8.0 | Add play similar songs function and display hot songs/singers |
| *12/05/2024* | *Cai Jiechao* | 9.0 | Optimize the code |
|  |  |  |  |

1. Table of Contents

[1. Document Change Log 2](#_Toc394839007)

[2. Table of Contents 3](#_Toc394839008)

[3. Design Overview 4](#_Toc394839009)

[4. Tools and Standards 5](#_Toc394839010)

[4.1. Tools 5](#_Toc394839011)

[4.2. Standards 5](#_Toc394839012)

[5. User Interface Design 6](#_Toc394839013)

[5.1. Usage Scenario 1 6](#_Toc394839014)

[5.2. Usage Scenario 2 6](#_Toc394839015)

[6. Database Design 7](#_Toc394839016)

[7. Diagrams 8](#_Toc394839017)

1. Design Overview

*<Please provide a general overview of the project in terms of the product or service that is being built, so that anyone reading this section is able to gain an understanding of what functionality is provided and how this functionality will be technically implemented. Mention any partners that may be involved in the effort.>*

Make a music website that contains search songs, play songs, play previous/next songs, pause/resume songs, download songs, add songs to my favorite, display the current time and total time of the song, adjust the volume, load songs to the main interface from the data we crawled, show the top ten hits from different charts, show top singers by a chart, show lyrics and word cloud(contains lyrics data for 200 songs we have crawled, or the designated song).

1. Search Functionality

The search function filters the music list based on user input. It looks for matches in song attributes such as title, artist, and album. This is achieved through string matching within the music list data structure.

2. Play Songs

The application uses the pygame library to stream and play audio files. When a user selects a song, its URL is fetched, and the audio stream is played using pygame.mixer.music.

3. Play Previous/Next Songs

The application maintains an index of the current song. The previous and next song functions adjust this index to move through the music list and load the respective song URLs for playback.

4. Pause/Resume Songs

The pause and resume functionalities are managed through pygame.mixer.music methods pause() and unpause(), which control the playback state.

5. Download Songs

The download feature uses the requests library to fetch the song file from its URL and save it to the user's specified directory. It handles file I/O operations to write the binary content to a file.

6. Add Songs to Favorites

The application maintains a list of favorite songs. When a user adds a song to favorites, the song's data is appended to this list.

7. Display Current Time and Total Time of the Song

While playing a song, pygame provides the current position in the audio stream. The application calculates the minutes and seconds from this position and updates the UI accordingly.

8. Adjust Volume

Volume adjustment is handled through a tk.Scale widget for the user interface, which sends a value to the control\_voice function. This function sets the volume using pygame.mixer.music.set\_volume().

9. Load Songs to the Main Interface

Songs are loaded from an external data source (e.g., an Excel file or an API). The application parses this data and populates the music list, which is then displayed in a ttk.Treeview widget.

10. Show Top Hits from Different Charts

The application retrieves top hit songs from various charts by making API requests to a music service provider. The data is then processed and displayed in the UI.

11. Show Top Singers by a Chart

Similar to top hits, the application fetches data on top singers, likely from an API, processes this data, and presents it in a user-friendly format, possibly using matplotlib for visualization.

12. Show Lyrics and Word Cloud

Lyrics are loaded from Crawler fetch data. The application uses jieba for Chinese text segmentation and WordCloud from the wordcloud library to generate a word cloud visualization.

13. Data Crawling

The application uses libraries like requests and BeautifulSoup to scrape song data and lyrics from music websites. The scraped data is then structured into a usable format, such as a pandas DataFrame.

14. Audio Processing

The mutagen library is used to handle audio files, such as getting the length of a song for display.

1. Tools and Standards

## Tools

*<Specify which technologies will be used to develop the application, e.g. Python packages such as Numpy, Pandas, etc.>*

Packages include Numpy, PIL, matplotlib, json, os, tkinter, io, pygame, requests, pandas, mutagen.mp3, re, jieba, sklearn.feature\_extraction.text, sklearn.metrics.pairwise, random, re, BeautifulSoup, time, openpyxl

Use wordcloud package to generate the cloud image.

## Standards

*<Specify what standards will be followed in this project, e.g. runs on a Windows platform, etc.>*

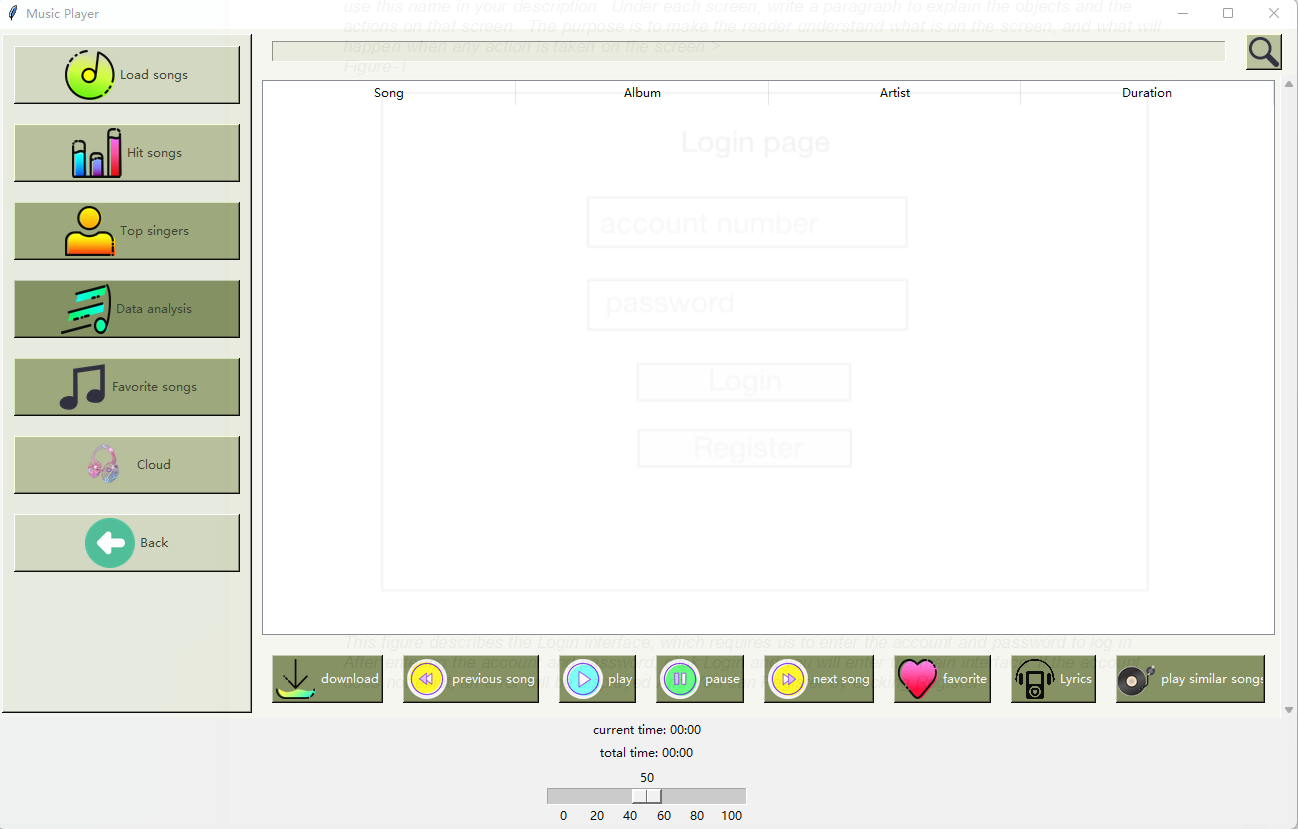
*runs on a Windows platform*

1. User Interface Design

## Usage Scenario 1

*<Put screen images here to describe each usage scenario step by step. Name each screen Figure-n and use this name in your description. Under each screen, write a paragraph to explain the objects and the actions on that screen. The purpose is to make the reader understand what is on the screen, and what will happen when any action is taken on the screen.>*

*Figure-1*

**

This figure is the main interface of our group project which contains multiple buttons and followings are the explanation:

In the left sidebar of the main screen, “Load songs” can choose a file we have already crawled to load the music and all the songs will appear in the middle of the interface.”Hit songs” displays the top ten hit songs from different charts.”Top singers” displays the top ten singers in the songs we have crawled.”Data analysis” displays the word cloud of 200 songs we have crawled.”Favorite songs” displays the list of songs we have added to favorite.”Cloud” displays the word cloud of the song we selected.”Back” will back to the song list we have loaded from the file we choose.

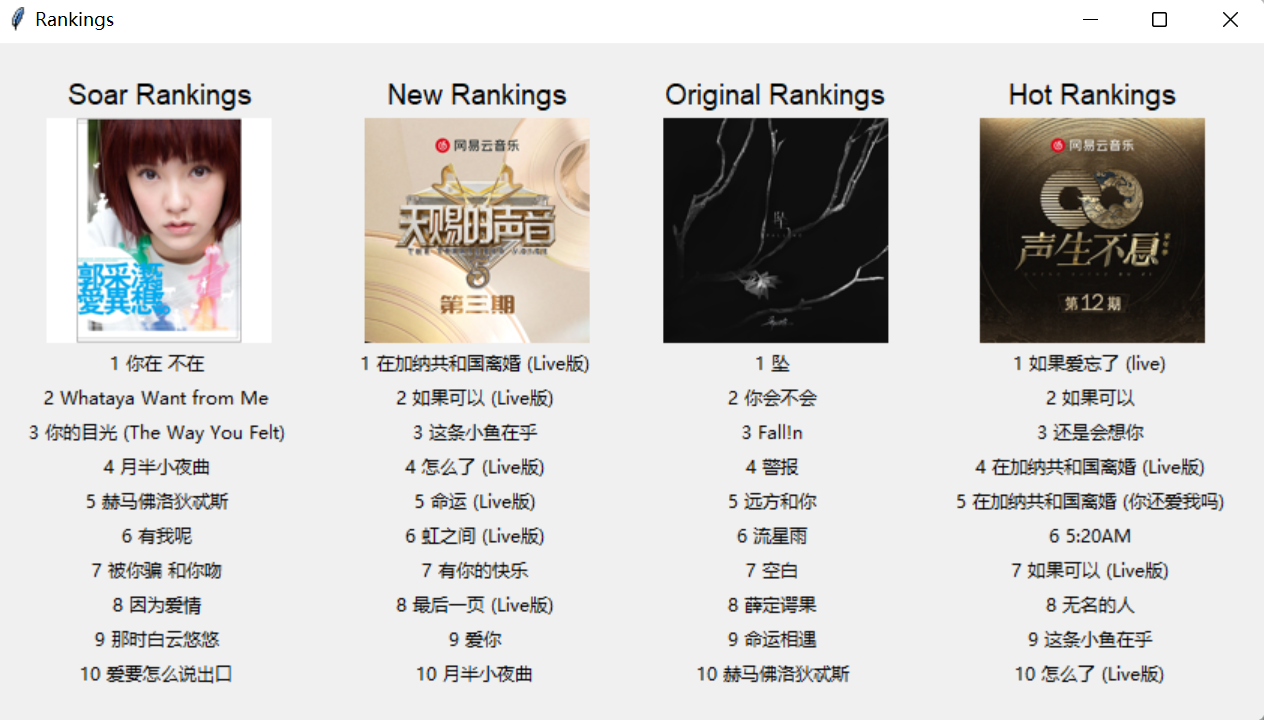
In the top button control panel, we only have a search box and a search button.You can type the name of the song or artist in the search box to search the song you want by click search button.

In the bottom control panel, there are 8 kinds of function.When click ”download” button, we will choose a folder to store the song we want to download.If the song we selected are not in the folder we choose, the song will appear in that folder and the message “song name download successfully!” will appear on the main interface.If the song we selected are already in that folder we choose, a message “song name already exists, skip download!” will appear.“play” will play the selected song and ”previous song” and “next song” will play the previous/next song of the selected song.”pause” will pause the song which is playing and this button will change with “resume”.”resume” will resume the song which is paused by click “pause” button and this button will change with ”pause”.”favorite” is to add the selected song into favorite song list with message “收藏了歌曲: song name”.And if the song we selected are in favorite song list, a message with “该歌曲已在我的收藏中”.”lyrics” shows the lyrics of the selected song with corresponding time.”current time” and “total time” displays the current time and total time of selected song.We have a scroll bar to adjust the volume of the song.

## Usage Scenario 2

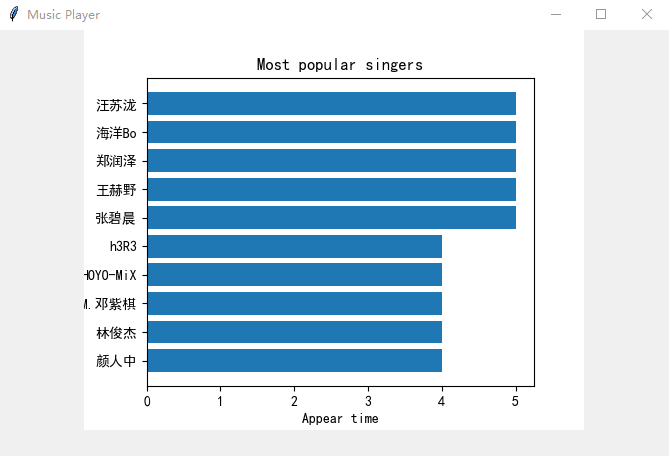
*<Do the same as Usage Scenario 1. Add additional usage scenario n below as needed.>*

*Figure-2*

**

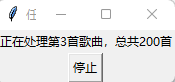
This figure appears after clicking “Hit songs” and displaysthe top ten hit songs from different charts.

*Figure-3*

**

This figure appears after clicking“Top singers” anddisplays the top ten singers in the songs we have crawled.

*Figure-4*



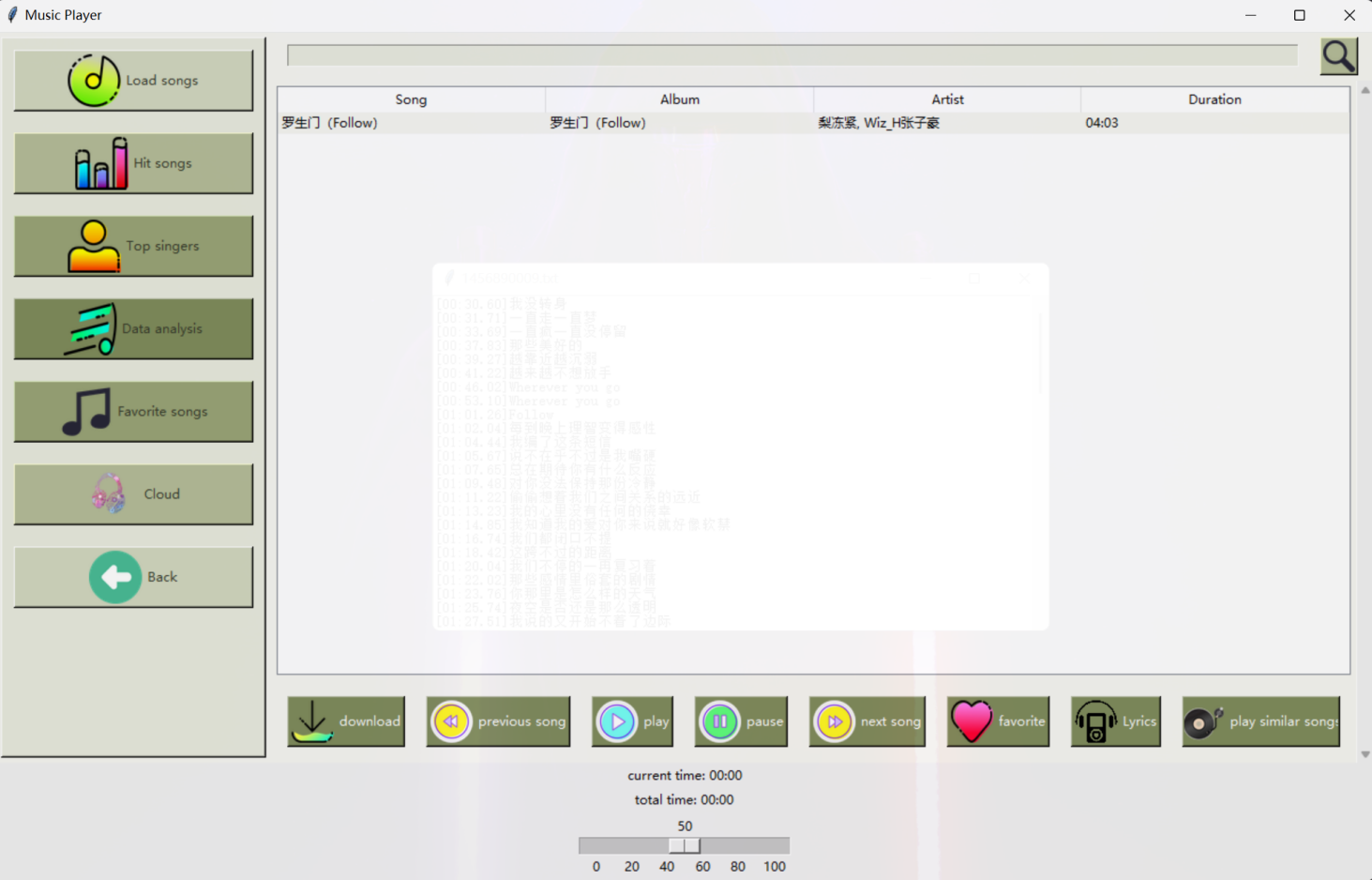
This figure appears after clicking “Data analysis” and it takes some time to processes the lyrics of 200 songs and after that a word cloud is generated to show the frequency of the lyrics of these 200 songs.And “停止” button is to stop the generating and back to the main interface.

*Figure-5*



This figure is the frequency of the lyrics of these 200 songs. Larger words it is more frequent in the song, which can just like the hot words. Therefore if we generate word cloud from the hot song play list, its word cloud can show what keywords is hot and fashion these days.

*Figure-6*



This figure shows the songs in my favorite.

*Figure-7*



Word cloud for song 我想念, just as the image, 我想念 is the largest character because it is the most frequent and key words in lyrics of this song.

This figure shows the frequency of the lyrics of the selected song.

*Figure-8*



This figure shows the lyrics of the selected song with corresponding time.

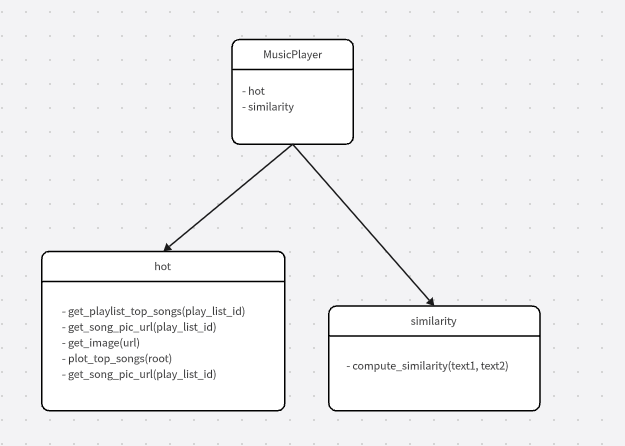
1. Database Design (If Applicable)

*<Describe your database design here. You can put your ER diagram in this section.>*

1. Diagrams

*<C Programming course: Control Flow diagram; Java: Class Diagram>*

Relation of hot.py, similarity.py and MusicPlayer.py



## MusicPlayer.py

## 67994f5b5c5c1ae0073891756131069

bs.py

