



# Everest Radio Defense Document

project name: Radio Everest

Team members: Zhang Ruichen, Wang Yishuo, Lu Shaohua,

Xu Wenbo, Liu Xulei, Li Tianhui

College: School of Mechanical, Electrical and Information Engineering

major: Computer Science, Software Engineering

2023.7. 9

## 1.1Project users

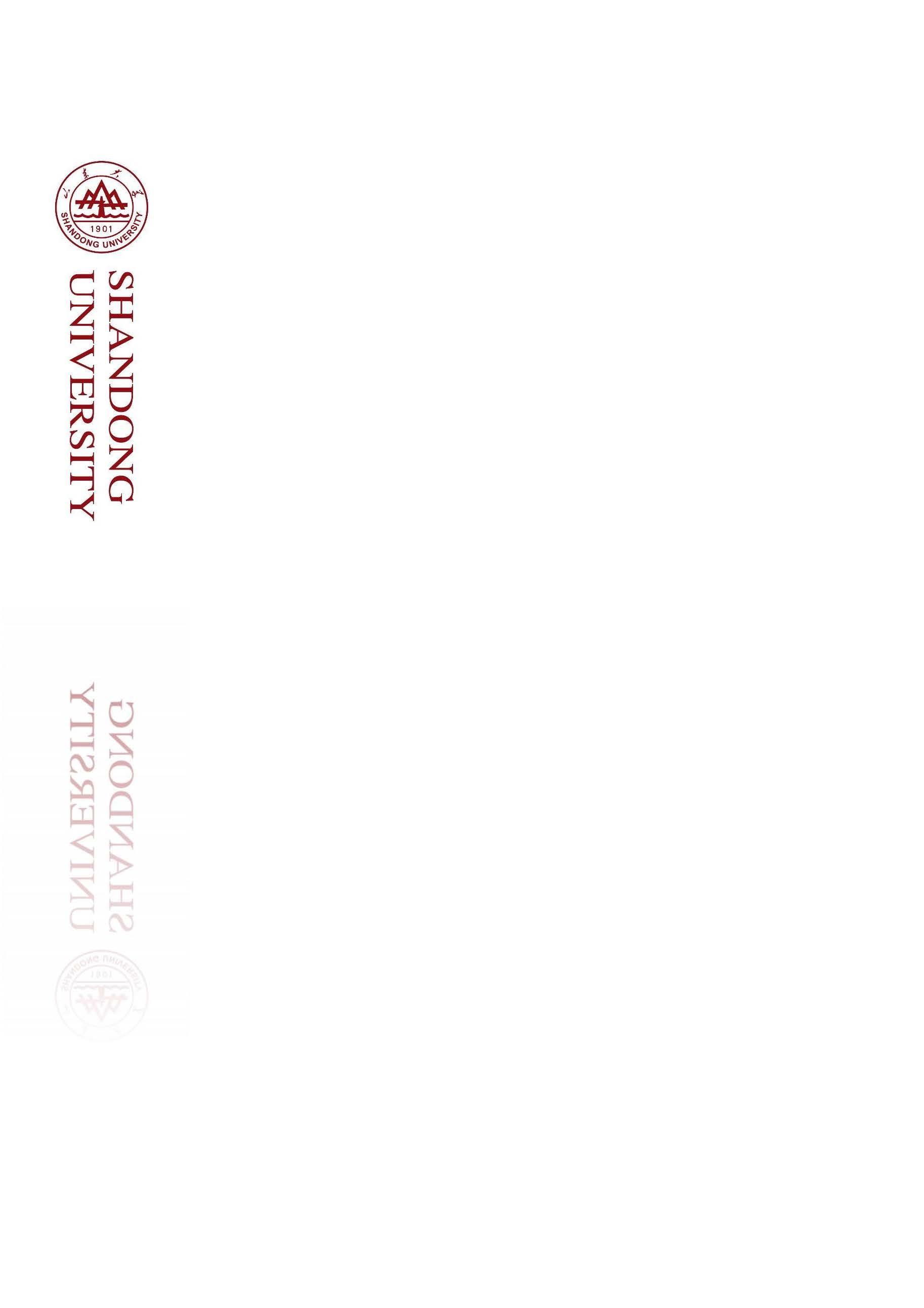
Radio Everest's user base is broad and diverse.

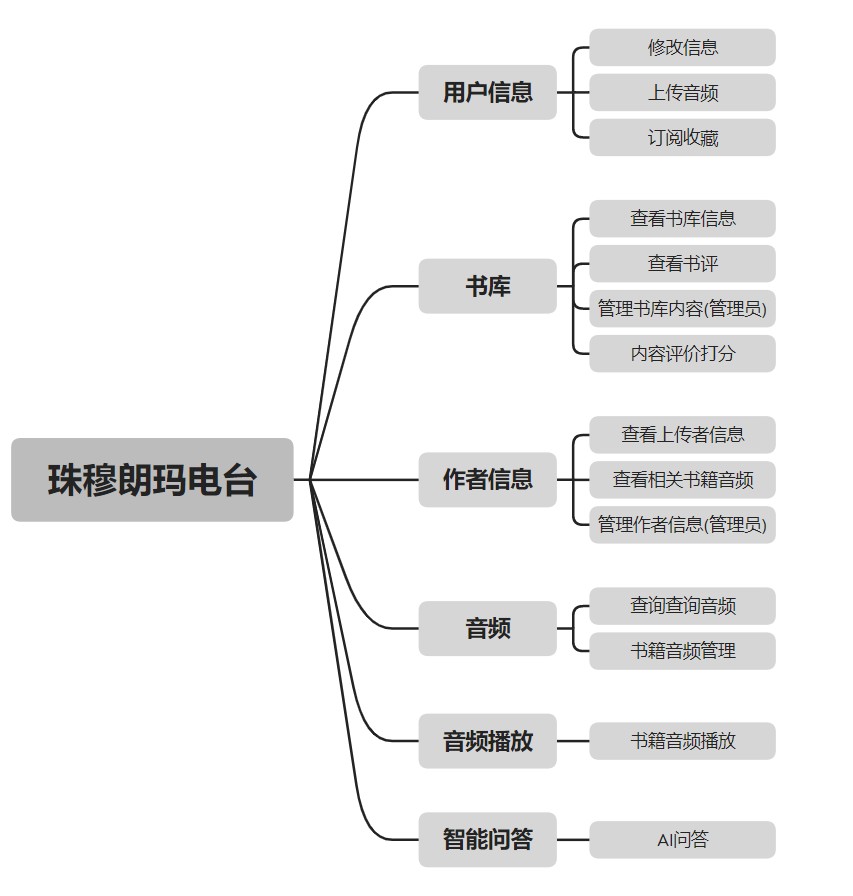
First, literature lovers can enjoy audio interpretations of classic literary works and best-selling novels through Everest Radio . Business people can improve their professional knowledge and management skills by listening to books. Students can use Everest Radio to supplement their learning and reading , and enhance their language expression and comprehension skills.

In addition, commuters listen to audio books on buses and subways to improve the efficiency of fragmented time. The elderly can maintain their reading habit by listening to books and enjoy a relaxing and enjoyable reading experience.

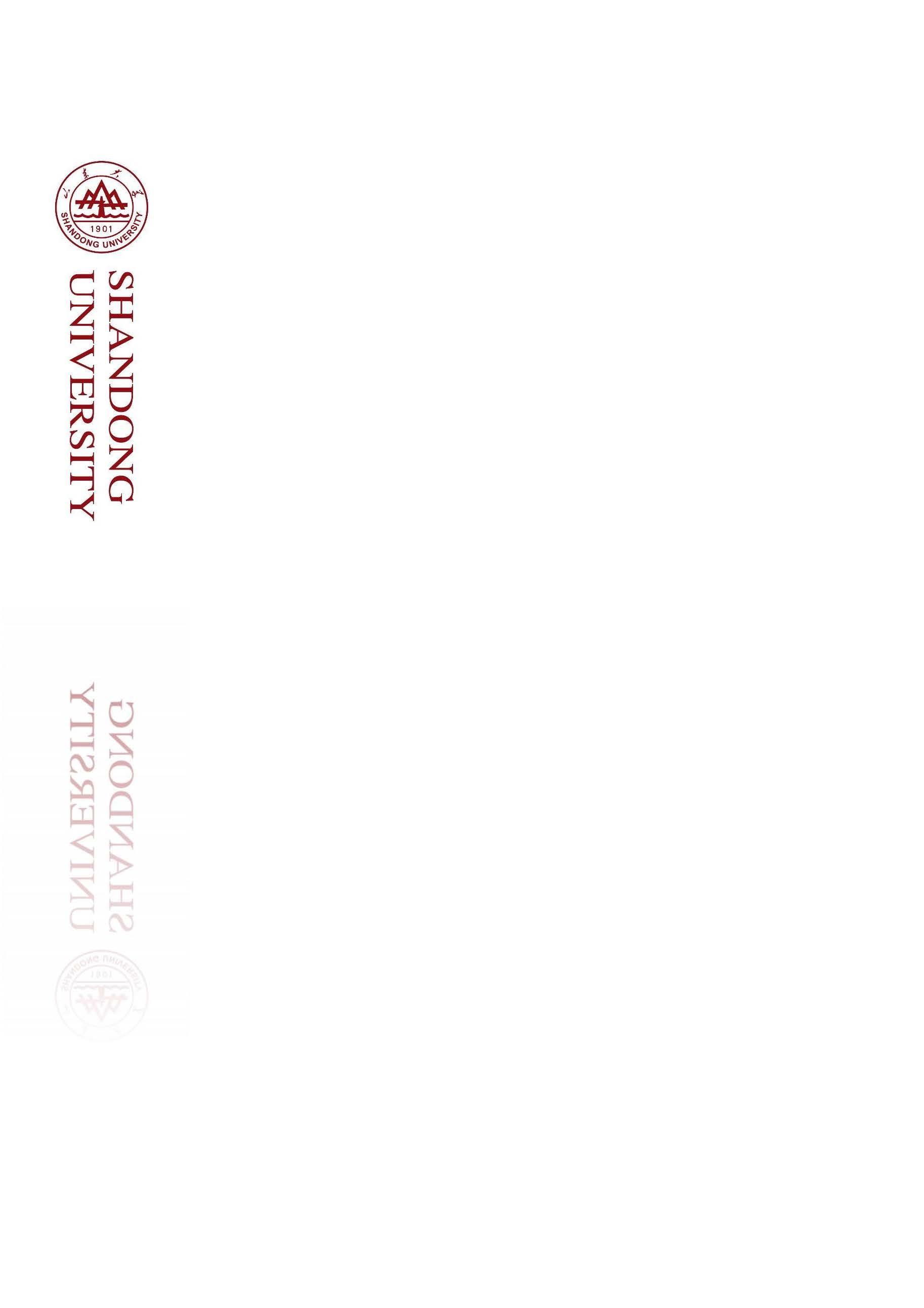
In short, Radio Everest is suitable for anyone who loves reading and pursuing knowledge.

## 1.2 Project functions





1.3 Analysis of each functional requirement



### 1. User information management

1. User registration and login:

- Users can register a new account and provide necessary personal information.

- Registered users can log in to the system using the provided username and password. 2. Enter and modify user information:

- Registered users can fill in and update personal information, such as name, gender, date of birth, contact information, etc.

- Users can upload avatars or personal photos, and edit and change avatars. 3. Password modification:

- Users can change their login password, provide the original password for authentication, and enter a new password for update. 4. Book audio collection and subscription:

- Users are able to browse the book audio library and select to add book audio to the favorites list.

- Users can subscribe to audio updates for specific books to receive the latest chapters or content. 5. Upload audio:

- Users can upload their own audio files, such as reading works, solo music, etc.

- The system should provide suitable audio format support and limit file size and duration. 6. User information display:

- The system can display the user's basic information, such as name, gender, date of birth, etc.

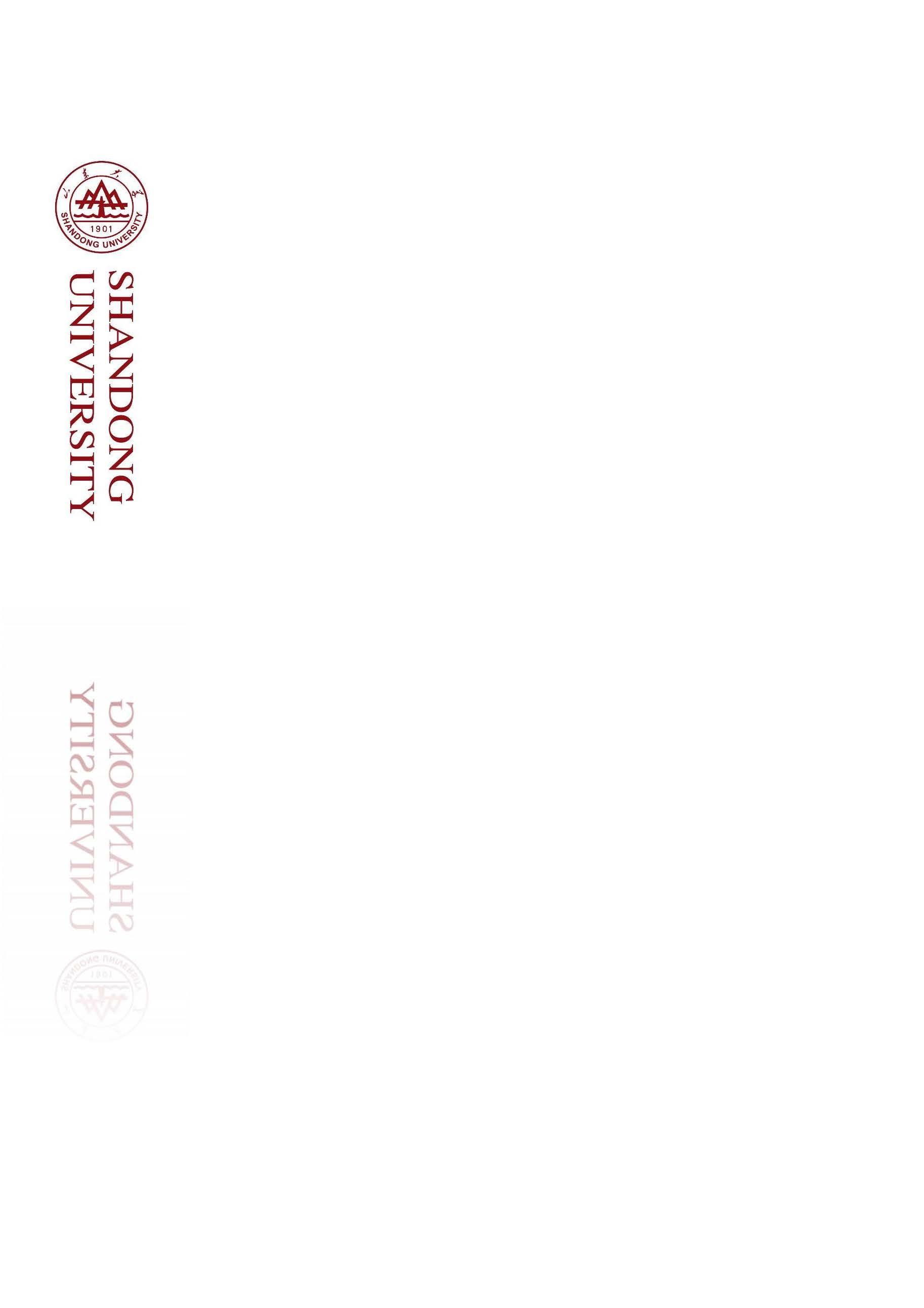
- Users can view and edit their own information on the profile page, and view their uploaded audio and collected book audio.

7. Data backup and recovery:

- The system can regularly back up user data, including user information, collected audio books and uploaded audio, etc.

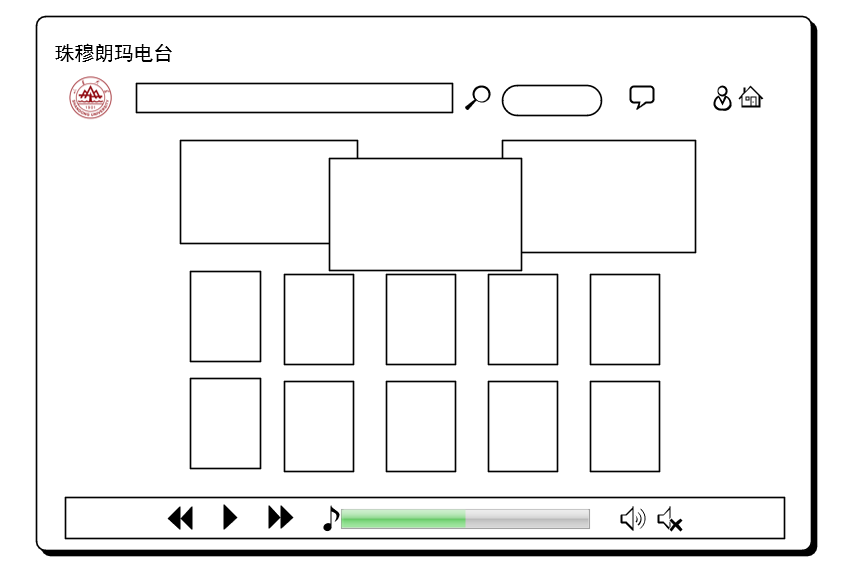
- Provides data recovery capabilities to prevent accidental data loss. 8. Data statistics and reporting:

- The system can generate statistical reports on user information, such as the number of users, statistics on collected books and audio, etc.



- Provide data visualization tools so that users can more intuitively understand changes and trends in user information.

### 1.4 Library



The following is a functional requirements analysis based on the Everest library: 1. Check the library audio information:

- Users can browse audio information in the Everest library, including book title, author, category, description, etc.

- Users can search and browse based on keywords, categories or other filters. 2. View book reviews:

- Users can view other users' reviews and ratings of audios in the library.

- Users can sort book reviews by rating or chronological order to find more valuable reviews. 3. Manage library content ( administrator function ) :

- Administrators can add, edit and delete audio information in the library.

- Administrators can manage audio categories, tags and descriptions so that users can better search and browse. 4. Content evaluation and scoring:

- Users can evaluate and rate the audio in the library.

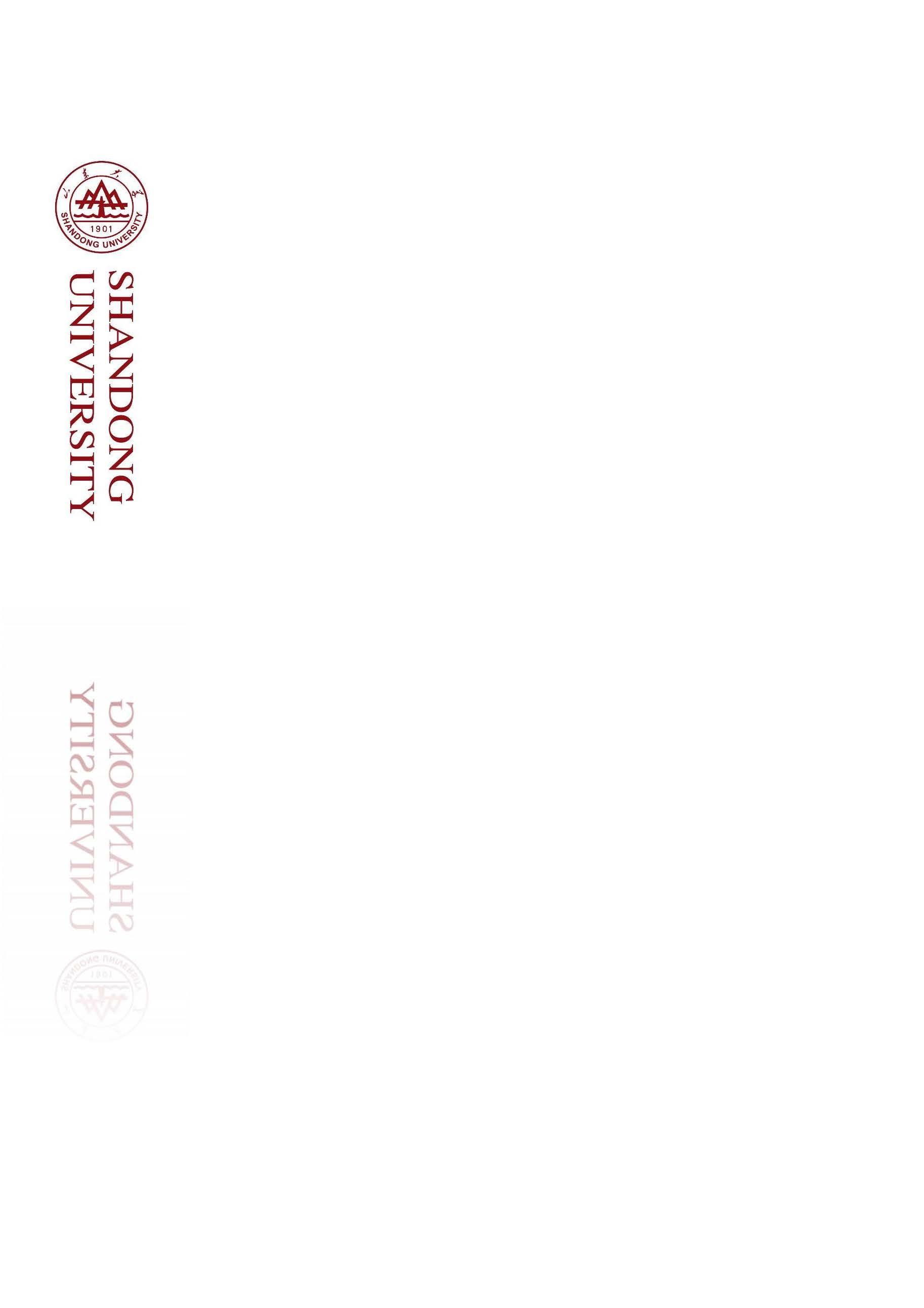
- Users can write book reviews to express their opinions and experiences on audio content.

- Users can provide ratings for each audio for reference by other users. 5. User collections and subscriptions:

- Users can add their favorite audio to a favorites list for review and access at any time.

- Users can subscribe to updates for specific audios to receive the latest chapters or content. 6. Data statistics and reporting ( administrator function ) :

- Administrators can generate statistical reports on the library, including the number of audios, the number of audios in different categories, etc.



- Provide data visualization tools so that administrators can more intuitively understand the content distribution and trends of the library. 7. User rights management ( administrator function ) :

- Administrators can manage user accounts, including reviewing new user registrations, freezing or unfreezing accounts, etc.

- Provide different user roles, such as ordinary users and administrators, to distinguish different operation permissions.

### 1.5 Author

1. Check the uploader information:

- Authors can view the information of users who uploaded their works, including user names, personal profiles, etc.

- Authors can understand the background and interests of users who upload their works. 2. Ratings and reviews:

- Authors can view user ratings and reviews of their works.

- Ratings and reviews help authors understand readers’ feedback and opinions about their work. 3. View related book audio information:

- Authors can view audio information about books related to their works.

- This includes audio that is related to or has similar themes to your own work. 4. Data statistics and reporting:

- Provide statistical reports on the author's works, such as the number of works, average ratings, etc.

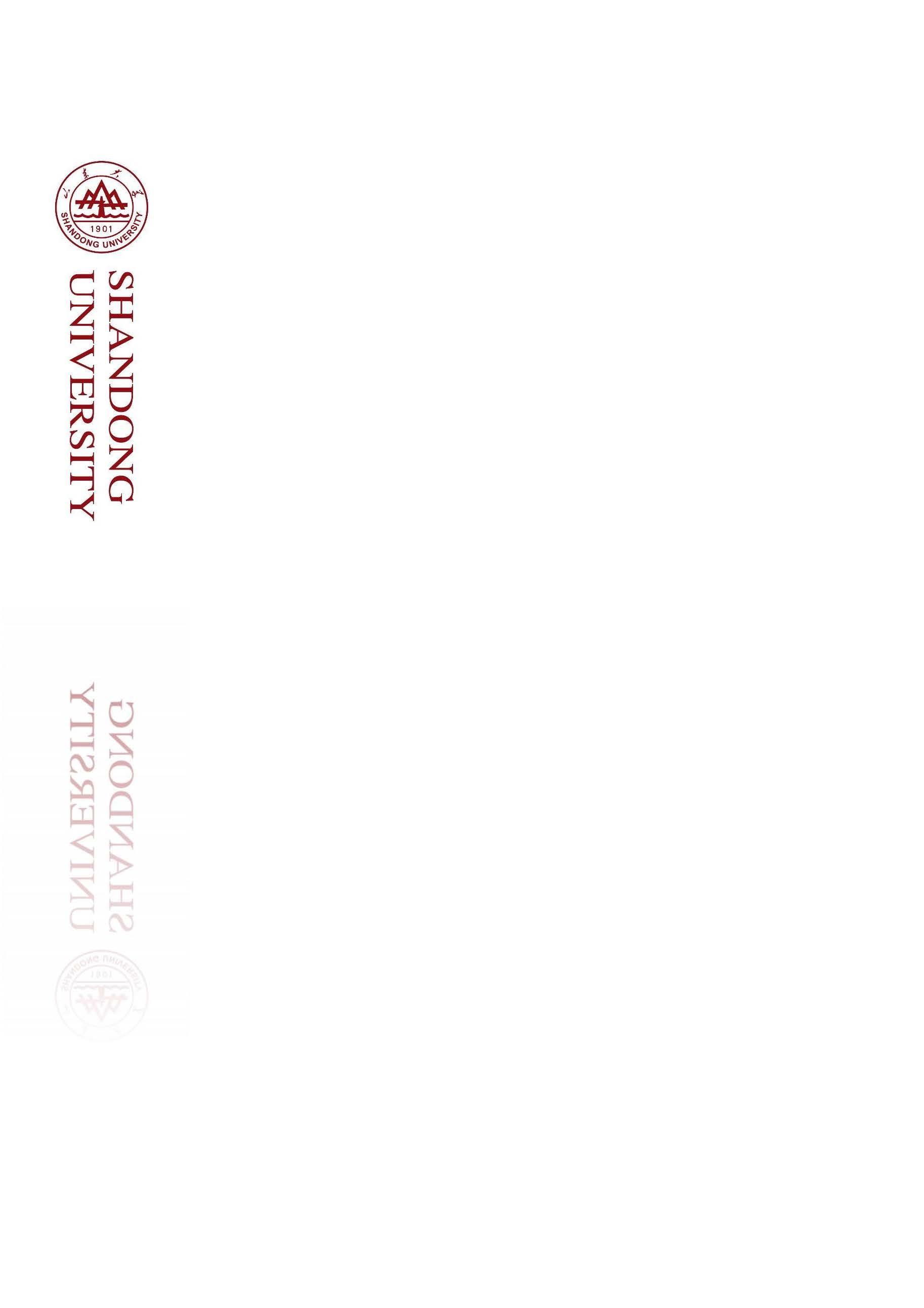
- Provide data visualization tools so that authors can more intuitively understand the performance and popularity of their works. 5. Interact with readers:

- Authors can interact with readers, such as responding to readers' comments and questions.

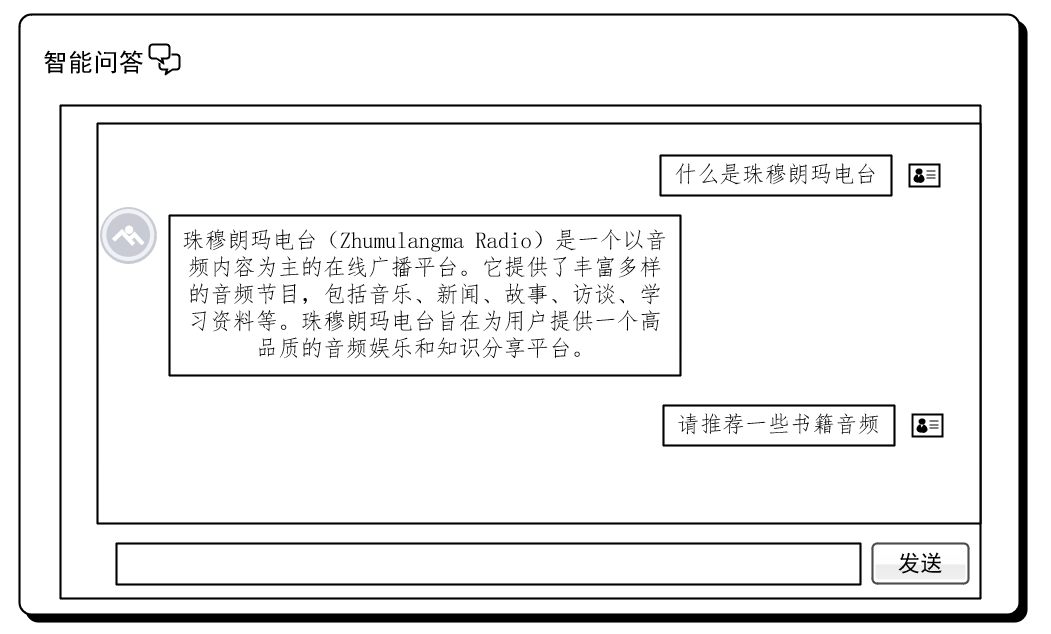
- This helps authors build closer relationships with readers and get more feedback and suggestions. 6. Work management:

- Authors can manage their uploaded works, including editing work information, deleting works, etc.

- The author can update the description, cover image and other information of the work.



### 1.6 Intelligent Q&A



1. Problem identification and understanding:

- The system can identify key information from questions raised by users and conduct semantic understanding and analysis of the questions.

- The system can identify question types, such as factual questions, definitional questions, reasoning questions, etc. 2. Knowledge base and data sources:

- The system needs to have an extensive and reliable knowledge base or data source to provide accurate answers.

- Knowledge base can include structured databases, document collections, information on the Internet, etc. 3. Answer generation and sorting:

- The system is able to generate accurate and clear answers based on the type of question and relevant knowledge.

- The system can rank multiple candidate answers based on their relevance, credibility and authority to present the optimal solution.

4.Multi -language support:

- The system should support multiple languages so that users can ask questions in the language they are familiar with.

- The system is able to understand and handle semantic and cultural differences between different languages. 5. User interaction and feedback:

- The system should provide a user-friendly interface for users to ask questions and view answers.

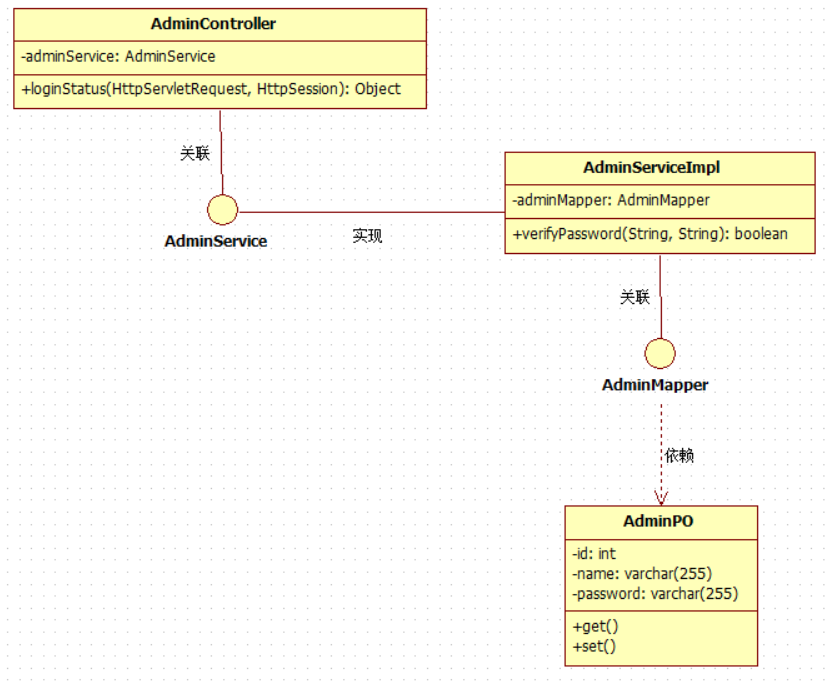
- The system can handle user feedback and inquiries, providing further explanations and details. 6. Real-time updates and maintenance:

- The system needs to regularly update and maintain the knowledge base to maintain the latest information and accuracy of answers.

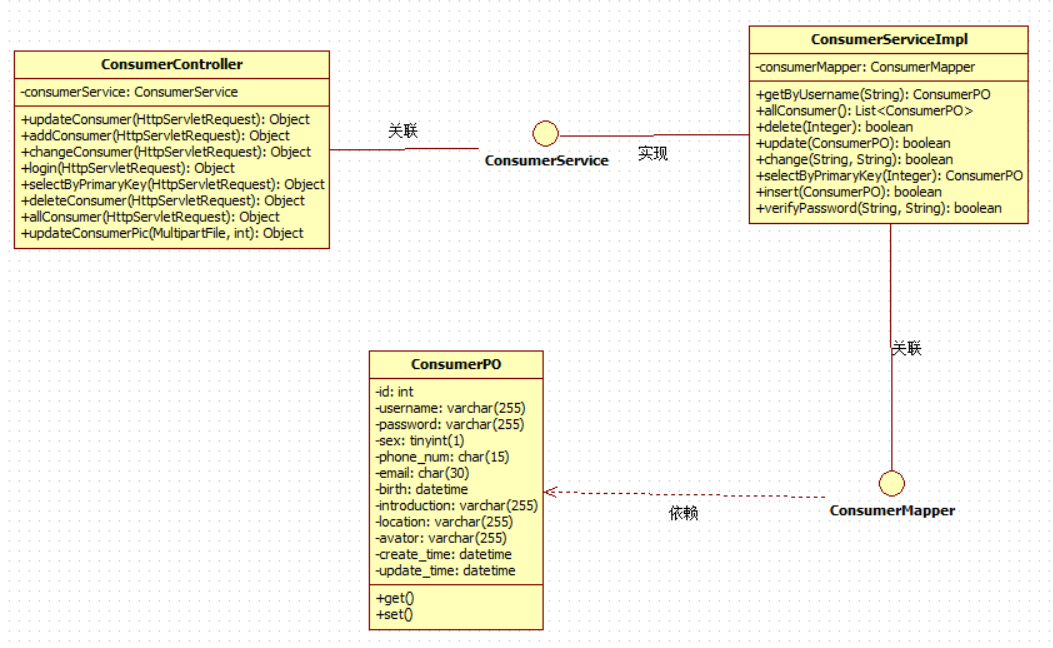
- The system should be able to acquire new data and knowledge from various sources and integrate it into the knowledge base. 7. Privacy and Security:

- The system should ensure user privacy and data security, and follow relevant privacy policies and security standards during data processing and storage.

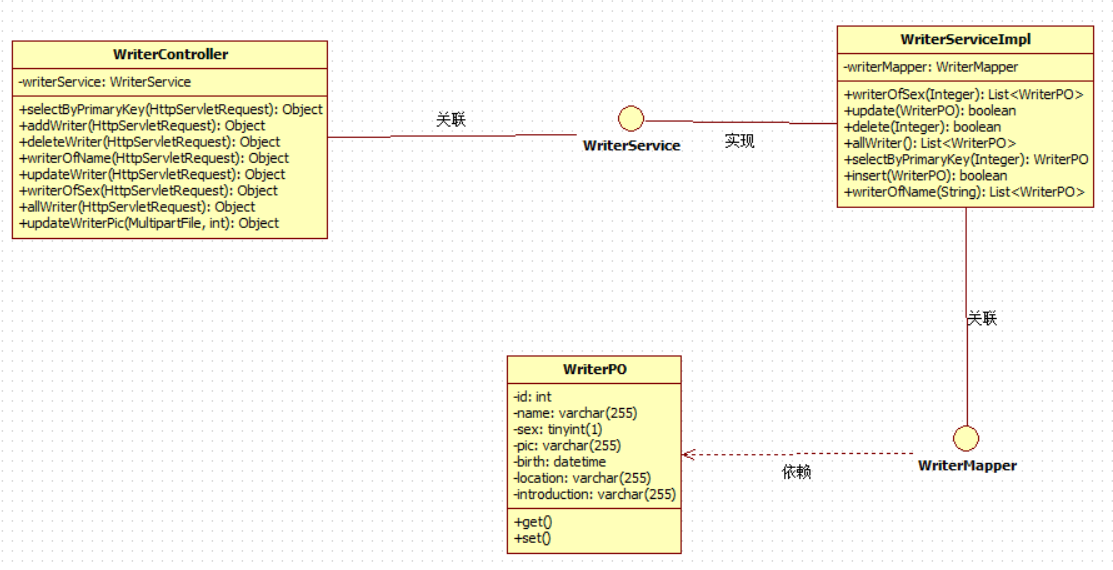
### 2.1 Admin class diagram



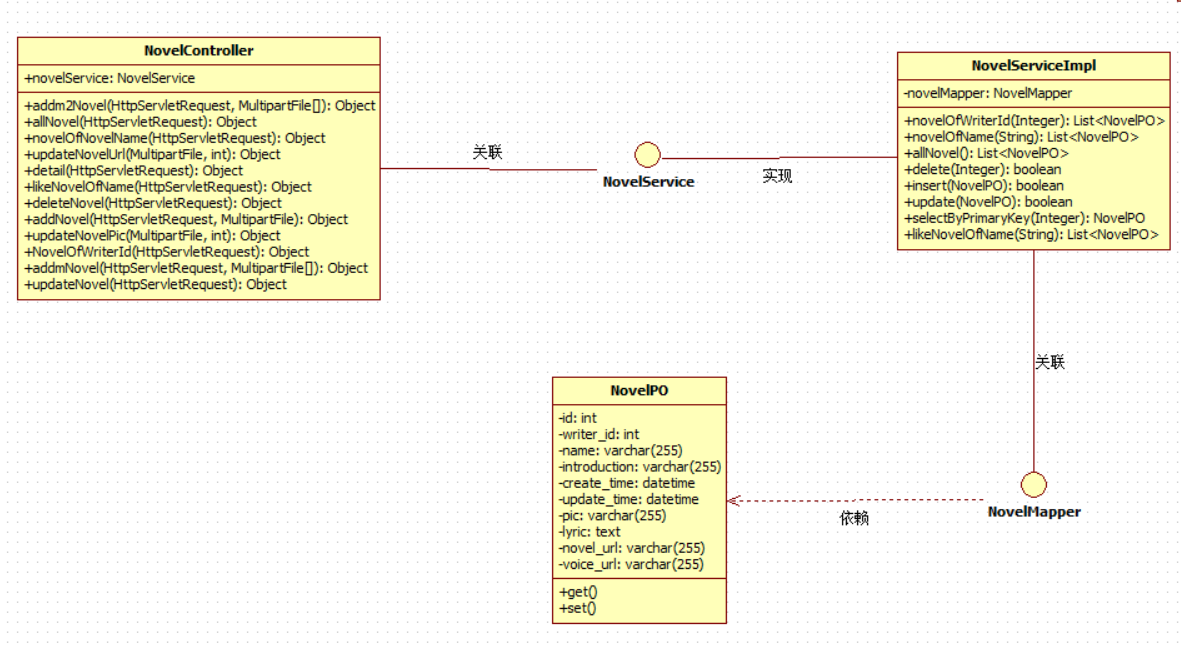
### 2.2 Consumer class diagram



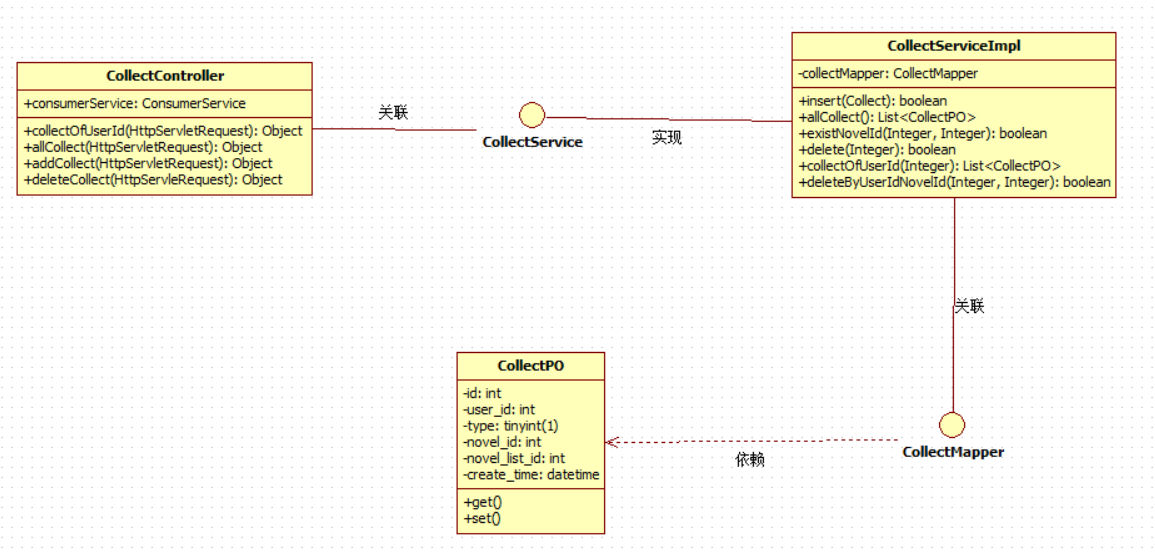
### 2.3 Writer class diagram



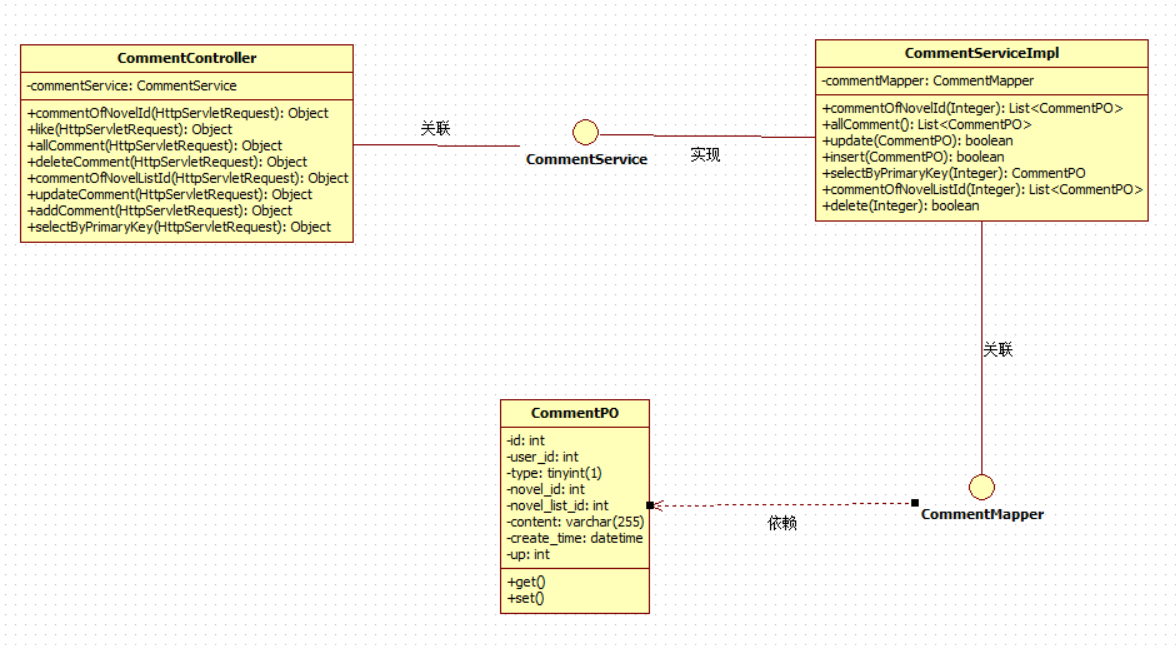
### 2.4 Novel class diagram



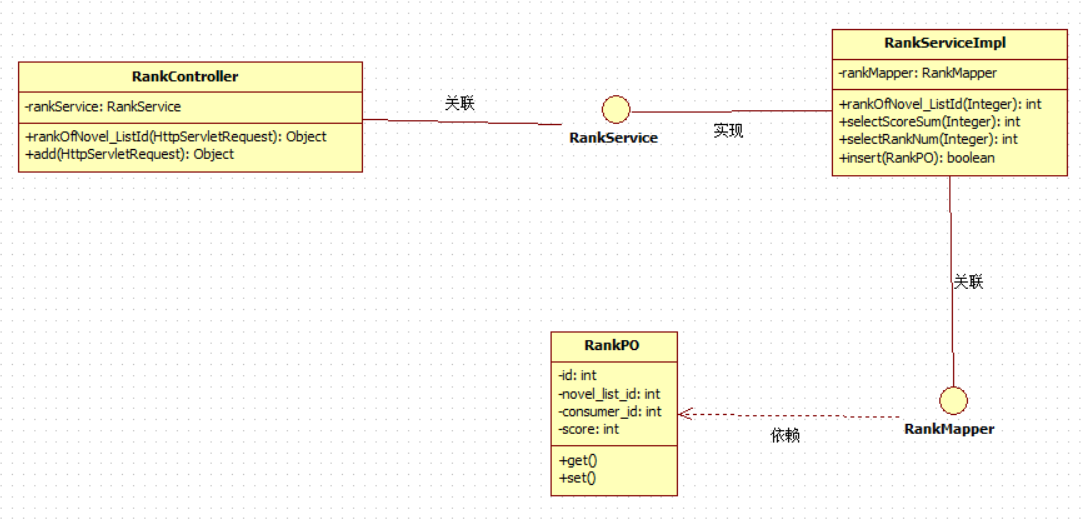
### 2.5 Collect class diagram



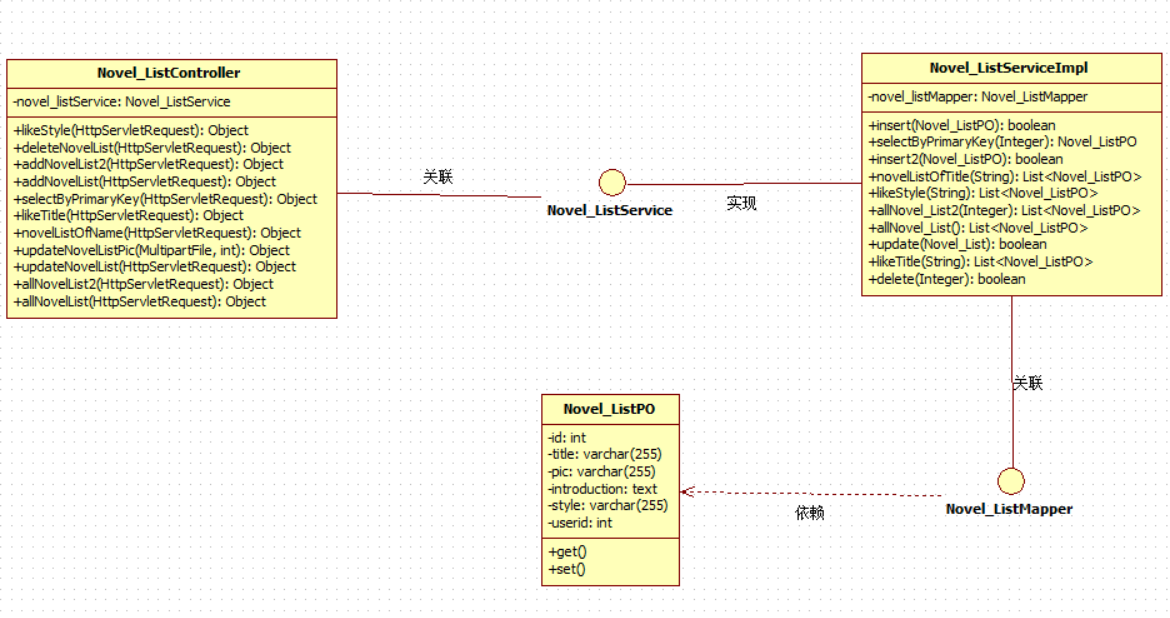
### 2.6 Comment class diagram



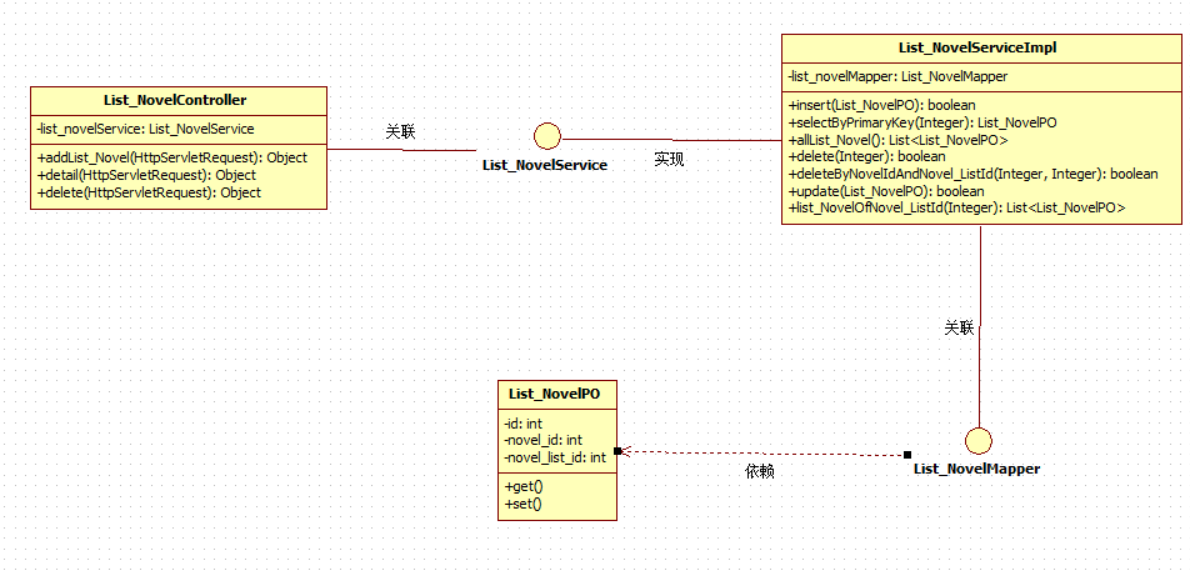
### 2.7 Rank class diagram



### 2.8 Novel\_List class diagram



### 2.9 List\_Novel class diagram



3.1 Administrator information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field type** | **Can it be empty** | **Remark** |
| ID \_ | ID | int | N | Primary key (auto-increment) |
| No. | admin\_jobnum | varchar | N |  |
| Administrator email | admin\_email | varchar | N | Maximum 30 words |
| Administrator level | admin\_level | int | N | Default level 3 |
| Administrator phone number | admin\_tel | varchar | N | Maximum 30 words |
| Administrator avatar | admin\_pic | varchar | N | Store relative path |
| Username | admin\_telname | varchar | N |  |
| password | admin\_telpassword | varchar | N | The password is letters, numbers, and  crossed out mixed composition |

3.2 Collection information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **Remark** |
| Collection ID | ID | int | N | primary key |
| User ID | user\_id | int | N | Foreign key ( consumer : id ) |
| Novel ID | novel\_id | int | N | foreign key ( novel : id ) |
| Collection time | create\_time | datetime | Y |  |
| Bookshelf ID | novel——list\_id | int | N | foreign key ( novellist : id ) |
| type | type | tinyint | N | 0 Novels 1 Bookshelf |

3.3 Comment information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **Remark** |
| Comment ID | ID | int | N | primary key |
| User ID | user\_id | int | N | Foreign key ( consumer :  id ) |
| Bookshelf ID | novel\_id | int | Y | foreign key ( novel\_list : id ) |
| comments | content | varchar | Y | Maximum 255 words |
| Comment time | create\_time | datetime | Y |  |
| Number of comment likes | up | int | Y | Default value 0 |
| Novel ID | novel\_id | int | Y | foreign key ( novel : id ) |
| type | type | tinyint | Y | 0 Novels 1 Bookshelf |

3.4 User information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **Remark** |
| User ID | ID | int | N | primary key |
| Username | username | varchar | N |  |
| password | password | varchar | N |  |
| gender | sex | tinyint | Y | 1 represents male, 0 represents female |
| Mobile phone number | phone\_num | char | Y |  |
| Email | email | char | Y |  |
| Date of birth | birth | datetime | Y | Accurate to year - month - day |
| User signature | introduction | varchar | Y | Maximum 255 words |
| area | location | varchar | Y |  |
| User avatar | avator | varchar | Y | Store relative path |
| Registration time | create\_time | datetime | N |  |
| Change the time | update\_time | datetime | N | Last modified time |

3.5 Writer information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **Remark** |
| Writer ID | writer\_id | int | N | primary key |
| Author name | writer\_name | varchar | N |  |
| gender | writer\_sex | tinyint | Y |  |
| Writer avatar | writer\_pic | varchar | Y | relative path |
| Date of birth | writer\_birth | datetime | Y | Accurate to year - month - day |
| area | writer\_location | varchar | Y |  |
| Introduction | introduction | varchar | Y | Maximum 255 words |

3.6 Novels information

ID ID

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名** | **字段代码** | **字段类型** | **可否为空** | **备注** |
| ⼩说ID | music\_id | int | N | 主键 |
| 作家ID | writer\_id | int | N | 外键（writer：  id） |
| 发⾏时间 | create\_time | datetime | N | ⼩说发⾏时间 |
| ⼩说图⽚ | pic | varchar | Y | 相对路径 |
| 内容 | content | text | Y |  |
| ⼩说⽂件 | music\_file\_src | varchar | N | 相对路径 |
| ⼩说作家 | melody\_author\_id | int | N | 外键 |
| 内容作家 | lyric\_author\_id | int | Y | 外键 |

3.7 Bookshelf information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **Remark** |
| Bookshelf ID | list\_id | int | N | primary key |
| bookshelf name | list\_name | varchar | N | Maximum 30 words |
| bookshelf cover | list\_pic | varchar | Y | relative path |
| Bookshelf Introduction | list\_title | varchar | Y | Maximum 300 words |
| Grid label | labal | varchar | Y | Maximum 30 words |
| Bookshelf novel id | music\_id | int | Y | foreign key |

3.8 Rating information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **Remark** |
| Bookshelf Rating ID | ID | int | N | primary key |
| Bookshelf ID | novel\_list\_id | int | N | foreign key ( novel\_list : id ) |
| User ID | user\_id | int | N | Foreign key ( consumer : id ) |
| Score | score | smallint | Y | 1-5 points |

3.9 Bookshelf Novels List

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **Remark** |
| Bookshelf Novels ID | ID | int | N | primary key |
| Novel ID | novel\_id | int | N | foreign key ( novel : id ) |
| Bookshelf ID | novel\_list\_id | int | N | foreign key ( novel\_list : id ) |

|  |  |  |
| --- | --- | --- |
| 4. Simulated data |  | |
| 4.1 \_ Administrator information  **Field name field code field type** | **Can it be empty** | **simulated data** |
| Identity ID id int | N | 1 |
| Job ID admin\_jobnum varchar | N | 123 |
| Administrator email admin\_email varchar | N | [123@163.com](mailto:123@163.com) |
| Administrator level admin\_level int | N | 3 |
| Administrator phone number admin\_tel varchar | N | 18661585821 |
| Administrator avatar admin\_pic varchar | N | Store relative path |
| Username admin\_telname varchar | N | admin1 |
| passwordadmin\_telpassword varchar \_ | N | admin1 |

4.2 Collection information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **simulated data** |
| Collection ID | ID | int | N | 1 |
| User ID | user\_id | int | N | 1 |
| Novel ID | novel\_id | int | Y | 1 |
| Collection time | create\_time | datetime | Y | 2023/7/5 |
| Bookshelf ID | novel——list\_id | int | Y | null |
| type | type | tinyint | N | 0 |

4.3 Comment information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **simulated data** |
| Comment ID | ID | int | N | 1 |
| User ID | user\_id | int | N | 1 |
| Bookshelf ID | novel\_id | int | Y | null |
| comments | content | varchar | Y | This is a test comment |
| Comment time | create\_time | datetime | Y | 2023/7/5 |
| Number of comment likes | up | int | Y | Default value 0 |
| Novel ID | novel\_id | int | Y | 1 |
| type | type | tinyint | Y | 0 |

4.4 User information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **simulated data** |
| User ID | ID | int | N | 1 |
| Username | username | varchar | N | user1 |
| password | password | varchar | N | user1 |
| gender | sex | tinyint | Y | 1 |
| Mobile phone number | phone\_num | char | Y | 18661585821 |
| Email | email | char | Y | [123@163.com](mailto:123@163.com) |
| Date of birth | birth | datetime | Y | 2002/12/10 |
| User signature | introduction | varchar | Y | This is a user signature |
| area | location | varchar | Y | Weihai City, Shandong Province |
| User avatar | avator | varchar | Y | Store relative path |
| Registration time | create\_time | datetime | N | 2023/7/5 |
| Change the time | update\_time | datetime | N | 2023/7/5 |

4.5 Writer information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **simulated data** |
| Writer ID | writer\_id | int | N | 1 |
| Author name | writer\_name | varchar | N | Writer A |
| gender | writer\_sex | tinyint | Y | 1 |
| Writer avatar | writer\_pic | varchar | Y | relative path |
| Date of birth | writer\_birth | datetime | Y | 2002/12/10 |
| area | writer\_location | varchar | Y | Weihai City, Shandong Province |
| Introduction | introduction | varchar | Y | This is writer A Introduction |

4.6 Novels information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **simulated data** |
| Novel ID | music\_id | int | N | 1 |
| Writer ID | writer\_id | int | N | 1 |
| Release time | create\_time | datetime | N | 2023/7/5 |
| Storytelling pictures | pic | varchar | Y | relative path |
| content | content | text | Y | This is the content of the novel |
| Novels files | music\_file\_src | varchar | N | relative path |
| Novel Writer ID | melody\_author\_id | int | N | 1 |

4.7 Bookshelf information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **simulated data** |
| Bookshelf ID | list\_id | int | N | 1 |
| bookshelf name | list\_name | varchar | N | Urban sentiment |
| bookshelf cover | list\_pic | varchar | Y | relative path |
| Bookshelf Introduction | list\_title | varchar | Y | This is from the urban romance novels bookshelf |
| Grid label | labal | varchar | Y | Urban sentiment |
| Bookshelf novel id | music\_id | int | Y | 1 |

4.8 Rating information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **simulated data** |
| Bookshelf Rating ID | ID | int | N | 1 |
| Bookshelf ID | novel\_list\_id | int | N | 1 |
| User ID | user\_id | int | N | 1 |
| Score | score | smallint | Y | 1 |

4.9 Bookshelf Novels List

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field name** | **Field code** | **Field Type** | **Can it be empty** | **simulated data** |
| Bookshelf Novels ID | ID | int | N | 1 |
| Novel ID | novel\_id | int | N | 1 |
| Bookshelf ID | novel\_list\_id | int | N | 1 |

# 5. Code display

5.1 Administrator information

CREATE TABLE `admin` (

`id` int NOT NULL AUTO\_INCREMENT COMMENT '主键',

`name` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT '账号',

`password` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT '密码',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8mb3 ROW\_FORMAT=DYNAMIC COMMENT='管理员';

5.2 Collection information

CREATE TABLE `collect` (

`id` int NOT NULL AUTO\_INCREMENT COMMENT '主键',

`user\_id` int DEFAULT NULL COMMENT '⽤户id',

`type` tinyint(1) DEFAULT NULL COMMENT '收藏类型（0⼩说1书架）',

`novel\_id` int DEFAULT NULL COMMENT '⼩说id',

`novel\_list\_id` int DEFAULT NULL COMMENT '书架id',

`create\_time` datetime DEFAULT NULL COMMENT '收藏时间',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=62 DEFAULT CHARSET=utf8mb3 ROW\_FORMAT=DYNAMIC COMMENT='收藏';

5.3 Comment information

CREATE TABLE `comment` (

`id` int NOT NULL AUTO\_INCREMENT COMMENT ' primary key ',

`user\_id` int DEFAULT NULL COMMENT ' userid ',

`type` tinyint(1 ) DEFAULT NULL COMMENT ' Comment type ( 0 novels 1 bookshelf) ',

`novel\_id` int DEFAULT NULL COMMENT ' ⼼ id',

`novel\_list\_id` int DEFAULT NULL COMMENT ' bookshelf id',

`content` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Comment content ',

`create\_time` datetime DEFAULT NULL COMMENT ' Collection time ',

`up` int DEFAULT '0 ' COMMENT ' Number of likes ',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=62 DEFAULT CHARSET=utf8mb3 ROW\_FORMAT=DYNAMIC COMMENT=' Comment ';

5.4 User information

CREATE TABLE `consumer` (

`id` int NOT NULL AUTO\_INCREMENT COMMENT ' primary key ',

`username` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' account ',

`password` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Password ',

`sex` tinyint(1 ) DEFAULT NULL COMMENT ' Gender ( 1Male0 \_

⼥) ',

`phone\_num` char(15) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Telephone ',

`email` char(30) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Mailbox ',

`birth` datetime DEFAULT NULL COMMENT ' Birth date ',

`introduction` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Signature ',

`location` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Region ',

`avator` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' avatar ',

`create\_time` datetime DEFAULT NULL COMMENT ' When created

room ',

`update\_time` datetime DEFAULT NULL COMMENT ' update time ',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=44 DEFAULT CHARSET=utf8mb3 ROW\_FORMAT=DYNAMIC COMMENT=' front-end user ';

5.5 Writer information

CREATE TABLE `writer` (

`id` int NOT NULL AUTO\_INCREMENT COMMENT ' primary key ',

`name` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' name ',

`sex` tinyint(1 ) DEFAULT NULL COMMENT ' Gender ( 0 ⼼ female 1 male 2

Combination 3 is unknown) ',

`pic` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' avatar ',

`birth` datetime DEFAULT NULL COMMENT ' Birth date ',

`location` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Region ',

`introduction` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Introduction ',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=1002 DEFAULT CHARSET=utf8mb3 ROW\_FORMAT=DYNAMIC COMMENT=' writer ';

5.6 Novels information

CREATE TABLE `novel` (

`id` int NOT NULL AUTO\_INCREMENT COMMENT ' primary key ',

`writer\_id` int DEFAULT NULL COMMENT ' writer id',

`name` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Book title ',

`introduction` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Introduction ',

`create\_time` datetime DEFAULT NULL COMMENT ' Creation time ',

`update\_time` datetime DEFAULT NULL COMMENT ' update time ',

`pic` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Seal ',

`lyric` text CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci COMMENT ' content ',

`novel\_url` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Book address ',

`voice\_url` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' audio address ',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=137 DEFAULT CHARSET=utf8mb3 ROW\_FORMAT=DYNAMIC COMMENT=' Novel ';

5.7 Bookshelf information

CREATE TABLE `novel\_list` (

`id` int NOT NULL AUTO\_INCREMENT COMMENT ' primary key ',

`title` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Title ',

`pic` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' Novel cover ',

`introduction` text CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci COMMENT ' Introduction ',

`style` varchar(255) CHARACTER SET utf8mb3 COLLATE utf8mb3\_general\_ci DEFAULT NULL COMMENT ' grid ',

`userid` int DEFAULT NULL, PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=92 DEFAULT CHARSET=utf8mb3 ROW\_FORMAT=DYNAMIC COMMENT=' bookshelf ';

5.8 Rating information

CREATE TABLE `rank` (

`id` int NOT NULL AUTO\_INCREMENT COMMENT '主键',

`novel\_list\_id` int NOT NULL COMMENT '书架id',

`consumer\_id` int NOT NULL COMMENT '⽤户id',

`score` int DEFAULT NULL COMMENT '评分',

PRIMARY KEY (`id`) USING BTREE,

UNIQUE KEY `consumer\_id` (`novel\_list\_id`,`consumer\_id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=43 DEFAULT CHARSET=utf8mb3 ROW\_FORMAT=DYNAMIC COMMENT='评价';

5.9 Bookshelf Novels List

CREATE TABLE `list\_novel` (

`id` int NOT NULL AUTO\_INCREMENT COMMENT '主键',

`novel\_id` int DEFAULT NULL COMMENT '⼩说id',

`novel\_list\_id` int DEFAULT NULL COMMENT '书架id',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=242 DEFAULT CHARSET=utf8mb3 ROW\_FORMAT=DYNAMIC COMMENT='书架包含书

⽬列表';

# 6. Database table diagram

