**ASSIGNMENT 2 FRONT SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number and title** | Unit 9: Software Development Life Cycle | | |
| **Submission date** |  | **Date Received 1st submission** |  |
| **Re-submission Date** |  | **Date Received 2nd submission** |  |
| **Student Name** | Phan Minh Tri | **Student ID** | GCC18015 |
| **Class** | GCC0701 | **Assessor name** | Nguyen Thai Nghe |
| **Student declaration**  I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice. | | | |
|  |  | **Student’s signature** |  |

**Grading grid**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P5 | P6 | P7 | M3 | M4 | M5 | M6 | D3 | D4 |
|  |  |  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **❒ Summative Feedback: ❒ Resubmission Feedback:** | | |
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Internal Verifier’s Comments:** | | |
| **Signature & Date:** | | |

**ASSIGNMENT 2 BRIEF**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number** | Unit 9: Software Development Life Cycle | | |
| **Assignment title** | Undertake a software development lifecycle | | |
| **Academic Year** | 2019 – 2020 | | |
| **Unit Tutor** | LE Minh Duc | | |
| **Issue date** |  | **Submission date** |  |
| **Name and date** |  | | |

|  |  |
| --- | --- |
| **Submission Format:** | |
| *Format:* | The submission is in the form of 1 document.  You **must** use the *Times font* with *12pt size*, turn on *page numbering*; set *line spacing to 1.3* and *margins* to be as follows: left = 1.25cm, right = 1cm, top = 1cm, bottom = 1cm. Citation and references must follow the Harvard referencing style.  **Word limit**: 3000 words (excluding figures and references). Submissions that exceed this limit will be rejected. |
| *Submission:* | You **must** submit the assignment **by the due date** and follow the submission method specified by the Tutor. The submission form is **soft copy**, which is to be uploaded to the following URL: [http://cms.greenwich.edu.vn](http://cms.greenwich.edu.vn/). |
| *Note:* | Your assignment *must* be your own work, and not copied by or from another student or from other sources, such as book etc. If you use ideas, quotes or data (such as diagrams) from books, journals or other sources, you must reference the sources, using the Harvard style. Make sure that you know how to reference properly and that you understand the plagiarism guidelines. **Plagiarism is a very serious offence**, which will result in a failing grade. |
| **Unit Learning Outcomes:** | |
| **LO3** Undertake a software development lifecycle.  **LO4** Discuss the suitability of software behavioural design techniques. | |
| **Assignment Brief and Guidance:** | |
| **Tasks**  At this stage, you have convinced Tune Source to select your project for development. Complete the following tasks to analyse and design the software.  **Task 1 – Analysis (1)**   1. (P5.a) Identify the stakeholders, their roles and interests in the case study.   Review the requirement definition of the project. Clearly indicate which stakeholder(s) provide what requirements.  *Word limit: 150 - 200*  Identify FRs and NFRs of TuneSource Project  Discuss the relationships between the FRs and NFRs.  *Word limit: 300 – 400 words*   1. (P5.b) Discuss the technique(s) you would use to obtain the requirements.   If needed, you may state suitable additional assumptions about the project in order to justify the technique(s) that you choose.  ***Techniques: JAD, Interview, Observation, etc …***  ***Demonstrate how to collect requirements based on chosen technique***  *Word limit: 700 - 1000*   1. (M3) Discuss how you would trace these requirements throughout the project.   *Word limit: 400 – 500 words*  **Task 2 – Analysis (2)**  (P6) Analyse the requirements that you identified in Task 1 using a combination of structural and behavioural modelling techniques that you have learnt.  *Scope*: you only need to construct following items for the system. You will have to include   * Use Case Diagram for the whole system * Use Case specification for 2 Use cases * Context Diagram for the whole system * Data Flow Diagram – Level 0 for the whole system * ERD for the whole system   *Worl limit: 1000 – 1200 words*  **Task 3** **– Design**  Based on the analysis result, discuss how you would conduct the design phase:   1. (P7) Discuss how the user and software requirements are addressed in the design phase.    * You will explain how Mockup and Wireframe are used in the project. You should include some of the mockup or wireframe (at least 5) design of the TuneSource project to justify that it matches users’ requirements    * You will explain which architecture (client – server, n-tier, microservices, etc.) is suitable for the project with clear illustrations and why    * Then you will address which solution stack could be suitable to implement the project with clear explanations 2. (M5) Discuss how activity diagram and pseudocode are used to specify the software behaviour. 3. (M6) Discuss how UML state machine can be used to specify the software behaviour. Differentiate between FSM And extended FSM using the case study. 4. (D4) Discuss how the data-driven approach improves the reliability and effectiveness of software.   *Word limit: 400 - 1500*  **Task 4 – Software quality management**   1. (M4.a) Discuss two software quality attributes that are applicable to the project. 2. (M4.b) Discuss two quality assurance techniques that can help improve the software quality in the project. 3. (D3) Discuss how the design techniques and approaches that you have used can help improve the software quality.   *Word limit: 400 - 1500* | |

|  |  |  |  |
| --- | --- | --- | --- |
| Learning Outcomes and Assessment Criteria | | | |
| Pass | Merit | Distinction | |
| **LO3 Undertake a software development lifecycle** | | | **D3** Critically evaluate  how the use of the  function design paradigm  in the software  development lifecycle can  improve software quality. |
| **P5** Undertake a software  investigation to meet a  business need.  **P6** Use appropriate  software analysis  tools/techniques to carry  out a software  investigation and create  supporting  documentation. | **M3** Analyse how software  requirements can be  traced throughout the  software lifecycle.  **M4** Discuss two  approaches to improving  software quality. |  | |
| **LO4 Discuss the suitability of software behavioural**  **design techniques** | | | **D4** Present justifications  of how data driven  software can improve the  reliability and  effectiveness of software. |
| **P7** Explain how user and  software requirements  have been addressed. | **M5** Suggest two software  behavioural specification  methods and illustrate  their use with an  example.  **M6** Differentiate between  a finite state machine  (FSM) and an extended-  FSM, providing an  application for both. |  | |

Table of Contents

# **P5**. Undertake a software investigation to meet a business need……………………………………………………………………………………………………………………………………….9.

# P7 Explain how user and software requirements have been addressed…………………………24

[References 30](#_Toc63893557)

# **P5. Undertake a software investigation to meet a business need.**

* 1. **Plan of the Tune Source Project**
* We carry out this project in compliance with the needs of our stakeholders. The stakeholders include an on-demand music retailer and a partner with whom we build online music systems. In the method, we implement protection and accessibility specifications.
  1. **The function of interested parties**

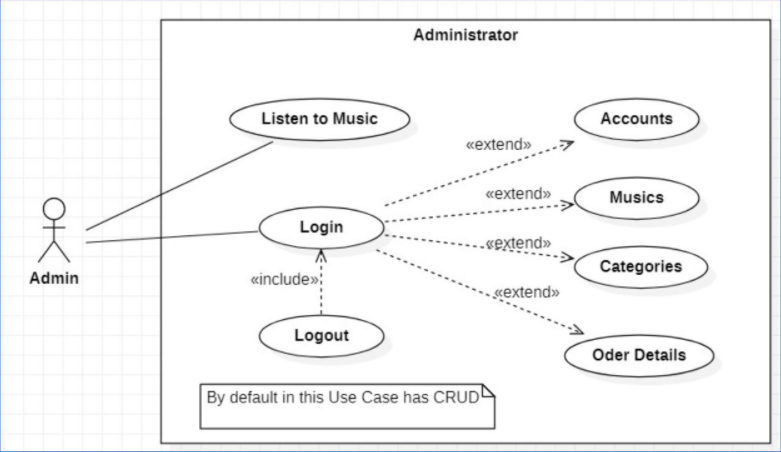
Stakeholders shall include:

* On-demand music supplier: to supply music styles at the request of the consumer.
* Customer: have functional specifications, helpful interface functionality.
  1. **Functionality:**
* Login/ Signup.
* Search for favorite tracks, artists, albums, etc.
* Create and customize playlists.
* Control the music with usual controls such as play, pause, skip, shuffle, repeat, etc.
* Adjust the volume using the volume bar.
* Update user details / Logout.
* Note: All of the management function has included CRUD.
  1. **Non-Functional requirement**
* System features such as stability, reliability, performance, maintenance, scalability, and usability are defined by non-functional specifications (NFRs). Various backlog profiles act as constraints or limits on the nature of the device. We ensure the usability and performance of the whole device. Failure to meet all of these criteria means that the framework does not meet the internal needs of the organization, the consumer, or the industry, or that the regulatory authorities or consumers do not agree with the mandatory requirements.
* Our services are being run and stored on cloud storage. We suggest the collaborators use Microsoft Azure cloud systems. It guarantees the protection and operability of the system.
  1. **Security**
* Secure SQL Injection: SQL Injection is a popular web attack formula based on those managing websites, the reason is that the material is mostly not officially coded and hacking tools take advantage of running vulnerabilities destructive. The most prevalent avoidance of SQL injection consists of two parts. The first is to upgrade and patch all servers, services, and applications on a regular basis, then create and use the source code well and test the source code of the website in order not to allow SQL commands to existing signs that are irregular. (mona.media, n.d.)
* Approve / validate server-side website security
* Validation can often be performed both on the browser and on the server-side, the browser will detect basic errors, such as the appropriate fields that are not left vacant or where you insert text in the numeric fields. However, they can also be overlooked, and you must be careful to verify these assertions as if they did not do so may result in a scenario where the malicious code being introduced into the database could trigger the effects. Unexpected effects on your website. (mona.media, n.d.)
  1. **Observation**
* Several outlets from trendy music sites, including Zing MP3 and Spotify, SoundCloud were consulted. The main content will be quickly reviewed in Table 1.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Product | Cost | Quality  (Song) | Playlist | Music preview | Management | User Experience |
| Zing MP3 | Medium | Low | Yes | Yes | No | Normal |
| Spotify | High | High | Yes | Yes | Yes | Good |
| SoundCloud | Medium | Low | Yes | Yes | No | Normal |
| Our product | Medium | Medium | Yes | Yes | Yes | Good |

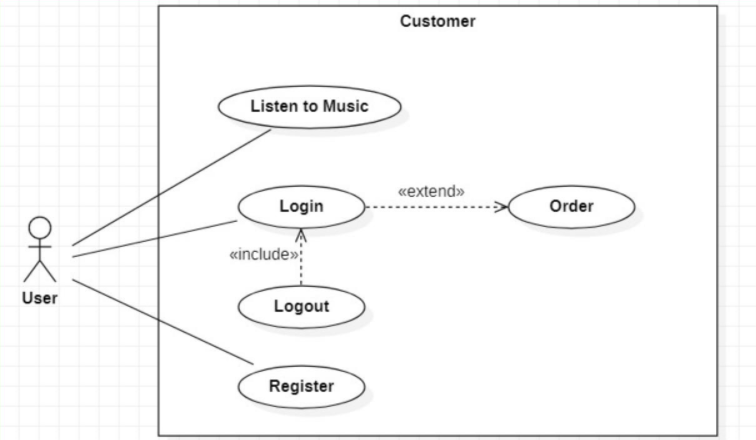
*Table 1: Statistical tables*

* We have received this feedback from actual users. These comparative reviews only tell us what needs to be fixed and changed so we can do better by mistakenly delivering the best product to our customers.
  1. **Use case diagram for the whole system**
* UC Admin



*Figure 1: Administrator*

* UC Customer



*Figure 2: Customer*

* 1. **Use case:**
* **UC Login**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case name:** Login | **ID:** UC-01 | | **Priority:** High |
| **Actors:** Admin | | | |
| **Description**: An administrator gets log in through log in page. | | | |
| **Trigger**: Admin use their information to get in to their Tune Source account.  **Type:** ☒ External ☐Temporal | | | |
| **Preconditions**: required exist data account. | | | |
| **Normal Course:**  1. Log in at Tune Source page.  2. Input data account (Username and Password).  3. Sending next page, if:   * Successfully log in: Send to the home page. * Failed to login: re-load the login page and the input data account (Username and Password).   Otherwise, the use case ends! | | **Information for steps:** | |
| **Post-Conditions:**  1. Online account.  2. Admin can get other functions when they're logged in. | | | |

*Table 2: UC-01*

* **UC Listen to Music**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case name:** Listen to Music | **ID:** UC-02 | | **Priority:** Medium |
| **Actors:** Admin, User. | | | |
| **Description**: User and Admin can listen to music on the Home page | | | |
| **Trigger**: User and Admin can listen to music on the Home page  **Type:** ☒ External ☐Temporal | | | |
| **Preconditions**: Does not require signing in to the system | | | |
| **Normal Course:**   * Access the Home page and listen to music.   Otherwise, the use case ends! | | **Information for steps:** | |
| **Post-Conditions:** | | | |

*Table 3: UC-02*

* **UC Logout**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case name:** Logout | **ID:** UC-03 | | **Priority:** High |
| **Actors:** Admin, User. | | | |
| **Description**: Admin and User can get log out when they were logged in before. | | | |
| **Trigger**: Admin and User can get log out when they were logged in before.  **Type:** ☒ External ☐Temporal | | | |
| **Preconditions**: Required logging into the system | | | |
| **Normal Course:**  1. Access the Login page and input data account (Username and Password).  2. Click the button Log out.  Otherwise, the use case ends! | | **Information for steps:** | |
| **Post-Conditions:** | | | |

*Table 4: UC-03*

* **UC Account**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case name:** Logout | **ID:** UC-04 | | **Priority:** High |
| **Actors:** Admin. | | | |
| **Description**: Administrators who have previously logged in to the system will be able to access the UC - Account | | | |
| **Trigger**: UC-Account provides the following functions: Edit user account metadata, remove user accounts if necessary.  **Type:** ☒ External ☐Temporal | | | |
| **Preconditions**: Required logging into the system | | | |
| **Normal Course:**  1. Logged in to the system with an administrator account.  2. Access the Account page.  Otherwise, the use case ends! | | **Information for steps:** | |
| **Post-Conditions:**  1. Account online.  2. Admin can get other functions when they are logged in | | | |

*Table 5: UC-04*

* **UC Music**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case name:** Music | **ID:** UC-05 | | **Priority:** High |
| **Actors:** Admin. | | | |
| **Description**: Administrators who have previously logged in to the system will be able to access the UC - Music | | | |
| **Trigger**: UC-Music provides the following functions: add a new music song, edit details on a music song, or remove a music song if necessary.  **Type:** ☒ External ☐Temporal | | | |
| **Preconditions**: Required logging into the system | | | |
| **Normal Course:**  1. Logged in to the system with an administrator account.  2. Access the Music page.  Otherwise, the use case ends! | | **Information for steps:** | |
| **Post-Conditions:**  1. Account online.  2. Admin can get other functions when they are logged in | | | |

*Table 6: UC-05*

* **UC Category**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case name:** Category | **ID:** UC-06 | | **Priority:** High |
| **Actors:** Admin. | | | |
| **Description**: Administrators who have previously logged in to the system will be able to access the UC - Category | | | |
| **Trigger**: The UC-Category includes the following functions: Add a new music category, edit a music category information or delete a music category if necessary.  **Type:** ☒ External ☐Temporal | | | |
| **Preconditions**: Required logging into the system | | | |
| **Normal Course:**  1. Logged in to the system with an administrator account.  2. Access the Category page.  Otherwise, the use case ends! | | **Information for steps:** | |
| **Post-Conditions:**  1. Account online.  2. Admin can get other functions when they are logged in | | | |

*Table 7: UC-06*

* **UC Order Details**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case name:** Order Details | **ID:** UC-07 | | **Priority:** High |
| **Actors:** Admin. | | | |
| **Description**: Administrators who have previously logged in to the system will be able to access the UC – Order Details | | | |
| **Trigger**: The UC- Order Details includes the following functions: Edit an order details information or delete order details if necessary.  **Type:** ☒ External ☐Temporal | | | |
| **Preconditions**: Required logging into the system | | | |
| **Normal Course:**  1. Logged in to the system with an administrator account.  2. Access the Order Details page.  Otherwise, the use case ends! | | **Information for steps:** | |
| **Post-Conditions:**  1. Account online.  2. Admin can get other functions when they are logged in | | | |

*Table 8: UC-07*

* **UC Order**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case name:** Order. | **ID:** UC-08 | | **Priority:** Medium |
| **Actors:** User. | | | |
| **Description**: The user that have previously logged into the system will be able to buy a music song. | | | |
| **Trigger**: User can buy any song they like in the Order.  **Type:** ☒ External ☐Temporal | | | |
| **Preconditions**: Required logging into the system | | | |
| **Normal Course:**  1. Logged in to the system with a user account.  2. Access the Order page.  Otherwise, the use case ends! | | **Information for steps:** | |
| **Post-Conditions:**  1. Account online.  2. Admin can get other functions when they are logged in | | | |

*Table 9: UC-08*

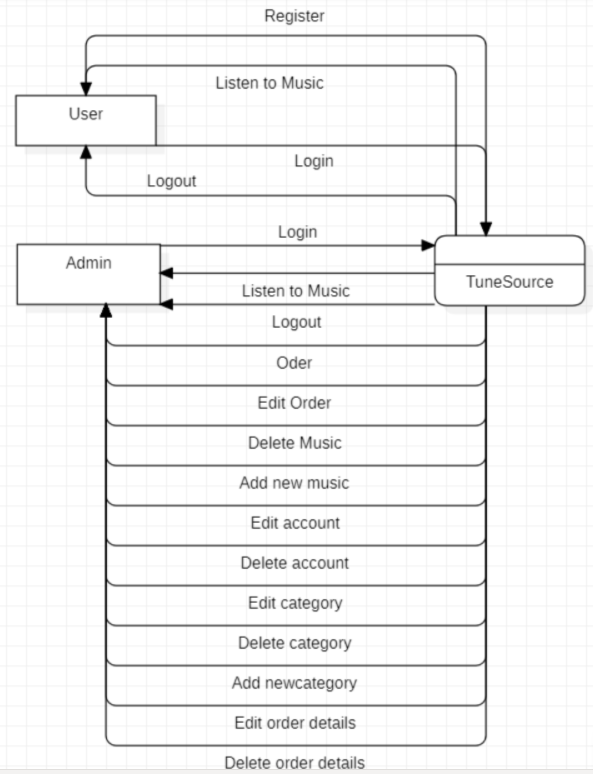
* **UC Register**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case name:** Register | **ID:** UC-09 | | **Priority:** High |
| **Actors:** User. | | | |
| **Description**: Users can register membership accounts through the page "Register" on the website. | | | |
| **Trigger**: Upon active authentication, the user can log in via the login page and buy music.  **Type:** ☒ External ☐Temporal | | | |
| **Preconditions**: Does not require logging into the system. | | | |
| **Normal Course:**  1. Access to the Register page on the website.  2. The input data account has to be registered.  3. To send, press the button.  Otherwise, the use case ends! | | **Information for steps:** | |
| **Post-Conditions:** | | | |

*Table 10: UC-09*

# **DFD (Data Flow Diagram)**

# Level 0 DFD: This DFD show how admin and user relation to Tune Source.



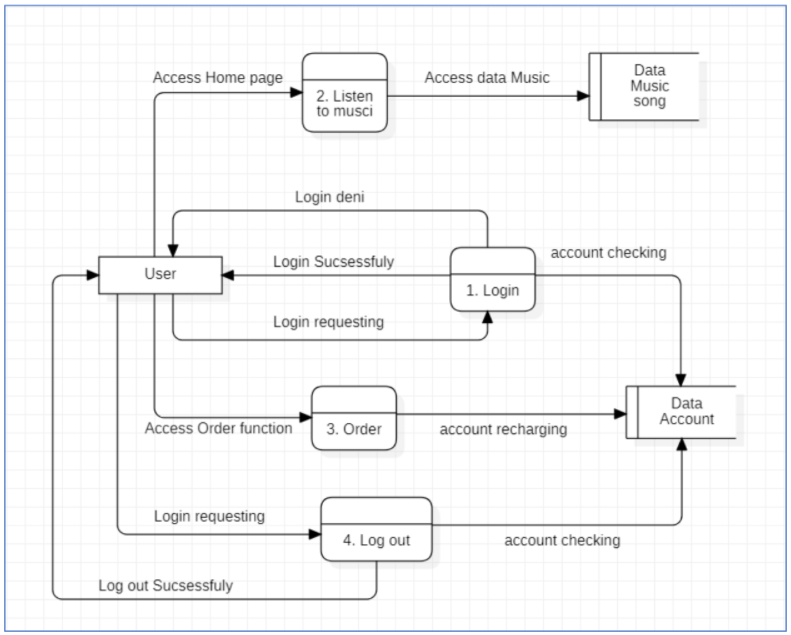
*Figure 3: DFD-level 0*

* Level 1 DFD: more detailed DFD, Admin will interact with each process, and the system will interacts with the database.



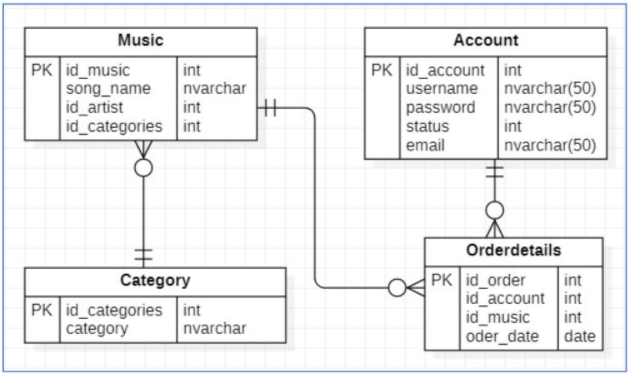
*Figure 4: DFD-level 1 of admin*

* Level 1 DFD: more comprehensive DFD, the user interacts with each process, and the device interacts with the database.



*Figure 5: DFD-level 1 of user*

* 1. **ERD (Entity Relationship Diagram)**



*Figure 6: ERD*

# 

*Figure 7: Database Diagram*

# P7 Explain how user and software requirements have been addressed.

# At this point, in order to conform to the specifications, set out above, we will elaborate. I select Tune Source functions for the diagram as-Register, Login/Log Out, Music Listening, Account Management, Music Management, Category Management, Order Information Management.

# Login to the browser:

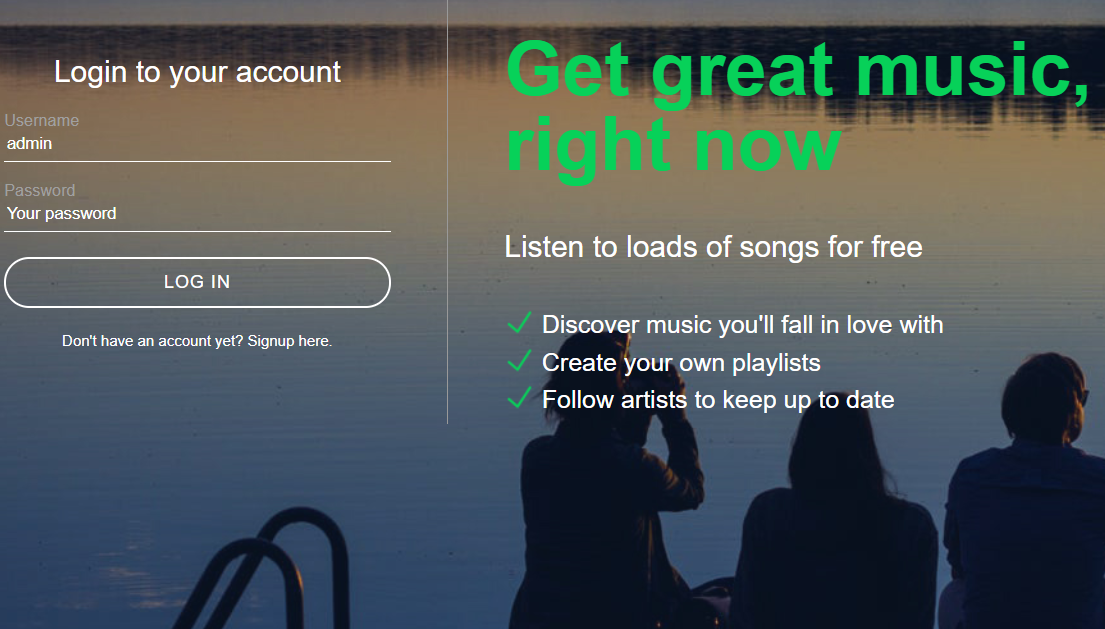
# -There are two text boxes in the login window to describe the login info.

# -This is the link to the registration window and the Tune Source theme.

# -Link to the SQL database to verify the login info

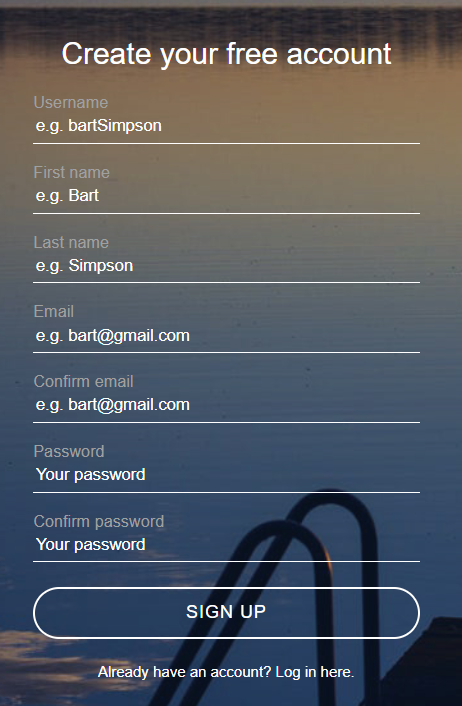
# -Set the char to \* for your password

* 1. **Login**



*Figure 8:* *Login Page*

* 1. **Register page**



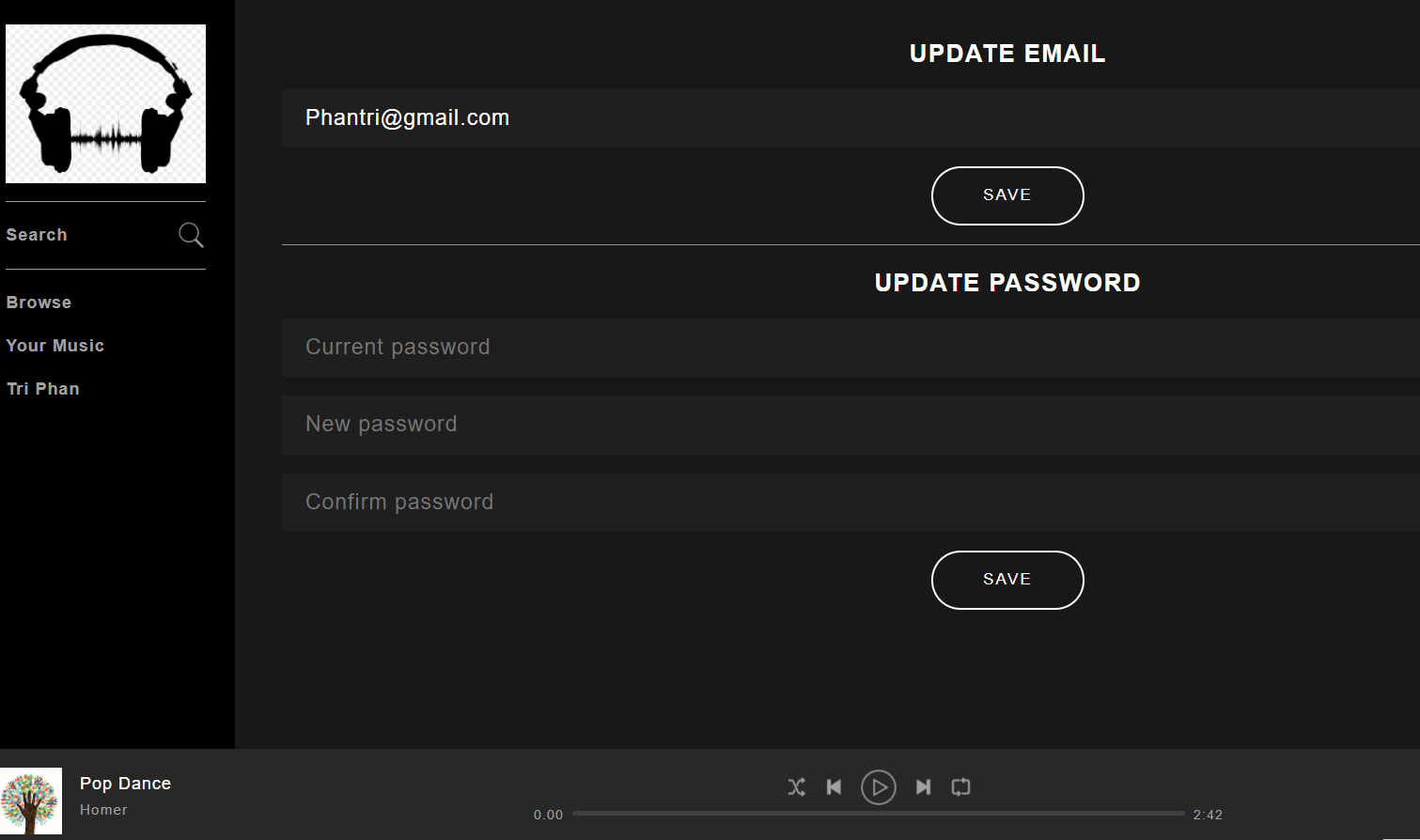
*Figure 9:* *Register page*

* 1. **Home page of admin account**



*Figure 10:* *Home page of admin account*

* 1. **Management Account**



*Figure 11:* *Management account*

1. **Management Music**

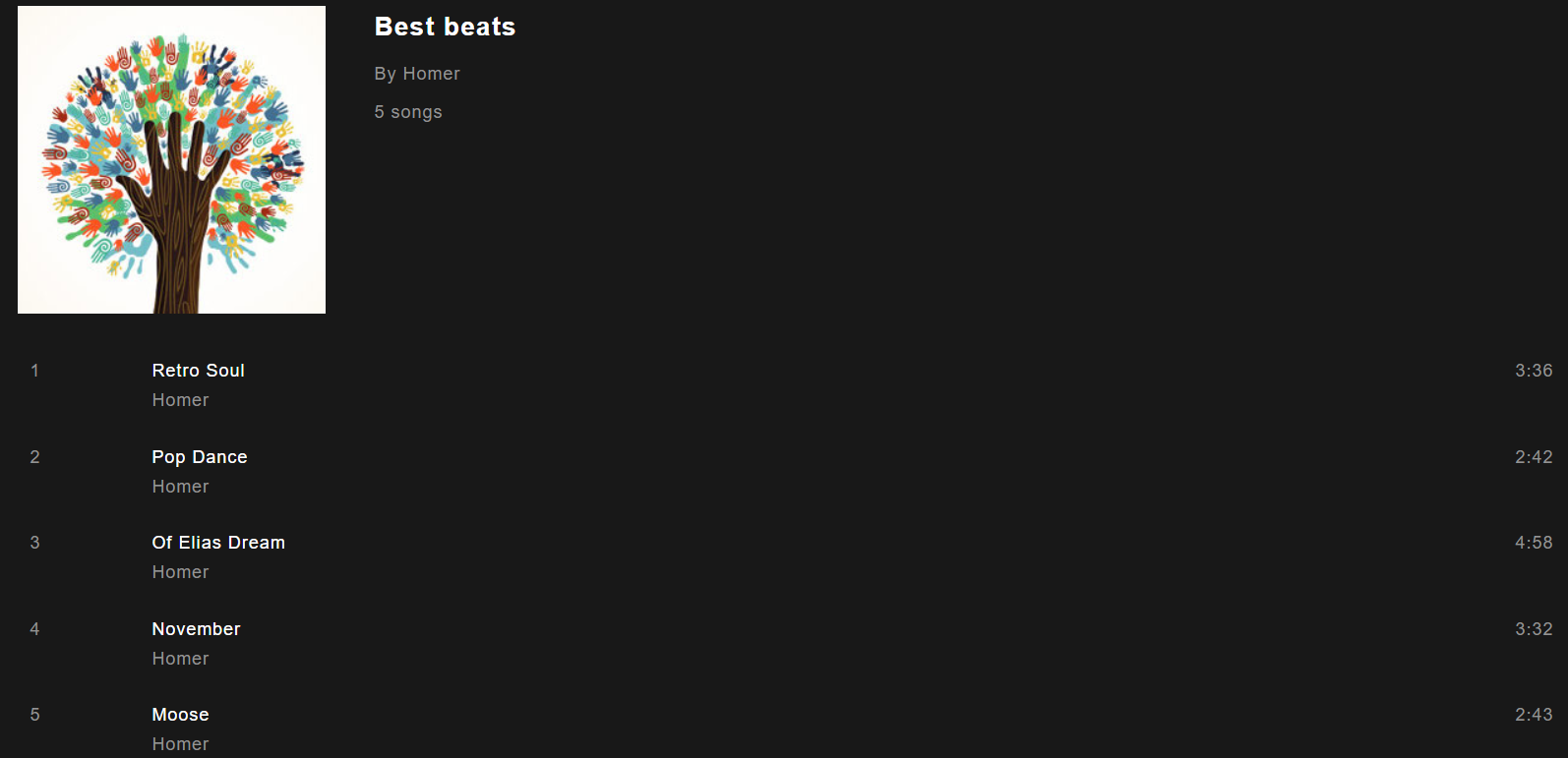
****

Figure 12: Management music

1. **Management Category**

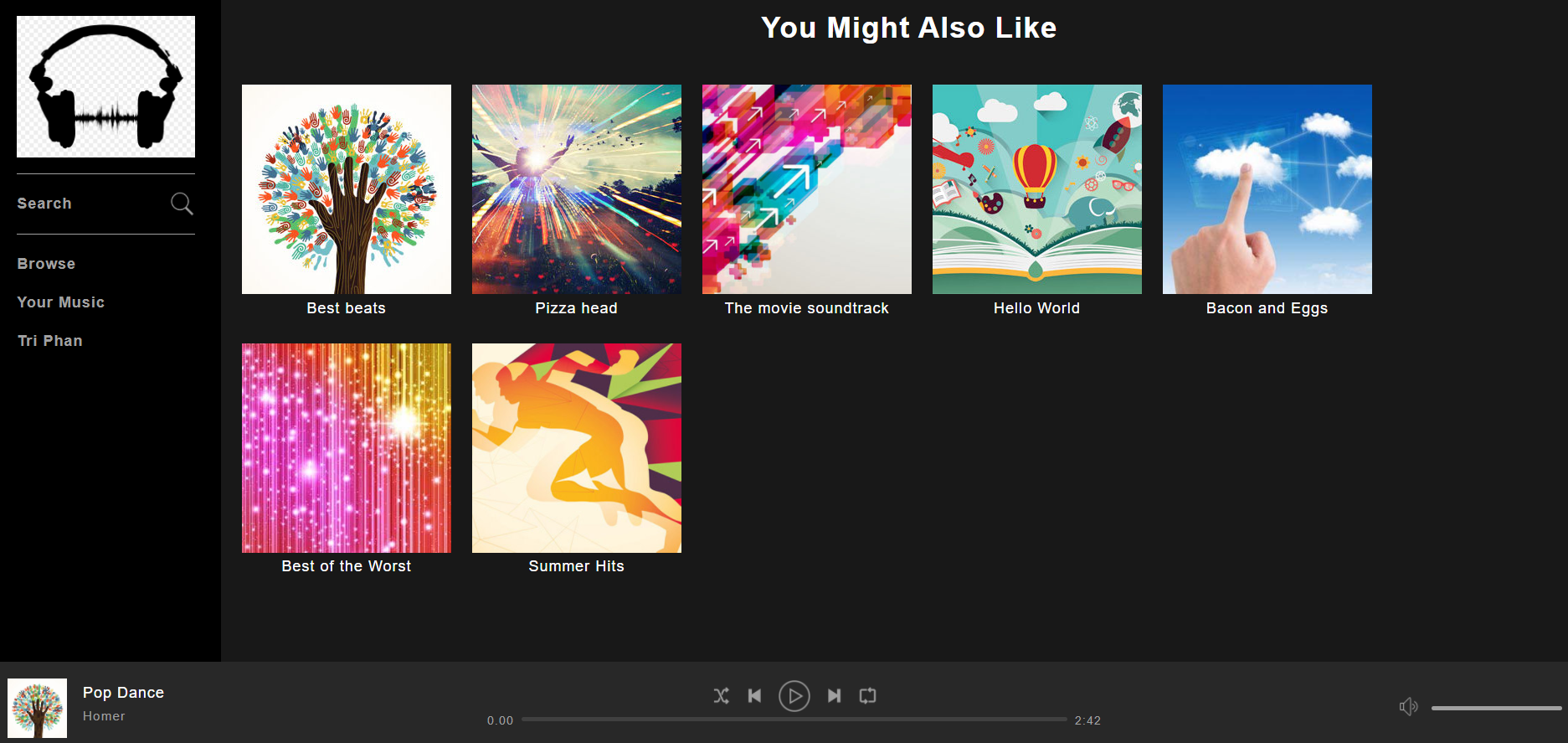
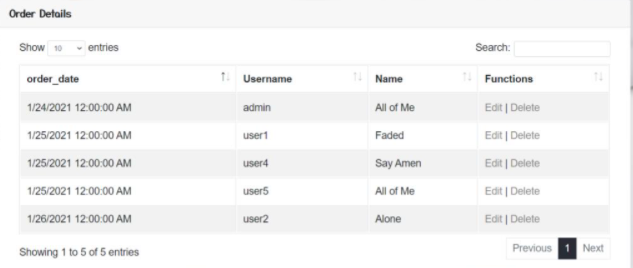
****

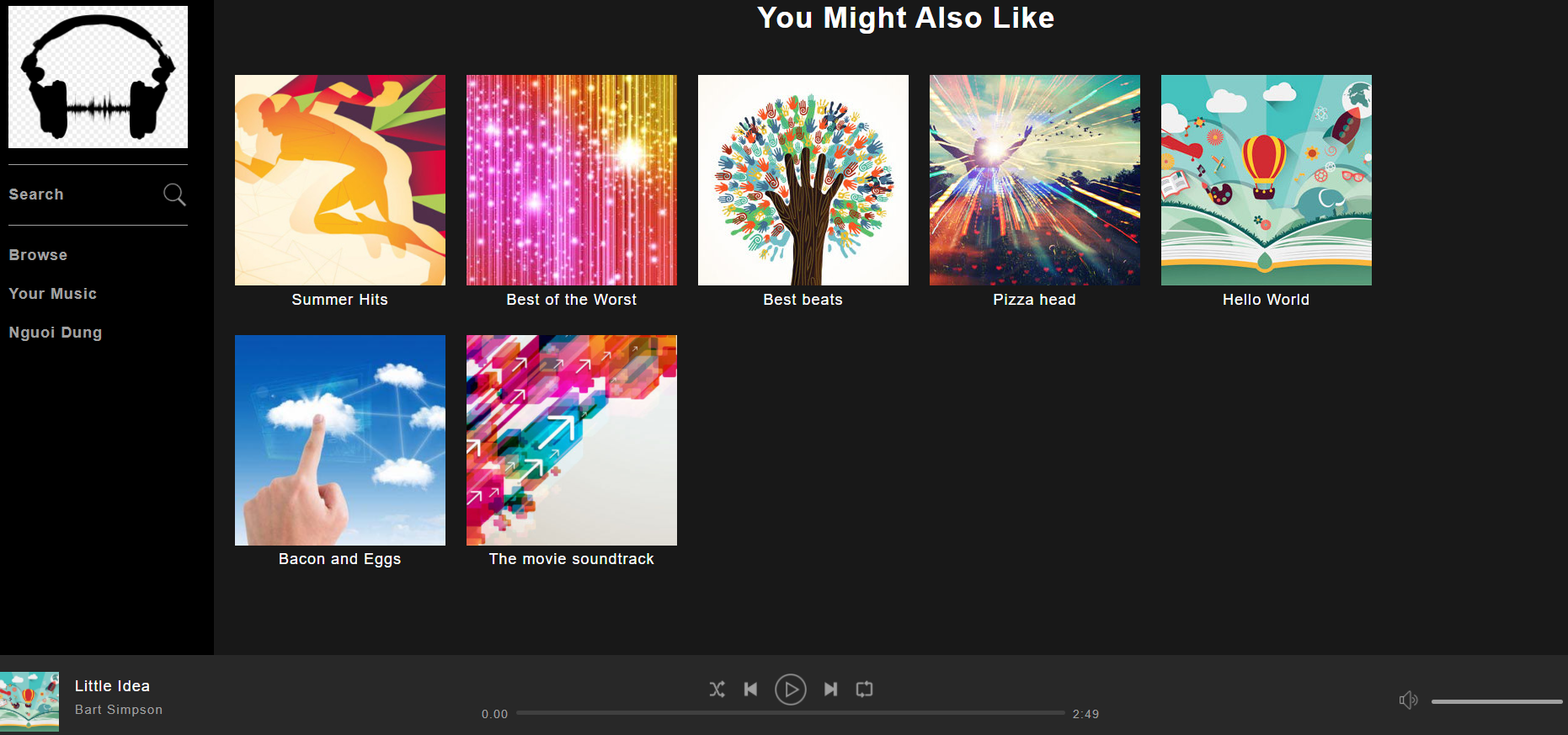
Figure 13: Management Category

1. **Management Order Details**

****

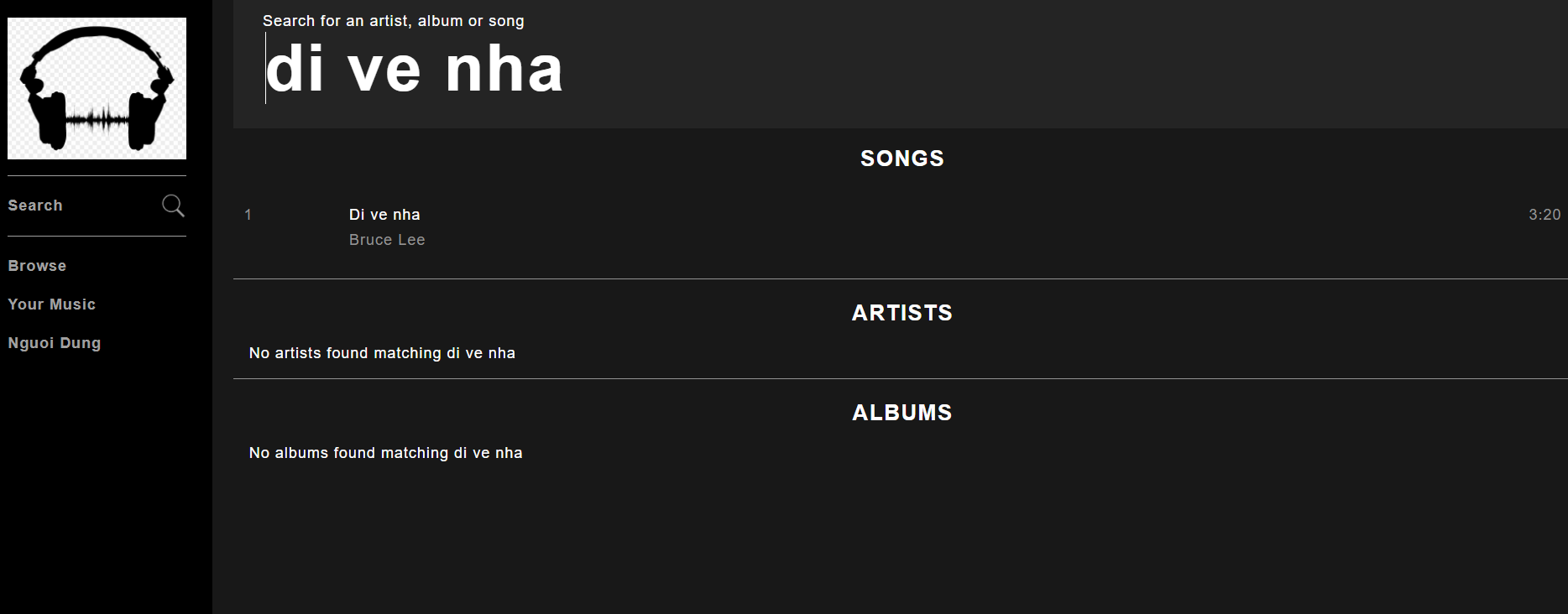
*Figure 14: Management Order Details*

1. **Home page of the User account**

****

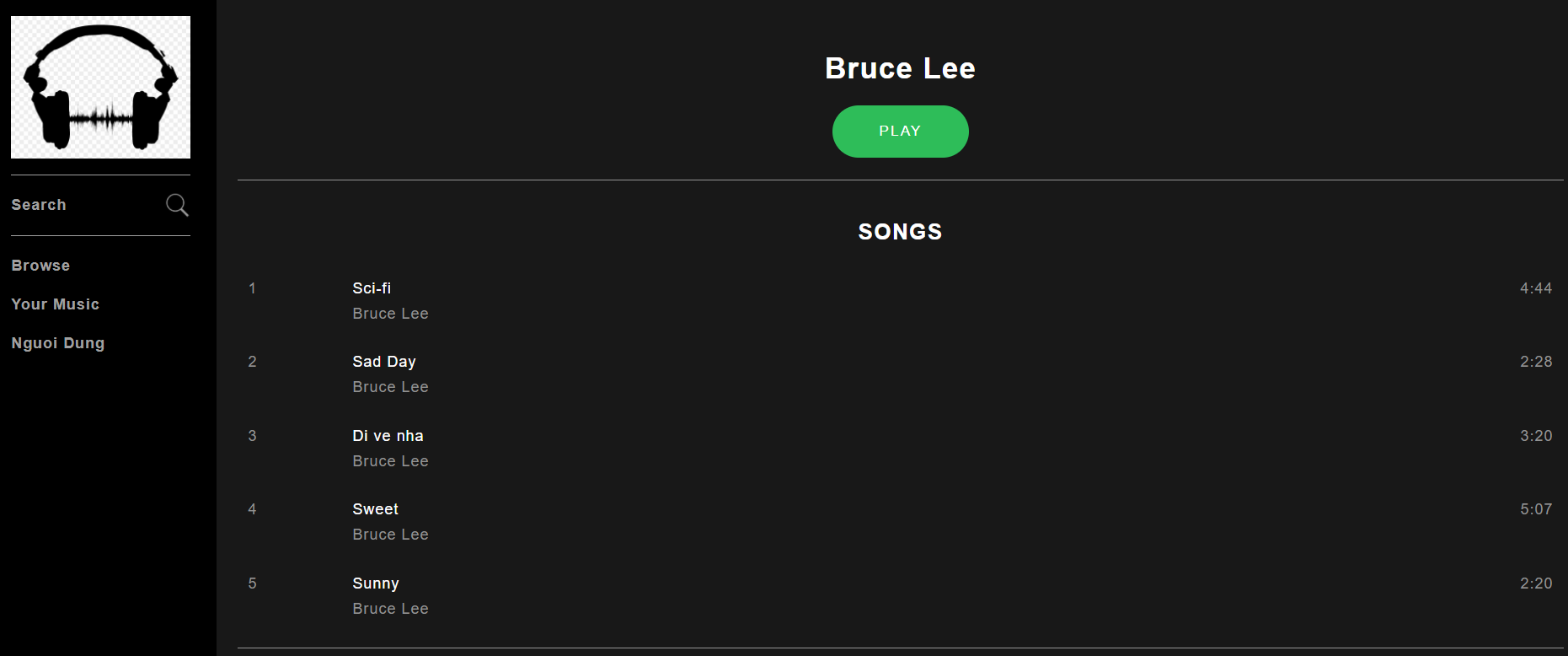
*Figure 15: Home page of the User account*

1. **Search for artist, album or song**

****

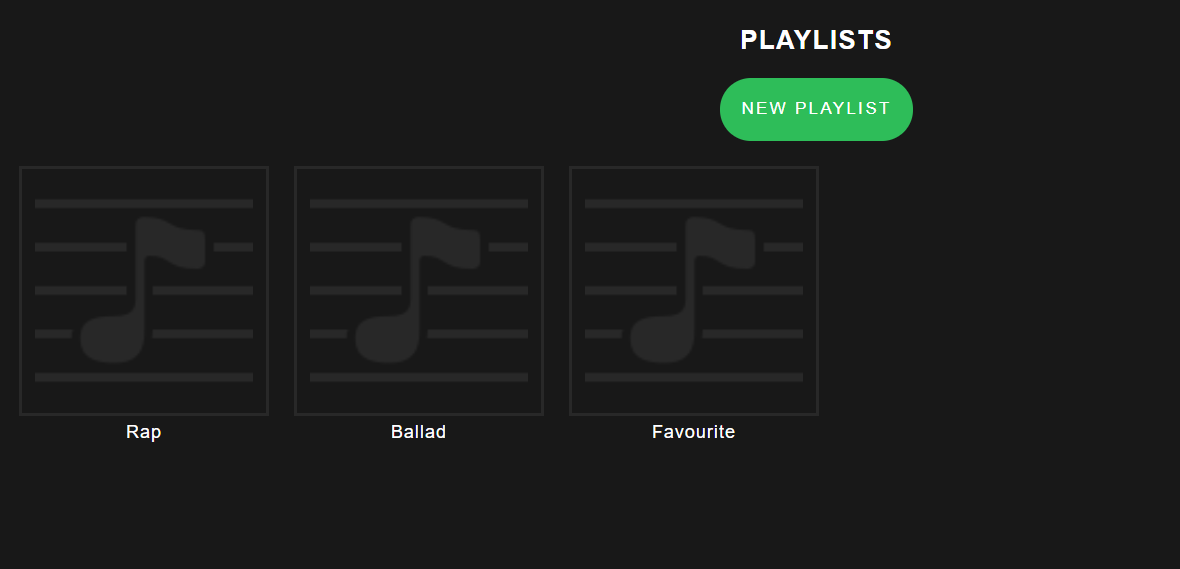
*Figure 16: Search for artist, album or song*

1. **Sorted by artist name**

****

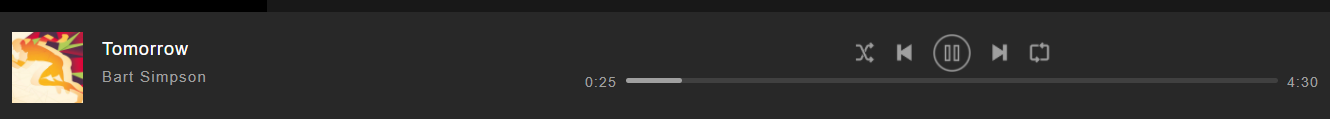
*Figure 17: Sorted by artist name*

1. **Playlists of user**

****

*Figure 18: Playlists of user*

1. **Listening to Music basic**

****

*Figure 19: Listening to Music basic*

* Note: Without having to log in, individuals will listen to the effects of basic home page machine songs.
* Conclusion

We are based on functional requirements and have fulfilled these roles based on the functional criteria and suggested by partners. We have consulted other websites for listening and have developed a user-friendly GUI. For further compatibility, we'll upgrade the management GUI

# References

mona.media, n.d. [Online]   
Available at: https://mona.media/bao-mat-website-va-nhung-dieu-can-biet-de-toi-uu/  
[Accessed 9 2 2020].