ASSIGNMENT 2 FRONT SHEET

Qualification	**BTEC Level 5 HND Diploma in Computing**		
Unit number and title	Unit 9: Software Development Life Cycle		
Submission date	22/03/2021	**Date Received 1st submission**	
Re-submission Date	30/03/2021	**Date Received 2nd submission**	
Student Name	PHAM CAO NGUYEN	**Student ID**	GCC18074
Class	GCC0801	**Assessor name**	NGUYEN HUNG DUNG
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	CAONGUYEN

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.
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

2. (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

3. (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.	

Contents

Chapter 3 Undertake a software development lifecycle

Tune Source is a company headquartered in southern California. Tune Source is the brainchild of three entrepreneurs with ties to the music industry: John Margolis, Megan Taylor, and Phil Cooper. Originally, John and Phil partnered to open a number of brick and mortar stores in southern California specialising in hard-to-find and classic jazz, rock, country, and folk recordings. Megan soon was invited to join the partnership because of her contacts and knowledge of classical music. Tune Source quickly became known as the place to go to find rare audio recordings. Annual sales last year were $40 million with annual growth at about 3%–5% per year. Tune Source currently has a website that enables customers to search for and purchase CDs. This site was initially developed by an Internet consulting firm and is hosted by a prominent local Internet Service Provider (ISP) in Los Angeles. The IT department at Tune Source has become experienced with Internet technology as it has worked with the ISP to maintain the site.

V. Undertake a software investigation to meet a business need. [1]

a. Stakeholders, their roles, and interests.

- **The stakeholders include:**
  - **Customer:**
    - **Roles**
      - Provide functional requirements, useful features on the system.
      - Purchasing to get music.
      - Use system Report system error when available Provide comments on completing the project.
    - **Interests**
      - Listen to and download music from the system Receive care services from system owners
      - Can listen to it out before you buy and download.
  - **Admin:**
    - **Roles**
      - Managing the music
      - Managing the web
      - Managing the customer
    - **Interests**
      - Manage and delete music and customer information.
      - Access to the website under the freed music download admin.
  - **Investors:**
    - **Interests**
      - Provide support to build the system
    - **Interests**
      - After the system is up and running, you can receive income from the system provider.
  - **System Owner:**

- **Roles**
  - System management
  - Ensuring system performance
  - System maintenance
  - System update
  - Provide customer care services
- **Interests**
  - Receive benefit directly from the system

o FRs and NFRs of Tune Source Project

✓ Functional requirement

- Log in / Log out / Register
- Listen to Music
- Order
- Manage product
- Manage Music
- Manage Account
- Manage Order Details
- Note: All of the management function has included CRUD.

✓ Non-Functional requirement

- **Operational**
  - **Descriptions:** The technical environments in which the system will operate
  - **Requirements:**
    - The system can run on handheld devices
    - The system should be able to work on any Web browser
    - The system should be able to integrate with the existing inventory system
- **Security**
  - **Descriptions:** Restrict access and require information to ensure safety
  - **Requirements:**
    - Only the administrator can view the customer's profile as well as change some information about the website.
    - The system has anti-virus measures to disable and block malicious software such as hackers.
    - Customer can see their downloaded history at all times
- **Performance**
  - **Descriptions:** The speed, quality, capacity, and reliability of the system.
  - **Requirements:**
    - The system will have to operate 24/24
    - The system downloads new status parameters immediately as a change.
- **Cultural and Political**
  - **Descriptions:** Legal and cultural factors affect the interests and interests of the system.
  - **Requirements:**
    - Personal information is protected in compliance with the Data Protection act

        - Country managers are permitted to authorize custom user interfaces within their units.
    - Our systems are operated and hosted on Cloud Computing. We recommend for partners to use the cloud systems of Microsoft Azure. It ensures the security and operability of the system.

○ Relationships between the FRs and NFRs.

- Software requirements are divided into two parts, FRs and NFRs. FRs determine the functionality, while, NFRs determine how a system is supposed to be. In the literature, we have identified that most of the work is related to FRs. NFRs have received less attention from the goal-oriented requirements engineering community. The aim of this paper is to present a taxonomy of non-functional requirements so that the requirements analyst can easily identify different types of NFRs according to their needs in the early phase of requirements engineering.
- Non-Functional Requirements (NFRs) have been increasingly accepted as crucial to the success of software projects.
- However, the current state of industrial practice is still focusing mainly on functional requirements (FRs) using UML use cases as the main tool for requirements elicitation and modeling. In order to encourage practitioners to focus more on much deserved NFRs, there is a need for frameworks to provide a smooth transition from the use case modeling. This paper proposes such a framework for integrating NFRs with FRs in the use case model. It proposes that key use case model elements, specifically, actor, use case, actor-use case association, and system boundary, be used as association points to provide intuitive context for the NFRs. The framework specifies the scope of each type of NFR association through the formalization of NFR scope propagation rules that take advantage of relationships between use case model elements (specialization, generalization, extends, includes). A process and illustration are presented to demonstrate how to apply the method to a simplified pricing system.

○ Tune Source Project

- We are carried out in accordance with the needs of our partners. An on-demand music provider and a partner for whom we build streaming music systems are among the stakeholders. Throughout the operation, we implement security and accessibility specifications.

○ Requirement

- **Requirements describe:** A requirement is a statement of what the system must do or what characteristics it needs to have.
    - What the business needs (business requirements)
    - How the system should be built (system requirements)
    - What the users need to do (user requirements)
    - Characteristics the system should have (non-functional
    - What the software should do (functional requirements)

b. In Tune Source, I chose interview because:

○ The technique we would use to obtain the requirements will be an interview.

- **Search interview object.**
    - To chose interviewees, we must understand their position in the project as well as the intent of the interview. Meeting times with each interviewee will be divided.
    - Develop a list of customer interview questions you will need to develop a list of interview questions before you conduct customer interviews and you

should seek to understand all stages of how a customer makes a purchasing decision.

- **Interviewees:** Manager, Customer/User, Business Analyst.
- **Designing Interview Questions:** The interview will take the form of a scripted interview, in which we, the interviewers, will ask a preset list of questions in a top-down interview format.
    - How do customers purchase and download music?
    - What information the customer must give when register?
    - How many users active at peak time?
    - What problem you usually get when using the web?
    - What improvement you would like to have for the web in the future?
    - Why do you think so?
    - Can you give some more examples?
    - Can you describe it in more detail?
    - Do you have any suggestions and want to improve?
    - Do you feel satisfied listening to music on our website?
- **Preparing for the Interview:** Include Mathematics, Ethics, legal culture, religious Knowledge, and Indigenous Knowledge.
- **Reason for interview:** To gather, gather opinion, satisfaction, the future market, and sort out the information necessary for Tune Source system
- **Areas of discussion:** About features, functions, and the future goals of Tune Source Website
- **When conducting the Interview:** Appear to be professional and unbiased, friendly, record all information, be sure you understand the issues that are discussed, give interviewee time to ask questions, and briefly explain, enthusiasm in serious interviews, and always answer honestly.
- **After the interview:** The interviewer must write an interview summary, which has a concise history of the interview, the location where it was held, and the interview's topic or themes. Interview comments may also be included in the survey. Both interviewees are given the report with instructions to read it and notify the analyst of any clarifications or changes.

o Interview

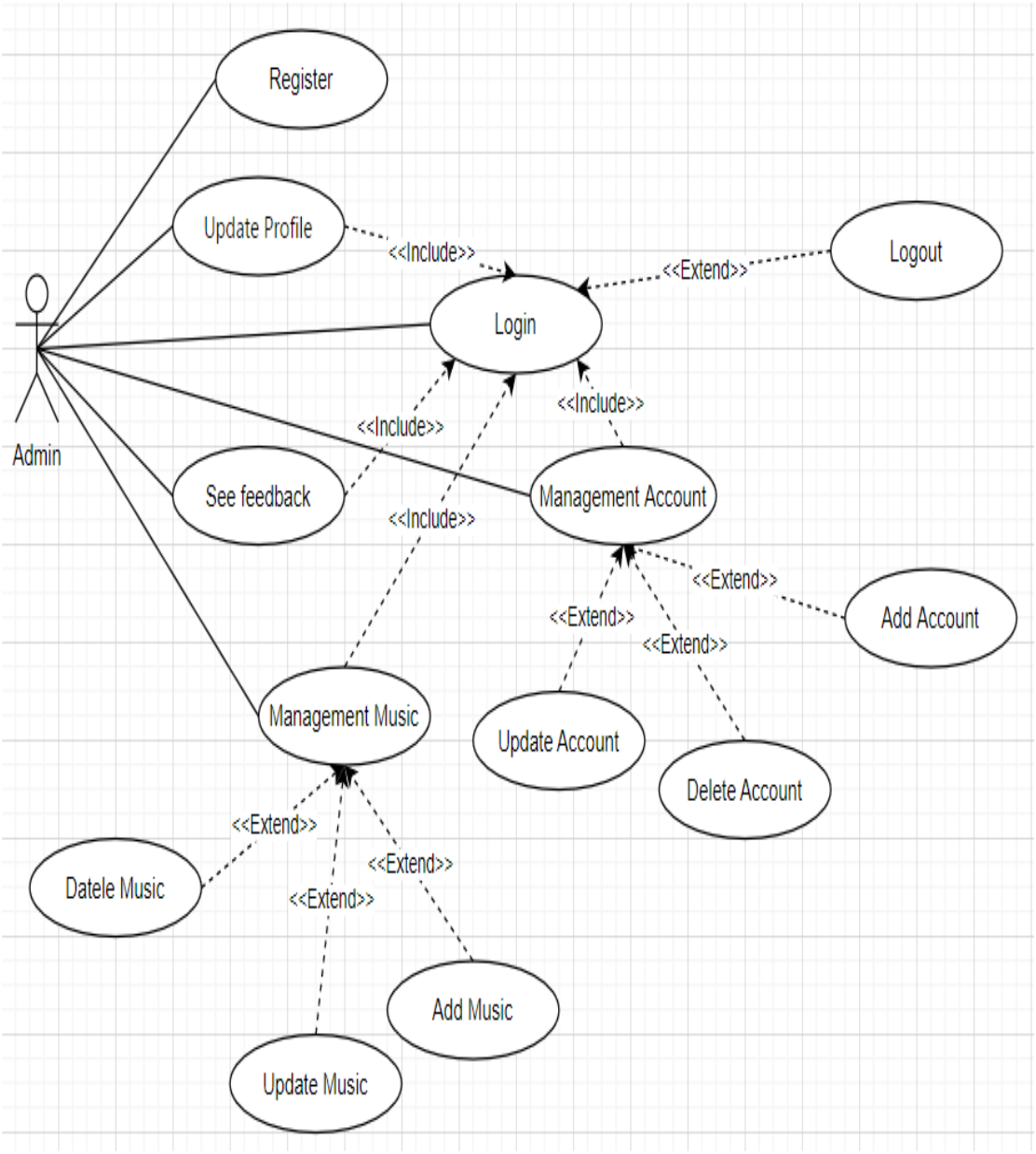
- Interviewing customer help eliminate guesswork. It removes opinions from your marketing strategy and replaces them with facts.
    - It may be appropriate to find undiscovered requests
    - Convergence in a few common requirements will create a repository of usage requirements
    - Throughout the project
    - Doubt will not be substituted for an interview.
- How to avoid prejudice of users when meeting their requirements question? We use questions about user natural problems without regard to any scope.
    - Who is the user?
    - Who is the customer?
    - Do they need a change?
    - Where else can I find a solution to this problem?

- **With question:**
    - There are a few questions I need to apply to my survey to improve the results, but I'll start with these:

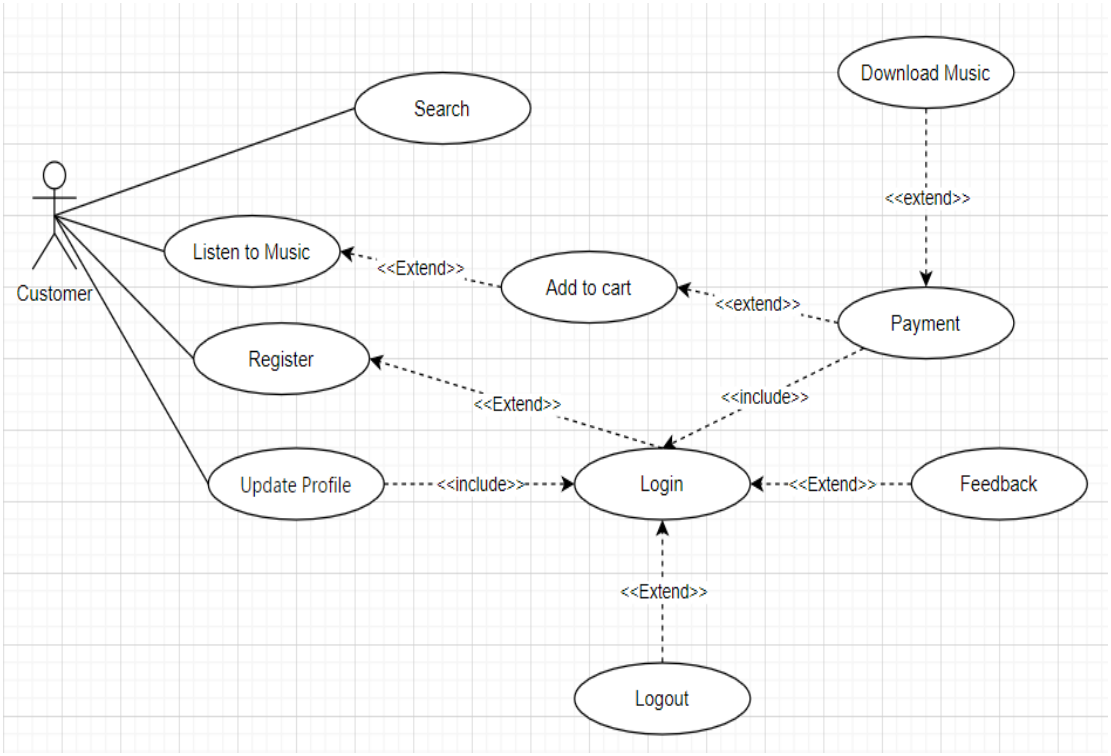
- To you, how much does a normal song cost?
    - People think about price first when they choose to purchase or pay for something, so this query will help us properly value our product and tailor it to the vast majority of users.
- Would you like to listen to music sample?
    - Based on the requirement of the brief, this question is to make sure that the function we are going to make it necessary for the user
- Do you listen to music online often?
    - The aim is to determine whether or not this person is familiar with online music software and possesses the necessary skills to use online music software to detect website feature errors.
- Should the Tune Source website have a music product review function?
    - The wrong goal is to allow users the right to experience and rate themselves so that if this app is bad, we can both fix the bug and develop the benefits if it is good.
- When using the website to buy music online, what requirements do you need to meet your needs?
    - The goal is to learn the requisite web functions, to listen to music online. To be able to build the most full web for users.
- Do you accept that the app should have the ability to save your favorite song to your account?
    - They want to listen to their favorite song and listen again and again without searching many times. With this additional function, the online music website attracts more customers

VI. Use appropriate software analysis tools/techniques to carry out a software investigation and create supporting documentation.

- **User Admin**



- **User Customer**



- **UC - Login**

Use Case Name: Login	ID: UC1	Priority: High
Actor: Admin		
Description: An administrator gets login through login page.		
Trigger: Admin use their account to get login Tune Source		
Type: External		
Precondition: o required exist data account. o Internet connection is required.		
Normal Course: 1. Log in at Tune Source page. 2. Input data account (Username and Password) and click "login". 3. The system authorizes login and allows the administrator to use update delete add information, see the feedback of customer the site. 4. The system records the login activity into a log.		
Post condition: o Account online. o Admin logins successfully. o Admin can get other functions when they are logged in. o The system records the login activity successfully.		
Exceptions: o The administrator forgot the account. o The administrator decides to disable the account. o The system authorizes login unsuccessfully and displays a message.		

- **UC - Register**

Use Case Name: Register	ID: UC2	Priority: High
Actor: Customer		
Description: Users can register membership accounts through the page "Register" on the website.		
Trigger: Customer registers for an account. Upon successful registration, the customer can log in through the login page and can purchase music.		
Type: External		
Precondition: ○ Customer access the website and choose to register ○ Internet connection is required		
Normal Course: ○ Access to the Register page on the website. ○ Customer access the website and choose to register ○ Customer enter personal information (name, email, phone number, gender, address, password, date of birth) ○ Click the button to submit. ○ The system will store the customer information ○ After successfully register, the customer return to the homepage.		
Post condition: Customer information is stored in the database.		
Exceptions: ○ If the customer's information is inappropriate, the system refuses to create an account and display a message. • Customer cancels the register • Customer re-enter the information ○ If the customer's information is identical to the information already saved, the system refuses to create an account and display a message. • Customer cancels the register • Customer re-enters the information		

- **UC - Logout**

Use Case Name: Logout	ID: UC3	Priority: High
Actor: Admin, Customer		
Description: Admin and User can get log out when they were logged in before.		
Trigger: Admin and customers can log out of the website when they are not in use.		
Type: External		
Precondition: Required logging into the system.		
Normal Course: ○ Click on the "logout" icon in the toolbar of the website to log out if the customer has logged in before. ○ Otherwise, use case end.		
Post condition:		
Exceptions: Obligatory the admin and customer login to the website before.		

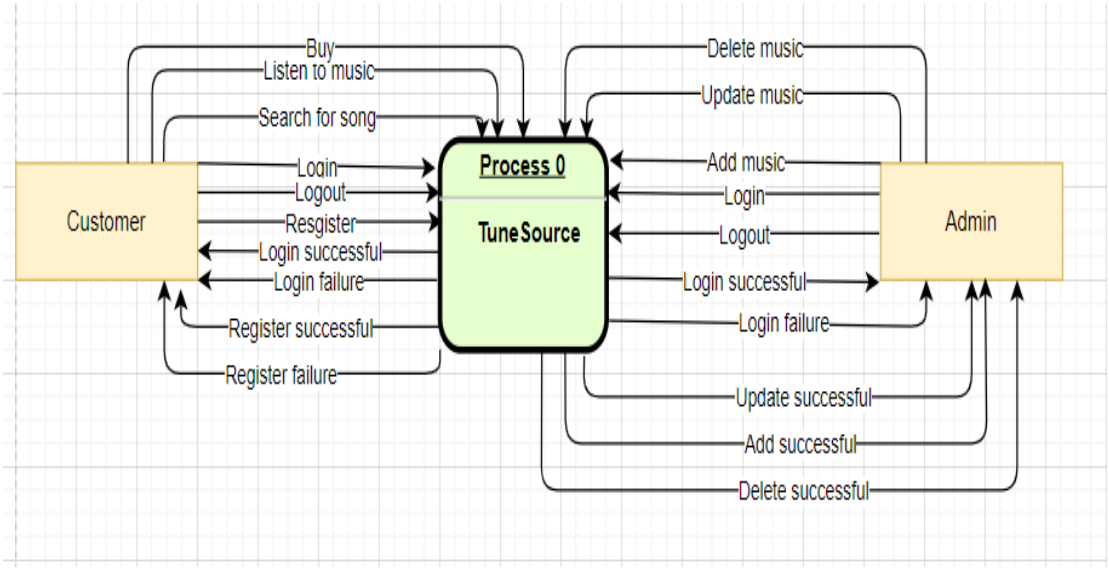
- **UC - Feedback**

Use Case Name: Feedback	ID: UC4	Priority: High
Actor: Customer		
Description: This function allows the customer to send feedback for products		
Trigger: Customer gives feedback for the product.		
Type: External		
Precondition: o The customer logged in successfully. o The customer's device is connected to the internet.		
Normal Course: o Customers choose the feedback tab. o The customer enters content to feedback. o Customer click "post" button. o The system will save feedback into the database.		
Post condition: Customer give feedback successfully		
Exceptions: o The feedback content is below 1000 characters. o If customers enter the feedback content over 1000 characters. The system will display customers to shorten the content feedback. • The system will notify customers to enter more than 1000 characters and request customers to re-enter. • Customer cancel feedback		

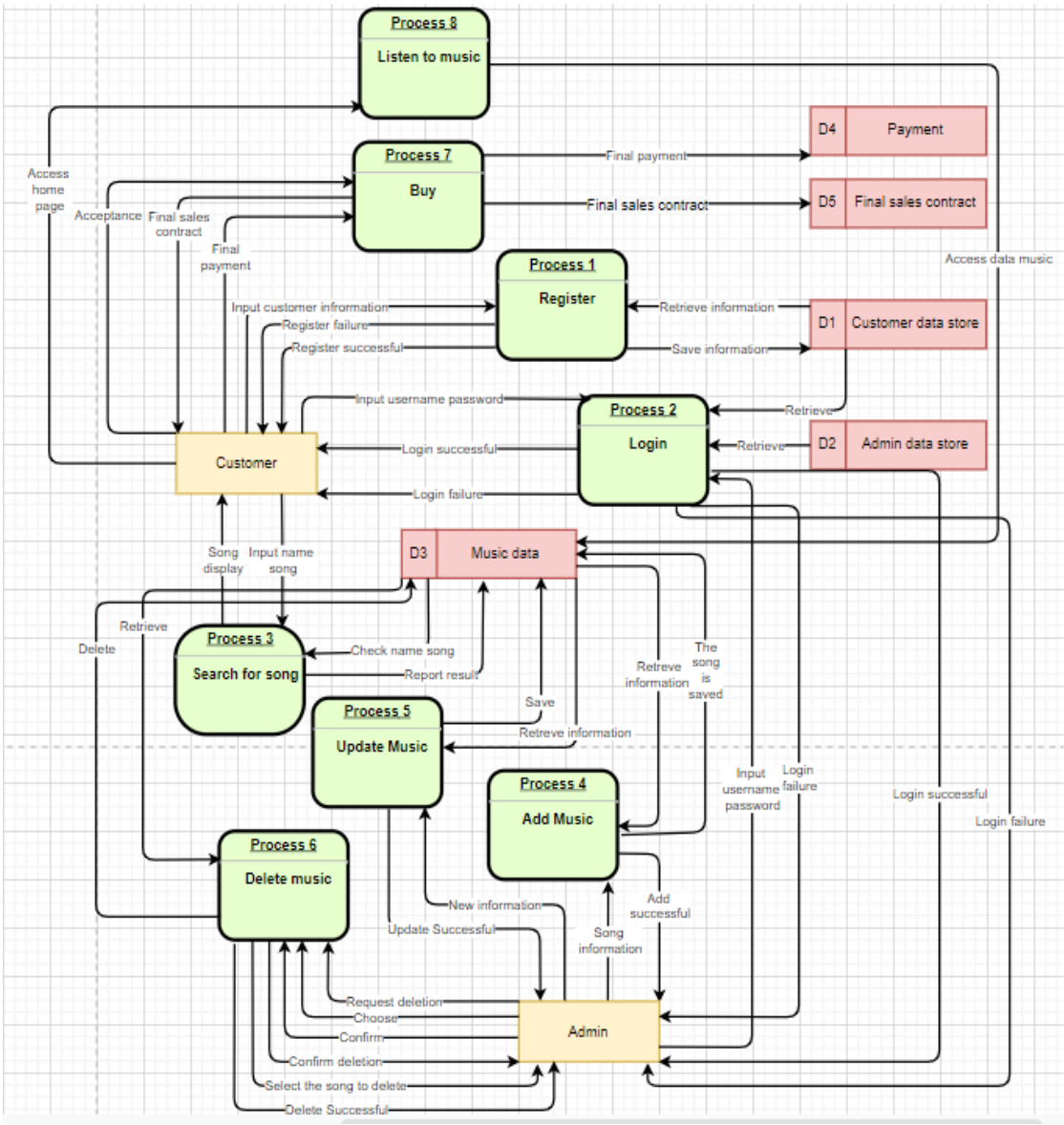
- **UC - Order**

Use Case Name: Order	ID: UC5	Priority: High
Actor: Customer		
Description: Customer that have previously logged into the system will be able to buy music songs.		
Trigger: Customers can buy any song they like in the Order.		
Type: External		
Precondition: o Required logging into the system. o Internet connection is required.		
Normal Course: o Logged into the system with a Customer account. o Customer can listen to music before choosing Order		
Post condition: Online account required		
Exceptions: o Ask the customer to provide the age when ordering, there may be age-restricted music genres. o The system will display a message. when the customer provides the age • The customer has successfully ordered. • Customer order failed (cancel).		

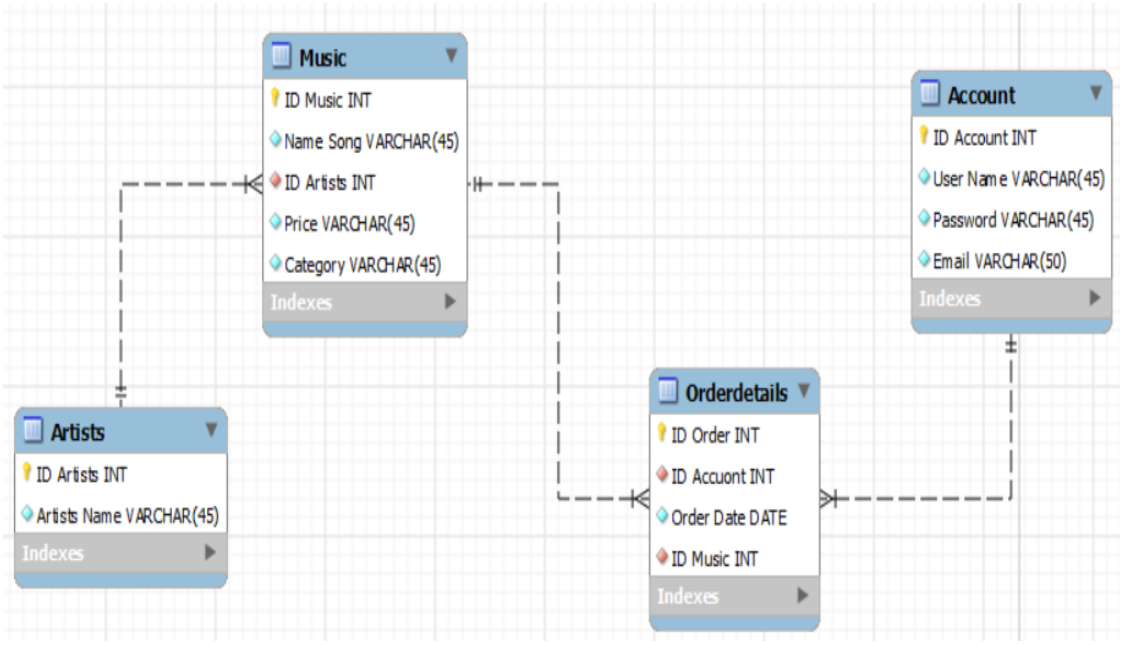
- **Context Diagram for the whole system**



- **Data Flow Diagram - Level 0 for the whole system**



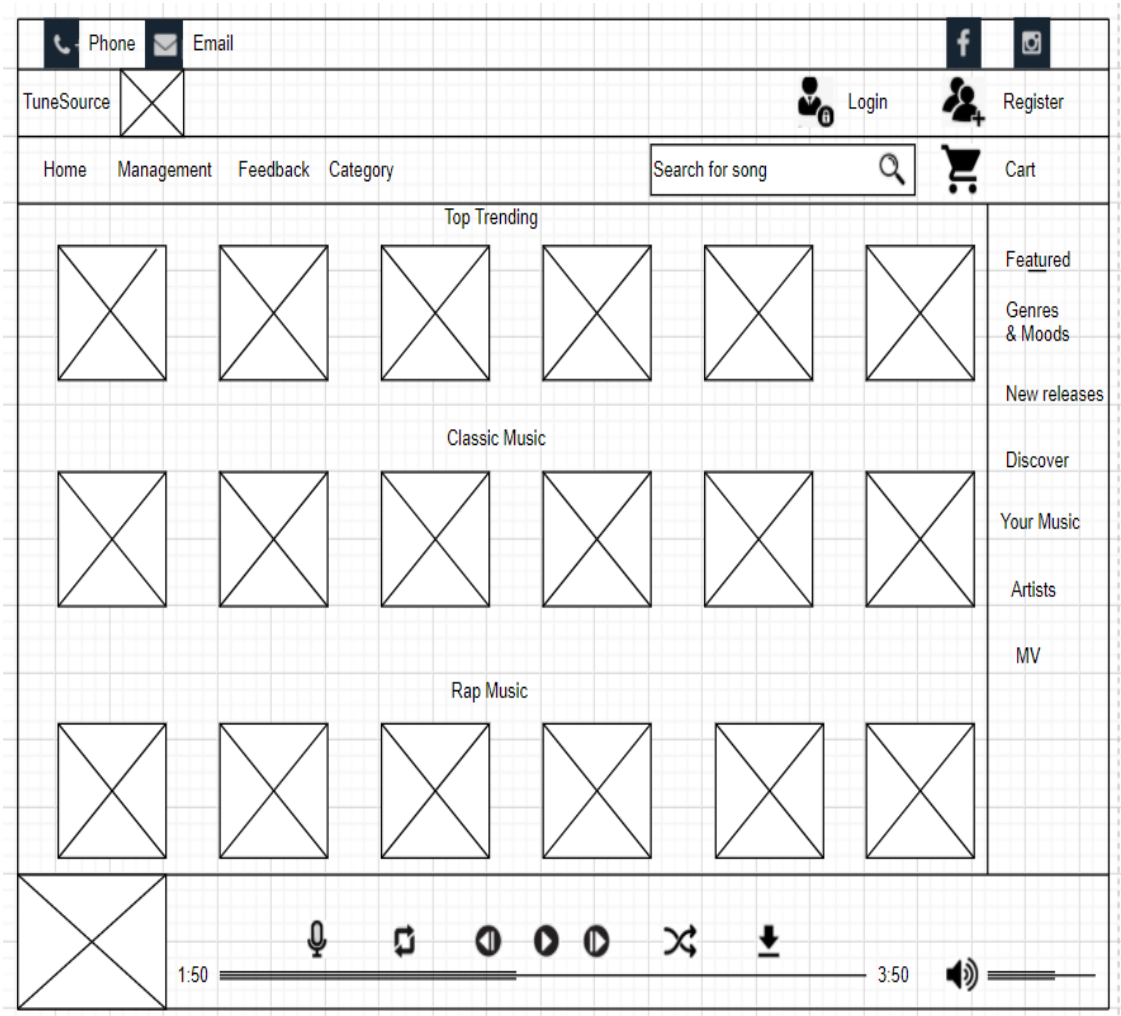
- **ERD for the whole system**



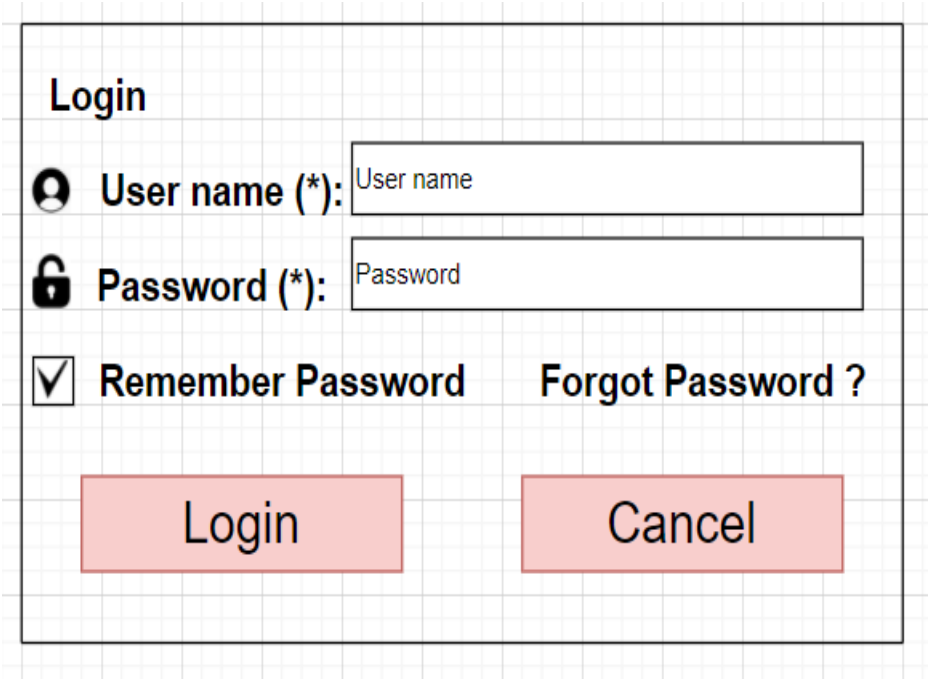
VII. Explain how user and software requirements have been addressed.

- In the Tune Source system, there are functions that can satisfy the user's requirements.
  - Register
  - Login
  - Logout
  - Update Profile
  - Search for song
  - Listen to music
  - Payment/ Download

- **Home Page**
  - Customer and admin can log in here, users can view products and register their accounts, listen to music, search for a song, Feedback and update information, but administrators can perform the function as add, edit, see feedback and delete music, account, and may perform the function as add, edit, and delete in "Management Music", "Management Account".

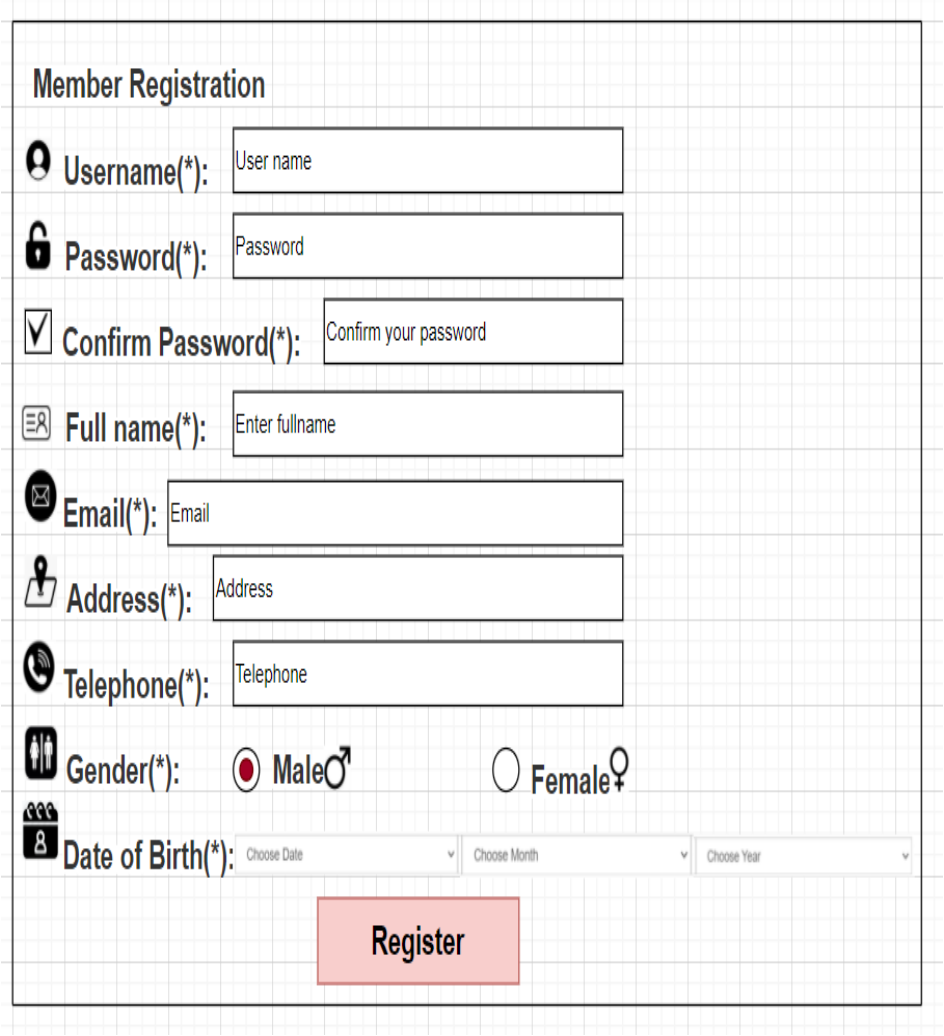


- **Login page**
  - The login page created for the purpose of accessing the web for both customers and administrators to access the site carry out the necessary issues.



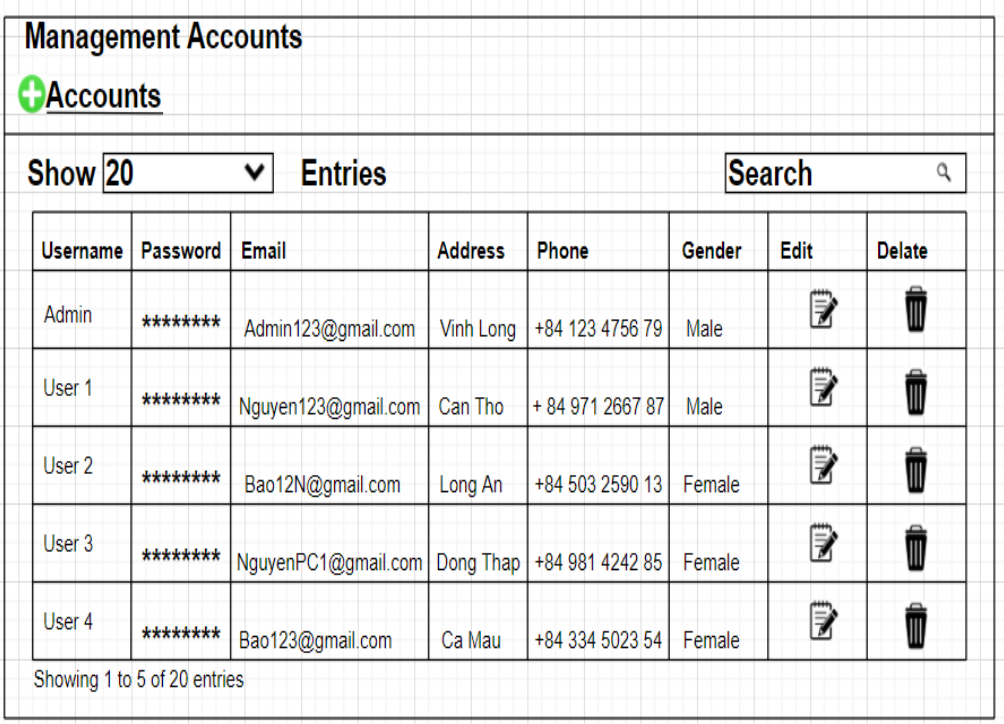
○ Register page

- In this register form, the customers must do input all personal information including Username, Password, Confirm Password, Full name, Address, Email, Telephone, Gender, and Date of birth.
- When customers input not enough one of the requests then the system will be sent the notification on the screen report they know the information not enough or the password not match



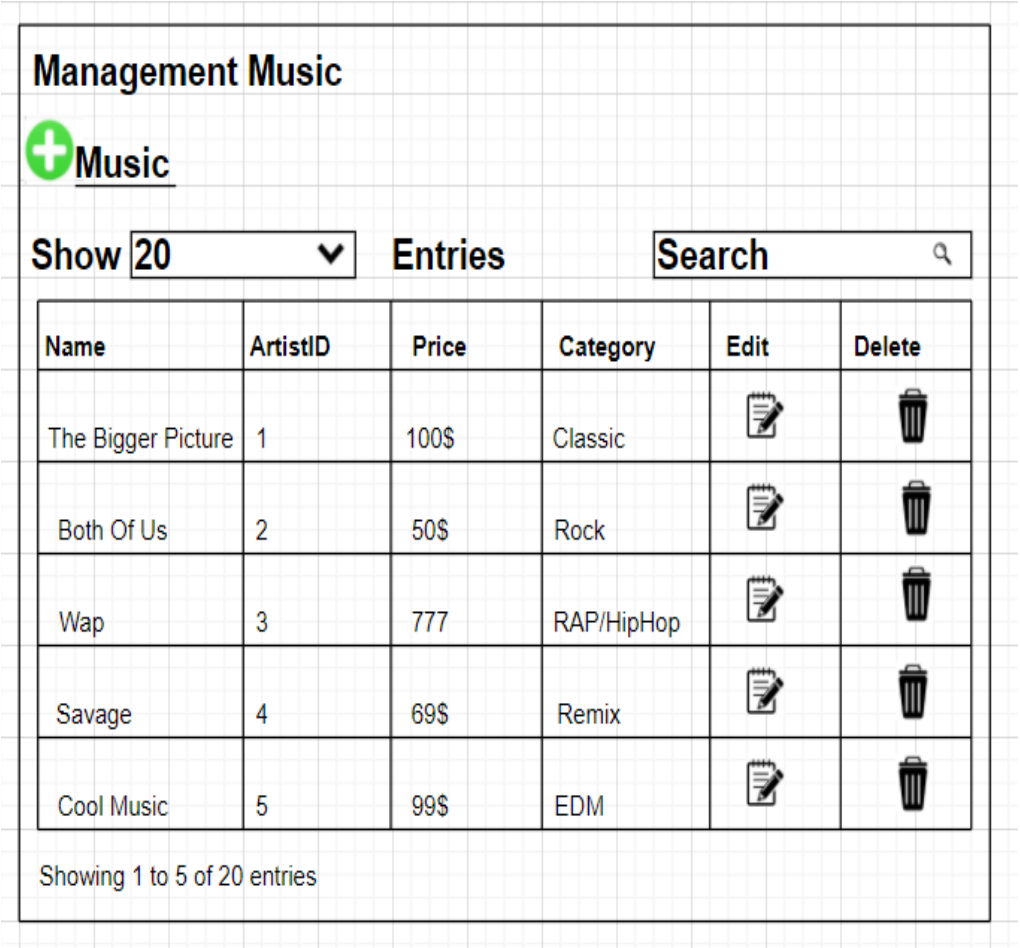
○ Management Accounts page

- With this account management page, only the admin and new authorized accounts can access the "account management page" and perform additional information, update or delete the customer account.



○ Management Music page

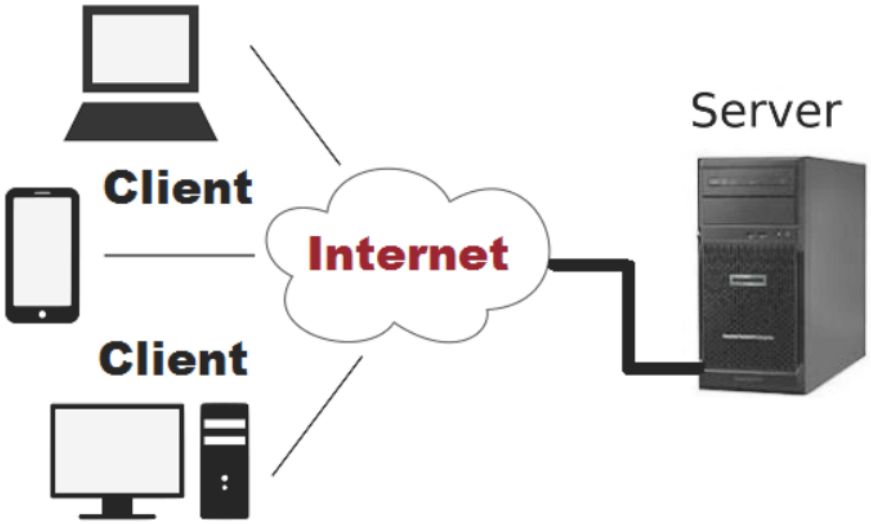
- With this music management page, only the admin and new authorized accounts can access the "music management page" and perform additional information, update add edit delete all types of music.



○ Client – server [2]

- **Definition**

Client-server architecture is a computing model in which the server hosts, delivers and manages most ofthe resources and services to be consumed by the client. This type of architecture has one or more client computers connected to a central server over a network or internet connection. Client-server architecture is also known as a networking computing model or client-server network because all the requests and services are delivered over a network.



- The main benefit of a three-tier architecture is that because each tier runs on its own infrastructure, each tier can be simultaneously developed by a separate development team and can be updated or extended when necessary without affecting the other floors. I choose the server architecture that is suitable for the project as the client-server architecture. In addition, other benefits include:
  - Client-server architectures balance the processing between client devices and one or more server devices.
  - The client is responsible for the presentation logic, whereas the server is responsible for the data access logic and data storage.
  - A thick or fat client contains all or most of application logic, whereas a thin client contains a small portion
  - Improved reliability: Outage crashes at one level are less likely to affect availability or performance for others.
  - Scalable: Any level can be expanded independently of other levels if needed.
  - Using middleware, various types of clients and servers may be served.
  - Easy to design all applications
  - Maximum user satisfaction
  - Implementation of Homogeneous Environment
  - Best performance
  - The presentation logic, device logic, and data processing logic should all be separate from one another.
  - If a server fails, only the applications requiring that sever are affected. The only major limitation of client-server architectures is their complexity.

Solution stack [3]

- **What does Solution Stack mean?**
  - A solution stack is a set of different programs or application software that are bundled together in order to produce a desired result or solution. This may refer to any collection of unrelated applications taken from various subcomponents working in sequence to present a reliable and fully functioning software solution. Many computer companies like Microsoft and Linux provide different solution stacks to clients.
  - Linux (operating system): Linux is reliable and secure. Programmers and developers are frequently fixing issues, which in turn lessens security risks.
  - Apache (webserver): Apache is open-source software, which ensures the original source code can be used and collaborated on for free.
  - MySQL: MySQL is extremely scalable, as seasonal demands fluctuate, resource use can be customized to minimize waste and maximize performance. Financial transactions performed by MySQL are secured by being treated as a unified entity.
  - PHP, JavaScript (programming languages): JavaScript is the client-side scripting language and PHP is the server-side scripting language. JavaScript is used client-side to check and verify client details and PHP is server-side used to interact with the database.
  - This is the only approach for TuneSource in the early stages of production and service. This solution stack, which includes PHP and SQL servers, is the best solution stack for early project web development. When the website and server have been up and running for a while, and the framework data has reached a point of completion and consistency, we will use the.NET programming language in conjunction with the solution stack to design the project's application.

References

[1]Ragunath, P.K., Velmourougan, S., Davachelvan, P., Kayalvizhi, S. and Ravimohan, R., 2010. Evolving a new model (SDLC Model-2010) for software development life cycle (SDLC). *International Journal of Computer Science and Network Security*, *10*(1), pp.112-119.

[2]Sinha, A., 1992. Client-server computing. *Communications of the ACM*, *35*(7), pp.77-98.

[3]Benke, I., Marković, B.E., Pavlović, I., Milošević, M. and Grbić, R., 2019, May. Software solution stack for data transfer on a frame grabber platform. In *2019 Zooming Innovation in Consumer Technologies Conference (ZINC)* (pp. 39-43). IEEE.