**ASSIGNMENT 02 FRONT SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number and title** | Unit 09: Software Development Life Cycle | | |
| **Submission date** | 7/9/2022 | **Date Received 1st submission** |  |
| **Re-submission Date** |  | **Date Received 2nd submission** |  |
| **Student Name** | Do Huu Duy | **Student ID** | GCC200018 |
| **Class** | GCC0903 | **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** | huuduy |

**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 Brief 02 (RQF)

## Higher National Certificate/Diploma in Business

|  |  |
| --- | --- |
| **Student Name/ID Number:** |  |
| **Unit Number and Title:** | **Unit 09: Software Development Life Cycle** |
| **Academic Year:** |  |
| **Unit Assessor:** |  |
| **Assignment Title:** | **Undertake a software development life cycle** |
| **Issue Date:** | **07/12/2020** |
| **Submission Date:** |  |
| **Internal Verifier Name:** |  |
| **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.   Submission:   * Students are compulsory to submit the assignment in due date and in a way requested by the Tutor. * The form of submission will be a soft copy posted on <http://cms.greenwich.edu.vn/>. * Remember to convert the word file into **PDF** file before the submission on CMS.   Note:   * The individual Assignment must be your own work, and not copied by or from another student. * If you use ideas, quotes or data (such as diagrams) from books, journals or other sources, you must reference your sources, using the Harvard style. * Make sure that you understand and follow the guidelines to avoid plagiarism. Failure to comply this requirement will result in a failed assignment. |
| **Unit Learning Outcomes:** |
| **LO3** Undertake a software development lifecycle.  **LO4** Discuss the suitability of software behavioral 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 analyze and design the software.  **Task 1 – Analysis (1)**   1. 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 Tune Source Project.  Discuss the relationships between the FRs and NFRs.  *Word limit: 300 – 400 words.*   1. 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. Discuss how you would trace these requirements throughout the project by using Requirement Traceability matrix. You will have to provide real usage of it.   *Word limit: 400 – 500 words.*  **Task 2 – Analysis (2)**  Analyze the requirements that you identified in Task 1 using a combination of structural and behavioral 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.   For each diagram, you will have to explain properly.  *Word limit: 1000 – 1200 words.*  **Task 3** **– Design**  Based on the analysis result, discuss how you would conduct the design phase:   1. Discuss how the user and software requirements are addressed in the design phase.    * You will explain how Mock-up, and Wireframe are used in the project. You should include some of the mockup or wireframe (at least 5) design of the Tune Source 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 technical solution stack could be suitable to implement the project with clear explanations. 2. Discuss how activity diagram and pseudocode are used to specify the software behavior. 3. Discuss how UML state machine can be used to specify the software behavior. Differentiate between FSM and extended FSM using the case study. 4. Discuss how the data-driven approach improves the reliability and effectiveness of software.   *Word limit: 800 – 1500.*  **Task 4 – Software quality management**   1. Discuss two software quality attributes that are applicable to the project. 2. Discuss two quality assurance techniques that can help improve the software quality in the project. 3. 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 (Assignment 02):** | | | |
| Learning Outcome | Pass | Merit | Distinction |
| **LO3** Undertake a software development lifecycle | **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** Analyze how software requirements can be traced throughout the software lifecycle.  **M4** Discuss two approaches to improving software quality. | **D3** Critically evaluate how the use of the function design paradigm in the software development lifecycle can improve software quality. |
| **LO4** Discuss the suitability of software behavioral design techniques | **P7** Explain how user and software requirements have been addressed. | **M5** Suggest two software behavioral 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. | **D4** Present justifications of how data driven software can improve the reliability and effectiveness of software. |

Table of Contents

[I. Undertake a software investigation to meet a business need 6](#_Toc113181154)

[1. Identify the stakeholders, their roles and interests in the case study 6](#_Toc113181155)

[1.1. Define requirement 6](#_Toc113181156)

[1.2. What is stakeholder 6](#_Toc113181157)

[1.3. Roles and interests in the case study 6](#_Toc113181158)

[1.4. Identify FRs and NFRs of Tune Source Project 6](#_Toc113181159)

[2. Discuss the technique(s) that would use to obtain the requirements 7](#_Toc113181160)

[II. Use appropriate software analysis tools/techniques to carry out a software investigation and create supporting documentation 9](#_Toc113181161)

[1. Use case diagram 9](#_Toc113181162)

[2. Use Case specification for 2 Use cases 10](#_Toc113181163)

[2.1. Purchase music download gift cards 10](#_Toc113181164)

[2.2. Purchase individual download 12](#_Toc113181165)

[3. Context Diagram for the whole system 14](#_Toc113181166)

[4. Data Flow Diagram – Level 0 for the whole system 14](#_Toc113181167)

[5. ERD for the whole system 15](#_Toc113181168)

[III. Explain how user and software requirements have been addressed 15](#_Toc113181169)

[1. Mock-up of the project 15](#_Toc113181170)

[1.1. Sign up function 15](#_Toc113181171)

[1.2. Login function 16](#_Toc113181172)

[1.3. Search music 17](#_Toc113181173)

[1.4. Listen to music 17](#_Toc113181174)

[1.5. Purchase individual downloads 18](#_Toc113181175)

[1.6. Purchase music download gift cards 18](#_Toc113181176)

[1.7. Subscribe function 19](#_Toc113181177)

[2. Explanation architecture in the project 19](#_Toc113181178)

[3. Address which technical solution stack could be suitable to implement the project with clear explanations 19](#_Toc113181179)

[IV. Analyze how software requirements can be traced throughout the software lifecycle 20](#_Toc113181180)

[V. Discuss two approaches to improving software quality 21](#_Toc113181181)

[VI. Suggest two software behavioral specification methods and illustrate their use with an example 21](#_Toc113181182)

[VII. Differentiate between a finite state machine (FSM) and an extended-FSM, providing an application for both 22](#_Toc113181183)

[1. What is finite state machine (FSM) 22](#_Toc113181184)

[2. Extended-finite state machine (EFSM) 23](#_Toc113181185)

[VIII. Critically evaluate how the use of the function design paradigm in the software development lifecycle can improve software quality 23](#_Toc113181186)

[IX. Present justifications of how data driven software can improve the reliability and effectiveness of software 24](#_Toc113181187)

[1. What is Data-Driven 24](#_Toc113181188)

[2. How Data-Driven can improve the reliability and effectiveness of software. 24](#_Toc113181189)

[X. Conclusion 24](#_Toc113181190)

[References 25](#_Toc113181191)

# I. Undertake a software investigation to meet a business need

## 1. Identify the stakeholders, their roles and interests in the case study

### 1.1. Define requirement

Project requirements are conditions or tasks that must be met to ensure the success or completion of a project. They give a clear picture of the work to be done. They are designed to align project resources with organizational goals. The benefits of effectively gathering project requirements include reduced costs, increased project success rates, more effective change management, and improved communication between stakeholders (Raynor, 2021)

Requirement of the Tune Source: There are 5 require function in the system

* Search for music in our digital music archive.
* Listen to music.
* Purchase individual downloads at a fixed fee per download.
* Establish a customer subscription account permitting unlimited downloads for a monthly fee.
* Purchase music download gift cards.

### 1.2. What is stakeholder

A stakeholder is a party that has an interest in the company for a project and can influence or be affected by the project's business. The main stakeholders of a typical company are investors, employees, customers, and suppliers (FERNANDO, 2022)

### 1.3. Roles and interests in the case study

For the Tune Source project, the stakeholder includes users and singers. For the users, they will use the access to the website of Tune Source to register an account and they can listen to music online on there. In addition, when they register an account success, they can log in and search the music and download the music that they like but they must pay fees. Besides, they can buy some gift card music to send to their relatives. However, to perform the above things they have to register an account and have money in their account. For singers, they will be the person that supplies the songs for the Tune Source project. Besides, when the Tune Source project gets any music, they must be allowed by the owner of that music. In addition, they can buy the copyright of that music. To sum up, the users they can access the Tune Source website to register an account and log in, they can listen to music online, download music, buy gift card music when their account have money. On the other hand, the singers will the person that supplies songs for the Tune Source and the Tune Source may buy the copyright of them.

### 1.4. Identify FRs and NFRs of Tune Source Project

1.4.1 Define FR and NFR

**FR:** A Functional Requirement (FR) is a description of the services that the software must provide. Describes a software system or its components. Functions are nothing more than inputs to a software system, its actions, and outputs. This could be a calculation, data manipulation, business process, user interaction, or some other specific function that defines what the system is expected to do. Functional requirements in software engineering are also called functional specifications (Martin, 2022)

**NFR:** Non-Functional Requirements (NFRs) are a set of specifications that describe the operational capabilities and limitations of a system and seek to enforce its functionality. These are basically requirements that describe good performance, including speed, security, reliability, data integrity, and so on (altexsoft, 2022)

1.4.2 Function Requirements (FRs) and Non-Function Requirements (NFRs) of Tune Source Project

**Function Requirements (FRs):** For the Tune Source project, the functional requirement (FR) is to describe a software system and its components. The software of the Tune Source project is performing the listen to music online website with some functions such as the user can register an account and log in with that account. In addition, users can purchase individual download music, listen to music online, and share gift card music with the constraints that users must have an account and that account must have money. This project will use the Symfony framework comes with PHP languages programming to develop the project.

To sum up, in this project will have some the following function:

* Search for music in our digital music archive.
* Listen to music.
* Purchase individual downloads at a fixed fee per download.
* Establish a customer subscription account permitting unlimited downloads for a monthly fee.
* Purchase music download gift cards.

**Non-Function Requirements (NFRs):** For the Tune Source project, the non-functional requirement is to perform checking about performance, including speed, security, reliability, and data integrity in the project. Checking the specifications such as the server of the system while performing the project. Besides, identify about operational capabilities and limitations of a system, and check whether the specifications used in the project are suitable or not.

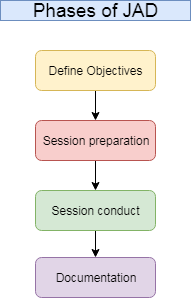
1.4.3 Relationship of Function Requirements (FRs) and Non-Function Requirements (NFRs)

FR and NFR have a relationship intimately with each other. For a project that can be well performed, the FR and NFR in the project cannot lack. Before performing a project, the FR is the first work that we need to perform, we need to collect requirements from the users to know in the project what need we to do, so FR in a project is important. Besides, NFR cannot lack in a project, when we collect the FR from the users, the next work that we need to do is to check the system and specifications whether it is suitable for the project or not. When we perform a project but we ignore one of two of these steps, the project will get lots of risks. For example, if we don't perform functional requirements (FR) from users, the project may be wrong about function and how we know the project will perform which is the function, whether the function that we performing is suitable for the project or user's requirements. On the other hand, non-functional requirements (NFR) too. If we don't perform this, how do we know whether the system and specifications are suitable for the project and the risk is given when we use this technology for the project? To perform a project, we need to get requirements from users, and from that, we will consider the specifications to apply for the project. To sum up, these two functions have a relationship intimately and are supported in a project.

## 2. Discuss the technique(s) that would use to obtain the requirements

There are some techniques to collect the functional requirements of users such as interview, observation, and JAD

* **Interview technique:** Interviews are the initial requirements-gathering technique. This can be done through a meeting, phone call, or email. In this requirement-gathering technique, project managers interview stakeholders to obtain requirements. Checklists and surveys can be prepared, and project managers can freely express their expectations of the project to stakeholders. Project managers record and store requirements received from project stakeholders (masterofproject, 2022)
* **Observation technique:** A requirements gathering technique called observation involves observing potential users of the product to identify their requirements. For example, consumers can be observed to determine the user experience and most used features of an e-commerce shopping website. You can identify or prioritize project needs based on the steps consumers take (masterofproject, 2022)
* **Joint Application Development (JAD):** Collaborative application development is primarily used for gathering, designing, and developing software requirements. This method consists of holding a series of joint meetings, known as JAD sessions, which can last from several hours to several weeks. The picture 1 is the stages of JAD (Zehra, 2022)

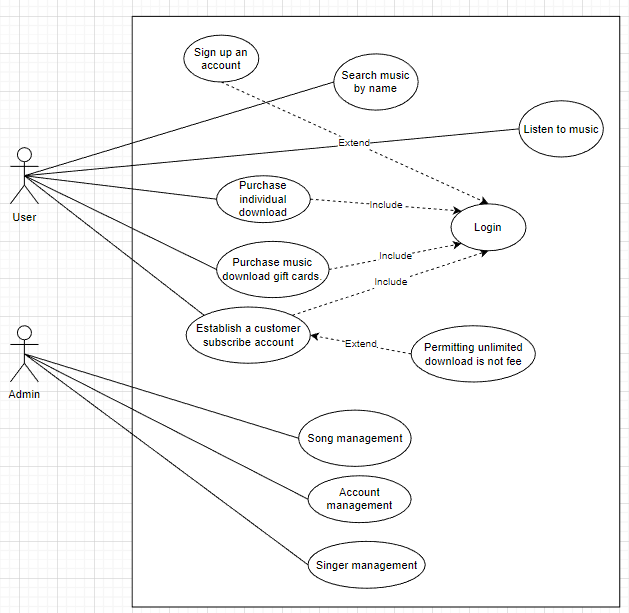


**Image 1. The stages of JAD (study, 2021)**

For the Tune Source project, I will choose the interview technique to perform collection requirements from users. I will perform some meet to talk about the project and get ideas from stakeholders. In addition, I will send emails and call the phone users to perform the surveys and collect their expectations of the project stakeholders. To sum up, in the Tune Source project, I will choose the interview technique to perform surveys and collect requirements from users and stakeholders.

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

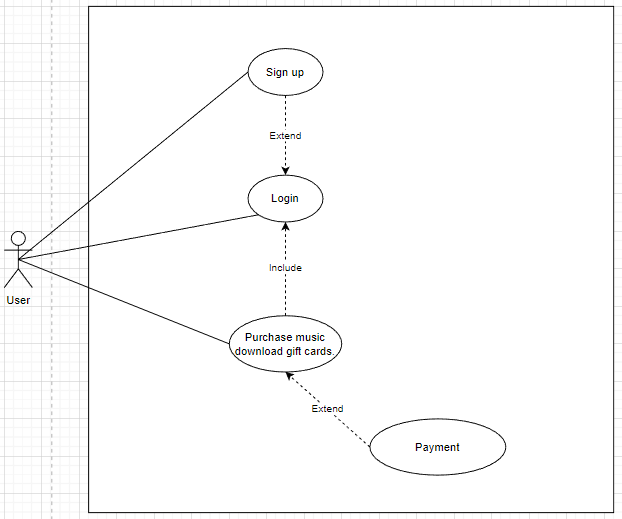
## 1. Use case diagram



**Image 2. General use case**

## 2. Use Case specification for 2 Use cases

### 2.1. Purchase music download gift cards

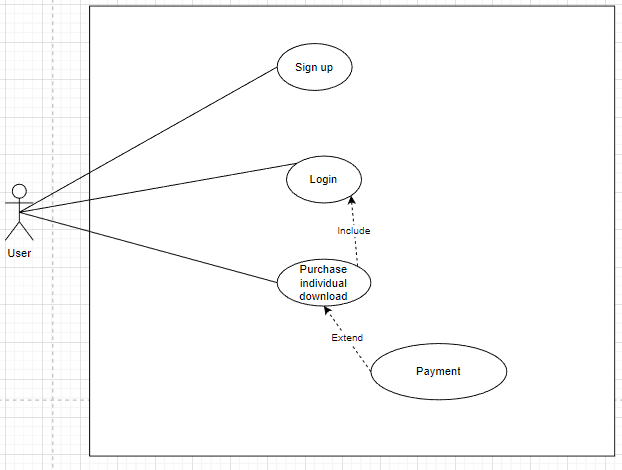


**Image 3. Purchase music download gift card use case**

**Table 1. Purchase music download gift cards**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case Name: Purchase download music gift card | | | ID: UC 2 | | Priority: High |
| Actor: User | | | | | |
| Description: User can log in with their account and purchase download music gift cards | | | | | |
| Trigger: The user needs purchase download music gift cards  Type:  External  Temporal | | | | | |
| Preconditions:  1. The user needs purchase download music gift cards  2. The user wants to send the music gift card to their friend or relative  3. User’s account must have money  4. Available music  5. User must be login successfully | | | | | |
| Normal Course:  1. User need to sign up an account if they don’t have account  2. User need to log in with the account that they signed up  3. User need to click on “Music” on navbar link to go to Music page  4. User will choose the song that they want to send  5. User need to click on “Purchase music gift cards” on navbar link  to go to Purchase music gift card page  6. User need to fill information to all field in the form with  “Name of music”, “Receiver email”, “Receiver phone”, and the message that user wants to send  7. Click on “Send” button to send the gift card music to the person who the user wants to send | | | Information for steps:  Sign up form  Login form  Switch to music page  List of music  Switch to purchase download music gift card page  Send gift card music form | | |
| Alternative Course:  1. The system will require users to recharge money if user accounts do not have money to purchase music  2. The system will require users to decrease songs if the money in their account of users not enough to purchase music  3. The system will require cancel if the user account do not enough money to perform “Purchase download music gift card” function | | | Switch to recharge page  Decrease songs  Cancellation | | |
| Postconditions:  1. After the user purchase individual download, money in their account will be decrease  2. The number of music gift card purchases will increase | | | | | |
| Exceptions:  1. The system will display message “Insufficient balance”  2. The system will switch to login form if the user hasn’t logged in yet  3. The system will display message “Updating song” | | | | | |
| Summary Inputs | Source | Outputs | | Destination | |
| Information to sign up  Enter the account that signed up to login  Click on “Music” on navbar link  Click on “Purchase download music gift cards” on navbar link  Fill all information to send the gift card music  Cancellation | User  User  System  System  User | Display message “Sign up successfully”  Display message “Login successfully”  Display music page  Display purchase download music gift cards page  Display form to fill information of gift card and send | | Login  Home  Music page  Purchase download music gift card page  The form to send gift cards music | |

### 2.2. Purchase individual download

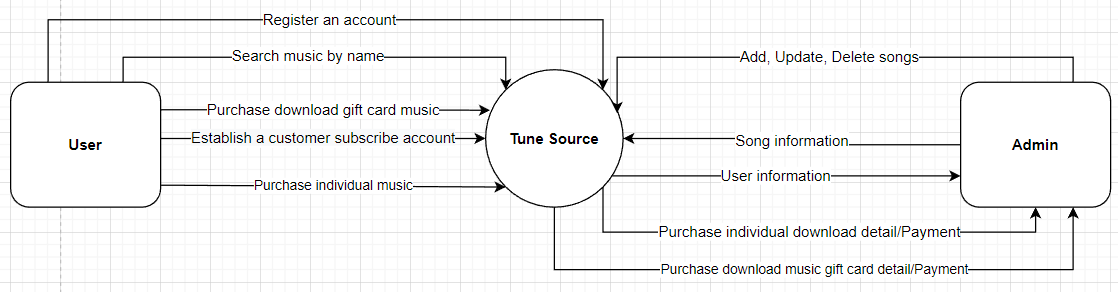


**Image 4. Purchase individual download use case**

**Table 2. Purchase individual download**

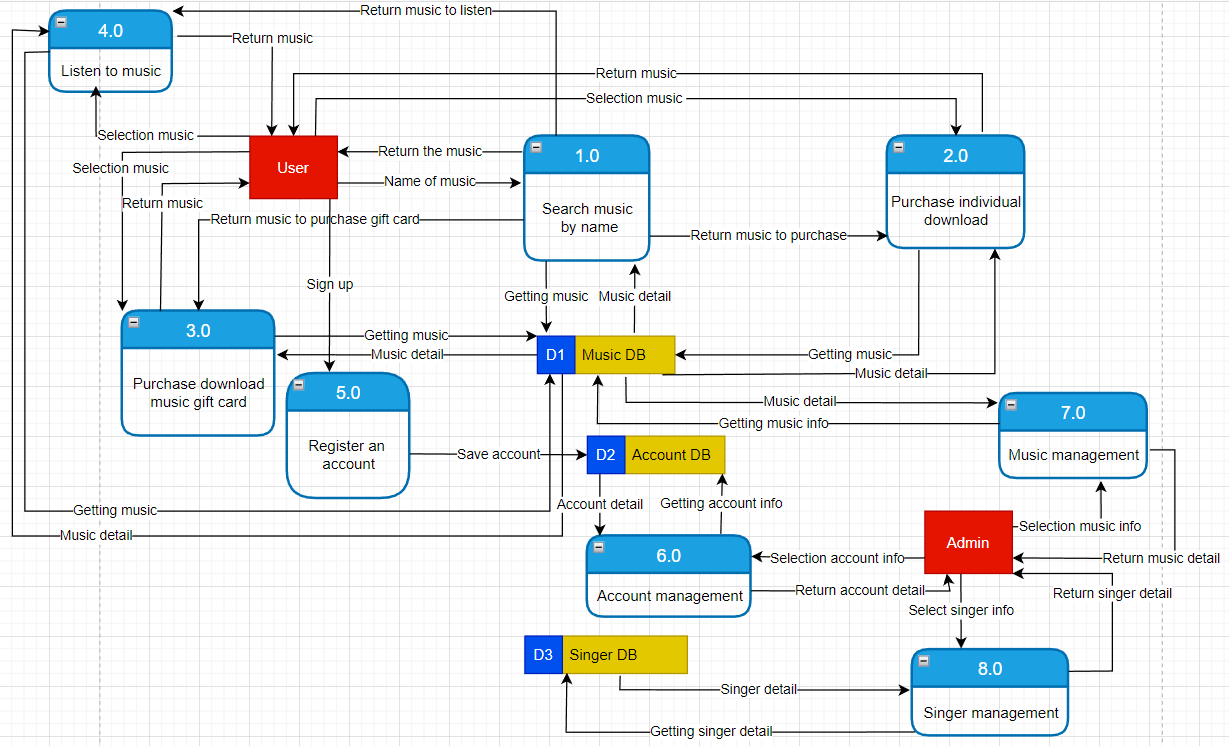
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case Name: Purchase individual download | | | ID: UC 3 | | Priority: High |
| Actor: User | | | | | |
| Description: The user will register an account and login with that account, then they can purchase individual download and payment for that music | | | | | |
| Trigger: The user needs purchase individual download  Type:  External  Temporal | | | | | |
| Preconditions:  1. The system will display message “Insufficient balance”  2. The system will display message “Do not have money in your accounts”  3. The system will switch to login form if the user hasn’t logged in yet  4. The system will display message “Updating song”  5. User must be login successfully | | | | | |
| Normal Course:  1. User need to sign up an account if they don’t have account  2. User need to log in with the account that they signed up  3. User need click on “Music” on navbar link to go to Music page  4. User will choose the song that they want to listen and download  5. User need to click on the price of each song to purchase and download them | | | Information for steps:  Sign up form  Login form  Switch to music page  List of music | | |
| Alternative Course:  1. The system will require users to recharge money if user accounts do not have money to purchase music  2. The system will require users to decrease songs if the money in their account of users not enough to purchase music  3. The system will require cancel if the user account do not enough money to perform “Purchase individual download” function | | | Switch to recharge page  Decrease songs  Cancellation | | |
| Postconditions:  1. After the user purchase individual download, money in their account will be decrease  2. The number of music gift card purchases will increase | | | | | |
| Exceptions:  1. The system will display message “Insufficient balance”  2. The system will display message “Do not have money in your accounts”  3. The system will switch to login form if the user hasn’t logged in yet  4. The system will display message “Updating song” | | | | | |
| Summary Inputs | Source | Outputs | | Destination | |
| Information to sign up  Enter the account that signed up to login  Click on “Music” on navbar link  Click on the price of song to purchase and download  Cancellation | User  User  System | Display message “Sign up successfully”  Display message “Login successfully”  Display music page | | Login  Home  Music page | |

## 3. Context Diagram for the whole system



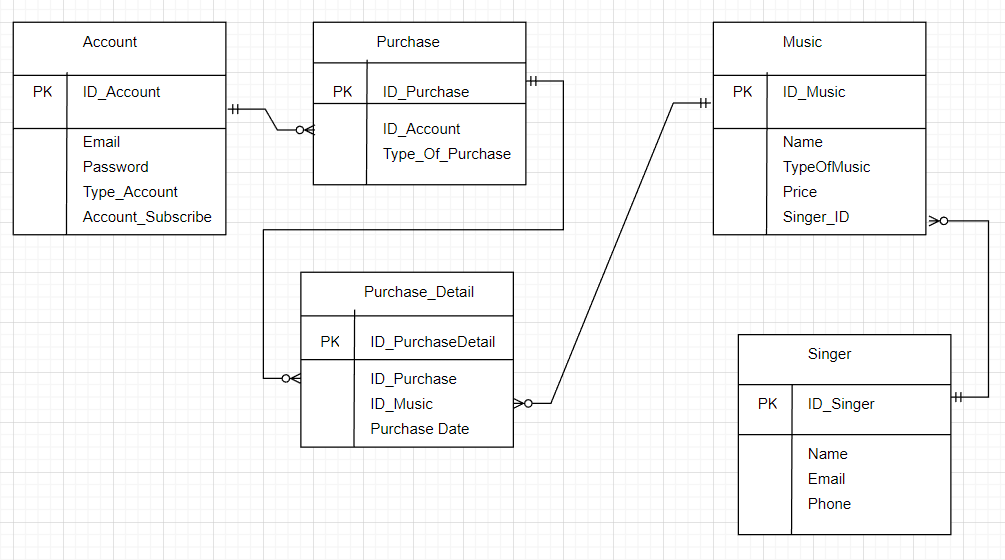
**Image 5. Context diagram**

## 4. Data Flow Diagram – Level 0 for the whole system



**Image 6. Data flow diagram - level 0**

## 5. ERD for the whole system



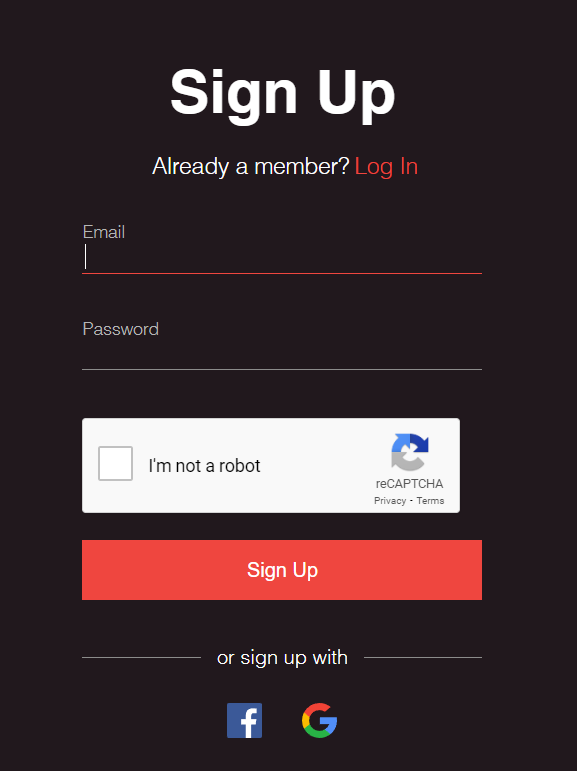
**Image 7. Entity relationship diagram**

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

## 1. Mock-up of the project

### 1.1. Sign up function

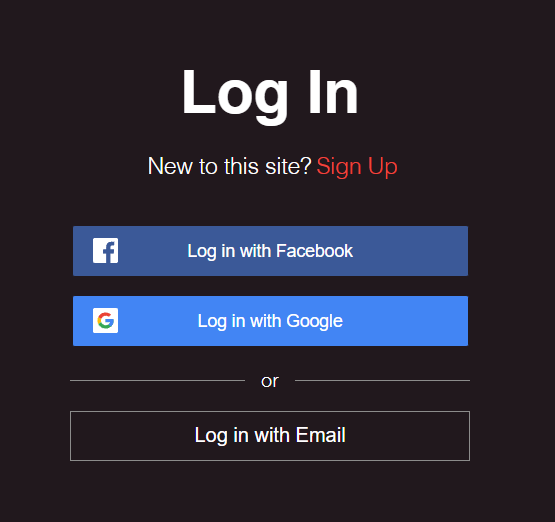
User can sign up an account by Email and Facebook.



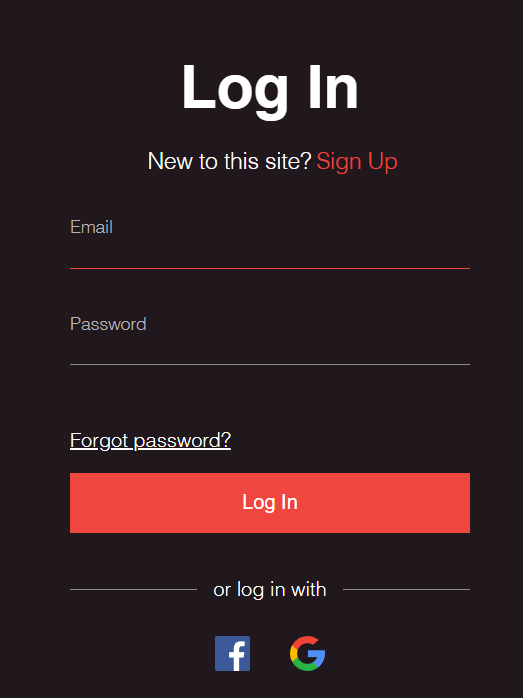
**Image 8. Sign up function**

### 1.2. Login function

User can login with the account that they registered.



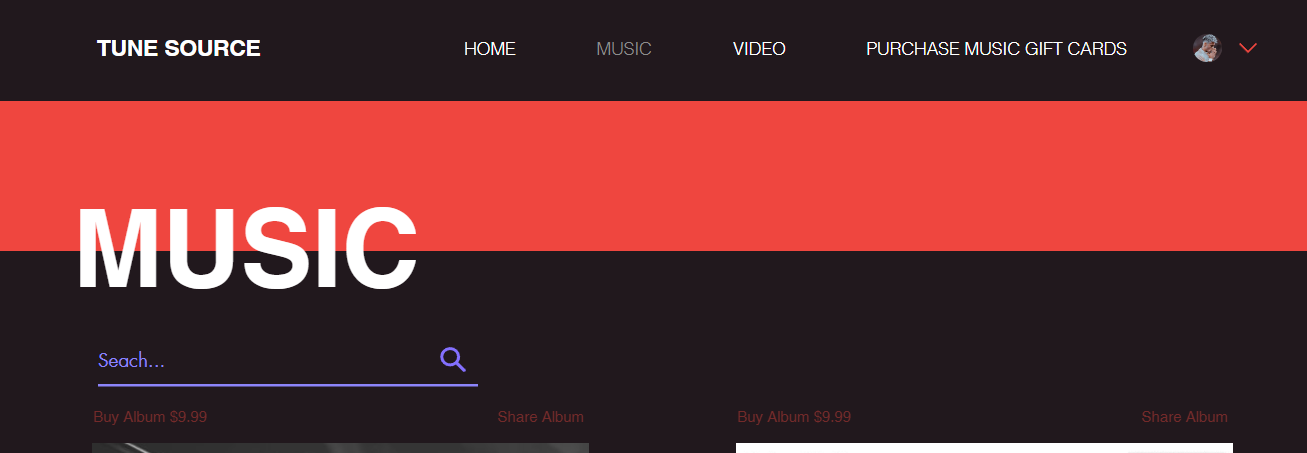
**Image 9. Login function with Facebook or Google**



**Image 10. Login function with Email**

### 1.3. Search music

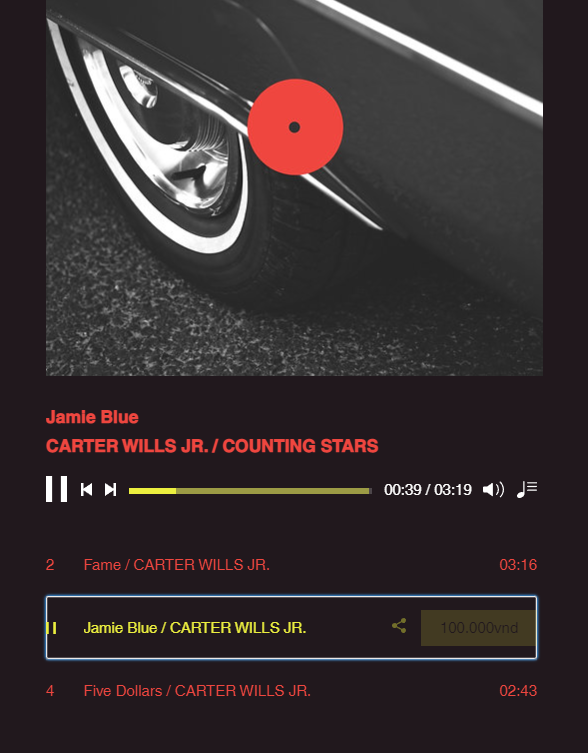
User can text the name of music in search text and click on search button to find the music that they want to search.



**Image 11. Search function**

### 1.4. Listen to music

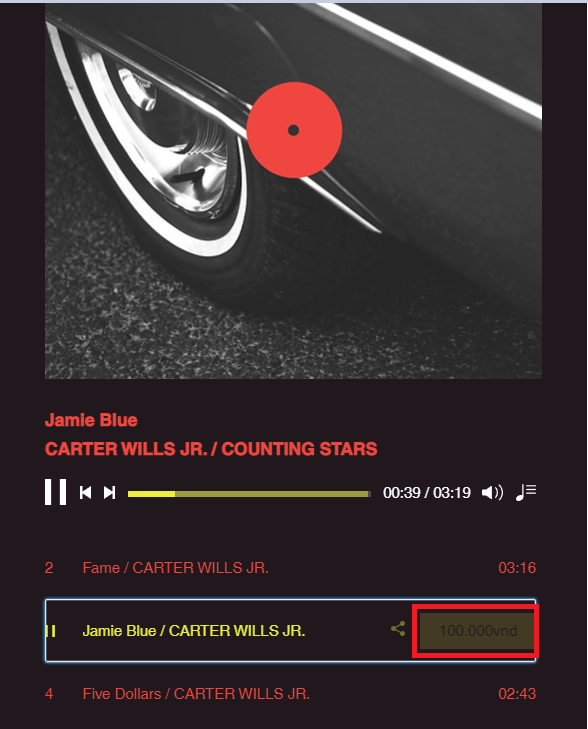
There are the list of music and user can listen to music online by click on the music that they want to listen. The picture is a description about listening to music online.



**Image 12. Listen to music**

### 1.5. Purchase individual downloads

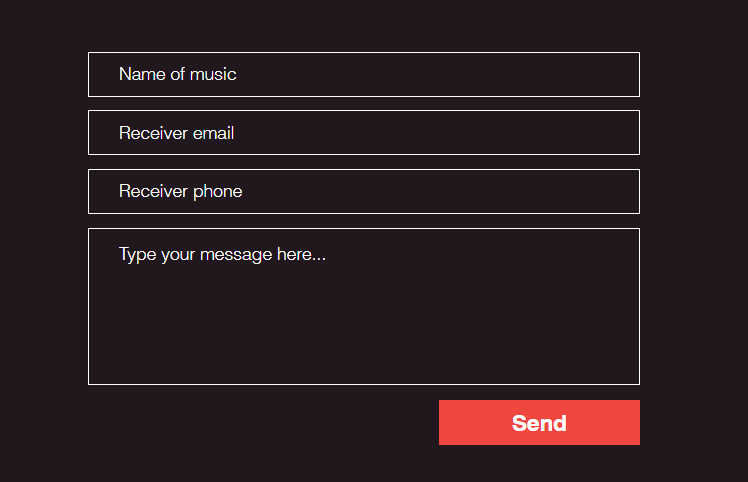
There is a price of music in each music and the user can purchase individual downloads by clicking on the price of each music in the list of music. The picture is a description about the purchase individual download function.



**Image 13. Purchase individual download function**

### 1.6. Purchase music download gift cards

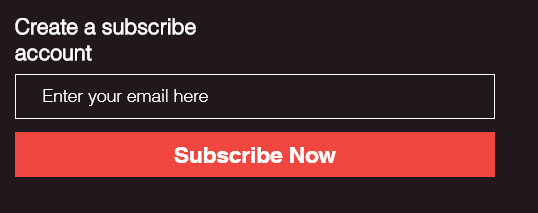
User can share gift card music by fill in the form with fields such as “Name of music”, “Receiver email”, “Receiver phone”, and “Type your message here”, then they must click on “Send” button to send the gift card music that they want to share to the person that they want to share. The picture is a description about share gift card music function.



**Image 14. Purchase music download gift card function**

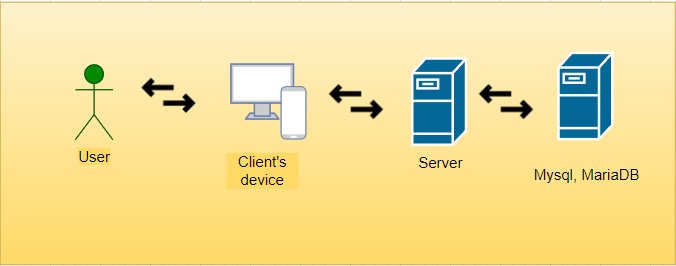
### 1.7. Subscribe function

Users can create a subscribe account to download music unlimited that isn’t fee. Users need to use the email that they signed up account before to perform create a subscribe account.



**Image 15. Subscribe function**

## 2. Explanation architecture in the project



**Image 16. Client-server model**

In this project, I will use the Client-server model because when I use this model, the data will be saved in a single place and less maintenance cost and data recovery are possible. Ensure security because the server is centralized in one place and easy to maintenance. In addition, the users can connect to access and use data easily without having to be near the server. Besides, this is a famous model and flexible, easy to use, and applied in a lot of projects. So, I think this model will be suitable with the Tune Source project and I will use it.

## 3. Address which technical solution stack could be suitable to implement the project with clear explanations

In this project, I will use the Symfony framework with PHP programing language. This is a famous framework and quite a development today. Besides, in numerous web projects, the programmer often uses this framework a lot because this framework is easy to use and has a frame available built. Moreover, symphony frame is a framework with a lot of built-in libraries, it will help writing code quickly and easily. We will be easy to maintain code and this framework is suitable some long-term project like Tune Source project. In addition, in this project, I will perform with the MVC (Model-View-Controller) model and the data of the project will save to MySQL. MySQL is a data base that is used a lot in web projects, it is easy to use. Moreover, MySQL is reliable and safe.

# IV. Analyze how software requirements can be traced throughout the software lifecycle

**Table 3. Requirement Traceability Matrix**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement Traceability Matrix | | | | | | |
| Project Name: Tune Source | |  | | | | |
| Business Requirement Document (BRD) | | Functional Requirement Document (FRD) | | Test Case Document | | |
| BR\_ID | BR use case | FR\_ID | FR use case | Priority | Test case | Status |
| BR\_1 | Listen to music | FR\_1 | Listen to music | High | Users can lick on the song that they want to listen and they can listen the song that they choose | Passed |
| BR\_2 | Search music | FR\_2 | Search music | High | Users can search music by name, when the users enter name of the song that they want to search in to text box search and click on button, the song that they search will display | Passed |
| BR\_3 | Purchase individual download | FR\_3 | Purchase individual download | High | Users can purchase individual download, when users log in with their account then they can purchase that they want to purchase and their account must have enough money | Passed |
| BR\_4 | Share music download gift card | FR\_4 | Share music download gift card | High | Users can share music download gift card to send for people that they want to share, they have to log in and then they will choose the song that they want to share and purchase them to share when the account have enough money | Passed |
| BR\_5 | Establish a customer subscription account | FR\_5 | Establish a customer subscription account | High | Users can establish a customer subscription account to be allowed unlimited downloads | Passed |

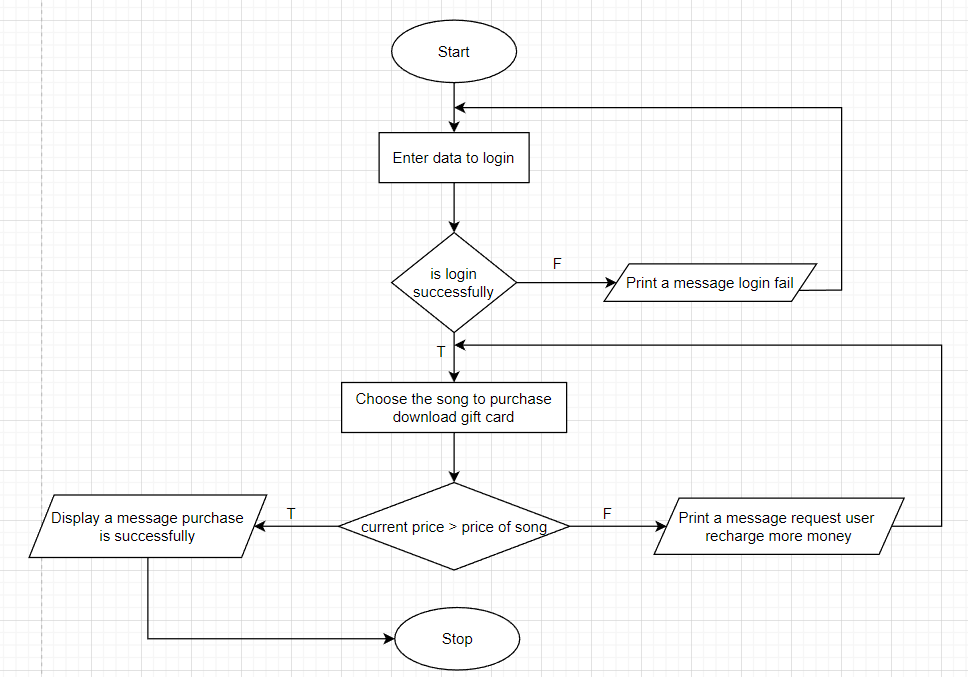
# V. Discuss two approaches to improving software quality

There are two approaches to improving software quality that I think it will be suitable with the Tune Source project this is “Test Early Stage” and “Implement Quality Assurance”.

* **Test early:** Testing early and often is key to improving software quality. Early testing ensures that errors do not become major problems. The more serious the defect, the more expensive it will be to fix. We will make the software quality better when we perform tests early to detect risks and solve them. Involving testers early in the software design process ensures that any problems or bugs that arise are addressed and fixed before the problem evolves exponentially (forcebolt, 2021)
* **Implement quality assurance:** Quality assurance is a process that I think must be present at all times during improve and development of software. We need to check and monitor that the techniques used to deliver outcomes have been tracked and are working correctly. This will help us capture the progress of software development and whether the techniques we are using are guaranteed or not. Thereby helping us to improve the software better (forcebolt, 2021)

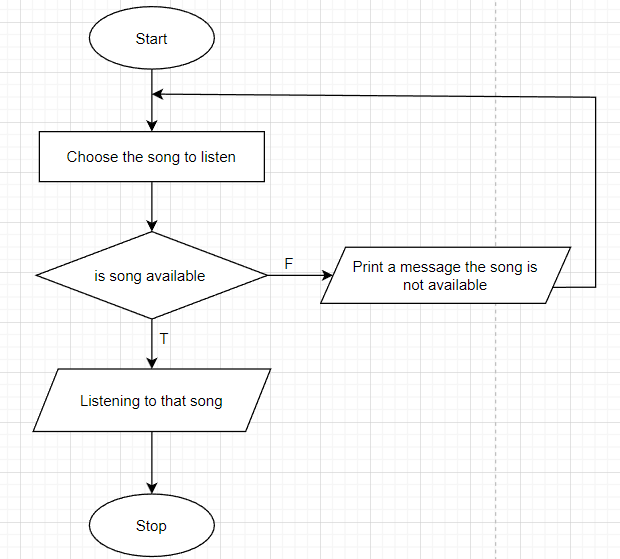
# VI. Suggest two software behavioral specification methods and illustrate their use with an example

* Purchase music download gift cards



First the user needs to login by their account. If they login fail the system will display a message login fail and they have to login again and when they login successfully, then they will choose the song that they want to purchase gift card and they must ensure that their money in their account must have enough or larger the price of the song that they want to purchase. If the money in their account not enough they have to recharge more money to purchase.

* Listen to music

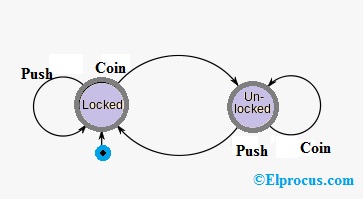


In this function, the user will choose the song that they want to listen and listen that song, but the song that they choose not available in the system, the system will display a message the song is not available and the user have to choose different song to listen.

# VII. Differentiate between a finite state machine (FSM) and an extended-FSM, providing an application for both

## 1. What is finite state machine (FSM)

Finite State Machine (FSM) is the term Finite State Machine (FSM), also known as Finite State Automation. FSM is a computational model that can be implemented using hardware or software. It is used to write sequential logic and some computer programs. FSMs are used to solve problems in fields such as mathematics, games, linguistics, and artificial intelligence. A system in which certain inputs can cause certain state changes is characterized using FSM. For example, the following state machine diagram describes the different states of a turnstile. Inserting a coin into the turnstile unlocks it, and pressing the turnstile locks it. Inserting a coin into an unlocked turnstile or pushing an otherwise locked turnstile does not change its state (elprocus, 2022)



* Advantages of finite state machine (FSM)
* Finite state machines are flexible
* Easy to move from a significant abstract to a code execution
* Low processor overhead
* Easy determination of reachability of a state
* Disadvantage of finite state machine (FSM)
* The expected character of deterministic finite state machines can be not needed in some areas like computer games
* The implementation of huge systems using FSM is hard for managing without any idea of design.
* Not applicable for all domains
* The orders of state conversions are inflexible.

## 2. Extended-finite state machine (EFSM)

# VIII. Critically evaluate how the use of the function design paradigm in the software development lifecycle can improve software quality

In this program, I perform 5 functions such as listening to music, searching for music, purchasing music download gift cards, purchasing individual download music, and establishing a subscribed account to download music unlimited these five functions have some impact on the system such as the system must have a big place to save data of songs to serve the functions of the system like search, listen, and purchase music. Besides, the system also saves the data of the user's account so the system must have a big place to save. I used MySQL to save the data I think, this is a database that will be a little inconvenient when must save big data. However, with the search for music and listen to music function the system will perform this function well because the music is saved in a single place and when the user performs these two functions the system will be easy to get data and respond to the user fast. In addition, two functions are purchase download gift card and purchase individual music the system will operate quite slowly because for the case that the money in the user's account do not enough to purchase the song the system will transfer to the recharge money page, to the user can recharge money in their account to purchase music. With the five above functions, I think these is some impacts that may happen with the system.

# IX. Present justifications of how data driven software can improve the reliability and effectiveness of software

## 1. What is Data-Driven

Data-driven is consumed to refer to a process or activity that is driven by data rather than just intuition or personal experience. In other words, decisions are based on solid empirical evidence rather than guesswork or instinct. The term is used in many fields, but most commonly in engineering and business (techopedia, 2018)

## 2. How Data-Driven can improve the reliability and effectiveness of software.

Data-driven essentially means that data dictates the actions of the people performing the events or processes. This is most evident in the field of big data, where data and information are the basis of all action, and data collection and analysis are central motivations. With data becoming easier to collect and less expensive to store, big data analytics has emerged as the business world's premier decision-making tool. Having so much data gives us powerful insights into the world, so we can manipulate the results. Projects using data-driven models are therefore more reliable and flexible and help the system operate effectiveness (techopedia, 2018)

# X. Conclusion

In this subject, I learned how to develop a system, and understand the steps to perform the development of a system by the structure correctly and logic. Understand the importance of developing a system and the impact that it gives and the know-how to draw a matrix diagram to determine some issues of the project. For example, in the Tune Source project during performing the project I understand and realize the importance when performing a life cycle development of the system and realize my own shortcomings and improved them. Understand how to set a plan to develop a system sequentially and structured appropriately. Know how to choose the right tools and techniques for the project to develop the system's effectiveness and suitability for the project's budget. To sum up, these are things that I learned and conclude when learning the Software Development Life Cycle subject and during perform develop a system like Tune Source.

# References

altexsoft, 2022. *altexsoft.com.* [Online]   
Available at: https://www.altexsoft.com/blog/non-functional-requirements/  
[Accessed 6 8 2022].

elprocus, 2022. *elprocus.com.* [Online]   
Available at: https://www.elprocus.com/finite-state-machine-mealy-state-machine-and-moore-state-machine/  
[Accessed 3 9 2022].

FERNANDO, J., 2022. *investopedia.com.* [Online]   
Available at: https://www.investopedia.com/terms/s/stakeholder.asp  
[Accessed 4 8 2022].

forcebolt, 2021. *forcebolt.com.* [Online]   
Available at: https://www.forcebolt.com/11-ways-to-improve-software-quality/  
[Accessed 31 8 2022].

Martin, M., 2022. *guru99.com.* [Online]   
Available at: https://www.guru99.com/functional-requirement-specification-example.html#:~:text=A%20Functional%20Requirement%20(FR)%20is,%2C%20its%20behavior%2C%20and%20outputs  
[Accessed 6 8 2022].

masterofproject, 2022. *masterofproject.com.* [Online]   
Available at: https://blog.masterofproject.com/collect-requirements-process/  
[Accessed 6 8 2022].

Raynor, G., 2021. *study.com.* [Online]   
Available at: https://study.com/academy/lesson/project-requirements-definition-types-process.html  
[Accessed 13 8 2022].

study, 2021. *study.com.* [Online]   
Available at: https://study.com/academy/lesson/joint-application-development-definition-phases-methodology.html  
[Accessed 6 8 2022].

techopedia, 2018. *techopedia.com.* [Online]   
Available at: https://www.techopedia.com/definition/18687/data-driven  
[Accessed 4 9 2022].

Zehra, Z., 2022. *educative.io.* [Online]   
Available at: https://js.educative.io/answers/what-is-joint-application-development-jad  
[Accessed 6 8 2022].