


## ASSIGNMENT 01 FRONT SHEET

<b>Qualification</b>	<b>BTEC Level 5 HND Diploma in Computing</b>		
<b>Unit number and title</b>	Unit 09: Software Development Life Cycle		
<b>Submission date</b>	04 – July – 2022	<b>Date Received 1st submission</b>	
<b>Re-submission Date</b>		<b>Date Received 2nd submission</b>	
<b>Student Name</b>	Trinh Duc Anh	<b>Student ID</b>	GCH210829
<b>Class</b>	GCH1002	<b>Assessor name</b>	Do Tien Thanh
<b>Student declaration</b> 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.			
		<b>Student's signature</b>	

### Grading grid

P1	P2	P3	P4	M1	M2	D1	D2

☐ **Summative Feedback:**

☐ **Resubmission**

**Feedback:**

**Grade:**

**Assessor Signature:**

**Date:**

**Internal Verifier's Comments:**

**Signature & Date:**

## Contents

I.	Introduction .....	5
II.	Analysis .....	5
1.	Requirement definition.....	5
2.	Stakeholders who can make functional and non-functional requirements and their interests.....	6
3.	Functional and non-functional requirements of the project.....	6
4.	Techniques used to gather requirements.....	9
4.1.	Interviews.....	9
4.2.	Join Application Development .....	9
4.3.	Questionnaires.....	10
4.4.	Document Analysis.....	10
4.5.	Observation.....	10
4.6.	Technique chosen .....	10
5.	Interview questions .....	11
6.	Use-case .....	12
6.1.	Signs up for a subscription .....	12
6.2.	Buys a gift card .....	13
7.	Data flow diagram.....	14
7.1.	DFD 0.....	14
7.2.	DFD 1 .....	15
8.	Entity relationship diagram.....	19
9.	Flowchart .....	20
10.	Class diagram and activity diagram .....	21
10.1.	Class diagram .....	21
10.2.	Activity diagrams.....	22
III.	Design.....	24
1.	Pseudocode.....	24
2.	Actual code .....	26
3.	Wireframes .....	28
3.1.	Log in.....	28
3.2.	Register .....	29
3.3.	Browsing song .....	30

3.4. Cart.....	30
3.5. Checkout/invoice .....	31
4. Test case.....	31
5. Architecture used.....	32
References .....	33

## I. Introduction

The SDLC model utilized for Tune Source has been illustrated in earlier sections, and some of the hazards that the Tune Source project may encounter have been studied, handled, and explained. And the feasibility analysis demonstrated that this idea is possible. The stakeholders and their interests in the case study are then identified in this part. The requirements description is evaluated, and it is demonstrated how stakeholders deliver the needs for this project. Finally, the connection between FRs and NFRs is examined. And in this section, the TS project's Requirement Definition (with explanations of the need/purpose) is stated, as well as what requirements the relevant party(s) supply.

## II. Analysis

### 1. Requirement definition

A requirement is a physical or functional necessity that is defined in product development and process optimization that a certain design, product, or process tries to meet. It is often used in formal engineering design, such as systems engineering, software engineering, or enterprise engineering. It is a wide notion that can refer to any required (or sometimes desirable) function, attribute, capacity, trait, or quality of a system in order for it to be valuable and useful to a customer, company, internal user, or other stakeholders.

A Functional Requirement (FR) describes the service that the program must provide. It refers to a software system or a component of one. A function is nothing more than the software system's inputs, behavior, and outputs. It might be a computation, data manipulation, business process, user interaction, or any other specialized activity that defines the function of a system. In software engineering, functional requirements are also known as functional specifications. (Martin, 2022)

Non-Functional Requirement (NFR) describes a software system's quality attribute. They evaluate the software system based on its responsiveness, usability, security, portability, and other non-functional characteristics that are crucial to its success. "How quickly does the website load?" is an example of a non-functional requirement. Non-functional requirements that are not met can lead to systems that do not meet the demands of the users. (Martin, 2022)

## 2. Stakeholders who can make functional and non-functional requirements and their interests

Stakeholder	Position	Interests
Carly Edwards	Assistant Vice president of the Marketing department, Sponsor of the project	When this project is finished, they will own stock in the firm and earn a part of the money generated by the newly established system.
John Margolis	Founder of Tune Source	When the project is finished, their income will grow dramatically due to additional clients and market share, and they will earn more when the project is put on the market.
Trinh Duc Anh	Business Analyst	A Business Analyst's goal is to develop a complete business analysis that defines a company's issues, prospects, and solutions. Forecasting and budgeting. From there, planning and monitoring will assist the firm to always rise and be rewarded by the corporation.
Customers	Customers, the source of revenue for the company	The customers want a quick, cheap, and convenient way to listen to music. Implementing an online music subscription would satisfy that want fully.
Development team	Implement an online store for Tune Source	Should the project be successful, they would earn tangible benefits in the form of bonuses along with intangible ones which are increased reputation and referral from Tune Source.

Table 1 Stakeholders who can create requirements and their interests

## 3. Functional and non-functional requirements of the project

ID	User stories	F/NF	Explanation
1	As a customer, I want to be able to browse Tune Source on my phone and make purchases on the device	F	This is a requirement that involves user interaction; thus, it is a functional requirement. As smartphones are the dominant internet device, a mobile-friendly online store will be able to bring and keep more customers, making the company prosper further. It is an essential aspect for an online store to have
2	As the chief accountant, I want Tune Source to be able to connect and work with payment services like PayPal, Momo, bank apps	F	The implementation of this feature would allow the user to have more choices in how they pay for the items from the store, positively affecting the user interaction, and making this a functional process. Furthermore, shaking hands with third-party payment systems would eliminate the need for an in-house solution, keeping the development cost and time lower while giving the customer peace of

			mind and ease of use due to how simple it is to use these services and how famously secure they are.
3	As the warehouse manager, I want Tune Source to integrate third-party shipping services, allowing the customers to choose what shipping service provider they want when checking out	F	This is a requirement for the user's experience along with the business process, making this a functional one. Traditionally, the company has already worked with shipment services to deliver the products to the customers' hands so it is natural that the online store would also shake hands with them. Furthermore, integrating these services into the store would automate the information handling between the store, the shipping service, and the customers, reducing time and cost along with mitigating any error in the process.
4	As the operation director, I want the Tune Source website to be aesthetically pleasing	F	This is a functional requirement due to the fact that it revolves around the experience of the users. Aesthetics is important as the website needs to show the store's uniqueness in design, attracting more customers.
5	As a customer, I want to be able to create a Tune Source account	F	This is a functional requirement because it is about user interaction. Allowing the customer to create an account would let them more easily track the orders that they've made, manage their cart should they switch devices, set up default information so they do not have to manually fill their billing and delivery information every time they want to make a purchase.
6	As a user, I want to be taken to the login page when I try to play a song before being logged in	F	This is a requirement that is about user interaction, making this a functional requirement. Redirecting the user to the login page when they try to play a song without logging in would save them the time of finding a log-in button, thus improving the user experience and satisfaction
7	As a customer, I saw this cool song and think that my friends would enjoy it, I want a short link to share it and the preview thumbnail should show concise information about the product	F	This is a user interaction requirement; therefore, it is functional. The product's URL should be short so that social media platforms do not automatically flag it as spam. When a URL is shared on social media, a thumbnail of the page would also be provided. This thumbnail should be designed so that a high definition of the product should be displayed when the URL is shared so that the

			visitor will have a more positive impression of the website before visiting it.
8	As the operation director, I want the website to load as quickly as possible	NF	This is the requirement regarding the performance of the website. A fast-loading website improves the customer's user experience. Long loading time frustrates the customers and makes them leave the store. To make sure that the customers stay in the store as long as possible, a fast-loading time is essential to the system's performance
9	As the Managing Director, I need the system to protect the customers' information along with all the data to be protected as tightly as possible	NF	This requirement regards the security of the system, making it non-functional. Security is an essential feature of every online store; Lizzie's is not an exception. When a customer creates an account, they have entrusted us with their personal information, we need to do our best to protect it. Other data such as employee accounts should also be secured to protect our hard workers.
10	As the operating director, I want the website to be as accessible as possible	NF	This is a requirement for the usability of the website, making it non-functional. Accessibility is important as the customers should be able to perform actions with ease on the website, this factor would keep the customer on the website for longer and increase sales for the company.
11	As the Managing Director, I need the system's maintenance to be as easy as possible	NF	This is a requirement for the effectiveness of the website, making it non-functional. Every time the system needs to be maintained; the website would be down. Downtime is bad for every online business, and developing good maintainability minimizes the downtime of the system, making the practice an important aspect of the project.
12	As the Managing Director, I need the system's expansion to be as easy as possible	NF	This is a requirement for the effectiveness of the system, making it non-functional. The website will grow over time, and when it is time to scale up, a scalable system will allow the development time and labor to be shorter, thus saving cost while giving the website more features in a shorter time.

Table 2 Functional and non-functional requirement



## 4. Techniques used to gather requirements

Requirements elicitation, also known as requirements collecting, is one of the most complex, error-prone, and communication-intensive phases of software development. It can only be successful with strong customer-developer cooperation. It is necessary to understand what the users truly require. (Lunalovegood, 2021)

### 4.1. Interviews

Interviews with stakeholders and users are essential for developing effective software. We are unlikely to satisfy users and stakeholders unless we grasp their aims and expectations. We must also identify each interviewee's point of view in order to appropriately assess and handle their contributions. Listening is the talent that allows a great analyst to extract more value from an interview than an average analyst. (tutorialspoint, 2022)

There are five basic steps to doing an interview:

1. Selecting interviewees
2. Designing interview questions
3. Preparing for the interview
4. Conducting the interview
5. Post-interview follow-up

### 4.2. Joint Application Development

The process of starting a new project is known as joint application development. The JAD approach is utilized for technology projects with Application Management that may be implemented for internal organizational operations or for external commercialization. (Avendano, 2021 )

This necessitates workshops or meetings with stakeholders, both IT professionals and business users, in order for them to participate and improve the system in the best way possible. Having both views that might generate inventive ideation for technology is a critical success component. (Avendano, 2021 )

- IBM created the JAD methodology in the late 1970s, and it is frequently the most effective method for gathering information from consumers.
- JAD is an organized procedure in which 10 to 20 users gather under the supervision of a JAD facilitator.
- A facilitator is someone who creates the meeting agenda and conducts the discussion but does not participate in it.
- The facilitator should be knowledgeable about both group process approaches and system analysis and design methodologies.
- A scribe or two assists the facilitator by taking notes, producing copies, and so forth.
- The JAD group meets for numerous hours, days, or weeks until all topics have been explored and all necessary information has been gathered.
- The majority of JAD meetings are held in a specially equipped conference room.

- The conference space is frequently set up in a U form so that all attendees can see each other easily.

Steps to conduct a JAD are:

1. Select participants
2. Design the JAD session
3. Prepare for the JAD session
4. Conduct the JAD session
5. Post-JAD follow-up

#### 4.3. Questionnaires

- A questionnaire is a collection of written questions designed to elicit information from persons.
- Questionnaires are frequently employed when a significant number of people's information and views are required.
- When compared to paper questionnaire distribution, electronic dissemination can save a large amount of money.

#### 4.4. Document Analysis

- Document analysis is frequently used by project teams to understand the current system.
- Unfortunately, most systems are not thoroughly documented because project teams fail to record their projects as they progress, and once the projects are completed, there is no time to go back and document them.
- There are several useful papers in the organization:
- Reports, memorandums, policy guides, user training manuals, organizational charts, and forms on paper

#### 4.5. Observation

- Observation, or the act of observing processes take place, is a strong technique for gaining insight into the current system.
- Observation is an excellent technique to validate information obtained from other sources such as interviews and surveys.
- The idea is to keep a low profile, not disrupt those who are working, and not affect those who are being monitored.

#### 4.6. Technique chosen

For Tune Source, the technique chosen to gather requirements is Interview. With this method of requirements gathering, the member of the development team is able to work with each stakeholder individually, helping them focus more well on the process of communicating the requirements they need from the system.

## 5. Interview questions

No	Stakeholder	Question
1	Customer	What do you want to happen when you want to share a song to your friends through social media?
2	Customer	What do you want to happen when you try to listen to a song but you are not logged into the website yet?
3	Operation director	How do you want the website to be like appearance-wise?
4	Warehouse manager	What feature do you want to system to have relating to your job at the company?
5	Customer	On what device do you plan on using Tune Source's services?
6	Operation director	What is your expectation of the performance of the to-be-implemented system?
7	Managing Director	What do you expect of the system regarding maintainability?
8	Managing Director	What do you expect of the system regarding expandability?

*Table 3 Interview questions*

## 6. Use-case

### 6.1. Signs up for a subscription

Signs up for subscription	ID: 01	Actor: Customer
<b>Description:</b> A user with/without an account sign up for Tune Source music subscription. The customer provides their billing information, and their account is granted the subscription status allowing them to limitlessly download and listen to songs on the system.		
<b>Trigger:</b> A customer decided to subscribe to the Tune Source music service.		
<b>Type:</b> <input checked="" type="checkbox"/> External <input type="checkbox"/> Temporal		
<b>Precondition:</b> The website, account database, user database is online, payment system is online.		
<b>Normal Course</b> 1) A customer visits the website to sign up for Tune Source music subscription. 2) Log-in/Sign-up step a) The customer clicks on the "Sign up" button. b) The customer clicks on the "Sign in" button 3) A sign-up/log-in page welcomes the customer, prompting them to enter their account information. 4) The customer fills in their information for account verification. a) For new customer without account, they would have to input their email address, phone number, password and its retype to create a new account b) For existing customer with an account, they only have to input their email address and password to log into the website. 5) After the user have successfully created or log into their account. a) The system displays a success message telling the customer that their account has been created, then automatically log the user into the website. b) The system displays a success message telling the customer that they have logged into their Tune Source account, then redirects them to the home page 6) The customer wants to subscribe to Tune Source after their account is logged in. a) After the customer created a new account, the website then displays a message asking if the customer wants to sign up for Tune Source's monthly subscription, the customer clicks "Yes." b) After the customer logged into an existing account, they navigate to account settings, subscription plan, and clicks "Add subscription". 7) The customer is taken to a site containing information of each subscription plan. 8) The customer chooses a plan that they want. 9) The website displays a form for the customer to fill in their billing information including credit card information, legal name, and billing address. 10) The sign-up is successful, and the subscription status of the customer's account is turned on.		<b>Information for step</b>  ← Phone/Password/Email ← Email/Password → New Account  ← Plans information → Modified Account
<b>Exception:</b> 1) The customer's credit card is declined (Occurs at step 9) a) System displays a message saying that the customer's credit card has been declined when the system is trying to make a payment with it. b) System asks the user to re-input a valid credit card number. 2) Account is not valid (Occurs at step 4b) a) System displays a message saying that the email/password combination is invalid. b) System asks the user to retype their account credential or to reset password via the "forgot password" or customer service.		
<b>Postconditions:</b> 1) After a customer has signed up for a subscription a) A new account is created with subscription status assigned to it along with credit card information for recurring payments. b) An existing account is assigned subscription status along with credit card information for recurring payments. 2) The account owner can enjoy the perks of the subscription plan just by logging into their account. 3) A receipt of the transaction must be made, stored on the system, and sent to the customer.		

Figure 1 Use case: Signs up for subscription

## 6.2. Buys a gift card

Buys a gift card		ID: 02	Actor: Customer	
<b>Description:</b> A customer comes to the Tune Source website to by a gift card. After the payment is complete, the customer is provided with a code that can be redeemed for the credit that they have paid for.				
<b>Trigger:</b> A customer decided to buy a gift card from Tune Source website.				
<b>Type:</b> <input checked="" type="checkbox"/> External <input type="checkbox"/> Temporal				
<b>Precondition:</b> The website, account database, user database is online, payment system is online.				
<b>Normal Course</b> 1) A customer visits the Tune Source website. 2) The customer clicks on the Gift Card section. 3) The customer chooses the card(s) that they want to purchase. 4) The customer clicks "Add to cart". 5) The customer clicks on the cart symbol, the website takes them to their cart. 6) The customer clicks "Checkout". 7) After clicking "Checkout", the user is prompted with a log in form. a) The customer fills in their account information to log into their account b) The customer clicks on the "Sign up" button and signs up for a new account before making a purchase. c) The customer clicks on the "Check out as guest" button and makes a purchase without a Tune Source account. 8) The website displays a payment information form. a) For customer that logged into an account with saved information, the payment information is automatically filled, the customer needs to double-check to ensure that the information is correct. b) For customer that checks out anonymously or has an account without saved payment information, they need to fill out each of the information field to make a payment. 9) The customer clicks on the submit buttons and the payment is successful. 10) The website displays a success message saying that the payment has been made for their order. 11) After the payment is made a) For logged-in customer, the gift card codes can be accessed through their account information section. b) For all customers, the gift card code is emailed via the email address and phone number that they provided in step 8.			<b>Information for step</b>  ← Password/Email ← Phone/Email/Password → New Account ← Payment information → New Order → Modified account	
<b>Exception:</b> 1) The customer's credit card is declined (Occurs at step 9) a) System displays a message saying that the customer's credit card has been declined when the system is trying to make a payment with it. b) System asks the user to re-input a valid credit card number. 2) Account is not valid (Occurs at step 7a) a) System displays a message saying that the email/password combination is invalid. b) System asks the user to retype their account credential or to reset password via the "forgot password" or customer service.				
<b>Postconditions:</b> 1) The account that makes the purchase will have the gift card stored in the account. 2) Gift card code must always be sent via email and text message via the address and number that the customer provides during the checkout stage. 3) A receipt of the transaction must be made, stored on the system, and sent to the customer.				

Figure 2 User case: Buys a gift card

## 7. Data flow diagram

### 7.1. DFD 0

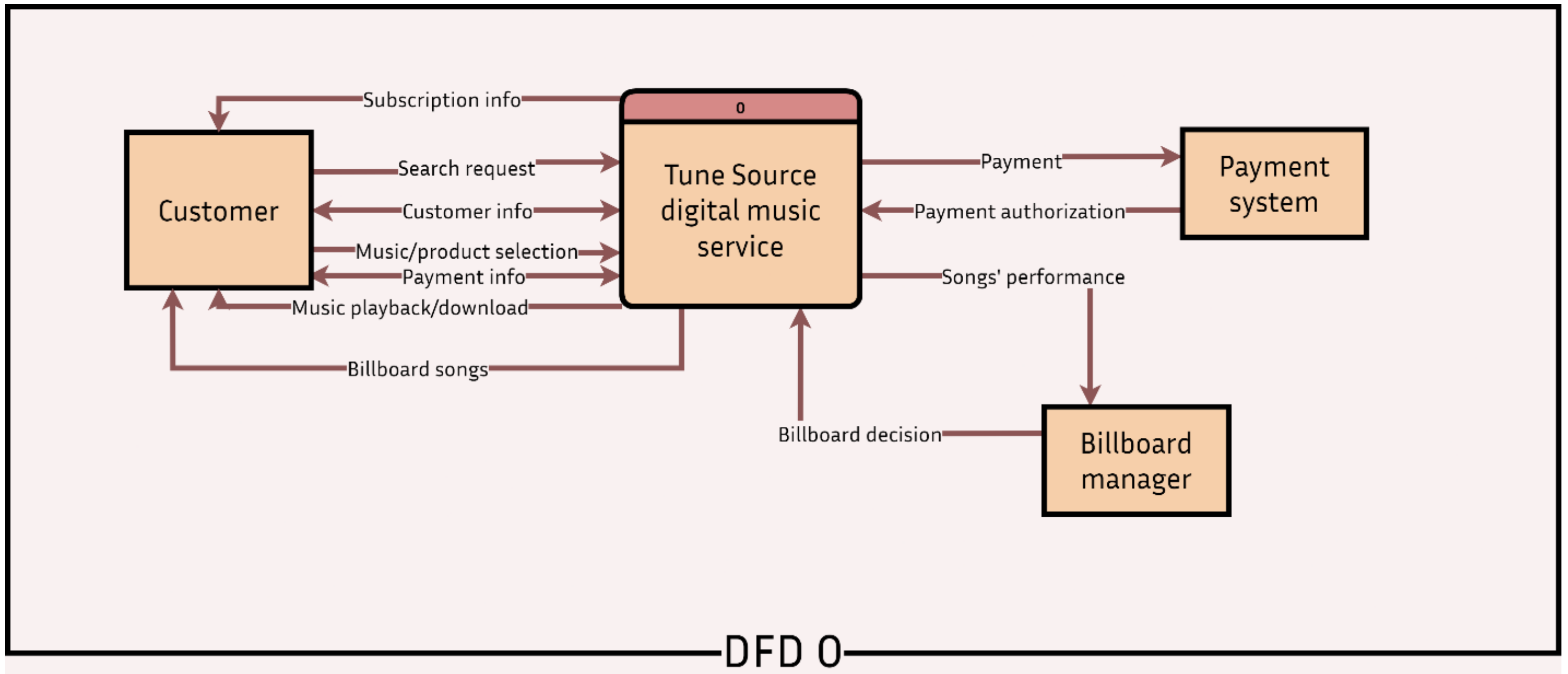
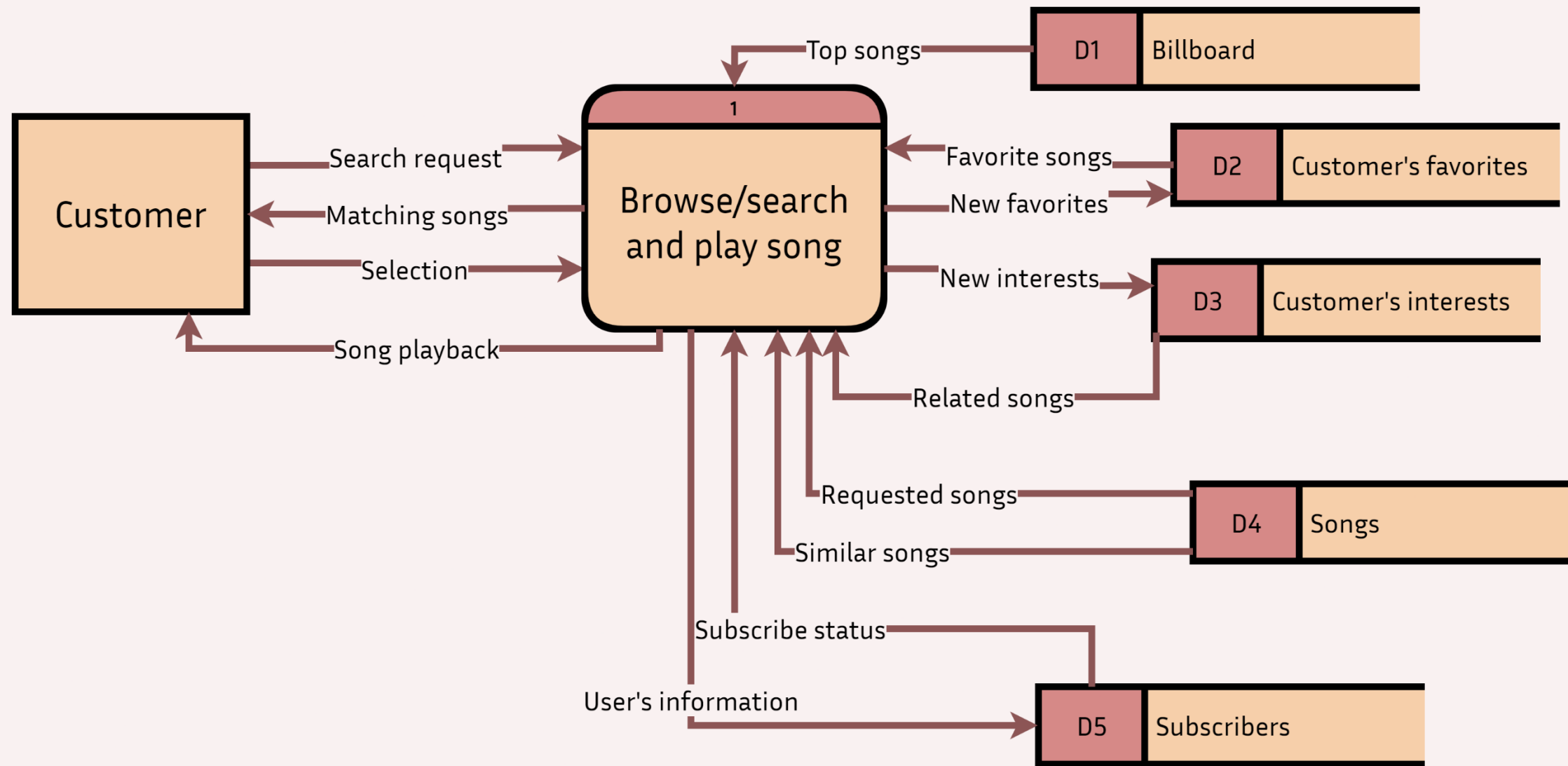


Figure 3 Context diagram (DFD 0)



DFD 1 Fragment

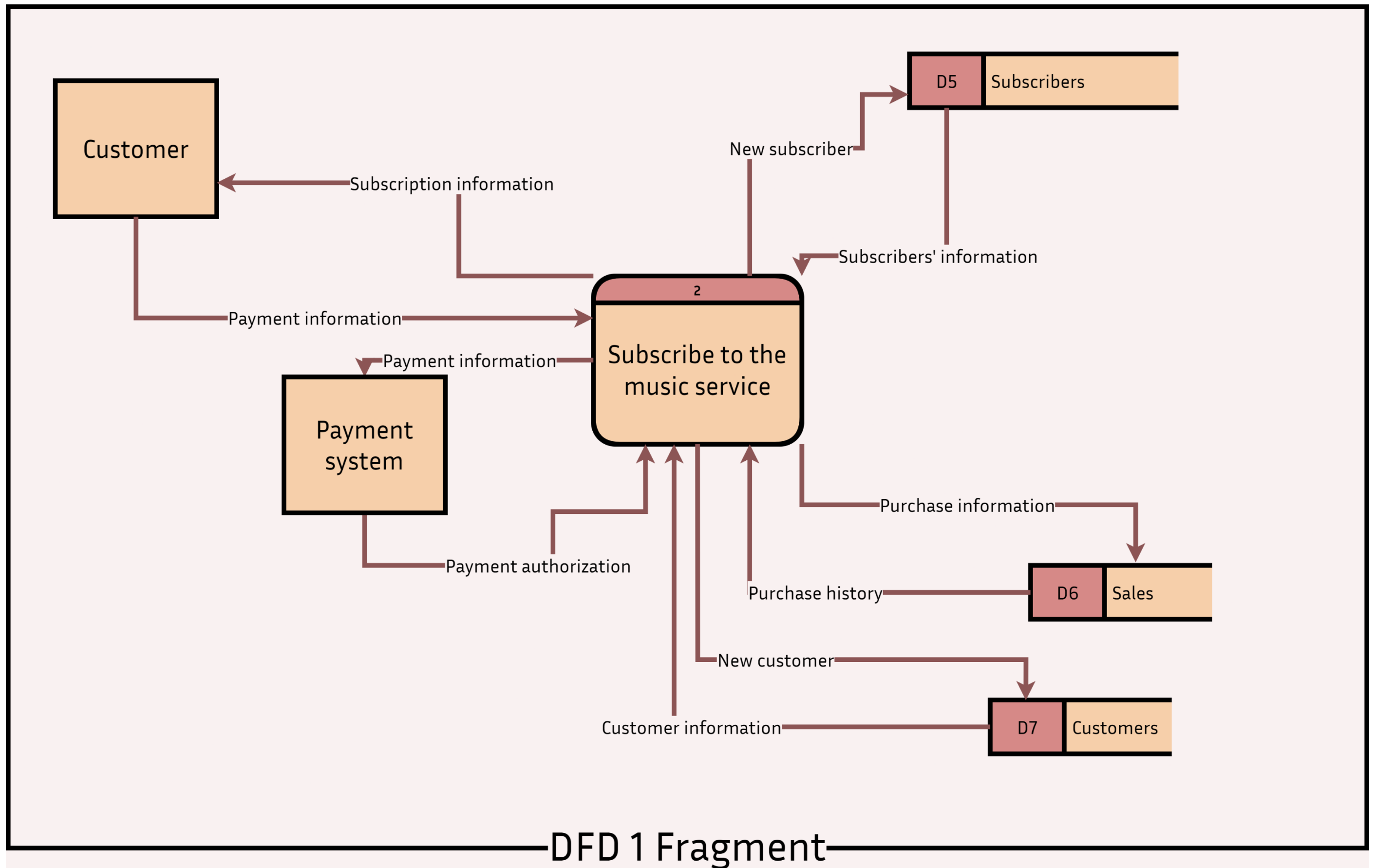


Figure 5 DFD Fragment 1 - Subscribe to the music service



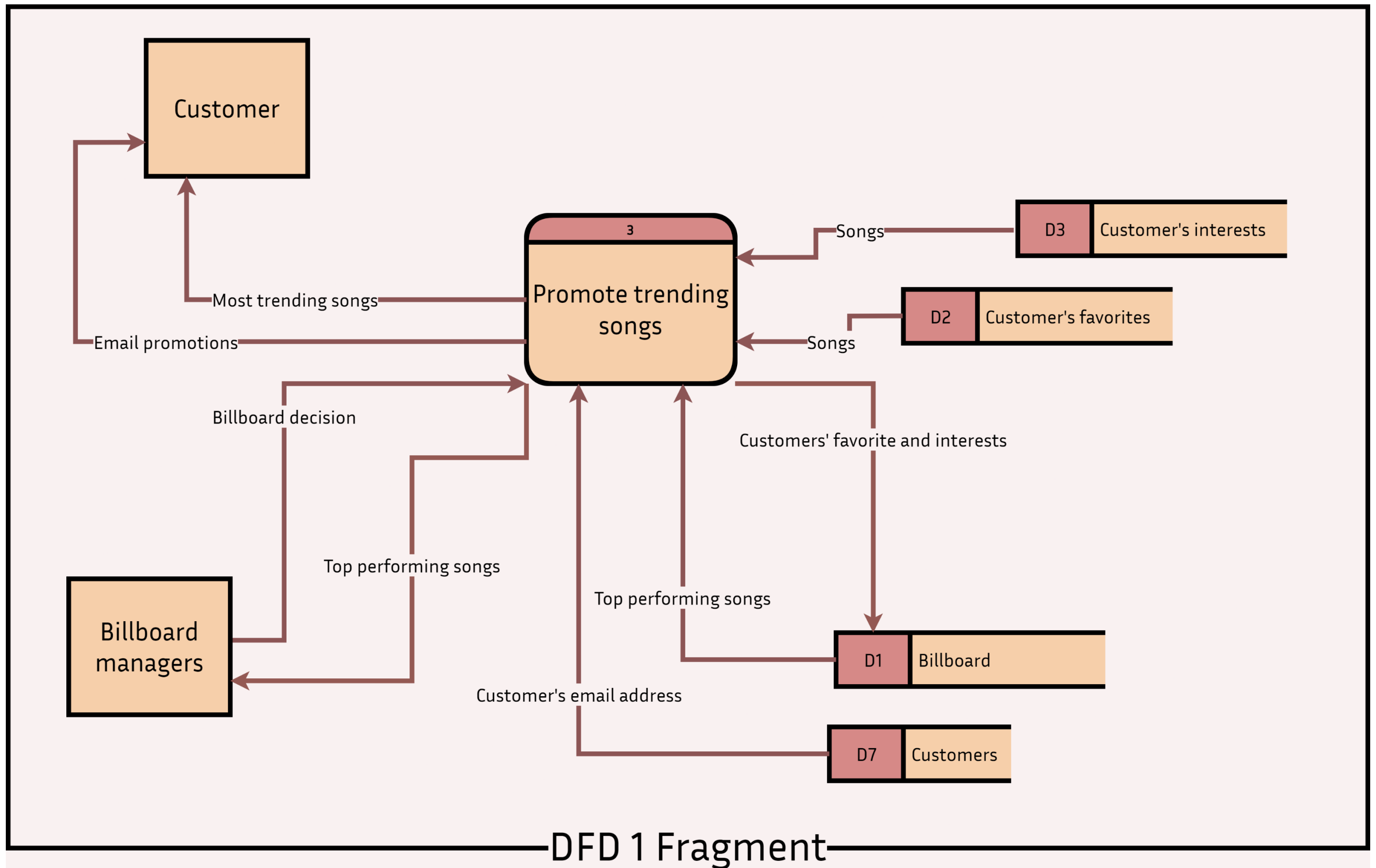


Figure 6 DFD 1 Fragment - Promote trending songs

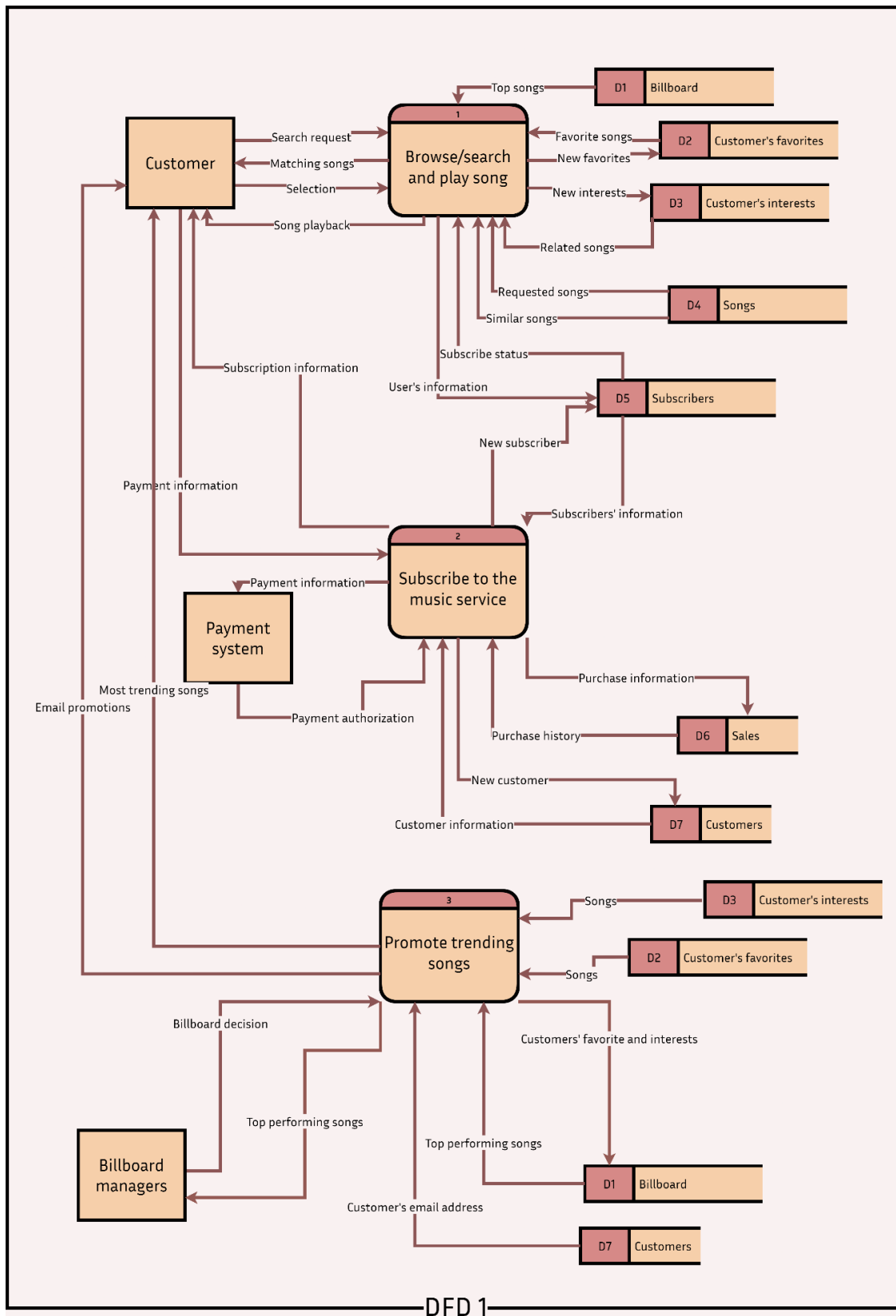


Figure 7 DFD 1

## 8. Entity relationship diagram

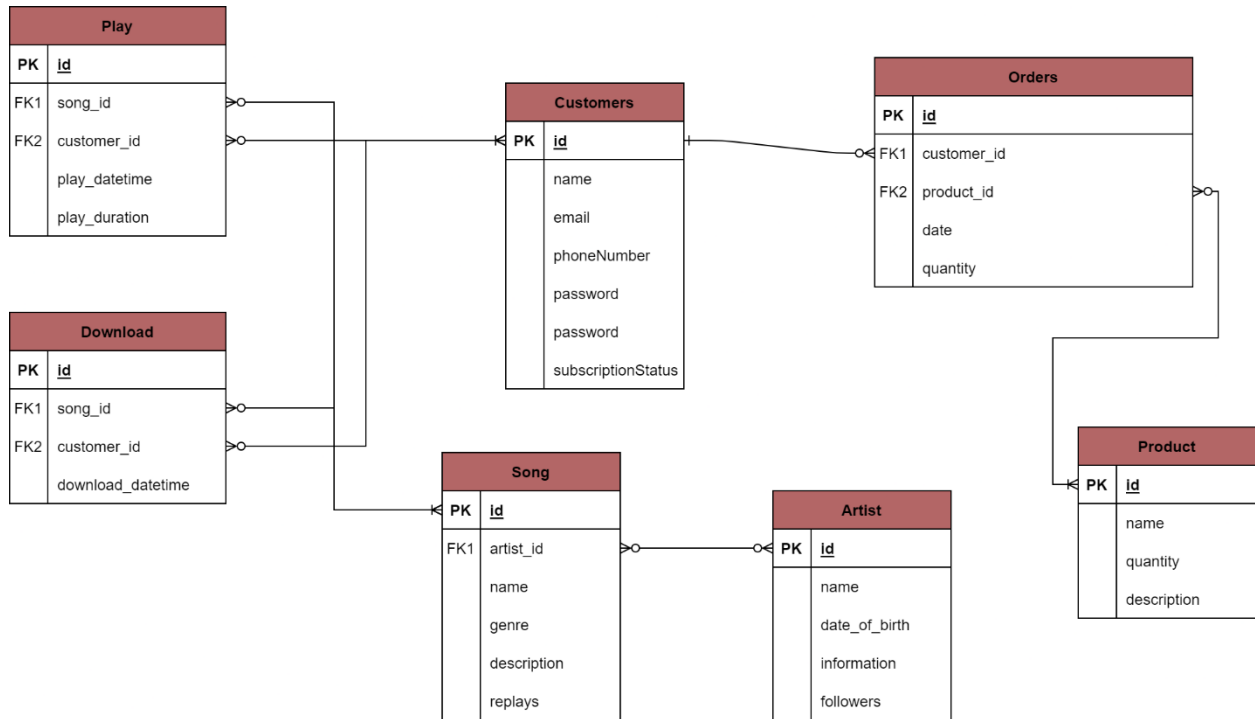


Figure 8 Entity-relationship diagram

## 9. Flowchart

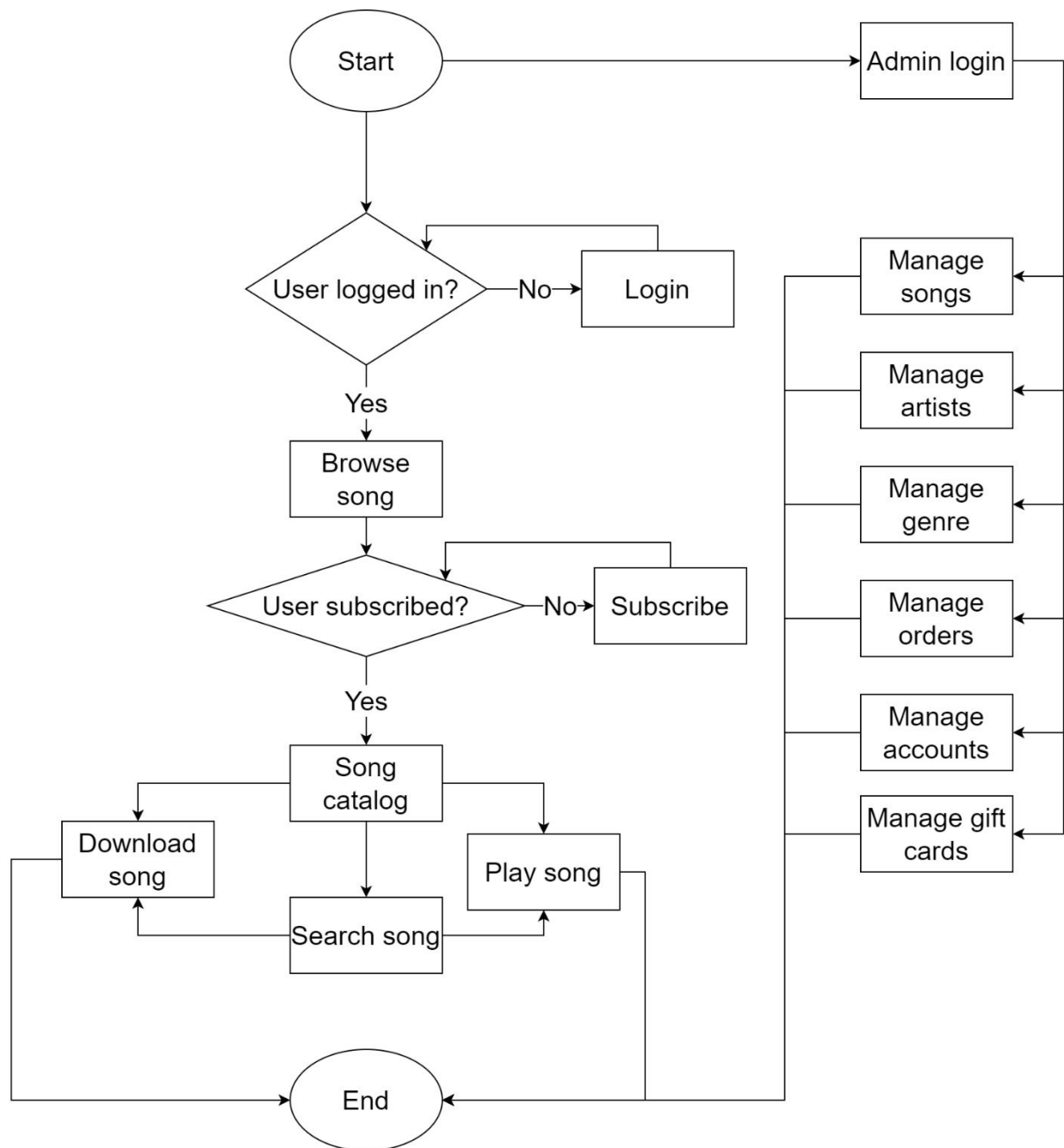


Figure 9 Flowchart

10. Class diagram and activity diagram

10.1. Class diagram

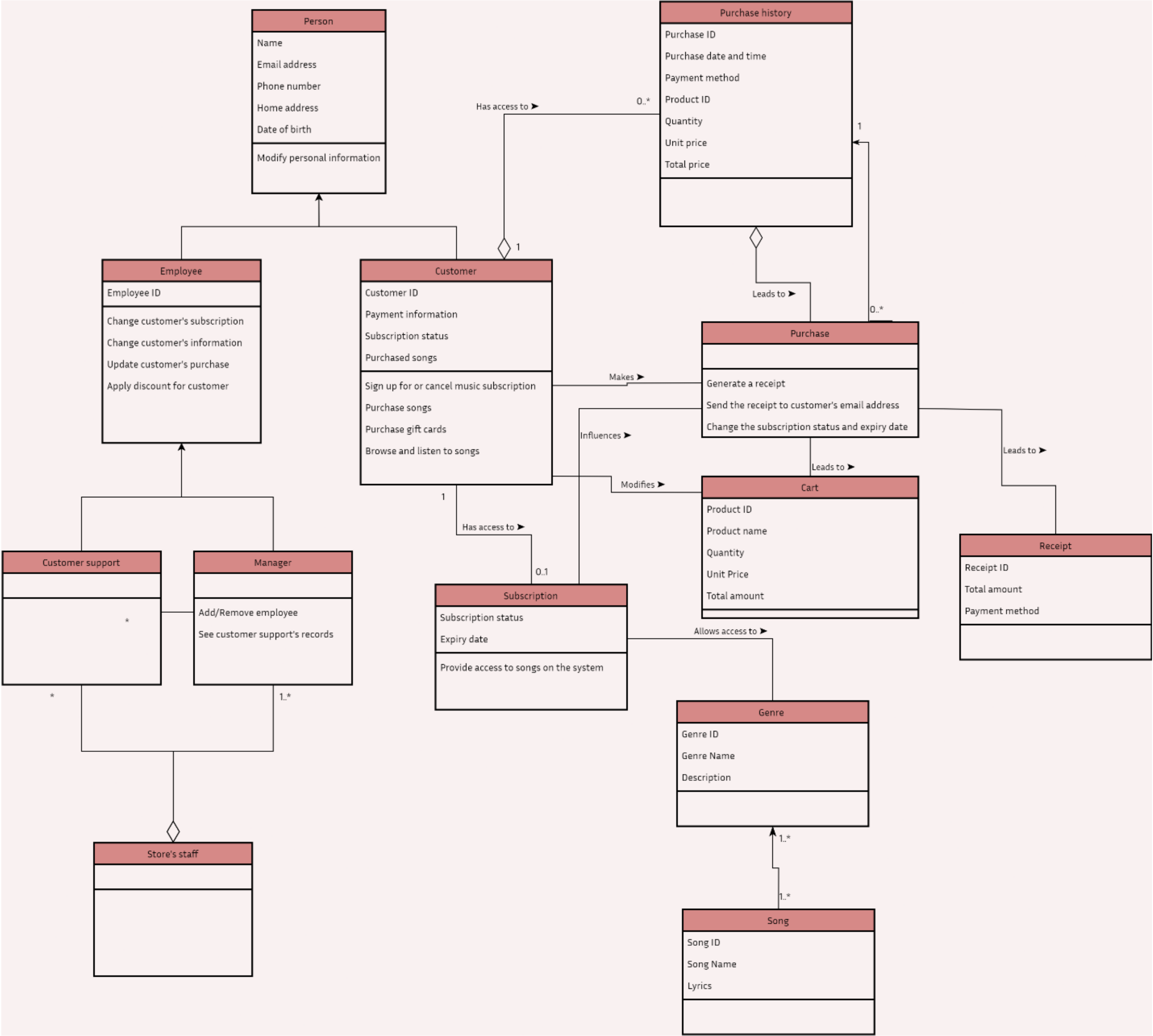


Figure 10 Class diagram

## 10.2. Activity diagrams

### *Subscribing to the music streaming service*

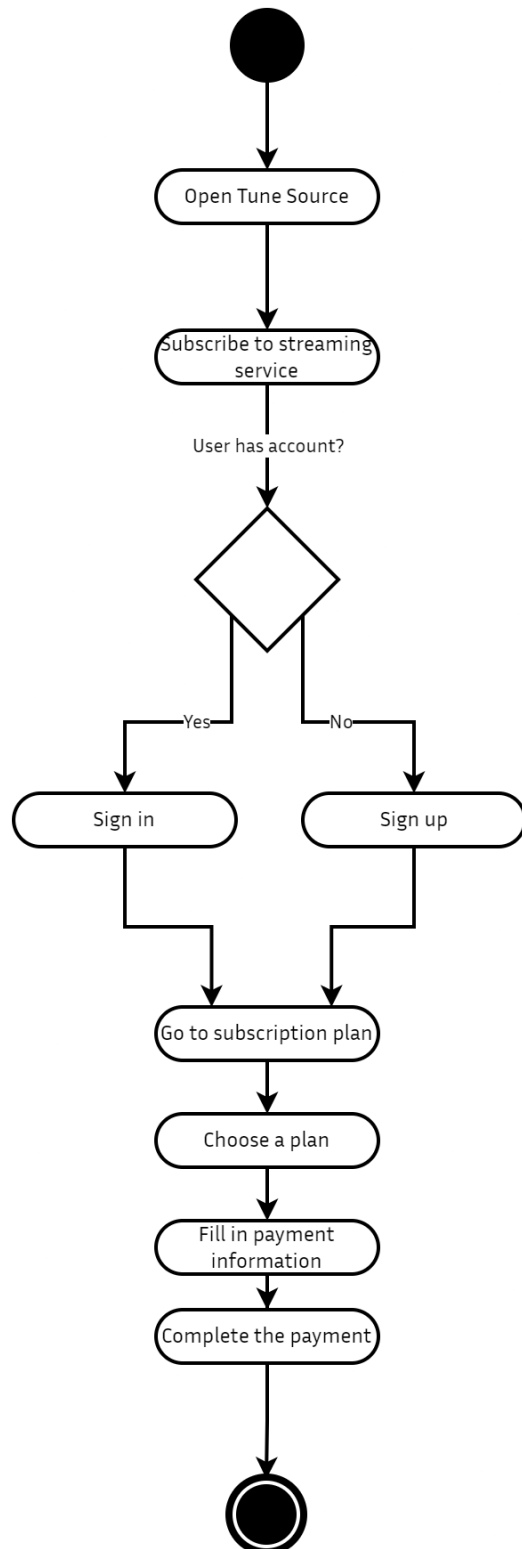


Figure 11 Activity diagram of signing up for subscription

### Buying gift card

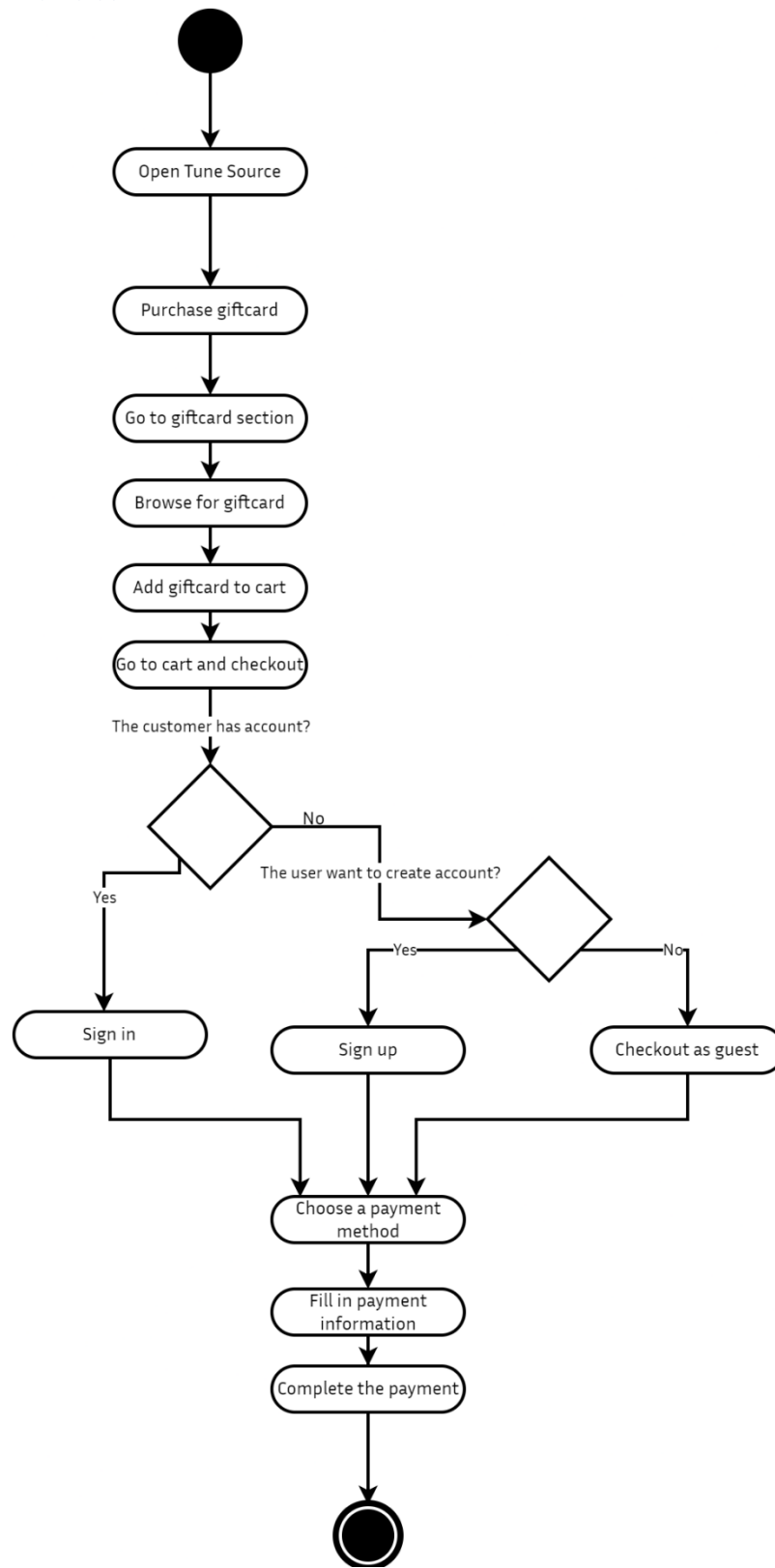


Figure 12 Activity diagram for buying a gift card

### III. Design

#### 1. Pseudocode

```
// Login Pseudocode
begin
valid input_password:=false
valid input_username:=false
logged_in:false
while(logged_in=false)do
begin
print("Please enter Username:")// input Username
readln(inputted_username)
writeln("Please enter Password:")// input Password
readln(inputted_password)
// begin matching process if inputted_username is in
Login_Dictionary then
// match inputted login details with each existing valid
set of Login details from the Login Dictionary
valid input_username:=true
if
else
end if else
then
    inputted_password is in Login_Dictionary then
    valid input_password:true // Username and
Password are both matched
    logged_in:=true
    writeln("Logins details have been validated")
else
    valid input_password:=false // If username was
matched but password wasn't
    logged_in:false
```



```
        writeln("Error your username or password was
invalid")
    end if
end while
if
end
    valid input_username=false // If username wasn't
matched.
    logged_in:=falsewriteln("Error your username or
password was invalid")
    logged_in=true
    writeln("You have been logged in")
    writeln("You could not be logged on due to false
Login details")
    end if
end
```

## 2. Actual code

```
public static void loginUser(String userName, String
userPassword) {

    String fileName = "user.csv";
    readUserFromFile(fileName, userList);
    isLogin = false;
    for (UserModel user : userList) {
        if (user.getUserName().equals(userName) &&
user.getUserPassword().equals(userPassword)) {
            isLogin = true;
            break;
        }
    }
    if (isLogin) {
        JOptionPane.showMessageDialog(null, "Login
successfully");
        // go to StudenListGUI
        StudentListGUI studentListGUI = new
StudentListGUI();
        studentListGUI.setVisible(true);

    } else {
        JOptionPane.showMessageDialog(null, "Login
failed");
        // reopen LoginGUI
        LoginGUI loginGUI = new LoginGUI();
        loginGUI.setVisible(true);
    }
}
```

```

        // Check if the input userName and userPassword is
correct
        public void checkLogin(String userName, String
userPassword) {

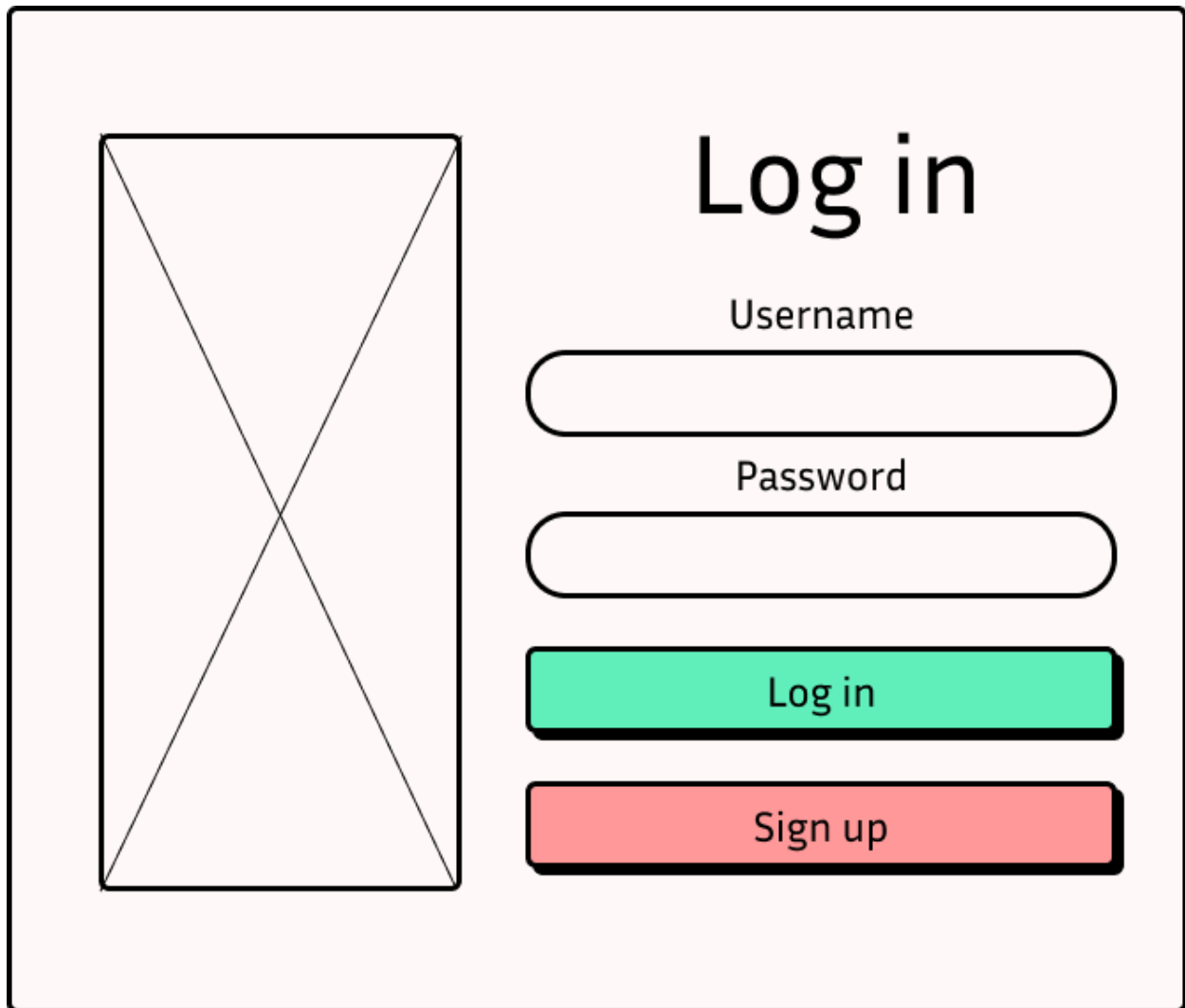
            String fileName = "user.csv";
            readUserFromFile(fileName, userList);
            isLogin = false;
            for (UserModel user : userList) {
                if (user.getUserName().equals(userName) &&
user.getUserPassword().equals(userPassword)) {
                    isLogin = true;
                    break;
                }
            }
            if (isLogin) {
                JOptionPane.showMessageDialog(null, "Login
successfully");
            } else {
                JOptionPane.showMessageDialog(null, "Login
failed");
            }
        }
    }
}

```

This is a piece of code for the login function. The system first would load the file that contains all the Tune Source's accounts. Then it takes the login credentials that the user has input, matches it with the data in the file. If the system is able to find a match, the user would be logged into Tune Source. Otherwise, the user is stuck on the login page with an error message asking the user to re-enter valid account credentials.

### 3. Wireframes

#### 3.1. Log in



A wireframe for a login page. On the left is a large rectangle with a diagonal 'X' inside, representing a placeholder for a logo or image. On the right, the text 'Log in' is displayed in a large font. Below it are two input fields: the first is labeled 'Username' and the second is labeled 'Password'. At the bottom of the form are two buttons: a green 'Log in' button and a red 'Sign up' button.

Log in

Username

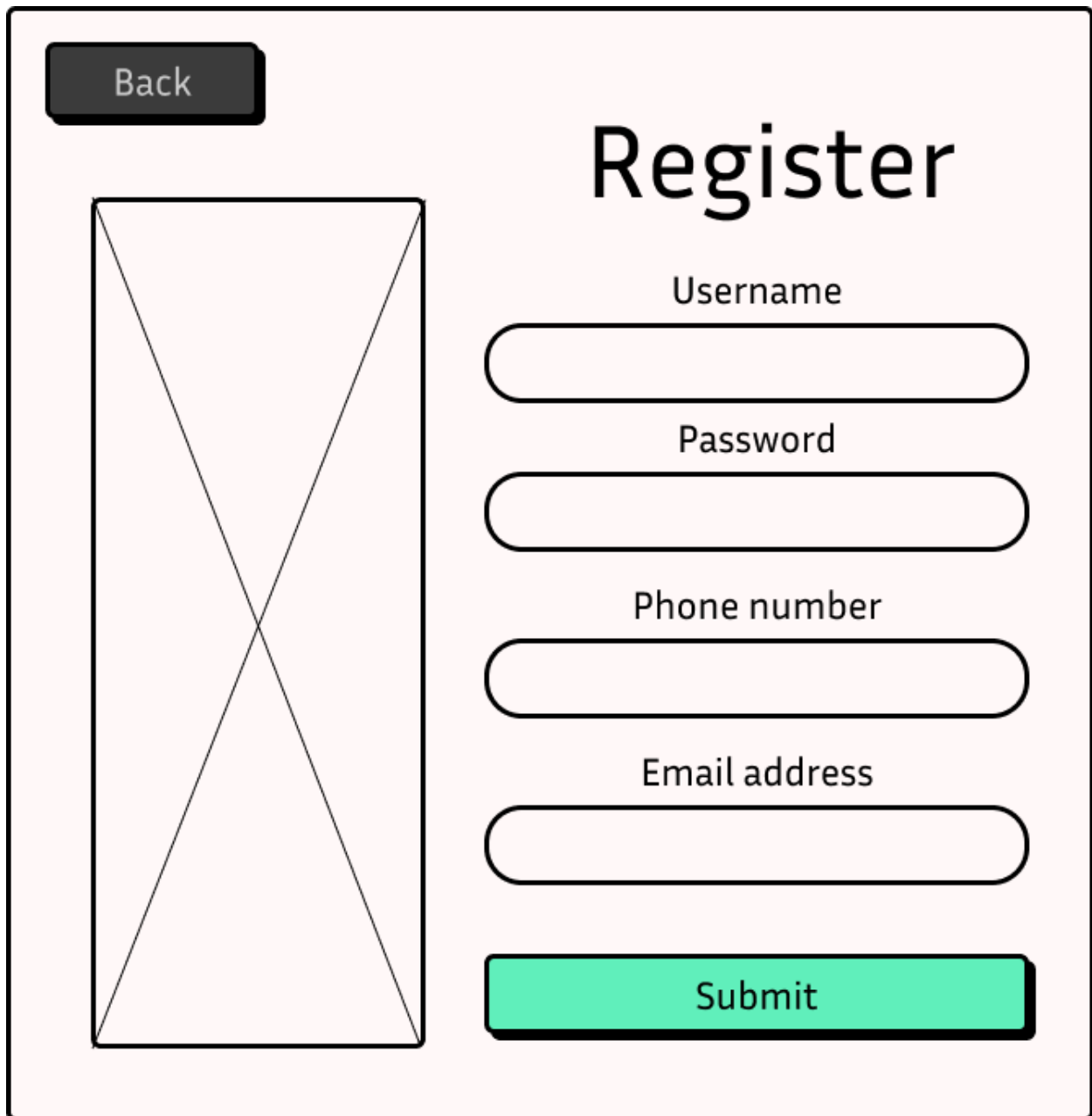
Password

Log in

Sign up

Figure 13 Wireframe for log in

### 3.2. Register



A wireframe for a registration form. The form is contained within a light pink rectangular box with a black border. In the top-left corner of the box is a dark gray button with the text "Back". To the right of this button, the word "Register" is displayed in a large, bold, black font. Below the title, there are four input fields, each consisting of a label and a rounded rectangular box. The labels are "Username", "Password", "Phone number", and "Email address", arranged vertically. At the bottom of the form is a green button with the text "Submit". On the left side of the form, there is a large, empty rectangular box with a black border and a large 'X' drawn across it from corner to corner.

Back

# Register

Username

Password

Phone number

Email address

Submit

Figure 14 Wireframe for registration

### 3.3. Browsing song

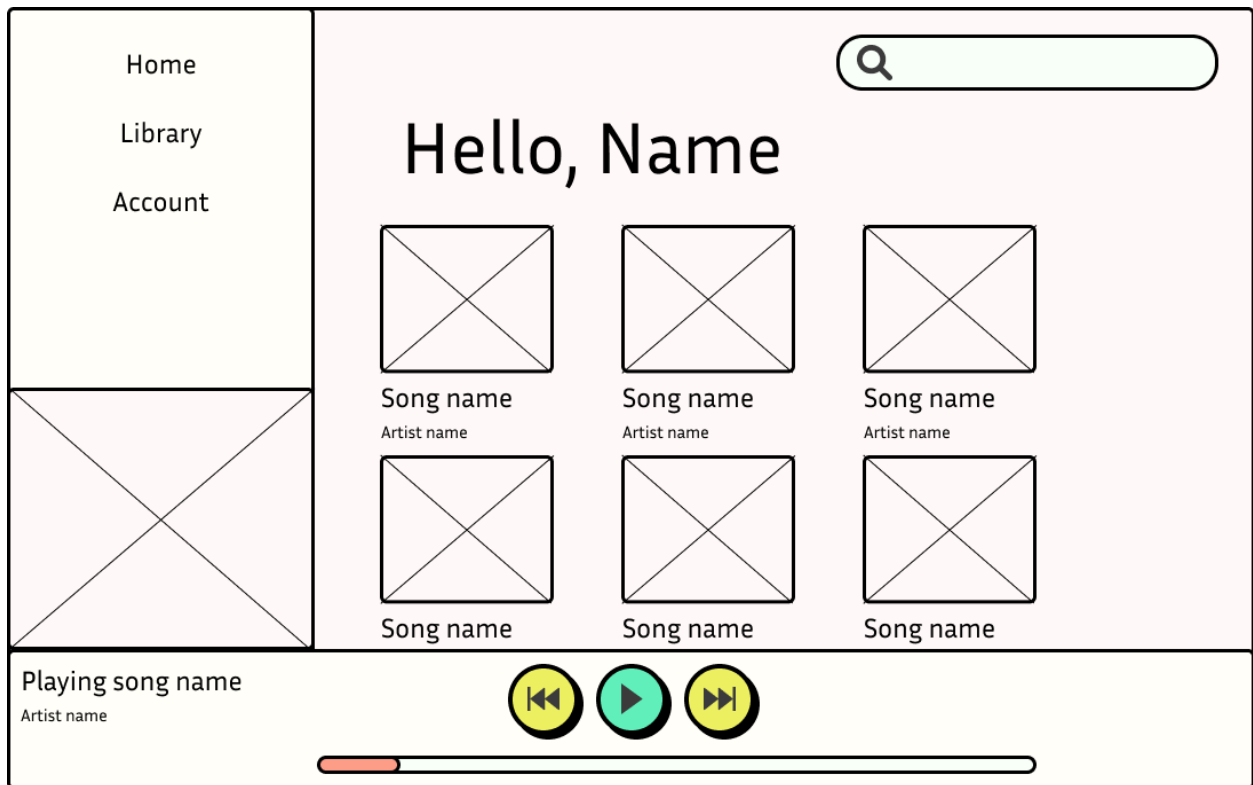


Figure 15 Wireframe for browsing songs

### 3.4. Cart

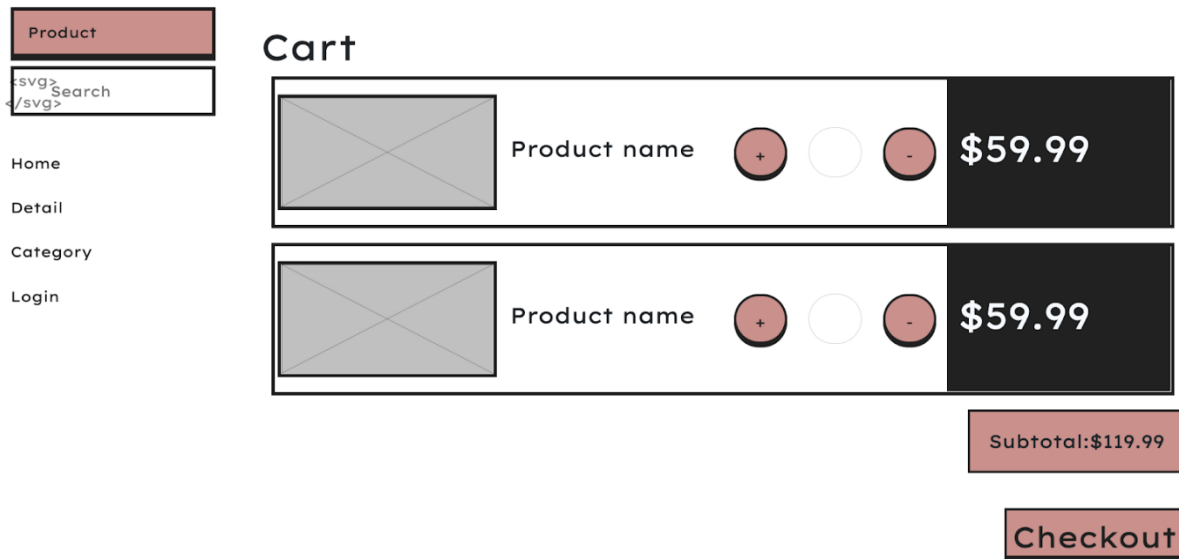


Figure 16 Wireframe for cart

### 3.5. Checkout/invoice



Figure 17 Wireframe for checkout/invoice

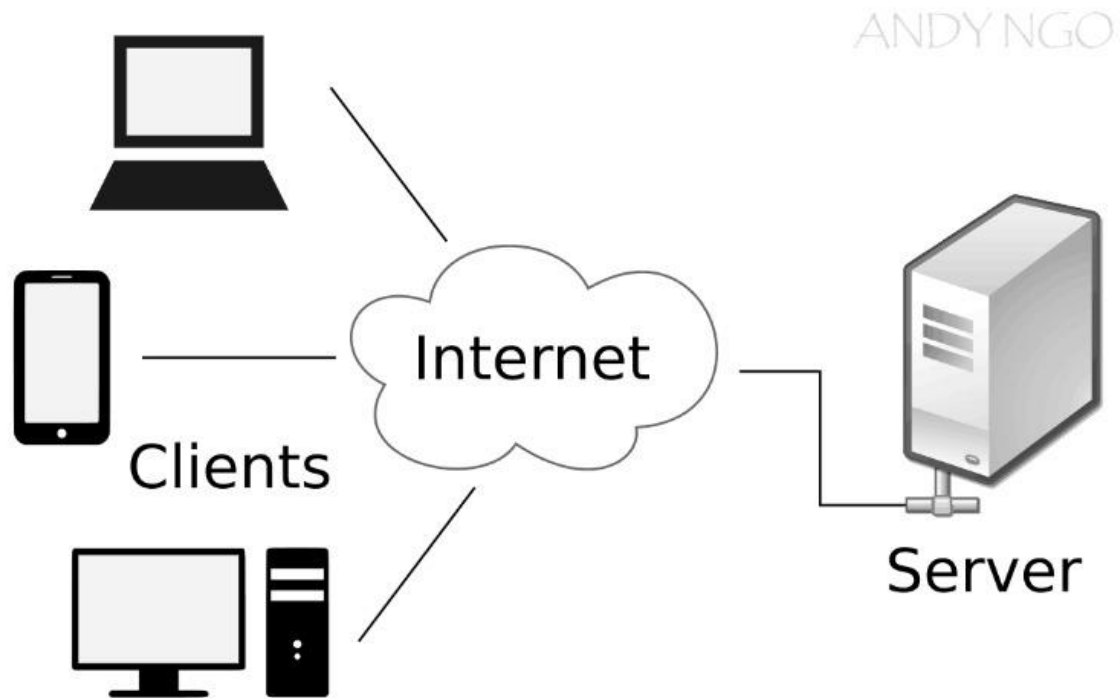
### 4. Test case

No	Description	Expected result
1	Test login function with valid data	The user is logged in and able to access the account's information
	Test login function with invalid data	The user is not logged into any account
2	Test playing a song with a subscribed account	The song plays
	Test playing a song with an account that has not subscribed	The song does not play, and the user is redirected to a page where they can sign up for the subscription
3	Test buying a gift card with valid credit card information	The transaction goes through, a code for the gift card is sent to the customer's email address and is accessible through their account
	Test buying a gift card with invalid credit card information	The transaction does not go through, the customer is prompted to re-enter a valid credit card information

Table 4 Test case table

## 5. Architecture used

For the Tune Source project, the thin client-server architecture would be used to develop the website. As Tune Source has a website for users to play, download music and buy gift card, a client-server architecture is needed to establish communication between Tune Source and the user to transfer back and forth the data that each party needs and wants such as music files or credit card information.



*Figure 18 Client-server model*

Thin client is a computer inside the server that handle a portion of the application before the data is finally sent to the user instead of the data being sent straight from the server's storage. This way Tune Source website and application would take less resource on the user's device, making it more accessible on lower performance devices such as low-end smartphones.



## References

Avendano, O., 2021 . *Joint Application Development (JAD)*. [Online]

Available at: <https://www.toolshero.com/information-technology/joint-application-development/>  
[Accessed 03 July 2022].

Lunalovegood, 2021. *Software Engineering | Requirements Elicitation*. [Online]

Available at: <https://www.geeksforgeeks.org/software-engineering-requirements-elicitation/>  
[Accessed 03 July 2022].

Martin, M., 2022. *What is a Functional Requirement in Software Engineering? Specification, Types, Examples*. [Online]

Available at: <https://www.guru99.com/functional-requirement-specification-example.html>  
[Accessed 03 July 2022].

Martin, M., 2022. *What is Non-Functional Requirement in Software Engineering? Types and Examples*. [Online]

Available at: <https://www.guru99.com/non-functional-requirement-type-example.html>  
[Accessed 03 July 2022].

tutorialspoint, 2022. *Requirement Gathering Techniques*. [Online]

Available at:  
[https://www.tutorialspoint.com/business\\_analysis/business\\_analysis\\_requirement\\_gathering\\_techniques.htm](https://www.tutorialspoint.com/business_analysis/business_analysis_requirement_gathering_techniques.htm)  
[Accessed 03 July 2022].