Srs spotify

Software Engineering (Lovely Professional University)

SOFTWARE REQUIREMENTS SPECIFICATION

For

SPOTIFY

Version 1.0 approved

Prepared by : Akshansh Singh

Roll No. : RK21PBA18

Reg. No. : 12102459

LOVELY PROFESSIONAL UNIVERSITY

JALANDHAR , PUNJAB.

CONTENT

1. INTRODUCTION

1.1 Purpose

This software requirement specification document has been created for the program Spotify, which is a musical app used for playing music formats and chiptunes.

It is a free app but also came with payed services for playing music without ads.

1.2 Definitions and Abbreviations

User:-

Any living being who interacting with the software is a user.

Library:-

The collection of tracks inside a directory or across multiple directories forms up a library.

Store:-

This is the persistence layer of whole system.

Classifier:

An algorithm that implements classification, especially in a concrete implementation. It is a part of recommender system.

Tags:-

A label attached to track which gives extra information about it.

1.3 Abbreviations
        none

1.4 Scope

        The product Spotify is an imaginary product that will support listening to streamed music and handle playlists on mobile phones even in off-line mode.

        It plays music and provide suggestions based on track which user listening to from both offline library which is available in user's machine as well as on internet.

1.5 References.
- Googel.com.
- Github com.
- Spotify.com

2. Overall Descriptions

2.1 Product Perspective

        This system consists of three component packaged as one desktop application:

Music Player: for playing music from local library.

Classifier: On the bases of present track playing, classifier will generate suggestions.

It provide on end-user interface. The major logical components are
- List of playlist

- Playlist of music
- music Player
- Search Screen
- Login / signup page.

2.2 Product Functions

Using this app user can play track available in offline library. While playing music user can get a list of suggested tracks which are most closely related with the present track in term of their singers, release year, rating etc. These track may be present in offline and online sources.

User can perform following functions:

- Play / pause / stop / seek, control volume
- Add tracks / remove tracks
- Update metadata.
- Get recommendation.
- Simple interface
- Repeat playlist.

2.3. User characteristics

User group 1 : Kids

Kids are consumers in the age 7-17 are very active in building and sharing playlists. They might have difficulty understanding English, so default setting and visual interaction is a must.

User group 2 : Adults

Adults are in the age 18 upwards and have well organized playlists for different modes and occasions playlists for different programs. Whenever new feature are released these groups need to be notified, they will not enjoy discovering things by curiosity.

2.4 Operating Environment

System Requirement

- Operation System should be capable of playing music and have any of mentioned OS installed.

- Internet connection is required for suggestions and metadata updation.

Supported Version

Android :	Android OS 5.0 or above.
iOS	iOS 13 or above
Mac	OSX 10.13 or above
Windows	windows 7 or above

Processor :- 1 gigahertz or faster

Ram :- 1 (GiB).

The web player

Supported web browser
- chrome
- Firefox
- Edge
- Opera
- Safari

2.5 Design and Implementation constraints

- User mail id should be valid.
- Users register no. should be valid.
- Users should be any country of the world
- Password should be minimum 8 digit with different uppercase, lowercase, and no in it
- Date of Birth format DD|MM|YY
- A valid Google and facebook account if you login with Google and facebook

2.6 Assumption And Dependencies

- Assumptions

  - Users have access to reliable Internet service.
  - Users have existing spotify accounts.

- Dependencies.

  - The 'host' of parties must have a valid spotify Premium account.
  - The 'host' of parties must be using a non-mobile device.

2.7 User Documentation

User can use the help regarding any query about the website. He /she can also ask contact us and go through the frequently asked question available on the website at the bottom of the front page.

3. External Interface Requirements

3.1 User Interface

The Spotify user interface is specially design for taking the user in mind. They make sure that the user learn to max spend most of the time to use the system not to learn how to use this system. For making it easy they use two buttons on the initial page of Netflix and those are Sign in and Join for free for a month. So, if a customer is already taken a plan then he/she can sign in and enjoy the service. and if they don't have any plan then they can click on join free for free month and enjoy the service free for a month.

3.2 Hardware Interfaces

The spotify use a well-developed data base for storing huge data amount of data use good algorithm to fetch the data from the database to the web browser of the user in different of the world.

3.3 L Software Interfaces

Spotify runs on any operating system having web browser and also have good internet speed. The spotify can run on windows, ios, android, Mac etc. If th the user has touch, then he/she can use the tou for clicking the buttons and they also click by mouse and cursor. The web component commun

with the database in order to get user rating data of the songs system. It communicates with the database in the database and the used component reads/modify operation concerning both reading and modifying the data while the communication the Music Recommender and the website consist of reading a JSON object from the recommender to the website to be displayed on the webpage.

3.4 Communication Interfaces

User can open their Spotify account on their computer, mobiles if they have the good internet speed they can listen the songs, user can change their plans easily. They can cancel anytime from anywhere in the world as a user can watch anywhere. Whenever the user open the app for listening songs he/she has to login their account if they have already make it, else they have to subscribe a plan or they have to join free from a month to listen.

4   Functional Requirement

4.1 Functional requirement 1

- Description -
    ask user to sign-up in his/her account

- Input
    Name, Email Id, Contact number and password

- Processing-

  Store all the information in database and send a confirmation code on contact number.

- Output -

  Display a message that user is register successfully and redirect user to homepage

4.2 Functional requirement 2

- Description -

  Ask user to login in then the account

- Input -

  E-mail Id and password.

- Processing-

  Store all the information in database and send a confirmation code on contact number.

- Output -

  Display a message that signed in successfully and redirect user to homepage.

4.3 Functional requirement 3

- Description -

  Search option for categories.

- Input -

  Song name or type of song

- Processing

  System check in database and match query with data

- Output -

  Display all related song data to user

4.4 Functional Requirement 4

- Description-
  Display Song

- Input-
  Click on subcategory and then on audio song.

- Processing-
  System will check in database for song related to that category.

- Output-
  All the song related to that category displayed to user.

4.5 Functional Requirement 5

- Description-
  Download

- Input-
  click to Download button

- Processing-
  System allows the user to download the song

- Output-
  Display user download song.

4.6 Functional requirement 6

- Description-
  Add to playlist.

- Input
  click on add to playlist.

* Processing:

  System will add the audio to your playlist.

* Output -

  Display playlist page.

4.7 Feed Functional requirement 7

* Description -

  Subscription page

* Input -

  Bank name, card name, no...one time password.

* Process -

  System saves your information from database

* Output -

  Display that user subscribed with the package.

4.8. Functional requirement 8.

* Description -

  Feed back

* Input -

  Name, Address, contact number and feedback

* Process -

  Save feedback into database

* Output -

  Display feedback submitted successfully message

5. Non-Functional Requirement

5.1.1 Performance requirement

Performance requirement 1.
- Description

If user is not giving response to subscription package for the amount of time it was issued to him/her the subscription will be ended.

Performance requirement 2.
- Description

If song is not there, then display audio not found

Performance requirement 3
- Description

The subscription package will be of 1 month or 3 month only

5.2. Attributes

5.2.2.1 Availability

The spotify is available 24 hours a day.

5.2.2.2 Security
- User's accessibility is censured in all the ways.
- Users are advised to change their password on regular basis.

5.2.2.3 Software Quality Attributes
- Security.
- Performance

5.2.2.4 Maintainability

Only maintainers are allowed to add new song to app.
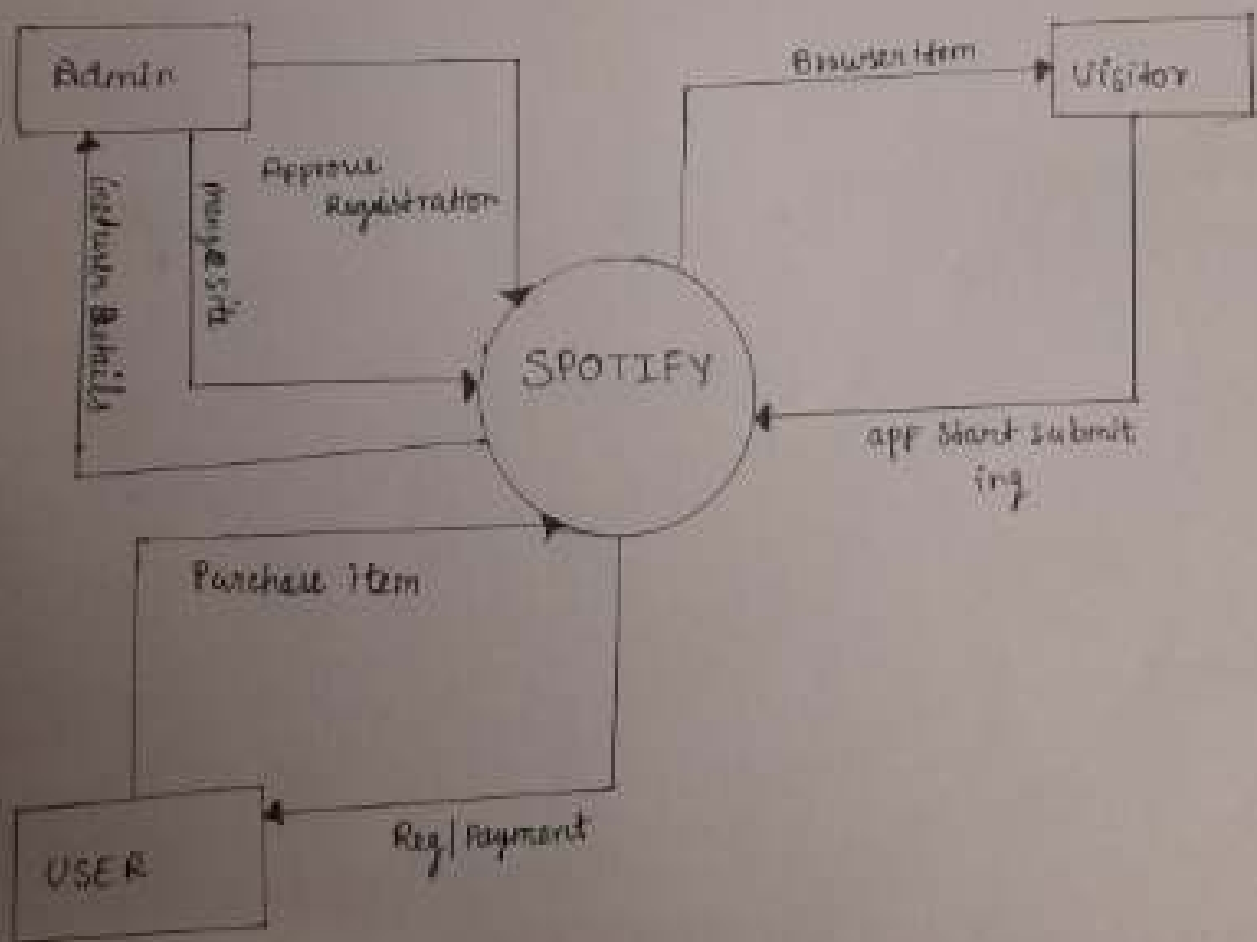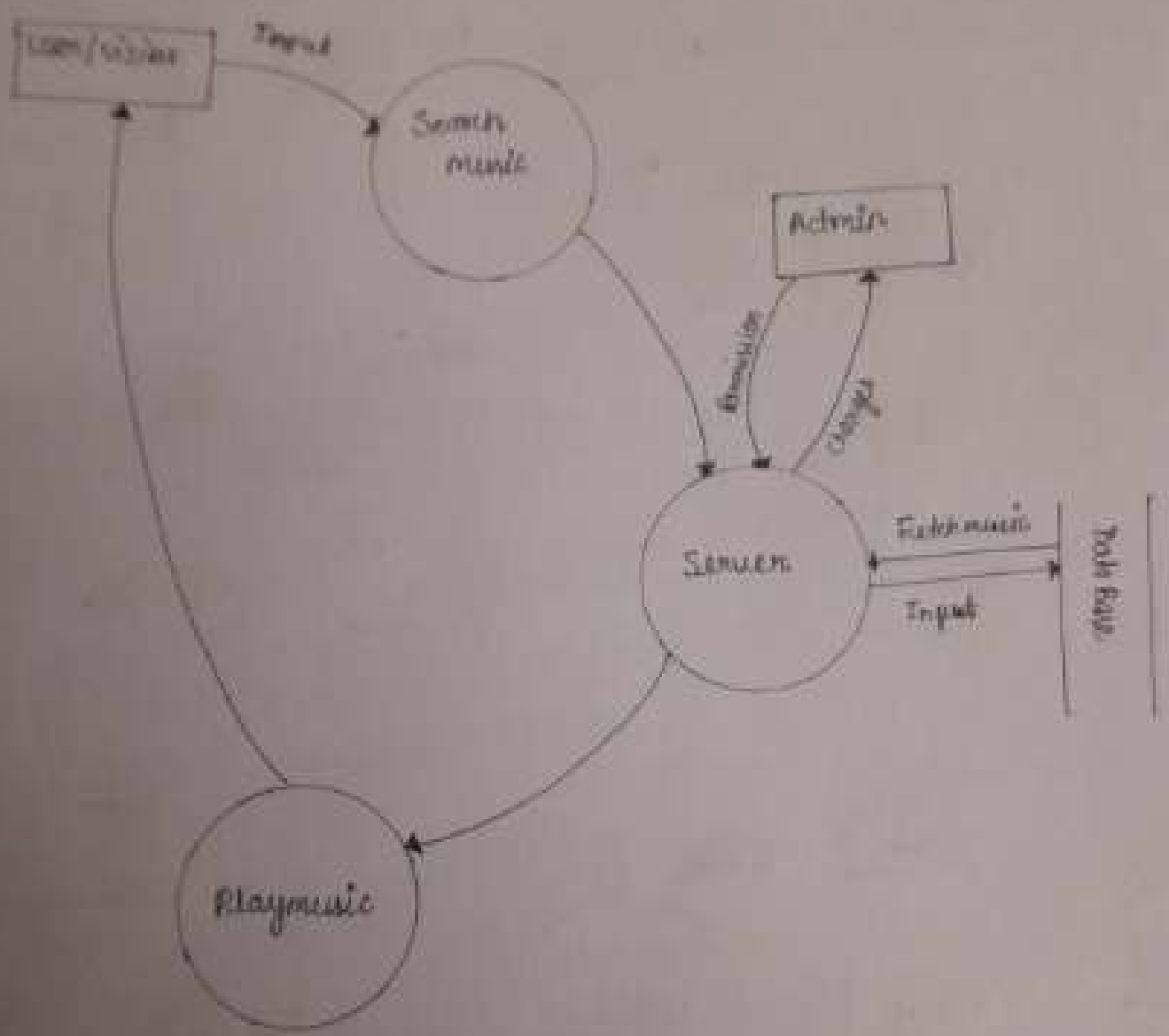
5.2.3 Other requirements

Database

- Website must able to store every kind of data means every format of data

- Website should be frequently connectable to database on it must be database sensitive

6. DATA FLOW DIAGRAM

FIRST LEVEL

FIRST LEVEL

SECOND LEVEL
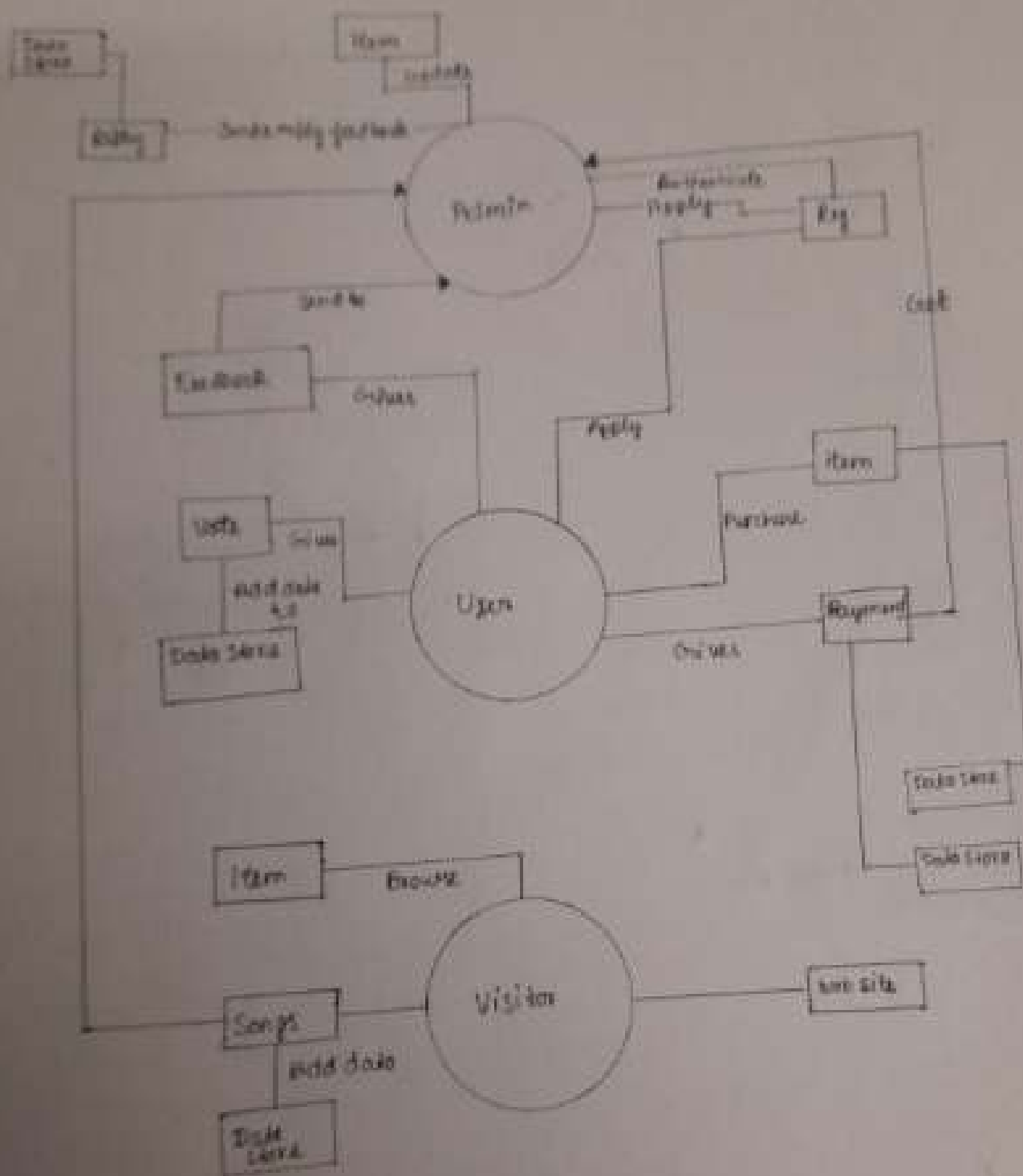
Test case # :1

System : Computer

Designed by : Akhansh Singh

Executed by: ARhansh Singh

Short Description : Test the login process

Test case Name : Login

Subsystem : Spotify

Design Date : 5/10/22

Execution Date : 8/10/22

Pre-Condition

The user has a well established internet connection

The user is already registered with spotify

The system display the login page

Step	Action	Executed system response	Pass/Fail	comment
1	Enter user ID	User ID text box furnits white	Pass	
2	Enter Password	Password text box turn white	Pass	
3	click login	The screen displays successfully login	Pass	
	Check post conditions			

Post Conditions

The screen display Home page of Spotify

Test case # : 2

Test case Name : Song search

System : Computer

Subsystem : Spotify.com

Designed by : Akshansh Singh

Design Date : 21/9/22

Executed by : Akshansh Singh

Execution date : 25/9/22

Short Description : Search for songs

Pre conditions

The user is already login to Spotify

Step	Action	Expected system response	Pass/Fail	comment
1.	Click on Search button	System displays various song	Pass	
2.	Choose songs	Select the song	Pass	
3.	Click next button	System displays the chosen song	Pass	
4.	Check post condition 1			

Post condition

The user will get the information about different songs

Test case # : 3

Test case Name : Buy subscription

System : Computer

Sub System : Spotify.com

Designed by : Akshansh Singh

Designed date : 23/9/22

Executed by : Akshansh Singh

& executed date: 24/9/22

Short Description : Test case for buying subscription policy (pack).

Pre-conditions

The user already registered with spotify

The user wants to buy a subscription pack

Step	Action	Expected System Response	Pass/Fail	comment
1.	Click on menu button	System ask to type sub-scription	Pass	
2.	Click on the subscription button	System will display various subscription	Pass	
3.	Choose subscription	System display information of the pack	Pass	
4	Click on buy button	Redirect to Payment Page	Pass	

Post Condition

The screen will display Acknowledgement Paf

Test case: 4

System : Computer

Designed by : Akshansh Singh

Executed by : Akshansh Singh

Short description : Test case for search history

Testcase Name : check history
Subsystem : spotify.com
Design Date : 26/9/22
execution date : 30/9/22

Pre-condition

The user want to search history.

The user is already registered to spotify

Step	Action	Expected System Response	Pass/Fail	Comment
1.	Click on main menu	System display different menu	Pass	
2.	Click on History	System display the history of songs search	Pass	
3.	Click on one song	play the song in the history	Pass	
4.	Check post condition 1			

Post condition

The user get history menu

Test case #: 5   Test case Name: Recomendation
System: Computer   Subsystem: spotify.com
Designed by: Akshansh Singh   Design date: 1/10/22
Executed by: Akshansh Singh   Execution date: 3/10/22
Short description: Testing of Recomendation Interface.

Pre condition

The user wants to use recomendation.

The user has a valid account.

Step	Action	Expected system response	Pass/Fail	Comment
1.	Click on Recommendation button	System display the recomendation interface	Pass	
2.	Click on the recomen-dated song	Select the song user want to play.	Pass	
3.	Check post condition 1			

Post condition

The user get recomendation interface.

Test case# : 6

System: Computer

Designed by : Abishansh Singh

Executed by : abishansh Singh

Short description: Test of cancel subscription

Test case Name : Test of cancel subscription

Subsystem: spotify com

Design Date : 4/20/22

execution date : 6/20/22

Pre-condition:

User already have registred on spotify

User want cancel subscription

Step	Action	Expect system-response	Pass/Fail	comment
1.	click on main menu	System display different option	Pass	
2.	click subscription	System displays the subscription details.	Pass	
3.	click on Cancel button	System display the user subscri	Pass	
4.	check post condition			

Post condition

The user get cancilation slip.

Test case # : 7

System: Computer

Designed by: Akshansh Singh

Executed by: Akshansh Singh

Short description: Test case for support assistance.

Pre-condition

The user want to enquire.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	Click on support button	System want the query	Pass	
2	Enter query	System display the query & ask for feedback	Pass	
3.	Enter feedback	system displays feedback submitted	Pass	
4.	Check post condition			

Post condition

The user get solution of query.

Test case # : 8   Test case Name : change Password
System : Computer   Subsystem : Spotify.com
Designed data by : Abhishek Singh   Design date : 3/29/22
Executed by : Abhishek Singh   Execution date : 12/10/22
Short description : Test case for change of account
                   password.

Pre - condition

The user have a valid spotify account.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	Click on my account	System display menu	Pass	
2.	Click on account setting	system displays account setting	Pass	
3.	change Password	system display 'otp send' and ask to enter it.	Pass	
4.	enter OTP	system display successful	Pass	
5.	check post condition			

Post condition
-the new password will be saved at database.

Test case: 3

System: Computer

Designed by: Siddharth Singh

Executed by: Siddharth Singh

Short description: Test case for downloading song

Test case Name: Song for download

Subsystem: spotify.com

Design date: ...

Execution date: ...

Pre-condition

The user wand to download the song
The user is already have account on spotify

Step	Action	Expected system Response	Pass/Fail	Comment
1	Click on song	System display the song	Pass	
2	click on download button	system downloaded the song.	Pass	
3	check post condition 1			

Post condition

The user get the song downloaded in his/her account

Test case # : 10

System : computer

Designed by : Akshansh Singh

Executed by : Akshansh Singh

Short description : Test case for edit profile

Test case Name : edit profile

Subsystem : spotify.com

Design date : 15/10/22

execution date : 18/10/22

Pre-condition

The user have valid account on spotify

Step	Action	expected system response	Pass / Fail	commen
1.	click on edit profile	System displays user profile	Pass	
2.	enter the name + valid no.	System create new profile	Pass	
3.	check post condition L			

Post condition L

User get new profile.