



Srs spotify

Software Engineering (Lovely Professional University)

SOFTWARE REQUIREMENTS SPECIFICATION

For

SPOTIFY

Version 8.1.0 approved

Prepared by : Akshansh Singh

Roll No. : RK21PBA18

Reg. No. : 12102459

LOVELY PROFESSIONAL UNIVERSITY
JALANDHAR, PUNJAB.

CONTENT

1. Introduction

- 1.1 Purpose
- 1.2 Definitions and Abbreviation
- 1.3 Abbreviation
- 1.4 Scope
- 1.5 Reference

2. Overall Description

- 2.1 Product Perspective
- 2.2 Product function
- 2.3 User Characteristics
- 2.4 Operating Environment
- 2.5 Design and Implementation constraints
- 2.6 Assumption And Dependencies
- 2.7 User Documentation

3. External Interface Requirement

- 3.1. User Interface
- 3.2. Hardware Interface
- 3.3. Software Interface
- 3.4. Communication Interface

4. Functional Requirement

- 4.1. Functional requirement 1
- 4.2. Functional requirement 2
- 4.3. Functional requirement 3
- 4.4. Functional requirement 4
- 4.5. Functional requirement 5
- 4.6. Functional requirement 6
- 4.7. Functional requirement 7

4.8 Functional requirement 8

5. Non-Functional Requirement

S-1. Performance requirement

S-2. Attributes

S-3. Maintainability

S-4. Other requirements

6. Data Flow Diagram

6.1 Zero level dfd

6.2 First level dfd

6.3 Second level dfd

7. Test cases

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 come with payed services for playing music without ads.

1.2 Definition 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.

13 Abbreviations

none

14 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.

15 References

- Google.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 basis of present track playing, Classifier will generate suggestions.

It provide an end-user interface. The major logical components are

- List of playlist

- Playlist of music
- Music Player
- Search Screen
- Login / Sign up 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 moods and occasions playlists for different programs. Whenever new features 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 GB.

The web player

Supported web browser:

- Chrome
- Firefox
- Edge
- Opera
- Safari

2.5 Design and Implementation constraints

- User mail id should be valid.
- User register no. should be valid.
- User should belong country of the world.
- Password should be minimum 8 digit with different upper case, lower case 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 available 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/customer 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 and good algorithm to fetch the data from the database to the web browser of the user in different of the world.

3.3 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 the user has such, then he/she can use the tool for clicking the buttons and they also click by mouse and cursor. The web component commun

with the database in order to get some history logs of the changes system. The communication with the database in the database and the web component handles operation concerning with inserting and modifying the data, while the communication like Music Recommendation and the website consist of sending a HTML object from the database to the website to be displayed on the webpage.

3.4 Communication Interfaces

User can open their Spotify account on their computer, mobile if they have the good internet speed they can listen the songs, user can change their plans easily. They can access anytime from anywhere in the world as a user can watch anywhere. Whenever the user open the app for listening songs, it asks 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 3 month to listen.

4 Functional Requirement

4.1 Functional requirement 1

- Description -

Ask user to sign-up in his/her account.

- Input

Name, E-mail 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 their 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 video songs.
- 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. Feb 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 -

Feedback

- 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 used 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 ensured in all the ways.
- Users are advised to change their password on regular basis.

5.2.2.3. Software Quality Attributes

- Security.
- Performance.

S.2.2.4 Maintainability

Only maintainers are allowed to add new song to app

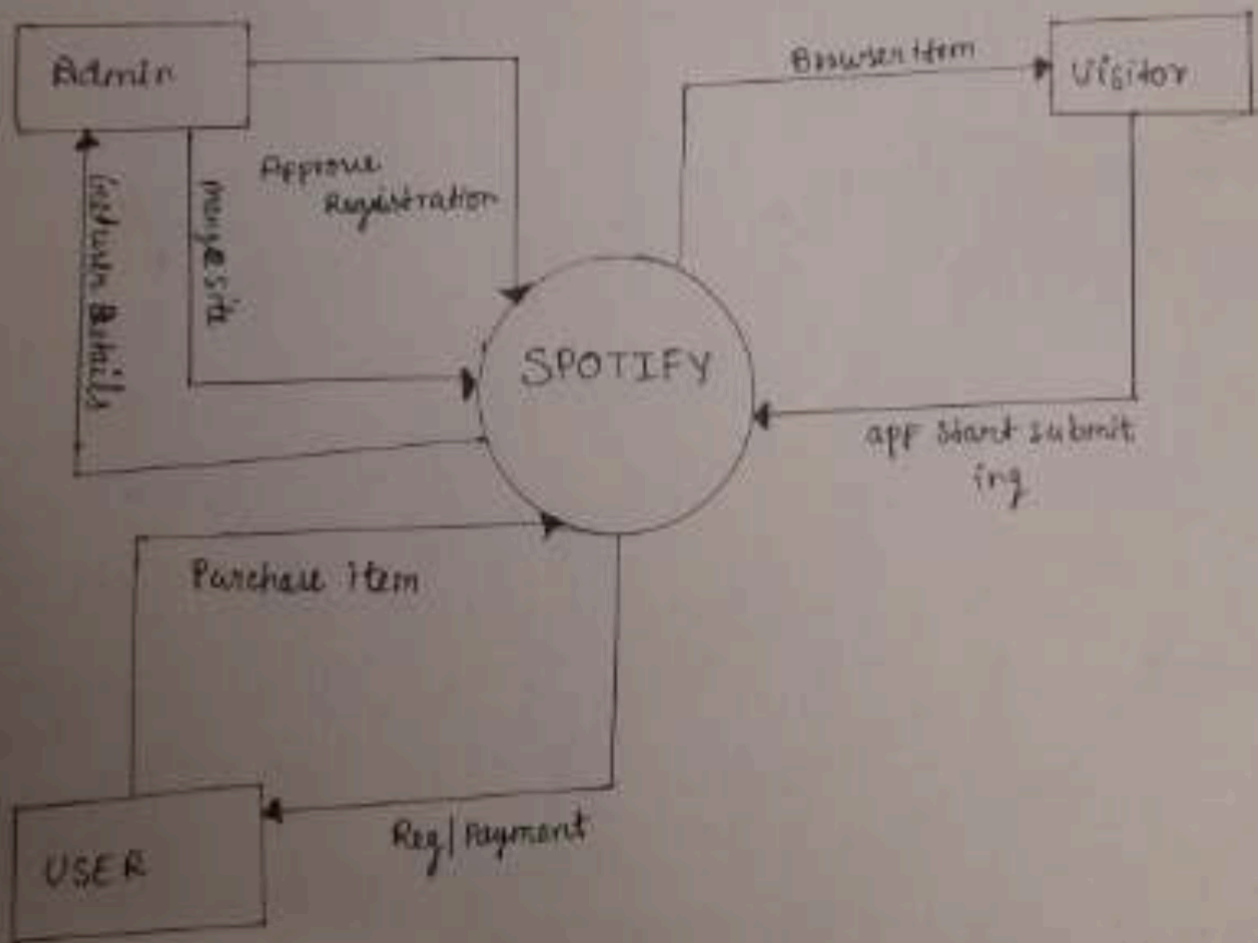
S.2.3 Other requirements

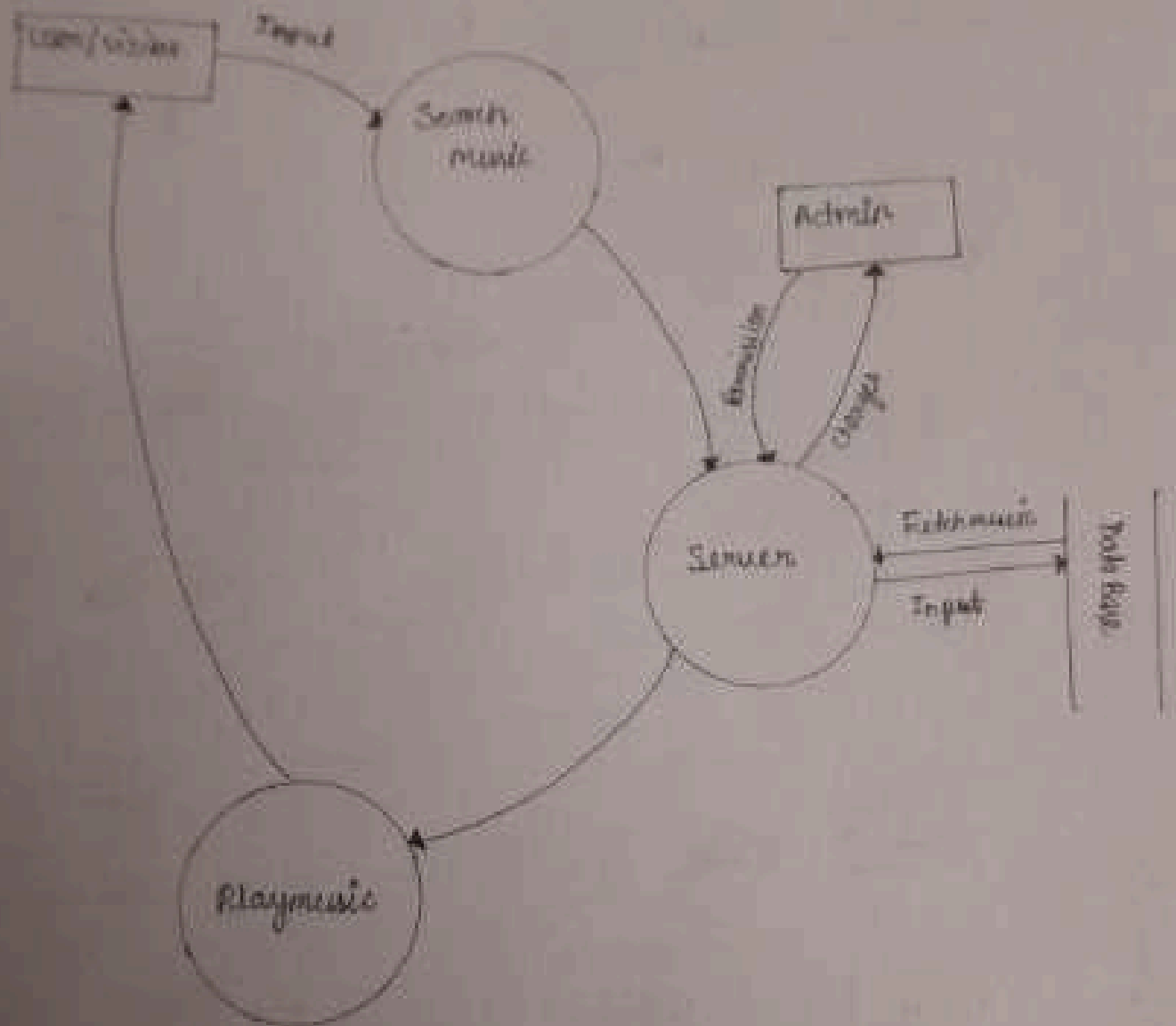
Database

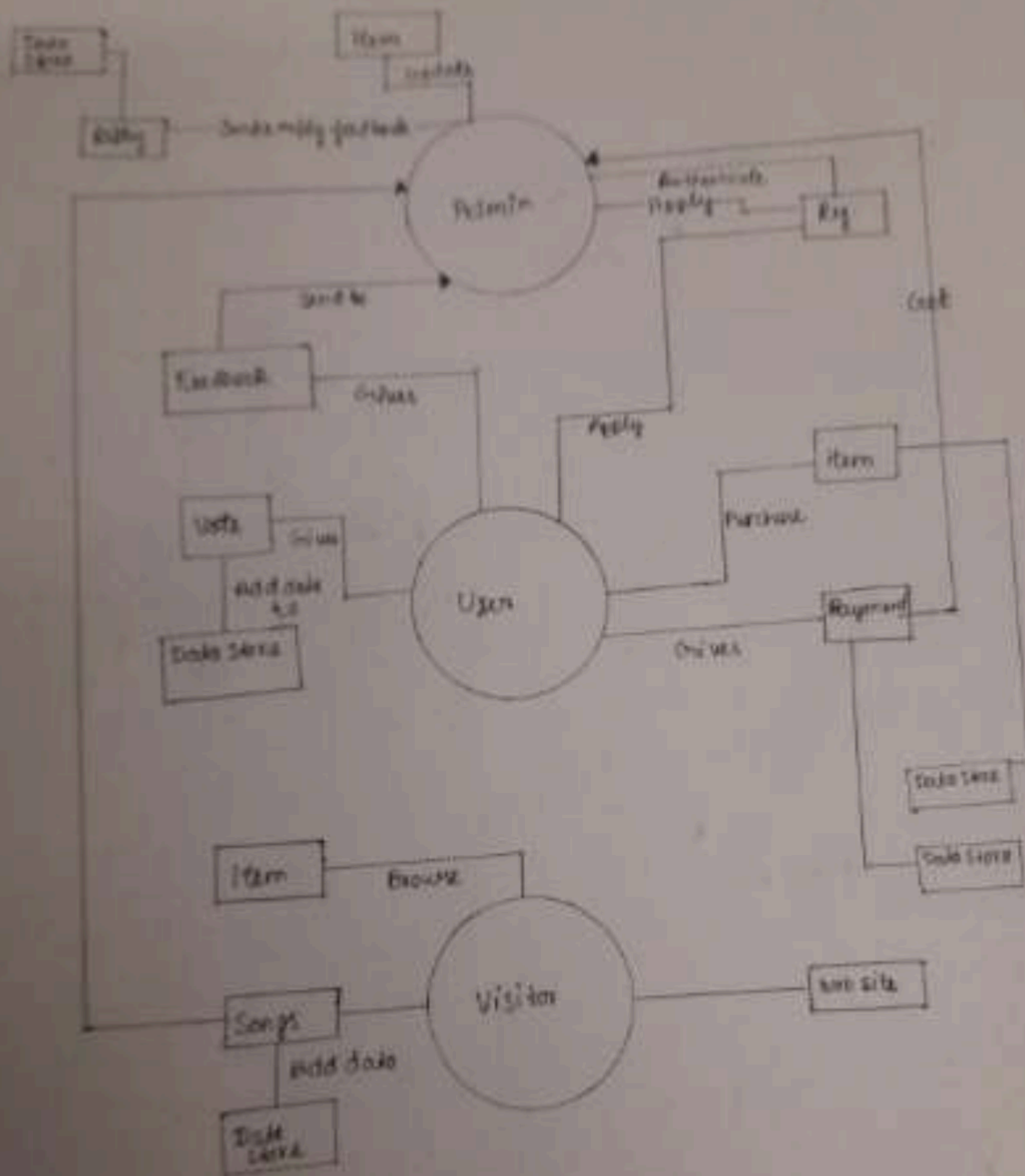
- Website must able to store every kind of data means every format of data
- Website should be frequently connectable to database or it must be database sensitive.

6. DATA FLOW DIAGRAM

FIRST LEVEL







Test case # : 1

System : Computer

Designed by : Adithyan Singh

Executed by : Adithyan Singh

Short Description : Test the login process

Test case Name : Login

Subsystem : Spotify

Design Date : 5/10/22

Execution Date : 8/20/22

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 turn white	Pass	
2	Enter Password	Password text box turn white	Pass	
3	click login	The screen displays successfully login	Pass	
	check post condition			

Post-Conditions

The screen display home page of Spotify

Test case # : 2

System : Computer

Designed by : Akshay Singh

Executed by : Akshay Singh

Short Description : Search for songs

Test case Name : Song search

Subsystem : Spotify.com

Design date : 21/9/22

Execution date : 25/9/22

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 song	Select the song	Pass	
3.	Click next button	system displays the chosen song	Pass	
4.	check post condition			

Post condition

The user will get the information about different songs

Test case #: 3

System: Computer

Designed by: Akshay Singh

Executed by: Akshay Singh

Short Description: Test case for buying subscription pack.

Test case name: Buy subscription
Sub System: Spotify.com
Designed date: 23/05/22
Executed date: 24/5/22

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 subscription	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 Pop

Test case ID: TC-001

System: Computer

Designed by: Akshay Singh

Executed by: Akshay 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			

Post condition

The user get history menu

Test case #1: 5

System: Computer

Designed by: Akshansh Singh

Executed by: Akshansh Singh

Short description: Testing of Recommendation Interface.

Test case Name: Recommendation

SubItem: spotify.com

Design date: 3/20/22

Execution date: 3/20/22

Pre condition

The user wants to see recommendation.

The user has a valid account.

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

Post condition

The user get recommendation interface.

Test case ID: 6

System: Computer

Designed by: Abhinav Singh

Executed by: Abhinav Singh

Short description: Test of cancel subscription

Test case Name: Test of cancel subscription

Subsystem: Spotify.com

Design Date: 4/20/22

Execution Date: 6/10/22

Pre-condition:

User already have registered 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 subscrii	Pass	
4.	check post condition			

Post condition

The user get cancellation slip.

Test case #: 2

System: Computer

Designed by: Akshay Singh

Executed by: Akshay Singh

Short description: Test case for support enquiries

Test case Name: Support

Subsystem: Spotify

Design date: 7/10/22

Execution date: 8/10/22

Pre-condition

The user want to inquire

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

System: Computer

Designed date by: Akshay Singh

Executed by: Akshay Singh

Short description: Test case for change of account password.

Test case Name: change Password

Subsystem: Spotify.com

Design date: 3/20/22

Execution date: 12/10/22

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 displays successful	Pass	
5.	check post condition			

Post condition

The new password will be saved at database.

Test case 3

System : Computer

Designed by : Akshay Singh

Executed by : Akshay Singh

Description : Test case for download song.

Test case Name : Test for download
Subscription : spotify.com
Design date : 23/10/20
Execution date : 24/10/20

Pre-condition

The user want 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 : Akshay Singh

Executed by : Akshay 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	comment
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 1			

Post condition 1

User get new profile.