Theresa Pham

CSC 415: Software Engineering

Final Evaluation – Take Home Component

Take Home Component Questions & Diagrams

1. The Spotify WorldView app does little to breach the ACM Code of Ethics. It satisfies both general moral imperatives as well as specific professional responsibilities as outlined in the Code of Ethics. Point 1.1, to "contribute to society and human well-being", entails creating a project that will be used in a socially responsible way and will not harm others. Spotify WorldView has complied with this imperative as by being a means of searching for albums and finding where in the world they are available, there is no chance of communication with other people in a way that would lead to harm, and no harmful information being transmitted to users after searching. Also with the new log in functionality, even if someone were able to log into another person's Spotify, there is little a hacker could do change anything with the user's account in my application. The app allows users to become more aware of where their favorite albums are available in the world. Point 1.6, to "give proper credit for intellectual property" means to give credit where credit is due. In the case of my application, it is very clear that the database of music and its associated information comes from Spotify from the application title, and the log in occurs through Spotify, indicating that my application is working off of the Spotify Web API and that the music library does not come from myself. Because the project is also working off an existing open source project, credit is given to the creator of the original open source project.

   Looking at the more specific professional responsibilities, point 2.1, to "strive to achieve the highest quality, effectiveness and dignity in both the process and products of professional work" means striving for quality and excellence with the system at hand and ensure there are no serious consequences that could occur because of a poor quality system. For Spotify WorldView, because it is working off of the Spotify API, it is built off of a system that is already of high quality and excellence. The Spotify WorldView app specifically though provides many means of searching for an album via the search bar, entering the URI onto the map interface, and even taking the saved albums from a specific

Spotify user's account. A user can also simply view a list of the countries that an album is available in, or go to the map interface to visualize the countries where an album is available. Point 2.2, to "acquire and maintain professional competence", involves the developer taking initiative to learn and gain knowledge in order to engineer their application to the best of its abilities, whether that is done through classes, independent study, or attending seminars. While courses helped me in my foundation of programming, for the majority of the project, independent studying had to be conducted in order to learn the languages needed for the project as well as learn how to ensure everything would interface properly and be functional. For my application, I had to learn to use Ruby, Ruby on Rails, HTML/CSS, and even Javascript as that was the language of the original open source code. I also had to learn to work with the Spotify Web API, which was helped greatly through usage of a Ruby wrapper called RSpotify that made calls through the Spotify API in a way that functioned well with the Ruby language.

2. a) Name of Project: Spotify WorldView

   Language Used: Ruby (on Rails)

   Major Stakeholders/Users: Music Listeners, Spotify Users, Record Label Employees

   New User: Spotify User (w/ Account)

   - Before all users and major stakeholders all had the same access to functionalities they could do, but now I am implementing a new functionality that will be exclusive to a Spotify User with an account.

   New Functionality: Log in through Spotify and view information on saved albums

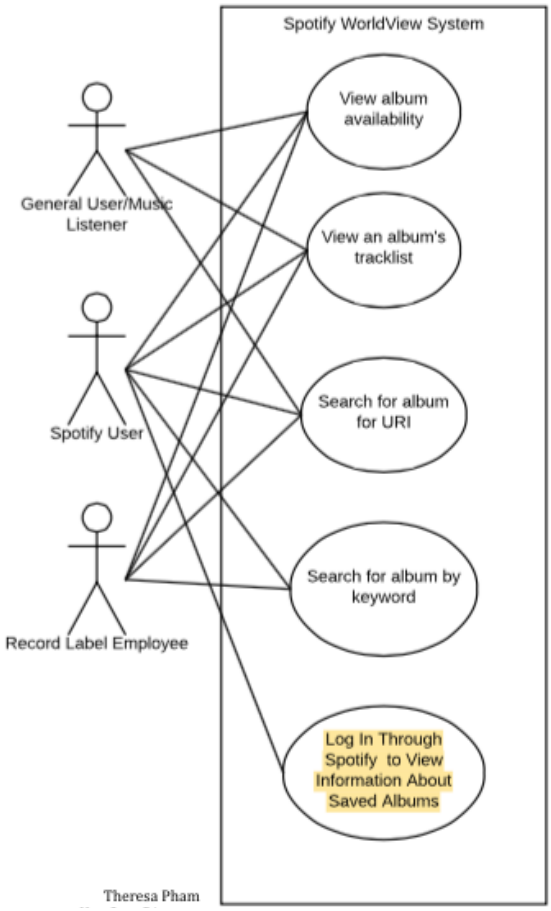
   - This new functionality will pull a user's saved albums from their account so they can view album tracklists and availability using the map interface

   Github URL: https://github.com/itstheresa/spotifyWorldView

   All updated diagrams are below as well as uploaded to the Github docs folder.

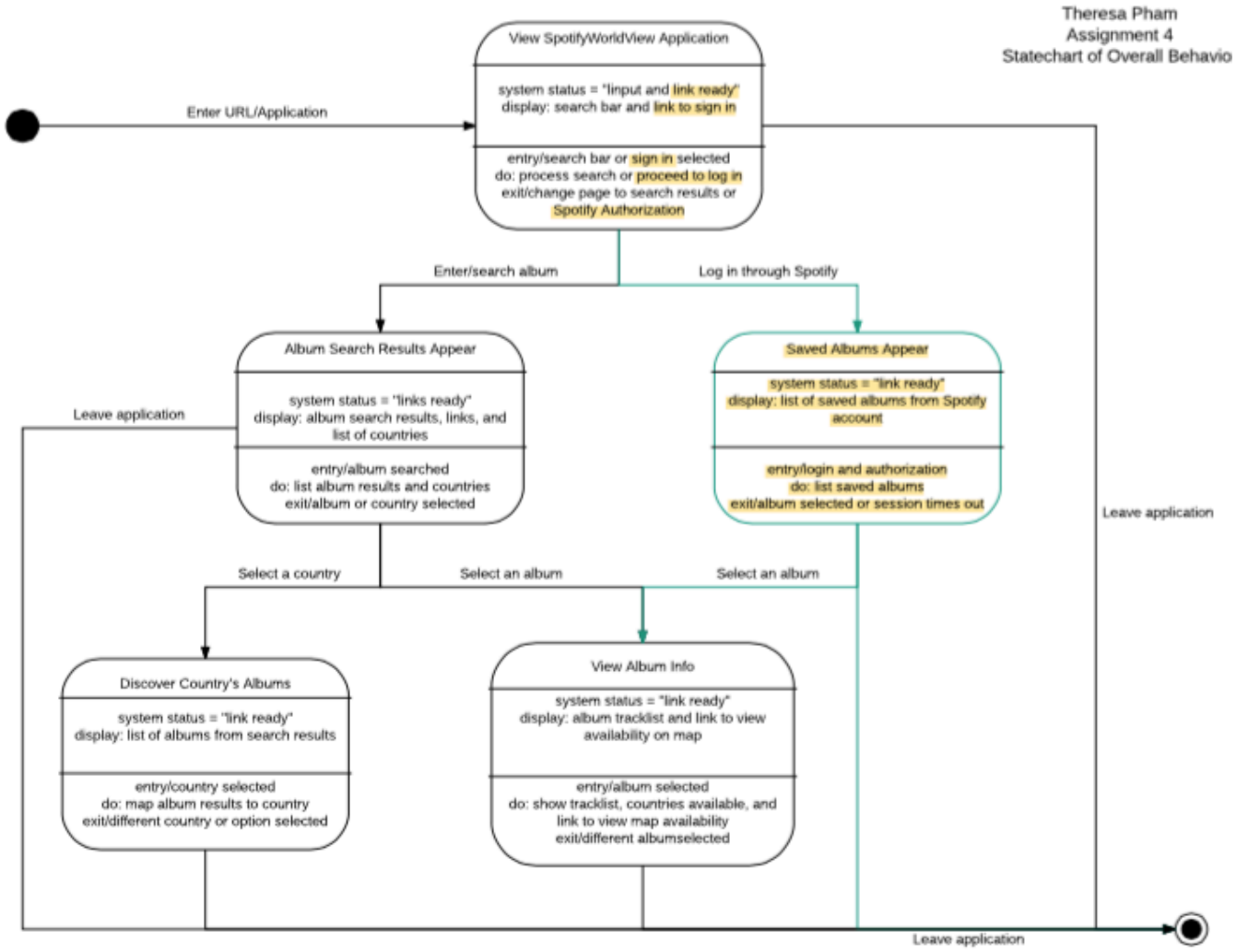
Note: All changes made to existing diagrams are highlighted or made to be a different color.

b) Updated Use Case Diagram + New Use Case Description:



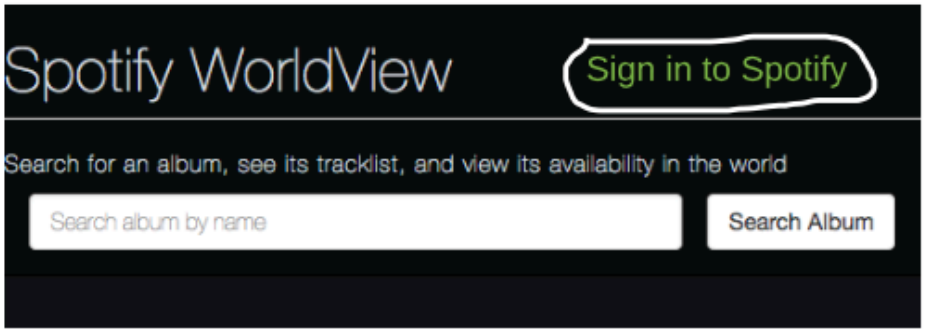
Name	Log In Through Spotify to View Information About Saved Albums
Summary	Users with Spotify accounts can log in to pull up a list of saved albums from their account and view the tracklist and availability of those albums
Actors	Spotify Users
Preconditions	1. The user has entered the application by entering the URL 2. The user has a Spotify account
Normal Scenario	1. The user goes to the website 2. The user signs into their Spotify account 3. List of saved albums in account appear 4. User selects an album to view information about 5. The tracklist and availability of the album are shown
Exception Scenario	No albums saved in account

c) Updated System State Chart Diagram:

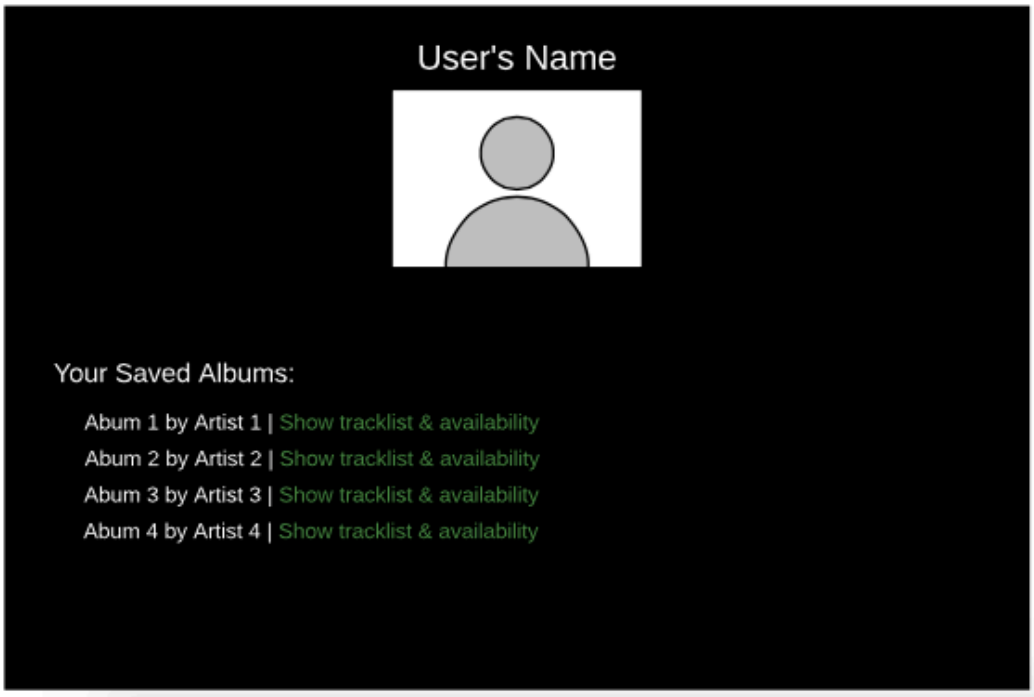


d) UI Mock-Up for New Functionality:

Add link to sign into Spotify on the front page



After signing in, account information will appear to inform user that login was successful, and saved albums from Spotify will appear in a list with links to view tracklist and availability



e) Test Case Design for New Functionality:

When testing to see whether an application as a whole works, system testing is used. For system testing, the black box method is used most often, where I acted as if I was a user and testing all possible cases that could be encountered. This was the method used to test my new functionality. For the new functionality implemented of logging in and viewing information on saved albums, there is little room for a user to break the system. Since the functionality relies on the Spotify API, there already exists error handling for invalid username/password when trying to log in.

Test Cases

Functionality Tested	**Inputs**	**Expected Outputs**	**Actual Outputs**
Log in to Spotify	Click link + Enter valid Spotify username & password	Redirects to page with user's name, profile picture, country, and a list of their saved albums	Works – redirects to show account info and saved albums from Spotify
Indication to user that they have no albums saved (if this is the case)	Valid log in information	Shows message "No albums saved"	Works – displays message
Show tracklist of a searched album	Keyword search + Album selection	List of songs in album	Works – redirects
Link to map availability interface for an album	Select an album + Click link to view availability	Redirect to map interface page with countries available highlighted for album selected	Works – redirects properly to Github page with correct album on map interface
Logout	Click link to logout	Redirects user to home page	Works - redirects user to home page

3. b) See actual output for test cases above