

## PlaySwap Milestone 5 (user testing)

Latest commit:

<https://github.com/jwoody02/PlaySwap/commit/6f161f0f7bb1a71e9ba9d4dae4fa1909e13c8ad3>

### Team Members:

- Jordan Wood
- Clay Fricke
- Jack Higgins
- Logan Chayet
- Preston Dotsey

### Test Plans

- Create a document that describes how at least three features within your finished product will be tested.

### Features to be Tested:

- Paste links for playlists
  - Function to test (ViewController.swift): `@objc func textFieldDidChange(_ textField: UITextField)`
  - When Transferring from spotify
    - Code to test: `if(str.contains("open.spotify") && str.contains("playlist"))`
      - Explanation: checks the search query (or the search bar's text) to see if it has any spotify-based url elements that could indicate whether its a playlist url the user has pasted in
    - How to test:
      - (1) Paste spotify playlist urls repeatedly to make sure it loads all songs and no ui bugs happen
      - (2) paste spotify PRIVATE playlist urls to see if it will load properly (or document whether this works)
      - (3) paste urls from other spotify elements (e.g. the url to a song on spotify) to see how the app reacts + fix accordingly
      - (4) paste invalid urls to see how app reacts (paste links without the playlist ID, etc)
    - When transferring from apple music
      - Code to test: `if(str.contains("music.apple") && str.contains("playlist"))`
        - Explanation: similar to Spotify login, but checks for a specific URI and then posts a query to the Apple music API. It will only then check if it meets certain string requirements
      - How to test:
        - (1) Paste Apple music playlist urls repeatedly to make sure it loads all songs and no ui bugs happen
        - (2) paste Apple music PRIVATE playlist urls to see if it will load properly (or document whether this works)
        - (3) paste urls from other Apple music elements (e.g. the url to a song on Apple music) to see how the app reacts + fix accordingly

- (4) paste invalid urls to see how app reacts (paste links without the playlist ID, etc)
- Login to both services
  - Function to test (ViewController.swift):`hideLoginpage1Elements()`, `loadSpotifyLogin()`, `addLoginpage2Elements()`
  - Spotify Login
    - How to test:
      - (1) Click login button for spotify service
      - (2) Enter login information
      - (3) Check if faulty login information gives the correct information
      - (4) Check if correct login information is validated and allows access to the user's library
  - Apple Music Login:
    - Code to test (line 970): `SKCloudServiceController.requestAuthorization`
      - & test response codes + error codes
    - How to test:
      - (1) Click login button for Apple Music service
      - (2) Enter login information
      - (3) Check if faulty login information gives the correct information
      - (4) Check if correct login information is validated and allows access to the user's library
- Search for Playlists
  - After logging in to either spotify or apple music, the user should be presented with a search bar
  - Tests:
    - (1) click search bar and it should animate and expand toward the top of the screen
    - (2) tap again and begin typing the name of a playlist
    - (3) playlist search results should show up and should show the playlist cover photo, the playlist name, and the author of the playlist
    - (4) the user should then be able to click a playlist and view the playlists' contents + info

## Individual Contributions

- Jordan Wood: Worked with Jack & Clay to get the apple music playlist creation to work and verify that removing the warnings didn't break any other code. Transferring to Apple music still needs some testing and refinement but the base code is working with little to no errors  
 Jordan's Latest Push:  
<https://github.com/jwoody02/PlaySwap/commit/6f161f0f7bb1a71e9ba9d4dae4fa1909e13c8ad3>

- Clay Fricke: Did more testing for our current UI and added solutions with Jordan. Worked with Jordan and Logan to resolve the errors. Additionally, fixed warnings that occurred in the code with Logan. Ensuring that variables that were dynamically established variables were converted to 'var' and statically established variables were converted to 'let'. Additionally, assisted with Apple Music transfer with Jordan and Jack.

Clay's Latest Push:

<https://github.com/jwoody02/PlaySwap/commit/fea676c79cdbcdc25cb24b15dc186251d09fdc46>

[Helped Logan with Warnings \(pushed from his account\)](#)

- Jack Higgins: Worked with Clay + Jordan to integrate url requests into the actual code for Apple music playlist creation and adding songs.

Jack's Latest Push:

<https://github.com/jwoody02/PlaySwap/commit/6f161f0f7bb1a71e9ba9d4dae4fa1909e13c8ad3>

- Logan Chayet: Fixed warnings that popped up in the code. Made sure variables that were dynamic were converted to 'var' variables and static variables were converted to 'let'

Logan's Latest Push:

<https://github.com/jwoody02/PlaySwap/commit/ce56165846f24d5ba08a6153fca91377aa5e91b0>

- Preston Dotsey: contributed to testing criteria brainstorming, testing code to check for potential errors, UI improvement

