

Team Names (Github):

- Scott Young (scottlyoung)
- Amjad Alharbi (amjaad)
- Joe Alsko (JAlsko)
- Jonah Jacobsen (jjjacobsen)

Project Title: Web-based Music Player

Project Summary: A simple web-based Music player system that allows users to listen to their favourite songs, create and edit their own playlists.

Completed Features:

| User Requirements | | |
|-------------------|---|----------|
| ID | Description | Priority |
| UR01 | As a user, I want to be able to search for songs by song title | Medium |
| UR02 | As a user, I want to be able to search for songs by artist name | Medium |
| UR03 | As a user, I would like to create my own playlist | High |
| UR04 | As a user, I want to add a song to a playlist that has been previously created | High |
| UR05 | As a user, I want to remove a song from a playlist that has been previously created | Low |
| UR06 | As a user, I want to play a song | High |
| UR07 | As a user, I want to play a playlist | High |
| UR08 | As a user, I want to view a song's info | High |
| UR09 | As an admin, I want to remove a song from the database | Low |
| UR10 | As an admin, I want to add a song (w/ attributes) to database | High |

\

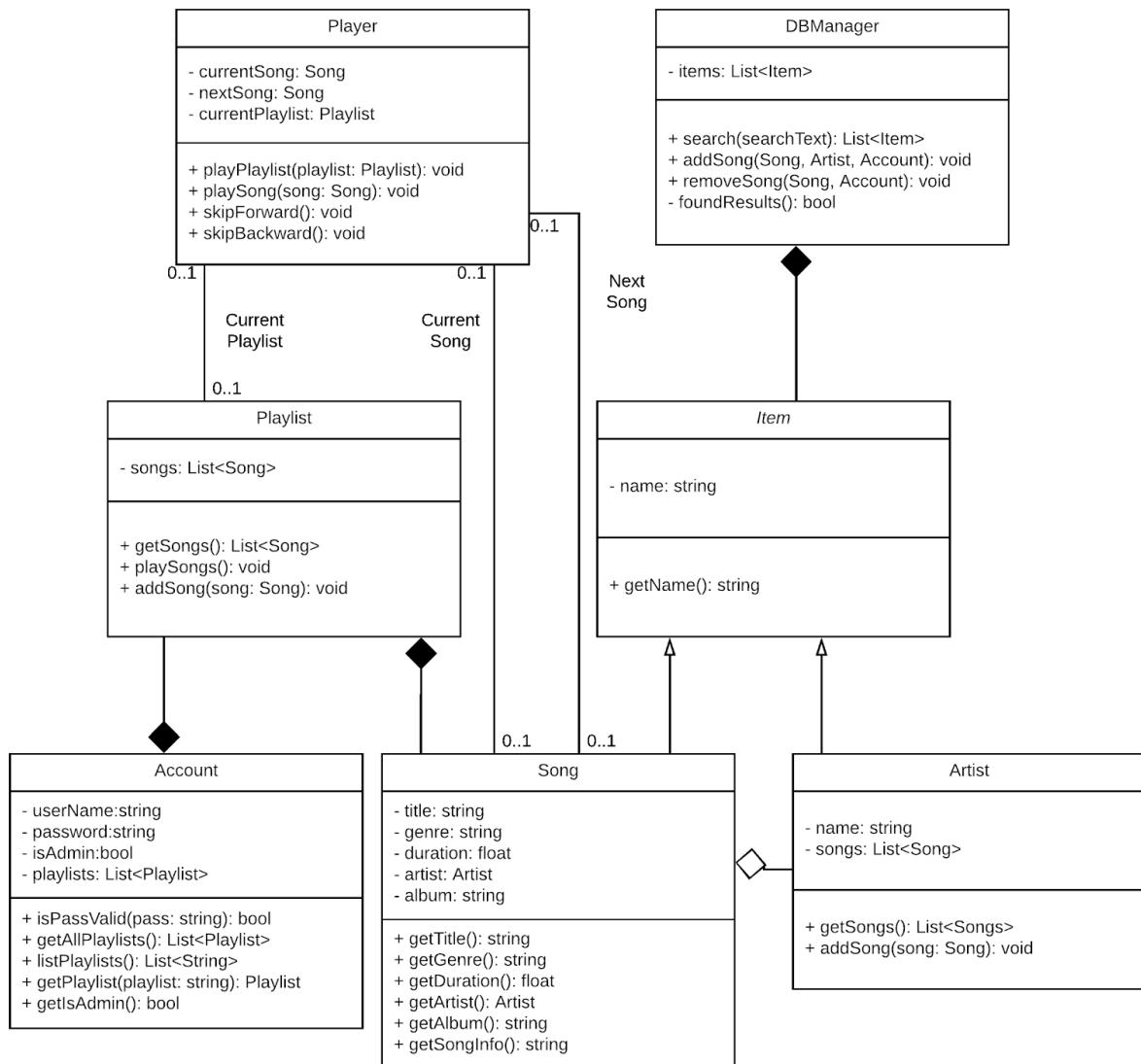
| Functional Requirements | | |
|-------------------------|---|----------|
| ID | Description | Priority |
| FN01 | Searching for songs/artists should autocomplete based on user input | Medium |
| FN02 | Adding songs to the database includes updating a list of existing song names/artists for the purpose of searching | Medium |
| FN03 | Maintain list of users and their passwords | High |

| Non-Functional Requirements | | |
|-----------------------------|------------------------------|----------|
| ID | Description | Priority |
| NF02 | Pages load within 10 seconds | Medium |

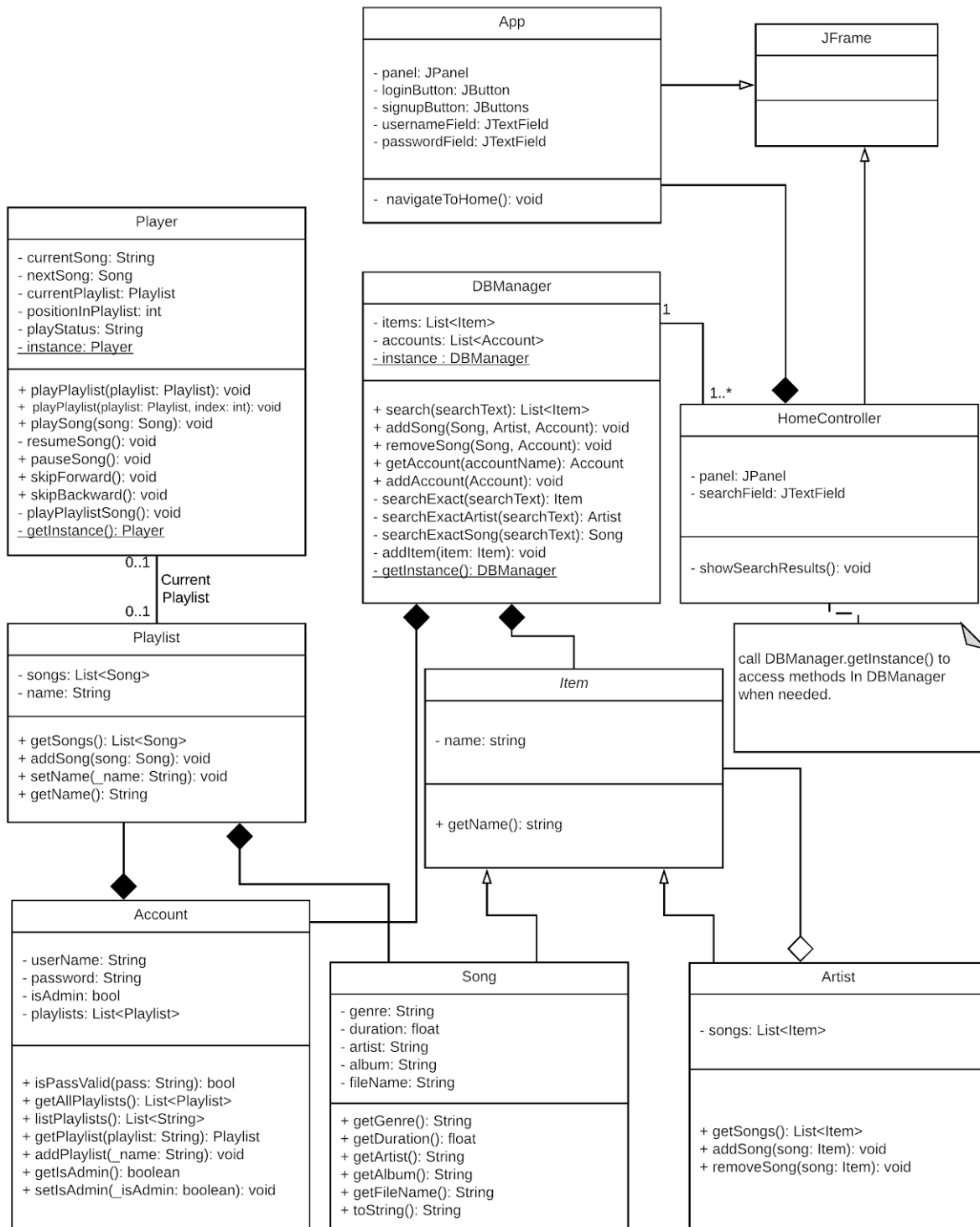
Incomplete Features:

| Non-Functional Requirements | | |
|-----------------------------|---|----------|
| ID | Description | Priority |
| NF01 | Pages should be aesthetically pleasing | Medium |
| NF03 | Search autocomplete searches within 3 seconds | Medium |
| NF04 | Songs stream faster then they play | Medium |

- **Previous Class Diagram:**



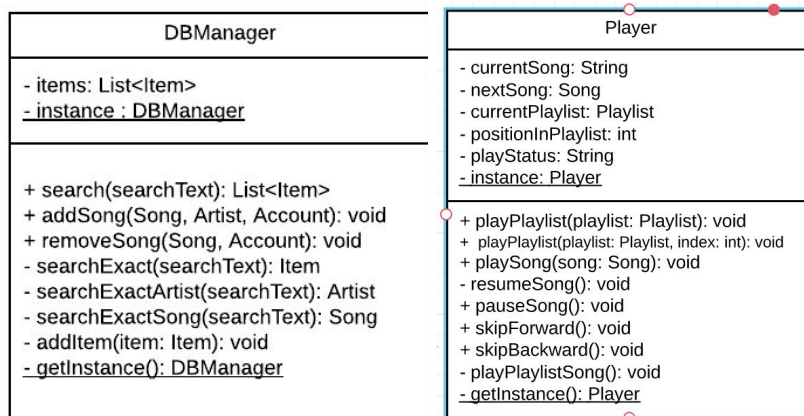
- **What changed? Why? If it did not change much, then discuss how doing the design up front helped in the development.**



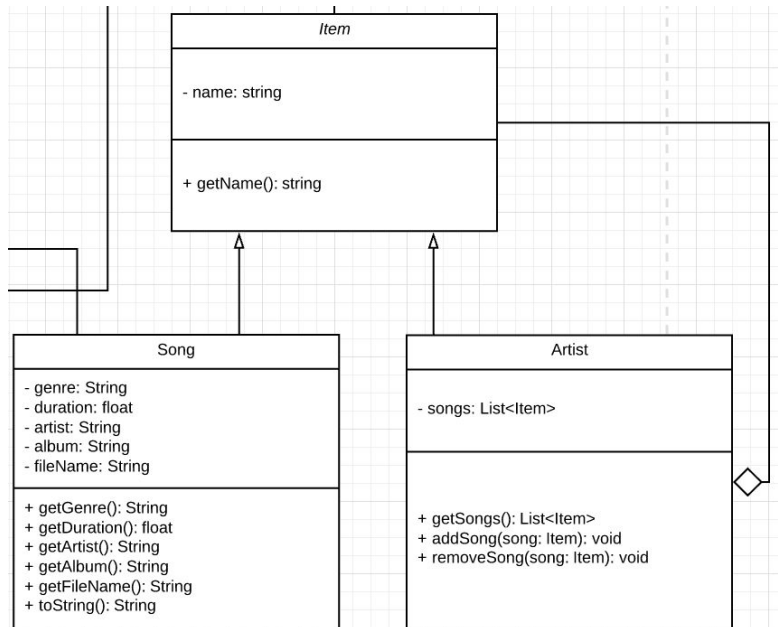
- DBManager and Player classes are both implemented as singleton, since only one instance of each class should exist in at any time.
- Added views-controllers classes so that the client doesn't directly interact with the database proxy.
- Not much changed with the Item class and its subclasses. This is because we determined a need for the composite design pattern early on in order to implement the search function (so that artists *and* songs showed up in the search results)
- Because of the lack of major changes to a large part of our class diagram, there ended up being very little work to do in regards to the implementation of the project.

5. Design pattern (each design pattern as a separate image in the .PDF).

Singleton



Composite



6. What have you learned about the process of analysis and design now that you have stepped through the process to create, design and implement a system?

- As well as functional requirements, user requirements are very important for determining system structure and which design patterns to use.
- Getting the basic objects' structure right early on makes implementing other, larger classes that handle these objects much easier and reliable. With the song/playlist/artist classes complete, implementation of methods in the Player class was much simpler and more scalable
- It can be very important to think about potential expansion to the software in the early stages of development. If the software is designed wrong, expansion can be nearly impossible without completely redoing the code. We encountered this with the implementation of network communication for our software. We had originally planned it around having the client and server being the same program. Later, when we wanted to expand this to be a proper client and server, we found that our lack of planning for this made it an extremely difficult task, and we had to abandon the idea of expanding our software in this way.