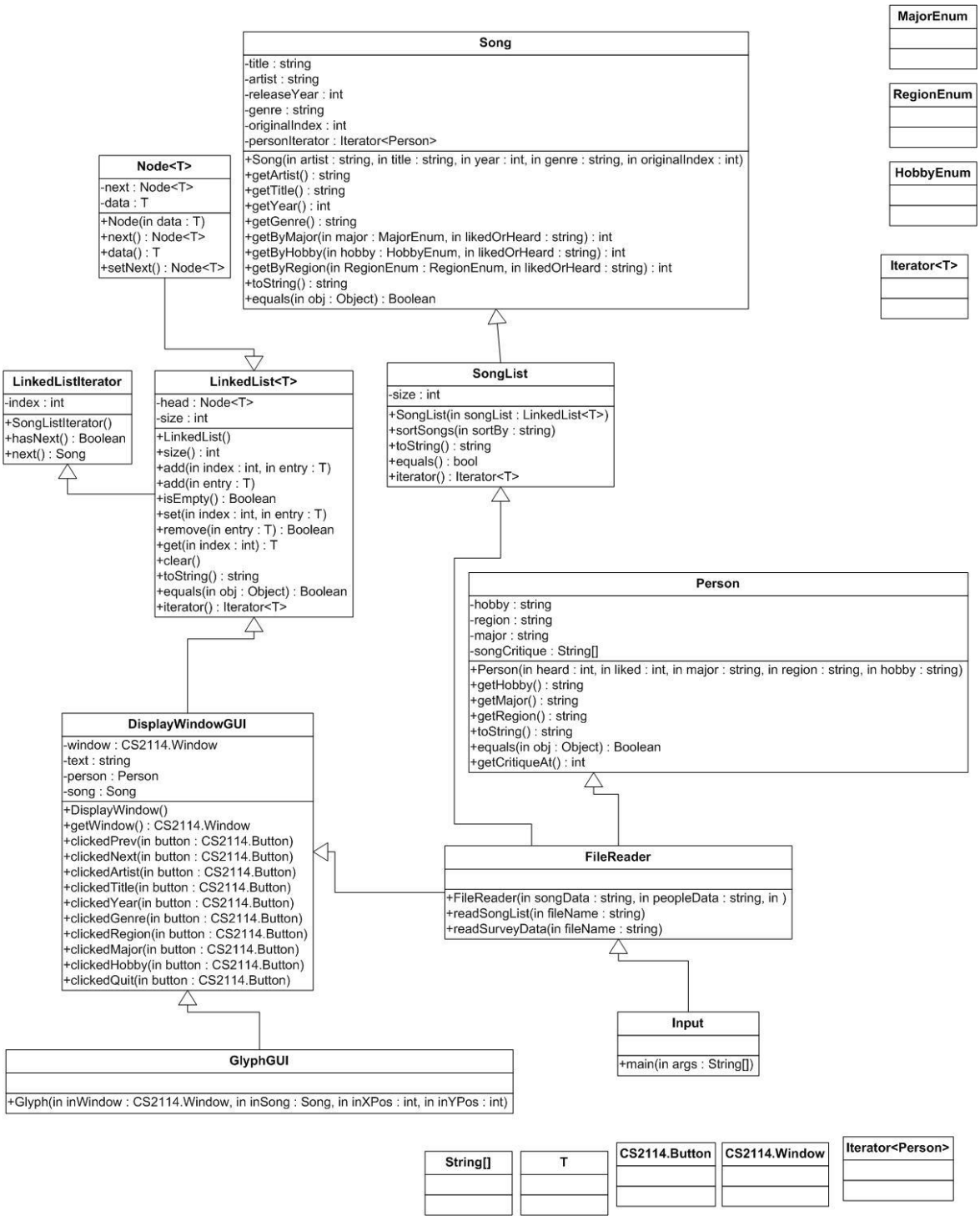
Sean McHugh (seanfmch)

Matt Fishman (feesh96)

Mostafa Elemary (melemary)

Group 6 - UML Version 2



ProjectRunner.java

Instantiates new FileReader depending on the program arguments (song file name and person file name, respectively). By default, instantiates FileReader with “SongList.txt” and “MusicSurvey.txt”.

FileReader.java

Reads both file inputs and creates a new linked list of songs by taking the MusicSurvey data and SongList data to create the songs to put in the linked list. Uses the linked list of songs to create a new SongList and passes the SongList as an argument to the window, instantiating the window.

Person.java

Holds each person’s major, region, and hobby, along with an array of their song critiques, which is used to calculate the amount listened/liked for each song. Has getters for each field and a getter for a critique at a specified index in the critique array.

Song.java

Holds each song’s title, genre, artist, release year, original index in the list of songs (which correlates to every Person’s critique array), and an iterator of type Person. It uses the iterator to iterate over the linked list of people to calculate the amount liked/listened by a given quality of person (region, hobby, major).

SongList.java

It is given a list of songs from the FileReader and has multiple versions of sorts, based on different elements of the songs. These are useful for when the window buttons are pressed. It sorts the list of songs depending on a given “sortCriteria”, such as “artist” or “genre” using an insertion sort.

DisplayWindowGUI.java

Instantiated by FileReader with a list of Songs. It can loop through the SongList to create the Glyphs – one song for each Glyph.

GlyphGUI.java

Takes a Song and coordinates to be placed. The coordinates are the top left of the Glyph. Glyph uses the Song's originalIndex field and the Song's Iterator<Person> field to calculate how many likes/heards there are, and adjusts bar sizes accordingly.

CHANGES FROM VERSION 1

In Version 1, we kept track of all likes/heards using an extensive amount of fields for each Song. Now, we added an Iterator<Person> field that can iterate through all the survey data, and we added an originalIndex field that marks where the Song's likes/heards can be found in all the data. These two fields can be used to calculate likes/heards. This calculation is done in the newly added Glyph class.