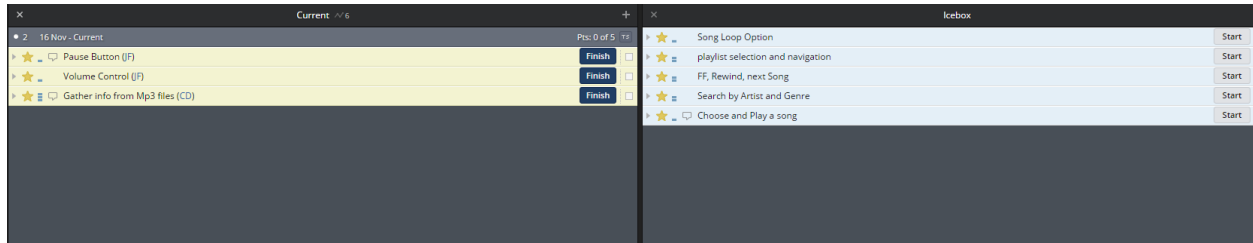


## Pivotal Tracker Progress:



## GitHub / SourceTree branches and merges:

Graph	Description
	Small Method Changes
	searchForFilesAndGetInfo method added
	in middle of genericfying the code
	play/pause and volume controls do something now
	First Iteration Deliverables
	made it so it is impossible to add repeat entries into the database updated gitignore to ignore sqllite databases
	misc
	added a Interface for media types, created database handler and can add things to the database.
	added globals file, fixed search boxes to ask for the correct thing
	Merge branch 'master' of https://github.com/jfelts1/SoftwareEngineering
	wat
	added resharper setting export
	Merge branch 'master' of https://github.com/jfelts1/SoftwareEngineering

Note: The pause button and volume control are in place and working, but we can't really use a screenshot to capture it.

## Search for media files code above get info from files:

```

61
62 private IList<IMediaEntry> searchForFilesAndGetInfo()
63 {
64     //Jess' code starts here
65     string[] mp3Files = System.IO.Directory.GetFiles(System.IO.Directory.GetCurrentDirectory(), "*.mp3");
66     string[] aviFiles = System.IO.Directory.GetFiles(System.IO.Directory.GetCurrentDirectory(), "*.avi");
67     Console.WriteLine("Current Working Directory: " + System.IO.Directory.GetCurrentDirectory());
68     ArrayList files = new ArrayList(mp3Files.Length + aviFiles.Length);
69     if (mp3Files.Length == 0) Console.WriteLine("Error: No_MP3_Doge"); // (✓ "0") ✓ ^ +
70     else
71     {
72         foreach (string t in mp3Files)
73         {
74             Console.WriteLine(t);
75             files.Add(t);
76         }
77     }
78     if (aviFiles.Length == 0) Console.WriteLine("Error: No_AVI_Doge"); // (✓ "0") ✓ ^ +
79     else
80     {
81         foreach (string t in aviFiles)
82         {
83             Console.WriteLine(t);
84             files.Add(t);
85         }
86     }
87     Console.WriteLine("ArrayList: ");
88     foreach (object curLine in files)
89     {
90         Console.WriteLine(curLine);
91     }
92     Console.ReadLine();
93
94

```

## Get info from files code:

```
98     string filePath = "";
99     IList<IMediaEntry> mediaEntries = new List<IMediaEntry>();
100
101     if(files.Count == 0) Console.WriteLine("Error: No_Files");
102     else
103     {
104         foreach (object filePathObject in files)
105         {
106             filePath = filePathObject.ToString();
107
108             using (FileStream fs = File.OpenRead(filePath))
109             {
110                 //Meaning that tags have been added to the file, ie artist, genre, etc
111                 if (fs.Length >= 128)
112                 {
113                     MusicID3Tag tag = new MusicID3Tag();
114                     fs.Seek(-128, SeekOrigin.End);
115                     fs.Read(tag.TAGID, 0, tag.TAGID.Length);
116                     fs.Read(tag.Title, 0, tag.Title.Length);
117                     fs.Read(tag.Artist, 0, tag.Artist.Length);
118                     fs.Read(tag.Album, 0, tag.Album.Length);
119                     fs.Read(tag.Year, 0, tag.Year.Length);
120                     fs.Read(tag.Comment, 0, tag.Comment.Length);
121                     fs.Read(tag.Genre, 0, tag.Genre.Length);
122                     string theTAGID = Encoding.Default.GetString(tag.TAGID);
123
124                     if (theTAGID.Equals("TAG"))
125                     {
126                         string mediaTitle = Encoding.Default.GetString(tag.Title);
127                         string mediaArtist = Encoding.Default.GetString(tag.Artist);
128                         string mediaAlbum = Encoding.Default.GetString(tag.Album);
129                         string mediaYear = Encoding.Default.GetString(tag.Year);
130                         string mediaComment = Encoding.Default.GetString(tag.Comment);
131                         string mediaGenre = Encoding.Default.GetString(tag.Genre);
132                         //long Length = Encoding.Default.GetString(tag.)
133
134                         Console.WriteLine();
135                         Console.WriteLine("Title: " + mediaTitle);
136                         Console.WriteLine("Artist: " + mediaArtist);
137                         Console.WriteLine("Album: " + mediaAlbum);
138                         Console.WriteLine("Year: " + mediaYear);
139                         Console.WriteLine("Comment: " + mediaComment);
140                         Console.WriteLine("Genre: " + mediaGenre);
141                         Console.WriteLine();
142
143                         // MusicEntry(string genre, string title, long length, string artist, string filePath)
144                         MusicEntry m = new MusicEntry(mediaGenre, mediaTitle, 0, mediaArtist, filePath);
145                         mediaEntries.Add(m);
146                     }
147                 }
148             }
149         }
150     }
151
152     return mediaEntries;
153 }
154 }
```