CIS 111B FitList

May 8, 2022

Team Members: Chris Bonner, Geoffrey Gaines, Julia Hansford

FitList is a program that lets a user decide what type of workout they would like to do and then choose what exercise(s) they want to do for the day. We chose to work with GUIs because they seemed as like they would be a challenge and they are used in large companies today. We started off this project deciding how we wanted this program to work, Julia started on the GUI, Chris started on the API, and Geoffrey started on the code. We started off working on the JSON.

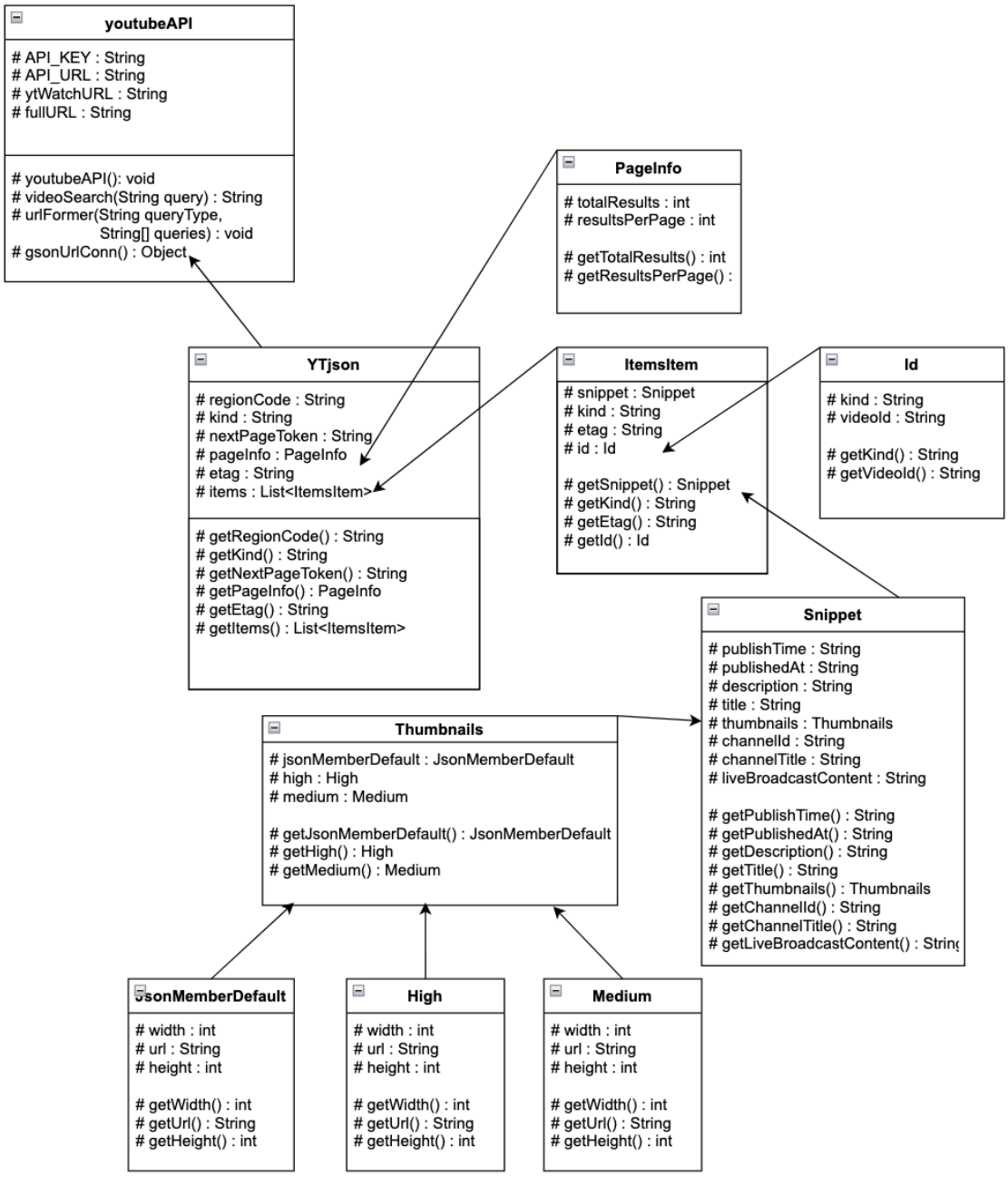
Creating a GUI allowed us to practice and work with different aspects we were unfamiliar with such as JFrames, JLabels, JButtons, JComoBox, JCheckBox and JTextAreas. These allowed us two really make everything visual for the program. The JFrame allowed us to make a structure that would be able to hold all the other aspects that makes the program work. The JLabels allowed us to place text and give the user directions on what they should do in the window. The JComoBox allowed us to make a dropdown box that would give a list of options for the user to choose from, and the JButton when clicked would allow us to open a different portion of the program. The JCheckBoxes allowed us to check which exercise(s) the user wanted to do that day and then the JTextArea displayed the URL for the video corresponding to the exercise.

All activities in the GUI create the frames, labels, buttons, checkboxes, dropdown box, text area, and the driver are run in it. Once the GUI is running the HomeFrame opens that holds a dropdown box with different workouts in it. The user selects either cardio, arms, legs, push or pull then hits next. A new frame opens with 5 checkboxes relating to the previous selection. The user selects which exercise(s) they would like to do and clicks the search button that returns a video for each exercise selected.

As for the rubric, FitList uses different internal Java APIs java.awt.event.ActionEvent, java.awt.event.ActionListener, java.util.ArrayList, and javax.swing.*. FitList also uses external APIs like a GSON and JSON to help connect the program to the internet. Our API was written by our database lead Chris, which was then tested out by our programming lead Geoffrey. Geoffrey made sure everything could flow together from the API into the GUI the Julia made.

As a group we had to make timing decisions on things that we had to cut out from the program because we did not initially think it would be as difficult as it was. If we had more time, we would have tried to work with Android Studios and make the program an actual app. We would also have the app make a personalized schedule on days to work out and when the user would need a rest day, but unfortunately, we did not have the time to put the app in motion.

FitList UML


youtubeAPI
API_KEY : String
API_URL : String
ytWatchURL : String
fullURL : String
youtubeAPI(): void
videoSearch(String query) : String
urlFormer(String queryType, String[] queries) : void
gsonUrlConn() : Object
PageInfo
totalResults : int
resultsPerPage : int
getTotalResults() : int
getResultsPerPage() :
YTjson
regionCode : String
kind : String
nextPageToken : String
pageInfo : PageInfo
etag : String
items : List<ItemsItem>
getRegionCode() : String
getKind() : String
getNextPageToken() : String
getPageInfo() : PageInfo
getEtag() : String
getItems() : List<ItemsItem>
ItemsItem
snippet : Snippet
kind : String
etag : String
id : Id
getSnippet() : Snippet
getKind() : String
getEtag() : String
getId() : Id
Id
kind : String
videoId : String
getKind() : String
getVideoId() : String
Snippet
publishTime : String
publishedAt : String
description : String
title : String
thumbnails : Thumbnails
channelId : String
channelTitle : String
liveBroadcastContent : String
getPublishTime() : String
getPublishedAt() : String
getDescription() : String
getTitle() : String
getThumbnails() : Thumbnails
getChannelId() : String
getChannelTitle() : String
getLiveBroadcastContent() : Strin
Thumbnails
jsonMemberDefault : JsonMemberDefault
high : High
medium : Medium
getJsonMemberDefault() : JsonMemberDefault
getHigh() : High
getMedium() : Medium
JsonMemberDefault
width : int
url : String
height : int
getWidth() : int
getUrl() : String
getHeight() : int
High
width : int
url : String
height : int
getWidth() : int
getUrl() : String
getHeight() : int
Medium
width : int
url : String
height : int
getWidth() : int
getUrl() : String
getHeight() : int