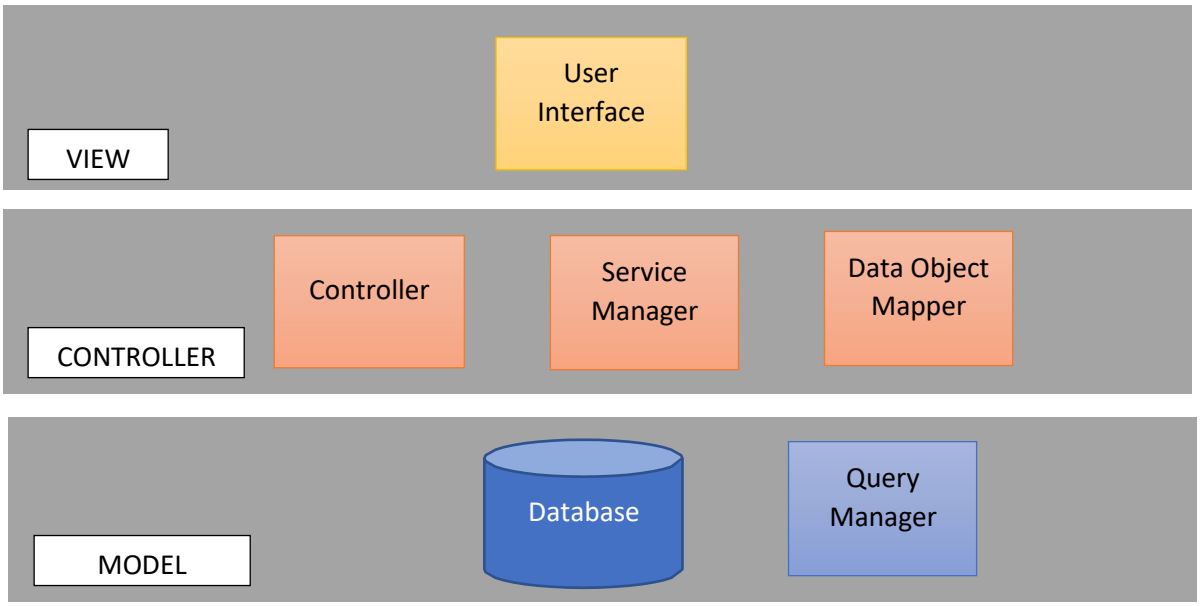
Contents

Team Details:
Sindoora S Rao – 40199155
Sourav Ganesha – 40228888

GitHub : https://github.com/SouravGanesh/APP_Project_2022/

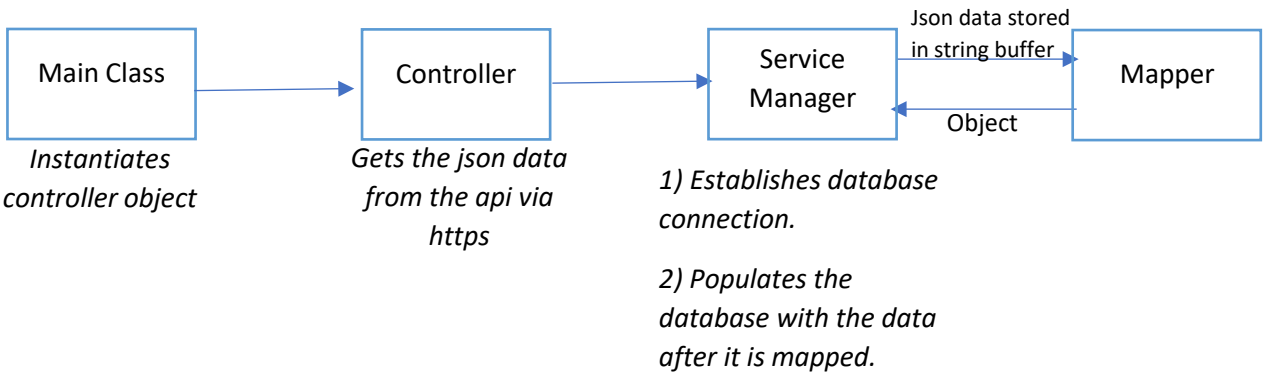
Architectural Overview



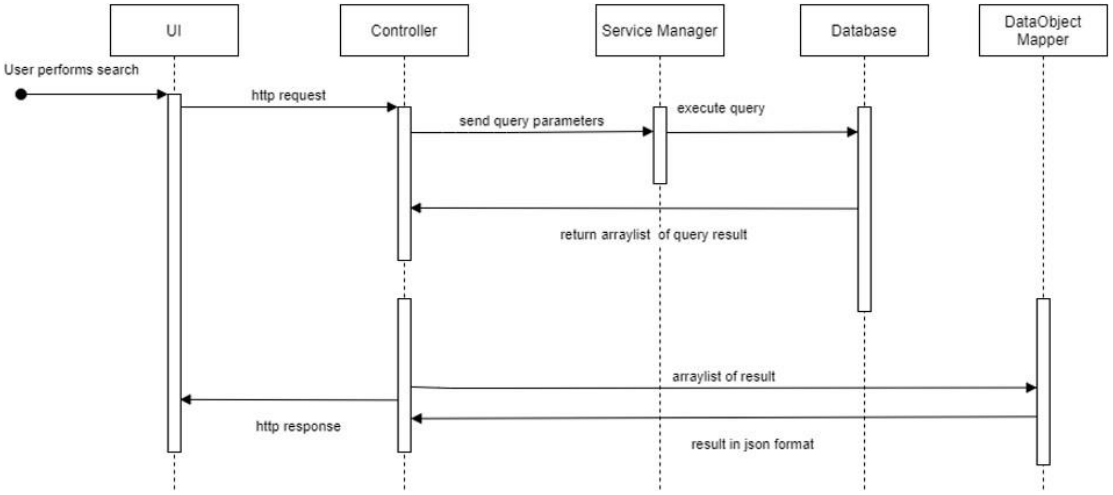
Flow of control

Application Initialization

- On execution of main, invocation happens in the following sequence and the dataset gets populated in the database.

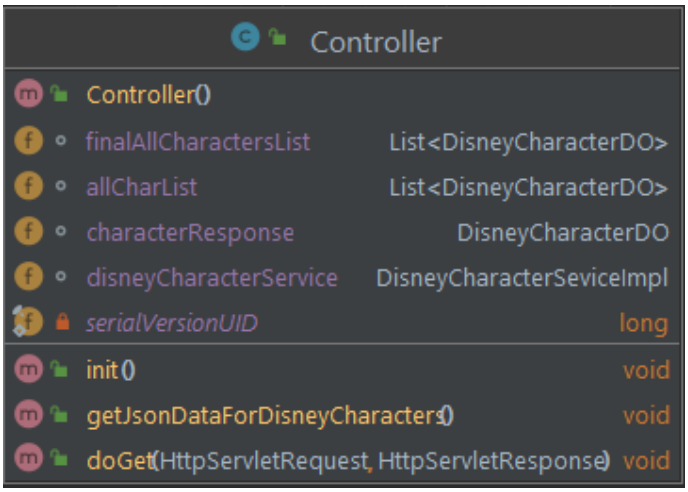


Execution of a user action



Overview on entities and its methods

Controller class



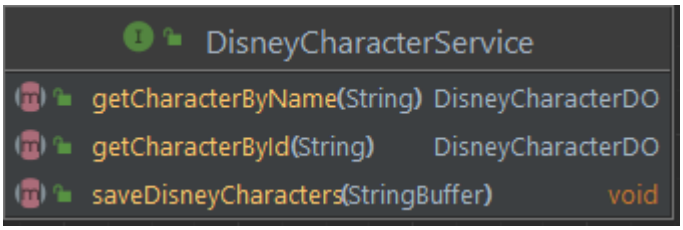
1. *Init ()*
   - called when servlet is first created. It is called only once during the first creation and not called for each request from user.
   - Init method is responsible for starting the server and makes a call to *getJsonDataForDisneyCharacter ()* internally.

2. *getJsonDataForDisneyCharacter ()*
   - responsible initiating a http URL connection to an API having the dataset of Disney characters.
   - The json data and received and passed on to the service class, *disneyCharacterService*

3. *doGet (HttpServeletRequest request, HttpServeletResponse response)*

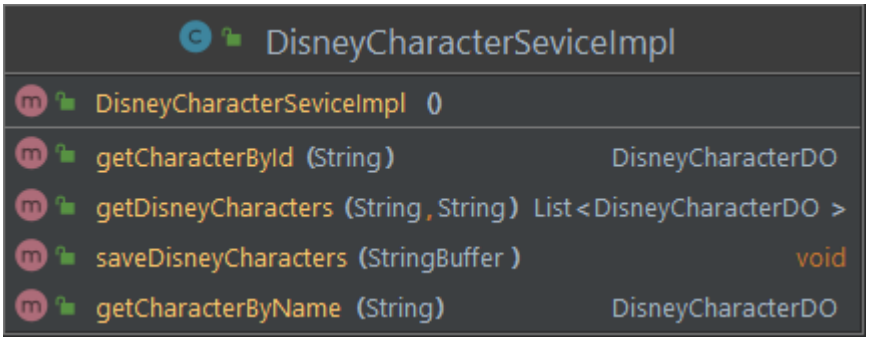
   - Handles the user request and is responsible for sending back the response to a certain request.
   - It reads the user input set via java script file and passes these values to the *disneyCharacterService*
   - It is also responsible for converting the result received from the service class to json format and print it on to the UI.

Service Interface



   - Provides the declaration for Service methods to save the data into database and fetch the data based on user input.

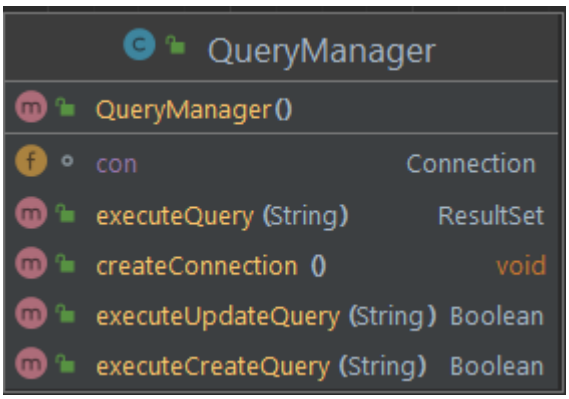
Service Class



1. *saveDisneyCharacter (String)*
   - gets called by *getJsonDataForDisneyCharacter ().*
   - saves all the data that is received by a json API after it is been mapped.

2. *getDineyCharacters (), getCharacterByName (), getCharacterById ()*
   - *frames a string of query statement based on user input and sends it to the query manager for execution*

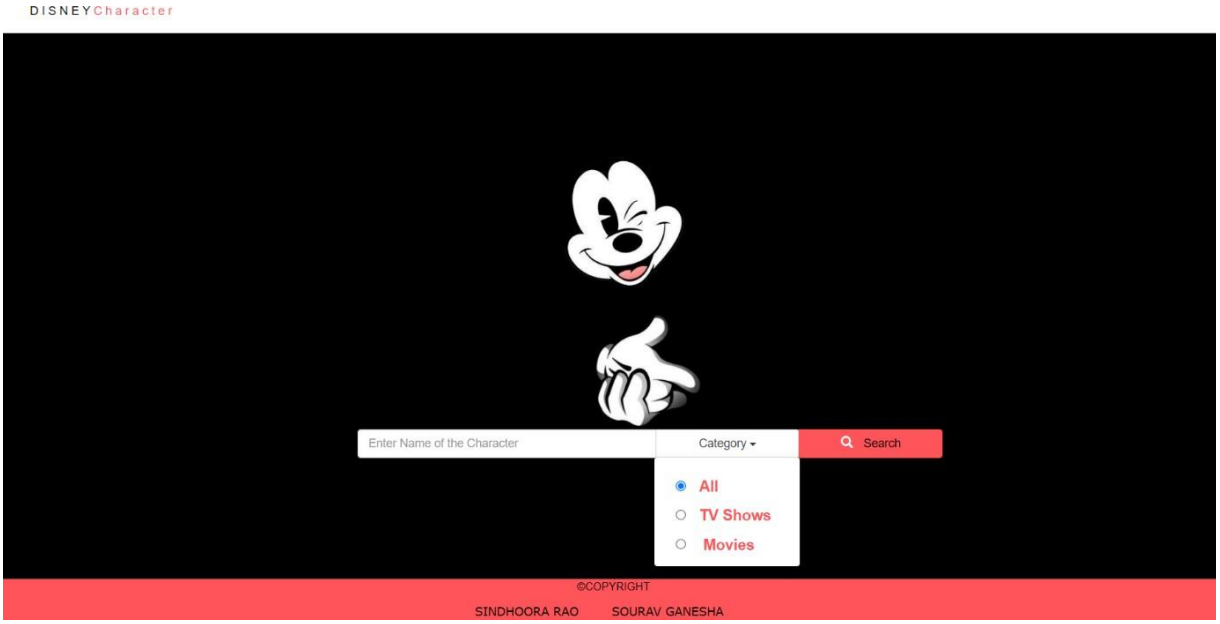
Query Manager



1. *createConnection ()*
   - *is responsible to establish jdbc connection to mysql.*
2. *executeUpdateQuery (String), executeQuery (String)*
   - *executes query on the tables in connected database and returns the result of execution.*

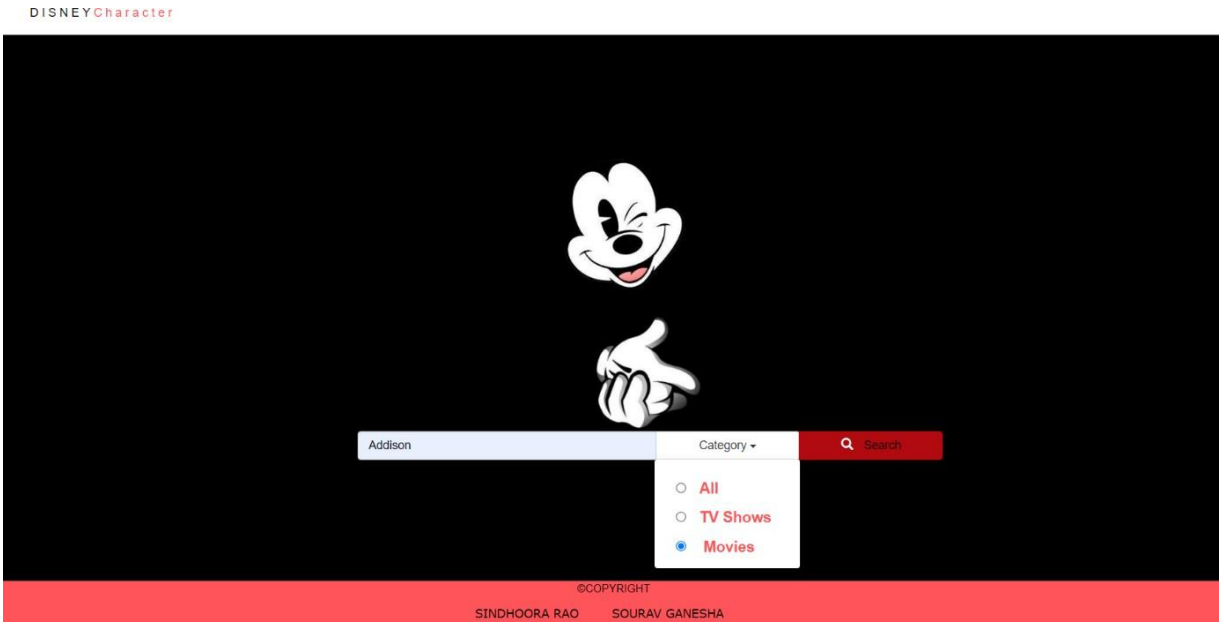
User Interface

Home Page



Result of search action from user

-performing search



-result of the search

Testing

-We have used Junit Test framework for testing our application. The following screenshots show the detailed coverage result from test execution,