IMDB CLON (MovieMenia)

**Objective:**

* Develop a movie review and rating application in the web with all the functionality of social media interaction
* Use Non personalized recommendation for visitors and personalized recommendation for users
* Provide users to follow other users in the domain of the application with instant messaging feature and notification system for better communication

**Target Technology:**

* ReactJs/NextJs framework for frontend
* Java Spring Boot for sessionless/stateless backend api
* Microservice Architecture with spring cloud integration
* Docker as virtual environment and app container
* AWS for cloud deployment

**Consumer Classification of Application:**

There will be total 3 types of consumer in the application

**Visitor:** as a visitor one can access the application to see the ratings and reviews of all the movies and also visit the profiles of other users in the application.

**User:** along with all the priviledges of a visitor a user can create his/her own profile interact with other user rate and review any movie

**Admin:** with all the priviledges of a user an admin can upload new movies with all the necessary attributes (e.g. title, length, poster, description).

**Frontend Views:**

The following list is about all the necessery Pages (UI Views) of the frontend application.

**Sign-up:**

To allow visitors to create a new account in the domain of the application.

**Sign-in:**

To allow users to sign-in with the credentials they provided.

**Home:**

Root page of the application provides segregated view of the movies available

(e.g. trending, top rated etc.).

**Movie:**

To provide a broad and descriptive view of a particular movie with all its available information , ratings and reviews.

**WatchList:**

To show a stacked views of all the movies a user have marked to watch or already watched.

**Trailer:**

Shows the trailer of the movie in a broader view.

**Profile:**

Shows the Account of a user with all the information of the user (e.g. name, profile picture, movies rated ) with an option to follow him/her.

**MyProfile:**

Shows the View where one can access all of his/her information and customize them .

**Search:**

To provide a stacked view of the search result .

**Upload:**

A convenient view for uploading new movies in the application

**Consumer/View access table:**

Following table classifies the users with respect to their access level in the application.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Visitor | User | Admin |
| Sign-up | Yes | Yes | Yes |
| Sign-in | Yes | Yes | Yes |
| Home | Yes  (without personalized recommendation) | Yes | Yes |
| Trailer | Yes | Yes | Yes |
| Movie | Yes  (without rate or review priviledges) | Yes | Yes |
| Watchlist | No | Yes | Yes |
| Upload | No | No | Yes |
| Profile | Yes | Yes | Yes |
| MyProflie | No | Yes | Yes |
| Search | Yes | Yes | Yes |

**API DOCUMENTATION:**

Following is the list of all the api with url format, reqest format and structure and response format and structure.

**Related Document Structure:**

**Movie 🡪**

**{**

**“movie\_id” : “<movie id>”,**

**“movie\_title” : “<movie\_title>”,**

**“movie\_average\_rating” : “<average rating>”,**

**“movie\_poster” : “<link of the poster>”,**

**“rated\_by\_user” : “<rating that the use gave>”,**

**“total\_ratings” : “<total number of ratings>”,**

**“genres” : “<movie genres>”**

**}**

**Review 🡪**

**{**

**“review\_id” : “<review id>”,**

**“reviewer\_name” : “<reviewer name>”,**

**“reviewer\_profile\_picture” : “<profile picture link>”,**

**“content” : “<content of the review>”,**

**“no\_of\_reacts” : “<no of reacts>”,**

**“user react status” : “<wheather user reacted>”**

**}**

**Reply 🡪**

**{**

**“reply\_id” : “<reply id>”,**

**“replier\_name” : “<replier name>”,**

**“replier\_profile\_picture” : “<profile picture link>”,**

**“content” : “<content of the reply>”,**

**“no\_of\_reacts” : “<no of reacts>”,**

**“user react status” : “<wheather user reacted>”**

**}**

**Msg 🡪**

**{**

**“msg\_id” : “<msg id value>”**

**“type” : “<type of msg (e.g. authentication/validation)>”,**

**“msg” : “<msg based on type>”**

**}**

**MovieShort 🡪**

**{**

**“movie\_id” : “<movie id>”,**

**“movie\_title” : “<movie title>”,**

**“movie\_poster” : “<movie poster link>”**

**}**

**UserShort 🡪**

**{**

**“username” : “<name of the user>”,**

**“user\_profile\_pic” : “<link of profile picture>”**

**}**

**Error 🡪**

**{**

**“code” : “<integer error code for error classification>”,**

**“msg” : “<default msg associated to the request>”**

**“reason” : “a key value pair map for error field and error value Iteration”**

**}**

**Signup Endpoint:**

**URL:**

**/auth/signup**

**METHOD: POST**

**QUERY PARAMETER: NONE**

**HEADER: NONE**

**REQUEST:**

**{**

**“userId” : “<name of the user>”, (length: 10 to 30)**

**“password” : “<password of the user>”, (length: 10 to 20) “emailId” : “<email address value>” (email validation: true)**

**}**

**RESPONSE:**

**ONSUCCESS 🡪**

**{**

**“userId” : “<name of the user>”,**

**“token” : “<authentication token>”**

**}**

**ONERROR 🡪**

**{**

**“code” : “VALIDATION\_FAILED\_CODE”**

**“msg” : “SIGN\_UP\_FAILED\_MSG”,**

**“reason” : {**

**“field1” : “msg1”,**

**“field2” : “msg2”**

**}**

**}**

**------------------------------------------------------------------------------------------**

**{**

**“code” : “DATA\_LAYER\_ERROR”**

**“msg” : “SIGN\_UP\_FAILED\_MSG”,**

**“reason” : {}**

**}**

**STATUS CODE:**

**ONSUCCESS 🡪 CREATED(201)**

**ONERROE 🡪 BAD\_REQUEST(400), INERNAL\_SERVER\_ERROR(500), REQUEST\_TIMEOUT(504)**

**Signin Endpoint:**

**URL:**

**/auth/signin**

**REQUEST:**

**{**

**“userId” : “<username of the user>”, (length: 10 to 30)**

**“password” : “<password of the user>” (length: 10 to 20)**

**}**

**HEADER: NONE**

**METHOD: POST**

**QUERY PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**{**

**“userId” : “<username of the user>”,**

**“token” : “<authentication token>”**

**}**

**ONERROR 🡪**

**{**

**“code” : “VALIDATION\_FAILED\_CODE”**

**“msg” : “SIGN\_IN\_FAILED\_MSG”,**

**“reason” : {**

**“field1” : “msg1”,**

**“field2” : “msg2”**

**}**

**}**

**------------------------------------------------------------------------------------------**

**{**

**“code” : “DATA\_LAYER\_ERROR”**

**“msg” : “SIGN\_IN\_FAILED\_MSG”,**

**“reason” : {}**

**}**

**{**

**“code” : “CREDENTIAL\_NOT\_MATCH\_ERROR\_CODE”**

**“msg” : “SIGN\_IN\_FAILED\_MSG”,**

**“reason” : {**

**“userId” : “id value”**

**}**

**}**

**STATUS CODE:**

**ONSUCCESS 🡪 OK(200)**

**ONERROE 🡪 BAD\_REQUEST(400), INTERNAL\_SERVER\_ERROR(500), REQUEST\_TIME\_OUT(504)**

**Home Endpoint:**

**URL:**

**/movies/all/user\_id**

**REQUEST: NONE**

**HEADER : NONE**

**METHOD: GET**

**QUERY PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**{**

**“user\_info”: {**

**“username” : “<username of the user>”,**

**“profile\_picture\_url” : “<profile picture url>”**

**},**

**“recent” : {**

**“page\_id” : “<page id value>”,**

**“page\_size”: “<page size>”,**

**“no\_of \_pages” : “<number of pages>”**

**“list” : “<list of movie object>”**

**},**

**“toprated”: {**

**“page\_id” : “<page id value>”,**

**“page\_size”: “<page size>”,**

**“no\_of \_pages” : “<number of pages>”**

**“list” : “<list of movie object>”**

**},**

**“rate\_more” : {**

**“page\_id” : “<page id value>”,**

**“page\_size”: “<page size>”,**

**“no\_of \_pages” : “<number of pages>”**

**“list” : “<list of movie object>”**

**},**

**“suggested” : {**

**“page\_id” : “<page id value>”,**

**“page\_size”: “<page size>”,**

**“no\_of \_pages” : “<number of pages>”**

**“list” : “<list of movie object>”**

**}**

**}**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪 UNAUTHORIZED**

**Recent Released Movies Endpoint:**

**URL:**

**/movies/recent/page\_id**

**REQUEST: NONE**

**HEADER: NONE**

**METHOD: GET**

**QUERY PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS🡪**

**{**

**“page\_id” : “<page id value>”,**

**“page\_size”: “<page size>”,**

**“no\_of\_pages” : “<number of pages>”**

**“list” : “<list of movie object>”**

**}**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪 UNAUTHORIZED**

**Top rated Movies Endpoint:**

**URL:**

**/movies/toprated/page\_id**

**REQUEST: NONE**

**HEADER: NONE**

**METHOD: GET**

**QUERY PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS🡪**

**{**

**“page\_id” : “<page id value>”,**

**“page\_size”: “<page size>”,**

**“no\_of\_pages” : “<number of pages>”**

**“list” : “<list of movie object>”**

**}**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪 UNAUTHORIZED**

**Rate More Movies Endpoint:**

**URL:**

**/movies/ratemore/page\_id**

**REQUEST: NONE**

**HEADER: NONE**

**METHOD: GET**

**QUERY PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS🡪**

**{**

**“page\_id” : “<page id value>”,**

**“page\_size”: “<page size>”,**

**“no\_of\_pages” : “<number of pages>”,**

**“list” : “<list of movie object>”**

**}**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪 UNAUTHORIZED**

**Suggested Movies Endpoint:**

**URL:**

**/movies/suggested/user\_id/page\_id**

**REQUEST: NONE**

**HEADER: u\_id, token**

**METHOD: GET**

**QUERY PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS🡪**

**{**

**“page\_id” : “<page id value>”,**

**“page\_size”: “<page size>”,**

**“no\_of\_pages” : “<number of pages>”,**

**“list” : “<list of movie object>”**

**}**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪 UNAUTHORIZED**

**Search Movies Endpoint:**

**URL:**

**/movies/search/page\_id**

**REQUEST: NONE**

**HEADERS: NONE**

**METHOD: GET**

**QUERY PARAMETER: search\_query**

**RESPONSE:**

**ONSUCCESS:**

**{**

**“query\_string” : “<query string value>”,**

**“page\_id” : “<pages number>”,**

**“page\_size”: “<page size>”,**

**“page\_no” : “<number of pages>”,**

**“list” : “<list of movie object>”**

**}**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪 UNAUTHORIZED**

**Wishlist Movies Endpoint:**

**URL:**

**movie/wishlist/user\_id/page\_id**

**REQUEST: NONE**

**HEADER: u\_id, token**

**METHOD: GET**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**{**

**“page\_id” : “<page no value>”,**

**“page\_size”: “<page size>”,**

**“page\_no” : “<number of pages>”,**

**“list” : “<list of movie object>”**

**}**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪 UNAUTHORIZED**

**Watched Movies Endpoint:**

**URL:**

**movie/watched/user\_id/page\_id**

**REQUEST: NONE**

**HEADER: u\_id, token (compulsory)**

**METHOD: GET**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**{**

**“page\_id” : “<page no value>”,**

**“page\_size”: “<page size>”,**

**“page\_no” : “<number of pages>”,**

**“list” : “<list of movie object>”**

**}**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪 UNAUTHORIZED**

**View Other User Profile Endpoint:**

**URL:**

**users/profile/user\_id**

**REQUEST:**

**NONE**

**HEADER: NONE**

**METHOD: GET**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**{**

**“user\_person\_info”: {**

**“user\_id” : “<user id value>”,**

**“profile\_picture” : “<profile picture link>”,**

**“no\_of\_movies\_rated” : “<number of movie rated>”,**

**“no\_of\_movie\_reviewed” : “<no of movie reviewed>”,**

**“no\_of\_follower:” : “<number of followers>”,**

**“my\_following” : “<does he/she follow>”**

**},**

**“movie\_rated\_list” : “<list of movie objects>”,**

**“movie\_watch\_list” : “<list of movie objects>”,**

**“movie\_wish\_list” : “<list of movie objects>”**

**}**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪**

**UNAUTHORIZED (for token mismatch)**

**BAD\_REQUEST (for invalid userid)**

**View My Profile Endpoint:**

**URL:**

**users/account/userid**

**REQUEST: NONE**

**HEADER: u\_id, token**

**METHOD: GET**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**{**

**“user\_person\_info”: {**

**“user\_id” : “<user id value>”,**

**“profile\_picture” : “<profile picture link>”,**

**“no\_of\_movies\_rated” : “<number of movie rated>”,**

**“no\_of\_movie\_reviewed” : “<no of movie reviewed>”,**

**“no\_of\_follower:” : “<number of followers>”,**

**},**

**“movie\_rated\_list” : “<list of movie objects>”,**

**“movie\_watch\_list” : “<list of movie objects>”,**

**“movie\_wish\_list” : “<list of movie objects>**

**}**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪**

**UNAUTHORIZED (for invalid token)**

**BAD\_REQUEST (for userid invalid)**

**View Movie Endpoint:**

**URL:**

**movies/info/movie\_id**

**REQUEST: NONE**

**METHOD: GET**

**QUERY\_PARAMETER: NONE**

**HEADER: NONE**

**RESPONSE:**

**{**

**“movie\_id” : “<movie id valuie>”,**

**“movie\_title” : “<movie title>”,**

**“movie\_poster” : “<movie poster url>”,**

**“movie\_length” : “<length of the movie>”,**

**“average\_rating” : “<average rating of the movie>”,**

**“total ratings” : “<total no of ratings by the user>”,**

**“movie\_genre” : “comma separated list of genre”,**

**“movie\_trailer\_link” : “link of the movie trailer”,**

**“no\_of\_reviews” : “total no of reviews”,**

**“desc” : “description of movie”**

**“reviews”: {**

**“movie\_id” : “<movid id value>”**

**“page\_id” : “<page id value>”,**

**“page\_size” : “<page size>”,**

**“no\_of\_pages” : “<page no>”,**

**”list” : “<list of review objects>”**

**}**

**}**

**ONERROR 🡪**

**Msg Error Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪**

**UNAUTHORIZED (for invalid token)**

**BAD\_REQUEST (for movie id invalid)**

**View Movie Review Endpoint:**

**URL:**

**movies/reviews/movie\_id/page\_id**

**REQUEST: NONE**

**HEADER: NONE**

**METHOD: GET**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**{**

**“movie\_id” : “<movie id value>”,**

**“page\_id” : “<page id value>”,**

**“no\_of\_pages” : “<no of pages>”,**

**“page\_size” : “<page size>”,**

**“list”: “<list of review objects>”**

**}**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪**

**UNAUTHORIZED (for invalid token)**

**BAD\_REQUEST (for movieid invalid)**

**View Movie Review Reply Endpoint:**

**URL:**

**movies/review/reply/movie\_id/review\_id**

**REQUEST: NONE**

**HEADER: NONE**

**METHOD: GET**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**{**

**“movie\_id” : “<movie id value>”,**

**“review\_id” : “<review id value>”,**

**“page\_id” : “<page id value>”,**

**“page\_size” : “<page size>”,**

**“no\_of\_pages” : “<no of pages>”,**

**“list”: “list of reply objects”**

**},**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 OK**

**ONERROR 🡪**

**UNAUTHORIZED (for invalid token)**

**BAD\_REQUEST (for movieid/reviewid invalid)**

**Post Movie Endpoint:**

**URL:**

**movies/upload/**

**REQUEST:**

**{**

**“movie\_title” : “<movie title>”, (length: 5 to 200)**

**“movie\_poster” : “<movie poster img>”, (max-size: 4MB)**

**“movie\_length” : “<length of the movie>”, (type: int (30-180))**

**“movie\_genre” : “comma separated of genre”, (min-length:3)**

**“movie\_trailer” : “movie trailer video”, (max-size: 100MB)**

**“movie\_desc” : “movie description” (length 10 to 2000)**

**}**

**HEADER: u\_id, token**

**METHOD: POST**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**movie\_object**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪 CREATED**

**ONERROR 🡪 UNAUTHORIZED**

**Post Movie Rating Endpoint:**

**URL:**

**movies/rating/user\_id/movie\_id/**

**REQUEST:**

**{**

**“rating” : “<rating in scale>” (type: int (0-5))**

**}**

**HEADER: u\_id, token**

**METHOD: POST**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**Success Msg Object**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪**

**OK**

**ONERROR 🡪**

**UNAUTHORIZED (invalid token)**

**BAD\_REQUEST (invalid movieid)**

**Post Movie Review Endpoint:**

**URL:**

**movies/review/user\_id/movie\_id/**

**REQUEST:**

**{**

**“content” : “<content data>” (min-length:2)**

**}**

**HEADER: u\_id, token**

**METHOD: POST**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**Success Msg Object**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪**

**OK**

**ONERROR 🡪**

**UNAUTHORIZED**

**Post Movie Review Reply Endpoint:**

**URL:**

**movies/reviews/replys/user\_id/movie\_id/review\_id/**

**REQUEST:**

**{**

**“content” : “<content data>” (min-length: 2)**

**}**

**HEADER: u\_id, token**

**METHOD: POST**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**Success Msg Object**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪**

**OK**

**ONERROR 🡪**

**UNAUTHORIZED (invalid token)**

**BAD\_REQUEST (movieid/reviewid invalid)**

**Post Movie Review React Endpoint:**

**URL:**

**movies/review/react/user\_id/movie\_id/review\_id/**

**REQUEST:**

**BODY: NONE**

**HEADER: u\_id, token**

**METHOD: POST**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**Success Msg Object**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪**

**OK**

**ONERROR 🡪**

**UNAUTHORIZED (invalid token)**

**BAD\_REQUEST (ivalid movieid/reviewid)**

**Post Movie Review Reply react Endpoint:**

**URL:**

**movies/review/reply/react/user\_id/movie\_id/review\_id/reply\_id/**

**REQUEST:**

**BODY: NONE**

**METHOD: POST**

**HEADER: u\_id, token**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**ONSUCCESS 🡪**

**Success Msg Object**

**ONERROR 🡪**

**Error Msg Object**

**STATUS CODE:**

**ONSUCCESS 🡪**

**OK**

**ONERROR 🡪**

**UNAUTHORIZED (invalid token)**

**BAD\_REQUEST (invalid movieid/reviewid/replyid)**

**Follow/Unfollow User Endpoint**

**URL:**

**user/followrequest**

**REQUEST:**

**BODY: NONE**

**HEADER: u\_id, token**

**METHOD: POST**

**QUERY\_PARAMETER: target\_user\_id**

**RESPONSE:**

**ONSUCCESS 🡪**

**{**

**“followerId” : <userId of the user who hit the endpoint>,**

**“followingId” : <userId of the user who got followed or unfollowed>,**

**“noOfFollowers” : “<no of updated followers of the target user>”,**

**“status” : <true(followed) or false(unfollowed)>**

**}**

**ONERROR 🡪**

**{**

**“code” : “DATA\_LAYER\_ERROR”**

**“msg” : “FOLLOW\_REQUEST\_FAILED\_MSG”,**

**“reason” : {}**

**}**

**-------------------------------------------------------------------------------------------**

**{**

**“code” : “ENTITY\_NOT\_FOUND”**

**“msg” : “FOLLOW\_REQUEST\_FAILED\_MSG”,**

**“reason” : {**

**“current/target userid” : “id value”**

**}**

**}**

**STATUS CODE:**

**ONSUCCESS 🡪**

**OK**

**ONERROR 🡪**

**UNAUTHORIZED, BAD\_REQUEST, INTERNAL\_SERVER\_ERROR, REQUEST\_TIME\_OUT**

**Check UserId Exist Endpoint:**

**URL:**

**users/exists/user\_id**

**REQUEST: NONE**

**METHOD: GET**

**HEADER: NONE**

**QUERY\_PARAMETER: NONE**

**RESPONSE:**

**NONE**

**STATUS CODE:**

**NO\_CONTENT (if no such user exist)**

**OK (if user exist)**

**Get User Query Short Details Endpoint:**

**URL:**

**users/short/page\_id**

**REQUEST: NONE**

**METHOD: GET**

**HEADER : NONE**

**QUERY\_PARAMETER : search\_query**

**RESPONSE:**

**{**

**“page\_id” : “<page id>”,**

**“page\_size”: “<page size>”,**

**“no\_of\_pages” : “<no of pages>”,**

**“list”: “<list of user short objects>”**

**}**

**STATUS CODE:**

**OK**

**Get Movie Query Short Details Endpoint:**

**URL:**

**movies/short/page\_id**

**REQUEST: NONE**

**METHOD: GET**

**HEADER : NONE**

**QUERY\_PARAMETER : search\_query**

**RESPONSE:**

**{**

**“page\_id” : “<page id>”,**

**“page\_size”: “<page size>”,**

**“no\_of\_pages” : “<no of pages>”,**

**“list”: “<list of movie short objects>”**

**}**

**STATUS CODE:**

**OK**

**Add Movie To Wishlist Endpoint:**

**URL:**

**movies/wishlist/user\_id/movie\_id**

**REQUEST: NONE**

**HEADER: u\_id, token**

**QUERY PARAMETER: NONE**

**METHOD: UPDATE**

**RESPONSE:**

**ONSUCCESS 🡪 Success Msg Object**

**ONERROR 🡪 Error Msg Object**

**STATUS CODE:**

**UNAUTHORIZED (token invalid)**

**BAD\_REQUEST (userid/movieid invalid)**

**Add Movie To Watchedlist Endpoint:**

**URL:**

**movies/watchedlist/user\_id/movie\_id**

**REQUEST: NONE**

**HEADER: u\_id, token**

**QUERY PARAMETER: NONE**

**METHOD: UPDATE**

**RESPONSE:**

**ONSUCCESS 🡪 Success Msg Object**

**ONERROR 🡪 Error Msg Object**

**STATUS CODE:**

**UNAUTHORIZED (token invalid)**

**BAD\_REQUEST (userid/movieid invalid)**

**Remove Movie from Wishlist Endpoint:**

**URL:**

**movies/wishlist/user\_id/movie\_id**

**REQUEST: NONE**

**HEADER: u\_id, token**

**QUERY PARAMETER: NONE**

**METHOD: DELETE**

**RESPONSE:**

**ONSUCCESS 🡪 Success Msg Object**

**ONERROR 🡪 Error Msg Object**

**STATUS CODE:**

**UNAUTHORIZED (token invalid)**

**BAD\_REQUEST (userid/movieid invalid)**

**Remove Movie from Watchedlist Endpoint:**

**URL:**

**movies/watchedlist/user\_id/movie\_id**

**REQUEST: NONE**

**HEADER: u\_id, token**

**QUERY PARAMETER: NONE**

**METHOD: DELETE**

**RESPONSE:**

**ONSUCCESS 🡪 Success Msg Object**

**ONERROR 🡪 Error Msg Object**

**STATUS CODE:**

**UNAUTHORIZED (token invalid)**

**BAD\_REQUEST (userid/movieid invalid)**

**Remove Review of a Movie Endpoint:**

**URL:**

**movies/reviews/user\_id/movie\_id/reviewid**

**REQUEST: NONE**

**HEADER: u\_id, token**

**QUERY PARAMETER: NONE**

**METHOD: DELETE**

**RESPONSE:**

**ONSUCCESS 🡪 Success Msg Object**

**ONERROR 🡪 Error Msg Object**

**STATUS CODE:**

**UNAUTHORIZED (token invalid)**

**BAD\_REQUEST (userid/movieid/reviewid invalid)**

**Remove Reply of a Review Endpoint:**

**URL:**

**movies/reviews/replies/user\_id/movie\_id/reviewid/replyid**

**REQUEST: NONE**

**HEADER: u\_id, token**

**QUERY PARAMETER: NONE**

**METHOD: DELETE**

**RESPONSE:**

**ONSUCCESS 🡪 Success Msg Object**

**ONERROR 🡪 Error Msg Object**

**STATUS CODE:**

**UNAUTHORIZED (token invalid)**

**BAD\_REQUEST (userid/movieid/reviewid/replyid invalid)**

**Remove Movie Endpoint:**

**URL:**

**movies/movie\_id**

**REQUEST: NONE**

**HEADER: u\_id, token**

**QUERY PARAMETER: NONE**

**METHOD: DELETE**

**RESPONSE:**

**ONSUCCESS 🡪 Success Msg Object**

**ONERROR 🡪 Error Msg Object**

**STATUS CODE:**

**UNAUTHORIZED (token invalid)**

**BAD\_REQUEST (movieid invalid)**

**Remove User Endpoint:**

**URL:**

**users/user\_id**

**REQUEST: NONE**

**HEADER: u\_id, token**

**QUERY PARAMETER: NONE**

**METHOD: DELETE**

**RESPONSE:**

**ONSUCCESS 🡪 Success Msg Object**

**ONERROR 🡪 Error Msg Object**

**STATUS CODE:**

**UNAUTHORIZED (token invalid)**

**BAD\_REQUEST (movieid invalid)**

**Logout User Endpoint:**

**URL:**

**users/signout/user\_id**

**REQUEST: NONE**

**HEADER: u\_id, token**

**QUERY PARAMETER: NONE**

**METHOD: POST**

**RESPONSE:**

**ONSUCCESS 🡪 Success Msg Object**

**ONERROR 🡪 Error Msg Object**

**STATUS CODE:**

**UNAUTHORIZED (token invalid)**

**BAD\_REQUEST (movieid invalid)**

**Update User Password Endpoint:**

**URL:**

**users/password/user\_id**

**REQUEST:**

**{**

**“old\_password” : “<old password>”,**

**“new\_password” : “<new password>”**

**}**

**HEADER: u\_id, token**

**QUERY PARAMETER: NONE**

**METHOD: POST**

**RESPONSE:**

**ONSUCCESS 🡪 Success Msg Object**

**ONERROR 🡪 Error Msg Object**

**STATUS CODE:**

**UNAUTHORIZED (token invalid)**

**BAD\_REQUEST (movieid invalid)**