

The **Movie Booking** application will expose endpoints required to do the CRUD operations with **movies** database maintained in MongoDB.

The application will make use of several node package(s) including Express, Mongoose for server-side development.

Application should send or receive data only if the user is authenticated and authorized.

## MongoDB Collections

Below are the field level details in Mongo DB Collections:

Database		moviesdb	
Collection		genres	
Fields & Description(s)			
S. No	Field Name	Type	Description
1	genreid	Number	Unique Identity for each Genre
2	genre	String	A String value denoting genre name
Sample JSON			
[ {"genreid": 1, "genre": "comedy"}, {"genreid": 2, "genre": "drama"}, {"genreid": 3, "genre": "action"}, {"genreid": 4, "genre": "romance"}, {"genreid": 5, "genre": "horror"} ]			

Database		moviesdb	
Collection		artists	
Fields & Description(s)			
S. No	Field Name	Type	Description
1	artistid	Number	Unique Identity for each Artist
2	first_name	String	First Name of the Artist
3	last_name	String	Last Name of the Artist
4	wiki_url	String	Link to Artist's Wikipedia Page
5	profile_url	String	Link to Artist's Photo Image

6	movies	Array	List of Movies in which Artist Occurred. Default: Keep it blank
<b>Sample JSON</b>			
<pre>[     {         "artistid": 1, "first_name": "amitabh", "last_name": "bachchan",         "wiki_url": "https://en.wikipedia.org/wiki/Amitabh_Bachchan",         "profile_url": "https://wikibio.in/wp-content/uploads/2017/12/Amitabh-Bachchan.jpg", "movies": []},     {         "artistid": 2, "first_name": "nasiruddin", "last_name": "shah",         "wiki_url": "https://en.wikipedia.org/wiki/Naseeruddin_Shah",         "profile_url": "https://wikibio.in/wp-content/uploads/2019/06/Naseeruddin-Shah.jpg", "movies": []},     {         "artistid": 3, "first_name": "rajkumar", "last_name": "rao",         "wiki_url": "https://en.wikipedia.org/wiki/Rajkummar_Rao",         "profile_url": "https://i1.wp.com/wikifamouspeople.com/wp-content/uploads/2018/09/rajkumar-rao.jpg?fit=768%2C432&amp;ssl=1", "movies": []} ]</pre>			

Database		moviesdb	
Collection		movies	
Fields & Description(s)			
S. No	Field Name	Type	Description
1	movieid	Number	Unique Identity for each Movie
2	title	String	Name of the Movie
3	published	Boolean	Is Movie Published
3	released	Boolean	Is Movie Released
4	poster_url	String	Link to Movies Poster Image or Photo
5	release_date	String	Date on which Movie Released in string form
6	publish_date	String	Date on which Movie Published in string form
7	artists	Array	Artist(s) presentation from Artists Collection in JSON form. It can be used by combining Artist collection programmatically. However here to understand collection composition artists collection is part of movies collection
8	genres	Array	Genres(s) presentation from Genres Collection in JSON form. It can be used by combining Genres collection

			programmatically. However here to understand collection composition Genres collection is part of movies collection
9	duration	Number	Number denoting length of the movie in minutes
10	critic_rating	Number	Number denoting Critic's rating for the movie [Considered range from 1 to 5]
11	trailer_url	String	Link to Movies Trailer Video
12	wiki_url	String	Link to Movies Wikipedia Page
13	story_line	String	Story line or plot of the movie in short manner
14	shows	Array	"Shows" collection detailing where and at what time show is arranged. Sample explains it better

#### Sample JSON

[SCROLL for details]

```
[
  {
    "movieid":1,
    "title":"The Lodgers",
    "published":true,
    "released": true,
    "poster_url":"https://images-na.ssl-images-amazon.com/images/M/MV5BM2FhM2E1MTktMDYwZi00ODA1LWI0YTtN2NjZjM3ODFjYmU5XkEyXkFqcGdeQXVyMjY1ODQ3NTA@._V1_SY500_CR0,0,337,500_AL_.jpg",
    "release_date":"1/1/2020",
    "publish_date": "2/2/2020",
    "artists":[
      { "artistid": 1, "first_name": "amitabh", "last_name": "bachchan",
        "wiki_url":"https://en.wikipedia.org/wiki/Amitabh_Bachchan",
        "profile_url":"https://wikibio.in/wp-content/uploads/2017/12/Amitabh-Bachchan.jpg",
        "movies":[]},
      { "artistid": 2, "first_name": "nasiruddin", "last_name": "shah",
        "wiki_url":"https://en.wikipedia.org/wiki/Naseeruddin_Shah",
        "profile_url":"https://wikibio.in/wp-content/uploads/2019/06/Naseeruddin-Shah.jpg",
        "movies":[]},
      {"artistid": 3, "first_name": "rajkumar", "last_name": "rao",
```

```

"wiki_url": "https://en.wikipedia.org/wiki/Rajkummar_Rao",
"profile_url": "https://i1.wp.com/wikifamouspeople.com/wp-content/uploads/2018/09/rajkumar-rao.jpg?fit=768%2C432&ssl=1", "movies": []},

"genres": ["comedy", "drama", "action", "romance", "horror"],
"critic_rating": 3,
"trailer_url": "https://www.youtube.com/watch?v=ltIcW2xMuzs",
"wiki_url": "https://en.wikipedia.org/wiki/Main_Page",
"story_line":
    "1920, rural Ireland. Anglo Irish twins Rachel and Edward share a strange existence
    In their crumbling family estate. Each night, the property becomes the domain of a
    sinister presence (The Lodgers) which enforces three rules upon the twins"
"shows": [
    {
        "id": 1001,
        "theatre": {"name": "CityPride", "city": "Pune"},
        "language": "English",
        "show_timing": "1/1/2021",
        "available_seats": "5",
        "unit_price": 200
    },

    {
        "id": 1002,
        "theatre": {"name": "CityPride", "city": "Mumbai"},
        "language": "Hindi",
        "show_timing": "2/1/2021",
        "available_seats": "20",
        "unit_price": 100
    }
]

```

Database		moviesdb	
Collection		users	
Fields & Description(s)			
S. No	Field Name	Type	Description
1	userid	Number	Unique Identity for each User
2	email	String	A String value denoting user's email
3	first_name	String	A String value denoting user's first name

4	last_name	String	A String value denoting user's last name
5	username	String	A String value denoting username for login purpose. If the username is not given during registration from Registration UI then [first_name].[last_name] is used as username
6	contact	String	A String value denoting Mobile Number of the user
7	password	String	A String value denoting Password of the user used for Login purpose
8	role	String	A String value denoting Role of the user – “user” or “admin”
9	isLoggedIn	Boolean	Boolean value denoting if user has logged in
10	uuid	String	A universally unique identifier (UUID) denoting logged in user
11	accesstoken	String	Access token to be sent to client application which will be validated in each request
12	coupens	Array	An Array denoting discount coupons given to users
13	bookingRequests	Array	Denotes booking references including coupons used during registration along with show ID and seat details

#### Sample JSON

```
[
  {
    "userid": 1, "email": "a@b.com", "first_name": "user1", "last_name": "user1",
    "username": "test", "contact": "9898989898", "password": "test@123", "role": "user", "isLoggedIn": false,
    "uuid": "", "accesstoken": "",
    "coupens": [{"id": 101, discountValue: 101 }, {"id": 102, discountValue: 102 }],
    "bookingRequests": [{"reference_number": 29783, "coupon_code": 101, show_id: 1003, tickets: [1,3]},
      {"reference_number": 19009, "coupon_code": 201, show_id: 1002, tickets: [1]}
    ],
  },
  {
    "userid": 2, "email": "p@q.com", "first_name": "user2", "last_name": "user2", "username": "user",
    "contact": "9898989898", "password": "user@123", "role": "admin", "isLoggedIn": false, "uuid": "",
    "accesstoken": "",
    "coupens": [{"id": 103, discountValue: 103 }, {"id": 104, discountValue: 104 }],
    "bookingRequests": [{"reference_number": 29783, "coupon_code": 101, show_id: 1003, tickets: [1,3]},
      {"reference_number": 19009, "coupon_code": 201, show_id: 1002, tickets: [1]}
    ]
  }
]
```

## API Routes

Below is the list of APIs routes to be exposed from this application along with description:

**\*\*Note:** Parameter values to be passed in URL are mentioned in curly {} brackets

No	Movies Data related API Routes (needs to be prefixed with root URL)	Description
1	GET /movies?status=PUBLISHED	This API returns all movies from database with status set as "PUBLISHED"; this is handled through condition in code
2	GET /movies?status=RELEASED	This API returns all movies from database with status set as "RELEASED" "; this is handled through condition in code
3	GET /movie/{movieid}	This API returns details about movie with id passed as parameter "; this is handled through condition in code
4	GET/movies?status=RELEASED&title={title}&genres={genres}&artists={artists}&start_date={startdate}&end_date={enddate}	This API returns all movies data with filter criteria matching with parameters passed"; this is handled through condition in code
5	GET movies/{movieid}/shows	This API returns "shows" available for specific movie with id passed as parameter

No	Login & Booking related API Route	Description
1	POST /signup	This API will collect new users data and add it in DB
2	POST /login	This API will collect credentials and return access token if valid
3	POST /logout	This API will remove access token entries from DB
4	GET /coupons	This API will get discount coupons based on id
5	POST /bookings	This API will add show booking related entry and return a reference number stored in DB

No	Artist and Genres related API Route	Description
1	GET /artists	This API returns all artists from database
2	GET /genres	This API returns all genres from database