

**MYFLIX**

**API DOCUMENTATION**

## 1. Introduction

- Overview
- Base URL

## 2. Authentication

- Authentication Method
- How to Obtain Credentials

## 3. Services Overview

- List of Services

## 4. Endpoint Documentation

### 1. Content Service

- Add Movie
- Get All Movie
- Add TV Series
- Get All TV Series
- Add Documentary
- Get All Documentary

### 2. User Service

- Add General User
- Get All General Users
- Get User By Id
- Get User By Email
- Get All Admins

### 3. Review Service

- Add Review
- Get All Reviews
- Get Review By Content Id

### 4. Mail Service

- Send Registration Mail
- Send Content Update Mail

# 1. Introduction

## 1.1 Overview

Welcome to the “MTFLIX” Streaming Service Application Programming Interface (API) which is same as “NETFLIX”. This API provides access to key features of our streaming platform, enabling seamless integration and management of content, user information, reviews, and notifications.

Our API is structured around four main services:

- **Content Service:** Manage movies, TV series, and documentaries.
- **User Service:** Handle user registration, authentication, and user-related information.
- **Review Service:** Manage user reviews and ratings for content.
- **Mail Service:** Automatically send emails upon user registration and notify all registered users when new content is added.

This documentation will guide you through the available endpoints, request and response formats, and usage examples to help you effectively interact with our platform.

## 1.2 Base URL

Here are the Base URL for all API end points. You can GET, POST or DELETE data easily using this URL.

**`http://localhost/8081`**

## **2. Authentication**

### **2.1 Authentication Method**

There is no API key to access this API end points. You can easily get all data using relevant links with base URL.

### **2.2 How to Obtain Credentials**

There are no special credentials to obtained like API key requests, because there is no security on our API service.

## 3. Services Overview

### 3.1 List of Services

#### Content Service

The Content Service allows you to manage the various types of content available on the platform, including movies, TV series, and documentaries. This service provides endpoints to create, retrieve, update, and delete content, ensuring that the content catalog is always up-to-date and accurate.

#### User Service

The User Service handles all aspects of user management, including registration, authentication, and retrieval of user details. This service ensures secure and efficient management of user data, allowing users to register, log in, and maintain their profiles on the platform.

#### Review Service

The Review Service enables users to rate and review content on the platform. This service provides endpoints to add, update, and delete reviews, as well as to retrieve all reviews for specific content. By leveraging this service, users can share their opinions and help others discover great content.

#### Mail Service

The Mail Service is responsible for sending notifications via email. It performs two primary functions.

- **User Registration:** Sends a welcome email to users when they register on the platform.
- **Content Notifications:** Notifies all registered users when new content (movies, TV series, or documentaries) is added to the platform.

This service ensures that users stay informed about new additions to the content catalogue and remain engaged with the platform.

## 4. Endpoint Documentation

### 4.1 Content Service

#### 4.1.1 Add Movie

##### 4.1.1.1 End point

- Endpoint: "http://localhost:8081/addMovie"
- HTTP Method: "POST"

##### 4.1.1.2 Request Body Parameters

- **name** (String): The name of the movie.
- **overview** (String): A brief description of the movie.
- **genres** (Array of Strings): Genres associated with the movie.
- **backdrop\_path** (String): Path to the backdrop image.
- **poster\_path** (String): Path to the poster image.
- **adult** (Boolean): Indicates if the movie is for adults.
- **original\_language** (String): The original language of the movie.
- **release\_date** (String): The release date of the movie.
- **vote\_average** (Float): The average vote rating.
- **trailer** (String): URL to the trailer.
- **watch\_time** (Integer): Watch time in minutes.

##### 4.1.1.3 Request Example

```
{
  "adult": false,
  "backdrop_path": "planetApes",
  "genres": [
    "Science Fiction",
    "Adventure",
    "Action"
  ],
  "original_language": "English",
  "overview": "Several generations following Caesar's reign, apes - now the dominant species - live harmoniously while humans have been reduced to living in the shadows. As a new tyrannical ape leader builds his empire, one young ape undertakes a harrowing journey that will cause him to question
```

```

all he's known about the past and to make choices that will define a future for apes and humans
alike.",
    "poster_path": "planetApesPoster",
    "release_date": "2024-05-08",
    "name": "Kingdom of the Planet of the Apes",
    "vote_average": 6.994,
    "watch_time": "112",
    "trailer": "https://www.youtube.com/embed/XtFI7SntVpY?si=cvxRUiY8YrDK3QV0"
}

```

#### 4.1.1.4 Response Example

```

1  {
2    "adult": false,
3    "backdrop_path": "planetApes",
4    "genres": [
5      "Science Fiction",
6      "Adventure",
7      "Action"
8    ],
9    "movie_id": "MV001",
10   "original_language": "English",
11   "overview": "Several generations following Caesar's reign, apes – now the dominant species – live harmoniously while humans have been
               reduced to living in the shadows. As a new tyrannical ape leader builds his empire, one young ape undertakes a harrowing journey
               that will cause him to question all he's known about the past and to make choices that will define a future for apes and humans
               alike.",
12   "poster_path": "planetApesPoster",
13   "release_date": "2024-05-08",
14   "name": "Kingdom of the Planet of the Apes",
15   "vote_average": 6.994,
16   "watch_time": "112",
17   "trailer": "https://www.youtube.com/embed/XtFI7SntVpY?si=cvxRUiY8YrDK3QV0"
18 }

```

### 4.1.2 Get All Movies

#### 4.1.2.1 End point

- Endpoint: `http://localhost:8081/getAllMovies`
- HTTP Method: `GET`

#### 4.1.2.2 Request Body Parameters

Doesn't require a body.

### 4.1.2.3 Request Example

Doesn't require a body.

### 4.1.2.4 Response Example

```
1  [
2    {
3      "name": "Kingdom of the Planet of the Apes",
4      "overview": "Several generations following Caesar's reign, apes - now the dominant species - live harmoniously while humans have been reduced to living in the shadows. As a new tyrannical ape leader builds his empire, one young ape undertakes a harrowing journey that will cause him to question all he's known about the past and to make choices that will define a future for apes and humans alike.",
5      "genres": [
6        "Science Fiction",
7        "Adventure",
8        "Action"
9      ],
10     "backdrop_path": "planetApes",
11     "poster_path": "planetApesPoster",
12     "adult": false,
13     "original_language": "English",
14     "release_date": "2024-05-08",
15     "vote_average": 6.994,
16     "trailer": "https://www.youtube.com/embed/XtFI7SntVpY?si=cvxRUiY8YrDK3QV0",
17     "movie_id": "M1",
18     "watch_time": 112
19   },
20   {
21     "name": "Furiosa: A Mad Max Saga",
22     "overview": "As the world fell, young Furiosa is snatched from the Green Place of Many Mothers and falls into the hands of a great Biker Horde led by the Warlord Dementus. Sweeping through the Wasteland they come across the Citadel presided over by The Immortan Joe. While the two Tyrants war for dominance, Furiosa must survive many trials as she puts together the means to find her way home.",
```

This continues until the movie list ended.

## 4.1.3 Get Movie By Id

### 4.1.3.1 End point

- Endpoint: `"http://localhost:8081/getMovieById/{movie_id}"`
- HTTP Method: `"GET"`

### 4.1.3.2 Request Body Parameters

Doesn't require a body.

### 4.1.3.3 Request Example

Doesn't require a body.



### 4.1.3.4 Response Example

```
1 {
2   "name": "Descendants: The Rise of Red",
3   "overview": "After the Queen of Hearts incites a coup on Auradon, her rebellious daughter Red and Cinderella's perfect
4     Chloe join forces and travel back in time to try to undo the traumatic event that set Red's mother down her villain
5   "genres": null,
6   "backdrop_path": "riseOfRed",
7   "poster_path": "riseOfRedPoster",
8   "adult": false,
9   "original_language": "English",
10  "release_date": "2024-07-11",
11  "vote_average": 7.332,
12  "trailer": "https://www.youtube.com/embed/3Pk4-lzowDM?si=d5Cuj9_UP1aFB6N0",
13  "movie_id": "M4",
14  "watch_time": 130
15 }
```

## 4.1.4 Add TV Series

### 4.1.4.1 End point

- Endpoint: `"http://localhost:8081/addSeries"`
- HTTP Method: `"POST"`

### 4.1.4.2 Request Body Parameters

- `name` (String): The name of the TV Series.
- `overview` (String): A brief description of the TV Series.
- `genres` (Array of Strings): Genres associated with the TV Series.
- `backdrop_path` (String): Path to the backdrop image.
- `poster_path` (String): Path to the poster image.
- `adult` (Boolean): Indicates if the movie is for adults.
- `original_language` (String): The original language of the TV Series.
- `release_date` (String): The release date of the TV Series.
- `vote_average` (Float): The average vote rating.
- `trailer` (String): URL to the trailer.
- `num_of_seasons` (Integer): number of seasons.
- `num_of_episodes` (Integer): number of episodes.

#### 4.1.4.3 Request Example

```
{
  "backdrop_path": "sunny",
  "series_id": "TV001",
  "name": "Sunny",
  "overview": "The life of an American woman living in Kyoto, Japan, is upended when her husband and son disappear in a mysterious crash. As consolation, she's given Sunny, one of a new class of domestic robots made by her husband's electronics company.",
  "poster_path": "sunnyPoster",
  "adult": false,
  "original_language": "English",
  "genre_ids": [
    "Mystery",
    "Drama",
    "Comedy"
  ],
  "release_date": "2024-07-09",
  "vote_average": 7.3,
  "num_of_seasons": 3,
  "num_of_episodes": 31,
  "trailer": "https://www.youtube.com/embed/l-CcvSkft6Y?si=Dk9uUo4CAvRrLaP"
},
```

#### 4.1.4.4 Response Example

```
1  {
2    "backdrop_path": "sunny",
3    "series_id": "T15",
4    "name": "Sunny",
5    "overview": "The life of an American woman living in Kyoto, Japan, is upended when her husband and son disappear in a mysterious crash. As consolation, she's given Sunny, one of a new class of domestic robots made by her husband's electronics company.",
6    "poster_path": "sunnyPoster",
7    "adult": false,
8    "original_language": "English",
9    "genre_ids": [
10     "Mystery",
11     "Drama",
12     "Comedy"
13   ],
14   "release_date": "2024-07-09",
15   "vote_average": 7.3,
16   "num_of_seasons": 3,
17   "num_of_episodes": 31,
18   "trailer": "https://www.youtube.com/embed/l-CcvSkft6Y?si=Dk9uUo4CAvRrLaP"
19 }
```

### 4.1.5 Get All Series

#### 4.1.5.1 End point

- Endpoint: ["http://localhost:8081/getAllSeries"](http://localhost:8081/getAllSeries)
- HTTP Method: ["GET"](#)

#### 4.1.5.2 Request Body Parameters

Doesn't require a body.

#### 4.1.5.3 Request Example

Doesn't require a body.

#### 4.1.5.4 Response Example

```
[
  {
    "name": "Sunny",
    "overview": "The life of an American woman living in Kyoto, Japan, is upended when her husband and son disappear in a mysterious plane crash. As consolation, she's given Sunny, one of a new class of domestic robots made by her husband's electronics company.",
    "genres": null,
    "backdrop_path": "sunny",
    "poster_path": "sunnyPoster",
    "adult": false,
    "original_language": "English",
    "release_date": "2024-07-09",
    "vote_average": 7.3,
    "trailer": "https://www.youtube.com/embed/l-CcvSkft6Y?si=Dk9uUo4CAvRrLaP",
    "series_id": "T1",
    "num_of_seasons": 3,
    "num_of_episodes": 31
  },
  {
    "name": "House of the Dragon",
    "overview": "The Targaryen dynasty is at the absolute apex of its power, with more than 15 dragons under their yoke. Most empires crumble from such heights. In the case of the Targaryens, their slow fall begins when King Viserys breaks with a century of tradition by naming his daughter Rhaenyra heir to the Iron Throne. But when Viserys later fathers a son, the court is shocked when Rhaenyra retains her status as his heir, and seeds of division sow friction across the realm.",
    "genres": null,
    "backdrop_path": "hod",
    "poster_path": "hodPoster",
    "adult": false,
    "original_language": "English",
    "release_date": "2022-08-21",
    "vote_average": 8.4,
    "trailer": "https://www.youtube.com/watch?v=Kq0V8u8E8_8",
    "series_id": "T2",
    "num_of_seasons": 1,
    "num_of_episodes": 10
  }
]
```

This continues until the movie list ended.

### 4.1.6 Get Series By Id

#### 4.1.6.1 End point

- Endpoint: `"http://localhost:8081/getSeriesById/{series_id}"`
- HTTP Method: `"GET"`

#### 4.1.6.2 Request Body Parameters

Doesn't require a body.

#### 4.1.6.3 Request Example

Doesn't require a body.

#### 4.1.6.4 Response Example

```
1  {
2    "name": "Sunny",
3    "overview": "The life of an American woman living in Kyoto, Japan, is upended when her husband and son disappear in a
4    crash. As consolation, she's given Sunny, one of a new class of domestic robots made by her husband's electronics
5    company.
6    "genres": null,
7    "backdrop_path": "sunny",
8    "poster_path": "sunnyPoster",
9    "adult": false,
10   "original_language": "English",
11   "release_date": "2024-07-09",
12   "vote_average": 7.3,
13   "trailer": "https://www.youtube.com/embed/l-CcvSkft6Y?si=Dk9ulUo4CAvRrLaP",
14   "series_id": "T1",
15   "num_of_seasons": 3,
16   "num_of_episodes": 31
17 }
```

#### 4.1.7 Add Documentary

##### 4.1.7.1 End point

- Endpoint: ["http://localhost:8081/addDoc"](http://localhost:8081/addDoc)
- HTTP Method: ["POST"](#)

##### 4.1.7.2 Request Body Parameters

- [name](#) (String): The name of the Documentary.
- [overview](#) (String): A brief description of the Documentary.
- [genres](#) (Array of Strings): Genres associated with the Documentary.
- [backdrop\\_path](#) (String): Path to the backdrop image.
- [poster\\_path](#) (String): Path to the poster image.
- [adult](#) (Boolean): Indicates if the movie is for adults.
- [original\\_language](#) (String): The original language of the Documentary.
- [release\\_date](#) (String): The release date of the Documentary.
- [vote\\_average](#) (Float): The average vote rating.
- [trailer](#) (String): URL to the trailer.
- [watch\\_time](#) (Integer): Watch time in minutes.

### 4.1.7.3 Request Example

```
{
  "adult": false,
  "backdrop_path": "potter",
  "genre_ids": [
    "Documentary"
  ],
  "doc_id": "D2",
  "original_language": "English",
  "overview": "An enchanting making-of story told through all-new in-depth interviews and cast conversations, inviting fans on a first-person journey through one of the most beloved film franchises of all time.",
  "poster_path": "potterPoster",
  "release_date": "2022-01-01",
  "name": "    : Return to Hogwarts",
  "vote_average": 7.312,
  "watch_time": 70,
  "trailer": "https://www.youtube.com/embed/ffGS4zZWGoA?si=M-voeJsgrit82ADx"
},
```

### 4.1.7.4 Response Example

```
1  [
2    {
3      "adult": false,
4      "backdrop_path": "potter",
5      "genre_ids": [
6        "Documentary"
7      ],
8      "doc_id": "D2",
9      "original_language": "English",
10     "overview": "An enchanting making-of story told through all-new in-depth interviews and cast conversations, inviting fans on a first-person journey through one of the most beloved film franchises of all time.",
11     "poster_path": "potterPoster",
12     "release_date": "2022-01-01",
13     "name": "    : Return to Hogwarts",
14     "vote_average": 7.312,
15     "watch_time": 70,
16     "trailer": "https://www.youtube.com/embed/ffGS4zZWGoA?si=M-voeJsgrit82ADx"
17   }
18 ]
```

## 4.1.8 Get All Documentaries

### 4.1.8.1 End point

- Endpoint: `http://localhost:8081/getAllDoc`
- HTTP Method: `GET`

### 4.1.8.2 Request Body Parameters

Doesn't require a body.

### 4.1.8.3 Request Example

Doesn't require a body.

### 4.1.8.4 Response Example

```
1  [
2    {
3      "name": "Return to Hogwarts",
4      "overview": "An enchanting making-of story told through all-new in-depth interviews and cast conversations, inviting fans on a magical first-person journey through one of the most beloved film franchises of all time.",
5      "genres": null,
6      "backdrop_path": "potter",
7      "poster_path": "potterPoster",
8      "adult": false,
9      "original_language": "English",
10     "release_date": "2020-01-01",
11     "vote_average": 8.5,
12     "trailer": "https://www.youtube.com/embed/IFGS4zZWGoA?si=M-voeJsgrit82ADx",
13     "doc_id": "D1",
14     "watch_time": 70
15   },
16   {
17     "name": "The Servant",
18     "overview": "Domestic service worker Lera is accused of a crime. In a cell in the premises of a Civil Guard barracks, she meets Mihaela and Julia, two prostitutes with whom she shares the story of Santa Vicenta María, founder of the Religious of Mary Immaculate, which she says saved her life.",
19     "genres": null,
20     "backdrop_path": "servant",
21     "poster_path": "servantPoster",
22     "adult": false,
23     "original_language": "Spanish"
```

This continues until the movie list ended.

## 4.1.3 Get Documentary By Id

### 4.1.3.1 End point

- Endpoint: `"http://localhost:8081/getDocById/{movie_id}"`
- HTTP Method: `"GET"`

### 4.1.3.2 Request Body Parameters

Doesn't require a body.

### 4.1.3.3 Request Example

Doesn't require a body.

#### 4.1.3.4 Response Example

```
1  {
2    "name": "      : Return to Hogwarts",
3    "overview": "An enchanting making-of story told through all-new in-depth interviews and cast conversations, invitin
4      first-person journey through one of the most beloved film franchises of all time.",
5    "genres": null,
6    "backdrop_path": "potter",
7    "poster_path": "potterPoster",
8    "adult": false,
9    "original_language": "English",
10   "release_date": "2022-01-01",
11   "vote_average": 7.312,
12   "trailer": "https://www.youtube.com/embed/fFGS4zZWGoA?si=M-voeJsgrit82ADx",
13   "doc_id": "D1",
14   "watch_time": 70
15 }
```

## 4.2 User Service

### 4.2.1 Add General User

#### 4.2.1.1 End point

- Endpoint: "http://localhost:8081/addGuser"
- HTTP Method: "POST"

#### 4.2.1.2 Request Body Parameters

- **name** (String): The name of user.
- **email** (String): Email of the user.
- **password** (Array of Strings): User entered password.

#### 4.2.1.3 Request Example

```
1 {  
2   "name": "Tharanga Sandun",  
3   "email": "sandun@gmail.com",  
4   "password": "dsds65dss"  
5 }
```

#### 4.2.1.3 Response Example

```
1 {  
2   "name": "Tharanga Sandun",  
3   "email": "sandun@gmail.com",  
4   "password": "dsds65dss"  
5 }
```

### 4.2.2 Get All General Users

#### 4.2.2.1 End point

- Endpoint: "http://localhost:8081/getAllGuser"
- HTTP Method: "GET"



#### 4.2.2.2 Response Example

```
1  [
2    {
3      "name": "Tharanga Sandun",
4      "email": "tharanga@gmail.com",
5      "password": "1234",
6      "guser_id": "U1"
7    },
8    {
9      "name": "Tharanga Sandun",
10     "email": "sandun@gmail.com",
11     "password": "1234",
12     "guser_id": "U2"
13   },
14   {
15     "name": "Tharanga Sandun",
16     "email": "kumara@gmail.com",
17     "password": "1234",
18     "guser_id": "U3"
19   },
20   {
21     "name": "Tharanga Sandun",
22     "email": "madurap@gmail.com",
23     "password": "1234",
24     "guser_id": "U4"
25   }
26 ]
```

#### 4.2.3 Get User By Id

##### 4.2.3.1 End point

- Endpoint: `"http://localhost:8081/getGUserById/{id}"`
- HTTP Method: `"GET"`

##### 4.2.3.2 Response Example

```
1  {
2    "name": "Tharanga Sandun",
3    "email": "tharanga@gmail.com",
4    "password": "1234",
5    "guser_id": "U1"
6  }
```

## 4.2.4 Get User By Email

### 4.2.4.1 End point

- Endpoint: `"http://localhost:8081/getGUserByEmail/{email}"`
- HTTP Method: `"GET"`

### 4.2.4.2 Response Example

```
1 {  
2   "name": "Basuru",  
3   "email": "jagadakshibasuru@gmail.com",  
4   "password": "1234",  
5   "guser_id": "U8"  
6 }
```

## 4.2.4 Get All Admin Users

### 4.2.4.1 End point

- Endpoint: `"http://localhost:8081/getAllAdmin"`
- HTTP Method: `"GET"`

### 4.2.4.2 Response Example

```
1 [  
2   {  
3     "name": "Tharanga",  
4     "email": "admin@gmail.com",  
5     "password": "1234",  
6     "admin_id": "A2"  
7   }  
8 ]
```

## 4.3 Review Service

### 4.3.1 Add Review

#### 4.3.1.1 End point

- Endpoint: "http://localhost:8081/addReview"
- HTTP Method: "POST"

#### 4.3.1.2 Request Body Parameters

- `user_id` (String): user id of the user logged user.
- `contentId` (String): user clicked content id.
- `reviewText` (Array of Strings): entered review by user

#### 4.3.1.3 Request Example

```
25 {
26   "user_id": "U3",
27   "contentId": "M4",
28   "reviewText": "A powerful story of hope and friendship. Tim Robbins and Morgan Freeman deliver unforgettable performances.
29     classic."
30 }
```

#### 4.3.1.3 Response Example

```
25 {
26   "review_id": "R19",
27   "user_id": "U10",
28   "contentId": "M4",
29   "reviewText": "A powerful story of hope and friendship. Tim Robbins and Morgan Freeman deliver unforgettable performances.
30     classic.",
31   "timeStamp": "2024-08-04"
32 }
```

### 4.3.2 Get All Review

#### 4.3.2.1 End point

- Endpoint: "http://localhost:8081/getAllReview"
- HTTP Method: "GET"

#### 4.3.2.2 Request Body Parameters

#### 4.3.2.3 Request Example

#### 4.3.2.4 Response Example

```
{
  "review_id": "R3",
  "user_id": "U1",
  "contentId": "M2",
  "reviewText": "A mind-bending masterpiece! Christopher Nolan's direction and Hans Zimmer's score create a thrilling journey through dreams. Absolutely captivating!",
  "timeStamp": "2024-07-31"
},
{
  "review_id": "R4",
  "user_id": "U2",
  "contentId": "M2",
  "reviewText": "A powerful story of hope and friendship. Tim Robbins and Morgan Freeman deliver unforgettable performances. A timeless classic.",
  "timeStamp": "2024-07-31"
},
{
  "review_id": "R5",
  "user_id": "U2",
  "contentId": "M3",
  "reviewText": "Heath Ledger's Joker is iconic. This movie redefined the superhero genre with its dark, gripping narrative. Must-watch!",
  "timeStamp": "2024-07-31"
},
}
```

### 4.3.3 Get Review By Content Id

#### 4.3.2.1 End point

- Endpoint: ["http://localhost:8081/getReviewByContent/{contentId}"](http://localhost:8081/getReviewByContent/{contentId})
- HTTP Method: **"GET"**

#### 4.3.2.2 Request Body Parameters

#### 4.3.2.3 Request Example

#### 4.3.2.4 Response Example

```
1 [
2   {
3     "review_id": "R3",
4     "user_id": "U1",
5     "contentId": "M2",
6     "reviewText": "A mind-bending masterpiece! Christopher Nolan's direction and Hans Zimmer's score create a thrilling journey through dreams. Absolutely captivating!",
7     "timeStamp": "2024-07-31"
8   },
9   {
10    "review_id": "R4",
11    "user_id": "U2",
12    "contentId": "M2",
13    "reviewText": "A powerful story of hope and friendship. Tim Robbins and Morgan Freeman deliver unforgettable performances. A timeless classic.",
14    "timeStamp": "2024-07-31"
15  }
16 ]
```

## 4.4 Mail Service

### 4.4.1 Send Registration Mail

#### 4.4.1.1 End point

- Endpoint: `"http://localhost:8081/registration/{receiverMail}"`
- HTTP Method: `"POST"`

#### 4.4.1.2 Request Body Parameters

- `name` (String): name of the user.
- `subject` (String): subject for the mail.

#### 4.4.1.3 Request Example

```
1 {  
2   ... "name": "Tharanga",  
3   ... "subject": "Registration mail"  
4 }
```

#### 4.4.1.4 Response Example

```
1 Successfully sent the mail!!
```

### 4.4.2 Send Content Update Mail

#### 4.4.2.1 End point

- Endpoint: `"http://localhost:8081/contentUpdate/{receiverMail}"`
- HTTP Method: `"POST"`

#### 4.4.2.2 Request Body Parameters

- **name** (String): name of the user.
- **subject** (String): subject for the mail.
- **nameContent** (String): name of the content (movie name, series name).
- **content** (String): content type (movie, documentary, TV series).
- **overview** (String): overview for content.
- **original\_language** (String): speaking language of the content.
- **release\_date** (String): content release date.
- **vote\_average** (double): votes for the content.
- **trailer** (String): you tube trailer of the content.

#### 4.4.2.3 Request Example

```
1 {  
2   ...."subject": "Testing",  
3   ...."name": "Tharanga Sandun",  
4   ...."nameContent": "Godzilla",  
5   ...."original_language": "English",  
6   ...."release_date": "03.08.2024",  
7   ...."vote_average": 6.45,  
8   ...."trailer": "https://www.youtube.com/embed/lV100lGwExM?si=o0U3qktAt5uPv_kM",  
9   ...."overview": "As the world fell, young Furiosa is snatched from the Green Place of Many Mothers and falls into the hands of a great Bik  
    Horde led by the Warlord Dementus. Sweeping through the Wasteland they come across the Citadel presided over by The Immortan Joe.  
    While the two Tyrants war for dominance, Furiosa must survive many trials as she puts together the means to find her way home."  
10 }
```

#### 4.4.2.4 Response Example

```
1 Successfully sent the mail!!|
```