

REST API Design

Chitram Dasgupta

July 9, 2025

Contents

1	Scenario	2
2	Solution	2
2.1	Identify the core resources	2
2.2	Design the Endpoints	2
2.2.1	User endpoints	2
2.2.2	Bookmark endpoints	2
2.3	Request-Response Cycle	2
2.3.1	Signup	2
2.3.2	Login	3
2.3.3	Creating a Bookmark	3

1 Scenario

We want to create a bookmark manager api where users will be able to register and authenticate themselves. They will also be able to create, update, and delete bookmarks.

2 Solution

2.1 Identify the core resources

1. **User:** id, email, password (hashing)
2. **Bookmark:** id, userId, title, url

One user can create **many** bookmarks.
So, this is a **1-to-many** association.

2.2 Design the Endpoints

2.2.1 User endpoints

Method	Endpoint	Description
POST	/signup	Create an account
POST	/login	Authenticate, return token
GET	/me	Get logged-in user info

2.2.2 Bookmark endpoints

Method	Endpoint	Description
GET	/bookmarks	List bookmarks (for user)
POST	/bookmarks	Add new bookmark
GET	/bookmarks/:id	View one bookmark
PUT	/bookmarks/:id	Update a bookmark
DELETE	/bookmarks/:id	Delete a bookmark

2.3 Request-Response Cycle

2.3.1 Signup

Request

```
POST /signup
Content-Type: application/json
```

```
{
  "email": "chitramdasgupta@example.com",
  "password": "secret123"
}
```

Response

```
{
  "message": "User created",
  "user": {
    "id": 1,
    "email": "chitramdasgupta@example.com"
  }
}
```

2.3.2 Login

Request

POST /login
Content-Type: application/json

```
{
  "email": "chitramdasgupta@example.com",
  "password": "secret123"
}
```

Response

```
{
  "token": "jwt-or-session-token"
}
```

After successful authentication, we will send **Authorization: Bearer <token>** along with our requests.

2.3.3 Creating a Bookmark

Request

POST /bookmarks
Authorization: Bearer <token>

```
{
  "title": "Wikipedia",
  "url": "https://en.wikipedia.org"
}
```

Response

```
{
  "id": 43,
  "title": "Wikipedia",
  "url": "https://en.wikipedia.org"
}
```