Spam Detection System – API Usage Guide

1. Prerequisites

Before running the project, ensure you have the following installed:

- **Java 17**
- **Maven 3.6+**
- **MySQL**

Database Setup

1. Open MySQL and create a new schema using the following command:
   `CREATE SCHEMA spam_detection;`
2.
3. Update your `application.properties` file with your MySQL credentials:
   `spring.datasource.url=jdbc:mysql://localhost:3306/spam_detection`
4. `spring.datasource.username=<your-username>`
5. `spring.datasource.password=<your-password>`
6.

2. Running the Application

1. **Download the project:** Since you are not using GitHub, extract the project files in your IDE.
2. **Navigate to the project directory.**
3. **Build and run the application:**
   `mvn clean install`
4. `mvn spring-boot:run`
5.
6. **Access the application:** The API will be available at:
   `http://localhost:8080/`
7.
8. **Database Initialization:** The database will be automatically populated using `data.sql`.

3. API Endpoints & Usage

User Registration

Endpoint: `POST /api/register`
Description: Registers a new user.
Request Body (JSON):

```
{
    "name": "Prabhat Kumar",
    "phoneNumber": "9876543210",
    "email": "prabhat@example.com",
    "password": "Prabhat12@"
}
```

Authentication: ❌ Not Required

User Login

Endpoint: `POST /auth/login`
Description: Authenticates a user and returns a JWT token.
Request Body (JSON):

```
{
    "phoneNumber": "9876543210",
    "password": "securepassword"
}
```

Authentication: ❌ Not Required
Response: JWT Token (to be used in Authorization header)

Update User Profile

Endpoint: `PUT /api/update`
Description: Updates the user's name or email.
Request Header:

```
Authorization: Bearer <JWT-TOKEN>
```

Request Body (JSON):

```
{
    "name": "New Name",
    "email": "newemail@example.com"
}
```

Authentication: ✅ Required

Search User by Name

Endpoint: `GET /search/searchByName`
Description: Searches users by name.
Request Example:

```
GET /search/searchByName?name=Prabhat Kumar
```

Authentication: ✅ Required

Search User by Phone Number

Endpoint: `GET /search/searchByNumber`
Description: Searches for a phone number in the global database.
Request Example:

```
GET /search/searchByNumber?phone_number=9876543210
```

Authentication: ✅ Required

View User Profile

Endpoint: `GET /search/displayProfile`
Description: Displays the full profile of a user.
Request Example:

```
GET /search/displayProfile?phone_number=9876543210
```

Authentication: ✅ Required

Report a Number as Spam

Endpoint: `POST /spam/report`
Description: Marks a number as spam.
Request Example:

```
POST /spam/report?phone_number=9876543210
```

Authentication: ✅ Required

4. Authentication

For all endpoints that require authentication, include the JWT Token in the request header:

```
Authorization: Bearer <JWT-TOKEN>
```

```
mysql> show tables;

+---------------------+
| Tables_in_callerid |
+---------------------+
| contacts            |
| spam_reports        |
| users               |
+---------------------+
```