

REST API WITH SPRING BOOT

Design Diagram



EASY SETUP INSTRUCTIONS

- Create Spring Initializer Project at <https://start.spring.io/>
- Provide Group and Artifact as `com.solstice` and `contact`
- Unzip the artifact and create existing Maven Project in Eclipse
- Copy the src content from the source Zip code and `pom.xml`
- Modify the application properties for database credentials and access
- The Server will be run on `localhost:8080` while running the maven build
- The welcoming html shows some of the api functionalities
- Alternatively run `mvn package` and generate the jar file which can be easily deployable in AWS

UNIT TEST CASE LOCATIONS

- `src/test/java`

UI SCREENS

Add/Edit Contact Screen

My Contacts

File ▾

New Contact

First name

Last name

Upload Photo

Choose Files

No file chosen

Email

Phone Number

Address

City

Country

Choose...

State

Choose...

Zip

Save Contact

[Back to top](#)

List/Delete/Search Screen

My Contacts

File ▾

Search

Name : Pinky Sindhu
Email : pinky.sindhu18@gmail.com
Phone Number : 8479975828
Address : 1804 Oslo Ct, Mundelein,
Illinois, 60060, United States

EditDelete

Page created with [Bootstrap](#)

[Back to top](#)

REST API ENDPOINTS

Content-Type = application/json

1. To find the contact by ID

<http://192.168.0.105:8080/api/contact/{id}>

Request : GET : <http://192.168.0.105:8080/contact/1>

2. To find the list of all Contacts

<http://192.168.0.105:8080/api/list/>

Request : GET : <http://192.168.0.105:8080/api/list/>

Response

```
[{"id":4,"firstname":"Jane","lastname":"Doe","email":"jane.doe@gmail.com","phonenumber":"12345678","image":"123","city":"Chicago","state":"Illinois","country":"United States","zip":"54321","address":"128 Lans Lane"}]
```

3. To Delete all Contacts

<http://192.168.0.105:8080/api/delete>

Request : DELETE : <http://192.168.0.105:8080/api/delete>

4. To delete contact by Id

<http://192.168.0.105:8080/api/delete/{id}>

Request : GET : <http://192.168.0.105:8080/api/delete/1>

5. To search contact by Phone Number/Email

<http://192.168.0.105:8080/api/search/{value}>

Request : GET : <http://192.168.0.105:8080/api/contact/pinky@email.com>

6. To retrieve contact by State or City

<http://192.168.0.105:8080/api/retrieve/{value}>

Request : GET : <http://192.168.0.105:8080/api/contact/Illinois>

7. To Save a Contact

<http://192.168.0.105:8080/api/>

Request :

```
{  
  "id":4,  
  "firstname":"Jane",  
  "lastname":"Doe",
```

```
"email": "jane.doe@gmail.com",  
"phonenumber": "12345678",  
"image": "<base64 data>",  
"city": "Chicago",  
"state": "Illinois",  
"country": "United States",  
"zip": "54321",  
"address": "128 Lans Lane"  
}
```