

Mediscreen



REST API Specification

—

Name: Add Patient

URL: /patient/api/add

Method: post

URL Parameters: None

Data Parameters: JSON body.

```
{  
  "familyName": "string",  
  "givenName": "string",  
  "dob": "string",  
  "sex": "string",  
  "address": "string",  
  "phone": "string"  
}
```

Response (Success):

Code: 201

Content: Added Patient JSON

Response (Failure):

Code:400

Content: "Failed to add new entry"

Sample Call:

```
$ curl --header "Content-Type: application/json" --request POST --data '{"familyName":"famname","givenName":"givenname","dob":"2012-04-23","sex":"F","address":"testaddress","phone":"111-222-3333"}' http://localhost:8080/patient/api/add
```

Notes:

Input values must meet validation requirements:

Values must not be null

Sex must be 'M' or 'F'

Phone must be in the format '123-456-7890'

—

Name: Get Patient

URL: /patient/api/get/{id}

Method: get

URL Parameters: {id} = patientId

Data Parameters: None

Response (Success):

Code: 200

Content: Patient JSON

Response (Failure):

Code:404

Content: "Id {id} not found."

Sample Call:

```
$ curl http://localhost:8080/patient/api/get/13
```

Notes:

—

Name: Update Patient

URL: /patient/api/update

Method: put

URL Parameters: None

Data Parameters: JSON body.

```
{
  "patientId":int
  "familyName":"string",
  "givenName":"string",
  "dob":"string",
  "sex":"string",
  "address":"string",
  "phone":"string"
}
```

Response (Success):

Code: 200

Content: Updated Patient JSON

Response (Failure):

If provided Patient JSON is invalid

Code:400

Content: Provided Patient JSON for verification

If no Patient exists in system with provided patientId

Code:404

Content: Provided Patient JSON for verification

Sample Call:

```
$ curl --header "Content-Type: application/json" --request PUT --data '{"patientId":22,"familyName":"famname","givenName":"givenname","dob":"2012-04-23","sex":"F","address":"testaddress","phone":"111-222-3333"}' http://localhost:8080/patient/api/update
```

Notes:

Input values must meet validation requirements:

Values must not be null

Sex must be 'M' or 'F'

Phone must be in the format '123-456-7890'

—
Name: Add Note to Patient History

URL: /patient/note/api/add

Method: post

URL Parameters: None

Data Parameters: JSON body.

```
{  
  "patId":int,  
  "note":"string",  
}
```

Response (Success):

Code: 201

Content: Added Patient Note JSON

Response (Failure):

Code:400

Content: "Failed to add new entry"

Sample Call:

```
$ curl --header "Content-Type: application/json" --request POST --data  
'{"patId":"#","note":"patient note"}' http://localhost:8181/patient/note/api/add
```

Notes:

Input values must meet validation requirements:

Values must not be null

PatID must be an integer

—

Name: Get Patient Note

URL: /patient/note/api/get/{id}

Method: get

URL Parameters: {id} = patientNotelId

Data Parameters: None

Response (Success):

Code: 200

Content: Patient Note JSON

Response (Failure):

Code:404

Content: "Id {id} not found."

Sample Call:

\$ curl http://localhost:8181/patient/note/api/get/6298acf5cca3e84f668afb4c

Notes:

Note that PatientNotelD is a String rather than an Integer

—

Name: Get All Patient Notes for a Patient

URL: /patient/note/api/getbypatient/{id}

Method: get

URL Parameters: {id} = patId

Data Parameters: None

Response (Success):

Code: 200

Content: JSON of list of Patient Notes

Response (Failure):

Code:404

Content: "Id {id} not found."

Sample Call:

\$ curl http://localhost:8181/patient/note/api/getbypatient/1

Notes:

—

Name: Update Patient Note

URL: /patient/note/api/update

Method: put

URL Parameters: None

Data Parameters: JSON body.

```
{
  "patientNoteId": "string"
  "patId": "string",
  "note": "string",
}
```

Response (Success):

Code: 200

Content: Updated PatientNote JSON

Response (Failure):

If provided PatientNote JSON is invalid

Code: 400

Content: Provided Patient JSON for verification

If no PatientNote exists in system with provided patientId

Code: 404

Content: Provided Patient JSON for verification

Sample Call:

```
$ curl --header "Content-Type: application/json" --request PUT --data '{"patient-NoteId": "NOTEID", "patId": "123", "note": "sample note text"}' http://localhost:8181/patient/api/update
```

Notes:

Input values must meet validation requirements:

PatientNoteID must be a valid PatientNote's ID String

PatID must be an integer

Note must not be null

—

Name: Get Patient Assessment

URL: /assessment/api/get/{id}

Method: get

URL Parameters: {id} = patientId

Data Parameters: None

Response (Success):

Code: 200

Content: Patient Assessment as String

"Patient: {GivenName} {FamilyName} (age {age}) diabetes assessment is: {risk}"

Response (Failure):

Code:404

Content: "Patient {id} not found."

Code:404

Content: "Patient {id} has no notes."

Sample Call:

\$ curl http://localhost:8282/assessment/api/get/1

Notes:

Note that patient ID is an integer

If patient does not have any notes, an error is returned to inform of this rather than producing a 'no risk' assessment, to ensure it is clear why no risk has been assessed.