

Crime management project

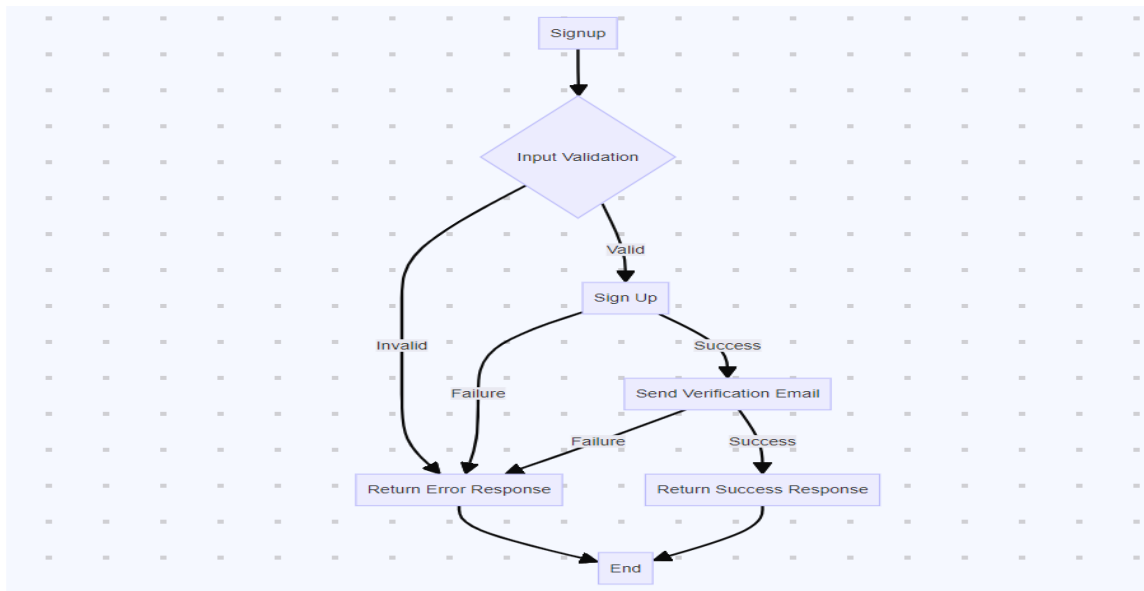
List of end points

authentication

```
1. @PostMapping("/signup")
```

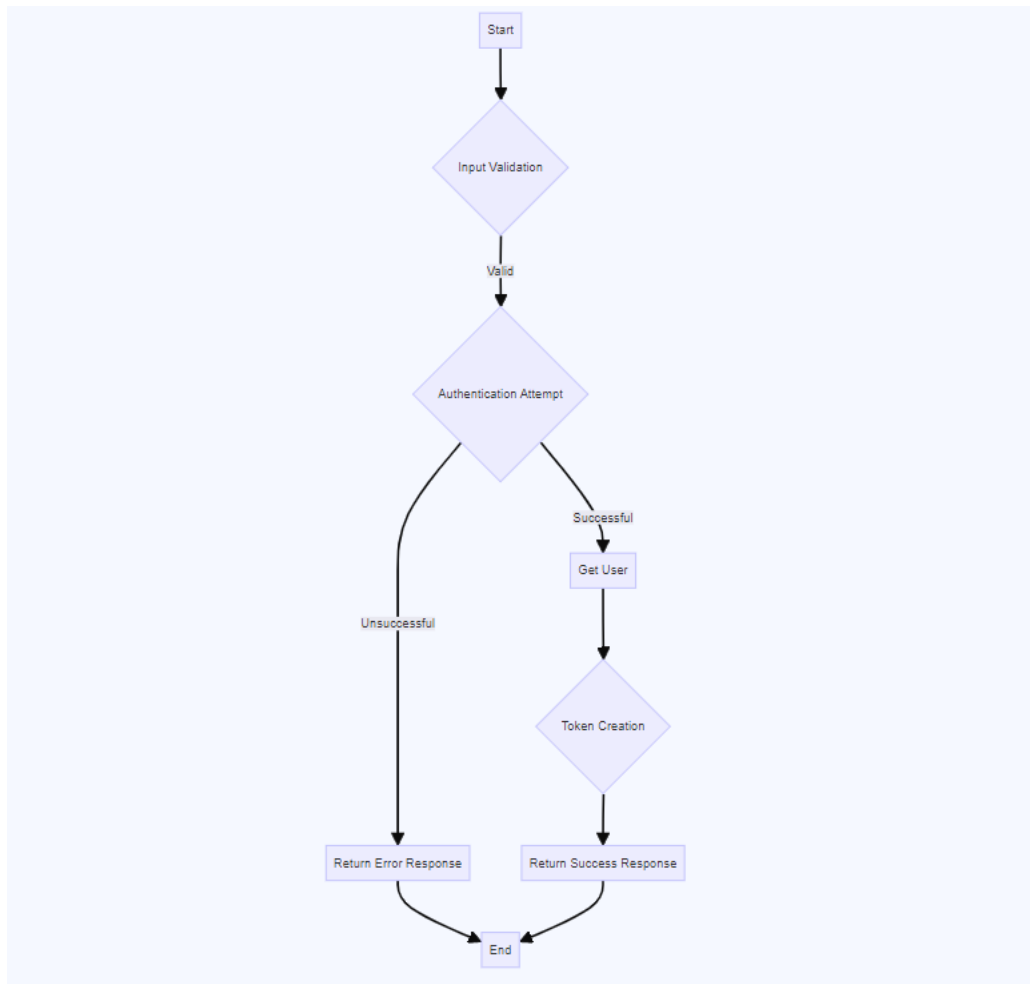
Signup expected data.

```
{  
  "email": "test@prison.com",  
  "password": "Olanrewaju1",  
  "firstName": "qudus",  
  "lastName": "ajoke",  
  "dateOfBirth": "24-12-1992",  
  "phoneNumber": "07035.....",  
  "address": "ogba"  
}
```



Login expected data

```
1. @PostMapping("/login")
{
    "email": "test@prison.com",
    "password": "Ajagbeejo1"
}
```



Verification

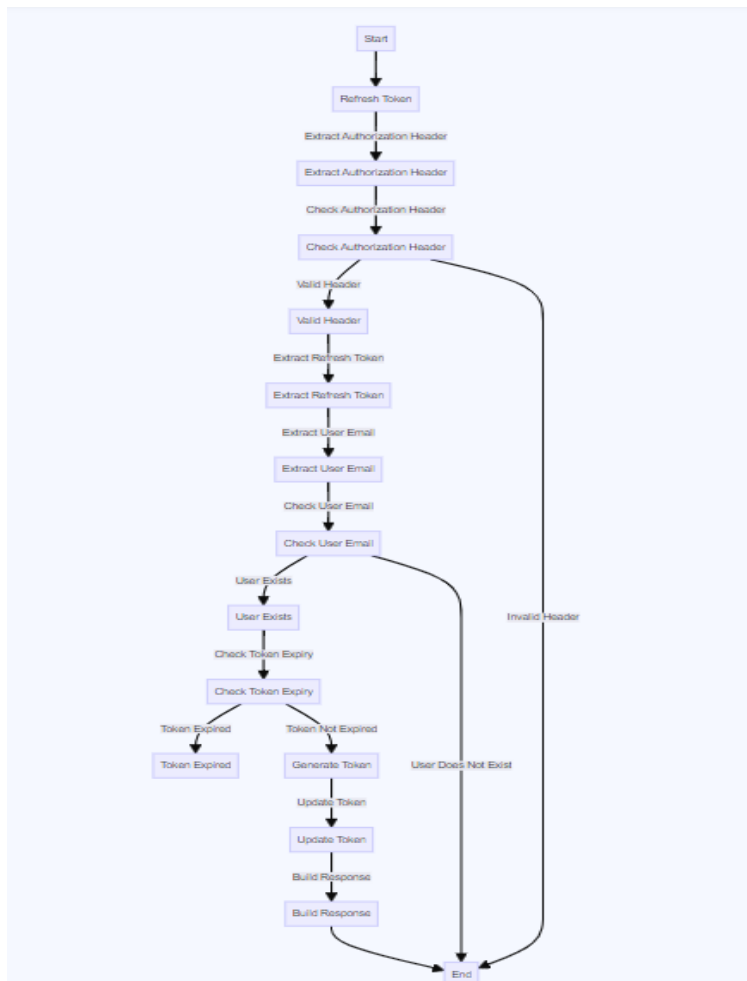
```
1. @GetMapping("/verification")
```

User needs to click the verification link send to the registered email before they can logging this verification link expired withing 10 minutes.

Refresh token

```
1. @PostMapping("/refresh-token")
```

The expired to ken is needed to generate new access token.

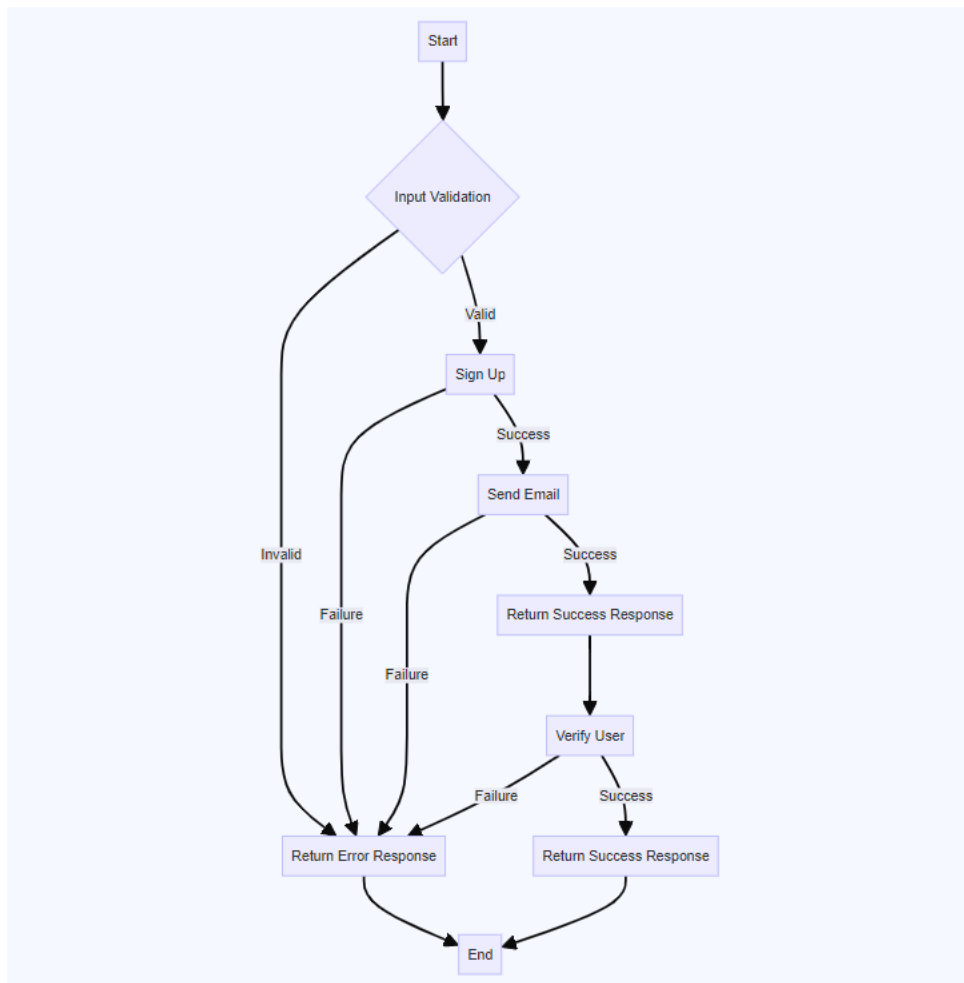


Signup by admin

```
2. @PostMapping("/signup-by-admin")
3.
```

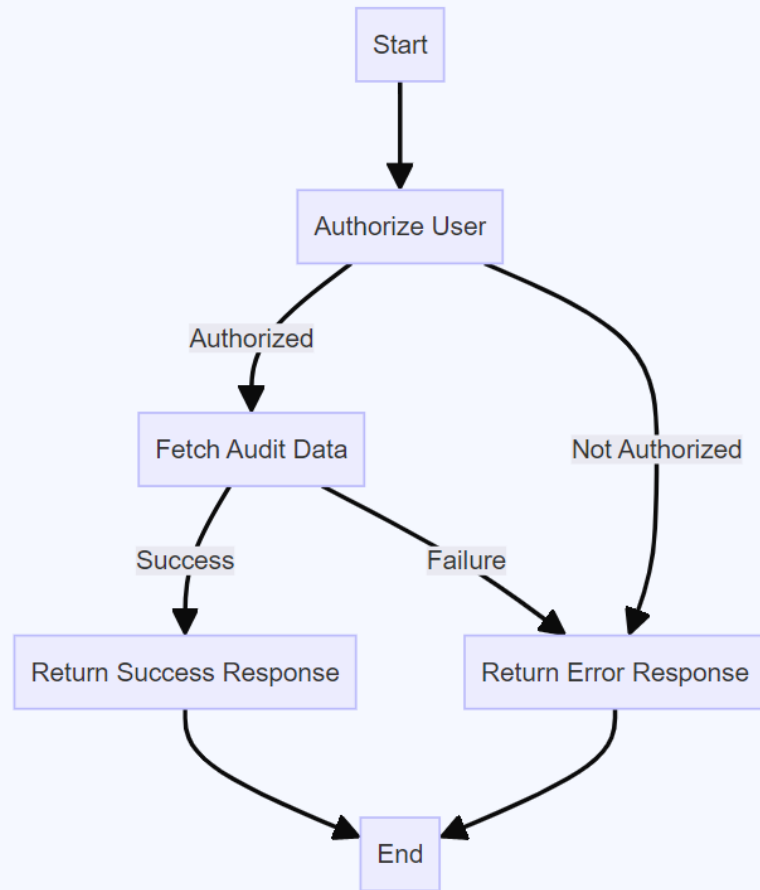
This is use to signup for a staff that doesn't already have a user account.

```
{
  "email": "test@prison.com",
  "password": "Lanrewaju1",
  "firstName": "qudus",
  "lastName": "olanrewaju",
  "dateOfBirth": "24-12-1992",
  "phoneNumber": "070357....",
  "address": "ikeja"
}
```

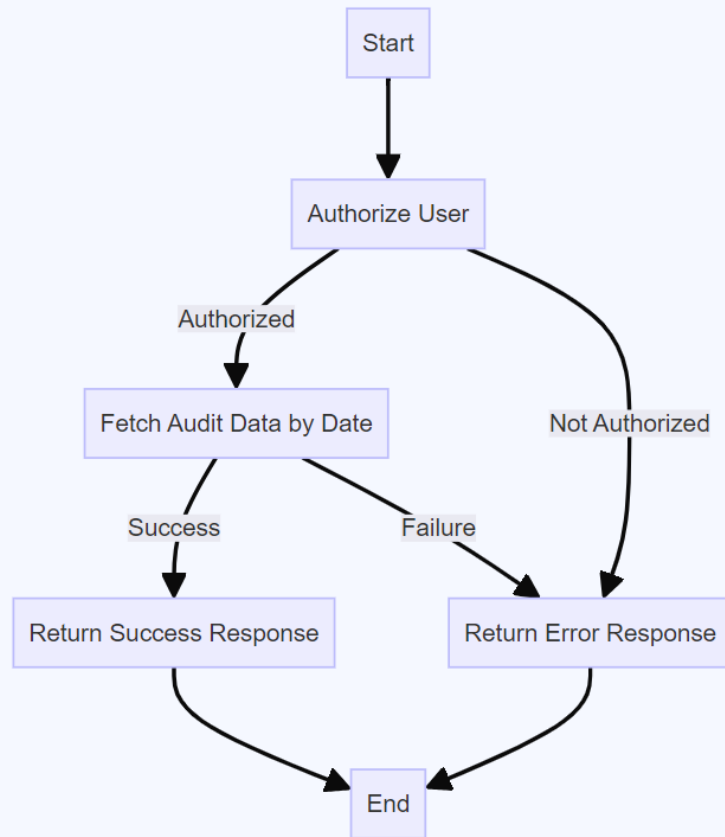


Audit trailing

```
1. @GetMapping("/all-audit")
```

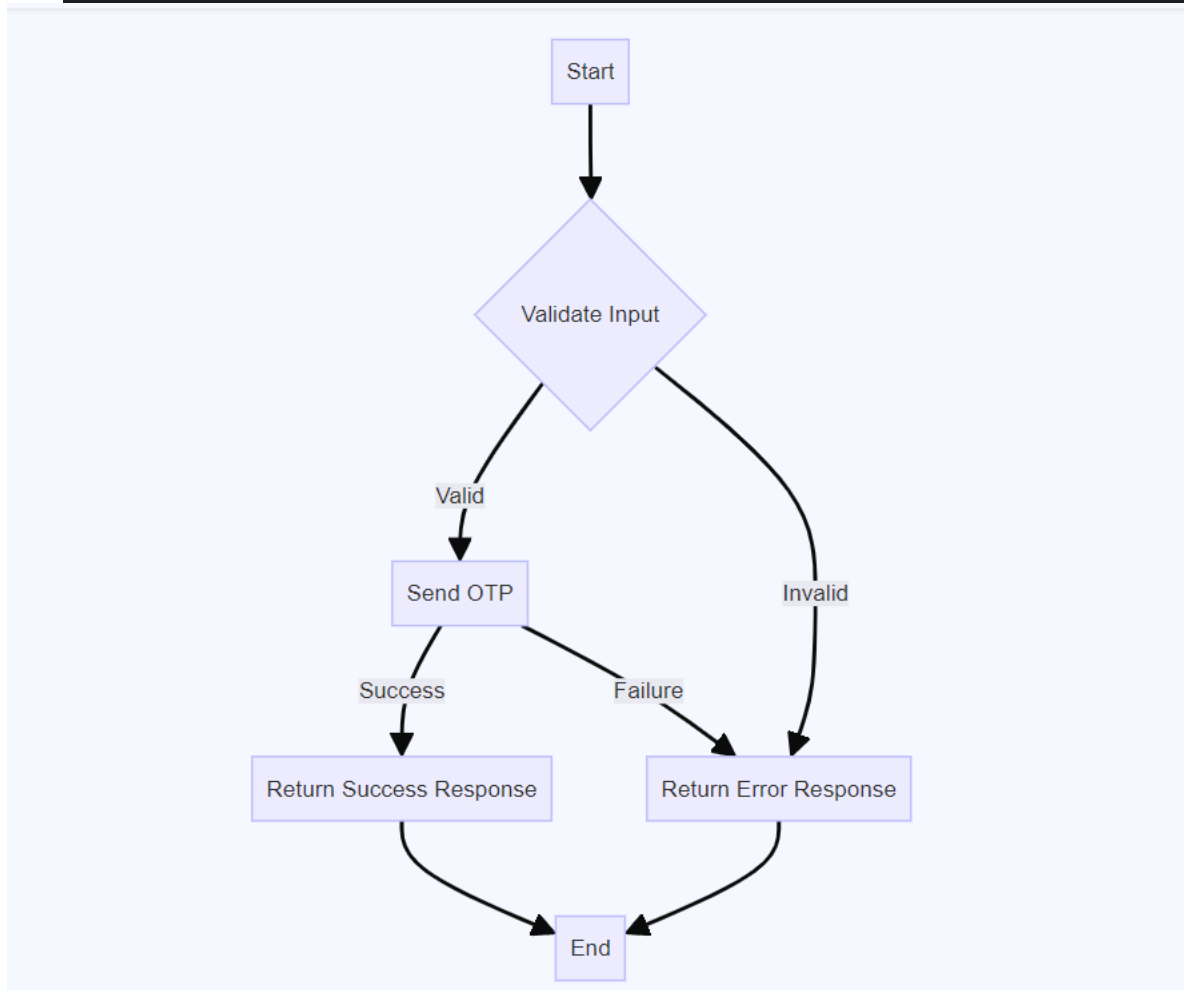


```
1. @GetMapping("/get-by-date")
```



Change Data

```
1. @PostMapping("/forgetpassword/inputOtp")
2. @PostMapping("/forgetpassword")
```



Forget password

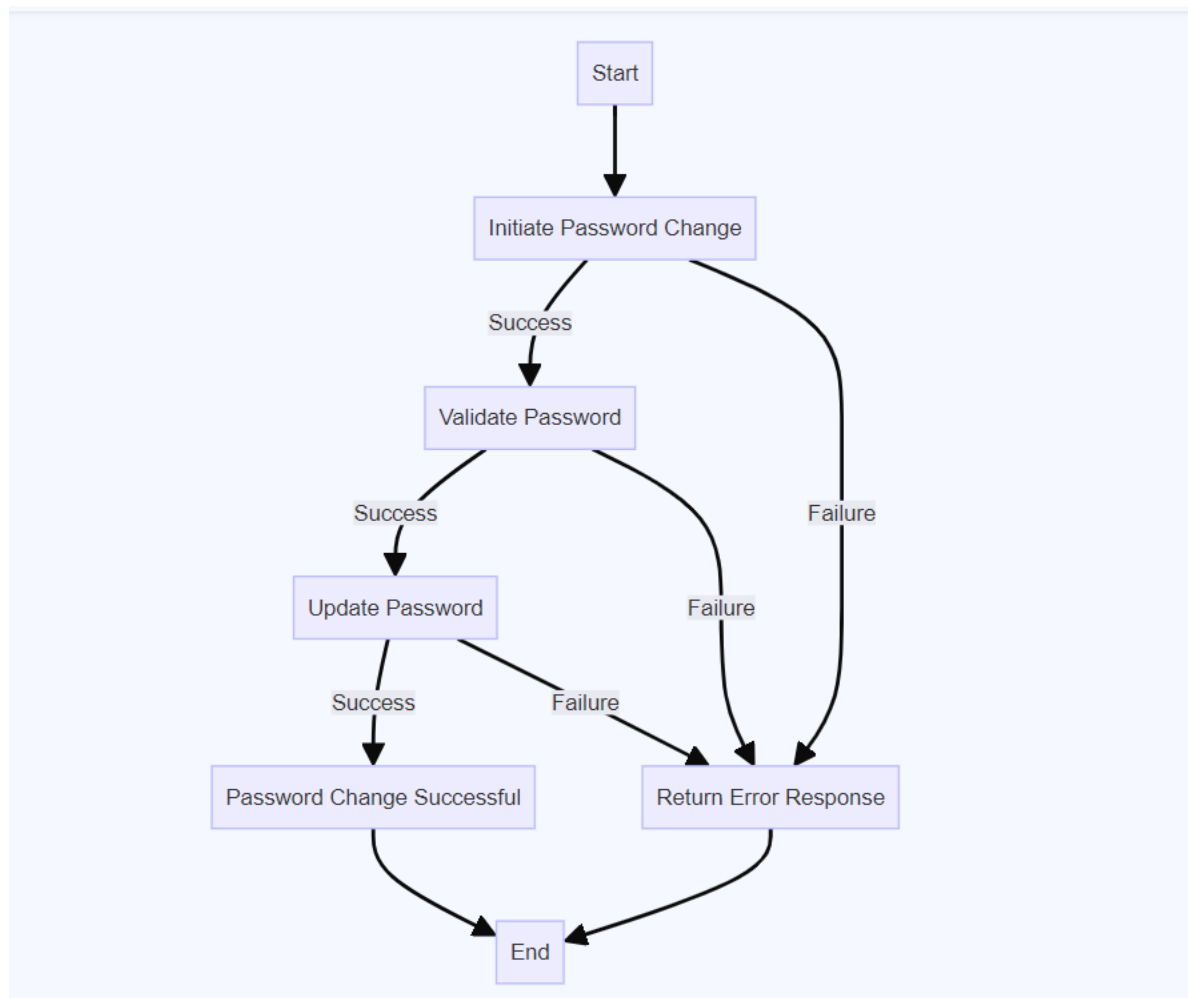
```
{
  "password":"ajagbeejo1",
  "confirmPassword":"ajagbeejo1",
  "email":"olanrewajuqudus5@gmail.com"
}
```

opt is sent to the user's phone number to verify the user before the change is made.

Forget password with otp

The sent otp is collected and validate before the change was made

```
3. @PutMapping("/changepassword")
```



Change password

```
{  
  "currentPassword":"Ojagbeejo1",  
  "password":"Ojagbeejo",  
  "confirmPassword":"Ojagbeejo"  
}
```

User can change password while logged in.

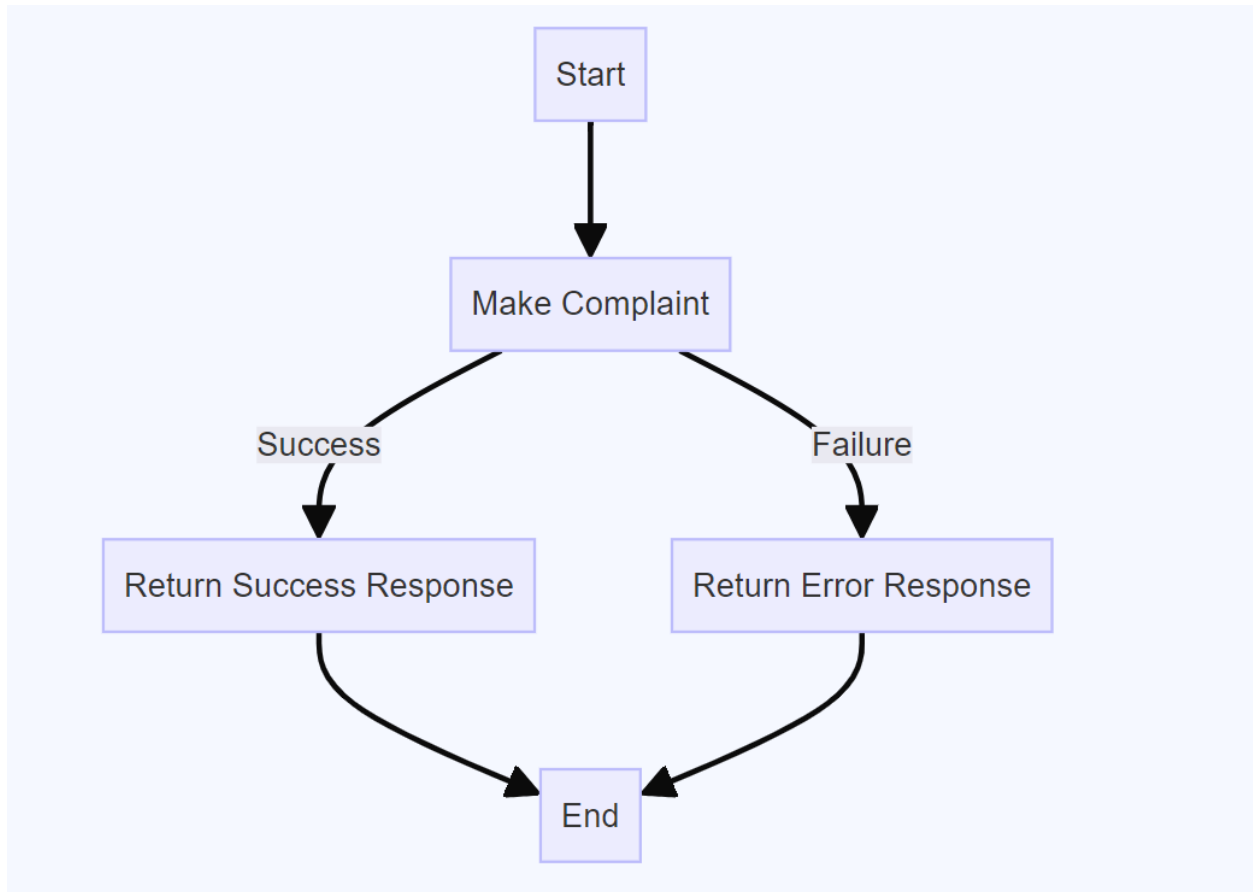
Change user role.

```
@PutMapping("/change-user-role")
```

Expect email of user and role to change to as a string params

Complaint

```
1. @PostMapping("/complaint")
```



Complaint

```
{  
  "location":"aja",  
  "crimeType":"theft",  
  "description":"wrist whatch "  
}
```

This is made by the user of the app.

Get highest complaint

```
1. @GetMapping("/highest-complaint")
```

Get all complaint

```
1. @GetMapping("/all-complaint")
```

Find complaint by data

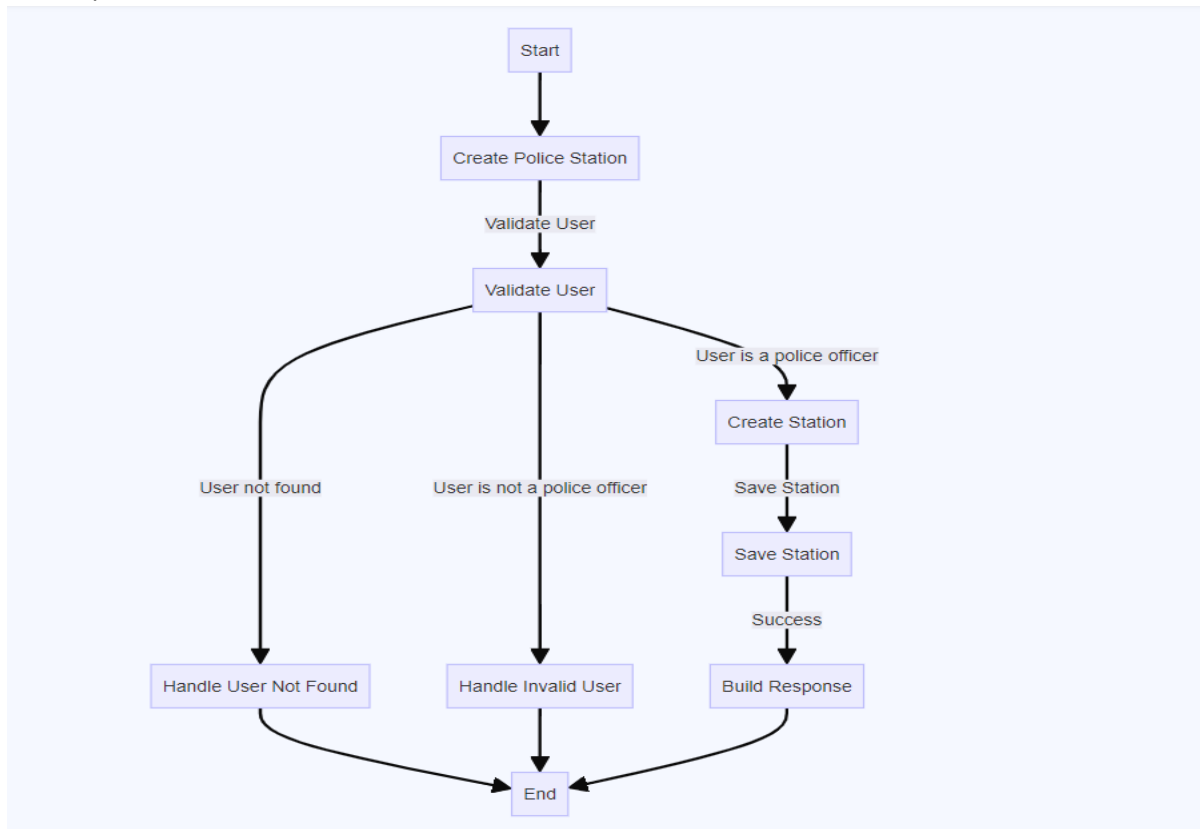
```
1. @GetMapping("/find-complaint-by-data")
```

The complaint can be get by any field of the complaint table.

Police

```
1. @PostMapping("/create-police-station/{userId}")
2.
```

Create police station

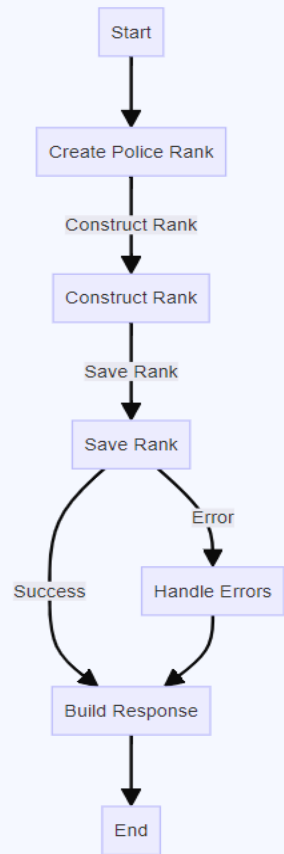


```
{
    "policeStationName":"alagbon",
    "location":"akala",
    "longitude":"1292891",
    "latitude":"7128291",
    "policeStationMaxStaffCapacity":"100",
    "currentCapacity":"0"
}
```

And user id of the head of police station.

Create rank

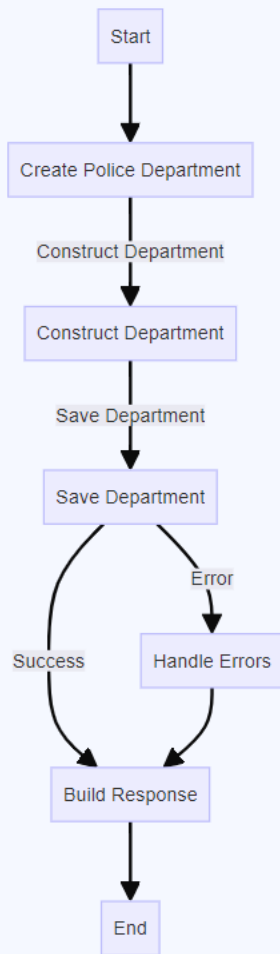
```
1. @PostMapping("/create-police-rank")
```



```
{  
  "rank": "Inspector General of Police (IGP)"  
}
```

Create police department

```
1. @PostMapping("/create-police-department")
```



```
{  
  "department":"legal"  
}
```

Update department by department id

```
1. @PutMapping("/update-police-department/{deptId}")
```

```
{  
    "department": "legal"  
}
```

Id the department to be changed.

Create occupation by department id

```
1. @PostMapping("/create-police-occupation/{deptId}")
```

Below data are passed through string param

Email: ,

Rank: ,

Station: ,

Occupation:

Find all police by rank

```
1. @GetMapping("/find-all-police-by-rank/{rankId}")
```

Rank id is expected to be provided as path variable.

Find all police officer in a single police station

```
1. @GetMapping("/find-all-police-by-station/{stationId}")
```

Police station id is expected to be provided as path variable.

Appoint police officer as a head of department

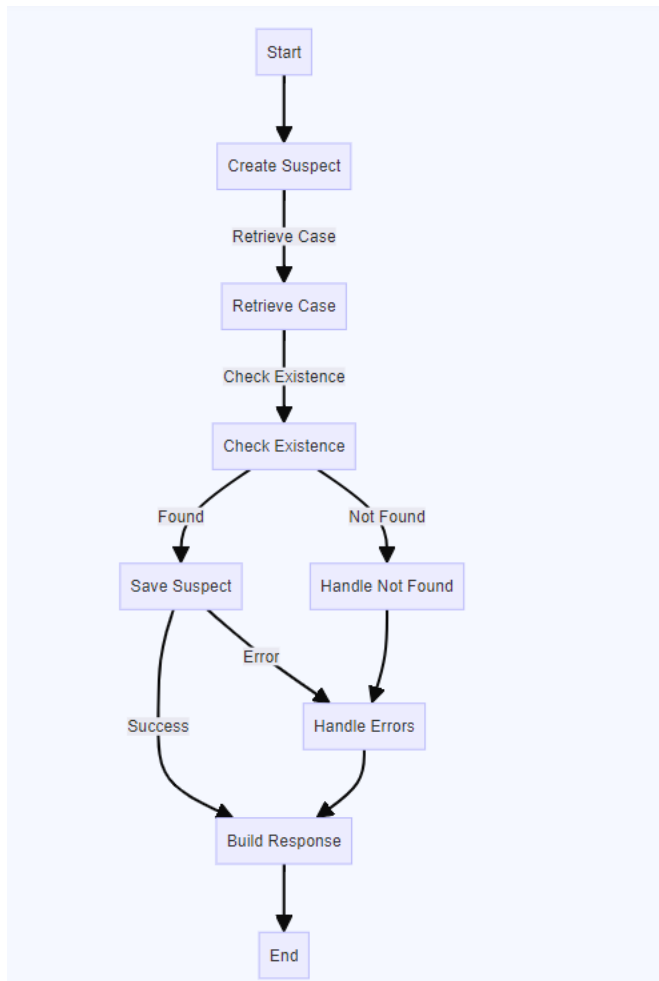
```
1. @PostMapping("/appoint-police-head-of-department/{userId}/{deptId}")
```

The officer id and the department id is expected to be provided as path variable.

Suspect of a crime

```
1. @PostMapping("/createSuspect/{caseId}")
```

Suspect of a crime can be created by getting the suspect data from a request body and the case id as a



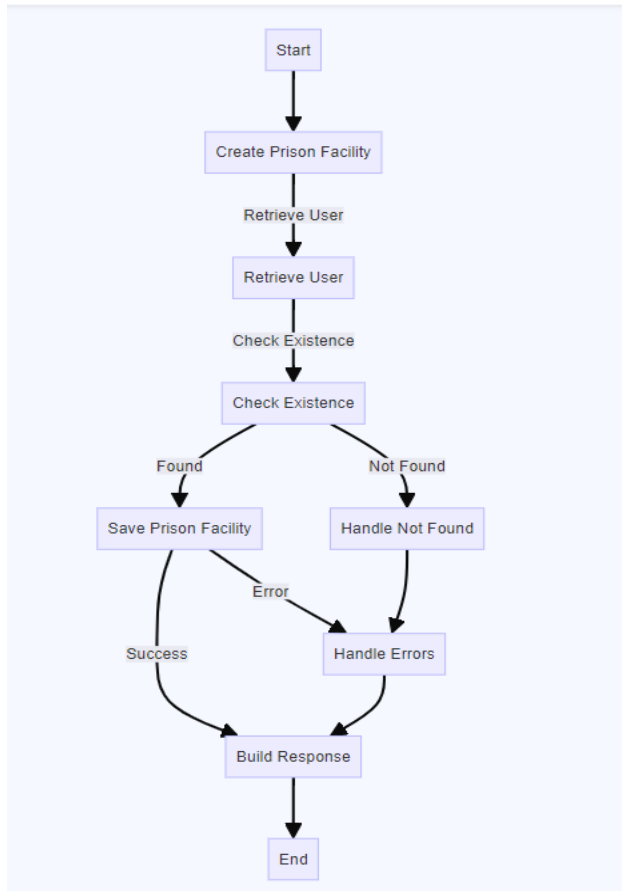
path variable, data expected is below.

```
{
  "firstName": "abiodun",
  "lastName": "olanrewaju",
  "dateOfBirth": "24-12-1992",
  "phoneNumber": "07035778452",
  "address": "oworo akala"
}
```

Prison

```
1. @PostMapping("/create-prison-facility/{userId}")
```

Create prison facility.



```
{  
  "prisonFacilityName": "gidaMai",  
  "location": "alkaleri",  
  "longitude": "1292841",  
  "latitude": "7128211",  
  "currentStaffCapacity": "0",  
  "prisonMaxStaffCapacity": "100",  
  "inmateMaxCapacity": "1000",  
  "currentInmateCapacity": "0"  
}
```

And user id of the head of prison facility.

Create prison rank

```
1. @PostMapping("/create-prison-rank")
{
    "rank":"Assistant Superintendent of Prisons I (ASP I)"
}
```

Create prison department

```
1. @PostMapping("/create-prison-department")
{
    "department":"legal"
}
```

Update department by department id

```
1. @PutMapping("/update-prison-department/{deptId}")
{
    "department":"legal"
}
```

Id the department to be changed.

Create occupation by department id

```
1. @PostMapping("/create-prison-occupation/{deptId}")
```

Below data are passed through string params

Email: ,

Rank: ,

Station: ,

Occupation:

Find all warden by rank

```
1. @GetMapping("/find-by-prison-rank/{rankId}")
```

Rank id is expected to be provided as path variable.

Find all warden in a single facility.

```
1. @GetMapping("/find-by-prison-facility/{facilityId}")
```

Police station id is expected to be provided as path variable.

Appoint warder as a head of department

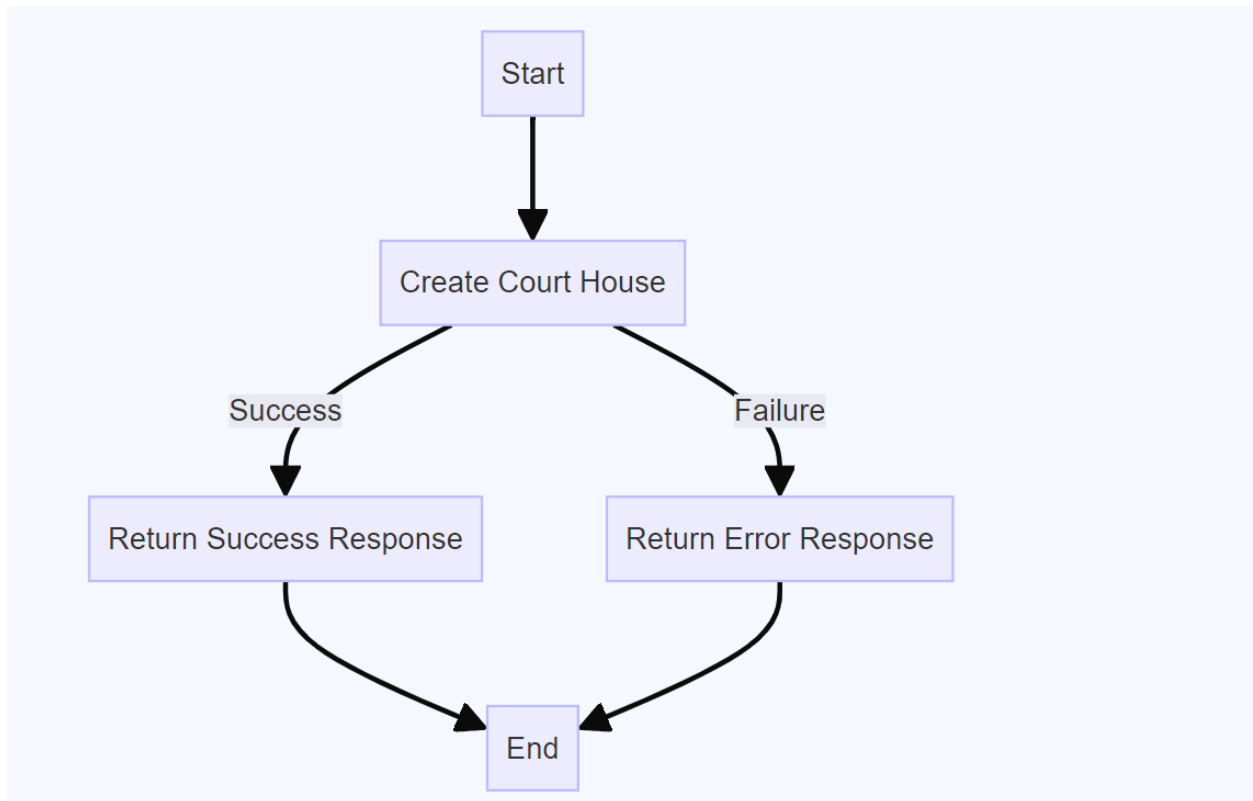
```
1. @PostMapping("/appoint-head-of-prison-department/{userId}/{deptId}")
```


The officer id and the department id are expected to be provided as path variable.

Court

```
1. @PostMapping("/create-court-House")
```

Create court house

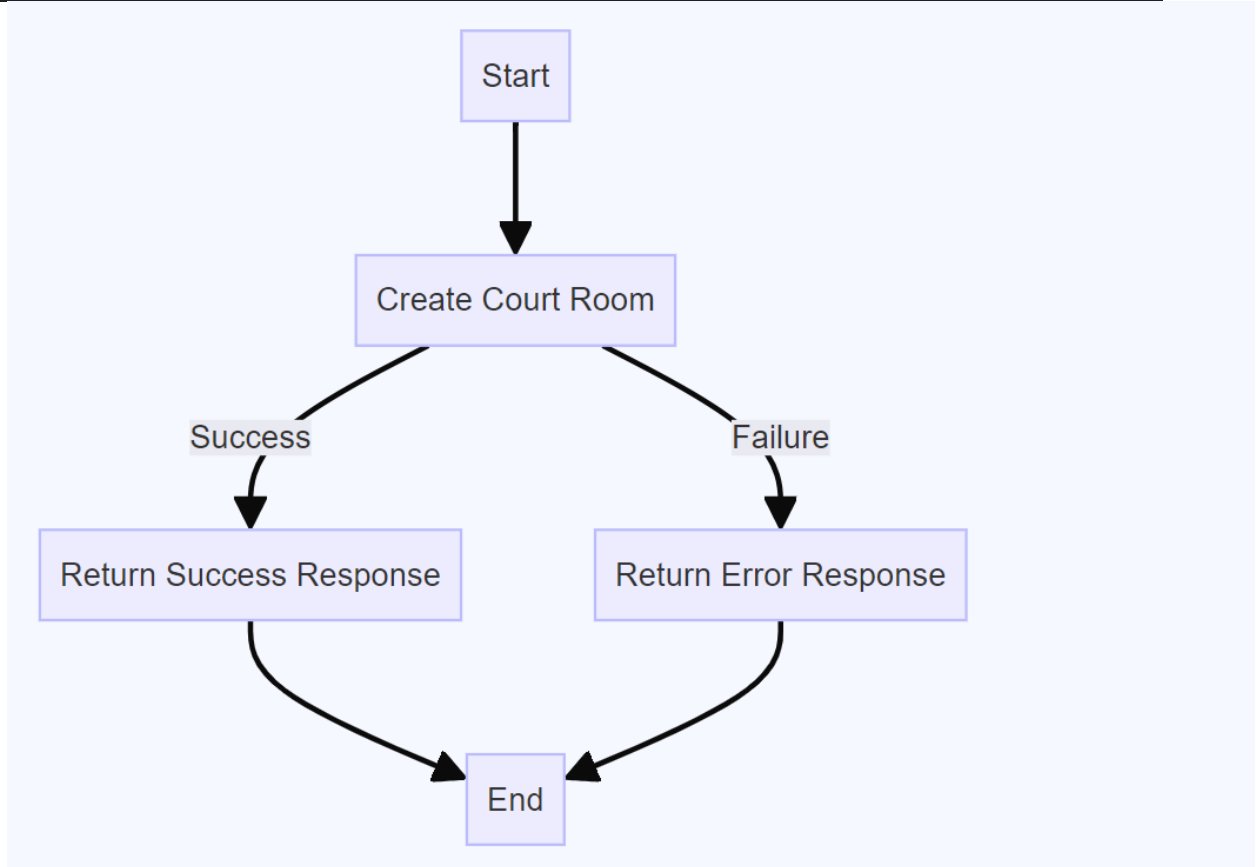


Expected as request body

```
{  
  "courtHouseName": "Ikorodu High court",  
  "location": "alagbon",  
  "longitude": "1292841",  
  "latitude": "7128211",  
  "numbersOfCourtRooms": "0",  
}
```

Create court room

```
1. @PostMapping("/create-court-room")
```



```
{  
  "roomNumber":"w63827"  
}
```

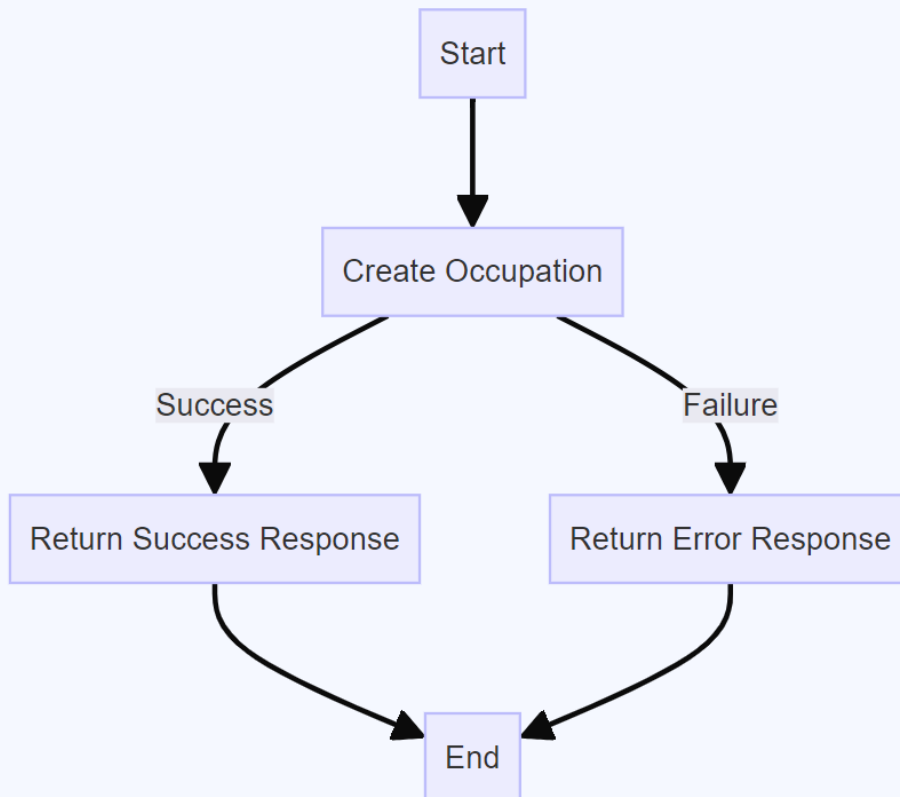
Assign case to judge

```
2. @PostMapping("/assign-case-to-judge/{roomId}/{caseId}")
```

roomId and caseId is passed as a path variable

Create occupation by court room id

```
1. @PostMapping("/create-occupation/{roomId}")
```



Below data are passed through string params

Email: ,

courtHouse: ,

Occupation:

Get all court house

```
1. @GetMapping("/all-court-house")
```

List of all court house are obtained.

Open case

```
1. @PostMapping("/save-to-match")
```

This is saved in the data base and used to later to match complaint by user in other to automatically open case by pattern matching.

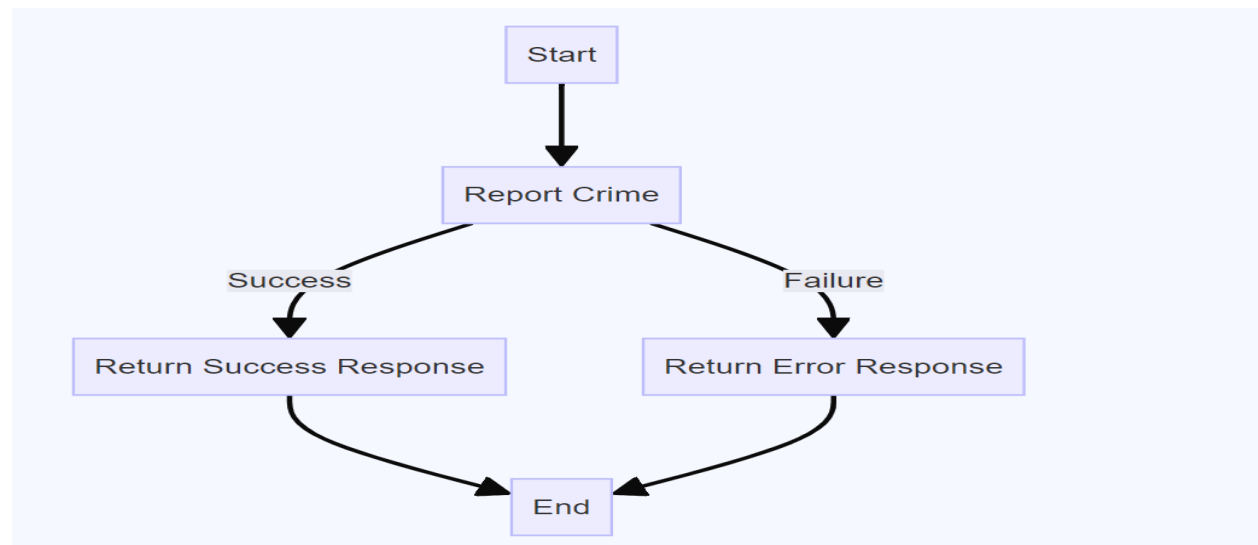
This obtained has request body

```
{  
  "toMatch":"killed"  
}
```

Crime

```
1. @PostMapping("/report-crime")
```

Report crime



This is sent as a multipart data

```
Data: {  
  "description":"the killer",  
  "location":"lagos",  
  "crimeType":"theft",  
  "date":"1-2-24",  
  "status":"open"  
}
```

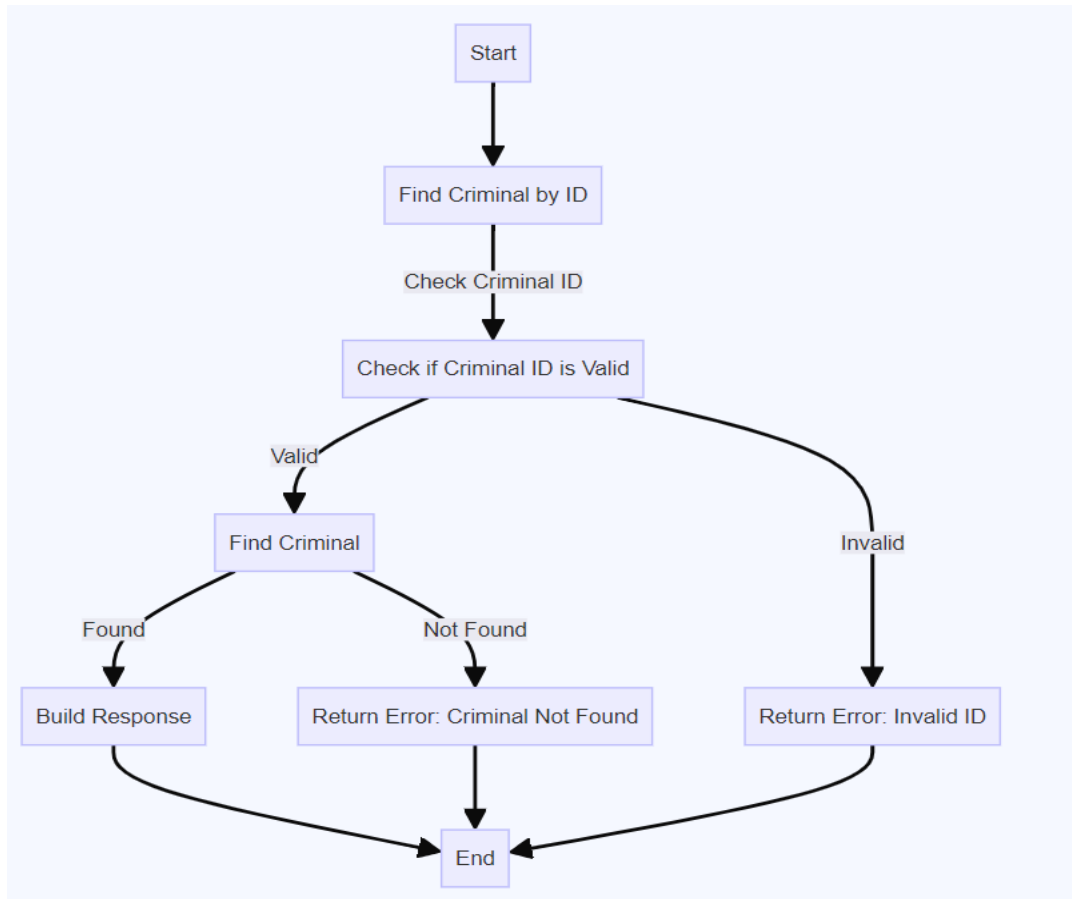
Image:

Video:

Find crime by crime id

```
1. @GetMapping("/find-crime/{crimeId}")
```

Crime id is passed as a path variable.



Find all crime by crime date

```
1. @GetMapping("/all-crime-by-date/{crimeDate}")
```

Crime date is pass as path variable.

Find all crime by crime location

```
1. @GetMapping("/all-crime-by-location/{crimeLocation}")
```

Crime location is pass as path variable.

Find all crime by crime type

```
1. @GetMapping("/all-crime-by-crime-type/{crimeType}")
```

Crime type is pass as path variable.

Find all crime by crime investigator

```
1. @GetMapping("/all-crime-by-investigator/{investigator}")
```

Crime investigatorId(userId) is pass as path variable.

Find all crime

```
1. @GetMapping("/all-crime")
```

Delete a crime by id

```
1. @DeleteMapping("/expunge-crime/{crimeId}")
```

Update a crime by id

```
1. @PutMapping("/update-crime-report/{crimeId}")
```

This is sent as a multipart data

Crime id is passed as a path variable

```
Data: {  
  "description": "the killer",  
  "location": "lagos",  
  "crimeType": "theft",  
  "date": "1-2-24",  
  "status": "open"
```

```
}
```

Image:

Video:

Criminal

```
1. @PostMapping("/save-criminal/{crimeId}")
```

Save criminal

This is expected as a multipart data

criminalImage: image,

criminalData: {

```
    "age": "27",
    "address": "adenuga",
    "firstName": "jane doe",
    "lastName": "jane doe",
    "eyeColor": "brown",
    "gender": "male",
    "contactNumber": "08098634548",
    "height": "1.57",
    "scars": "No",
    "nationality": "nija",
    "hairColor": "black",
    "dateOfBirth": "12/23/4",
    "tattoos": "No",
    "weight": "75"
}
```

Find criminal by id

```
1. @GetMapping("/find-criminal/{criminalId}")
```

Criminal id is expected as a path variable

Find criminal within a particular age range

```
1. @GetMapping("/find-criminal-age/{age}")
```

Age is expected as a path variable;

Find criminal by email

```
1. @GetMapping("/find-criminal-email/{email}")
```

Criminal email is expected as a path variable

Find criminal by age and gender

```
1. @GetMapping("/find-criminal-name-gender")
```

Criminal age and gender are expected as a string params

Delete criminal by email

```
1. @DeleteMapping("/expunge-criminal/{crimeId}")
```

Criminal email is expected as a path variable

Delete criminal by criminal id

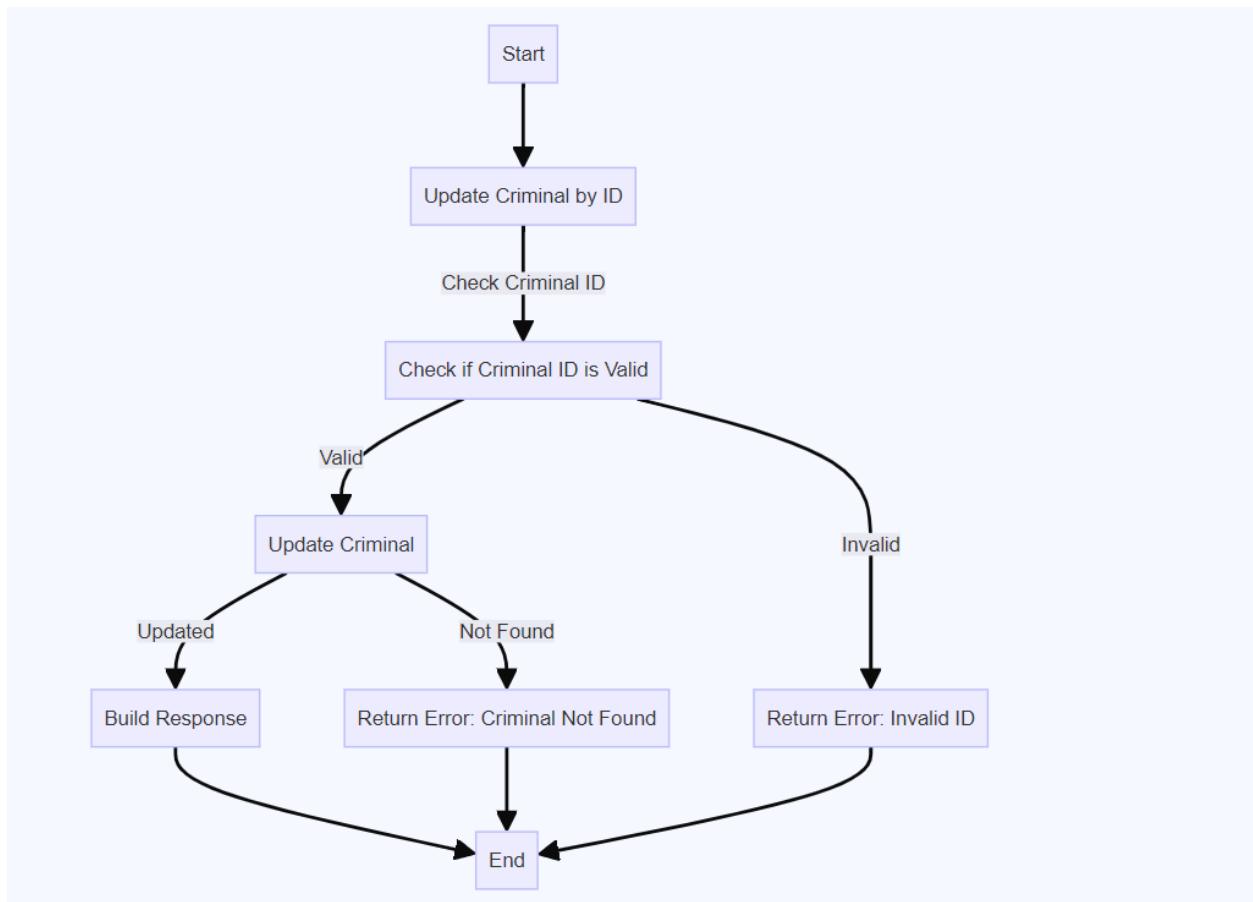
```
1. @DeleteMapping("/expunge-criminal-by-name/{email}")
```

Criminal id is expected as a path variable

update criminal by criminal id

```
1. @PutMapping(value = "/change-criminal/{crimeId}")
```

Criminal id is expected as a path variable and other as well



This is expected as a multipart data

criminalImage: image,

criminalData: {


```
"age": "27",
"address": "adenuga",
"firstName": "jane doe",
"lastName": "jane doe",
"eyeColor": "brown",
"gender": "male",
"contactNumber": "08098634548",
"height": "1.57",
"scars": "No",
"nationality": "nija",
"hairColor": "black",
"dateOfBirth": "12/23/4",
"tattoos": "No",
"weight": "75"
}
```

Get all judge

```
1. @GetMapping("/get-all-judge")
```

Charge case to court

```
1. @PostMapping("/charged-case/{caseId}")
```

This is done by the police legal department case id is expected as a path variable and courtHouseName as a String params.

Punishment

Save punishment

```
1. @PostMapping("/save-punishment/{chargedCaseId}")
```

Charged case id is expected as a path variable

Punishment is expected in this format

```
{
    "punishmentType": "fine",
    "startDate": "20-12-2024",
    "endDate": "21-12-2024",
    "punishmentDescription": "mush apy the fine",
}
```

```
"fineAmount":"255"
```

```
}
```

Punishment can only be any of the three (e.g., imprisonment, fine, community service)

```
1. @GetMapping("/find-all-by-punishment-type/{type}")
```

Fine amount is not required for other type

Find all punishment by type

Punishment type is expected as a path variable

Delete by punishment id

```
1. @DeleteMapping("/expunge-punishment/{punishmentId}")
```

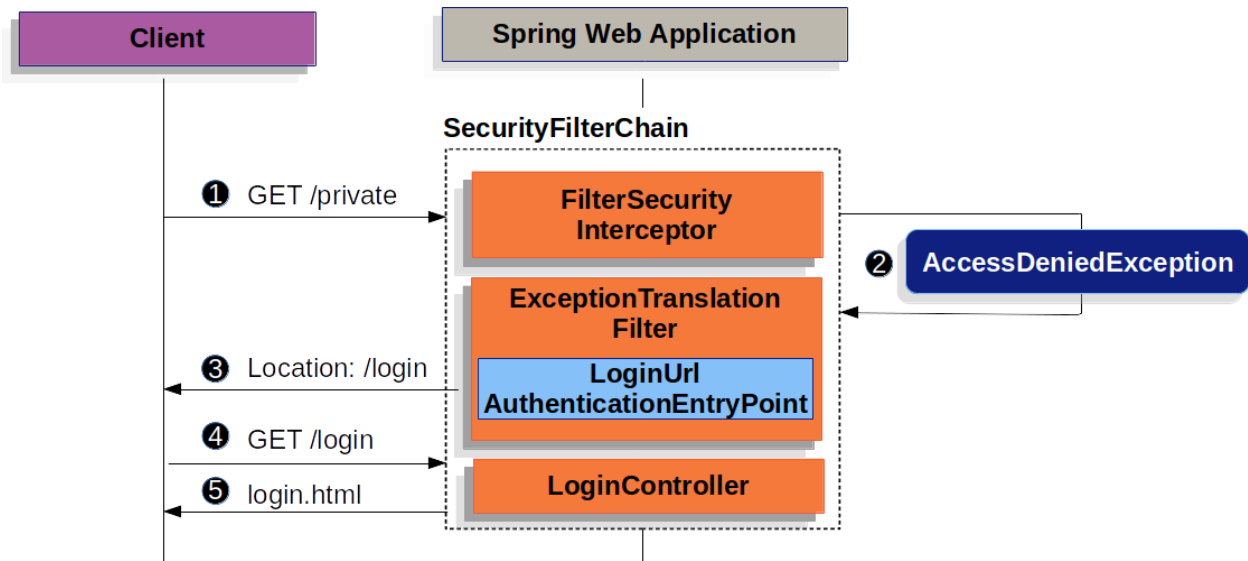
Punishment id is expected as a path variable.

Update punishment by id

```
1. @PutMapping("/change-punishment/{punishmentId}")
```

Punishment id is expected as path variable

This is the graphical representation of spring security implemented in this application.

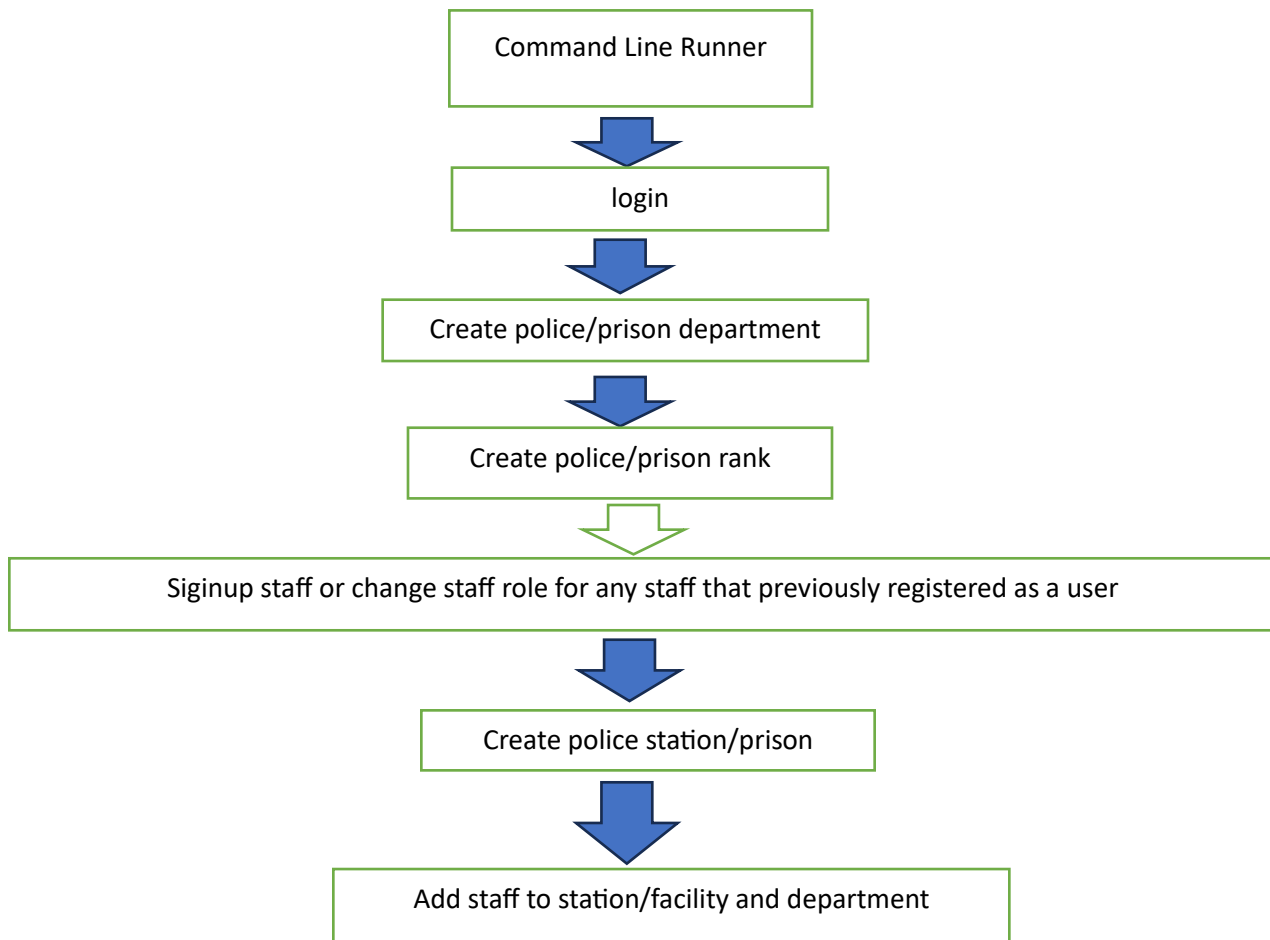


Source: <https://docs.spring.io/spring-security/site/docs/5.4.0/reference/html5/#servlet-securityfilterchain>

List of dependencies

- spring-boot-starter-mail
- spring-boot-starter-data-jpa
- org.projectlombok
- io.jsonwebtoken
- twilio
- spring-boot-devtools
- aws-java-sdk-s3
- spring-boot-starter-security
- spring-boot-starter-validation
- spring-boot-starter-web
- postgresql
- jackson-databind

USER FLOW CHART



USER FLOW CHART

