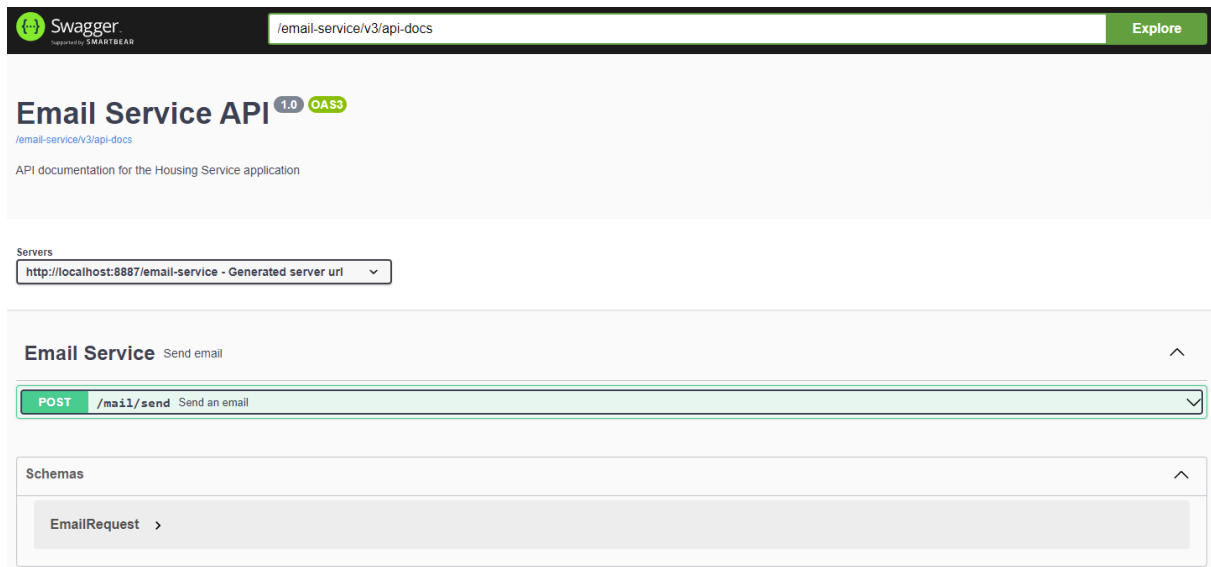


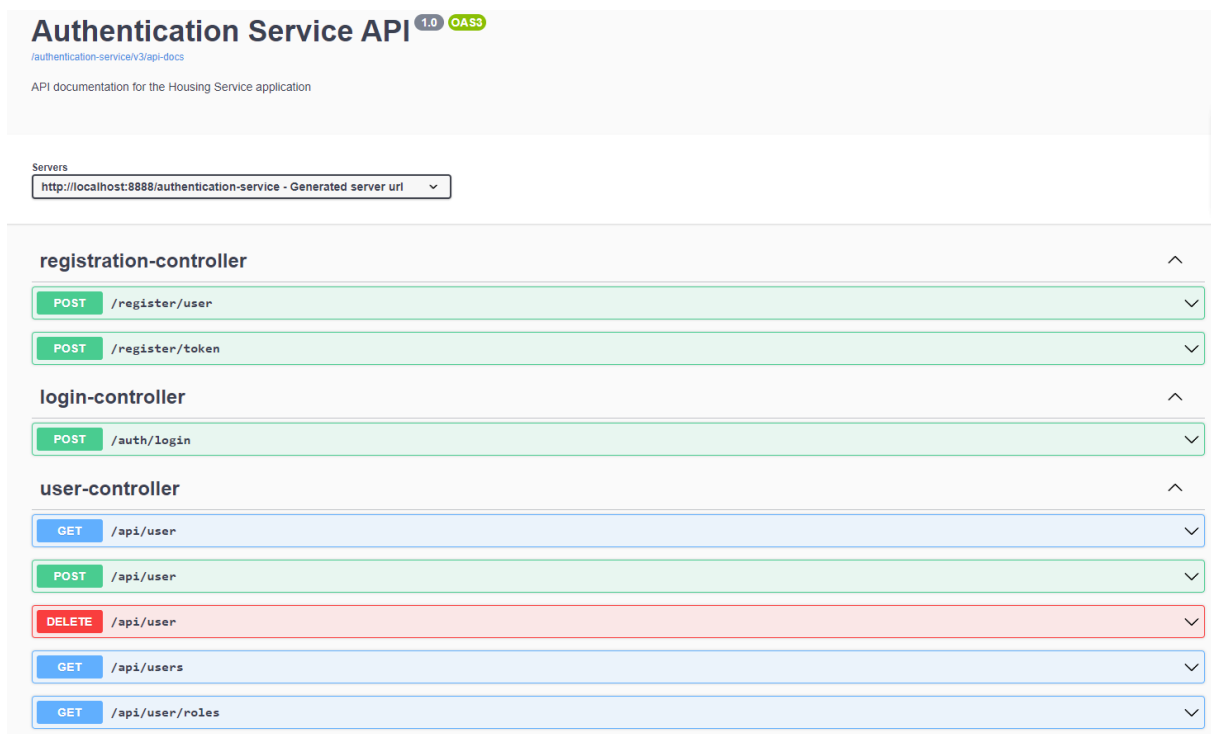
# Swagger ScreenShots

## Email Service



The screenshot shows the Swagger UI for the Email Service API. The top bar includes the Swagger logo, the URL `/email-service/v3/api-docs`, and an **Explore** button. The main header displays **Email Service API** with version **1.0** and **OAS3** status. Below this, the URL `/email-service/v3/api-docs` and a description "API documentation for the Housing Service application" are visible. A **Servers** dropdown menu shows `http://localhost:8887/email-service - Generated server url`. The **Email Service** section is expanded, showing a **POST** method for `/mail/send` with the description "Send an email". Below this, the **Schemas** section is expanded, showing the **EmailRequest** schema.

## Authentication Service



The screenshot shows the Swagger UI for the Authentication Service API. The top bar includes the Swagger logo, the URL `/authentication-service/v3/api-docs`, and an **Explore** button. The main header displays **Authentication Service API** with version **1.0** and **OAS3** status. Below this, the URL `/authentication-service/v3/api-docs` and a description "API documentation for the Housing Service application" are visible. A **Servers** dropdown menu shows `http://localhost:8888/authentication-service - Generated server url`. The **registration-controller** section is expanded, showing two **POST** methods: `/register/user` and `/register/token`. The **login-controller** section is expanded, showing one **POST** method: `/auth/login`. The **user-controller** section is expanded, showing four methods: `/api/user` (GET), `/api/user` (POST), `/api/user` (DELETE), `/api/users` (GET), and `/api/user/roles` (GET).

## Application Service

Servers

http://localhost:8081 - Generated server url

application-work-flow-controller	
POST	/api/submit
POST	/api/status
POST	/api/onboarding
GET	/api/status/{employeeId}
GET	/api/onboarding/{employeeId}
GET	/api/hr/status/{status}
GET	/api/application-workflows

## Employee Service

Servers

http://localhost:8080 - Generated server url

visa-status-controller	
PUT	/api/employees/visaStatus/{employeeId}
PUT	/api/employees/personalDocument/{employeeId}
employee-controller	
POST	/api/employees/createOrUpdate
GET	/api/employees
GET	/api/employees/{id}
GET	/api/employees/visa-status
GET	/api/employees/search
GET	/api/employees/profiles

## Visa Status Composite Service

Servers  
http://localhost:8180/VisaStatusCompositeService - Generated server url

### visa-status-controller

**POST** /composite/updateEmployeeDocument

Parameters Try it out

Name	Description
<b>employeeId</b> * required integer (int32) (query)	employeeId
<b>documentType</b> * required string (query)	documentType
<b>description</b> * required string (query)	description
<b>isRequired</b> * required boolean (query)	--

激活 Windows

## Housing Service

### Housing Service API <sup>1.0</sup> OAuth2

/housing-service/v3/api-docs

API documentation for the Housing Service application

Servers  
http://localhost:8082/housing-service - Generated server url

#### facility-report-controller

- PUT** /api/facility-reports/details/{detailId}
- DELETE** /api/facility-reports/details/{detailId}
- GET** /api/facility-reports
- POST** /api/facility-reports
- GET** /api/facility-reports/{id}/details
- POST** /api/facility-reports/{id}/details
- GET** /api/facility-reports/{id}

#### facility-controller

- GET** /api/facilities/{id}
- PUT** /api/facilities/{id}
- DELETE** /api/facilities/{id}
- GET** /api/facilities
- POST** /api/facilities

#### landlord-controller

- POST** /api/landlords

#### house-controller

- GET** /api/houses

GET

/api/facilities/{id}

▼

PUT

/api/facilities/{id}

▼

DELETE

/api/facilities/{id}

▼

GET

/api/facilities

▼

POST

/api/facilities

▼

landlord-controller

^

POST

/api/landlords

▼

house-controller

^

GET

/api/houses

▼

POST

/api/houses

▼

GET

/api/houses/{id}

▼

Schemas

^

FacilityReportDetailDTO >

FacilityDTO >

Landlord >

HouseDTO >

FacilityReportDTO >