Spring API

Overview

Api testing

Tags

Appointment

The Appointment APIs

Vehicle

The Vehicle APIs

MechanicalAction

The MechanicalAction APIs

Place

The Place APIs

Garage

The Garage APIs

Emails

The emails APIs

OpenDay

The OpenDay APIs

Paths

GET `/api/vehicles/{id}` Retrieve a Vehicle by Id

Get a Vehicle object by specifying its id. The response is Vehicle object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

PUT `/api/vehicles/{id}` Updates an existing vehicle

Updates an existing vehicle. The response is the updated Vehicle object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

DELETE `/api/vehicles/{id}` Deletes a vehicle

Deletes a vehicle.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK	No Links
404	Not Found	No Links

GET `/api/opendays/{id}` Retrieve a OpenDay by Id

Get a OpenDay object by specifying its id. The response is OpenDay object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

PUT `/api/opendays/{id}` Updates an existing open day

Updates an existing open day. The response is the updated OpenDay object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

DELETE `/api/opendays/{id}` Deletes a open day

Deletes a open day.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK	No Links
404	Not Found	No Links

GET `/api/mechanicalActions/{id}` Retrieve a MechanicalAction by Id

Get a MechanicalAction object by specifying its id. The response is MechanicalAction object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

PUT `/api/mechanicalActions/{id}` Updates an existing mechanicalAction

Updates an existing mechanicalAction. The response is the updated MechanicalAction object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

DELETE /api/mechanicalActions/{id} Deletes a mechanicalAction

Deletes a mechanicalAction.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK	No Links
404	Not Found	No Links

GET /api/garages/{id} Retrieve a Garage by Id

Get a Garage object by specifying its id. The response is Garage object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links

Code	Description	Links
404	Not Found	No Links

PUT `/api/garages/{id}` Updates an existing garage

Updates an existing garage. The response is the updated Garage object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

DELETE `/api/garages/{id}` Deletes a garage

Deletes a garage.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK	No Links
404	Not Found	No Links

GET `/api/appointments/{id}` Retrieve a Appointment by Id

Get a Appointment object by specifying its id. The response is Appointment object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

PUT `/api/appointments/{id}` Updates an existing appointment

Updates an existing appointment. The response is the updated Appointment object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

DELETE `/api/appointments/{id}` Deletes a appointment

Deletes a appointment.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Code	Description	Links
500	Internal Server Error	No Links
200	OK	No Links
404	Not Found	No Links

PUT `/api/appointments/vote/{id}` Modify vote

Modify vote

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK	No Links
404	Not Found	No Links

DELETE `/api/appointments/vote/{id}` Delete vote

Delete vote

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK	No Links
404	Not Found	No Links

GET `/api/vehicles` Retrieve all Vehicles

Get all Vehicles objects. The response is a list of Vehicles objects.

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

POST `/api/vehicles` Create a new vehicle

Creates a new vehicle. The response is the created Vehicle object.

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

POST `/api/user/changePassword`

Responses

Code	Description	Links
200	OK *Content* /	No Links

GET `/api/opendays` Retrieve all OpenDays

Get all OpenDays objects. The response is a list of OpenDays objects.

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links

Code	Description	Links
404	Not Found	No Links

POST `/api/opendays` Create a new open day

Creates a new open day. The response is the created OpenDay object.

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

POST `/api/opendays/filterDays` Retrieve a OpenDay by Garage Id and Days

Get a OpenDay object by specifying its garage id and days. The response is OpenDay object.

Parameters

Type	Name	Description	Schema
query	**sortDirection** *optional*		string
query	**size** *optional*		integer (int32)
query	**sortField** *optional*		string
query	**page** *optional*		integer (int32)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

GET `/api/mechanicalActions` Retrieve all MechanicalActions

Get all MechanicalActions objects. The response is a list of MechanicalActions objects.

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

POST `/api/mechanicalActions` Create a new mechanicalAction

Creates a new mechanicalAction. The response is the created MechanicalAction object.

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

GET `/api/garages` Retrieve all Garages

Get all Garages objects. The response is a list of Garages objects.

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

POST `/api/garages` **Create a new garage**

Creates a new garage. The response is the created Garage object.

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

POST `/api/auth/signup`

Responses

Code	Description	Links
200	OK *Content* /	No Links

POST `/api/auth/refreshToken`

Parameters

Type	Name	Description	Schema
query	**refreshToken** *required*		string

Responses

Code	Description	Links
200	OK *Content* /	No Links

POST `/api/auth/recoveryPassword`

Parameters

Type	Name	Description	Schema
query	**email** *required*		string

Responses

Code	Description	Links
200	OK *Content* /	No Links

POST /api/auth/login

Responses

Code	Description	Links
200	OK *Content* /	No Links

POST /api/auth/creaUtenteIniziale

Responses

Code	Description	Links
200	OK *Content* /	No Links

GET /api/appointments **Retrieve all Appointments**

Get all Appointments objects. The response is a list of Appointments objects.

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

POST /api/appointments Create a new appointment

Creates a new appointment. The response is the created Appointment object.

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

POST /api/appointments/{appointmentId}/state/{status} Update the appointment state

Update the appointment state. The response is the updated Appointment object. The status can be CONFIRMED, FINISHED, REJECTED

Parameters

Type	Name	Description	Schema
path	**appointment Id** *required*		string
path	**status** *required*		string

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

POST /api/appointments/vote Vote appointment

Vote appointment

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK	No Links
404	Not Found	No Links

POST /api/appointments/customAppointment **Handle custom technical evaluation**

Handle custom technical evaluation. The response is a list of Appointments objects.

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

POST /api/appointments/availableTimeSlots **Retrieve all available Appointments**

Get all available Appointments objects. The response is a list of Appointments objects.

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

GET /api/vehicles/user/{id} **Retrieve a list of Vehicles by User Id**

Get a list of Vehicles objects by specifying its user id. The response is a list of Vehicle objects.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

GET /api/user/me

Parameters

Type	Name	Description	Schema
query	**userPrincipal** *required*		UserPrincipal

Responses

Code	Description	Links
200	OK *Content* /	No Links

GET /api/public/test

Responses

Code	Description	Links
200	OK	No Links

GET /api/public/testRemoveFromCalendar

Responses

Code	Description	Links
200	OK	No Links

GET /api/public/testAddToCalendar

Responses

Code	Description	Links
200	OK	No Links

GET /api/public/populate

Responses

Code	Description	Links
200	OK	No Links

GET /api/places **Retrieve all Places**

Get all Places objects. The response is a list of Places objects.

Parameters

Type	Name	Description	Schema
query	**size** *optional*		integer (int32)
query	**page** *optional*		integer (int32)

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

GET /api/places/{id} **Retrieve a Place by Id**

Get a Place object by specifying its id. The response is Place object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK

Content

application/json | No Links |
| 404 | Not Found | No Links |

DELETE `/api/places/{id}` Deletes a place

Deletes a place.

Parameters

Type	Name	Description	Schema
path	**id**
required		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK	No Links
404	Not Found	No Links

GET `/api/places/findPlaceByRegionStartsWith` Get the place by region starts with the specified string

Get the place by region starts with the specified string

Parameters

Type	Name	Description	Schema
query	**region**
required		string

Responses

Code	Description	Links
200	OK

Content

application/json | No Links |
| 500 | Internal Server Error | No Links |

Code	Description	Links
404	Not Found	No Links

GET /api/places/findPlaceByProvinceStartsWith Get the place by province starts with the specified string

Get the place by province starts with the specified string

Parameters

Type	Name	Description	Schema
query	**province** *required*		string

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

GET /api/places/findPlaceByMunicipalityStartsWith Get the place by municipality starts with the specified string

Get the place by municipality starts with the specified string

Parameters

Type	Name	Description	Schema
query	**municipality** *required*		string

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links

Code	Description	Links
404	Not Found	No Links

GET /api/opendays/garage/{id} Retrieve a OpenDay by Garage Id

Get a OpenDay object by specifying its garage id. The response is OpenDay object.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

GET /api/garages/findGarageByPlaceRegionStartsWith Retrieve Garages by region starts with

Get the garages by place region starts with. The response is a list of garages by region starts with.

Parameters

Type	Name	Description	Schema
query	**region** *required*		string

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

GET /api/garages/findGarageByPlaceProvinceStartsWith
Retrieve Garages by province starts with

Get the garages by place province starts with. The response is a list of garages by province starts with.

Parameters

Type	Name	Description	Schema
query	**province** *required*		string

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

GET /api/garages/findGarageByPlaceMunicipalityStartsWith
Retrieve Garages by municipality starts with

Get the garages by place municipality starts with. The response is a list of garages by municipality starts with.

Parameters

Type	Name	Description	Schema
query	**municipality** *required*		string

Responses

Code	Description	Links
200	OK *Content* **application/json**	No Links
500	Internal Server Error	No Links
404	Not Found	No Links

GET /api/emails/sendNewAppointmentMail/{appointmentId}
Email the customer with the just created appointment details .

Parameters

Type	Name	Description	Schema
path	**appointment Id** *required*		integer (int64)

Responses

Code	Description	Links
200	OK	No Links

GET /api/emails/sendFinishedAppointmentData/{appointmentId}
Email the customer with the finished appointment details

Parameters

Type	Name	Description	Schema
path	**appointment Id** *required*		integer (int64)

Responses

Code	Description	Links
200	OK	No Links

GET /api/emails/sendDeletedAppointmentData/{appointmentId}
Email the customer with the deleted appointment details

Parameters

Type	Name	Description	Schema
path	**appointment Id** *required*		integer (int64)

Responses

Code	Description	Links
200	OK	No Links

GET /api/emails/sendCustomAppointmentRejectedMail/{appointmentId} Email the customer with the appointment details for the appointment rejection.

Parameters

Type	Name	Description	Schema
path	**appointment Id** *required*		integer (int64)

Responses

Code	Description	Links
200	OK	No Links

GET /api/emails/sendCustomAppointmentApprovedMail/{appointmentId} Email the customer with the appointment details for the appointment confirmation.

Parameters

Type	Name	Description	Schema
path	**appointment Id** *required*		integer (int64)

Responses

Code	Description	Links
200	OK	No Links

GET /api/auth/tokenResetPassword

Parameters

Type	Name	Description	Schema
query	**token** *required*		string

Responses

Code	Description	Links
200	OK *Content* /	No Links

GET /api/auth/session

Parameters

Type	Name	Description	Schema
query	**userPrincipal** *required*		UserPrincipal

Responses

Code	Description	Links
200	OK *Content* /	No Links

GET /api/appointments/myappointments Retrieve all Appointments by user principal

Get all Appointments objects by user principal. The response is a list of Appointments objects.

Parameters

Type	Name	Description	Schema
query	**userPrincipal** *required*		UserPrincipal

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

GET `/api/appointments/appointmentsUser/{id}` Retrieve all Appointments by user id

Get all Appointments objects by user id. The response is a list of Appointments objects.

Parameters

Type	Name	Description	Schema
path	**id** *required*		integer (int64)

Responses

Code	Description	Links
500	Internal Server Error	No Links
200	OK *Content* **application/json**	No Links
404	Not Found	No Links

Components

Schemas

VehicleDto

Properties

Name	Description	Schema
plate *optional*	Unique identifier of the Vehicle.	string
model *optional*	Model of the Vehicle.	string
brand *optional*	Brand of the Vehicle.	string
year *optional*	Year of the Vehicle.	integer (int32)

Name	Description	Schema
fuel *optional*	Fuel of the Vehicle.	string
isActive *optional*	Is the Vehicle active.	boolean
userId *optional*	User id of the Vehicle.	integer (int64)

Appointment

Properties

Name	Description	Schema
id *optional*		integer (int64)
openDay *optional*		OpenDay
mechanicalAction *optional*		MechanicalAction
comment *optional*		string
vehicle *optional*		Vehicle
status *optional*		enum (AWAITING_APPROVAL,FINISHED,CONFIRMED,REJECTED)
externalTime *optional*		TimePeriod
internalTime *optional*		TimePeriod
price *optional*		number (double)
idCalendarEvent *optional*		string
votes *optional*		< Vote > array
isMechanicalActionCustom *optional*		boolean
startTime *optional*		LocalTime

DayPlan

Properties

Name	Description	Schema
workingHours *optional*		TimePeriod
breaks *optional*	The list of breaks	< TimePeriod > array

Garage

Properties

Name	Description	Schema
id *optional*		integer (int64)
name *optional*		string
place *optional*		Place
address *optional*		string
cap *optional*		string
linkGoogleMaps *optional*		string
latitude *optional*		number (double)
longitude *optional*		number (double)
phone *optional*		string
email *optional*		string
website *optional*		string
logo *optional*		string
openDay *optional*		< OpenDay > array
votes *optional*		< Vote > array

LocalTime

Properties

Name	Description	Schema
hour *optional*		integer (int32)

Name	Description	Schema
minute *optional*		integer (int32)
second *optional*		integer (int32)
nano *optional*		integer (int32)

LocalTimeWrapper

The end time

Properties

Name	Description	Schema
hour *optional*		integer (int32)
minute *optional*		integer (int32)
second *optional*		integer (int32)
nano *optional*		integer (int32)

MechanicalAction

Properties

Name	Description	Schema
id *optional*		integer (int64)
name *optional*		string
description *optional*		string
price *optional*		number (double)
isActive *optional*		boolean

Name	Description				Schema
internalDuration *optional*	*Properties*				object
	Name	**Description**		**Schema**	
	seconds *optional*			integer (int64)	
	nano *optional*			integer (int32)	
	negative *optional*			boolean	
	zero *optional*			boolean	
	units *optional*			< object > array	
externalDuration *optional*	*Properties*				object
	Name	**Description**		**Schema**	
	seconds *optional*			integer (int64)	
	nano *optional*			integer (int32)	
	negative *optional*			boolean	
	zero *optional*			boolean	
	units *optional*			< object > array	

OpenDay

Properties

Name	Description	Schema
id *optional*		integer (int64)
garage *optional*		Garage
date *optional*		string (date)
maxParallelAppointments *optional*		integer (int32)
appointments *optional*		< Appointment > array

Name	Description	Schema
workPlan *optional*		DayPlan

Place

Properties

Name	Description	Schema
id *optional*		integer (int64)
istat *optional*		integer (int32)
municipality *optional*		string
province *optional*		string
region *optional*		string
garage *optional*		< Garage > array

SimpleGrantedAuthority

Properties

Name	Description	Schema
authority *optional*		string

TimePeriod

The list of breaks

Properties

Name	Description	Schema
start *optional*		LocalTimeWrapper
end *optional*		LocalTimeWrapper

User

Properties

Name	Description	Schema
id *optional*		integer (int64)

Name	Description	Schema
name *optional*		string
email *optional*		string
imageUrl *optional*		string
emailVerified *optional*		boolean
provider *required*		enum (local,facebook ,google,github)
providerId *optional*		string
votes *optional*	*unique items*	< Vote > array
vehicle *optional*		< Vehicle > array
authorities *optional*		< SimpleGranted Authority > array

Vehicle

Properties

Name	Description	Schema
id *optional*		integer (int64)
plate *optional*		string
model *optional*		string
brand *optional*		string
year *optional*		integer (int32)
fuel *optional*		string
isActive *optional*		boolean
user *optional*		User
reservation *optional*		< Appointment > array

Vote

Properties

Name	Description	Schema
id *optional*		integer (int64)
rating *optional*		integer (int32)
comment *optional*		string
user *optional*		User
garage *optional*		Garage
appointment *optional*		Appointment

OpenDayDto

Properties

Name	Description	Schema
garageId *optional*	The garage id	integer (int64)
date *optional*	The date of the open day	string (date)
maxParallelAppointments *optional*	The max number of parallel appointments	integer (int32)
workPlan *optional*		DayPlan

DurationWrapper

The external duration of the action

Properties

Name	Description	Schema
hours *optional*		integer (int32)
minutes *optional*		integer (int32)

MechanicalActionDto

Properties

Name	Description	Schema
name *optional*	The name of the action	string
description *optional*	The description of the action	string

Name	Description	Schema
price *optional*	The price of the action	number (double)
isActive *optional*	If it is active or not	boolean
internalDuration *optional*		[DurationWrapper](#)
externalDuration *optional*		[DurationWrapper](#)

GarageDto

Properties

Name	Description	Schema
name *required*	The garage name	string
placeId *required*	The linked Place	integer (int64)
address *required*	The address of the garage	string
cap *required*	The CAP of the garage address	string
linkGoogleMaps *required*	Link to Google Maps position	string
latitude *required*	The latitude of the garage	number (double)
longitude *required*	The longitude of the garage	number (double)
phone *required*	The phone number of the garage	string
email *required*	The email of the garage	string
website *optional*	The website of the garage	string
logo *optional*	The logo of the garage	string

AppointmentDto

Properties

Name	Description	Schema
openDayId *optional*	The open day id	integer (int64)
mechanicalActionId *optional*	The mechanical action id	integer (int64)

Name	Description	Schema
comment *optional*	The comment	string
vehicleId *optional*	The vehicle id	integer (int64)
timeSlotSelected *optional*		TimePeriod
status *optional*	The status of the appointment	string
isMechanicalActionCustom *optional*	Is mechanical action custom	boolean

AppointmentVote

Properties

Name	Description	Schema
id *optional*		integer (int64)
rating *optional*	**Maximum**: 5 **Minimum**: 0	integer (int32)
comment *optional*		string
appointmentId *optional*		integer (int64)

GrantedAuthority

Properties

Name	Description	Schema
authority *optional*		string

UserPrincipal

Properties

Name	Description	Schema
id *optional*		integer (int64)
email *optional*		string
password *optional*		string

Name	Description	Schema
authorities *optional*		< [GrantedAuthority](#) > array
attributes *optional*		object
name *optional*		string
enabled *optional*		boolean
username *optional*		string
accountNonExpired *optional*		boolean
credentialsNonExpired *optional*		boolean
accountNonLocked *optional*		boolean

FilterDay

Properties

Name	Description	Schema
dateFrom *optional*		string (date)
dateTo *optional*		string (date)
garageId *optional*		integer (int64)

SignUpRequestDto

Properties

Name	Description	Schema
name *required*		string
email *required*		string
password *required*		string

LoginRequestDto

Properties

Name	Description	Schema
email *required*		string
password *required*		string

AuthResponseDto

Properties

Name	Description	Schema
id *optional*		integer (int64)
email *optional*		string
name *optional*		string
role *optional*		< Role > array
token *optional*		string
refreshToken *optional*		string
duration *optional*		string

Role

Properties

Name	Description	Schema
id *optional*		integer (int64)
name *optional*		string

CustomAppointmentEvaluation

Properties

Name	Description	Schema
appointmentId *optional*	The id of the appointment	integer (int64)
internalTime *optional*		TimePeriod
externalTime *optional*		TimePeriod

Name	Description	Schema
price *optional*	The price of the appointment	number (double)
approved *optional*		boolean

AppointmentSearchDto

Properties

Name	Description	Schema
opendayId *optional*		integer (int64)
mechanicalActionId *optional*		integer (int64)
externalTimeslot *optional*		boolean