PROJET Microservice : TerrainSpot app

APERÇU DU PROJET	
RÉSUMÉ	Notre projet vise à concevoir et développer un système complet de réservation et de gestion de terrain, utilisant une architecture microservices pour garantir la flexibilité, la scalabilité et la résilience du système. L'application permettra aux utilisateurs de recherche et reserver des terrains, de rechercher des produits et passer des commandes et aux administrateurs de gérer efficacement les informations sur les différents éléments .
OBJECTIFS	Automatiser le processus de réservation de terrain, fournir des informations en temps réel sur les événements et les produits , et assurer la gestion efficace des réservations et des commandes.

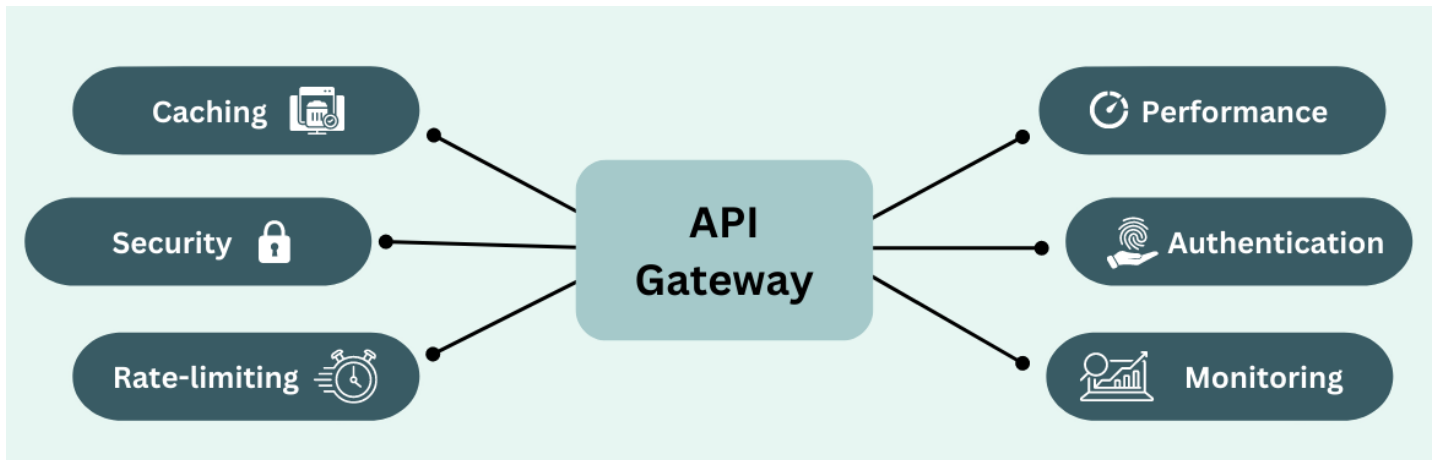| APPROBATION DES NOMS ET DES SIGNATURES DES PARTIES		
CLIENT		**MEMBRE DE PROJET**
Mme.Sarra Abidi		MohamedAmine Abdessayed roua mnissi rajhi zeineb sonia chalouah Eya Jomli Bayouli Oussama

TerrainSpotApp architecture
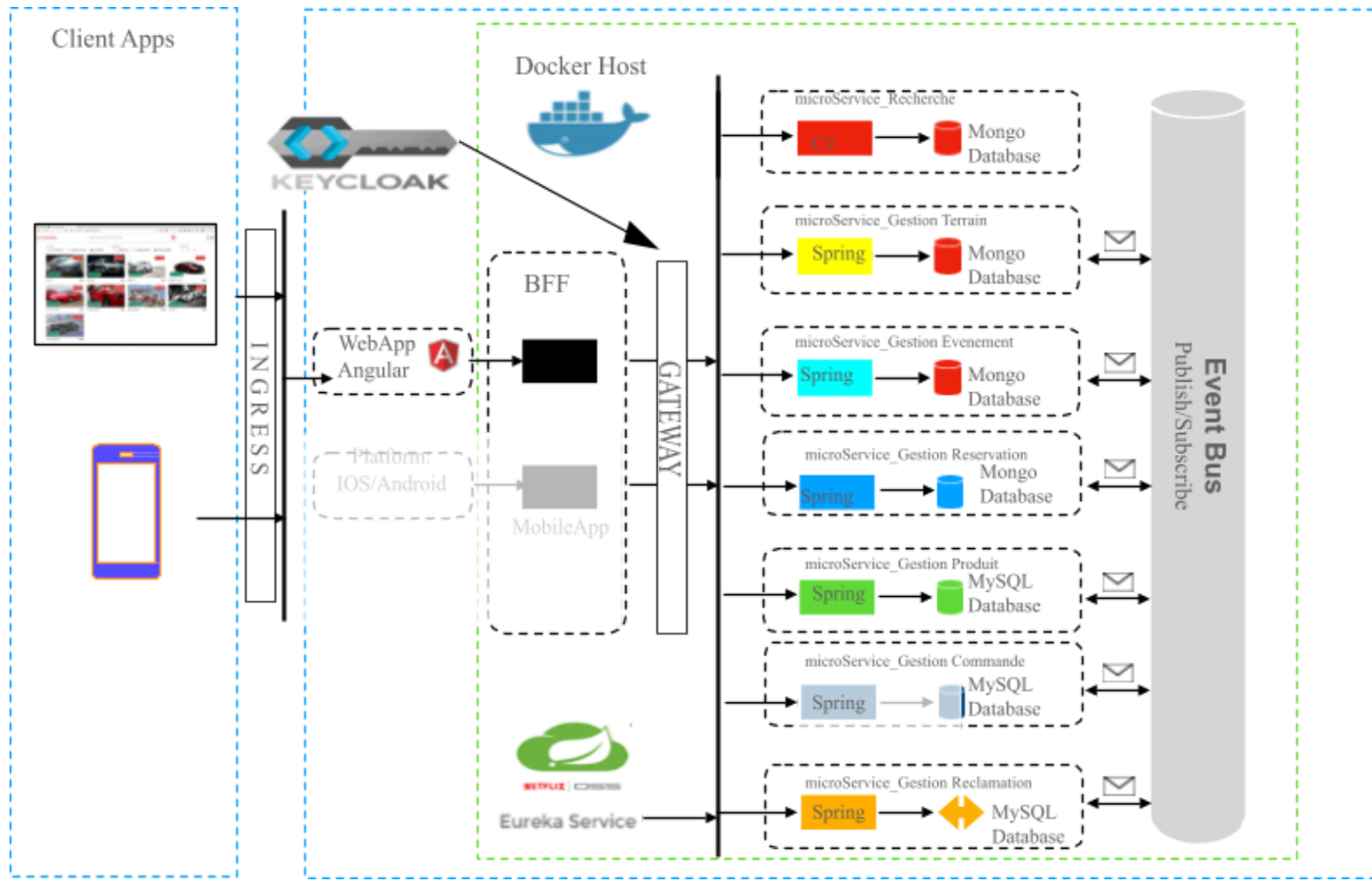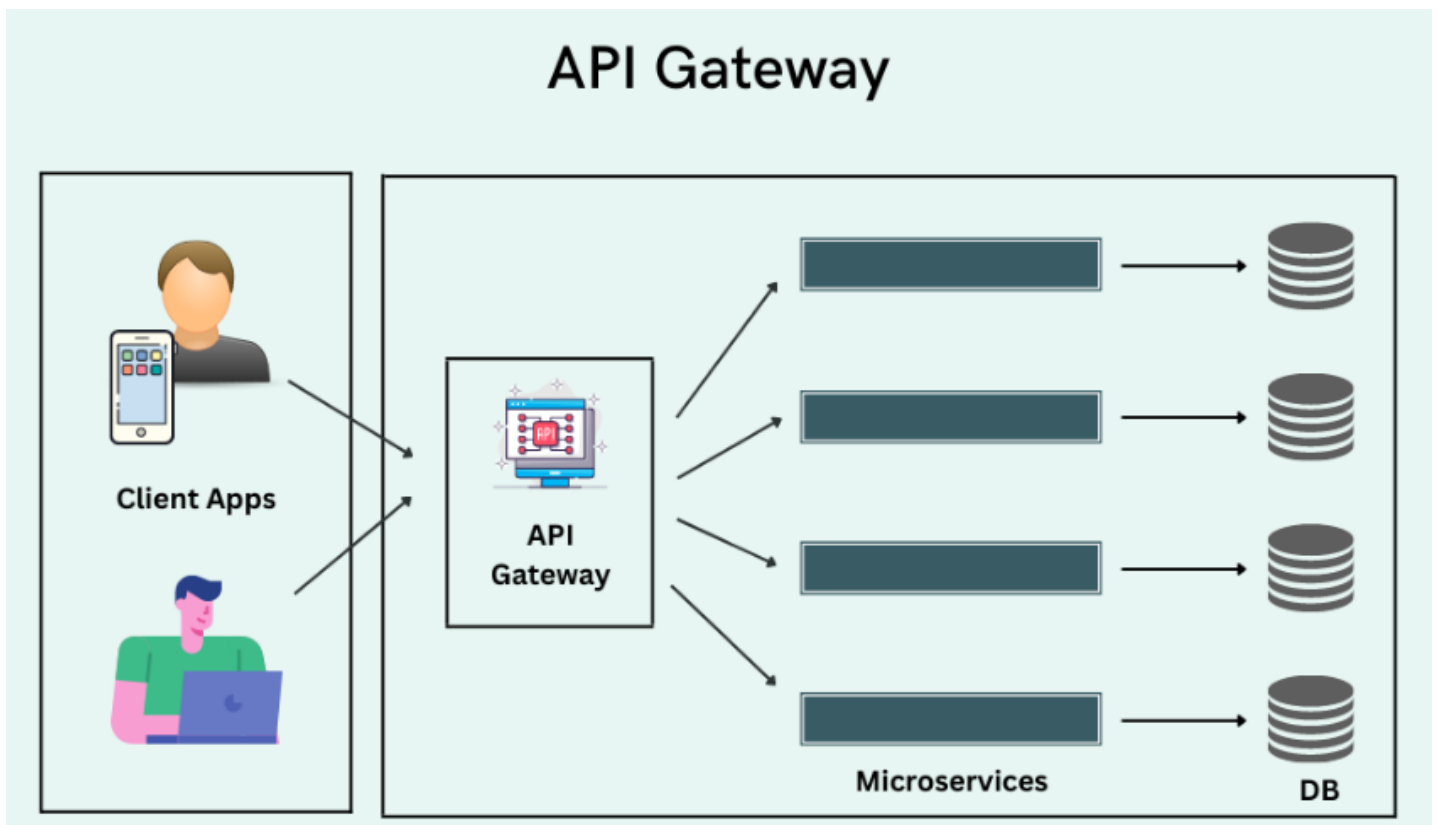
Eureka Service specification
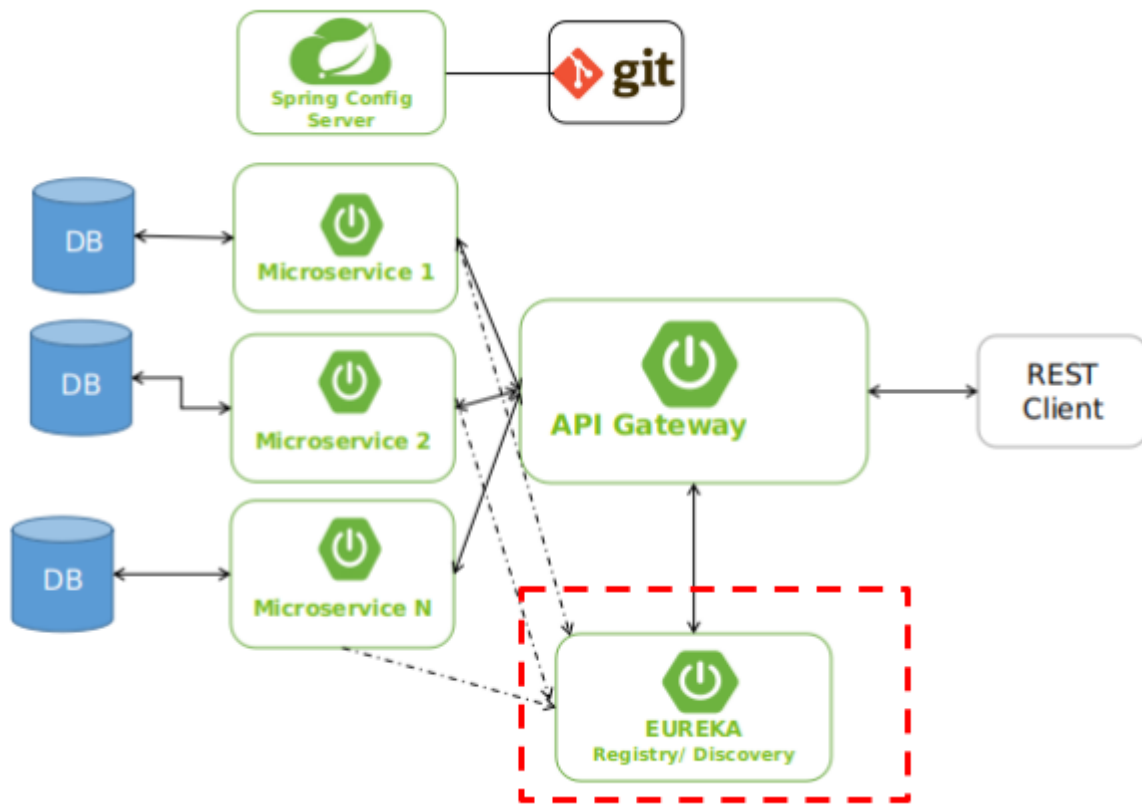
Infrastructure

Spring

Gateway

API Gateway

Infrastructure

Spring boot

Terrain Service specification

Infrastructure

Spring Boot

MySQL DB

Service Bus

External (User)

CreateTerrain - Creates Terrain.   Emits **saveCommandaddTerrain**

UpdateTerrain - Updates Terrain.   Emits **updateTerrain**

DeleteTerrain - Deletes Terrain.   Emits **deleteTerrainById**

Queries handled

getTerrainById - Gets Terrain by an ID.   Returns **TerrainDto**

getAllTerrains - Gets all Terrains.   Returns list of **TerrainDto**

Events emitted

TerrainCreated - When the Terrain is created, in response to CreateTerrain - Emits **TerrainDto**

 TerrainUpdated - When the Terrain is updated, in response to UpdateTerrain - Emits **CommandeDto**

 TerrainDeleted - When the Terrain is deleted, in response to DeleteTerrain - Emits **TerrainId**

API Endpoints

POST	/Terrain/addTerrain	Create Terrain	Auth
PUT	/Terrain/updateTerrain/{id}	Update Terrain	Auth
DELETE	/Terrain/removeTerrain/{idTerrain}	Delete Terrain	Auth
GET	/Terrain/listTerrains	Get Terrain s	Anon
GET	/Terrain/terrains/nom	Get Terrain by Nom	Anon
GET	/Terrain/terrains/type	Get Terrain by type	Anon

Models

Terrain.java

Property Name	Property Type	Default Value
Id	Long	
nom	String	
type	string	
emplacement	string	
capaciteSpectateurs	int	
eclairage	boolean	
prixLocation	double	

CreateTerrain

Property Name	Property Type	Default Value
nom	String	
type	string	
emplacement	string	
capaciteSpectateurs	int	
eclairage	boolean	
prixLocation	double	

UpdateTerrain

Property Name	Property Type	Default Value
nom	String	
type	string	
emplacement	string	
capaciteSpectateurs	int	
eclairage	boolean	
prixLocation	double	

DeleteTerrain

Property Name	Property Type	Default Value
Id	Long	

Evenement Service specification

Infrastructure

Spring Boot

MySQL DB

Service Bus

External (User)

CreateEvenement - Creates Evenement.   Emits **saveEvenement**

UpdateEvenement - Updates Evenement.   Emits **updateEvenement**

DeleteEvenement - Deletes Evenement.   Emits **deleteEvenementById**

Queries handled

getEvenementeById - Gets Evenement by an ID.   Returns **EvenementDto**

getAllEvenements - Gets all Evenements.   Returns list of **EvenementDto**

Events emitted

EvenementCreated - When the Evenement is created, in response to CreateEvenement
- Emits **EvenementDto**

EvenementUpdated - When the Evenement is updated, in response to UpdateEvenement
- Emits **EvenementDto**

 EvenementDeleted - When the Evenement is deleted, in response to DeleteEvenement
- Emits **EvenementId**

API Endpoints

POST	/event/add	Create Evenement	Auth
PUT	/event/{id}	Update Evenement	Auth
DELETE	/event/{id}	Delete Evenement	Auth
GET	/event/events	Get Evenements	Anon
GET	/event/{id}	Get Evenement by id	Anon

Models

Evenement.java

Property Name	Property Type	Default Value
Id	Long	
eventName	String	
description	string	
startDate	Date	
endDate	Date	
location	String	

CreateEvenement

Property Name	Property Type	Default Value
eventName	String	
description	string	
startDate	Date	
endDate	Date	
location	String	

UpdateEvenement

Property Name	Property Type	Default Value
eventName	String	
description	string	
startDate	Date	
endDate	Date	
location	String	

DeleteEvenement

Property Name	Property Type	Default Value
Id	Long	

Reservation Service specification

Infrastructure

Spring Boot

MongoDB

Service Bus

External (User)

CreateReservation - Creates Reservation.   Emits **saveReservation**

UpdateReservation - Updates Reservation.   Emits **updateReservation**

DeleteReservation - Deletes Reservation.   Emits **deleteReservationById**

Queries handled

getReservationById - Gets Reservation by an ID.   Returns **ReservationDto**

getAllReservations - Gets all Reservations.   Returns list of **ReservationDto**

Events emitted

ReservationCreated - When the Reservation is created, in response to CreateReservation
- Emits **ReservationDto**

 ReservationUpdated - When the Reservation is updated, in response to UpdateReservation
- Emits **ReservationDto**

 ReservationDeleted - When the Reservation is deleted, in response to DeleteReservation
- Emits**ReservationId**

API Endpoints

POST	/reservation/saveReservation	Create Reservation	Auth
PUT	/reservation/updateReservation	Update Reservation	Auth
DELETE	/reservation/deleteReservationById/{id}	Delete Reservation	Auth
GET	/reservation/getAll	Get Reservation	Anon
GET	/reservation/getReservationById/{id}	Get Reservation by id	Anon

Models

Reservation.java

Property Name	Property Type	Default Value
Id	String	
reservationName	String	
description	string	
startDate	Date	
endDate	Date	
status	string	

CreateReservation

Property Name	Property Type	Default Value
reservationName	String	
description	string	
startDate	Date	
endDate	Date	
status	string	

UpdateReservation

Property Name	Property Type	Default Value
reservationName	String	
description	string	
startDate	Date	
endDate	Date	
status	string	

DeleteReservation

Property Name	Property Type	Default Value
Id	string	

Produit Service specification

Infrastructure

Spring Boot

MongoDB

Service Bus

External (User)

CreateProduit - Creates Produit.   Emits **saveProduit**

UpdateProduit - Updates Produit.   Emits **updateProduit**

DeleteProduit - Deletes Produit.   Emits **deleteProduitById**

Queries handled

getProduitById - Gets Produit by an ID.   Returns **ProduitDto**

getAllProduits - Gets all Produits.   Returns list of **ProduitDto**

Events emitted

ProduitCreated - When the Produit is created, in response to CreateProduit   - Emits **ProduitDto**

 ProduitUpdated - When the Produit is updated, in response to UpdateProduit   - Emits **ProduitDto**

 ProduitDeleted - When the Produit is deleted, in response to DeleteProduit   - Emits **ProduitId**

API Endpoints

POST	/produits/addProduit	Create Produit	Auth
PUT	/produits/updateProduit	Update Produit	Auth
DELETE	/produits/deleteProduit/{id}	Delete Produit	Auth
GET	/produits/getAllProduits	Get Produit	Anon

Models

Produit.java

Property Name	Property Type	Default Value
Id	String	
Name	String	
description	string	
quantite	string	
prix	Number	
categorie	string	

CreateProduit

Property Name	Property Type	Default Value
Name	String	
description	string	
quantite	string	
prix	Number	
categorie	string	

UpdateProduit

Property Name	Property Type	Default Value
Name	String	
description	string	
quantite	string	
prix	Number	
categorie	string	

DeleteProduit

Property Name	Property Type	Default Value
Id	string	

Commande Service specification

Infrastructure

Spring Boot

MongoDB

Service Bus

External (User)

CreateCammande - Creates Commande.   Emits **saveCommand**

UpdateCammande - Updates Commande.   Emits **updateCommande**

DeleteCammande - Deletes Commande .   Emits **deleteCommandeById**

Queries handled

getCommandeById - Gets Commande by an ID.   Returns **CommandeDto**

getAllCommandes - Gets all Commandes.   Returns list of **CommandeDto**

Events emitted

CommandeCreated - When the Commande is created, in response to CreateCommande
- Emits **CommandeDto**

 CommandeUpdated - When the Commande is updated, in response to UpdateCommande
- Emits **CommandeDto**

 CommandeDeleted - When the Commande is deleted, in response to DeleteCommande
- Emits **CommandeId**

API Endpoints

POST	/commandes/saveCommande	Create Commande	Auth
PUT	/commandes/updateCommande	Update Commande	Auth
DELETE	/commandes/deleteCommandeById/{id}	Delete Commande	Auth
GET	/commandes/getAll	Get Commandes	Anon
GET	/commandes/getCommandeById/{id}	Get Commande by id	Anon

Models

Commande.java

Property Name	Property Type	Default Value
Id	String	
nomClient	String	
telephone	string	
adresse	string	
dateCommande	Date	
dateLivraison	Date	
PrixTotal	float	

CreateCommande

Property Name	Property Type	Default Value
nomClient	String	
telephone	string	
adresse	string	
dateCommande	Date	
dateLivraison	Date	
PrixTotal	float	

UpdateCommande

Property Name	Property Type	Default Value
nomClient	String	
telephone	string	
adresse	string	
dateCommande	Date	
dateLivraison	Date	
PrixTotal	float	

DeleteCommande

Property Name	Property Type	Default Value
Id	string	

Reclamation Service specification

Infrastructure

Spring Boot

MySQL DB

Service Bus

External (User)

CreateReclamation - Creates Reclamation.   Emits **createReclamation**

UpdateReclamation - Updates Reclamation.   Emits **updateReclamation**

DeleteReclamation - Deletes Reclamation.   Emits **deleteReclamationById**

Queries handled

getReclamationById - Gets Reclamation by an ID.   Returns **ReclamationDto**

getAllReclamations - Gets all Reclamations.   Returns list of **ReclamationDto**

Events emitted

ReclamationCreated - When the Reclamation is created, in response to CreateReclamation
- Emits **ReclamationDto**

 ReclamationUpdated - When the Reclamation is updated, in response to UpdateReclamation
- Emits **ReclamationDto**

 ReclamationDeleted - When the Reclamation is deleted, in response to DeleteReclamation
- Emits **ReclamationId**

API Endpoints

POST	/reclamation/add	Create Reclamation	Auth
PUT	/reclamation/update/{id}	Update Reclamation	Auth
DELETE	/reclamation/delete/{id}	Delete Reclamation	Auth
GET	/reclamation/list	Get Reclamation	Anon
GET	/reclamation/getById/{id}	Get Reclamation by id	Anon

Models

Reclamation.java

Property Name	Property Type	Default Value
Id	int	
sujet	String	
description	string	
email	string	

CreateReclamation

Property Name	Property Type	Default Value
sujet	String	
description	string	
email	string	

UpdateReclamation

Property Name	Property Type	Default Value
sujet	String	
description	string	
email	string	

DeleteReclamation

Property Name	Property Type	Default Value
Id	int	