



**Jorge Angulo, Mikel Mañeru eta Alvaro Viguera**

[**1.-Proiektuaren azalpen orokorra.**](#_7eokgy9suyyx) **2**

[**2-Teknologien instalazioa eta azalpenak.**](#_3tldsjav8rl9) **3**

[**2.1-ASP.NET:**](#_4mwf8qtd1bcy) **3**

[**2.2-RESTful API:**](#_ui54qk4hfh2e) **10**

[**2.3- MongoDB:**](#_alkzdnzcfw1i) **17**

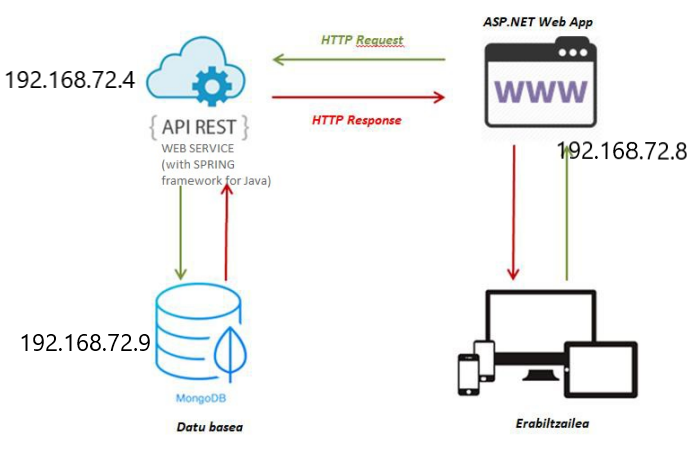
[**3-Datu Basea.**](#_vqrgnb8fcdsp) **18**

[**4-Ondorioak:**](#_vrfwaiuau9v) **20**

# 

# **1.-Proiektuaren azalpen orokorra.**

Erronka honen helburua datu-base baten biltegiratutako datuekin lan egingo duen web orri bat garatzea da.Horretarako 3 teknologia erabiliko ditugu.Hasteko ASP.NET teknologiarekin web orria garatuko dugu, honek, Javaren Spring Boot Frameworkean garatutako RESTful API bati eskariak egingo dizkio eta azkenik Restful zerbitzu honek, MongoDB datu-base baten biltegiratutako datuak atzituko ditu(gure kasuan hotelen informazioa kudeatuko dugu).Gainera hiru teknologiak bakoitza ordenagailu batean kokatuko da.

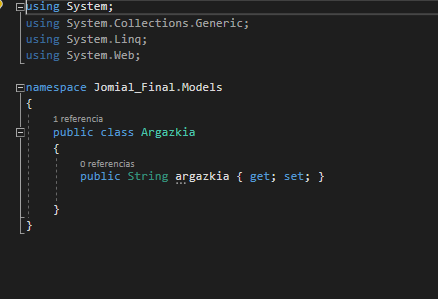


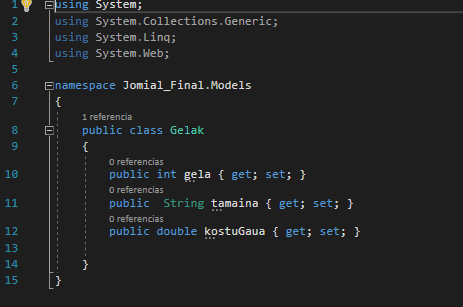
# **2-Teknologien instalazioa eta azalpenak.**

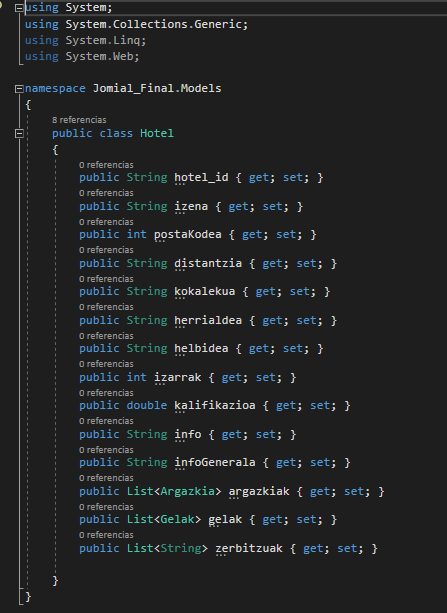
## **2.1-ASP.NET:**

Gure ASP.NET egiteko lehengo ASP.NET-etik, HTTPClient klasea erabiliz, REST API bat nola kontsumitu daitekeen bilatu dugu eta konekzioak egin ditugu tutorial honen bidez: [ASP.NET Web API REST zerbitzua kontsumitzea ASP.NET MVC-n HttpClient erabiliz](https://www.c-sharpcorner.com/article/consuming-asp-net-web-api-rest-service-in-asp-net-mvc-using-http-client/)

Erabili ditugun klaseak hauek dira:(Argazkia,Gelak,Hotel)

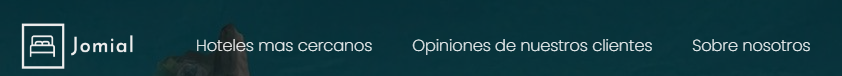






Gure Asp nabigazioa horrelako itxura dauka:

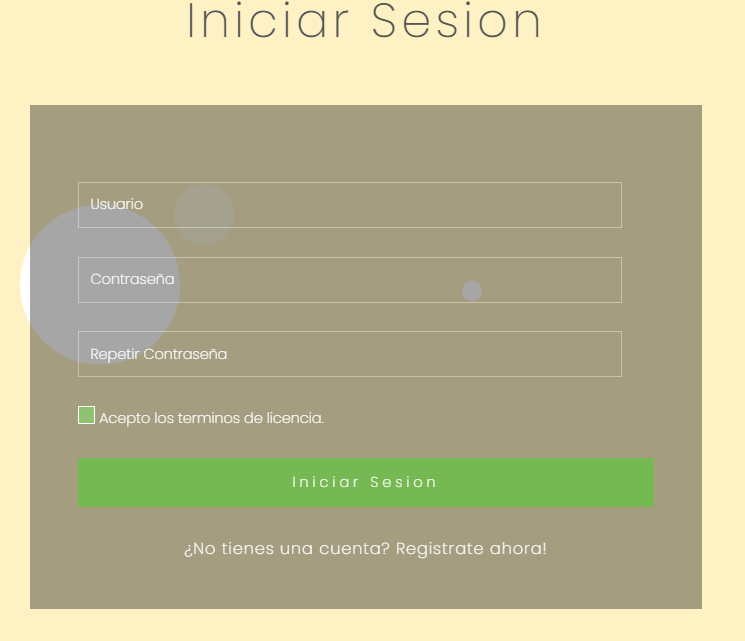
Hasieran Home controller bat daukagu gure web orriaren index-a exekutatzen duena.Index exekutatzean hainbat atal ditugu.





Ataletan sartzeko, lehenengo web orrian logeatu behar da,bestela abisu bat emango dizu.





Kontu korronterik ez badaukazu erregistratzeko aukera eskaintzen du.



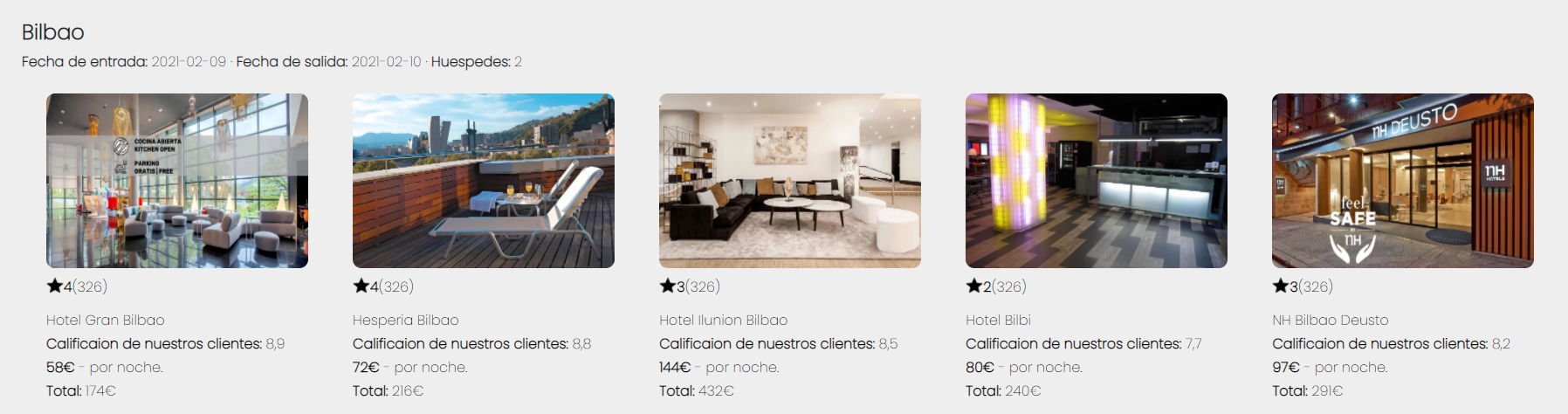
Kontu korronte batekin logeatzen zarenean ataletatik nabigatzeko aukera desblokeatzen da.

Index-eko orrian hotel bilatzaile bat daukagu.Bilatzaile honetan datu guztiak sartu behar dituzu bestela ezin izango da bilaketa egin.



Hiria Bilbao jartzerakoan Bilboko top 5 hotelak aterako dira.Hau ondo funtzionatzeko

BilatuController-eko BusquedaHotel metodoa erabiltzen du eta horrek sartutako parametroekin BusquedaHotel.cshtml zabaltzen du,Adibidez Bilbao sartzen badugu:



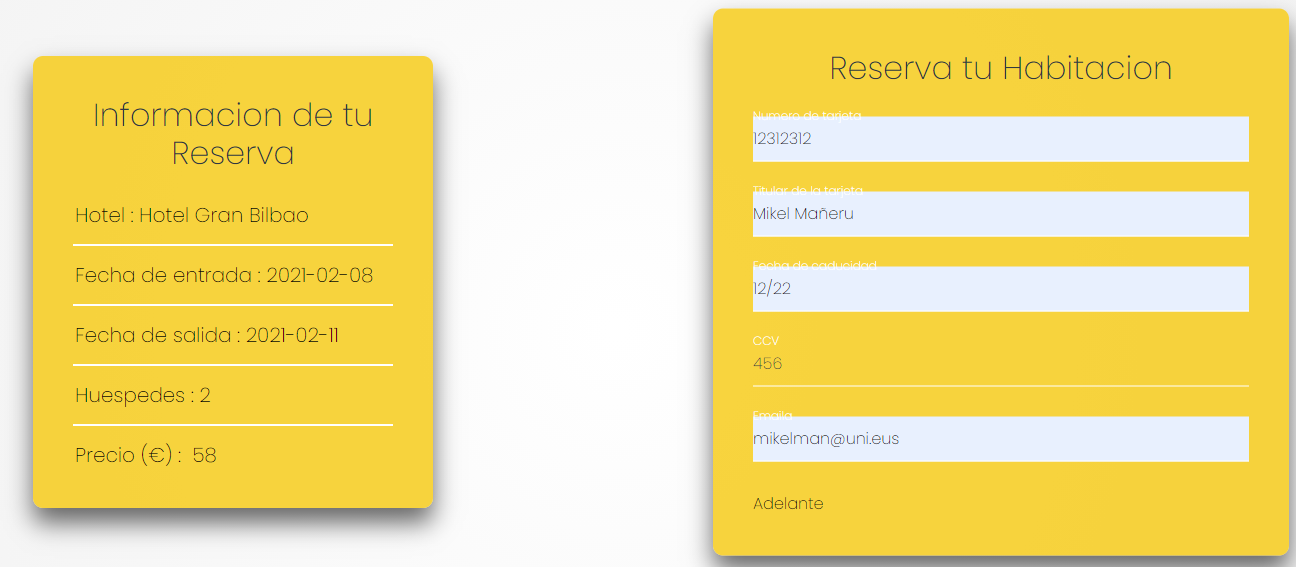
**Hoteles mas cercanos---->** hau klikatzen dugunean index-a href="@Url.Action("Hoteles", "Home")" egiten du. Hau da HomeController-eko Hoteles metodoa exekutatzen du eta Hoteles metodoa Hoteles.cshtml vista zabaltzen da.Bertan, 5 hiriko hotelak agertzen dira.



Bat klikatzen badugu bere informazioa ikusi dezakegu(alokatzeko aukera ere badago)



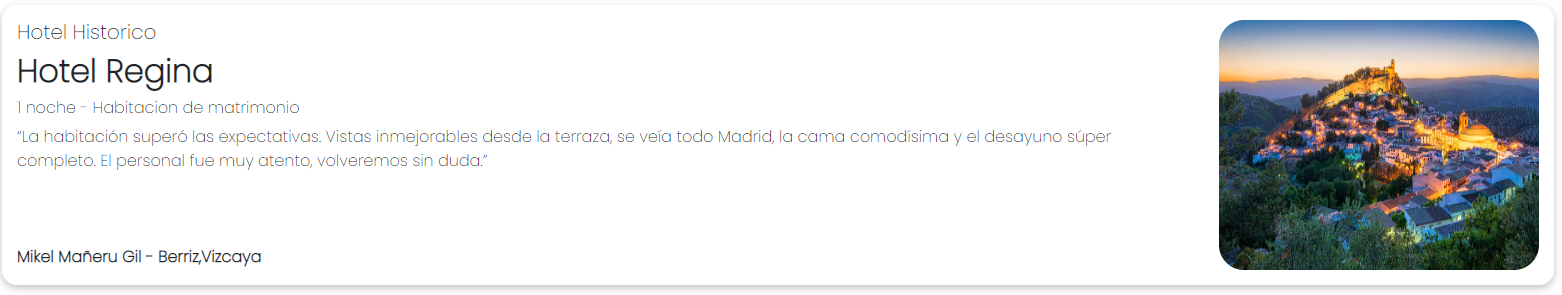
Gela bat alokatzeko datuak sartu eta adelante emanez egingo da.Gero zure alokairuaren informazioa eta zuk erosteko datuak sartzeko agertuko dira.



Azkenik adelante emanez alokairuaren informazioa agertzen da.Bat baino gehiago alokatzen badituzu ere agertuko dira(karrito bezalako bista da).



**Opinones de nuestros clientes---->**hau klikatzen dugunean index-a href="@Url.Action("Opinion", "Home")" egiten du.Hau da HomeController-eko Opinion metodoa exekutatzen du eta Opinion metodoa Opinion.cshtml vista zabaltzen da.Bertan, bezero batzuen iritziak agertzen dira.



**Sobre nosotros---->**hau klikatzen dugunean index-a href="@Url.Action("About", "Home")" egiten du.Hau da HomeController-eko About metodoa exekutatzen du eta About metodoa About.cshtml vista zabaltzen da.Bertan sortzaileen eta enpresaren informazioa agertzen da.

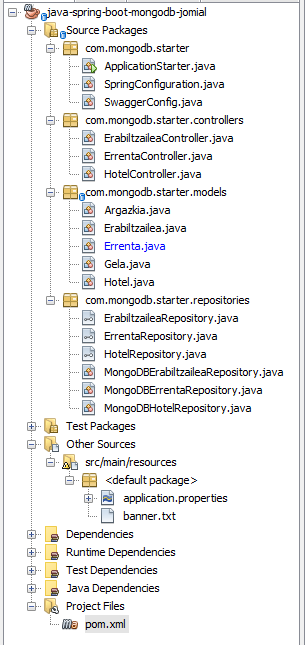
## **2.2-RESTful API:**

Gure REST APIa egiteko, urrats hauek jarraitu ditugu:

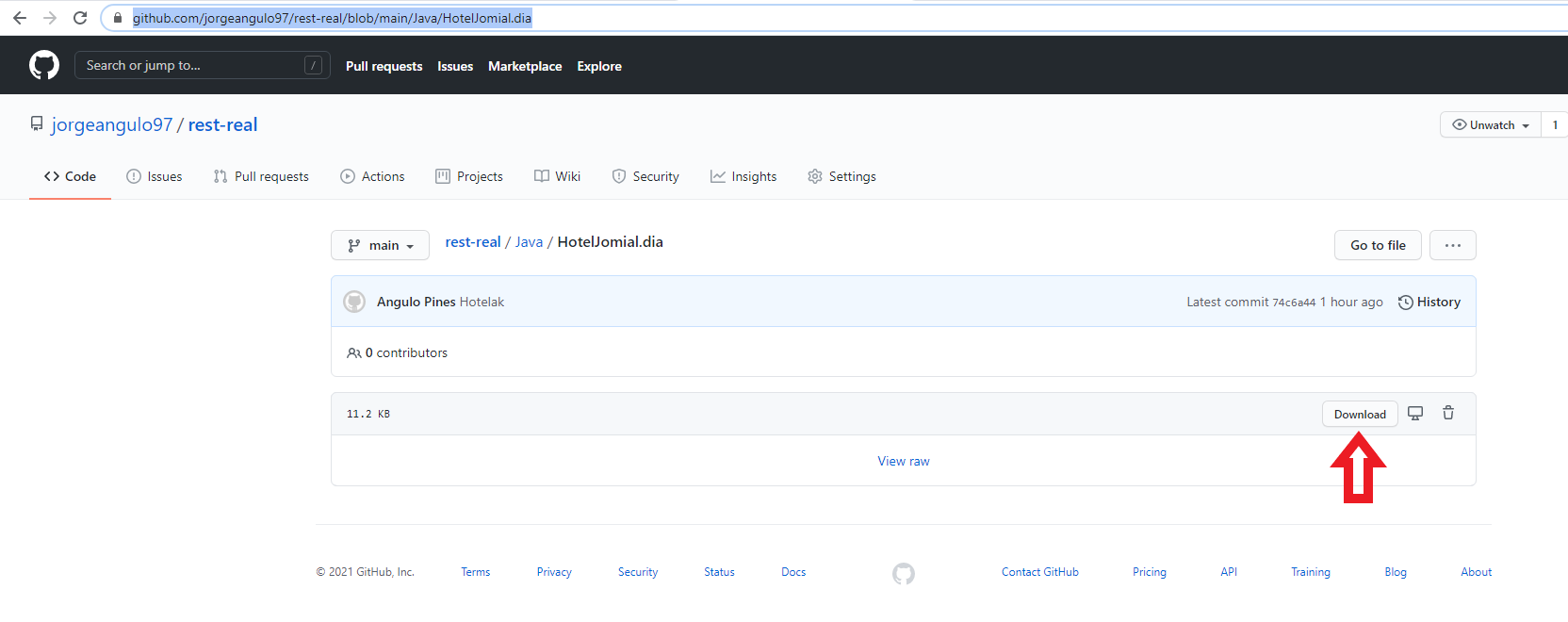
* Maxime Beugneten [Web Zerbitzua](https://www.mongodb.com/blog/post/restapis-with-java-spring-boot-and-mongodb) deskargatu dugu, ulertu eta Swagger bidez erabili dugu gure proiektua egiteko.
* Zerbitzua ordenagailu batean kokatu dugu IP honekin : 192.168.72.4
* Datu-basearen datuak hartuko ditugu ordenagailu honetatik IP: 192.168.72.9

Gure zerbitzuan datuak hartzeko **application.properties** artxiboa aldatu/gehitu behar dugu hau:

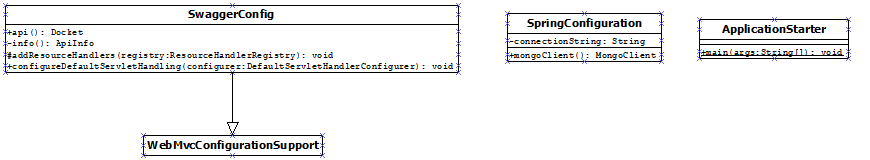
* spring.data.mongodb.uri=mongodb://192.168.72.9
* spring.data.mongodb.port=27017
* Zerbitzuaren egitura hurrengoa da:



* Zerbitzuaren klase diagrama oso handia denez gomendatzen dugu gure GitHub link honetatik deskargatzea: [Jomial Hotel Diagrama](https://github.com/jorgeangulo97/rest-real/blob/main/Java/HotelJomial.dia)

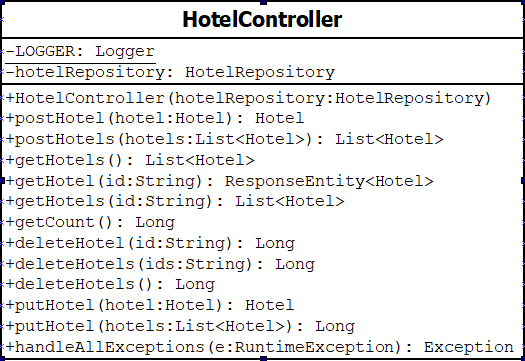


* Diagrama zabaltzerakoan ikusi ahal dugu, gure zerbitzariaren datuak nola dauden kokatuta:
* Starter: hemen Swagger, Spring, Application starter eta mvc konfigurazioa egongo da.

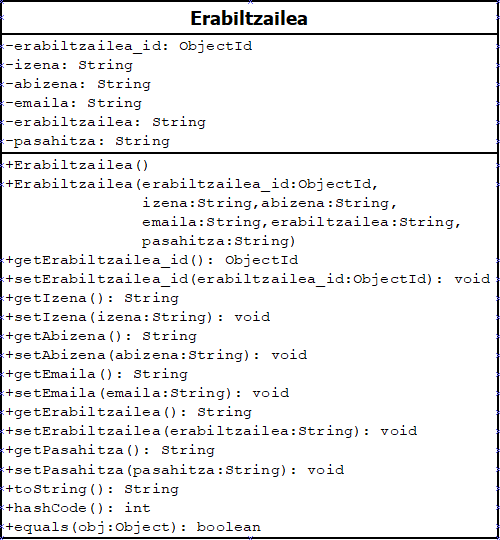


* Controlers: hemen Erabiltzailea, Errenta eta Hotel controllers egongo dira.

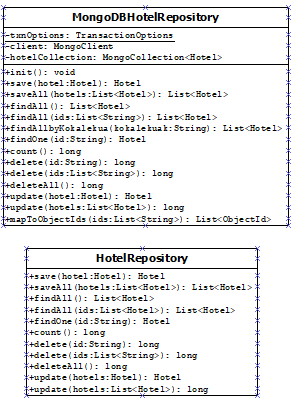
Erabiltzen ditugun metodoak eukiko ditu. Datuak eskatzeko, ezabatzeko, bilatzeko…



* Models: mongoDB Jomial colletions egituraren klaseak edukiko ditugu, bere metodekin.



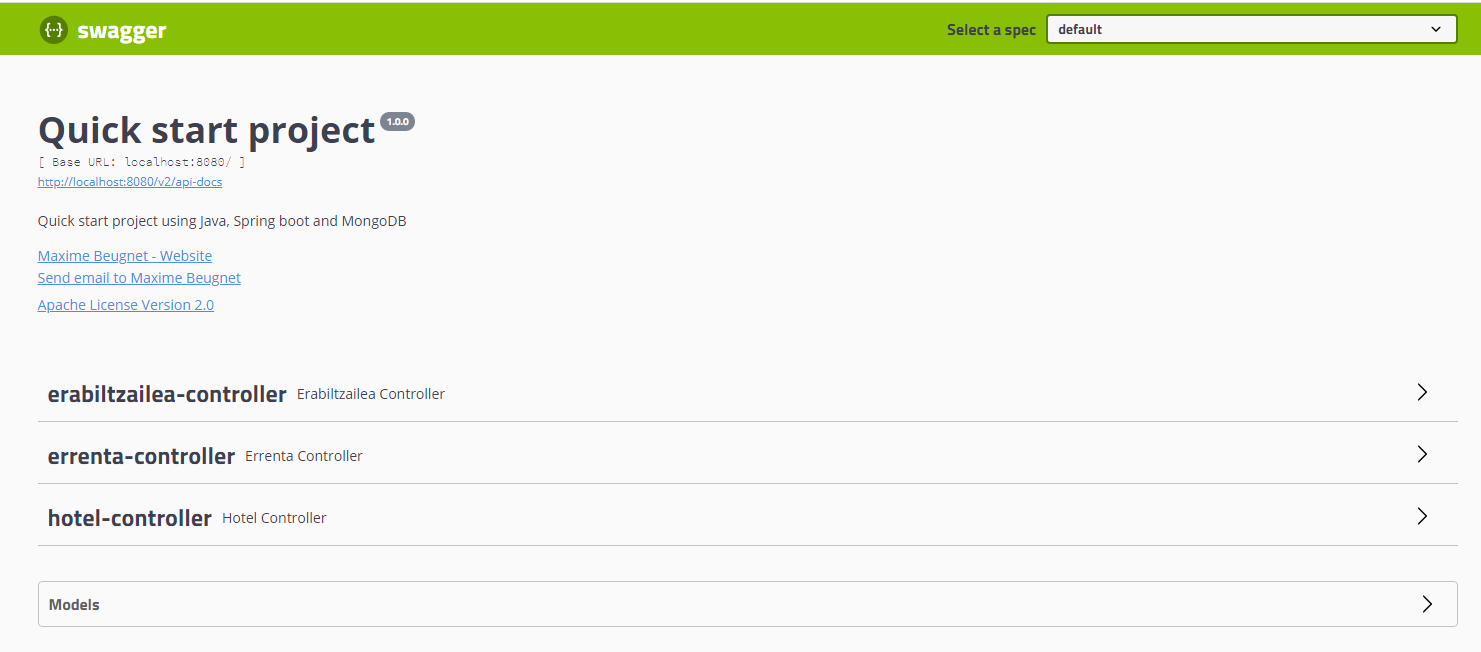
* Repositories: gure datu basearen errepositoriak edukiko ditu, hemen metodoak egongo dira, controladoretik dei bat egiten dugunean hona etortzeko.



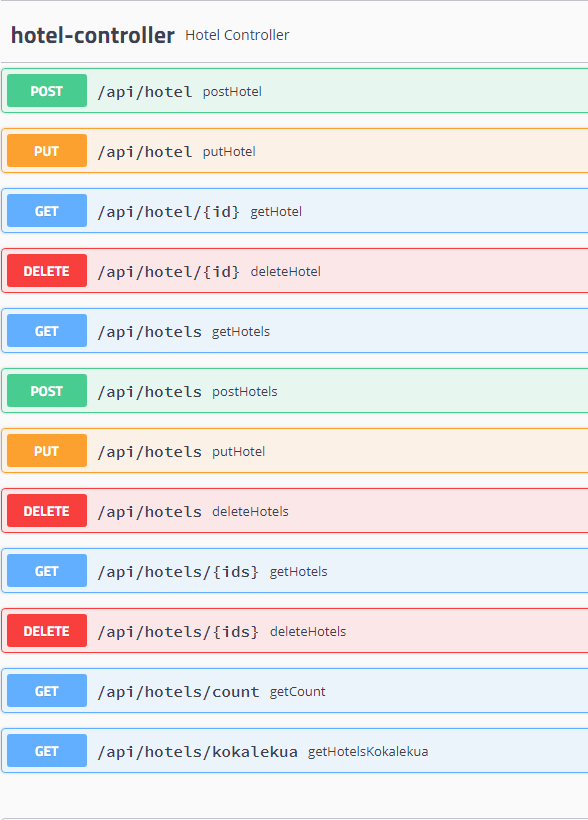
* Other Sources: hemen gure application.properties konfigurazioa egongo da.

## 

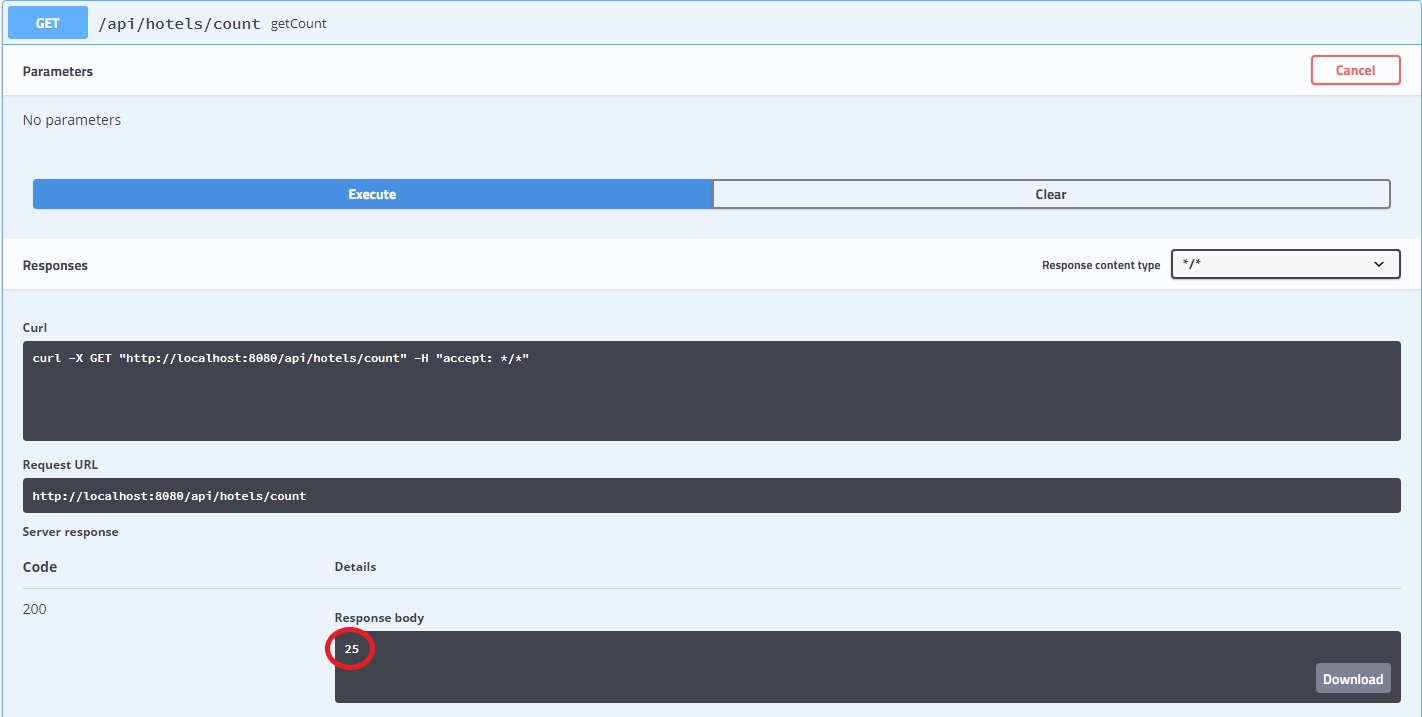
Gure Api Rest-real egitura horrela ikusten da Swaggerran:

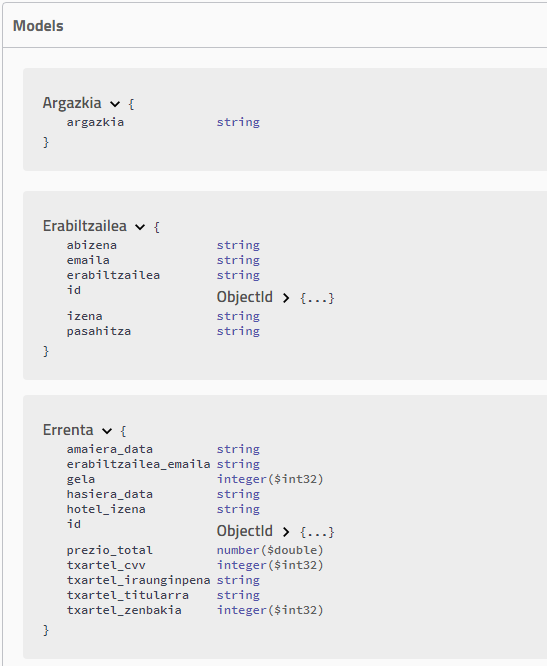


Swaggerren klaseen-kontroladoreak ikusi ahal ditugu egituratua:



Swaggerren gure zerbitzarian dauden metodoak exekutatu ahal ditugu eta emaitza ikusi :



Swaggerren gure klasearen modeloak ikusi ahal ditugu eta bertan baliabideak bere balio-ekin: 



Gainera gure zerbitzuaren Javadoc bat daukagu:

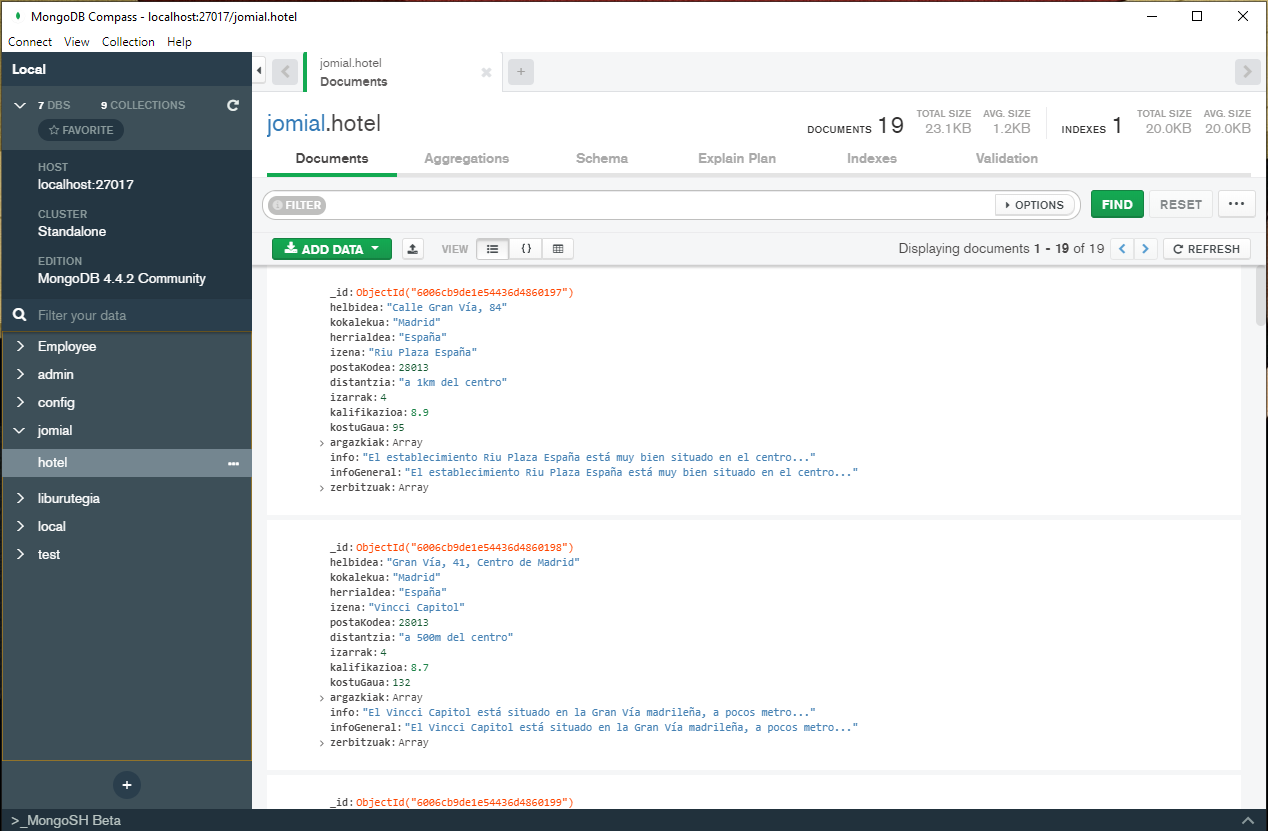
Hemendik deskargatu ahal duzu: <https://github.com/jorgeangulo97/rest-real> “Javadoc/apidocs/index.html”

## 

## **2.3- MongoDB:**

Mongo instalatzeko web orri hau erabili dugu ([Mongoren instalazioa](https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows/)). Web orri horretan mongo windows-ean instalatzeko pausuak azaltzen ditu.

Ondoren mongoDB instalatu dugunean, datuak era grafiko batean ikusteko mongo Compass deskargatu dugu ([Mongo Compass Deskargatu](https://www.mongodb.com/try/download/compass))



Azkenik mongoDB tools instalatu dugu([MongoDB tools deskargatu](https://www.mongodb.com/try/download/database-tools#:~:text=The%20MongoDB%20Database%20Tools%20are,soon%20as%20they%20are%20available.)), horrela hainbat abantaila irabazten dugu, Adibidez gure kasuan mongo import erabili ahal izateko.

Hau instalatzeko erraza da MongoDb tools descargatzerakoan rar bat deskargatzen da.Rar horrek karpeta bat dauka.Rar horren karpetan sartu eta gero “bin” karpetako ‘.exe’ guztiak kopiatu behar dira. ‘.exe’ guztiak kopiatutakoan MondoDB instalatutako karpetara joan behar da. Bertan bin karpeta aurkitu eta itsatsi ‘.exe’ guztiak.

# **3-Datu Basea.**

Esan dugun bezala guk hotelen errentarako proiektu bat egin dugu. Horretarako 3 datu base erabili ditugu(hotel,erabiltzailea eta errenta). Datu baseak sortzeko json eran egin ditugu eta gero mongoDB-ra importatu ditugu mongo import tool erabiliz.

Json egiturak mongora importatzeko hurrengoa egin behar da; ‘.json’ artxiboa copiatu eta mongoDB karpetako bin karpetan itsatsi behar da.Itsatsi duzunean CMD-a zabaldu eta bin karpeta horren helbidea copiatu(gure kasuan: cd C:\Program Files\MongoDB\Server\4.4\bin).Karpeta horretan zaudenean cmd bitartez hurrengo codigo sartu behar duzu.

(mongoimport --host=localhost:27017 --db= ZureDatuBasea --collection= ZureKolekzioa --file=artxiboa.json)

Guk erabili ditugun mongoinportak hurrengoak dira eta hemendik deskargatu ahal dira:

* <https://github.com/jorgeangulo97/rest-real> “MongoDB/artxiboa.json”
* mongoimport --host=localhost:27017 --db= Jomial --collection= hotel --file=hotel.json
* (mongoimport --host=localhost:27017 --db= Jomial --collection= erabiltzailea --file=erabiltzailea.json
* mongoimport --host=localhost:27017 --db= Jomial --collection= errenta --file=errenta.json

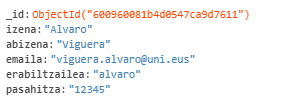
Hotel baten json egitura:

{"helbidea": "", "kokalekua": "", "herrialdea": "", "izena": "", "postaKodea": , "distantzia": "", "izarrak": , "kalifikazioa": ,"argazkiak":[{"argazkia": ""},{"argazkia": ""}, {"argazkia": ""}, {"argazkia": ""}, {"argazkia": ""}, {"argazkia": ""}], "info":", "infoGenerala":"","gelak":[{"gela": , "tamaina": "Para 2 personas camas dobles","kostuGaua": },{"gela": , "tamaina": "Habitacion individual","kostuGaua": },{"gela": , "tamaina": "Habitacion de matrimonio","kostuGaua": }],"zerbitzuak":["","","","",""]}

Erabiltzaile baten json egitura:

{"izena": "", "abizena": "", "emaila": "", "erabiltzailea": "", "pasahitza": ""}

Erabiltzaile baten egitura datu basean sartuta:



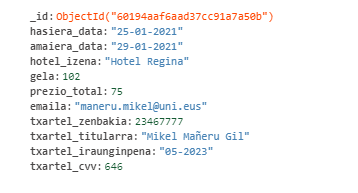
Hotel baten egitura datu basean sartuta:



Errenta baten json egitura:

{"hasiera\_data": "", "amaiera\_data": "", "hotel\_izena": "", "gela": , "prezio\_total": , "emaila”: “" ,"txartel\_zenbakia": , "txartel\_titularra": "", "txartel\_iraunginpena": "", "txartel\_cvv": }

Errenta baten egitura datu basean sartuta:



# **4-Ondorioak:**

Erabilitako iturriak:

* [Booking Web-orria](https://www.booking.com/index.es.html)
* [AirBnB web orria](https://es-l.airbnb.com/)
* [Css adibideak](https://freefrontend.com/)

Egindako gauza zailak:

* Logela bat erreserbatzeko orduan , zaila izan da bi formulariotetik datuak hartzea , datuak dena batera insertatzeko datu basean.
* JavaScript zaila izan da funtzioak egiterakoan.
* Diseinua zaila eta denbora handia hartu du.
* Java Spring zerbitzuan metodo batzuk egiteko zailak izan dira.
* Hasieran, datu baseko erregistroen egitura bat zehaztu genuen baina, denbora pasatu ahala ASP-en datuak erabiltzeko datu basearen egitura aldatu behar izan genuen.

## 

Egin gabe geratu diren gauzak edo hobetu ahal direnak:

* Gela bat alokatzeko momentuan 3 gelen artean aukeratu daiteke baina alokairua amaitzerakoan beti lehengo gela hartzen du
* Gela bat alokatzeko momentuan datetan iraganeko egun bat jarri ahal da.
* Java script bidez ez dakigu nola kalkulatu daten arteko egunen aldea gero prezioa kalkulatzeko.

Teknologien balorazioa:

* Erabili ditugun teknologiak, ez dira txarrak baina mongoDB “erlazionala”

ez denez, klase batzuen konexioak egiteko arazoak izan ditugu adibidez:

* Hotel eta Errenta klaseenkin erlazioa egiteko, ezin izan dugu hotelaren objektu guztia errenta batean sartu, beraz, hotelaren izena bakarrik sartu dugu.
* Eman diguten teknologien informazioa pixka bat abstraktua izan da eta gauzak asko gure kabuz bilatu behar izan dugu.