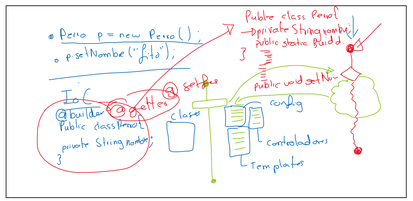
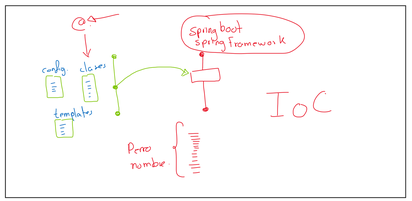
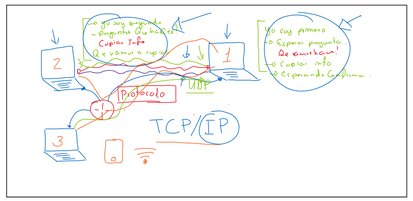
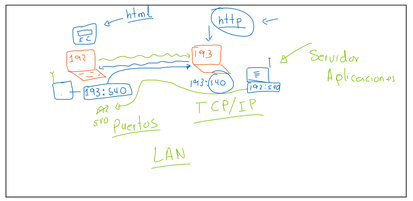
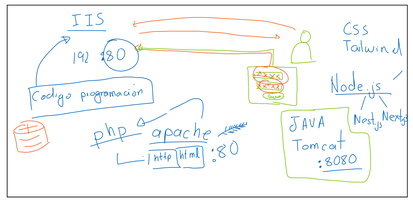
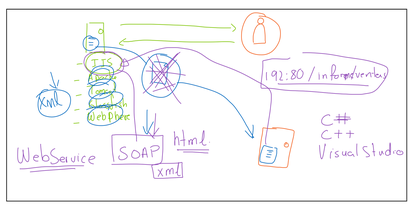

Perro p = new Perro();
p.setNombre("fido");
IoC
@Builder
@getter
@setter
Public class Perro
private String nombre;
Public class Perro
private String nombre;
public static Build
Public void setNo
clase
config
controladores
Templates

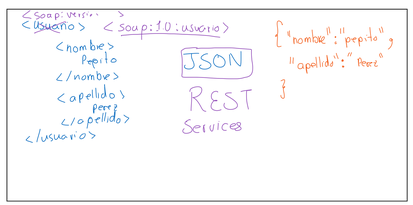

config
clases
templates
Spring boot
Spring framework
IoC
Perro
nombre

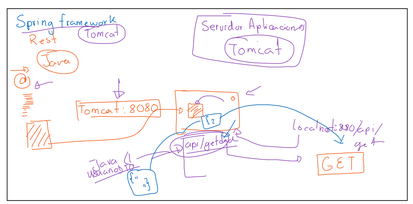

Protocolo
UDP
TCP/IP
LAN

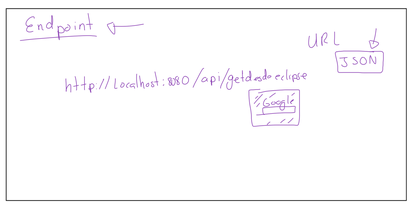

html
http
Servidor Aplicaciones
Puertos
TCP/IP
LAN

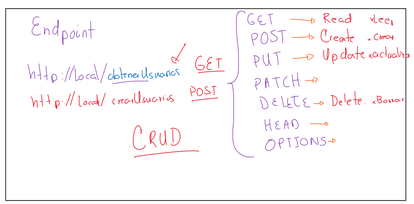

IIS
:80
Codigo programacion
CSS
Tailwind
Node.js
Next
Nestjs
JAVA
Tomcat
:8080
php
apache
:80
http
html

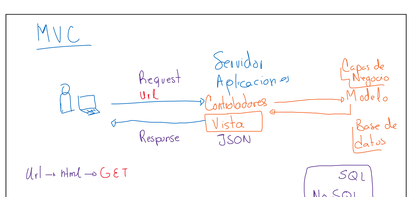

IIS
XML
WebDline
192:80/infoproventas
C#
C++
Visual Studio
WebService
SOAP
html
xml

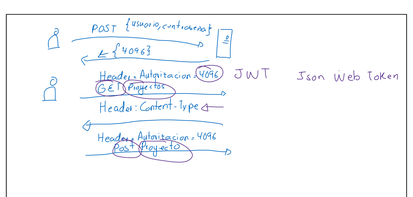

<soap:10:usuario>
<usuario>
<nombre>
Pepito
</nombre>
<apellido>
Perez
</apellido>
</usuario>
JSON
REST
Services
{"nombre": "pepito",
"apellido": "Perez"
}

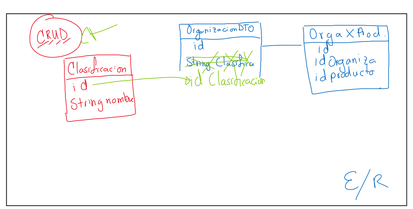

Spring framework
Rest
Tomcat
Java
Servidor Aplicacion
Tomcat
Tomcat: 8080
Localhost:8080/api/get
GET

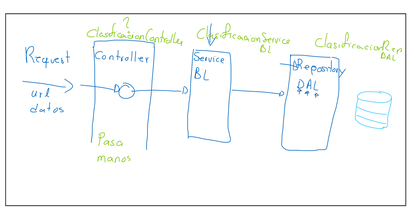

Endpoint
URL
JSON
http://Localhost:8080/api/getdatadeeclipse
Google

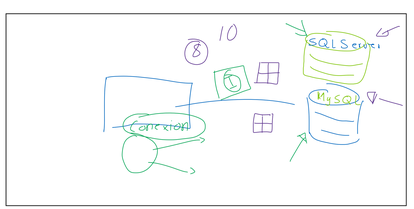

Endpoint
http://local/obtenerUsuarios
GET
http://Local/crearUsuarios
POST
CRUD
GET → Read
POST → Create
PUT → Update
PATCH →
DELETE → Delete
HEAD →
OPTIONS →

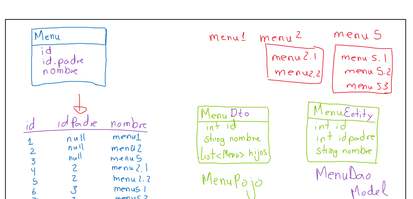

MVC
Request
Url
Servidor Aplicacion
Controladores
Capas de Negocio
Modelo
Vista
JSON
Base de datos
Response
Url → html → GET
SQL
No SQL

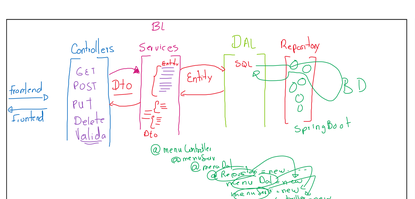

POST {usuario, contraseña}
{"49963"}
Header: Autorizacion 4096
GET Proyectos
Header: Content-Type
Header: Autorizacion: 4096
POST Proyecto
JWT
Json Web Token

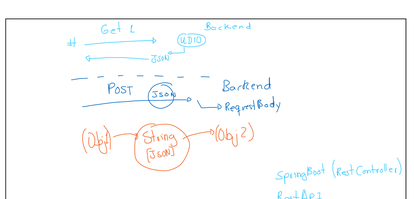

CRUD
Clasificacion
id
String nombre
OrganizacionDTO
id
String Clasificacion
id Clasificacion
OrgaXProd
id
id Organiza
id producto
E/R

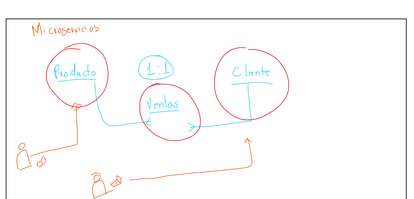

Request
url
datos
ClasificacionController
Controller
Pasa manos
ClasificacionServicio
Service
BL
ClasificacionRepo
Repository
DAL

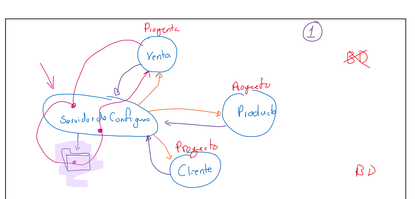

IO
SQLServer
MySQL
Conexion


Menu
id
id_padre
nombre
menu1
menu2
menu5
menu2.1
menu2.2
menu5.1
menu5.2
menu5.3
id | idPadre | nombre
1 | null | menu1
2 | null | menu2
3 | null | menu5
4 | 2 | menu2.1
5 | 2 | menu2.2
6 | 3 | menu5.1
7 | 3 | menu5.2
8 | 3 | menu5.3
MenuDto
int id
String nombre
List<Menu> hijos
MenuPojo
MenuEntity
int id
int id padre
String nombre
MenuDao
Model


BL
Controllers
Services
DAL
Repository
GET
POST
PUT
Delete
Valida
Dto
Entity
SQL
BD
SpringBoot
frontend
Fronted


Get 1
Backend
UDTO
JSON
POST
Backend
RequestBody
(Obj1)
String (JSON)
(Obj2)
SpringBoot (RestController)
RestApi


Microservicios
Producto
1:1
Ventas
Cliente


Proyecto
Venta
Proyecto
Producto
Servidor de Configuraciones
Proyecto
Cliente
BD