

# PokéUni

## Equipo Back-End

- Aguirre Cristian
- Condor Torres
- Colcas Victor
- Palomino Gerald
- Sarmiento Daniel



# Índice



**INTRODUCCIÓN**



**TECNOLOGÍAS  
USADAS**



**DIAGRAMAS**



**TABLAS**



**CONSULTAS**

# 1.INTRODUCCIÓN

La aplicación PokeUni está diseñada por diferentes grupos de trabajo, cada uno de estos grupos son responsable de una parte del sistema que se les encomienda como por ejemplo los encargados del front-end o también los encargados de las preguntas.

Nosotros somos responsables del diseño y la implementación del back-end la cual se encarga principalmente de proveer, administrar y responder las consultas que puedan requerir el front-end de la aplicación tanto de la web como el móvil.

Aplicaremos también los casos de usos de la consulta a la base de datos y veremos los diferentes diagramas.



## 2. Tecnología Usada

### NodeJS:

Es un entorno JavaScript de lado de servidor que utiliza un modelo asíncrono y dirigido por eventos. Uno de los puntos fuertes de Node.js es su capacidad de mantener muchas conexiones abiertas y esperando.

### MySQL:

MySQL es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL).

### NPM:

Es un gestor de paquetes bajo el lenguaje JavaScript, a través del cual podemos obtener cualquier librería con tan solo una sencilla línea de código, lo cual nos permitirá agregar dependencias de forma simple, distribuir paquetes y administrar eficazmente tanto los módulos como el proyecto a desarrollar en general.

### PostMan:

Está compuesta por diferentes herramientas y utilidades gratuitas que permiten realizar tareas diferentes dentro del mundo API REST.



## 2.Arquitectura de Diseño

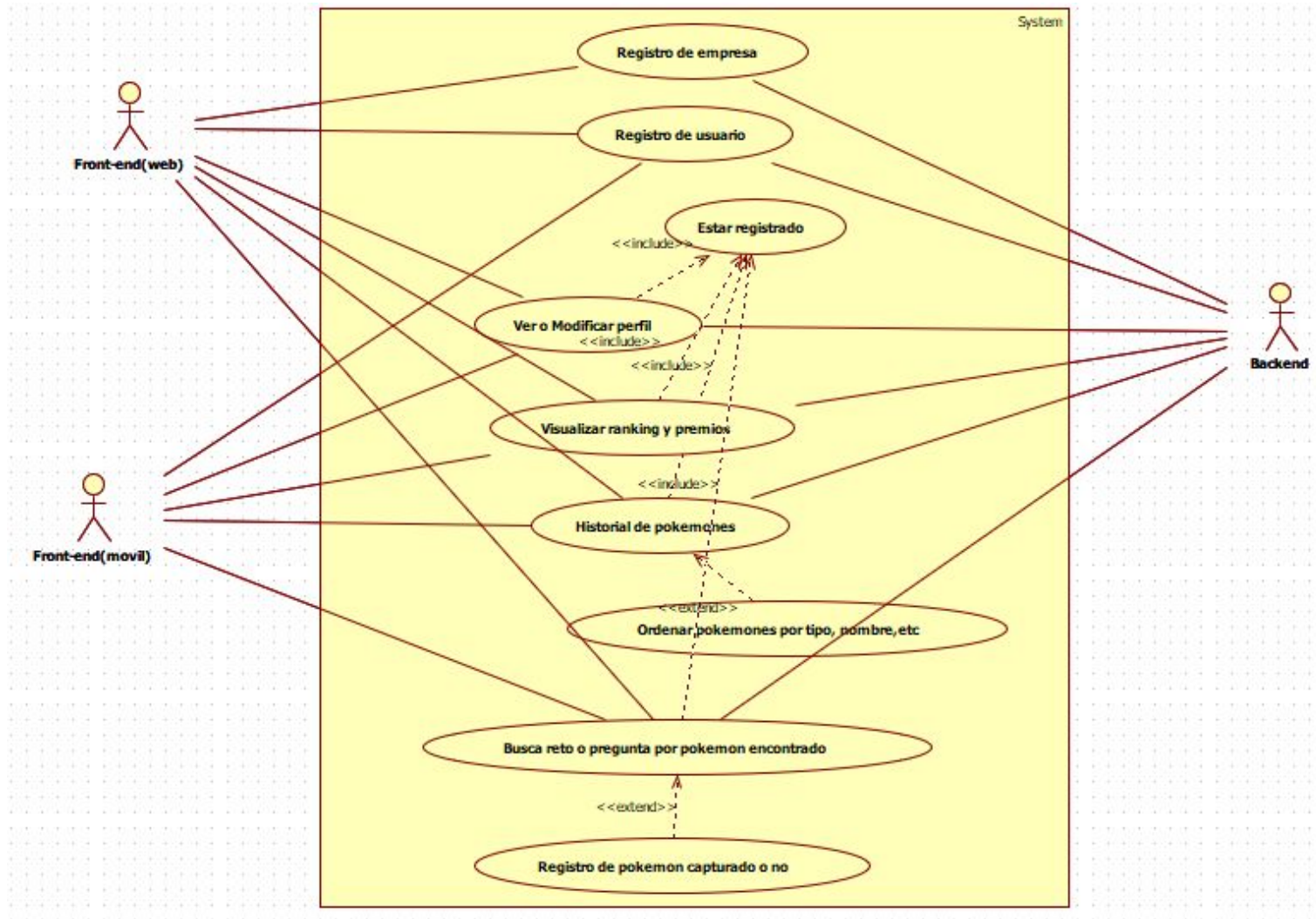
Está formado de 3 modulos de codigo y 1 segmento de base de datos .

- Módulo de APP: para recibir y enviar respuestas.
- Módulo middleware: Dar las rutas para utilizar la API y las lógicas de estas.
- Módulo de DataBase: Comunicación con la capa de Base de datos.
- Segmento de base de datos MySQL

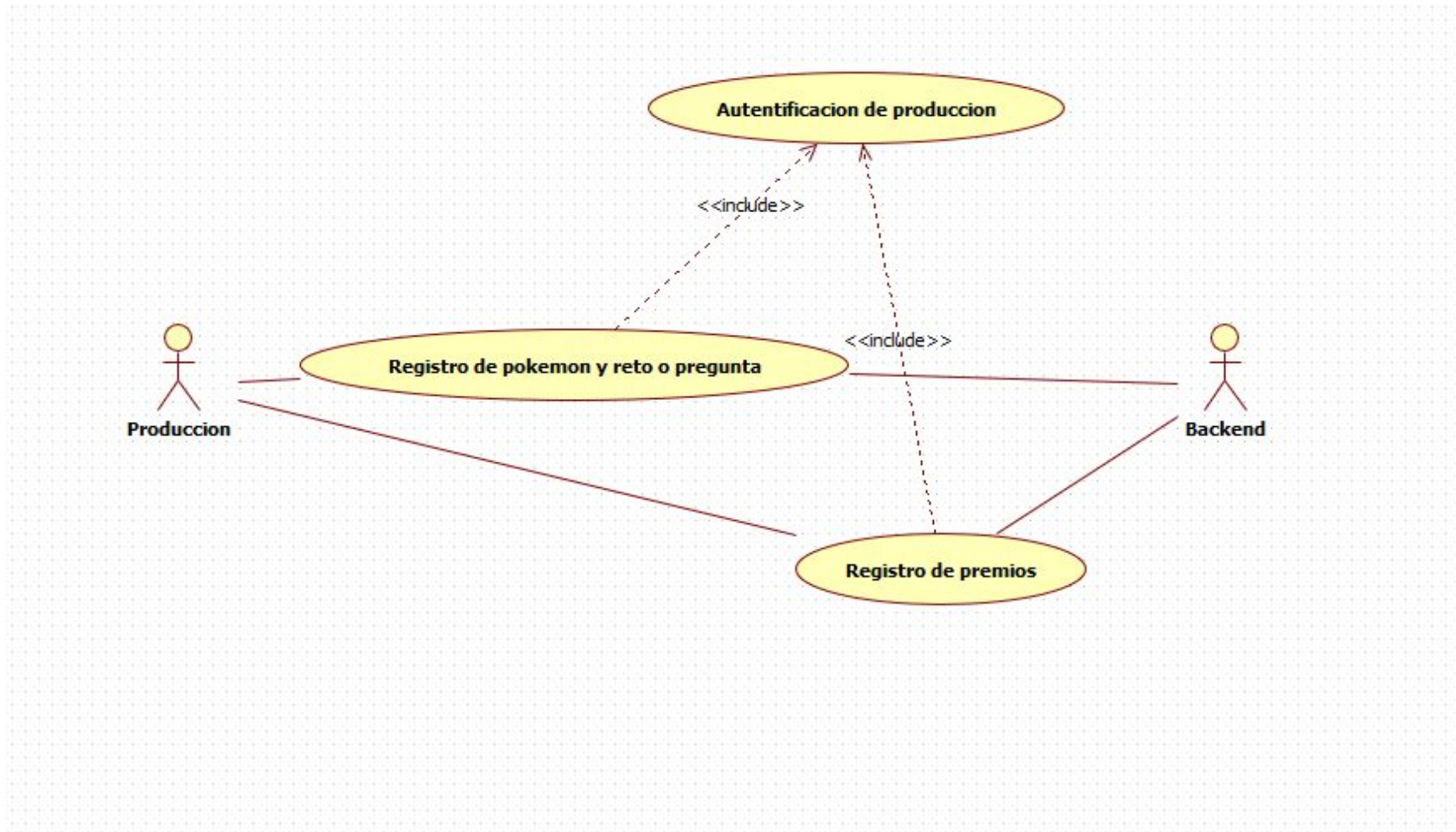




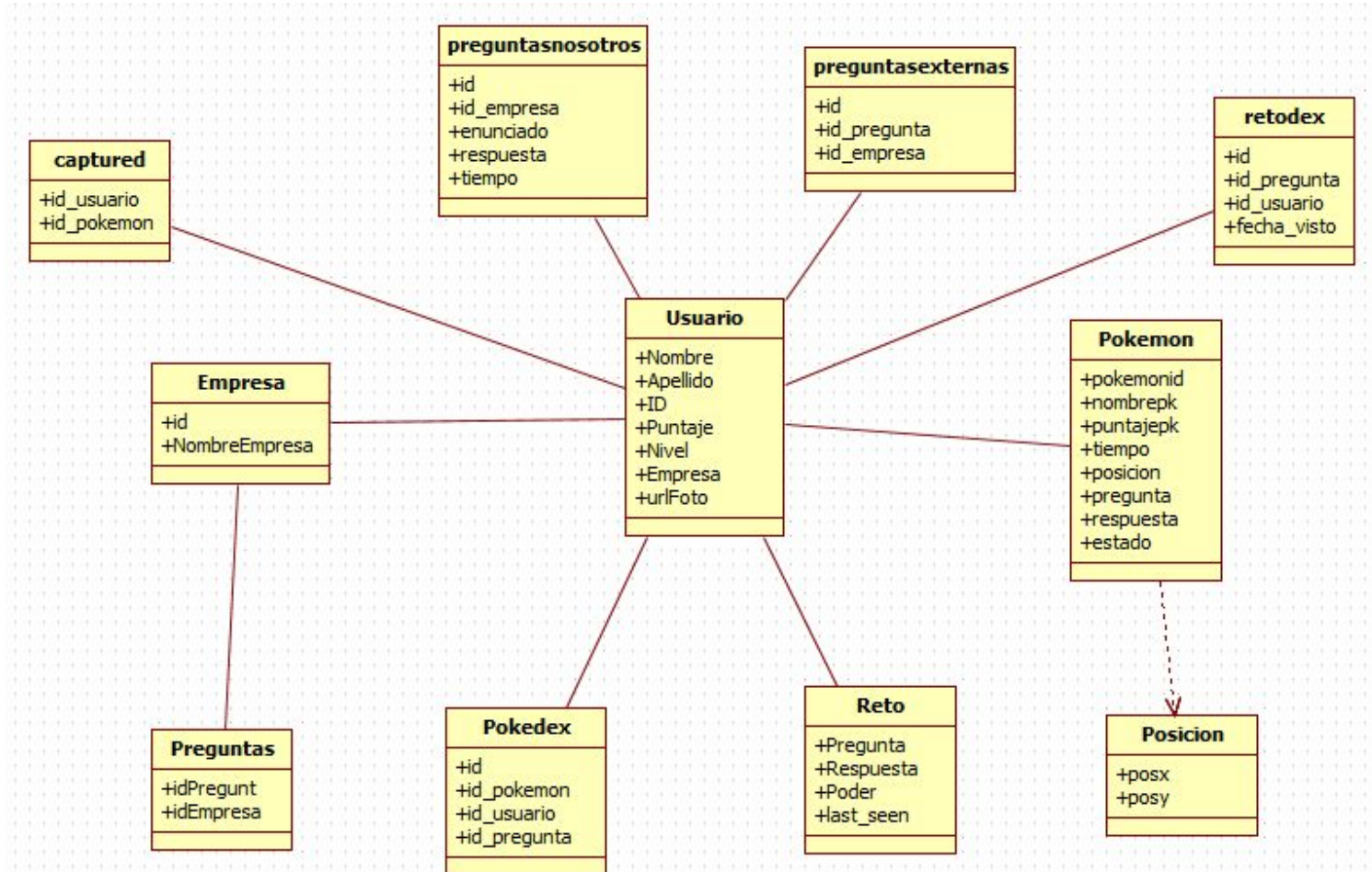
# 3.DIAGRAMA CASOS DE USOS



### 3. DIAGRAMA CASOS DE USOS

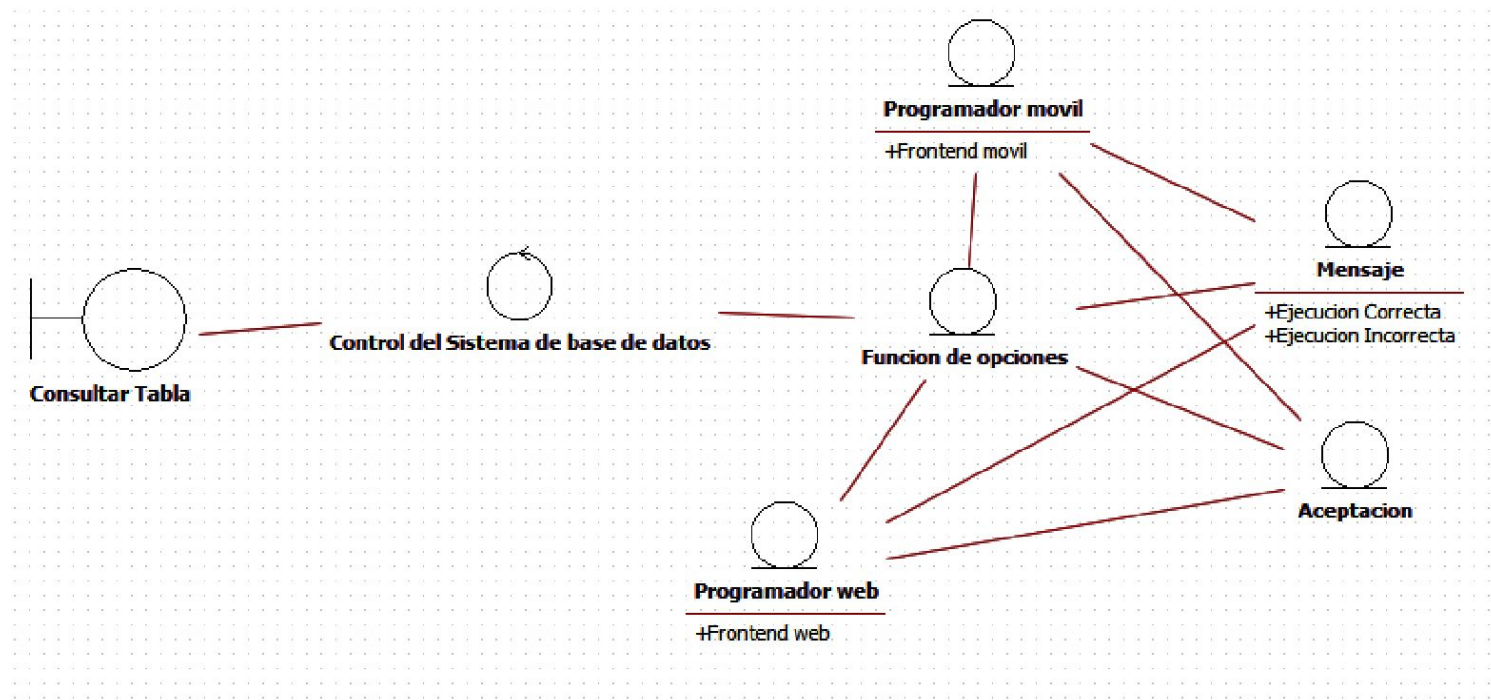


### 3. DIAGRAMA MODELO DE DISEÑO

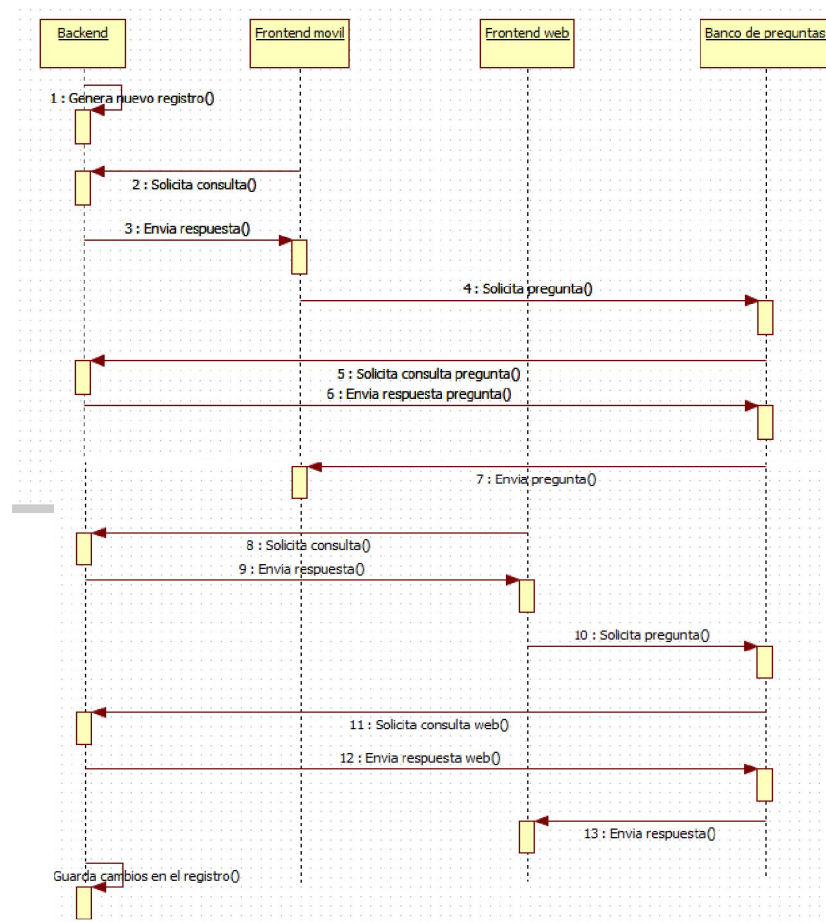




### 3. DIAGRAMA DE ROBUSTEZ



### 3. DIAGRAMA DE SECUENCIA



### 3. TRABAJO DEL BACK-END INTERNAMENTE

Funcionamiento de la parte interna desde que se recibe la información en el Back-end APP y ver que tipo de datos se consulta con el MySQL



# 4.TABLAS

empresa		pokemon		pokedex
id nombre		id		id_pokemon
		nombre		id_usuario
		posicionx		id_preguntas
		posiciony		id
		estado		
retodex		usuario		preguntas
id_pregunta		id		idpregunta
id_usuario		nombre		idempresa
id_pregunta		apellido		
		username		
		puntaje		
		nivel		
		id_empresa		

captured		preguntasnosotros		preguntasexternas
id_usuario		id		id
id_pokemon		id_empresa		id_empresa
		enunciado		id_pregunta
		respuesta		
		tiempo		
reto				
pregunta				
respuesta				
poder				
last_seen				



# 5.CONSULTAS

```
setUser(nombre:string,apellido:string,username:string,password:string,puntaje:Number,nivel:Number,id_empresa:Number) {  
  
    conection.query('insert into usuarios (nombre,apellido,username,password,puntaje,nivel,id_empresa) values (''+nombre+'',''+apellido'  
  
}  
  
/  
setUserAdmin(nombre:string,apellido:string,username:string,password:string,puntaje:Number,nivel:Number,id_empresa:Number) {  
    var tipo:string = "administrador"  
  
    conection.query('insert into usuarios (nombre,apellido,username,password,puntaje,nivel,id_empresa) values (''+nombre+'',''+apellido'  
  
}  
  
/  
setPokedex(id_pokemon:number,id_usuario:number,id_pregunta:number){  
    conection.query('insert into pokedex (id_pokemon,id_usuario,id_pregunta) values (''+id_pokemon+'',''+id_usuario+'',''+id_pregunta+''  
  
}
```





# 5.CONULTAS

```
setCompany(nombre :string) {  
    //var consulta =  
  
    conection.query('insert into empresa (nombreEmpresa ) values ( '"+nombre+"' );', (err, res) => {  
        console.log(res)  
    })  
  
    conection.query(' select * from empresa ', (err, res) => {  
        console.log(res)  
    })  
  
    //conection.end()  
}
```

```
getIDEmpresaByName(name:string):number{  
    var ret:number=-1;  
    conection.query('select id from empresa as e where e.nombreEmpresa="'+name+"'",(err,res)=>{  
        console.log(res,res)  
        ret=res  
    })  
    return ret  
}
```

