



UNIVERSIDAD AUTONOMA DE COAHUILA  
FACULTAD DE INGENIERÍA MECÁNICA Y  
ELÉCTRICA

NOVIEMBRE, 2022

# PROYECTO FINAL

TECNOLOGIAS DE  
INFORMACIÓN

SILVA GAYTAN YULISSA  
PEDRAZA RIOS VANESSA JAQUELINE

# Índice

Bases de Datos .....	4
5 Entidades .....	4
Modelo Entidad Relación .....	4
Modelo Relacional .....	5
Diccionario de Datos .....	6
Código SQL Creación .....	7
Código SQL Consultas Acción .....	8
Redes .....	9
Configuración IP fija W/L .....	9
Configuración de firewall por puertos W/L .....	11
Prueba de ping en ambos sentidos W/L .....	12
Acceso W/L SSH .....	13
Sistemas Operativos .....	13
SO W/L Instalado y Configurado .....	14
SMDB con Cliente W/L .....	15
Lenguaje de Programación e IDE W/L .....	16
Aplicación .....	16
Conexión Acceso a la Aplicación W/L .....	17
Uso de POO W/L .....	17
Logín W/L .....	17
CRUD para cada Tabla W .....	17
CRUD para cada Tabla L .....	18
Exportar a 5 Tipos de Archivos Diferentes .....	18
Código Disponible en Github Público .....	19

# Introducción

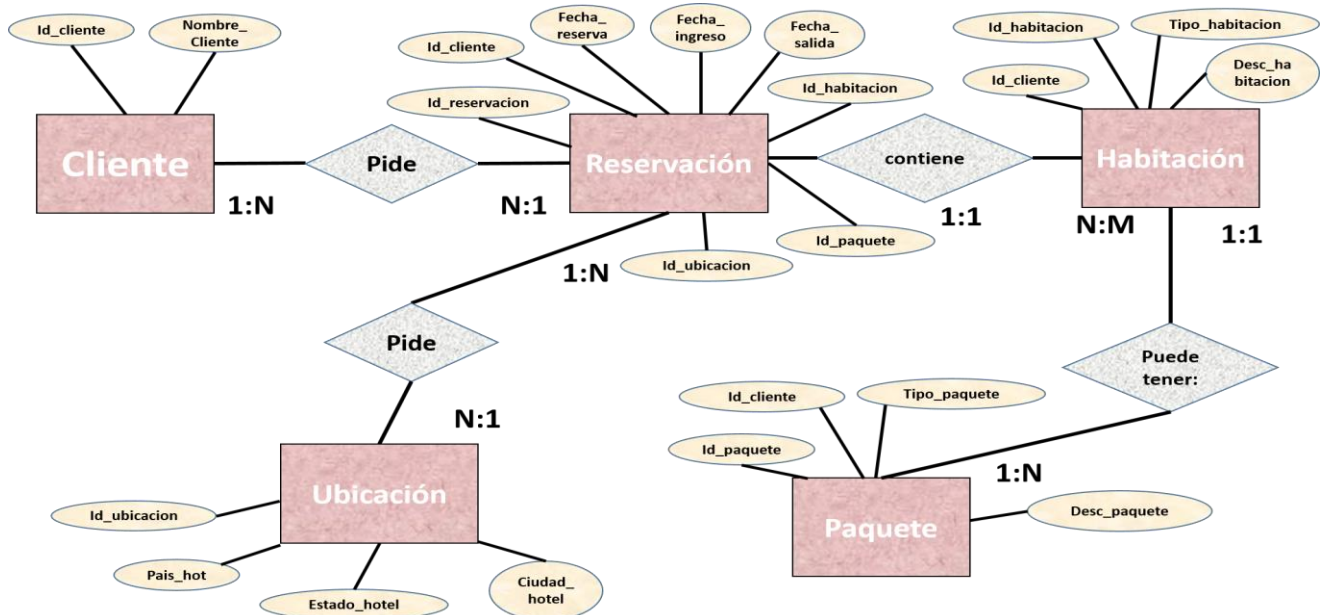
En el presente documento abarcaremos cuatro temas que van encaminados a los conocimientos que nos ha dejado la carrera a lo largo de los años, aunque se elige en noveno semestre la especialidad, semestres antes se van viendo estas especialidades como una entrada a lo que la especialidad tendría, gracias a estas es que los estudiantes pueden escoger libremente su especialidad, ya que van con la seguridad de que probaron las diferentes especialidades en la carrera.

A lo largo de este archivo podremos ser testigos de los cuatro temas que se irán desarrollando poco a poco, logrando capturar un poco de la esencia de lo importante que es cada uno para un estudiante, específicamente, un programador.

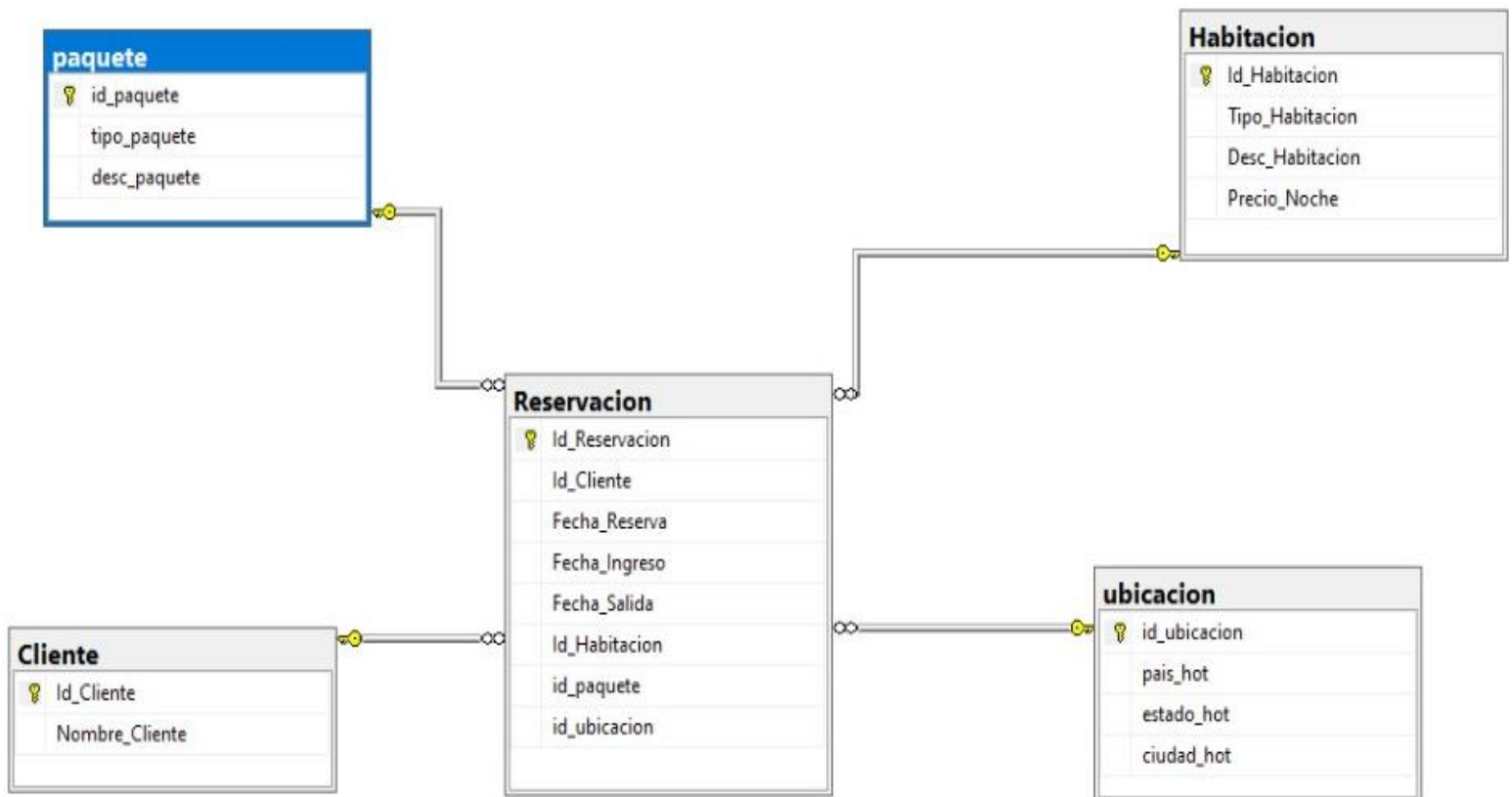
Se mostrarán evidencias del porque es importante cada tema en los estudiantes de programación, las bases de datos, redes, sistemas operativos y aplicaciones son temas que se han abarcado con anterioridad en la carrera.

# Bases de Datos

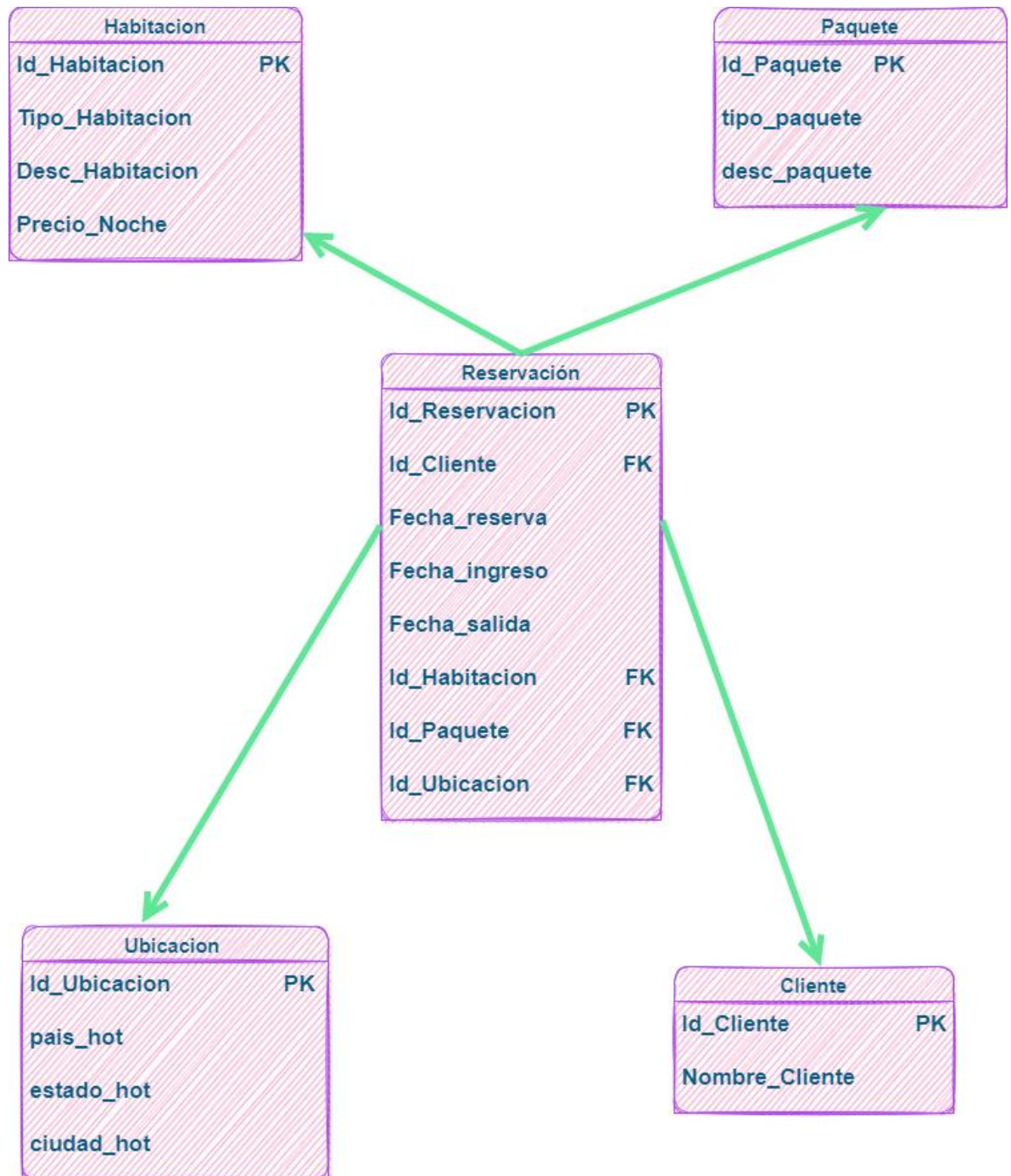
## 5 Entidades



## Modelo Entidad Relación



## Modelo Relacional



## Diccionario de Datos

Campo	Tamaño	Tipo de Dato	Descripción
<u>Cliente</u>			
Id_Cliente	8	Númeroico	Número de identificación de cada cliente
Nombre_Cliente	50	Int	Nombre completo de los clientes
<u>Habitacion</u>			
Id_Habitacion	8	Int	Número de identificación de cada habitación en el hotel
Tipo_Habitacion	50	Varchar	Descripción de las habitaciones existentes en el hotel(sencilla, matrimonial, suite etc.)
Desc_Habitacion	70	Varchar	Se describe lo que hay en una habitación así como las cosas a las que tiene acceso
Precio_Noche	50	Int	
<u>reservacion</u>			
Id_Reservacion	8	Int	Número de identificación que se da a cada una de las reservaciones hechas en el hotel
Id_Cliente	8	Int	Número de identificación de cada cliente
Fecha_Reserva	50	Datetime	Registro del día en el que se realizó la reservación
Fecha_Ingreso	50	Datetime	Registro del día en el que ingresó al hotel el cliente
Fecha_Salida	50	Datetime	Registro del día en el que se registró la salida del hotel del cliente
Id_Habitacion	8	Int	Número de identificación de cada habitación en el hotel
Id_paquete	8	Int	Agregados que se le dan a la reservación, como comida gratis o uso de limosina
Id_ubicacion	8	Int	Lugar en el que se encuentra el hotel donde se van a hospedar
<u>paquete</u>			
Id_paquete	8	Int	Agregados que se le dan a la reservación, como comida gratis o uso de limosina
Id_Cliente	8	Int	Número de identificación de cada cliente
tipo_paquete	70	Varchar	Se obtienen los datos de todos los paquetes existentes en el hotel
desc_paquete	100	Varchar	Se describe los plus y beneficios que adquiere el cliente en cada paquete seleccionado
<u>ubicacion</u>			
Id_ubicacion	8	Int	Lugar en el que se encuentra el hotel donde se van a hospedar
pais_hot	30	Varchar	País en el que se encuentran los hoteles existentes
estado_hot	30	Varchar	Estados donde existen extensiones de hotel para hospedarse
ciudad_hot	30	Varchar	ciudades donde existen extensiones del hotel

## Código SQL Creación

```
create database Hotel
```

```
create table Cliente (  
Id_Cliente int identity primary key not null,  
Nombre_Cliente varchar (50))
```

```
create table Habitacion(  
Id_Habitacion int primary key not null,  
Tipo_Habitacion varchar (50),  
Desc_Habitacion varchar (70),  
Precio_Noche int)
```

```
create table reservacion(  
Id_Reservacion int primary key not null,  
Id_Cliente int foreign key references Cliente (Id_Cliente),  
Fecha_Reserva datetime,  
Fecha_Ingreso datetime,  
Fecha_Salida datetime,  
Id_Habitacion int foreign key references Habitacion(Id_Habitacion),  
id_paquete int foreign key references paquete (id_paquete),  
id_ubicacion int foreign key references ubicacion (id_ubicacion))
```

```
create table paquete(  
id_paquete int primary key not null,  
Id_Cliente int foreign key references Cliente (Id_Cliente),  
tipo_paquete varchar (70),  
desc_paquete varchar (100))
```

```
create table ubicacion(  
id_ubicacion int primary key not null,  
pais_hot varchar (30),  
estado_hot varchar (30),  
ciudad_hot varchar (30))
```



## Código SQL Consultas Acción

SQLQuery1.sql - L...NHJ1\USUARIO (52))* X Hotel.sql - LAPTO...NHJ1\USUARIO (56))*		SQLQuery1.sql - L...NHJ1\USUARIO (52))* X Hotel.sql - LAPTO...	
select * from Habitacion		select * from Cliente	
100 %		100 %	
Results Messages		Results Messages	
Id_Habitacion	Tipo_Habitacion	Id_Cliente	Nombre_Cliente
1 101	Sencilla	1	Fatima Najera
2 102	Sencilla Doble	2	Vanessa Pedraza
3 103	Doble General	3	Yulissa Silva
4 104	Doble Matrimonial	4	Macario Gutierrez
5 105	Suite	5	Joel Alderete
6 106	Suite Presidencial	6	Adrian Escalante
7 107	Doble Matrimonial	7	Daniela Rodriguez
		8	Julia Cabrera
		9	Alejandro Zapata
Desc_Habitacion	Precio_Noche		
Planta baja, cuenta con una cama matrimonial	650		
Planta baja, cuenta con dos camas individuales	650		
Piso dos, cuenta con cama matrimonial y dos cam...	900		
Piso dos, cuenta con una cama matrimonial	1000		
Cuenta con cama king size y balcón	1500		
Cuenta con cama king size y vista al mar	1800		
Segundo piso	530		

SQLQuery1.sql - L...NHJ1\USUARIO (52))\* X Hotel.sql - LAPTO...NHJ1\USUARIO (56))\*

select \* from reservacion

100 %

Results Messages

	Id_Reservacion	Id_Cliente	Fecha_Reserva	Fecha_Ingreso	Fecha_Salida	Id_Habitacion	id_paquete	id_ubicacion
1	401	1	2020-01-12 00:00:00.000	2022-09-11 00:00:00.000	2022-11-20 00:00:00.000	102	203	304
2	402	3	2021-11-11 00:00:00.000	2022-09-23 00:00:00.000	2022-09-28 00:00:00.000	104	202	305
3	403	5	2022-01-12 00:00:00.000	2022-12-03 00:00:00.000	2022-12-15 00:00:00.000	106	205	303
4	404	2	2019-09-16 00:00:00.000	2022-01-11 00:00:00.000	2022-01-23 00:00:00.000	101	204	301
5	405	4	2022-09-04 00:00:00.000	2022-09-07 00:00:00.000	2022-09-10 00:00:00.000	104	202	305

SQLQuery1.sql - L...NHJ1\USUARIO (52))\* X Hotel.sql - LAPTO...NHJ1\USUARIO (56))\*

select \* from paquete

100 %

Results Messages

	id_paquete	tipo_paquete	desc_paquete
1	201	Familiar	Cuenta con una habitación doble general y un paseo turístico a dos lugares históricos
2	202	Matrimonial	Cuenta con una habitacion doble matrimonial y una cena gratis en el Hotel
3	203	Amigos	cuenta con una habiatacion sencilla doble y entrada a bares del hotel gratis
4	204	Sencilla	cuenta con una habitacion sencilla y comidas gratis en el hotel durante toda la estadia
5	205	Elegante	cuenta con una habitacion suite presidencial, comida gratis y acceso a limosina del hotel

SQLQuery1.sql - L...NHJ1\USUARIO (52))\* X Hotel.sql - LAPTO

select \* from ubicacion

100 %

Results Messages

	id_ubicacion	pais_hot	estado_hot	ciudad_hot
1	301	México	Coahuila	Monclova
2	302	México	Hidalgo	Pachuca
3	303	México	Nuevo Leon	Monterrey
4	304	México	Nuevo Leon	Montemorelos
5	305	México	Coahuila	Saltillo



# Redes

## Configuración IP fija W/L

### Configuración IP fija w

[illegible][illegible]

## Configuración IP fija L

```
vanestay@vanestay-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::f00d:4a44:f060:ca0d prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:21:f9:10 txqueuelen 1000 (Ethernet)
    RX packets 44802 bytes 54283426 (54.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8787 bytes 754697 (754.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

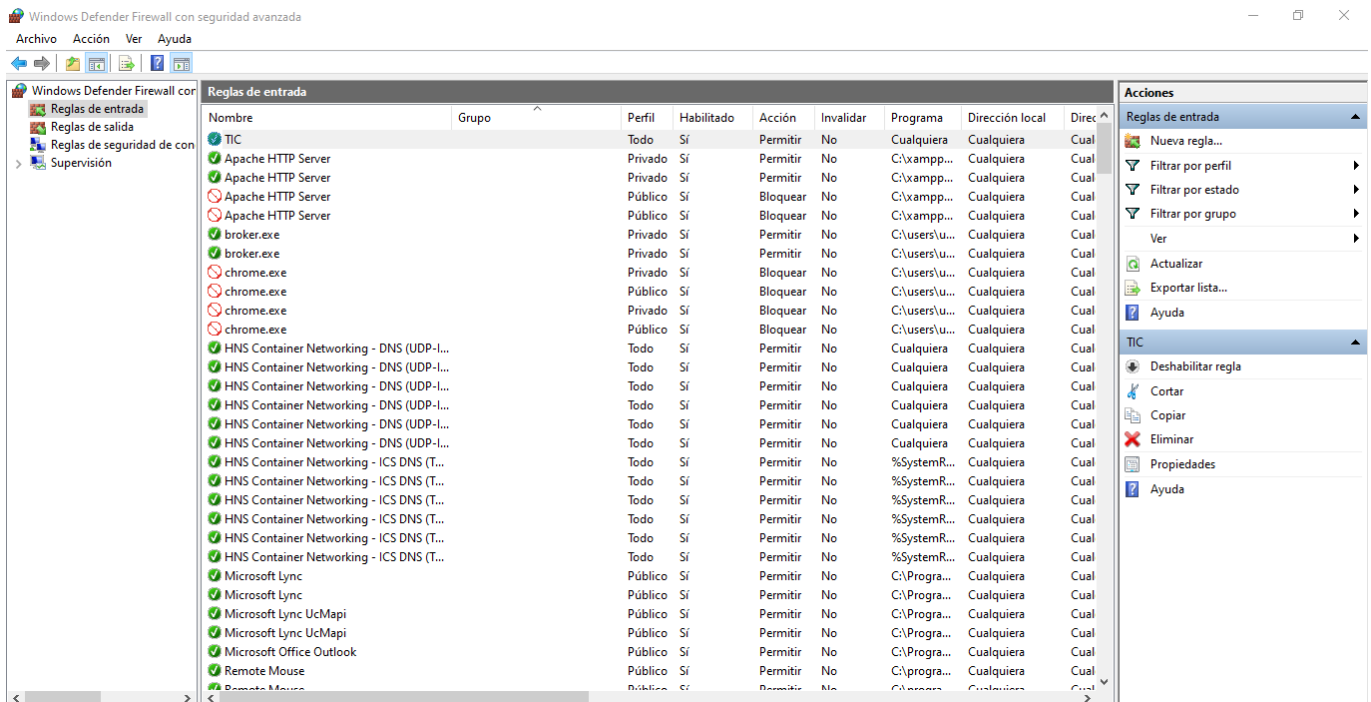
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Bucle local)
    RX packets 719 bytes 455331 (455.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 719 bytes 455331 (455.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
vanestay@vanestay-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.100 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::f00d:4a44:f060:ca0d prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:21:f9:10 txqueuelen 1000 (Ethernet)
    RX packets 44835 bytes 54288344 (54.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8893 bytes 768681 (768.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

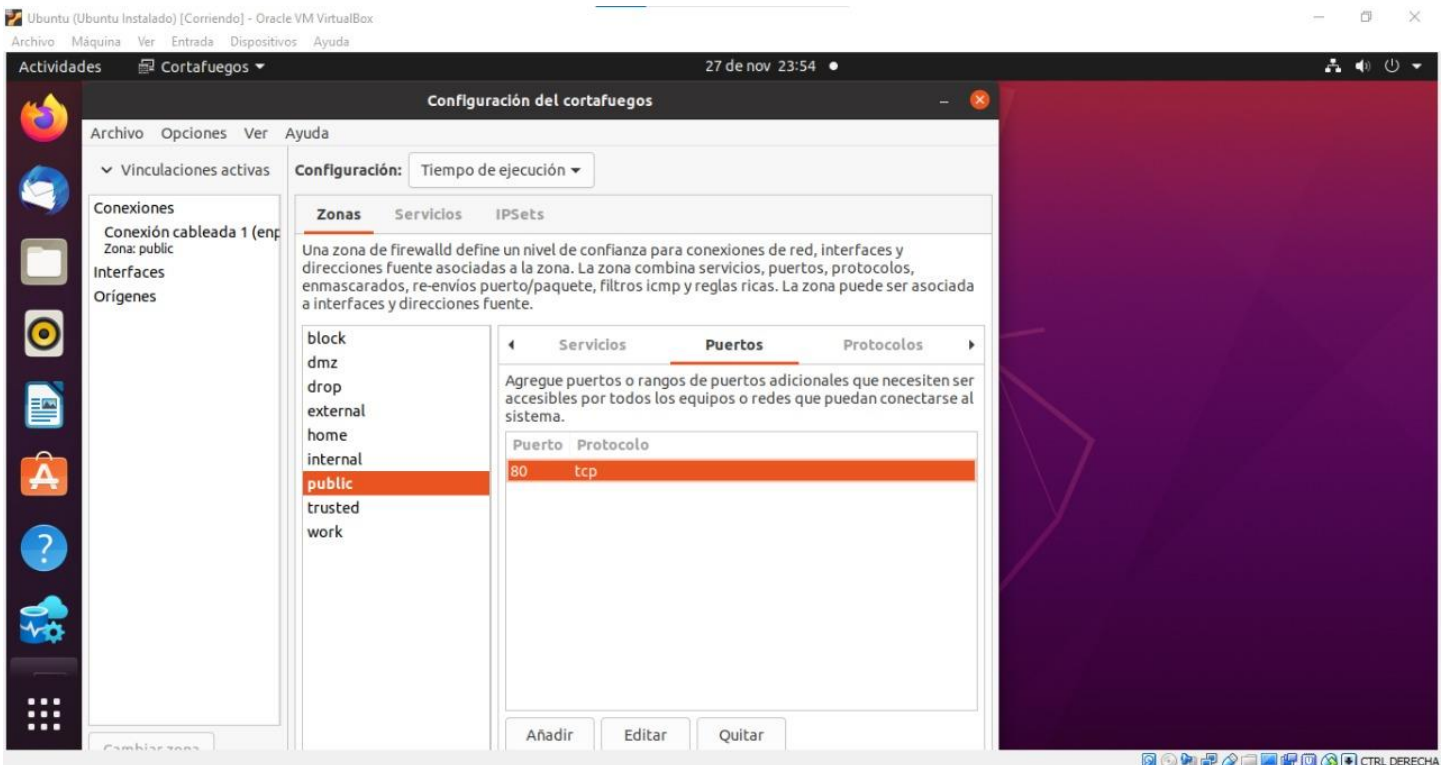
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Bucle local)
    RX packets 841 bytes 466405 (466.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 841 bytes 466405 (466.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

# Configuración de firewall por puertos W/L

## Configuración de firewall por puertos W



## Configuración de firewall por puertos L



## Prueba de ping en ambos sentidos W/L

### Prueba de ping en W

```
C:\WINDOWS\system32\cmd.exe

C:\Users\USER>ping 192.168.1.80

Haciendo ping a 192.168.1.80 con 32 bytes de datos:
Respuesta desde 192.168.1.80: bytes=32 tiempo<1m TTL=128
Respuesta desde 192.168.1.80: bytes=32 tiempo<1m TTL=128
Respuesta desde 192.168.1.80: bytes=32 tiempo<1m TTL=128
Respuesta desde 192.168.1.80: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 192.168.1.80:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms

C:\Users\USER>ping www.google.es

Haciendo ping a www.google.es [2607:f8b0:4012:808::2003] con 32 bytes de datos:
Respuesta desde 2607:f8b0:4012:808::2003: tiempo=45ms
Respuesta desde 2607:f8b0:4012:808::2003: tiempo=42ms
Respuesta desde 2607:f8b0:4012:808::2003: tiempo=42ms
Respuesta desde 2607:f8b0:4012:808::2003: tiempo=42ms

Estadísticas de ping para 2607:f8b0:4012:808::2003:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 42ms, Máximo = 45ms, Media = 42ms

C:\Users\USER>
```

### Prueba de ping en L

```
vanestay@vanestay-VirtualBox: ~
$ ping 10.0.2.100
PING 10.0.2.100 (10.0.2.100) 56(84) bytes of data.
64 bytes from 10.0.2.100: icmp_seq=1 ttl=64 time=0.057 ms
64 bytes from 10.0.2.100: icmp_seq=2 ttl=64 time=0.088 ms
64 bytes from 10.0.2.100: icmp_seq=3 ttl=64 time=0.088 ms
64 bytes from 10.0.2.100: icmp_seq=4 ttl=64 time=0.088 ms
64 bytes from 10.0.2.100: icmp_seq=5 ttl=64 time=0.089 ms
^C
--- 10.0.2.100 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4085ms
rtt min/avg/max/mdev = 0.057/0.082/0.089/0.012 ms
vanestay@vanestay-VirtualBox:~$
```

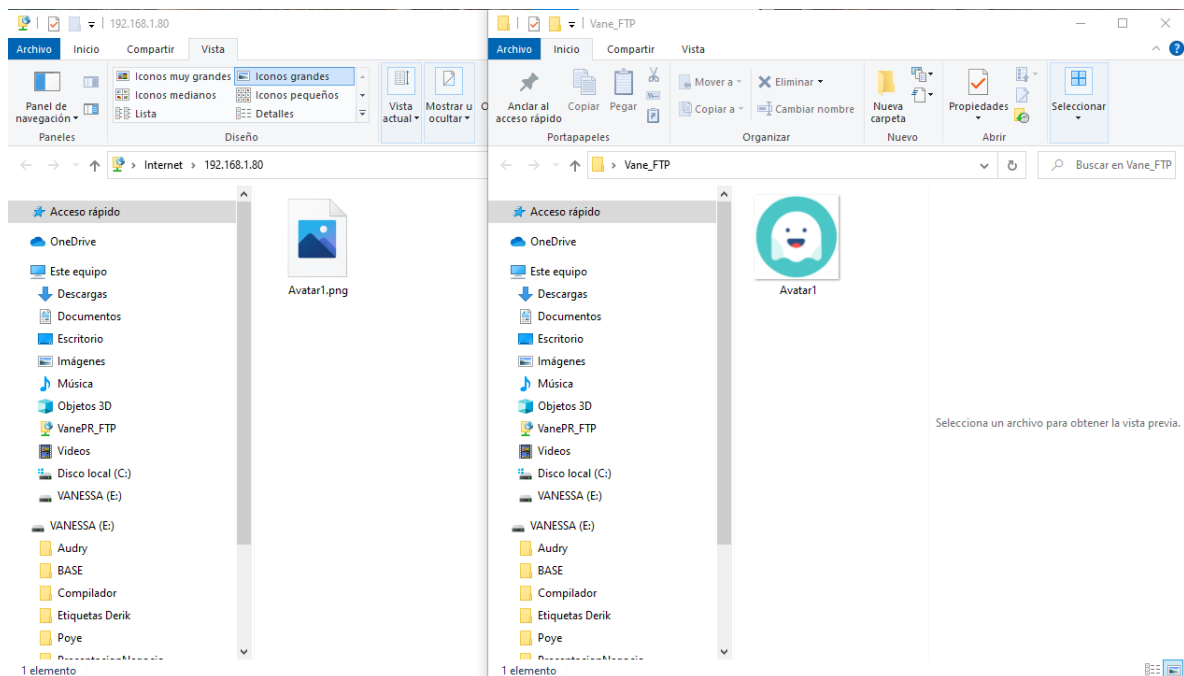
## Acceso W/L SSH

Acceso W SSH

Acceso L SSH

## Acceso W/L FTP o Similar

Acceso W FTP o Similar

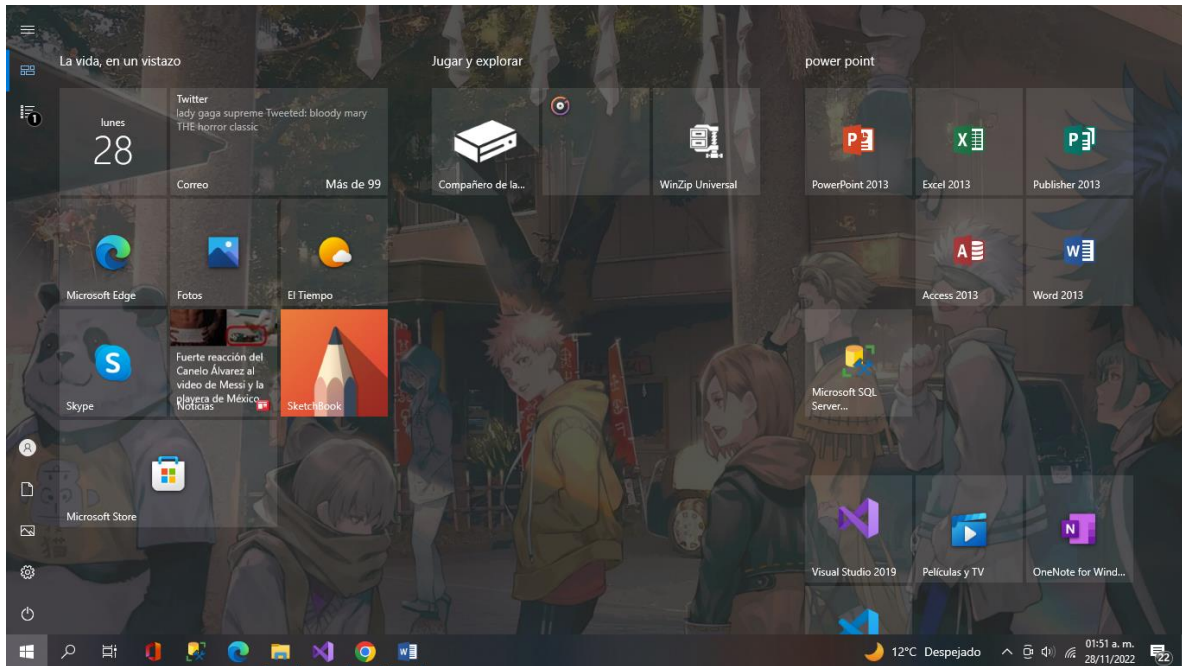


Acceso L FTP o Similar

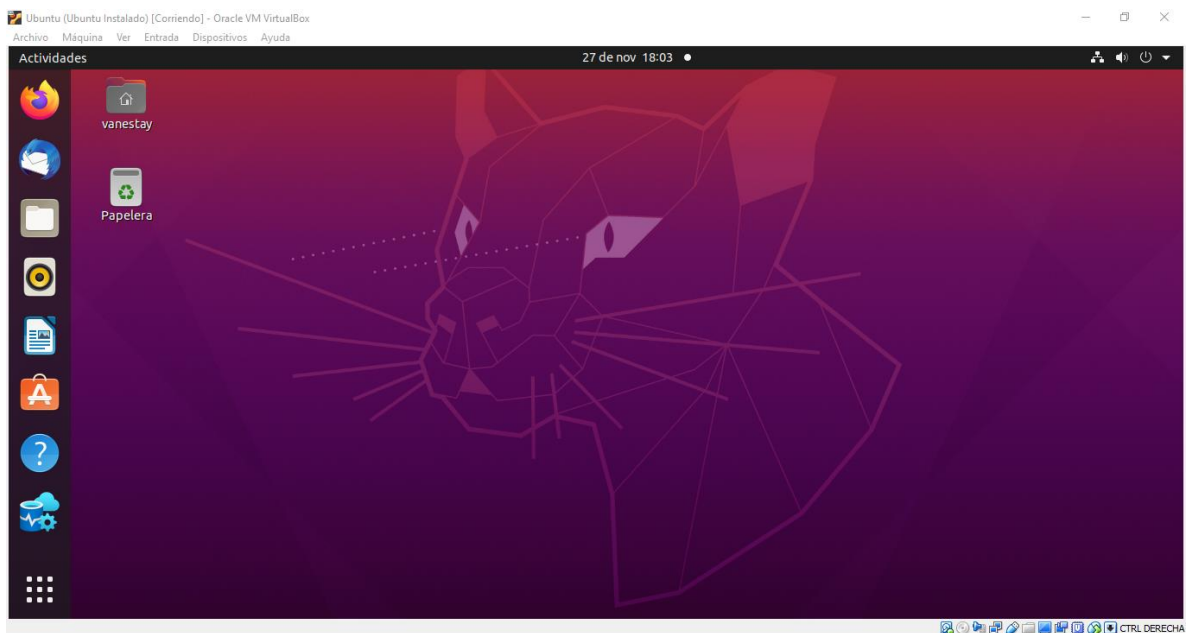
# Sistemas Operativos

SO W/L Instalado y Configurado

SO W Instalado y Configurado



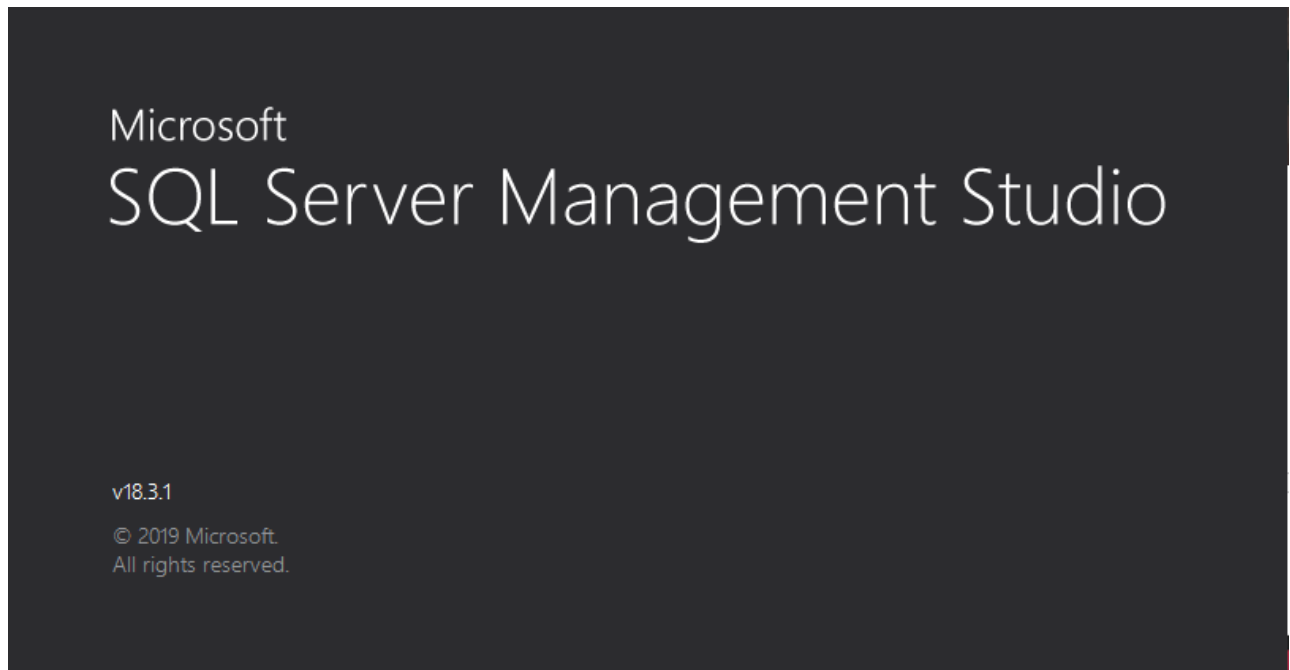
SO L Instalado y Configurado



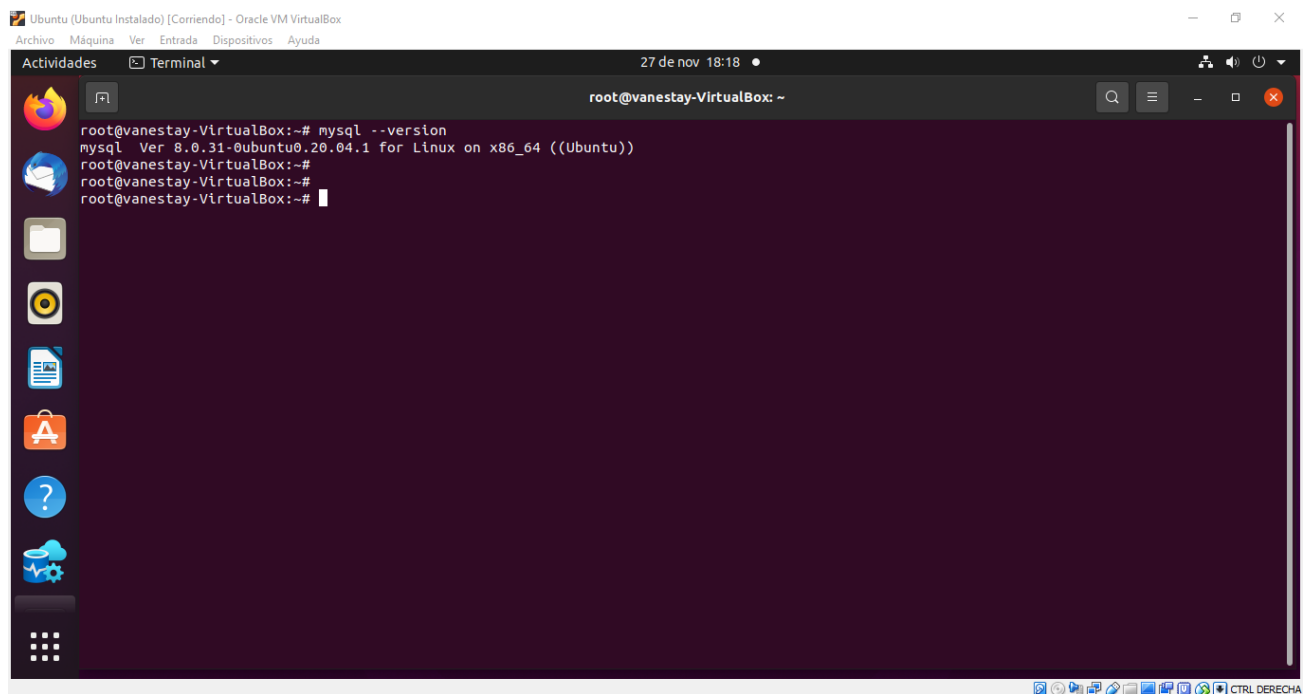


## SMDB con Cliente W/L

SMDB con Cliente W

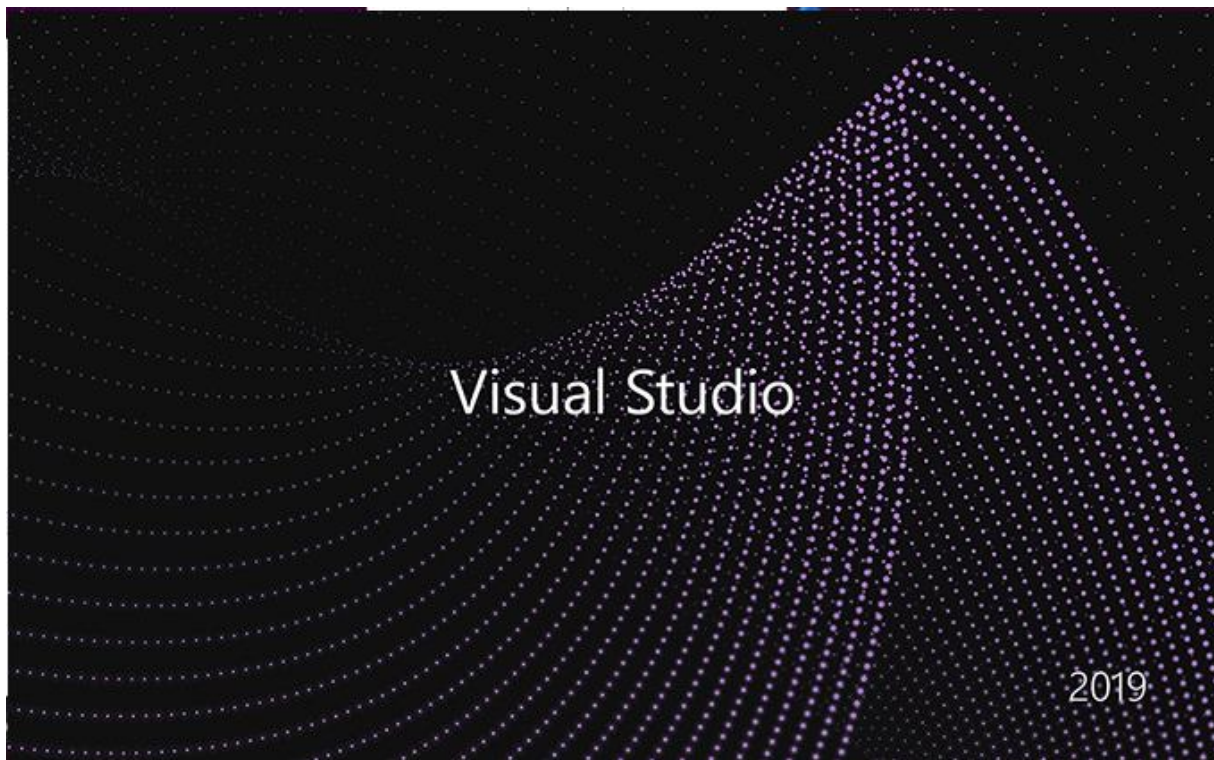


## SMDB con Cliente L

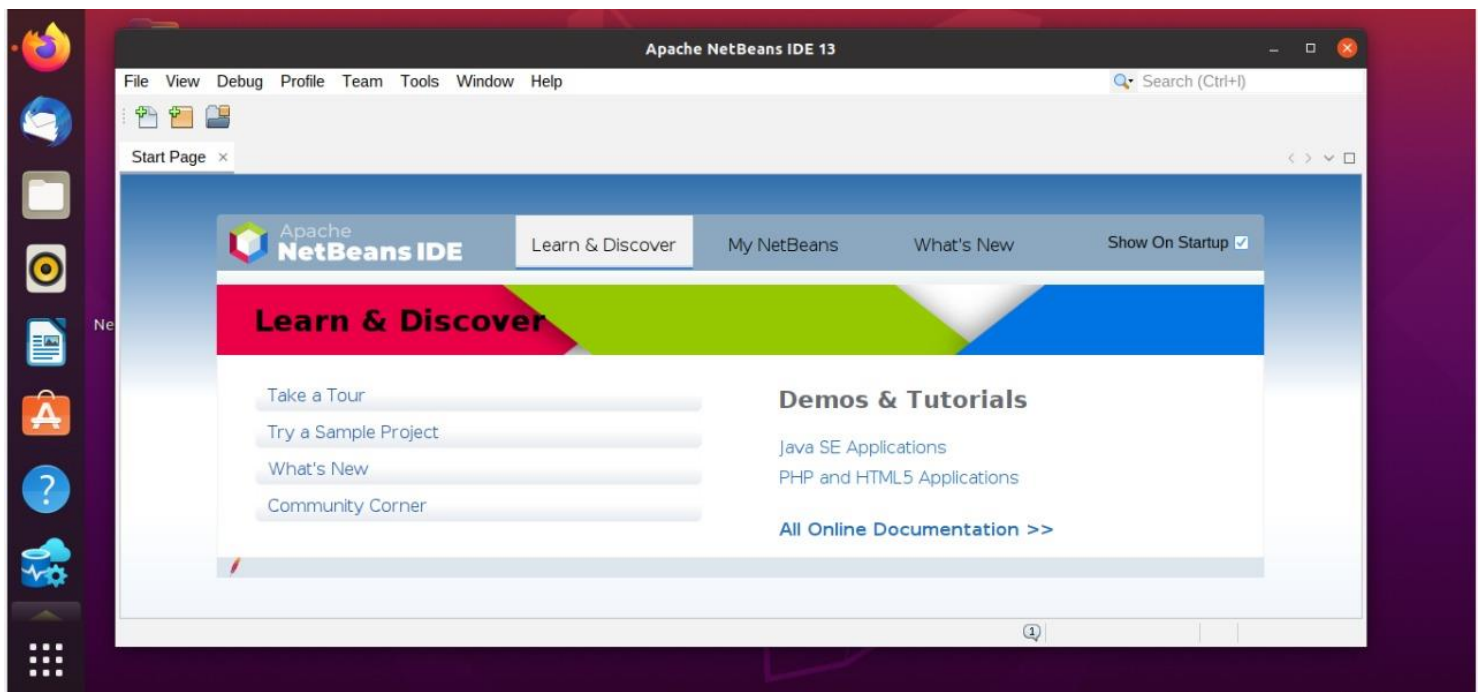


# Lenguaje de Programación e IDE W/L

## Lenguaje de Programación e IDE W



## Lenguaje de Programación e IDE L



# **Aplicación**

Conexión Acceso a la Aplicación W/L

Conexión Acceso a la Aplicación W

Uso de POO W/L

Logín W/L

Logín W



The image shows a login interface. On the left is a large logo consisting of a white outline of a hexagon containing a stylized white letter 'H' on a dark red background. To the right of the logo is a light-colored rectangular area containing the word **LOGIN** in a dark red, italicized serif font. Below this are two input fields: the first is labeled **USUARIO** and the second is labeled **CONTRASEÑA**, both in a dark red serif font. At the bottom of this area is a yellow rectangular button with the word **ACCEDER** in a dark red serif font.

CRUD para cada Tabla W

HHotelería

Reservaciones

Habitaciones

Clientes

Paquetes

Ubicaciones

Habitaciones

Habitación:  Tipo:

Descripción:

Precio:

Agregar

Actualizar

Eliminar

Buscar

Id_Habitacion	Tipo_Habitacion	Desc_Habitacion	Precio_No
101	Sencilla	Planta Baja	650
102	Sencilla Doble	Planta Baja	900
103	Doble General	Piso dos	1000

pdf

word

txt

excel

CRUD para cada Tabla L

Exportar a 5 Tipos de Archivos Diferentes

HHotelería

Reservaciones

Habitaciones

Clientes

Paquetes

Ubicaciones

Habitaciones

Habitación:  Tipo:

Descripción:

Precio:

Agregar

Actualizar

Eliminar

Buscar

Id_Habitacion	Tipo_Habitacion	Desc_Habitacion	Precio_No
101	Sencilla	Planta Baja	650
102	Sencilla Doble	Planta Baja	900
103	Doble General	Piso dos	1000

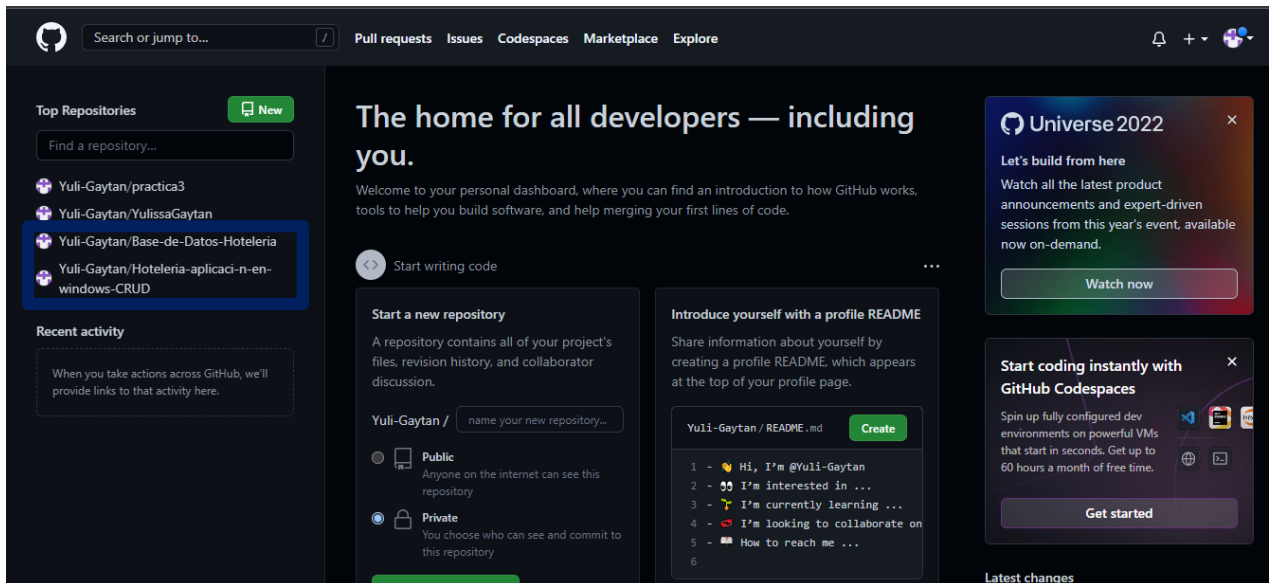
pdf

word

txt

excel

## Código Disponible en Github Público



**Link CRUD Windows:** <https://github.com/Yuli-Gaytan/Hoteleria-aplicaci-n-en-windows-CRUD.git>

**Link base de datos:** <https://github.com/Yuli-Gaytan/Base-de-Datos-Hotelaria.git>

# Conclusiones

Culminamos el trabajo con la experiencia de recordar los temas vistos en semestres anteriores y aunque algunos de ellos fueron enseñados hace años, podemos recordar las enseñanzas que cada uno de ellos nos dejó.

Primero empezamos con base de datos, esta especialidad nos enseña a diseñar una base de datos desde cero, y todo lo que ella conlleva, en este texto podemos apreciar los elementos que contiene una base de datos.

Después seguimos con las redes, que es son diferentes conexiones que se realizan manualmente, en esta podemos configurar desde direcciones ip, hasta códigos, una vez más podemos apreciar lo que nos brinda la materia en este documento.

Luego nos adentramos en un tema que no es especialidad, más sin embargo juega un papel importante en cada una de ella, el sistema operativo es el principal recurso con el que cuenta un programador para poder realizar su trabajo, sin esto no se podría hacer la gran cosa.

Por último, pero no menos importante dejamos la programación pura, que gracias a ella podemos crear aplicaciones en donde ponemos en práctica todo lo enseñado.



# Bibliografía

Charte, F. (210). *Ubuntu*. 02: 23.

Gutierrez, H. (20 de 08 de 2017). *campusMVP.es*. Obtenido de <https://www.youtube.com/watch?v=C6gT24GmNeQ&t=195s>

Karla. (07 de 09 de 2020). *Karla's Projects*. Obtenido de <https://www.youtube.com/watch?v=1Ivc7h0DJE0>

Olomos, M. D. (209). *Ubuntu Linux*. México: Ra-Ma Editorial, S.A.

Ortiz, M. (15 de 01 de 2020). *OneTronic1*. Obtenido de <https://www.youtube.com/watch?v=Efat15XYLvE>

Rodriguez, S. (09 de 07 de 2021). *Redes Plus*. Obtenido de <https://www.youtube.com/watch?v=lthfi8noKtw>

Sandoval, L. (18 de 12 de 2015). *Informatica pa novatos*. Obtenido de <https://www.youtube.com/watch?v=AK5I115Savc>