

Materia: Introducción a Bases de Datos(COM112)

Profesor: Sofia Ortiz Valenzuela

Fecha de entrega: 16/11/22

Ciclo: 1228

Nombre del proyecto: UP-SITE 2.0

Miembros del Equipo		
ID	Nombre	Carrera
0234666	Juan Manuel Pulido Moreno	ITISI
0241825	Rodrigo Lucas Nieto	IIDC
0244643	Sara Rocío Miranda Mateos	IIDC
0212511	Lorenzo Reinoso Fuentes	IIDC

Rúbricas							
ID	1-identify		4-impact	5-teams			
	IP	ASA	II	ER	TC	ME	CDTD

Contexto y Planteamiento problema

Dentro de la universidad panamericana hay tres sitios importantes para los estudiantes, **moodle**, el cual sirve para la entrega de tareas y proyectos así como para la entrega de material a los estudiantes, **UP4u**, el cual te da los datos importantes de la carrera, tales como las calificaciones, faltas, entre otras cosas. Y te permite la consulta de datos dentro de la universidad de manera limitada pero con una interfaz amigable e intuitiva. Y por último esta **UP-Site** el cual es un sitio creado con peoplesoft, el cual tiene acceso a toda la información de los estudiantes y del campus, sin embargo no tiene una interfaz como la UP4U, si no que es facil perderse y usualmente no se encuentran las respuestas que uno busca parate de que no es amigable al momento de realizar consultas importantes.

Debido a esto decidimos hacer un UPSITE, más amigable en cuanto a la funcionalidad, en el que puedes ver los registros del alumno, el servicio social, las materias que lleva e incluso las evaluaciones que se le han hecho a los profesores, para saber si es recomendable meter la materia con ellos, aparte de que te da una calificación por profesor para saber, la percepción de las personas que han cursado con el. También tenemos un registro de todas las opciones de materia, las cuales se pueden inscribir y al hacer el registro te dirán si es compatible con tu horario y si no excedes o careces de créditos por semestre, así como tener en cuenta el precio por semestre, y mes. De la misma manera nuestro programa contempla aspectos como la beca y el descuento a pago de contado, si se aplica el pago en una sola exhibición.

Propuesta de solución

Como solución a nuestro proyecto decidimos hacer una base de datos que contará con la información básica de los alumnos, así como información personal, materias, profesores, evaluaciones, becas, créditos cursando y servicio social. Con estas tablas y con ayudas de llaves foráneas podremos recopilar toda una base de datos que contenga los datos pertinentes por alumno y nos permite hacer consultas de datos que no se encuentran en las tablas de las bases de datos, por ejemplo, la cantidad de dinero mensual, el tipo de beca con el que se cuenta y si es que ya la perdiste o no, podremos consultar cuantas horas te faltan para completar tu servicio social, tu horario y podremos ver las evaluaciones de los profesores.

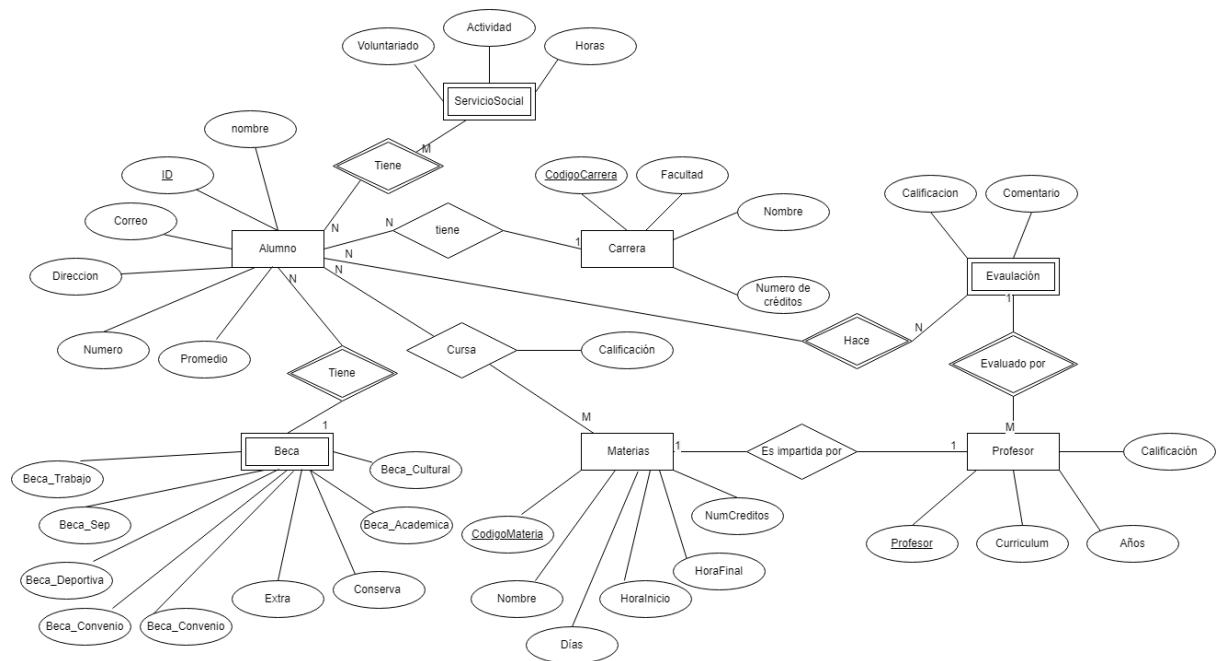
Valor de la solución

El valor de la solución está en que los estudiantes necesitan saber los costos de su carrera para hacer un presupuesto y contemplarlo en sus gastos mensuales y semestrales, y tambien es importante tener acceso a su registro para poder llevar un contro de su carrera y de su desempeño estudiantil. A su vez tambien es vital contrar con la retroalimentación de los

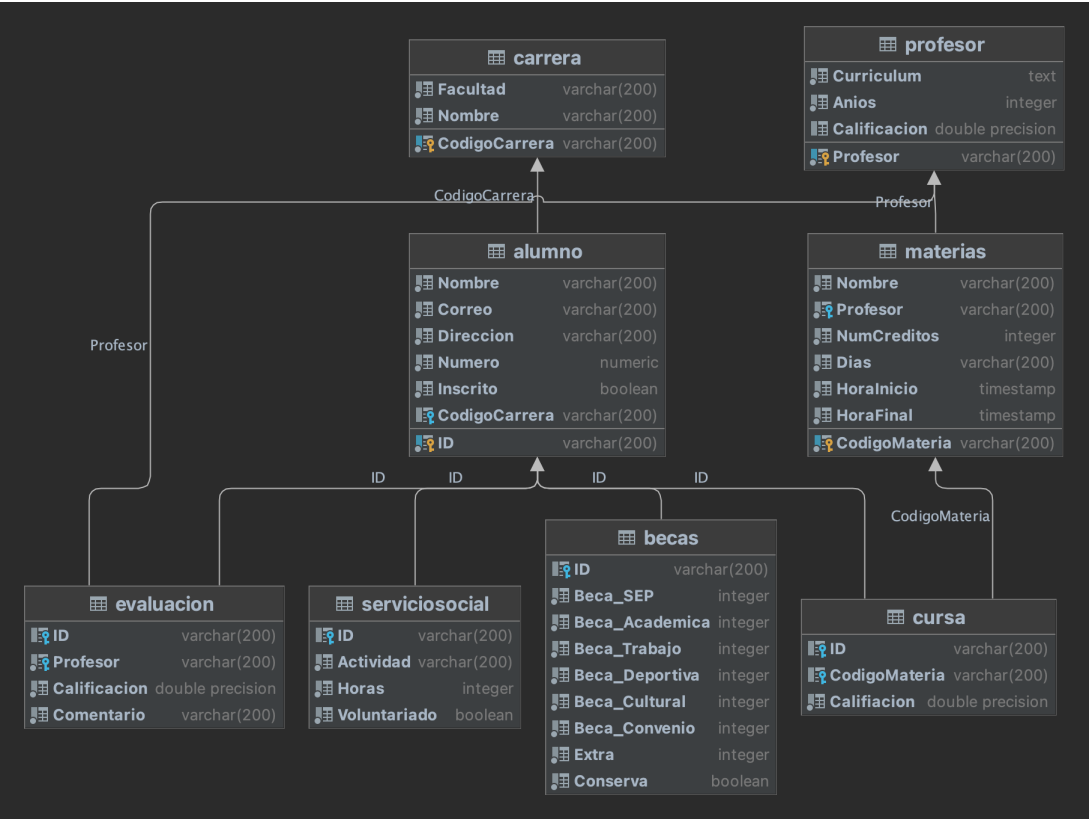
maestros para así poder tener mejores resultados y una mejora en la formación de los estudiantes

Implementación

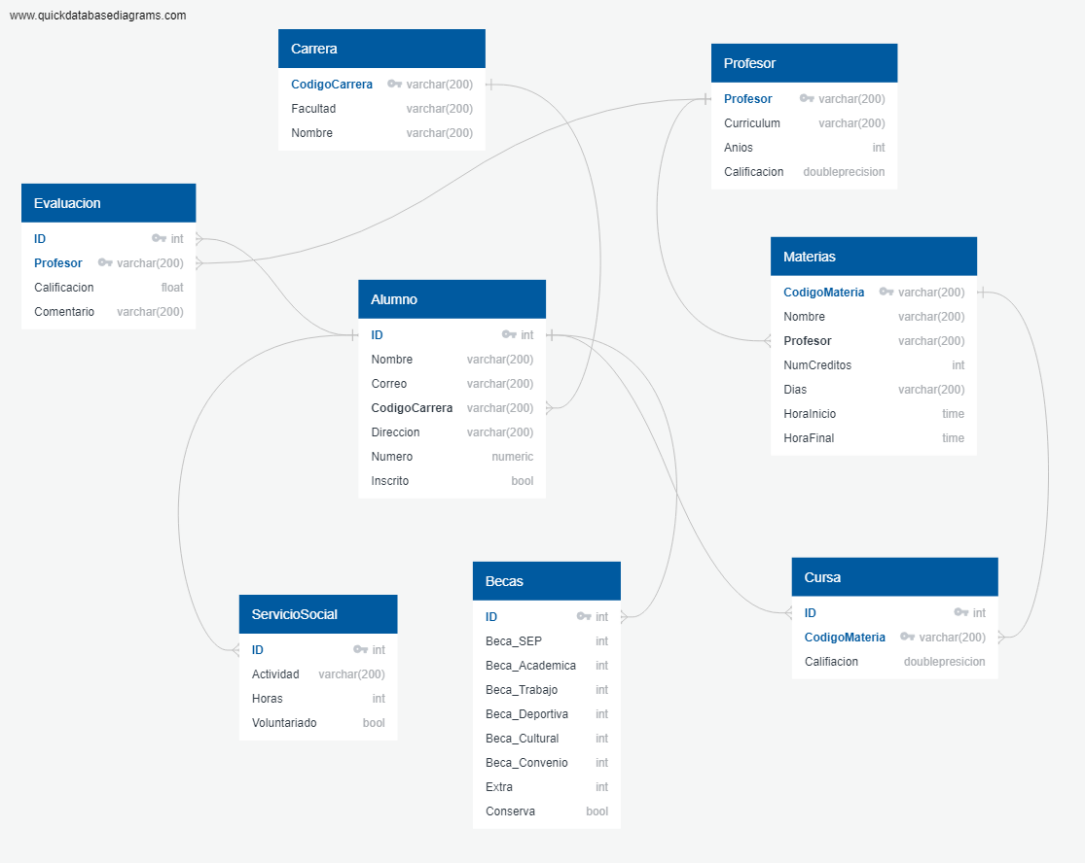
Diagrama E-R



Esquema relacional (Esquema físico)



V01



V02

Normalización

1FN => Cada campo tiene un valor único y no existen grupos repetidos

2FN => Los campos no clave dependen por completo de la llave primaria, sólo aplica para tablas con llaves compuestas.

3FN => Los campos no clave son accesibles únicamente por los campos clave.

Tabla	Dependencias Funcionales	1FN	2FN	3FN
Alumno	ID => Nombre, Correo, Direccion, Numero, Inscrito, CodigoCarrera	Ok	Ok	Ok
Becas	ID=> Beca_Sep, Beca_Academica, Beca_Trabajo, Beca_Deportiva, Beca_Cultural, Beca_Convenio, Extra, Conserva	Ok	Ok	Ok
Carrera	CodigoCarrera => Facultad, Nombre	Ok	Ok	Ok
Serviciosocial	ID=> Actividad, Horas, Voluntariado	Ok	Ok	Ok
Evaluación	ID, Profesor => Calificacion, Comentario	Ok	Ok	Ok
Cursa	ID, CodigoMateria => Calificacion	Ok	Ok	Ok
Materias	CodigoMateria => Nombre, Profesor, NumCreditos, Días, HoraInicio, HoraFinal	Ok	Ok	Ok
Profesor	Profesor => Curriculum, Años, Calificacion	Ok	Ok	Ok

Creación DB

Scripts de creación de tablas

```
Create SCHEMA up_SITE_2;  
CREATE TABLE up_SITE_2.Carrera (  
  CodigoCarrera varchar(200) PRIMARY KEY ,  
  Facultad varchar(200) NOT NULL,  
  Nombre varchar(200) NOT NULL  
);  
  
CREATE TABLE up_SITE_2.Profesor (  
  Profesor varchar(200) PRIMARY KEY,
```

```

Curriculum text NOT NULL,
Anios int NOT NULL,
Calificacion float
);
CREATE TABLE up_SITE_2.Materias (
CodigoMateria varchar(200) PRIMARY KEY ,
Nombre varchar(200) NOT NULL,
Profesor varchar(200) NOT NULL,
NumCreditos int NOT NULL,
Dias varchar(200) NOT NULL,
HoraInicio time NOT NULL,
HoraFinal time NOT NULL,
CONSTRAINT FK_Profesor FOREIGN KEY(Profesor) REFERENCES
up_SITE_2.Profesor(Profesor)
);

CREATE TABLE up_SITE_2.Alumno (
ID varchar(200) PRIMARY KEY ,
Nombre varchar(200) NOT NULL,
Correo varchar(200) NOT NULL,
Direccion varchar(200) NOT NULL,
Numero numeric NOT NULL,
Inscrito bool NOT NULL,
CodigoCarrera varchar(200),
Voluntariado bool NOT NULL,
CONSTRAINT FK_Carrera FOREIGN KEY(CodigoCarrera) REFERENCES
up_SITE_2.Carrera(CodigoCarrera)
);

CREATE TABLE up_SITE_2.Becas (
ID varchar(200) PRIMARY KEY,
Beca_SEP int NOT NULL,
Beca_Academica int NOT NULL,
Beca_Trabajo int NOT NULL,
Beca_Deportiva int NOT NULL,
Beca_Cultural int NOT NULL,
Beca_Convenio int NOT NULL,
Extra int NOT NULL,
Conserva bool NOT NULL,
CONSTRAINT FK_ID FOREIGN KEY(ID) REFERENCES up_SITE_2.Alumno(ID)
);

CREATE TABLE up_SITE_2.Cursa (
ID varchar(200),
CodigoMateria varchar(200),
Califiacion float NOT NULL,
CONSTRAINT FK_ID FOREIGN KEY(ID) REFERENCES up_SITE_2.Alumno(ID),
CONSTRAINT FK_Materia FOREIGN KEY(CodigoMateria) REFERENCES
up_SITE_2.Materias(CodigoMateria)
);

CREATE TABLE up_SITE_2.Evaluacion (

```

```

ID varchar(200),
Profesor varchar(200),
Calificacion float NOT NULL,
Comentario varchar(200) NOT NULL,
CONSTRAINT FK_ID FOREIGN KEY(ID) REFERENCES up_SITE_2.Alumno(ID) ,
CONSTRAINT FK_Profesor FOREIGN KEY(Profesor) REFERENCES
up_SITE_2.Profesor(Profesor)
);

CREATE TABLE up_SITE_2.ServicioSocial (
ID varchar(200),
Actividad varchar(200) PRIMARY KEY ,
Horas int NOT NULL,
CONSTRAINT FK_ID FOREIGN KEY(ID) REFERENCES up_SITE_2.Alumno(ID)
);

```

Scripts para insertar registros en DB

```

--CARRERA
INSERT INTO up_site_2.Carrera (CodigoCarrera, Facultad, Nombre) VALUES ('IID',
'INGENIERIAS', 'Innovación y Diseño');
INSERT INTO up_site_2.Carrera (CodigoCarrera, Facultad, Nombre) VALUES ('ECM',
'EMPRESARIALES', 'COMUNICACION');
INSERT INTO up_site_2.Carrera (CodigoCarrera, Facultad, Nombre) VALUES ('DER',
'HUMANIDADES', 'DERECHO');
INSERT INTO up_site_2.Carrera (CodigoCarrera, Facultad, Nombre) VALUES ('MC',
'SALUD', 'MEDICINA');
INSERT INTO up_site_2.Carrera (CodigoCarrera, Facultad, Nombre) VALUES ('IMT',
'INGENIERIAS', 'MECATRONICA');
INSERT INTO up_site_2.Carrera (CodigoCarrera, Facultad, Nombre) VALUES ('ENI',
'EMPRESARIALES', 'NEGOCIOS INTERNACIONALES');
-- PROFESOR
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Victoria Carreras', 'Ingeniero Ciencias Computacionales', '49', '9.8');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Sofia Ortiz', 'DBA', '28', '10');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Christian Coronel', 'Ingeniero TI', '34', '4');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Victor Isolino', 'Filosofo', '40', '8');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Rocio Mateos', 'Medico Pediatra', '45', '10');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Monica Fuentes', 'Dentista', '40', '10');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Juan Pulido', 'Ingeniero Ciberseguridad', '82', '2');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES

```

```

('Rodrigo Nieto', 'Abogado', '75', '1');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Diego Trevilla', 'Comunicologo', '30', '7');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Jyaru Lecona', 'Licenciado', '52', '10');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Sara Mateos', 'Directora Ingenierias', '29', '10');
INSERT INTO up_site_2.Profesor (Profesor, Curriculum, Anios, Calificacion) VALUES
('Lorenzo Fuentes', 'Asistente', '35', '9');

```

-- Materias

```

INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('COM01', 'Programacion de Algoritmos', 'Victoria
Carreras', 8, 'LUN-MIER-VIER', '8:30:00', '10:00:00');
INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('COM03', 'Intro Bases de Datos', 'Sofia Ortiz', 8,
'LUN-MIER-VIER', '8:30:00', '10:00:00');
INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('COM04', 'Python Básico', 'Christian Coronel', 8,
'LUN-MIER-VIER', '8:30:00', '10:00:00');
INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('HUM01', 'Ética', 'Victor Isolino', 8, 'LUN-MIER-VIER',
'8:30:00', '10:00:00');
INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('MED02', 'Histologia', 'Rocio Mateos', 8,
'LUN-MIER-VIER', '8:30:00', '10:00:00');
INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('MED01', 'Bioquimica', 'Monica Fuentes', 8,
'LUN-MIER-VIER', '8:30:00', '10:00:00');
INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('DER01', 'Derecho Penal', 'Rodrigo Nieto', 8,
'LUN-MIER-VIER', '8:30:00', '10:00:00');
INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('EMPC01', 'Principios de Fotografias', 'Diego Trevilla', 8,
'LUN-MIER-VIER', '8:30:00', '10:00:00');
INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('EMP02', 'Negocios', 'Jyaru Lecona', 8, 'LUN-MIER-VIER',
'8:30:00', '10:00:00');
INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('EMP01', 'Competencias Directivas', 'Sara Mateos', 8,
'LUN-MIER-VIER', '8:30:00', '10:00:00');
INSERT INTO up_site_2.Materias (CodigoMateria, Nombre, Profesor, NumCreditos, Dias,
HoralInicio, HoraFinal) VALUES ('HUM02', 'Antropologia Teologica', 'Lorenzo Fuentes', 8,
'LUN-MIER-VIER', '8:30:00', '10:00:00');

```

```

INSERT INTO up_site_2.Alumno (ID, Nombre, Correo, Direccion, Numero, Inscrito,
CodigoCarrera, Voluntariado) VALUES ('0212511', 'Lorenzo Reinos',
'0212511@up.edu.mx', 'Mier y Pesado', '5585530054', TRUE, 'IIDC', TRUE);
INSERT INTO up_site_2.Cursa (ID, CodigoMateria, Califiacion) VALUES ('0212511',
'EMP01', 5.5);
INSERT INTO up_site_2.Cursa (ID, CodigoMateria, Califiacion) VALUES ('0212511',
'COM04', 8.0);

```

```

INSERT INTO up_site_2.Carrera (CodigoCarrera, Facultad, Nombre) VALUES ('ITI',

```



```

'INGENIERIAS', 'Tecnologías de la Información');
INSERT INTO up_site_2.Alumno (ID, Nombre, Correo, Direccion, Numero, Inscrito,
CodigoCarrera, Voluntariado) VALUES ('0234666', 'Juan Manuel Pulido',
'0234666@up.edu.mx', 'Campana 3', '5582400549', TRUE, 'ITI', TRUE);
INSERT INTO up_site_2.Cursa (ID, CodigoMateria, Califiacion) VALUES ('0212511',
'EMP01', 10);
INSERT INTO up_site_2.Cursa (ID, CodigoMateria, Califiacion) VALUES ('0212511',
'COM01', 9.0);
INSERT INTO up_site_2.Cursa (ID, CodigoMateria, Califiacion) VALUES ('0212511',
'EMP02', 9.0);
INSERT INTO up_site_2.Cursa (ID, CodigoMateria, Califiacion) VALUES ('0212511',
'HUM02', 9.5);
INSERT INTO up_site_2.Cursa (ID, CodigoMateria, Califiacion) VALUES ('0212511',
'DER01', 10);
INSERT INTO up_site_2.Cursa (ID, CodigoMateria, Califiacion) VALUES ('0212511',
'EMPC01', 9.1);

INSERT INTO up_site_2.Becas (ID, Beca_SEP, Beca_Academica, Beca_Trabajo,
Beca_Deportiva, Beca_Cultural, Beca_Convenio, Extra, Conserva) VALUES ('0244643',
0, 20, 20, 0, 0, 0, 5, TRUE);
INSERT INTO up_site_2.Becas (ID, Beca_SEP, Beca_Academica, Beca_Trabajo,
Beca_Deportiva, Beca_Cultural, Beca_Convenio, Extra, Conserva) VALUES ('0212504',
0, 20, 20, 0, 0, 0, 5, TRUE);
INSERT INTO up_site_2.Becas (ID, Beca_SEP, Beca_Academica, Beca_Trabajo,
Beca_Deportiva, Beca_Cultural, Beca_Convenio, Extra, Conserva) VALUES ('0212511', 0,
20, 20, 0, 0, 20, 5, TRUE);
INSERT INTO up_site_2.Becas (ID, Beca_SEP, Beca_Academica, Beca_Trabajo,
Beca_Deportiva, Beca_Cultural, Beca_Convenio, Extra, Conserva) VALUES ('0212500',
0, 20, 20, 0, 0, 0, 5, TRUE);
INSERT INTO up_site_2.Becas (ID, Beca_SEP, Beca_Academica, Beca_Trabajo,
Beca_Deportiva, Beca_Cultural, Beca_Convenio, Extra, Conserva) VALUES ('0234666',
0, 40, 20, 0, 0, 0, 5, TRUE);
INSERT INTO up_site_2.Becas (ID, Beca_SEP, Beca_Academica, Beca_Trabajo,
Beca_Deportiva, Beca_Cultural, Beca_Convenio, Extra, Conserva) VALUES ('0212503',
0, 0, 0, 0, 0, 0, 5, TRUE);

```

Consultas SQL

```

--Muestra las becas que tiene un alumno (Lorenzo)
Select A.Nombre, A.id, b.Beca_sep, b.Beca_Academica, b.Beca_Trabajo,
b.Beca_Deportiva, b.Beca_Cultural, b.Beca_Convenio
From up_SITE_2.alumno as A
join up_SITE_2.becas b on A.id = A.id
where A.nombre = 'Lorenzo';

--Muestra el nombre y id del alumno con mayor calificacion de una materia (Calculo)
Select A.nombre, A.id ,C.Califiacion
From up_SITE_2.alumno as A
join up_SITE_2.cursa C on A.id = C.id

```

```

join up_SITE_2.materias M on C.codigomateria = M.codigomateria
where M.nombre = 'Calculo'
group by C.Califiacion,A.nombre, A.id;

--Muestra el nombre del profesor con la mayor calificacion de una materia (IBD)
Select P.Profesor, max(E.calificacion)
From up_SITE_2.profesor as P
join up_SITE_2.evaluacion E on P.profesor = E.profesor
join up_SITE_2.materias M on P.profesor = M.profesor
where M.nombre = 'Introduccion a Base de Datos'
group by P.Profesor;

--Muestra todos los alumnos de alguna carrera (Inteligencia de Datos)
Select A.nombre
from up_SITE_2.alumno as A
join up_SITE_2.carrera as C on A.CodigoCarrera = C.codigocarrera
where C.nombre = 'Inteligencia de Datos';

--Muestra a los alumnos con pocas horas de servicio
Select A.nombre, serv.horas
from up_SITE_2.alumno as A
join up_SITE_2.serviciosocial serv on A.id = serv.id
where serv.horas <= 10;

```

Funciones

```

CREATE OR REPLACE FUNCTION calcular_pago(
tipo_Pago VARCHAR(200),
ID_I varchar(200))
    RETURNS float
    LANGUAGE plpgsql
    AS $$
    DECLARE
        beca_total int;
        creditos int;
        pago float;
    BEGIN
        IF (Select B.conserva FROM up_SITE_2.becas B WHERE B.id = ID_I) THEN
            beca_total = (
                SELECT B.beca_academica + B.beca_convenio + B.beca_cultural +
                B.beca_deportiva + B.beca_sep + B.beca_trabajo + B.Extra
                From up_SITE_2.becas as B
                Where B.id = ID_I );
        else
            beca_total=0;
        end if;
    
```

```

        creditos = (
            SELECT sum(M.numcreditos)
            FROM up_SITE_2.materias M
            JOIN up_SITE_2.cursa C on M.codigomateria = C.CodigoMateria
            WHERE C.ID = ID_I
        );
        IF (22>creditos and creditos>42) THEN
            RAISE NOTICE 'Revisa tus materias tus creditos no son validos. Creditos:
%',creditos;
            Return NULL;
        end if;
        IF tipo_Pago = 'contado' THEN
            beca_total = beca_total + 5;
        end if;
        pago = 2873 * creditos * (beca_total/100);
        IF tipo_Pago = 'contado' THEN
            Raise NOTICE 'Se hará un pago de contado de % ',pago;
        else
            pago = pago/5;
            Raise NOTICE 'Se harán 5 un pagos de % ',pago;
        end if;

        RETURN pago;
    END;
    $$;

```

```

CREATE OR REPLACE FUNCTION recomendar_Profesor(
materia VARCHAR(200))
    RETURNS varchar(200)
    LANGUAGE plpgsql
    AS $$
    DECLARE
        Profesor varchar(200);
    BEGIN
        profesor = (SELECT P.profesor
        From up_SITE_2.profesor P
        JOIN up_SITE_2.Materias M on P.profesor = M.profesor
        WHERE M.nombre = materia and calificacion = (
            SELECT max(P.calificacion)
            From up_SITE_2.profesor P
            JOIN up_SITE_2.Materias M on P.profesor = M.profesor
            WHERE M.nombre = materia));
        Return Profesor;
    END;
    $$;

```

Resultados y conclusiones

Al hacer nuestra propia base de datos nos encontramos con los problemas de que no sabíamos cómo hacer consultas al 100%, es por ello por lo que comenzamos a investigar y aprender más de lo que ya sabíamos. Logramos trabajar en equipo y separar los tasks de una forma equitativa. Concluimos como equipo que las bases de Datos son muy útiles y nos pueden hacer procesos más eficientes, nos dimos cuenta de que SQL es un lenguaje de programación que es muy completo ya que a pesar de solo funcionar para base de datos nos permite hacer funciones y optimizar procesos más eficientemente que cualquier otro lenguaje de programación. Al comenzar este proyecto nadie quería hacer los diagramas de Entidad Relación ya que lo veíamos como una tarea tediosa por ello decidimos hacer esta tarea todos juntos y nos dimos cuenta en lo fácil que fue pasar las tablas a nuestro query después de tener nuestro diagramas de entidad relación ya hechos, descubrimos que la parte visual de las bases de datos a pesar de no verse muy importantes son la parte más importante para desarrollar nuestra base de datos.