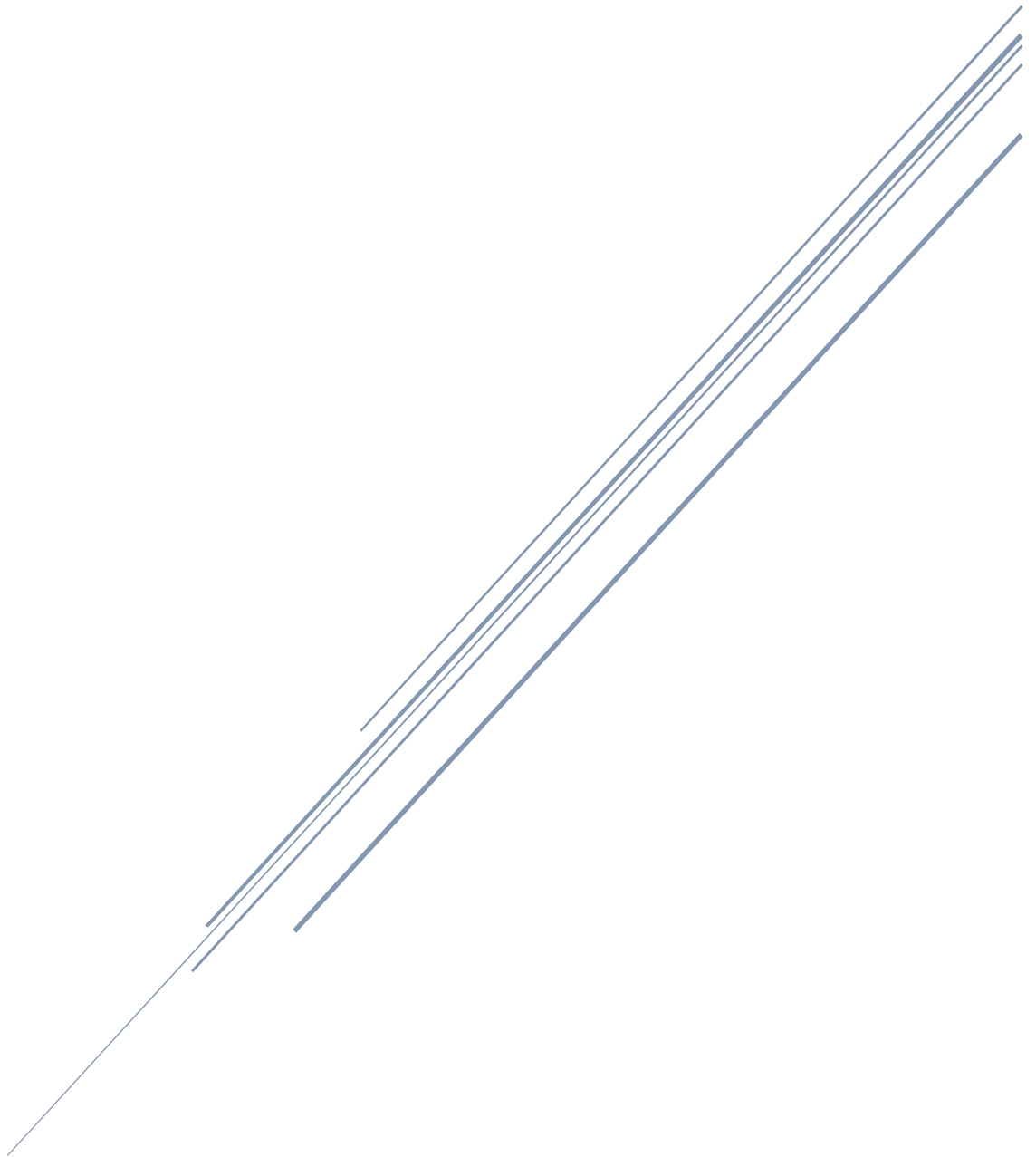


Documentación del proceso de desarrollo

Proyecto Inicial



Trabajo de Fin de Grado
Víctor García Murillo

Índice

Entrega de propuestas	3
Aplicación para gestionar los datos sobre los circuitos de la Fórmula 1	3
Plataforma de gestión de información para empresas de mercancías peligrosas.....	4
Aplicación para la gestión de portátiles del centro de estudios Ceinmark...	5
Aplicación para citar pacientes en un hospital	6
Aplicación para gestionar las competiciones de un karting	7
Selección de una propuesta	9
Creación del diagrama entidad relación	9
Conjuntos de entidades y sus respectivos atributos	9
Relaciones y cardinalidad entre los elementos	10
Creación del diagrama relacional estático.....	11
Tablas	11
Implementación de la base de datos en MySQL.....	13
Fases	13
Tipos de datos de los campos.....	15
Generación del diagrama de la base de datos.....	16
Pasos	16
Generación de datos	16
Datos seleccionados	16
Datos generados por expresiones regulares (regex)	17
Datos generados mediante archivos de texto	17
Generación de documentación automática	19
Pasos	19
Creación de la copia de seguridad de la base de datos.....	19
Pasos	19
Propuestas de ampliación	20
Sistema de pago mensual para el alquiler de portátiles	20
Gestión de regletas	20
Sistema de gestión de conexiones WiFi.....	21

Entrega de propuestas

En la etapa inicial, llevé a cabo una depuración de las propuestas que había estado contemplando durante el verano, descartando aquellas que no alcanzaban un nivel de desarrollo adecuado. Tomando como base el ejemplo proporcionado en la presentación, procedí a definir tanto las propuestas como sus respectivas reglas de negocio.

Las tres primeras propuestas que consideré fueron las siguientes:

Aplicación para gestionar los datos sobre los circuitos de la Fórmula 1

Se desea desarrollar un sistema de información para la gestión de los datos de los circuitos, pilotos, escuderías, grandes premios y temporadas.

- De un circuito, se busca almacenar la siguiente información: país, nombre, código, número de carreras celebradas en el circuito, ciudad, fecha de inauguración, cantidad de curvas, longitud y número de actualizaciones.
- En cuanto a los pilotos, se pretende almacenar su nombre, país de origen, fecha de nacimiento, debut en la Fórmula 1, apellido, abreviatura de su apellido, y sus premios ganados, que se considera un atributo multivaluado.
- Se desea establecer una relación donde varios pilotos pueden ganar en un mismo circuito, y un piloto puede alcanzar la victoria en varios circuitos. De esta relación, se almacenaría la fecha de la carrera y la posición en la que el piloto terminó en el podio (1º, 2º, 3º) solo si lograron este logro.
- Los circuitos pueden albergar múltiples Grandes Premios a lo largo de su historia, y un Gran Premio puede tener lugar en diferentes circuitos.
- Respecto a las escuderías, se quiere almacenar su nombre, fecha de fundación y país de origen.
- Un piloto ha podido pertenecer a varias escuderías durante su carrera, y las escuderías pueden contar con varios pilotos en su plantilla.
- En una temporada, se llevan a cabo múltiples Grandes Premios, y un mismo Gran Premio puede tener lugar en diversas temporadas.
- De cada temporada, se conserva el año, así como las fechas de inicio y finalización de cada temporada.
- Un piloto puede ganar múltiples temporadas, mientras que una temporada solo puede ser conquistada por un piloto, de la misma manera que ocurre con una escudería.

Plataforma de gestión de información para empresas de mercancías peligrosas

Se desea crear una aplicación web especializada en la gestión de documentos y noticias para usuarios que se dedican al transporte de mercancías peligrosas. Esta plataforma permitirá a los usuarios acceder de manera sencilla a información relevante sobre las mercancías que transportan, a través de noticias y carpetas específicas.

Con esta aplicación, los usuarios podrán estar al tanto de las últimas noticias relacionadas con las mercancías peligrosas que manejan, así como organizar de manera eficiente la documentación crucial que acompaña a su actividad.

- Se pretende almacenar información específica de las noticias, incluyendo su número de identificación, fecha de publicación, título y una breve descripción.
- Deseamos registrar información de los usuarios, incluyendo su identificación (DNI o CIF), nombre completo o nombre de la empresa, tipos de mercancías y dirección de correo electrónico.
- Una noticia puede ser accedida por múltiples usuarios, y a su vez, un usuario tiene la posibilidad de leer varias noticias.
- La aplicación contendrá varias carpetas de almacenamiento, y se desea conservar información sobre cada una de ellas, incluyendo su nombre, el tipo de mercancía peligrosa asociada y un código identificativo único.
- Múltiples usuarios pueden acceder a una misma carpeta de almacenamiento, y a su vez, un usuario tiene la capacidad de acceder a varias carpetas diferentes. Dentro de las carpetas, es posible guardar también otras carpetas.
- Los archivos se guardarán en carpetas, permitiendo que múltiples archivos se almacenen en una misma carpeta y que un archivo pueda estar presente en varias carpetas. En relación a los archivos, deseamos conservar información como su nombre, un código de identificación único y su extensión.
- La aplicación contará con notificaciones, y de ellas se pretende almacenar la fecha y el número identificativo de la notificación.
- Una notificación puede ser enviada a múltiples usuarios, y, por otro lado, un usuario puede recibir varias notificaciones.

Aplicación para la gestión de portátiles del centro de estudios Ceinmark

El objetivo de este proyecto es optimizar la gestión de los portátiles en mi centro educativo, que actualmente se caracteriza por su simplicidad y falta de automatización. Esta deficiencia conlleva confusiones y pérdidas de información en el proceso.

El proyecto se basará en una aplicación que no solo administrará la información relacionada con los dispositivos informáticos que el centro proporciona a los estudiantes, sino que también gestionará los datos de los propios alumnos y los recursos de almacenamiento disponibles para estos dispositivos electrónicos.

Esta aplicación se diseñará como una plataforma web, lo que garantizará su accesibilidad desde diversos dispositivos, ya sea un móvil o un ordenador, sin verse limitada por el sistema operativo utilizado.

A continuación, se presentará el sistema sobre el cual se construirá la aplicación:

- De los portátiles, se quiere saber, el modelo, procesador, memoria RAM, tipo de memoria ROM y capacidad de la misma, marca, código único, aplicaciones instaladas, si esta alquilado y si está en reparación.
- De los cargadores se quiere almacenar su número de serie (o código), potencia y su estado de alquiler.
- Cada portátil utiliza un cargador, y un mismo cargador puede ser utilizado por varios portátiles.
- En cuanto a los ratones, se busca almacenar su modelo, marca, código, tipo de conector y su estado de alquiler.
- Para los estudiantes, se desea guardar su DNI, nombre completo (nombre, primer apellido y segundo apellido), curso, tutor, aula correspondiente y turno de estudio (mañana o tarde).
- Cada estudiante tiene la opción de alquilar un portátil, cargador y ratón, pudiendo alquilar varios de estos simultáneamente. Asimismo, un portátil, cargador o ratón puede ser alquilado por varios estudiantes a la vez, siempre que no tengan el mismo turno.
- Se quiere almacenar de cada lugar de almacenamiento, su aula correspondiente, capacidad.
- Cada lugar de almacenamiento tiene la capacidad de guardar varios portátiles, mientras que un único portátil, ratón o cargador puede ser guardado en un solo lugar de almacenamiento.

Posteriormente, me puse en contacto con mi tutora para detectar posibles errores o deficiencias en cada una de las propuestas.

Durante la corrección de las propuestas, identifiqué que dos de las tres propuestas no cumplían con los requisitos mínimos. En consecuencia, opté por seleccionar la tercera y última propuesta relacionada con el alquiler de portátiles.

Dado que solo disponía de una propuesta, decidí idear dos alternativas adicionales en caso de que necesitara una opción adicional en el futuro.

Las siguientes son las propuestas adicionales:

Aplicación para citar pacientes en un hospital

El propósito de este proyecto es optimizar la gestión de citas de pacientes, con el fin de simplificar esta tarea. Debido a que actualmente, las aplicaciones diseñadas para este propósito carecen de intuición y ofrecen interfaces precarias.

El proyecto se basará en una aplicación que permitirá administrar las citas de los pacientes, coordinando de manera eficiente todas las partes involucradas en el proceso de consulta, incluyendo a médicos y enfermeros.

Esta aplicación se diseñará como una plataforma web, garantizando así su accesibilidad desde diversos dispositivos, ya sea un teléfono móvil o un ordenador, sin importar el sistema operativo que se utilice.

A continuación, se presentará el sistema sobre el cual se construirá la aplicación:

- En el sistema se almacena información de los pacientes, incluyendo su DNI, nombre, apellidos, fecha de nacimiento, número de seguridad social, dirección (calle, dirección y código postal), número de teléfono y el ambulatorio al que están asociados.
- Un paciente es asignado a un médico, y a su vez, cada médico puede tener varios pacientes asignados. Se desea almacenar la información del nombre del médico y los días de la semana en los que presta servicio a los pacientes.
- Un paciente será citado para una consulta en una sala médica, en una sala médica se atenderá a varios pacientes en una misma jornada. En esta relación, es importante registrar la fecha de la consulta.
- Los médicos pertenecen a una sala médica, de la cual se quiere conservar el código identificativo de cada sala. Cada sala médica cuenta con varios médicos, y cada uno de ellos atiende a los pacientes en determinados días de la semana. Además, deseamos registrar el número de pacientes atendidos en el día actual junto con el código único de cada sala médica.

- Las salas de enfermería en las que se atienden a los pacientes con cita están condicionadas por el código de la sala médica correspondiente, identificadas como (Sala médica → 1, Sala de enfermería → 1E).
- A una sala de enfermería son asignados varios enfermeros, y a su vez, cada enfermero está asignado en una sala de enfermería específica. En relación a estas salas de enfermería, se almacena el número de pacientes atendidos en el día actual y su código de identificación.
- Cada paciente será citado en una sala de enfermería, y en dicha sala, se atenderá a varios pacientes.
- De los enfermeros, se guardará su nombre y los días de la semana en los que prestan servicio a los pacientes.

Aplicación para gestionar las competiciones de un karting

El propósito de este proyecto es administrar las competiciones de karting y contribuir al enriquecimiento de un sector del automovilismo que se caracteriza por su profesionalismo y diversión.

El proyecto se centrará en el desarrollo de una aplicación que se encargará de la gestión de las carreras, los pilotos, las puntuaciones y la venta de entradas.

Esta aplicación se diseñará como una plataforma web, lo que garantizará su accesibilidad desde diversos dispositivos, ya sea un móvil o un ordenador, sin verse limitada por el sistema operativo utilizado.

A continuación, se presentará el sistema sobre el cual se construirá la aplicación:

- Un karting puede contar con uno o varios kartódromos, cada uno de los cuales debe registrar los siguientes atributos: nombre, dimensiones (ancho, que debe estar entre 8 y 15 metros, y longitud, que debe oscilar entre 600 y 1700 metros), tipo de kartódromo (interno o externo), número de curvas y número de escapatorias.
- El karting tiene varios boxes, de los cuales se quiere guardar, sus dimensiones (alto, largo y ancho).
- El karting dispone de múltiples boxes, de los cuales se quiere guardar, las dimensiones de cada uno de ellos, incluyendo alto, largo y ancho.
- En cada box, es posible almacenar varios karts y el equipamiento de los pilotos.
- De los karts, se quiere almacenar su número, que, junto con la categoría, servirá para identificar de manera única a cada kart. Además, se almacenarán los siguientes atributos: tipo de kart (normal, superkart o eléctrico), nombre y marca del motor, centímetros cúbicos del motor, tipo de neumáticos, capacidad del depósito de combustible en litros y modelo del volante.

- Cada kart está diseñado para competir en una categoría específica, pero también puede participar en otras categorías si lo permiten las reglas de la competición. Cabe destacar que en una categoría pueden competir varios karts.
- Las competiciones se subdividen en diversas categorías, como alevín, cadete, X30 junior, X30 senior y kz2, donde participan diferentes pilotos con sus karts. De cada categoría, se registrarán el nombre que la identifica y los rangos de edades permitidos para competir en esa categoría.
- En una competición, un piloto utiliza un kart, y un kart es empleado por un piloto en una competición. Además, un piloto con un kart puede tomar parte en múltiples competiciones. Los detalles que se desean almacenar de las competiciones incluyen el nombre, los patrocinadores, la fecha de inicio, la fecha de finalización y el piloto ganador.
- Los pilotos pueden practicar en cualquier momento en que el karting esté abierto. Se almacenará el precio, la fecha y el mejor tiempo obtenido durante la práctica. Cada piloto puede llevar a cabo múltiples prácticas en el karting, y cada práctica en el karting estará asociada a un piloto.
- Cada piloto dispondrá de un conjunto de equipamiento. Se registrarán los detalles del equipamiento, que incluirán el nombre, la zona del cuerpo a la que corresponde, el material utilizado y si el equipamiento es propiedad del karting o personal del piloto.
- Los pilotos deberán abonar al karting una tarifa mensual como costo de participación en las carreras del karting.
- Los pilotos deberán abonar al karting una tarifa por cada práctica realizada.
- De los pilotos se registrarán los siguientes datos: DNI, nombre, apellidos, fecha de nacimiento, localidad, licencia, edad y abreviatura del nombre.
- Todos los pilotos estarán clasificados, lo que permitirá visualizar su posición, diferenciando entre las distintas categorías en las que compiten.
- En una competición, un piloto obtiene una cantidad variable de puntos en función de su posición. Específicamente, se otorgan 20 puntos al ganador de la carrera, 18 puntos al segundo puesto, 16 puntos al tercero, 14 puntos al cuarto, 12 puntos al quinto, 10 puntos al sexto y 8 puntos al séptimo.
- Un kartódromo puede contar con múltiples gradas destinadas al público, y cada grada estará ubicada en un kartódromo específico. De estas gradas, se registrarán los siguientes detalles: código identificativo, asientos, disponibilidad de zonas VIP, presencia de techo, existencia de palco y la orientación de la grada (norte, sur, este u oeste).

- Los días de competición, el karting admite la entrada de espectadores que deberán abonar una tarifa al karting para presenciar la carrera. Se registrarán los datos del público, que incluirán el nombre y número de teléfono. Además, se guardará el precio de la entrada para el día de la competición.
- Un espectador del público adquirirá una entrada que le dará derecho a ocupar un asiento en una grada. Una persona del público puede en una grada comprar varios asientos. Cada asiento, que esté disponible en un día de competición, puede ser ocupado por una única persona.

Selección de una propuesta

De entre las tres propuestas iniciales, opté por seleccionar la única propuesta válida disponible en ese momento: una aplicación para gestionar los portátiles en el centro de estudios Ceinmark.

Creación del diagrama entidad relación

A partir del texto de la propuesta que seleccioné, procedí a identificar los distintos elementos del diagrama entidad relación.

Conjuntos de entidades y sus respectivos atributos

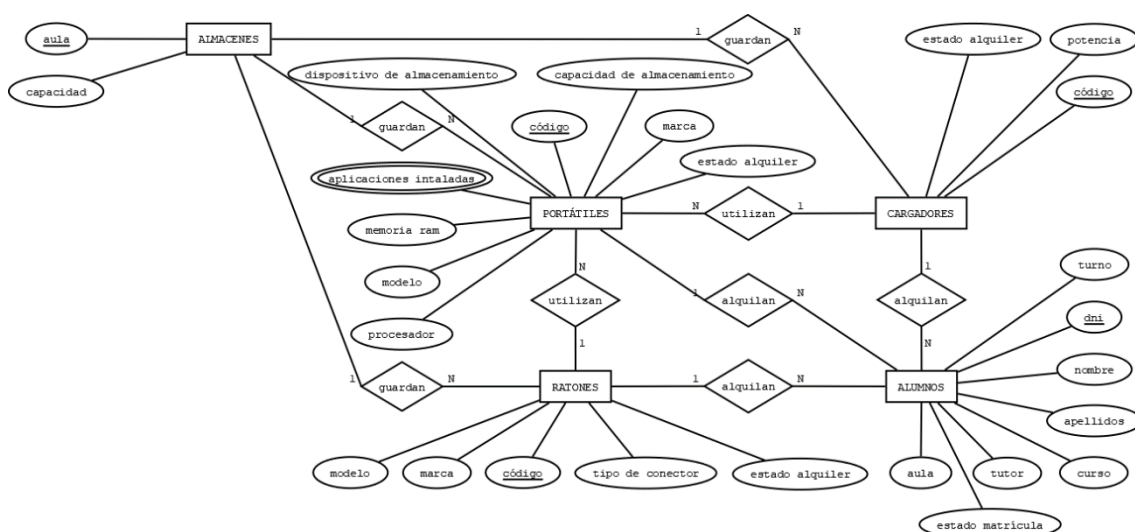
- Conjunto de entidades “Alumnos” con los siguientes atributos: DNI (atributo identificador del conjunto de entidades), nombre, apellidos, curso, tutor, aula, estado de la matrícula y turno de estudio.
- Conjunto de entidades “Portátiles” con los siguientes atributos: código (atributo identificador del conjunto de entidades), marca, modelo, procesador, cantidad de memoria RAM en GB, tipo de dispositivo de almacenamiento, capacidad del dispositivo de almacenamiento, aplicaciones instaladas (atributo multivaluado del conjunto de entidades) y estado del alquiler.
- Conjunto de entidades “Cargadores” con los atributos: código (atributo identificador del conjunto de entidades), potencia del cargador en vatios y estado del alquiler.
- Conjunto de entidades “Ratones” con los atributos: código (atributo identificador del conjunto de entidades), marca, modelo, tipo de conector del ratón y estado del alquiler.

- Conjunto de entidades “Almacenes” con los atributos: nombre del aula (atributo identificador del conjunto de entidades) y capacidad máxima de almacenamiento.

Relaciones y cardinalidad entre los elementos

- Un alumno tiene la posibilidad de alquilar un portátil, y un portátil puede ser alquilado por varios alumnos de diferentes turnos.
- El alumno puede alquilar un cargador y un cargador puede ser alquilado por varios alumnos de diferentes turnos.
- Un alumno puede alquilar un cargador, y un cargador puede ser alquilado por varios alumnos, de diferentes turnos.
- El alumno puede alquilar un ratón, y un ratón puede ser alquilado por varios alumnos, de diferentes turnos.
- Un portátil se almacena en un almacén, y un almacén tiene la capacidad de contener varios portátiles.
- El cargador se almacena en un almacén, y un almacén tiene la capacidad de contener varios cargadores.
- Un ratón se almacena en un almacén, y un almacén tiene la capacidad de contener varios ratones.

Una vez definidos todos los elementos del diagrama entidad relación, procedí a crearlo utilizando la aplicación Dia.



Para concluir esta fase, una vez que había finalizado el diagrama entidad relación, mi tutora lo revisó y aprobó.

Creación del diagrama relacional estático

Después de elaborar el diagrama entidad relación, procedí a representar cada conjunto de entidades mediante tablas, incluyendo también una tabla para el atributo multivaluado de aplicaciones. Acto seguido, asocié cada atributo a su tabla correspondiente.

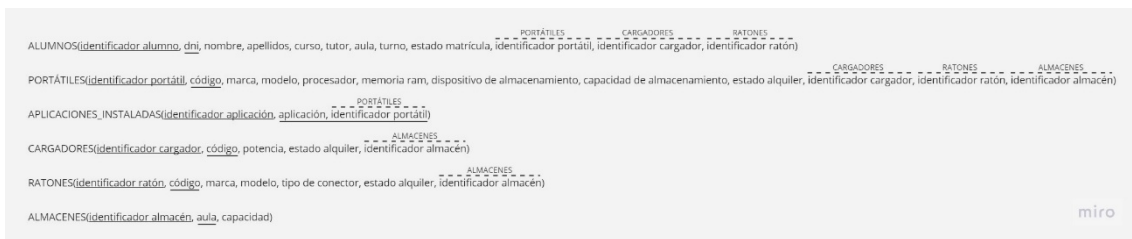
Posteriormente, definí la clave primaria y las claves foráneas de cada tabla, siguiendo la coherencia con la representación del diagrama entidad relación.

Finalmente, optimicé la estructura de cada tabla mediante la normalización, incorporando un atributo identificador (entero autoincremental) en cada una. Este nuevo identificador se convirtió en la clave primaria, sustituyendo la anterior por una clave única de la tabla.

Tablas

- Tabla “Alumnos”:
 - Identificador: identificador alumnos
 - Clave única: dni
 - Campos adicionales: nombre, apellidos, curso, tutor, aula y estado matrícula.
 - Claves foráneas: identificador portátil, identificador cargador e identificador ratón.
- Tabla “Portátiles”:
 - Identificador: identificador portátil
 - Clave única: código
 - Campos adicionales: marca, modelo, procesador, memoria ram, dispositivo de almacenamiento, capacidad de almacenamiento y estado alquiler.
 - Claves foráneas: identificador cargador, identificador ratón e identificador almacén.
- Tabla “Aplicaciones Instaladas”:
 - Identificador: identificador aplicación
 - Clave única de dos campos: aplicación e identificador portátil
 - Clave foránea: identificador portátil
- Tabla “Cargadores”:
 - Identificador: identificador cargador
 - Clave única: código
 - Campos adicionales: potencia y estado alquiler
 - Clave foránea: identificador almacén

- Tabla “Ratones”:
 - Identificador: identificador ratón
 - Clave única: código
 - Campos adicionales: marca, modelo, tipo de conector y estado alquiler.
 - Clave foránea: identificador almacén
- Tabla “Almacenes”:
 - Identificador: identificador almacén
 - Clave única: aula
 - Otro campo: capacidad



Tras realizar el diagrama relacional utilizando la aplicación Miro, lo presenté a mi tutora con la finalidad de identificar posibles fallos y, a continuación, cerrar esta fase del proyecto.

Implementación de la base de datos en MySQL

Utilizando dbForge, procedí a incorporar todos los elementos del diagrama relacional estático, siguiendo el ejemplo proporcionado. Posteriormente, llevé a cabo una revisión de los tipos de datos y las restricciones asociadas a cada tabla.

Fases

1. Creación de la base de datos

```
DROP DATABASE IF EXISTS alquiler_portatiles;  
CREATE DATABASE alquiler_portatiles;  
  
USE alquiler_portatiles;
```

2. Creación de tablas

- Tabla alumnos

```
CREATE OR REPLACE TABLE alumnos(  
    id_alumno integer AUTO_INCREMENT,  
    dni varchar(8),  
    nombre varchar(24),  
    apellidos varchar(48),  
    curso varchar(96),  
    aula varchar(4),  
    turno varchar(8),  
    estado_matricula varchar(16),  
    id_portatil integer,  
    id_cargador integer,  
    id_raton integer,  
    PRIMARY KEY (id_alumno)  
);
```

- Tabla portátiles

```
CREATE OR REPLACE TABLE portatiles(  
    id_portatil integer AUTO_INCREMENT,  
    codigo varchar(4),  
    marca varchar(24),  
    modelo varchar(24),  
    procesador varchar(24),  
    memoria_ram integer(4),  
    dispositivo_almacenamiento varchar(24),  
    capacidad_almacenamiento integer(8),  
    estado_alquiler varchar(24),  
    id_cargador integer,  
    id_raton integer,  
    id_almacen integer,  
    PRIMARY KEY (id_portatil)  
);
```

- Tabla aplicaciones instaladas

```
CREATE OR REPLACE TABLE aplicaciones_instaladas(  
    id_aplicacion integer AUTO_INCREMENT,  
    aplicacion varchar(32),  
    id_portatil integer,  
    PRIMARY KEY (id_aplicacion)  
);
```

- Tabla cargadores

```
CREATE OR REPLACE TABLE cargadores(  
  id_cargador integer AUTO_INCREMENT,  
  codigo varchar(4),  
  potencia integer(8),  
  estado_alquiler varchar(24),  
  id_almacen integer,  
  PRIMARY KEY (id_cargador)  
);
```

- Tabla ratones

```
CREATE OR REPLACE TABLE ratones(  
  id_raton integer AUTO_INCREMENT,  
  codigo varchar(4),  
  marca varchar(24),  
  modelo varchar(24),  
  tipo_conector varchar(8),  
  estado_alquiler varchar(24),  
  id_almacen integer,  
  PRIMARY KEY (id_raton)  
);
```

- Tabla almacenes

```
CREATE OR REPLACE TABLE almacenes(  
  id_almacen integer AUTO_INCREMENT,  
  aula varchar(4),  
  capacidad integer(4),  
  PRIMARY KEY (id_almacen)  
);
```

3. Creación de restricciones

- Restricciones de alumnos

```
ALTER TABLE alumnos  
  ADD CONSTRAINT uk_dni  
  UNIQUE KEY (dni),  
  
  ADD CONSTRAINT fk_alumnos_portatiles  
  FOREIGN KEY (id_portatil)  
  REFERENCES portatiles(id_portatil),  
  
  ADD CONSTRAINT fk_alumnos_cargadores  
  FOREIGN KEY (id_cargador)  
  REFERENCES cargadores(id_cargador),  
  
  ADD CONSTRAINT fk_alumnos_ratones  
  FOREIGN KEY (id_raton)  
  REFERENCES ratones(id_raton);
```

- Restricciones de portátiles

```
ALTER TABLE portatiles
  ADD CONSTRAINT uk_codigo
  UNIQUE KEY (codigo),

  ADD CONSTRAINT fk_portatiles_cargadores
  FOREIGN KEY (id_cargador)
  REFERENCES cargadores(id_cargador),

  ADD CONSTRAINT fk_portatiles_ratones
  FOREIGN KEY (id_raton)
  REFERENCES ratones(id_raton),

  ADD CONSTRAINT fk_portatiles_almacenes
  FOREIGN KEY (id_almacen)
  REFERENCES almacenes(id_almacen);
```

- Restricciones de aplicaciones instaladas

```
ALTER TABLE aplicaciones_instaladas
  ADD CONSTRAINT uk_aplicacion_portatil
  UNIQUE KEY (aplicacion, id_portatil),

  ADD CONSTRAINT fk_aplicaciones_portatiles
  FOREIGN KEY (id_portatil)
  REFERENCES portatiles(id_portatil);
```

- Restricciones de cargadores

```
ALTER TABLE cargadores
  ADD CONSTRAINT uk_codigo
  UNIQUE KEY (codigo),

  ADD CONSTRAINT fk_cargadores_almacenes
  FOREIGN KEY (id_almacen)
  REFERENCES almacenes(id_almacen);
```

- Restricciones de ratones

```
ALTER TABLE ratones
  ADD CONSTRAINT uk_codigo
  UNIQUE KEY (codigo),

  ADD CONSTRAINT fk_ratones_almacenes
  FOREIGN KEY (id_almacen)
  REFERENCES almacenes(id_almacen);
```

- Restricciones de almacenes

```
ALTER TABLE almacenes
  ADD CONSTRAINT uk_aula
  UNIQUE KEY (aula);
```

Tipos de datos de los campos

- El campo identificador de la tabla se define como integer con la propiedad auto_increment.
- Para los campos que representan valores numéricos, como memoria RAM, capacidad de almacenamiento y potencia se utiliza el integer.
- Los demás campos se clasifican como varchar, variando según el número máximo de caracteres.

Finalmente, verifiqué que el script se ejecutara correctamente y que la base de datos se creara con las tablas tal como las había concebido.

Generación del diagrama de la base de datos

Después de crear la base de datos, procedí a generar el diagrama de la base de datos utilizando dbForge.

Pasos

1. En la parte superior de dbForge, seleccioné la opción “Database” y posteriormente “Database Diagram”
2. Luego, arrastré la base de datos a la ventana recién abierta.
3. A continuación, realicé ajustes en la disposición de las tablas y reorganicé las fechas para mejorar la presentación general del diagrama.

Finalmente, revisé que todos los datos y relaciones estuvieran correctamente representados.

Generación de datos

Desde dbForge, he insertado 30 registros en cada tabla de la base de datos. Para esta tarea, en la configuración de la generación, definí el número específico de registros, en este caso, 30 registros por tabla. Además, especifiqué detalles como la base de datos correspondiente, la truncación de datos después de la generación, y el país asociado a los datos, optando por seleccionar Australia. Es relevante destacar que no incluí ni valores nulos ni vacíos en este proceso.

Datos seleccionados

- Configuré los identificadores autoincrementales como “ID (sequential)”, asignándoles valores del 1 al 30.
- Para los nombres y apellidos de los alumnos, utilicé las opciones “First Name (any gender)” y “Last Name”.

- Definí los datos enteros, como la potencia y la capacidad de almacenamiento, con el tipo “int”, especificando valores mínimos y máximos para su rango.

Datos generados por expresiones regulares (regex)

Configuré con regex los datos que utilizan caracteres aleatorios o patrones simples.

Aquí están las especificaciones:

- Para los códigos, utilicé el siguiente regex: “[0-9]{3}[A-Z]”.
- El DNI de los alumnos se definió con el regex: “[0-9]{7}[A-Z]”.
- Dado que había una amplia variedad de modelos de portátiles, diseñé este regex: “Vivobook [0-9A-Z]{4}|ExpertBook [0-9A-Z]{4}|EliteBookm840 G[0-9]|15S-[A-Z]{2}[0-9]{4}NS|Modern 15 B[0-9]{2}M-[0-9]{3}[A-Z]{3}XES|IdeaPad [0-9] [0-9A-Z]{6}|V[0-4][0-9] G[0-9] [A-Z]{3}|ThinkPad [A-Z][0-9]{3}|Extensa 15 EX215-[0-9]{2}|Aspire 3 A315-[0-9]{3}P-[0-9]{3}[A-Z]|TravelMate [A-Z][0-9] [A-Z][0-9]{3}|Go Start|Akoya [A-Z][0-9]{5}MD[0-9]{5}|Ioxbook [0-9A-Z]{3}|Latitude [1-9]{4}|Flex Advance|Neo|Vostro [1-9]{4}|Galaxy Book[1-9]|Voom Excellence|Netbook|NB[A-Z]-[1-9]{5}[A-Z]{5}|Satellite|U[1-9] [A-Z]{2}-[0-9]{2}ES[0-9]{3}[A-Z]{2}|LT[1-9]00E”.
- Para generar el procesador del portátil, definí el siguiente regex: “Intel I[3579] Gen 1[0-9]|Intel I[3579] Gen [4-9]|Intel Celeron|Intel Pentium|AMD Athlon|AMD Ryzen [3579] [34578][1-9]{3}”.
- Para definir el almacenamiento de la memoria RAM, utilicé este regex muy básico: “2|4|6|8|12|16|24|32|48|64|128”.
- Igualmente, para definir los modelos de los ratones hice algo parecido a lo anterior pero más breve “M[1-9]0|M[1-9]000|Easy Delta|Business|X[1-9]00|Basics|MC-[1-9]00|Flame|Mist|Tick|Crew”
- Asimismo, para los modelos de los ratones, seguí un enfoque similar pero más conciso con el regex: “M[1-9]0|M[1-9]000|Easy Delta|Business|X[1-9]00|Basics|MC-[1-9]00|Flame|Mist|Tick|Crew”.
- Por último, definí el tipo de conector del ratón con el siguiente regex: “USB|PS/2”.

Datos generados mediante archivos de texto

Para la creación de datos sin la presencia de caracteres aleatorios, elegí generarlos mediante archivos de texto que contienen las distintas opciones disponibles.

- Definí una variedad de cursos, incluyendo Servicios Administrativos (Primer y Segundo curso), Informática y Comunicaciones (Primer y Segundo curso), Gestión Administrativa (Primer y Segundo curso), Cuidados Auxiliares de Enfermería (Primer curso), Sistemas Microinformáticos y Redes (Primer y Segundo curso), Documentación y Administración Sanitarias (Primer y Segundo curso), Administración y Finanzas (Primer y Segundo curso), Desarrollo de Aplicaciones Multiplataforma (Primer y Segundo curso), Administración de Sistemas Informáticos en Red (Primer y Segundo curso), Educación Infantil (Primer y Segundo curso), Integración Social (Primer y Segundo curso), y Desarrollo de Aplicaciones Web (Primer y Segundo curso).
- Definí los turnos de los alumnos, siendo estos en la mañana o en la tarde.
- Establecí el estado de la matrícula de los alumnos, dividiéndolo en “Matriculado” y “No matriculado”.
- Las aplicaciones disponibles para los portátiles comprenden una amplia gama, desde herramientas de lectura como Acrobat Reader, utilidades de edición de audio como Audacity, programas de contabilidad como Contasol, hasta herramientas de desarrollo como dbForge, Eclipse y Visual Studio. También se incluyen navegadores como Google Chrome y Edge, herramientas de diseño gráfico como Figma, Gimp e Illustrator, editores de texto como Notepad++, y software de virtualización como VirtualBox. Además, se consideran reproductores multimedia como VLC, herramientas de compresión como WinRAR, y suites de oficina como Excel, PowerPoint y Word. En el ámbito de desarrollo específico, se encuentran entornos como Unity y Visual Studio Code, junto con Xampp, que combina Apache, MySQL, PHP y Perl para facilitar el desarrollo web.
- He establecido las distintas condiciones para el estado de alquiler, clasificándolo en “Alquilado”, “Disponible” y “Averiado”.
- En la selección de portátiles, se incluyen marcas diversas. Entre ellas se encuentran Acer, Alurin, Asus, Dell, Denver, Gigabyte, HP, Huawei, Innjoo, MSI, Lenovo, LG, Medion, Microsoft, Samsung, OEM, Polaroid, Primux, Prixton, Razer, Realme, Thomson, Toshiba y Vant.
- Las marcas de ratones ofrecen una diversidad de opciones, entre las cuales se encuentran Equip, Hama, HP, Logitech, NGS, Subblim y Trust.
- Se presentan distintas alternativas de dispositivos de almacenamiento, entre las que se incluyen HDD, SSD y NVMe.

La mayoría de los datos fueron recopilados de tiendas en línea especializadas en productos informáticos.

Después de configurar cada uno de los campos, pulsé sobre la fecha verde en la parte central superior y abrí el script al presionar el botón “Generate”. Posteriormente, añadí los datos a la base de datos ejecutando el script.

Generación de documentación automática

Desde la aplicación dbForge, generé la documentación siguiendo estos pasos:

Pasos

1. Desde la parte superior de dbForge, hice clic en “Tools” y luego en “New Documentation...”.
2. Elegí una conexión disponible
3. Marqué la casilla correspondiente a la base de datos “alquiler_portatiles” y sus respectivas tablas.
4. Hice clic en el botón superior “Generate...” para continuar.
5. Seleccioné el formato PDF y pulsé el botón “Generate” para crear la documentación.

Creación de la copia de seguridad de la base de datos

Con la aplicación dbForge, realicé la copia de seguridad de la base de datos.

Pasos

1. Desde el menú de exploración de las bases de datos, hice clic derecho en la base de datos y luego seleccioné “Backup and Restore” y “Backup Database...”.
2. En la configuración principal, definí la ruta y el nombre para la copia de seguridad.
3. En la sección de “options”, activé la casilla “Include CREATE DATABASE statement” para agregar la declaración de creación de la base de datos a la copia.
4. Finalmente, clic en el botón “Backup” para generar la copia de seguridad.

Propuestas de ampliación

Sistema de pago mensual para el alquiler de portátiles

Con el objetivo de mejorar la gestión de portátiles, propongo la incorporación de un sistema de pago para el alquiler de portátiles. Esta innovación tiene la intención de generar rentabilidad para el centro educativo.

La nueva funcionalidad permitirá a los estudiantes alquilar un portátil por un precio establecido, el cual se abonará de manera mensual, trimestral o anual.

A continuación, se definen las reglas de negocio de esta propuesta:

- En la relación entre los estudiantes y los portátiles, se registrará el precio mensual que cada estudiante debe abonar, incluyendo una fecha límite de pago y la especificación del tipo de abono, ya sea mensual, trimestral o anual.

Gestión de regletas

Con la finalidad de optimizar la gestión de recursos en nuestro entorno educativo, propongo la incorporación y control de regletas para cada aula del centro.

Cada aula estará equipada con varias regletas eléctricas, exclusivamente asignadas a un aula específica. Esta asignación permitirá simplificar la distribución y supervisión de la alimentación eléctrica para los dispositivos, contribuyendo así a mantener un entorno de aprendizaje ininterrumpido.

A continuación, se establecen las reglas de negocio para la propuesta:

- En el sistema, se definirá un nuevo conjunto de entidades denominado "Aulas". Para cada aula, se busca registrar su nombre (código), los cursos que se imparten en ella y los tutores de dichos cursos, quienes serán responsables de garantizar el correcto funcionamiento del sistema de regletas.
- Se establecerá una relación entre las aulas y los almacenes, ya que un almacén pueda encontrarse en un solo aula, y que, a su vez, una misma aula pueda albergar varios almacenes.
- Además, se establecerá una relación entre las aulas y los alumnos, ya que un alumno puede transitar por diferentes aulas a lo largo del turno, y, a su vez, en una misma aula pueden coincidir varios alumnos.

- Además, se contará con el conjunto de entidades llamado “Regletas”, donde se registrará la cantidad de enchufes de cada regleta, si tienen interruptor o no, y su código, el cual estará compuesto por el código del aula al que pertenecen y el número de la regleta (por ejemplo, aula 6A, regleta 2 → 6A2).
- El conjunto de entidades “Regletas” se relaciona con el conjunto de entidades “Aulas”, estableciendo que varias regletas pueden pertenecer a una sola aula, mientras que cada regleta está vinculada exclusivamente a una única aula.

Sistema de gestión de conexiones WiFi

Con el propósito de fortalecer la infraestructura tecnológica, propongo la implementación de un sistema de gestión de conexiones WiFi.

En el entorno educativo, la conectividad inalámbrica óptima es esencial, ya que una sola conexión WiFi puede abarcar varias aulas. Además, cada aula puede contar con varias conexiones WiFi estables, mejorando significativamente la velocidad de la red.

Esta mejora permitirá que cada aula disponga de múltiples conexiones, fomentando un entorno más conectado y adaptable a las necesidades tecnológicas actuales.

A continuación, se establecen las reglas de negocio para la propuesta:

- En el sistema, se implementará un nuevo conjunto de entidades denominado “Conexiones WiFi”, donde se registrará el nombre, velocidad de transferencia, frecuencia, tipo de seguridad y número máximo de conexiones simultáneas.
- Las conexiones WiFi se encuentran instaladas en un aula, con la posibilidad de contar con varias conexiones en una misma aula.
- Las conexiones WiFi proporcionan cobertura a varias aulas, y a su vez, un aula puede recibir señal de múltiples conexiones WiFi.
- Un portátil puede estar conectado simultáneamente a una única conexión WiFi, mientras que una conexión WiFi puede estar vinculada a varios portátiles.