



UNIVERSIDAD DE GUADALAJARA
CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E
INGENIERÍAS



DEPARTAMENTO DE CIENCIAS COMPUTACIONALES
SEMINARIO DE SOLUCIÓN DE PROBLEMAS DE INGENIERÍA DE
SOFTWARE.

SECCIÓN: D01

PROFESORA: KARLA ÁVILA CÁRDENAS

CALENDARIO 2021B

Práctica #7: Diseño estructural

Nombre: Ian Gerardo Horcasitas Pérez

Carrera: Ingeniería en Computación

Código: 216583618

ÍNDICE

INTRODUCCIÓN	3
DESARROLLO	5
Diagrama de bloques	5
Modelo Entidad-Relación	5
Diccionarios	6
Modelo relacional	6
Tarjetas CRC	7
Diagramas de clases.....	7
CONCLUSIONES	8
REFERENCIAS.....	8

INTRODUCCIÓN

Tras planear el funcionamiento e indicar las diferentes formas en que los usuarios pueden relacionarse con el programa mediante el diagrama de casos de uso, se debe comenzar el diseño estructural del programa. Ya se formaron las bases necesarias para comenzar a idear la manera en que estará construido el software, es por ello por lo que se emplean diferentes diagramas.

Cada uno de los diagramas pertenecientes a esta etapa representará diferentes aspectos importantes de la estructura del sistema final. En esta práctica se hará uso de 6 diferentes diagramas, los cuales pueden identificarse entre 3 diferentes objetivos.

En primer lugar, se encuentra el diagrama de bloques, cuyo objetivo es crear un mapa de los procesos que realizará el programa. El diagrama de bloques es el único perteneciente a esta clasificación. Como indica su nombre, se encuentra conformado solamente por bloques, dichos bloques indican la existencia de un proceso en específico, y estos se acomodan de forma ordenada, conectando aquellos que se relacionan de forma consecutiva.

Después se encuentran los diagramas que indican la estructura de la base de datos, estos son el diagrama Entidad-Relación, los diccionarios y el modelo relacional. El diagrama entidad relación muestra las relaciones que existen entre las diferentes tablas de la base de datos, cada tabla es una entidad (a menos que se trate de una relación de muchos a muchos). Los diccionarios son tablas que describen las características de cada campo de las tablas, indican su nombre, tipo de dato y tamaño, su descripción, si debería ser NULL o no y el tipo de campo que es (solo aplica si es llave primaria o llave foránea). Por último tenemos al modelo relacional, de acuerdo a Rus, editor de Economipedia “Un modelo relacional consiste en representar datos por medio de tablas relacionadas cuyas filas se llaman tuplas y las columnas variables, conformando así una base de datos” (2020). Este toma como base el modelo Entidad-Relación para mostrar los campos de cada tabla, y las llaves foráneas generadas por la relación entre las tablas.

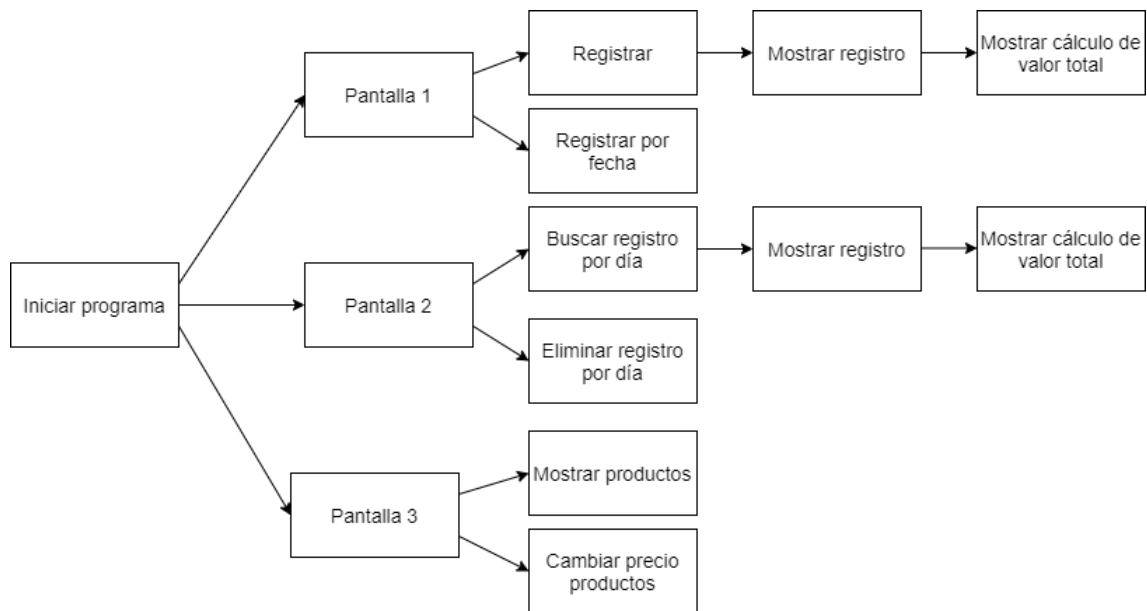
En el tercer caso se encuentran las tarjetas CRC y los diagramas de clases. Estos se encargan de describir la estructura de un programa orientado a objetos. Las tarjetas CRC indican los procesos que cada clase lleva a cabo, así como también con qué clases se relaciona para poder realizar sus tareas. Por otra parte, el diagrama de clase se encarga de describir totalmente el contenido de las clases, mostrando una tabla por cada clase, la cual contiene todos los atributos de dicha clase, indica si el atributo se encuentra privado, público, etc. También muestra los métodos e indica si dichos métodos están hechos para recibir o para regresar algún tipo de dato. Todas las tablas se encuentran relacionadas entre sí por medio de flechas, las cuales indican cuál clase interactúa con cuál.

Nota. Se omite la sección “Resultados” en el reporte pues no fue requerida para la actividad según su definición en el archivo “lineamientos de documentación.pdf”.

DESARROLLO

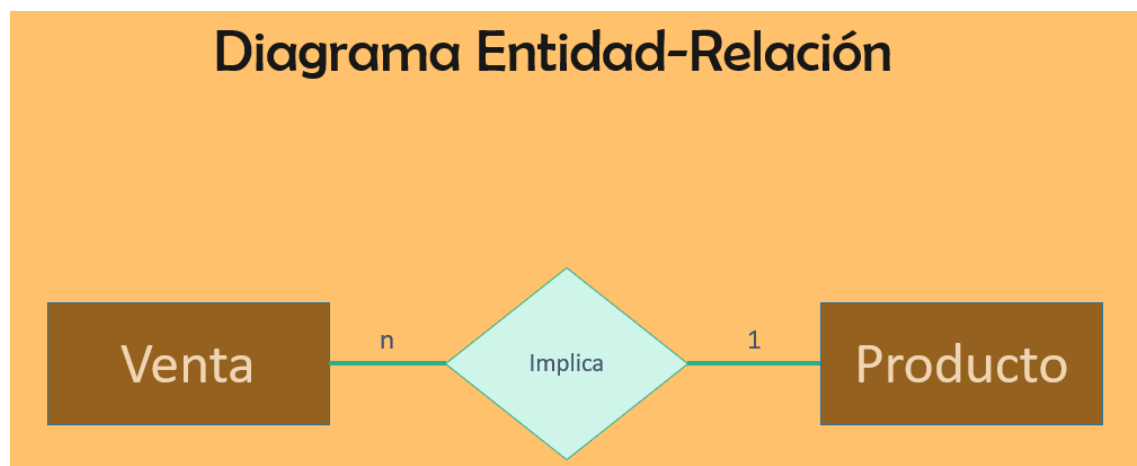
Diagrama de bloques

En este diagrama se representan los diferentes procesos del programa.



Modelo Entidad-Relación

Se puede apreciar la relación entre las dos tablas.



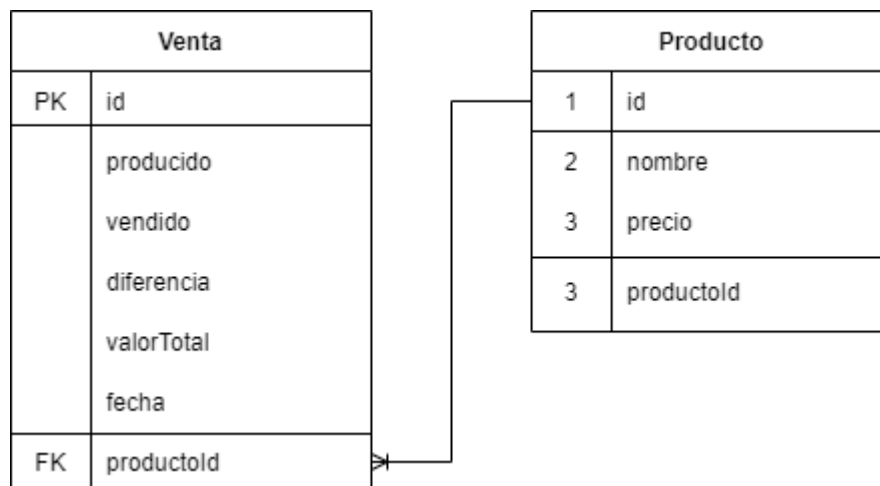
Diccionarios

Se anexan los diccionarios de datos de las tablas Venta y Producto.

Diccionario de Datos				
Venta				
Tipo	Columna	Tipo de dato	Descripción	Null
PK	id	INT	Número de identificación de la venta	NOT
	producido	SMALLINT	Cantidad de elementos producidos	NOT
	vendido	SMALLINT	Cantidad de elementos vendidos	NOT
	diferencia	SMALLINT	Diferencia entre producidos y vendidos	NOT
	valorTotal	DOUBLE(10,2)	Elementos vendidos x el precio	NOT
	fecha	DATE	Fecha cuando se realizó la venta	NOT
FK	productoid	INT	Número de identificación del producto vendido	NOT
Relaciones: productoid con Producto				
Producto				
Tipo	Columna	Tipo de dato	Descripción	Null
PK	id	INT	Número de identificación del producto	NOT
	nombre	VARCHAR(30)	Nombre del producto	NOT
	precio	DOUBLE(10,2)	Precio del producto	NOT

Modelo relacional

Se puede apreciar la relación entre ambas tablas, siendo que Ventas recibe el id del producto.



Tarjetas CRC

Debido a que no se planea que el programa utilice más de una clase, se anexa solo una tarjeta, la correspondiente a la clase que controla la interfaz.

Tarjeta CRC	
mainwindow	
Responsabilidades	Colaboradores
Permite ingresar la cantidad de elementos producidos y sobrantes de un día	
Muestra la información de los registros de un día	
Muestra el valor resultante de la sumatoria de todo lo vendido en el día	
Permite buscar los registros de un día	
Permite eliminar los registros de un día	
Muestra la fecha	
Muestra los productos del menú y su precio	
Permite cambiar el valor de los productos del menú	

Diagramas de clases

En el diagrama se puede apreciar los atributos y métodos de la clase mainwindow, es la única tabla en el diagrama debido a que se trata de la única clase en el programa.

mainwindow
+ui : Ui_MainWindow +scene : QGraphicsScene +venta : lista +producto : lista
limpiar():void insertarRegistro(lista):void insertarRegistroFecha(lista,int,int,int):void buscarRegistro(int, int, int):lista eliminarRegistro(int,int,int):void cambiarPrecioProducto(lista):void mostrarProductos():lista

CONCLUSIONES

Me costó más realizar esta actividad de lo que pensaba, los diagramas referentes a la base de datos fueron muy sencillos de llevar a cabo, sin embargo, tardé mucho en averiguar qué poner en las tarjetas CRC y diagrama de clases, pues no utilizaré mas que una clase para realizar el proyecto. Dicha clase es la encargada de crear y manejar la interfaz.

Estuve un largo tiempo debatiéndome si debería agregar una clase para las ventas y otra para los productos. Al final decidí no agregar las clases, pues no lo veía necesario para la manera en que tengo pensado realizar el programa.

Me agradó volver a realizar diagramas relacionados con bases de datos, sin embargo, fue extraño realizar los diagramas solo con dos tablas, esto debido a los requisitos. la mitad del programa utilizará clase (la de interfaz) y la otra mitad usará programación estructurada (la que hace la conexión a la base de datos).

Me pareció muy interesante aprender sobre los diagramas de clases y las tarjetas CRC, a pesar de que no las utilicé mucho, veo un gran potencial en estas herramientas para la programación orientada a objetos.

REFERENCIAS

Rus, E. (2020). *Modelo relacional*. Economipedia.

<https://economipedia.com/definiciones/modelo-relacional.html>