

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Estructuras de Datos
Vacaciones Primer Semestre de 2023
Proyecto



Ingeniero:

- Ing. Luis Espino

Auxiliares:

- Cristian Suy

EDD Creative

Proyecto Único - FASE II

Ciudad de Guatemala, 16 de Junio de 2023

Objetivos

Objetivo general:

- Aplicar los conocimientos del curso Estructuras de Datos en el desarrollo de las diferentes estructuras de datos y los diferentes algoritmos de manipulación de la información en ellas.

Objetivos específicos:

- Utilizar el lenguaje Go para implementar estructuras de datos no lineales
- Utilizar la herramienta Graphviz para graficar las estructuras de datos.
- Definir e implementar algoritmos de ordenamiento, búsqueda e inserción en las diferentes estructuras a implementar.

Resumen de estructuras a utilizar:

FASE II

- Matriz (fase anterior): Aplicación de Filtros
- Árbol AVL: Manejo de Pedidos Realizados
- Grafo No Dirigido: Sistema de Filtros Por Imagen
- Tabla Hash: Listado de Pagos
- Blockchain: Sistema de pagos

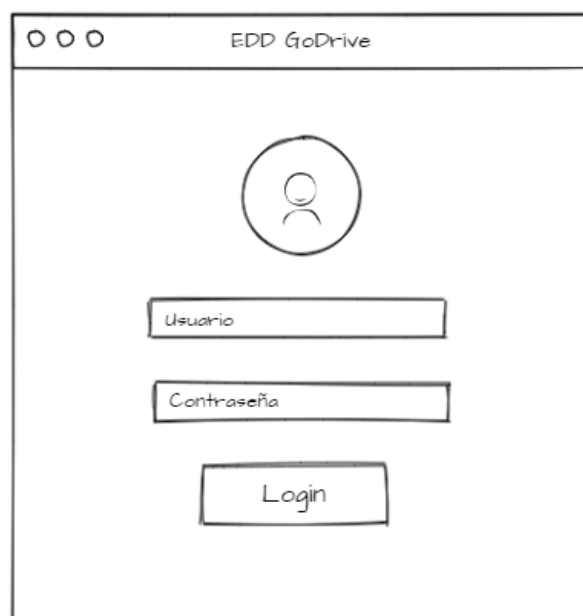
Definición del problema

La empresa EDD Creative, luego del éxito de la aplicación en consola presentado por usted como desarrollador, la empresa desea que la aplicación sea utilizada en un entorno web, y desean que usted cree una aplicación web y un servidor para poder almacenar y tener un ambiente más agradable para los usuarios como para los empleados, para ello usted debe presentar un prototipo local de la aplicación para considerar utilizarlo en servicios web. Se solicita que el sistema ya se incluya el sistema de pagos para los clientes, y el sistema de filtros para las imágenes

Descripción de la aplicación

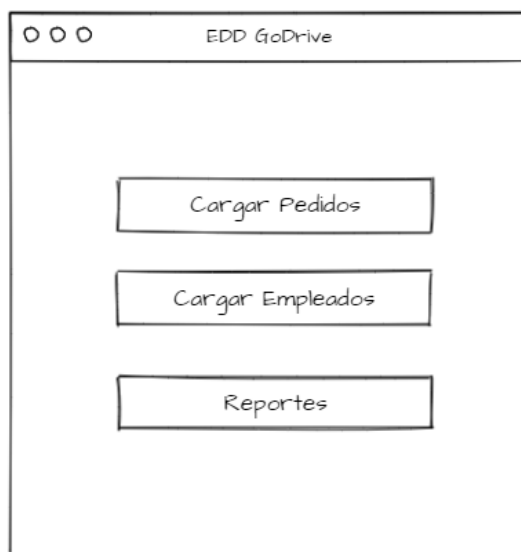
FASE II

Login: El sistema contará con un login principal para realizar el inicio de sesión de cada empleado de ventas que esté registrado en el sistema, este deberá validar que el usuario y contraseña sean correctos, habrá un usuario especial, el cual será el administrador del sistema este se llamara ADMIN_#Carnet y contraseña Admin, éste será el encargado de realizar la carga masiva con el archivo generado de la primera fase



A hand-drawn sketch of a login window titled "EDD GoDrive". The window contains a circular placeholder for a user profile picture at the top center. Below the picture are two rectangular input fields: the first is labeled "Usuario" and the second is labeled "Contraseña". At the bottom center of the form is a rectangular button labeled "Login".

Vista de Administrador: El administrador contará con el siguiente menú donde podrá elegir algunas de las opciones que se explicarán más adelante.



A hand-drawn sketch of an administrator menu window titled "EDD GoDrive". The window contains three rectangular buttons stacked vertically in the center. The top button is labeled "Cargar Pedidos", the middle button is labeled "Cargar Empleados", and the bottom button is labeled "Reportes".

Administrador: Tendrá como función principal cargar empleados para el cual usará la misma estructura de la fase anterior y asignar pedidos

- Cargar Pedidos: Este será el archivo JSON generado en la fase 1 y el factor para insertar al árbol será el id_cliente

La estructura a utilizar Arbol AVL

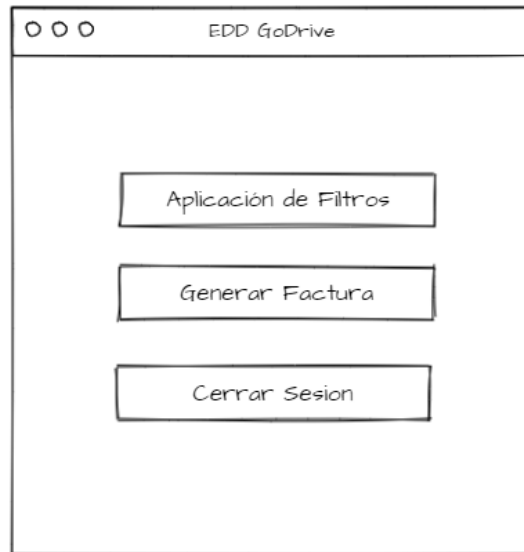
```
{
  "pedidos": [
    {
      "id_cliente": 12,
      "imagen": "bmo"
    },
    {
      "id_cliente": 12,
      "imagen": "mario"
    },
    {
      "id_cliente": 12,
      "imagen": "deadpool"
    },
    {
      "id_cliente": 12,
      "imagen": "ave"
    }
  ]
}
```

- Empleados: Id, nombre, cargo, contraseña. Se utilizará el archivo CSV de la

La estructura se usará la misma de la fase anterior

id	nombre	cargo	password
6385	Cristian Suy	Ventas	6385_Ventas
5372	Hector Jimenez	Ventas	5372_Ventas
4399	Pablo Velasquez	Diseño	4399_Diseño
1229	Jaquelin Gomez	Diseño	1229_Diseño
3607	Yadira Ruiz	Diseño	3607_Diseño
3518	Paula Fuentes	Ventas	3518_Ventas

Sistema Principal: Cuando el empleado a cargo del pedido realice el inicio de sesión contará con un menú especial, el cual tendrá las siguientes opciones:



- **Filtros de Imagen:** El usuario podrá ver ejemplos de los diferentes filtros que el sistema ofrece con una imagen de prueba, dicha imagen será proporcionado por el auxiliar para fines ilustrativos de la aplicación, luego que decida qué filtros aplicar se debe generar a partir de la imagen original los diferentes filtros solicitados (se debe conservar la imagen original tambien), tener en cuenta que el cliente puede elegir aplicar más de 1 filtro, ejemplo el cliente puede pedir que sea filtró en escala de grises y en modo espejo X, estos filtros se explicará más adelante.
- **Generar Factura de pago:** Luego de que el cliente haya decidido si desea su imagen con filtros u original se debe extender una factura para el manejo de pagos se utilizará el sistema de Blockchain para la seguridad e integridad del mismo, contará con una interfaz que solicitara nada mas los datos del pago a realizar, los demás datos ya debes ser insertados de manera automática se muestra a continuación un ejemplo de interfaz.

EDD Creative

Fecha
01-07-2024::16:50:03

Empleado Cobrador
6359

Usuario
5223

Pago
Q36.00

Realizar Pago

Regresar al Menú anterior

y para blockchain como estructura se utilizará el siguiente sistema de almacenamiento, su similitud es al de una lista doble enlazada de nodos, y este contendrá la siguiente información.

Index: Este número representa el número del bloque, para el bloque génesis tendrá el valor de index 0 y cada bloque que se cree será tendrá los valores 1, 2, 3, ... etc.

Timestamp: Representa la fecha y hora de creación exacta, este contará con la siguiente estructura (DD-MM-YY-::HH:MM:SS).

Biller: Este representará al empleado quien facturó, en este caso el carnet del mismo.

Customer: Este representará el ID del cliente que está pagando

Payment: Éste será la cantidad de dinero pagado por la imagen

PreviousHash: Este es el bloque previo y sirve para validar que la cadena del blockchain no esté corrupta. para el caso del bloque génesis, hash anterior deberá ser 0000

Hash: Este protege la información del mensaje y que este no este corrupto, para el caso de este se usará la encriptación SHA256 y creará una función para recibir com parametros parametros utilizara index, timestamp, transmitter, receiver, message.

Ejemplo:

SHA-256(index+timestamp+Biller+Customer+Payment)

Adicionalmente el empleado contará con una opción extra el cual mostrará las facturas extendidas solo por el, los datos importantes son el Hash generado de la factura en el inciso anterior y el id del cliente, adicional es importante mostrar constantemente el ID del empleado que inicio sesion o que facturo, las facturas de manera general serán almacenadas en una tabla hash de orden 5, **cuando la tabla llega a un 60% de utilización la tabla aumentará su capacidad hasta el siguiente número primo.** El valor de referencia para insertar en la tabla hash será el id del usuario el cual se deberá codificar para obtener un valor entero el cual servirá como parámetro para una función hash por división y así encontrar el índice en el cual se insertará el nodo, **Para las colisiones dentro de la tabla se utilizará una resolución direccionamiento abierto de doble dirección hash** Consiste en que una vez detectada la colisión se debe generar otra dirección aplicando la función hash a la dirección previamente obtenida. El proceso se detiene cuando se encuentra en una posición vacía para insertar el nodo.

Para el Algoritmo de este proceso, se deben crear 2 funciones hash con el cual en la primera función donde se encontrara el primer índice, se realizará en base al número de carnet del estudiante utilizando los últimos 2 dígitos, ejemplo: 202312345 el número a usar es 45, y se debe multiplicar el id de usuario por ese número, sumarle carnet y conseguir el modulo del mismo.

$\text{índice} = (45 * 1526 + 202312345) \% \text{tamaño_tablahash}$

Si el índice está ocupado, se realizará la siguiente función hash debe realizar el mismo procedimiento con la condición extra que se debe multiplicar 2 veces el número de carnet, de la siguiente forma:

$\text{índice} = (45 * 1526 + 202312345 * 2) \% \text{tamaño_tablahash}$

si el nuevo indice no esta vacio buscara el siguiente puesto vacio

Facturas Generadas por Empleado 8932	
ID Cliente	ID Factura
5963	80520+5ED6+A6+7CEF220207D1AC0ED359C90CAA39009DEEE0E9++BB3AF35D+
8996	++B+AE+BB50B60+C+552B0+0F0B0FC0+C+FD75060+77EC+D1F33E09D2AE05F
7952	2C0A95C2527E3ECBC5AA5A653790EB072A0C13056D376F73F5B+CEE9+5DEC02

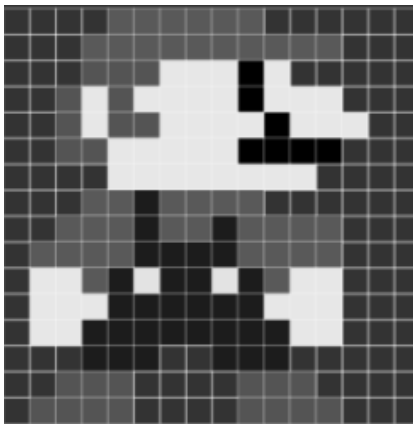
Regresar

Aplicación de Filtro: El cliente tendrá opción de aplicarle filtros a su imagen pixel art, para ello se debe generar el html y el css con los cambios solicitados para previa visualización, los filtros aceptados por la imagen serán:

- **Negativo:** Este filtro altera la imagen únicamente en sus colores, se debe de calcular el negativo específico de cada píxel y posteriormente pintar dichos píxeles con su negativo equivalente.

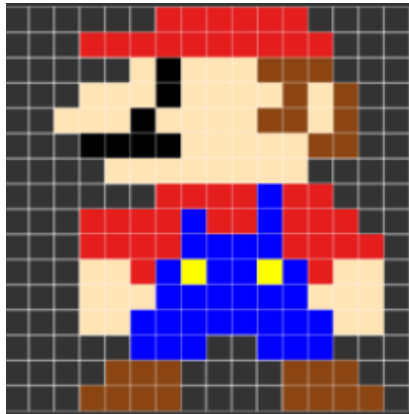


- **Escala de Grises:** Este filtro de igual manera únicamente afectará los colores de la imagen, aplicando un filtro de escala de grises a cada píxel de la imagen, de igual manera posteriormente se debe de pintar cada píxel con su gris equivalente.



- **Modo Espejos:** El filtro de Espejo no modificará los colores de la imagen, sin embargo si la posición de los píxeles de la imagen, por lo tanto se tendrán subsecciones de este filtro, el cual modifica la imagen en distintos ejes.
 - **Espejo X:** Este filtro modifica la orientación sobre el eje x de la imagen, por lo tanto se generará un espejo sobre el eje x. Este filtro no genera

modificación en el color de los píxeles de la imagen.

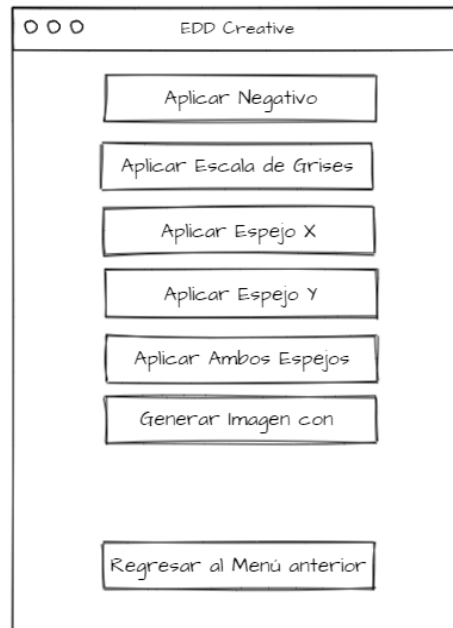


- Espejo Y: Este filtro modifica la orientación sobre el eje y de la imagen, por lo tanto se genera un espejo sobre el eje y. Este filtro no genera modificación en el color de los píxeles de la imagen.



- Doble espejo: Este filtro modifica la orientación sobre ambos ejes de la imagen, por lo tanto se genera un doble espejo. Este filtro es equivalente a aplicar el filtro espejo de eje x seguido del filtro de espejo de eje y. Este filtro no genera modificación en el color de los píxeles de la imagen.





Sistema de Ventas Completadas: Se podrá visualizar al empleado y las imágenes que ellos han vendido a los clientes, utilizando una matriz de adyacencia para el almacenamiento del mismo, y un grafo dirigido para visualizar esa relación entre empleado, clientes e imágenes vendidas, junto a los filtros aplicados. **Para ello mostrará la imagen de reporte de grafo y una tabla con la información de la matriz de adyacencia.**

EDD Creative

Solicitudes de filtros por clientes

ID Cliente	Imagen	Filtros
5963	mario	Espejo X
8996	carlin	Espejo Y
7952	tux	Negativo Espejo X

Reporte

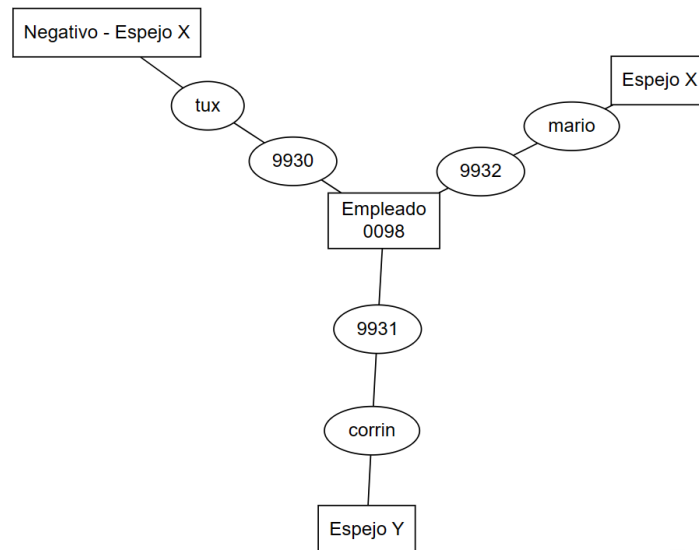
Regresar

Área de Reportes

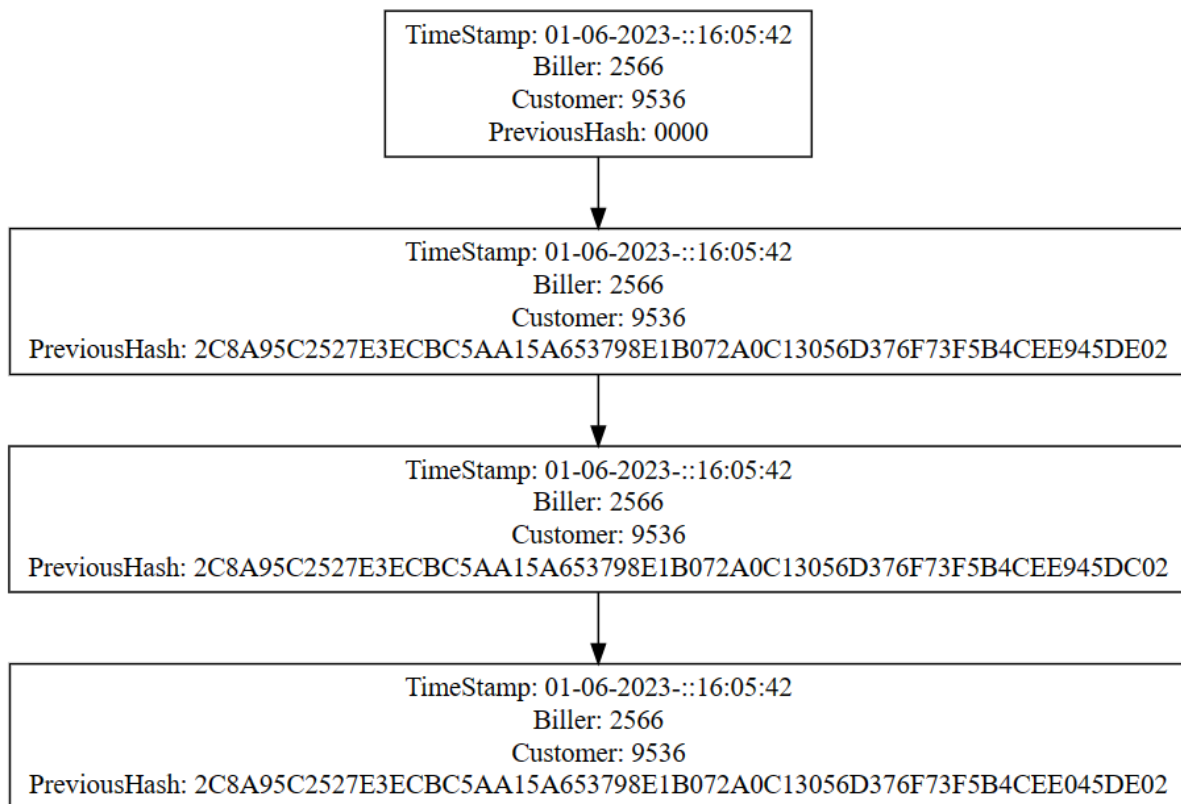
Se contará con un apartado especial en el que se puedan mostrar en forma de grafos las estructuras en tiempo real, generando la imagen utilizando el visualizador

de imágenes del sistema operativo. Los reportes sólo serán generados con la herramienta de Graphviz.

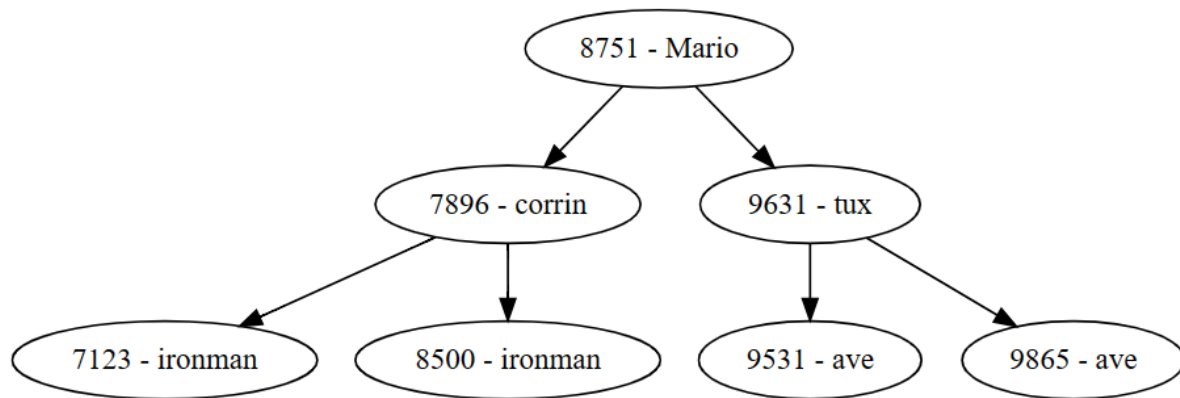
Grafo: Este reporte será asociado al apartado de Ventas Completadas



Reportes de Pagos: Y el reporte gráfico, mostrará todo los bloques pero con la siguiente estructura e información



Reporte de Árbol: con este reporte se mostrará los pedidos realizados de la fase anterior.



Tecnologías a utilizar

FASE II:

Realizar la aplicación Servidor en Golang:

Este se realizará por medio de un servidor, todas las estructuras serán realizadas en lenguaje GO y se comunicaran mediante peticiones API-REST con el frontend. Se permite el uso de cualquier package para el manejo de las rutas en Go, la más recomendada para facilidad <https://github.com/gorilla/mux>, en este repositorio se muestra el proceso de instalación. También se permite usar librerías nativas para la parte de codificación, decodificación y encriptación.

Realizar la aplicación Frontend:

Este se realizará en cualquier Framework (React, Angular, VueJS, etc) ya que este solo servirá para la interfaz gráfica de la aplicación, y realizar las peticiones al servidor de Go, se permite también el uso de plantillas o cualquier otro medio para darle estilos a su página.

Realizar los reportes en Graphviz:

Todos los reportes deberán estar realizados en graphviz y deben estar constantemente generando cuando se realice un cambio, para poder observar de forma visual el estado actual de las estructuras.

Restricciones

- Las estructuras deben de ser desarrolladas por los estudiantes **sin el uso de ninguna librería o estructura predefinida en el lenguaje**.
- Se permite el uso de cualquier paquete o repositorio de GO para la parte de blockchain en lo que respecta a codificación, decodificación y encriptación de datos.
- Se permite el uso de paquete para el manejo de la API-REST
- Los reportes son esenciales para verificar si se trabajaron correctamente las estructuras solicitadas, por lo que si no se tiene el reporte de alguna estructura se anularán los puntos que tengan relación tanto al reporte como a la estructura en cuestión.
- Se permite el uso de framework o librerías como Angular o React entre otros para el desarrollo del proyecto, solamente para el desarrollo de interfaz gráfica en esta Fase, toda las estructuras deben ser creadas en Go.

Observaciones

- El lenguaje para la fase 2 será Go y todas las salidas y entradas mediante una aplicación web
- Herramienta de desarrollo de reportes solamente **Graphviz** y **HTML** para visualizar imágenes.
- La entrega se realizará por medio de Github, el nombre del repositorio (Fase anterior) debe ser **EDD_VJ1S2023_PY_#carnet y para comodidad y recomendacion, separar por carpetas backend y frontend**. Y por medio de **UEDI** se hará entrega del link de su repositorio.
- Recordar tener sus repositorios en privado, para evitar copias o plagios de código.
- Agregar al repositorio como colaborador al Auxiliar, usuario **CristianMejia2198**
- Realizar los **manual de técnico y de usuario** en el README del repositorio o archivos separados
- Toda duda que se tenga durante el proceso, será realizada por medio de los foros de **UEDI**.
- Fecha de entrega:
 - FASE II: Sabado 1 de Julio del 2023 a las 23:59 hrs (Media Noche).