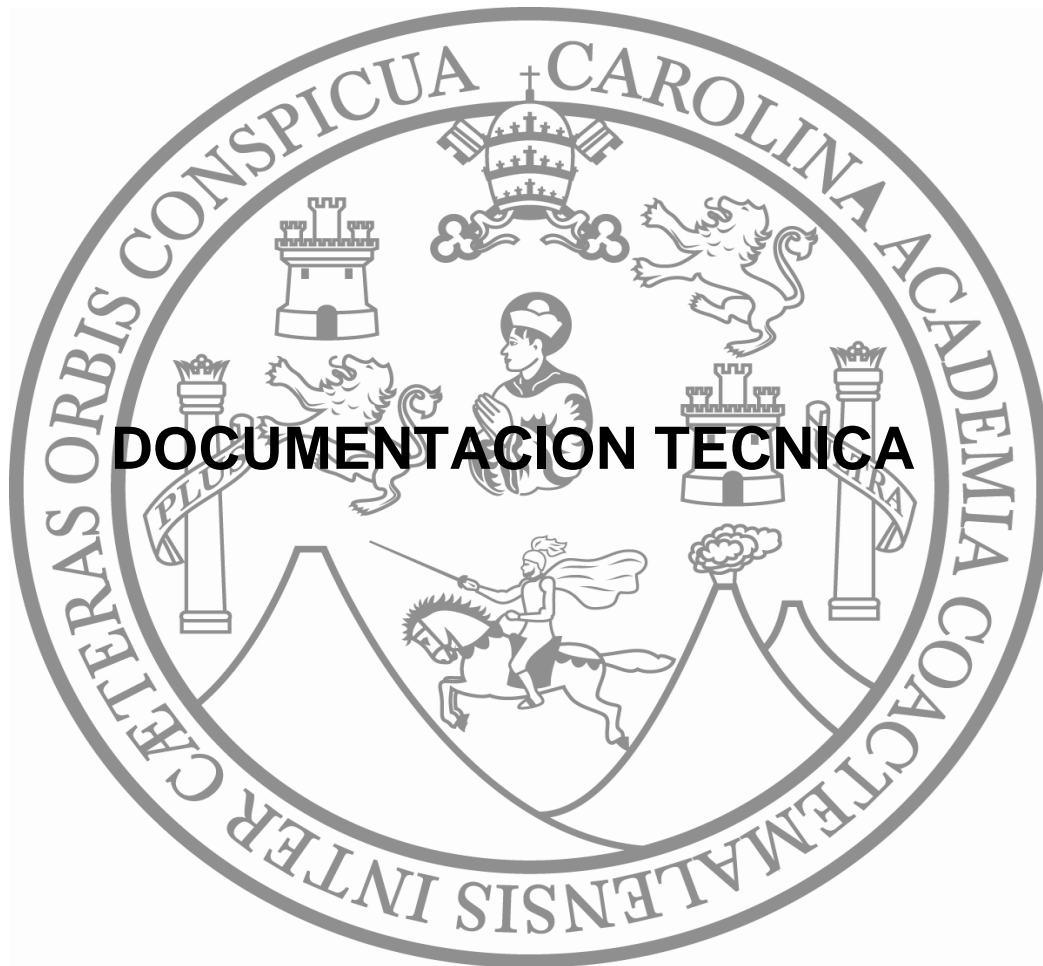


Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Estructuras de datos



Eduardo Abraham Barillas del Aguila

201903342

## ESTRUCTURAS IMPLEMENTADAS

**Lista Doblemente Enlazada:** La lista circular doblemente enlazada fue utilizada para poder manejar la lista de usuarios, esta esta basada en un template de clase T por lo que puede ser reutilizada para cualquier otro modelo/clase. Esta lista cuenta con varios métodos como lo puede ser:

**InsertFirst:** el cual recibe como parámetro una clase T definida en el template y se encarga de insertar al inicio de la lista.

**InsertLast:** el cual recibe como parámetro una clase T definida en el template y se encarga de insertar al final de la lista.

**Update:** el cual recibe 2 parametros, la claseT del objeto y la posición que se desea modificar;

**Get:** el cual retorna el objeto T;

**Remove:** el cual recibe la posición del objeto y se encarga de eliminarlo de la lista;

Y una propiedad accesible **Length** la cual nos provee de la longitud de la lista;

**Lista Simple:** Esta lista fue implementada con la finalidad de tener una lista genérica que nos permita realizar multiples funcionalidades como lo son los reportes y de la misma maanera fue utilizada para poder manejar una lista de listas para los articulso de tienda;

**InsertFirst:** el cual recibe como parámetro una clase T definida en el template y se encarga de insertar al inicio de la lista.

**Update:** el cual recibe 2 parametros, la claseT del objeto y la posición que se desea modificar;

**Get:** el cual retorna el objeto T;

Y una propiedad accesible **Length** la cual nos provee de la longitud de la lista;

**Stack o Pila;** Esta estructura fue utilizada para poder llevar el control de los movimientos o disparos realizados durante la partida, cuenta con los métodos:

**Peek:** Para obtener el primer ítem sin removerlos de la pila;

**Push:** Sirve para insertar a la pila un Objeto T que es pasado como parámetro;

**Pop:** al igual que el Peek nos retorna el valor que se encuentra hasta arriba de la pila pero este si lo remueve de la pila;

**Queue o Cola** Esta estructura fue utilizada para poder llevar el control de los movimientos o disparos realizados durante el tutorial para desplegarlos en orden;

**Peek:** Para obtener el primer ítem sin removerlos de la pila;

**Enqueue:** Sirve para insertar a la cola un Objeto T que es pasado como parámetro y lo inserta hasta el final de la cola;

**Deque:** al igual que el Peek nos retorna el valor que se encuentra hasta el inicio de la cola pero este si lo remueve de la cola;

### **Clases Utilizadas**

**User** Contiene los atributos username, password(Encriptada con SHA256), coins, skins y métodos setter y getter para poder manejar los atributos anteriormente descritos ya que se encuentran privados

**Node** Es un struct encargado de manejar la informacion del puntero siguiente y anterior asi como la información de la clase T;

**Articulo** Es un objeto el cual se encarga de poder manejar la información de los ítems que se encuentran en tienda;

**Shot:** Es un objeto el cual almacena en propiedades de tipo entero valores de coordenadas X e Y.

### **Librerias Utilizadas**

**Jsoncpp**

**SHA256**