

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
FACULTAD DE INGENIERIA  
ESCUELA DE CIENCIAS Y SISTEMAS  
MANEJO E IMPLEMENTACIÓN DE ARCHIVOS

# **MANUAL TECNICO**

JUAN ANTONIO SOLARES SAMAYOA  
201800496  
11 DE MAYO DE 2021

## DETALLES TECNICOS DE LA APLICACIÓN

- Lenguajes de programación utilizados: JavaScript, Golang

## BACKEND DE LA APLICACIÓN

- Golang

## FRONTEND DE LA APLICACIÓN

- Framework utilizado: React js

## BASE DE DATOS IMPLEMENTADA EN LA APLICACIÓN

- Oracle SQL database 18c Express Edition

## DEFINICIONES TECNICAS IMPORTANTES:

- **SERVIDOR HTTP:** Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor. El código recibido por el cliente es renderizado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo. Generalmente se usa el protocolo HTTP para estas comunicaciones, perteneciente a la capa de aplicación del modelo OSI. El término también se emplea para referirse al ordenador.
- **TRIGGER:** Un trigger o disparador es un objeto que se asocia con tablas y se almacena en la base de datos. Su nombre se deriva por el comportamiento que presentan en su funcionamiento, ya que se ejecutan cuando sucede algún evento sobre las tablas a las que se encuentra asociado.
- **PROCEDIMIENTO ALMACENADO:** Un procedimiento almacenado (almacenado físicamente en una base de datos. Su implementación varía de un gestor de bases de datos a otro. La ventaja de un procedimiento almacenado es que, al ser ejecutado, en respuesta a una petición de usuario, es ejecutado directamente en el motor de bases de datos, el cual usualmente corre en un servidor separado.

## DESCRIPCIÓN DE LOS PROCEDIMIENTOS ALMACENADOS MÁS IMPORTANTES DE LA APLICACIÓN

NOMBRE DE PROCEDIMIENTO ALMACENADO	DESCRIPCIÓN
CREAR_TEMPORADA	Ingresa una nueva temporada a la base de datos
CREAR_JORNADA	Ingresa una nueva jornada a la base de datos
CARGAR_TEMPORAL	Llena la tabla temporal de la base de datos los datos obtenidos a partir un archivo YAML
INICIAR_SESION	Verifica el usuario y la contraseña enviadas por medio de un formulario
PAGAR_MEMBRESIA	Paga la membresía de un cliente, para acceder a los beneficios que cada tipo de membresía.
INGRESAR_PREDICCION	El cliente almacena en la base de datos una predicción al resultado de un evento deportivo
INGRESAR_RESULTADO	EL administrador del sistema ingresa los resultados correspondientes a un evento deportivo
MOFICIAR_USUARIO	Modifica los datos de un cliente
ELIMINAR_DEPORTE	Elimina un deporte de la base de datos

## EJEMPLO DE EJECUCION DE PROCEDIMIENTOS ALMACENADOS EN GOLANG

```

44 // ----- PAGAR MEMBRESIA -----
45 func pagarMembresia(w http.ResponseWriter, r *http.Request) {
46
47     var transaccion PagoTransaccion
48
49     reqBody, err := ioutil.ReadAll(r.Body)
50
51     db, err := sql.Open("oci8", "TEST/1234@localhost:1521/ORCL18")
52     if err != nil {
53         log.Fatal(err)
54     }
55     defer db.Close()
56
57     json.Unmarshal(reqBody, &transaccion)
58
59     fmt.Println(transaccion.Id_temporada)
60     fmt.Println(transaccion.Id_cliente)
61     fmt.Println(transaccion.Tipo_membresia)
62     res, err := db.Exec("begin PAGAR_MEMBRESIA(:1, :2, :3);end;", transaccion.Id_cliente, transaccion.Tipo_membresia, transaccion.Id_temporada)
63
64     if err != nil {
65         fmt.Println(err)
66         return
67     }
68
69     fmt.Println(res)
70 }
71

```