**ITCR**

**Escuela de Ing. en Computación**

**Base de Datos 1**

**Prof. Franco Quirós**

**Diciembre 2020**

**Proyecto final**

[INSERTAR IMAGEN LOGO TEC]

* **Cantidad de horas trabajadas:**

15 horas.

* **Relato de los avances:**

Elaboración del modelo:

modelo competencias ciclísticas es un giro (varias carreras) ciclísticas que se llevan a cabo en un país, este modelo de datos se utilizara para guardar los puntos y tiempos de diferentes competencias ciclísticas donde hay premios para el mejor ciclista de montaña, al ciclista mas regular, al mejor ciclista, al mejor equipo.

En este modelo tuvimos que utilizar herencia por primera ves ya antes la habíamos visto en diferentes diagramas, pero en este proyecto ya la tuvimos que usar en un modelo físico y se nos complicó porque no sabíamos cómo hacer la relación, pero investigamos en algunas paginas y recordamos clases antiguas y nos pareció que la mejor opción para hacer la relación era de 1 a 1 o sea clase madre 1 a 1 con clase hija, esperemos que sea la mejor solución al problema.

Le preguntamos al profesor si había que hacer una pagina web para las consultas porque en el proyecto pasado nos habíamos confundido y el profesor nos pidió una vista SQL y nosotros entregamos una vista en c# está ves le preguntamos si podía ser una pagina web y el nos dijo que sí.

[INSERTAR IMAGEN DONDE EL PROFE DICE SI]

Una vez construimos el modelo según lo visto en clase empezamos a revisarlo para ver si no habíamos cometido un error, también cambiamos un poco el modelo con lo visto porque algunos datos de los xml venían distintos a lo planeado entonces agregamos número camisa, código carrera, código instancia giro para hacer los mapeos de mejor forma y los errores que teníamos era: orden erróneo de los atributos en las tablas, relaciones erróneas y tablas redundantes.

Cuando terminamos de revisar los errores del modelo empezamos a leer más indicaciones del proyecto final una es que cuando hagamos una actualización sobre tablas debemos hacerlo mediante invocación de store procedure y como la simulación necesita muchos inserts entonces comenzamos a crear varios store procedure para insertar datos, uno por cada tabla.

[INSERTAR IMAGEN DE LA INDICACION]

Luego de eso le preguntamos al profesor si el modelo estaba de la mejor forma, también le preguntamos la funcionalidad del número de camiseta ya que era muy similar al id del corredor el profesor nos dijo que había una tabla innecesaria en el modelo que la quitáramos (giroxequipo) y también nos dijo que el id corredor no es igual al número de camiseta porque un ciclista podía cambiar de equipo por ende tendría otro número de camiseta nosotros llegamos a la conclusión que id corredor no era la mismo que numero camiseta pero que el numero de camiseta si nos iba a servir para mapear en los equipos ciclísticos a determinado ciclista.

Se usaron los SP inserts de antes y logramos hacer el SP general de insertar datos catalogo desde el xml hacia la BD y también se comenzó la simulación

la cual funcionara de la siguiente manera primero lo que se hace es iterar por cada año en los xml y agregarlo en una tabla que tenga las instancias de giro,

luego itero por cada instancia de giro y obtengo los datos para luego ser insertados mediante los stored procedures.

Uso un sp para procesar las instancias de giro

Recibe como parámetro una variable xml con una instancia de giro.

[INSERTAR IMAGEN DEL CODIGO SIMULACION]

Una vez llevábamos un poco avanzado el proyecto nos interesó saber cuánto había que sacarnos en el proyecto para pasar el curso aunque no es algo relevante en el trabajo si es algo importante a saber por si surge algún error

por parte del profesor a la hora de entregar la nota, sin embargo, la fórmula en Excel que paso el profesor es un poco extraña y nos generó inquietudes por ende aun no estamos muy seguros.

[INSERTAR Imagen evaluación curso]  
Comenzamos a hablar un poco sobre los movimientos que era un "Debito por sanción", "Crédito por Tiempo Duración Carrera", entre otros…Entendimos que cuando ocurría una sanción se producía un debito por ende se le sumaban minutos al corredor y que cuando el corredor ganaba se hacía un crédito de minutos a su favor.

Luego se terminó la simulación pero aún falta actualizar los campos de “suma de tiempo” y “suma de puntos”, también realizar los movimientos en el momento correcto cual es el problema que para realizar estas funciones que bueno, que para la suma de pts o tiempo se requiere sacar como el listado de los corredores que ganaron cierta etapa y cada vez que un corredor gana se hace el movimiento, calcular cual corredor ha ganado y cual corredor hay que aplicarle sanción requiere un poco más de trabajo.

Mientras se realizaba la simulación otro compañero de grupo empezó a realizar las consultas que se pedían, pero aún no se ha hecho el portal ni se han probado a ver si funcionan de la forma más optima estas consultas son:

[INSERTAR IMAGEN DE LAS CONSULTAS]

Estas consultas lo que hacen es primero obtener el id giro por medio del nombre del giro luego obteniendo el id giro se busca el id de instancia giro con el id instancia giro se obtiene el Código instancia giro el cual con el seleccionamos en la tabla IGXEQXCORREDOR el nombre del corredor y la suma del tiempo que hizo en ese giro cabe destacar que la suma del tiempo ya se realiza en la simulación este mismo método nos sirve para conseguir los puntos del ranking regular y los puntos del ranking montaña cuando los conseguimos hacemos un select top 10 order by puntos o tiempo y así hacemos un select de los mejores ciclistas de determinado giro, en el caso del equipo funciona muy parecido pero hay que dirigirse a la tabla INSTANCIAGIRO porque en ella están los tiempos totales de los equipos, hay que recordar que para obtener datos de las tablas catálogos se utiliza inner join.

[INSERTAR IMAGEN STOREPROCEDURE]

En este proyecto algo esencial es el manejo de puntos, tenemos tres competencias principales: el ganador del giro, el corredor más regular y el ganador de montaña, pero ¿Cómo funcionan estas competencias? Bueno el giro lo gana el corredor que allá corrido todas las carreras de ese giro y además allá hecho menor tiempo por ende cada vez que llega hay que ir acumulando su tiempo. El ranking de corredores regulares funciona en que cada vez que el corredor gana una competencia va acumulando puntos en ese ranking si queda de segundo acumula menos por ende al finalizar el premio se le da al corredor que allá obtenido mayor a 0 en todas las competencias y tenga la mayor cantidad de puntos y por último el de montaña se reparte puntos a un único de ciclista el que llegue primero a la cima y dependiendo de la dificultad se le asigna 3 puntos, 2 puntos o 1 punto. Cada vez que los corredores lleguen hay que hacer los movimientos de ganancia de punto esto aún está en proceso, pero fue clave explicar bien todos estos pasos porque es el plato fuerte y lo más importante del proyecto.

Se realizo la suma de puntos y tiempo en los corredores la estrategia que utilizamos fue que cuando se le aplica a un corredor un movimiento ya sea regular, montaña o de tiempo se hace un update en la tabla IGXEQXCORREDOR con la suma si fuese un crédito si es un debito le hacemos la rebaja de puntos o tiempo.

Se arreglaron algunos errores de las consultas para conseguir el giro exacto.

* **Análisis de resultados:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Excelente | Bueno | Normal | Malo | Muy malo |
| Documentación | x |  |  |  |  |
| Base de Datos | x |  |  |  |  |
| Catálogos | x |  |  |  |  |
| Simulación |  |  | x |  |  |
| Consulta1 |  |  | x |  |  |
| Consulta2 |  |  | x |  |  |
| Consulta3 |  |  | x |  |  |
| Consulta4 |  |  | x |  |  |

En la documentación nos colocamos 10 porque le dedicamos bastante tiempo comparado a los otros proyectos y agregamos todas las partes requeridas.

El modelo lo hicimos según lo visto en clase y agregamos datos adicionales según lo pedía el xml.

Hicimos el insertado de catálogos en las tablas.

La simulación nos presenta algunos pequeños problemas a la hora que escribo está parte de la bitácora esperemos solucionarlos es a la hora de ingresar tiempo en la IGXEQXCORREDOR sumar los tiempos nos genera problema.

Sin la simulación de la mejor forma no podemos verificar las consultas por eso me pongo normal porque en si las consultas están bien a nivel de lógica, pero aún no pueden ser probadas.

* **Forma en que trabajo el equipo:**
* **Experiencias:**
* **Referencias:**

Sobre time: <https://www.lawebdelprogramador.com/foros/SQL-Server/702918-datetimeobtener-solo-la-hora-y-no-lafecha.html#i702918>

Sobre estándar SQL: <https://poorsql.com/>

Sobre Date: <https://docs.microsoft.com/en-us/sql/t-sql/functions/datediff-transact-sql?view=sql-server-ver15>

Sobre Concat: <https://docs.microsoft.com/en-us/sql/t-sql/functions/concat-transact-sql?view=sql-server-ver15>

Sobre Top(10): <https://social.msdn.microsoft.com/Forums/es-ES/f35ebf84-293f-48e7-ad44-e5bbcc93da0a/seleccionar-los-10-mejores-de-cada-grupo-sql-server-2008-r2-sp2?forum=sqlserveres>