TRABAJO PRÁCTICO OBLIGATORIO FINAL

**Materia :** Programación II

**Año :** **2016**

**Carrera : Analista de Sistemas**

**Profesor :** Claudio Alvarez

**Enc.Trab.Pcos. :** Raquel Bros

**Objetivo** **:** Integración de métodos

----------------------------------------------------------------------------------------------------------

**Normativas:**

* **El trabajo es personal y debe ser desarrollado por todos los alumnos que han cursado la materia en el 2016 o los alumnos que vayan a rendir libre durante las mesas del 2017.**
* El trabajo es obligatorio y representa los **contenidos prácticos** del examen final. Por lo tanto, no puede encontrarse desaprobado. Cuando se logre su aprobación, el alumno estará en condiciones de rendir los **contenidos teóricos** en las fechas que se establezcan para las mesas de exámenes finales regulares y los **contenidos teóricos** **/ prácticos** en las fechas que se establezcan para las mesas de exámenes finales libres.
* Su entrega, deberá ser coordinada con el profesor  **con una anticipación no menor a 10 días hábiles** a la fecha de la mesa examinadora. No se aceptarán presentaciones espontáneas ni entregas parciales. El trabajo NO dispone de vencimiento. La presencia del alumno es obligatoria.

**Formalidades exigidas:**

En una carpeta, se deberá entregar el siguiente material:

* Listado impreso del código fuente del presente trabajo práctico en hojas sueltas y numeradas, debiendo figurar el nombre del alumno y el año de cursado en c/u de ellas. La numeración debe ser considerada como un todo.
* Un **CD-RW** (regrabable) con:
  + el código fuente, debidamente identificado con los datos del alumno y la leyenda “**Programación II -** **TP Final año ……..**”
  + una versión ejecutable (.Exe) del programa. El alumno deberá verificar previamente su funcionalidad en el Centro de Cómputos del Instituto
* Cada librería deberá estar soportada en una Unidad
* Hoja de portada, con los datos personales del alumno y el año de cursado. La misma presentación como etiqueta del CD.
* Un índice de las unidades
* Un esquema gráfico (entidad – relación) minuciosamente explicado sobre como están conformados los diferentes ME y las relaciones que existen entre ellos.
* Documentación a nivel de Unidades y a nivel de código fuente.
* Copia de este enunciado

**Lenguaje a utilizar:** Delphi (cualquier versión).

**Fundamentos teóricos complementarios**

Sabemos que un árbol **trinario** dispone de nodos compuestos por:

* *la información propiamente dicha*
* *un enlace al hijo izquierdo (a un elemento de menor peso relativo)*
* *un enlace al hijo derecho (a un elemento de mayor peso relativo)*
* *un enlace al padre*
* *un enlace al hijo medio (a un elemento* ***de igual*** *peso relativo)*

Las estructuras trinarias tienen la gran ventaja de hacer fácil el manejo de claves repetidas, pues el almacenamiento de elementos duplicados se encuentren todos enlazados al medio (formando, por ejemplo, “una estructura arbórea binaria” clasificada por una clave secundaria) lo cual optimiza las búsquedas en los archivos de transacciones y/o movimientos (evidentemente, en un archivo de “datos maestros” -donde no hay repeticiones de claves- el tercer enlace no presenta ninguna ventaja).

**MercaTrucho**

Se requiere implementar un sistema de venta de artículos usados que denominaremos MercaTrucho.

La información será mantenida a través de archivos binarios reunidos bajo diversos Métodos Estructurales (ME). Estos son:

* ME de Usuarios (Hash abierto)
* ME de Categorías (a elección del alumno)
* ME de Publicaciones de Artículos (Arboles binarios índices AVL)
* ME de Compras Realizadas (Hash cerrado con enlaces dobles)
* ME de Mensajes compartidos entre usuarios (Árbol trinario con control de niveles)

Se recomienda que los ME residan en una carpeta compartida de red, cuyos archivos serán accesibles por todos los programas **MercaTrucho** que se necesiten correr desde las diferentes terminales.

Se aclara que se trata de terminales de red local. De esta forma, entre 2 o más ejecuciones simultáneas del programa se emularán los procesos de publicación, consultas, pagos, etc.

Queda en claro entonces que NO se trata de una aplicación web sino de red local. Se insiste que, a los efectos de poder hacer la defensa, el alumno deberá utilizar una carpeta compartida en entorno de red, donde colocar los ME.

**DEFINICION DE MÉTODOS**

1. **ME de Usuarios**

Tiene por objeto registrar a todos los usuarios de **MercaTrucho**. Como se ha dicho, es un ME compartido (por lo cual debe residir en una carpeta de red -local, por ejemplo- y así común a todas las conexiones).

Las personas que van a usar el servicio se registran con su email y algunos datos personales. Un usuario sólo podrá hacer operaciones (comprar y vender) en el sistema si está registrado aquí. Quienes accedan al sistema sin identificarse solo podrán consultar las publicaciones de ventas. En este ME también se podrá visualizar el Estado de cada usuario, su última fecha de conexión y si se encuentra o no en línea (conectado). Los usuarios actualizan esta información cada vez que se conectan a MercaTrucho.

Para ello tendremos un ME basado en un archivo Hash abierto (con colisiones) de MAX= 80 registros. La clave de búsqueda es el (email). Esta clave es la que debe usarse como entrada a la función Hash que dará la posición física del elemento.

Una vez registrado, el usuario podrá realizar publicaciones de ventas o ser comprador del sistema. También existirá la figura del administrador, esto es, una cuenta de usuario que tiene acceso a todos los datos y reportes que brinda el sistema.

1a) **el archivo USUARIOS.DAT** (datos, Hash con colisiones abierto) con los siguientes campos:

* Email : string. Debe validarse que la dirección disponga de un formato válido. Para el administrador se usará la dirección Admistrador@MercaTrucho.com
* Clave: string. Esto es, la clave secreta del usuario. Para el administrador, la clave por defecto será "palo\_y\_a\_la\_bolsa". Se entiende que todos podrán cambiarla cuando lo requieran. La clave deberá estar encriptada. Queda a criterio del alumno escribir el algoritmo de encriptación que considere más adecuado. El mismo deberá estar explicado en la documentación a entregar junto con este trabajo.
* Apellido y Nombre: string
* Domicilio: string
* Provincia: entero rango 1..25, que represente a todas las provincias de la Argentina. El alumno deberá utilizar una estructura de datos fija (por ejemplo un arreglo) donde se establecerá la relación entre el código y la provincia. Por ejemplo: 00:Capital Federal; 01:Buenos Aires;02:Córdoba; etc.
* FechaHora: date-time (indica cuando fue dado de alta , tomado del clock de la computadora)
* Foto: del usuario, por ejemplo, en formato .JPG
* **Id Usuario:** Se trata del número interno que identifica a este usuario. Es único. Lo genera automáticamente el sistema, tomándolo del campo <Ultimo ID interno> del archivo de Control. Este ID se utiliza para referenciar a cada usuario en los restantes ME.
* **Estado:** Indica si el usuario está conectado o no conectado (se actualiza cuando el usuario se conecta / desconecta a MercaTrucho)
* **Fecha-Hora ultima conexión:** time-date (se actualiza cuando el usuario se conecta a MercaTrucho)

2a) **el archivo USUARIOS.CON** (control) con los siguientes campos:

* Cantidad de claves: 0..MAX.
* Ultimo Id interno: longint. Autoincremental. Inicialmente en 0.

Queda a criterio del alumno establecer cuales algoritmos de exploración de área utilizar para este Hash. En la documentación a entregar, deberá justificar los motivos de la elección.

1. **ME de CATEGORIAS**

Tiene por objeto registrar las categorías de artículos que forman parte de la publicación. Por ejemplo: herramientas eléctricas, instrumentos musicales, etc. Las altas-bajas-modificaciones de este ME solo son accesibles para el usuario que tiene clave de administrador. No se podrán hacer modificaciones ni bajas, si la Categoría dispone de Artículos (vendidos o a vender).

Se deja a consideración del alumno la implementación del ME que considere más adecuado para este modelo de datos, según su eficiencia. En la documentación a entregar, deberá justificar los motivos de la elección.

1b) **el archivo CATEGORÍAS.DAT** (datos)

* Clave: un entero positivo.
* Nombre del Rubro (o categoría): string
* Comisión: es el porcentaje de la comisión que cobra MercaTrucho al vendedor, una vez efectuada la venta de un artículo
* (resto de campos a definir a criterio del alumno)

2b) **El archivo CATEGORÍAS.CON**

* (campos a definir según el modelo de datos que se opte)

1. **ME de ARTICULOS PUBLICADOS**

En este método se almacenan las publicaciones de venta que realizan los vendedores del sistema. Se trata de un ME que cuenta con 2 árboles binarios de búsqueda sobre un archivo de datos. Uno organiza los artículos por vendedores y el otro por categorías. Las estructuras de datos son los siguientes:

1 c) **el archivo PUBLICADOS.CON**

* Ultimo Id interno: longint. Autoincremental. Inicialmente en 0.
* Raíz Índice 1
* Raíz Índice 2
* Borrados Indice 1
* Borrados Indice 2
* Borrados Ventas

2 c) **el archivo PUBLICADOS.DAT**

* **Id Publicado:** Se trata del número interno que identifica a este artículo. Es único. Lo genera automáticamente el sistema, tomándolo del campo <Ultimo ID interno> del archivo de Control. Este ID se utiliza para referenciar a cada artículo publicado.
* Clave de la Categoría: tomado de CATEGORIAS.DAT
* Id\_Vendedor: en este caso, el Id\_Usuario del vendedor del producto (tomado de USUARIOS.DAT)
* Nombre del artículo: string
* Descripción de la publicación (detalle): string
* Precio de venta: real
* Fecha de la publicación
* Fecha de cierre de la publicación
* Tipo de artículos: 1=usado; 2=nuevo
* Estado: 1=publicado; 2=pausado 3=vendido ; 4=anulado; 5=bloqueado por el administrador del sistema

3 c) **el archivo PUBLICADOS.NTX\_1**

Organiza a **PUBLICADOS**.DAT a través del id\_vendedor

* Id\_vendedor
* Hijo Izquierdo, Hijo Derecho, Padre: enlaces
* Posición: posición del elemento en **PUBLICADOS**.DAT

4 c) **el archivo PUBLICADOS.NTX\_2**

Organiza a **PUBLICADOS**.DAT a través del id\_Categoría

* Id\_Categoría
* Hijo Izquierdo, Hijo Derecho, Padre: enlaces
* Posición: posición del elemento en **PUBLICADOS**.DAT

Los archivos índices son de tipo AVL. Por tal motivo todas las inserciones y eliminaciones deberán hacerse por rotación.

Los recorridos que se efectúen sobre **PUBLICADOS**.DAT deberán realizarse a través de algunos de los índices.

1. **ME de Ventas Realizadas**

Aquí se van trasladando las operaciones de venta, a medida que se van cerrando las diferentes operaciones (solo las ventas con Estado=3). Es decir, los artículos cuya publicación se vence o que son suspendidos por el usuario, NO se colocarán aquí.

Se trata de un ME basado en un hashing cerrado con enlaces dobles. El espacio de direccionamiento es de 50 posiciones.

1 d) **el archivo VENTAS.CON**

* Borrados en Ventas.Dat

2 d) **el archivo VENTAS.HASH**

* Primero, Ultimo: en Ventas.Dat, sobre los enlaces dobles ordenados
* Total: cantidad de elementos que colisionan en esta posición

3 d) **el archivo VENTAS.DAT**

* Id\_Comprador: en este caso, el Id\_Usuario del comprador del producto (tomado de USUARIOS.DAT)
* Id\_Publicado: el que identifica al producto publicado, esto es el Id de PUBLICADOS.DAT
* Nombre del artículo: string
* Precio de venta: real
* Fecha de la publicación
* Fecha de la venta
* Tipo de artículos: 1=usado; 2=nuevo
* Calificación del comprador: 0=Sin calificar aún; 1=Recomendable; 2=Neutral; 3=No recomendable
* Porcentaje comisión (obtenida de CATEGORIAS.DAT)
* Comisión cobrada: boolean. Indica si efectivamente MercaTrucho ya recibió la comisión por la venta realizada.
* Proximo, Anterior: Enlaces de los elementos que colisionan en la posición

1. **ME de Conversaciones**

Tiene por objeto ser el espacio donde los compradores hacen consultas al vendedor del artículo.

Se trata de ME con un **índice arbóreo trinario** que maneja **COLAS parciales** (los datos o mensajes intercambiados) en el Hijo Medio. Esto significa que cada conversación estará dada por un conjunto de mensajes almacenados en colas parciales en MENSAJES.DAT

El presente ME también lleva un control de balanceo del árbol (solo los nodos Hijo Izquierdo, Hijo Derecho; es decir, no se incluye al nodo medio).

1c) **el archivo MENSAJES.DAT** (datos de las colas parciales) conteniendo los mensajes de intercambio

* Número: indica el número del mensaje dentro de la conversación. Es autoincremental.
* Mensaje comprador : String o memo. Aquí se coloca la pregunta que hará el cliente.
* Mensaje vendedor: String o memo. Aquí se coloca la respuesta que dará el vendedor
* Fecha – Hora : date – time
* Siguiente: el enlaces entre los mensajes de la cola.

2c) **el archivo MENSAJES.NTX** (índice arbóreo trinario)

* Id interno de la publicacion (en PUBLICACIÓN.DAT)
* Id Cliente: en este caso, el Id\_Usuario del cliente interesado en el producto (tomado de USUARIOS.DAT)
* Ultimo número: indica el número del último mensaje generado en esta conversación. Inicialmente en cero.
* Primero, Ultimo: a la lista de mensajes en MENSAJES.DAT
* Hijo Izq, Hijo Der, Padre, Hijo Medio: los enlaces entre los usuarios

3c) **el archivo MENSAJES.NIV**

* Cantidad de nodos: longint

4c) **el archivo MENSAJES.CON** (control)

* Raíz: al árbol MENSAJES.NTX
* Borrados
* Ultimo Nivel: el último nivel en MENSAJES.NIV
* Porcentaje Niveles: rango 0..100. Indica cual es el porcentaje de niveles sobre los cuales se hará el control de balanceo.

|  |
| --- |
| **APLICACIÓN** |

Se le pide al alumno que desarrolle un programa que contenga en su interior el siguiente conjunto de opciones:

1. **USO PARA VISITANTES**

* Módulo de registración. Es un simple ALTA donde el visitante deberá ingresar los datos bajo los cuales se registrará en el sistema (email, clave, etc). El email del usuario NO puede estar repetido. Normalizar todo a mayúsculas o minúsculas.

1. **USO PARA USUARIOS REGISTRADOS**

* Acceso al sistema, a través de su email y su clave personal. Si el usuario está bloqueado, se le deberá notificar y no podrá continuar.
* Baja y modificaciones como usuario del sistema. Si el usuario tiene ventas publicadas NO podrá darse de baja hasta que no las cierre o pause.
* ABM (publicación) de artículos para la venta. No se puede dar de baja ni modificar las publicaciones cerradas, bloqueadas o pausadas. Se deberá incluir un mecanismo para subir una foto a la publicación.
* Listado de los artículos publicados, indicando fechas, estado de la publicación, datos del comprador, su calificación y si se ha pagado la comisión a MercaTrucho.
* Visión de los artículos publicados. Para ello, deberá elegir una categoría y allí filtrar a través de los siguientes criterios: artículos nuevos / artículos usados / ordenar por mayor - menor valor / una provincia / etc. También se deberá disponer de un método de búsqueda a través de los campos <nombre del artículo> y <descripción>, dándole prioridad al primero en el listado.
* Acceso a una publicación y poder allí formular preguntas, consultar las respuestas y, eventualmente, comprar el artículo.
* Acceder a todas sus compras realizadas, con fechas, artículo, vendedor, importe y resto de datos que sirvan para su identificación.

1. **USO PARA ADMINISTRADORES**

A este módulo, solo tendrán acceso aquellos usuarios que se acrediten con clave de administrador. Tiene la finalidad de generar la carga inicial a algunos ME del sistema, así como acceder a reportes globales y poder intervenir ante eventualidades.

Sobre el **ME de Categorías**

* ABM (manual) de categorías.
* Listado General de Categorías, indicando la cantidad de publicaciones que tiene cada una
* Listado de las <n> categorías con mayor cantidad de artículos, ordenadas de mayor a menor, desglosando: cantidad de artículos en venta; cantidad de ventas concretadas; cantidad de publicaciones pausadas; cantidad de publicaciones bloqueadas.

Sobre el **ME de Usuarios**

* Bloqueo / desbloqueo de cuentas de usuarios.
* Listado General de Usuarios bloqueados. Debe incluir nombre, Id\_Usuario, teléfono, etc. de cada uno (línea por línea).
* Listado de los <n> usuarios que hicieron más ventas en el sistema
* Listado los artículos comprados por un determinado usuario
* Listado de ventas según la calificación y entre fechas

Sobre los **ME de Publicaciones y Conversaciones**

* Bloqueo / desbloqueo de publicaciones
* Agregar (aleatoria y sistemáticamente) un grupo de conversaciones entre una publicación <x> y una cantidad <n> de usuarios existentes. Para el caso, las conversaciones tendrán siempre 2 mensajes fijos (testimoniales):
  + Comprador: “Hola. Me llamo“ + <nombre que corresponda> + “ - Fecha: “ + <Fecha – Hora>
  + Vendedor: “Gracias. Me llamo" + <nombre que corresponda> + " - Vendo " + <nombre del artículo que está vendiendo>

donde <x> y <n> serán solicitado previamente. Esto servirá para hacer una carga exhaustiva de conversaciones.

* Conversaciones de un usuario comprador. Se trata de un listado donde se muestran línea por línea, todas las conversaciones llevadas a cabo por un usuario como comprador. Incluyendo cantidad de mensajes y la última fecha / hora de producidos.
* Conversaciones de un usuario vendedor. Idem al anterior.
* Listado de publicaciones realizadas por un determinado vendedor, con su estado actual, comprador, comisión, etc.
* Listado de ventas realizadas entre 2 fechas. Línea por línea, se deberá detallar comprador y vendedor, el artículo, la fecha y el importe. También el cálculo de la comisión y el estado de cobro. Se deberán totalizar al pie.
* Listado de publicaciones de una categoría determinada que caducaron sin ventas. Entre dos fechas.

**Herramientas de los ME**

* Estado de los ME que contengan estructuras arboreas. En este caso, el de Publicaciones y Conversaciones. Se debe mostrar, línea por línea, los nodos presentes en el árbol (no se incluyen los “Hijos al medio”, en el caso del trinario). Finalmente, para el ME que utilice un archivo de control de niveles, se debe sacar una conclusión respecto a si el árbol debe ser rebalanceado o no. Se informa cuantos niveles quedan excluidos.
* Rebalanceo:proceso donde se realiza el rebalanceo del ME de MENSAJERÍA.
* Un test de Dispersión para los ME que utilicen Hash**.**  Mostrar la dispersión en general y agrupada por <n> bloques definidos por el usuario.

## IMPORTANTE

* Las entradas y salidas deben estar validadas, rechazando valores incorrectos sin que se cancele la aplicación.
* Las entradas y salidas deben ser consistentes y no alterar las relaciones lógicas entre las tablas. .
* El alumno debe desarrollar este programa en **Delphi (cualquier versión)** atendiendo las normativas de la programación estructurada respecto a los Métodos Estructurales, con un completo control de errores (especialmente en lo que respecta a la implementación de la librería operacional).
* Se valorará especialmente toda extensión de los alcances funcionales del presente TP Obligatorio, a los fines de que incluya otras opciones presentes en los programas de venta de artículos por Internet. **Para el caso, se recomienda armar ME a partir de las LO ya existentes.**
* Antes de entregar este trabajo, el alumno deberá verificar el correcto funcionamiento del presente sistema en el Centro de Cómputos de la Institución. Al momento de hacer la defensa, se deberá instalar los ME y ejecutar al menos 2 usuarios en terminales diferentes, que tengan consultas y hagan uso del resto de las opciones de la aplicación. A los efectos, es necesario que la RUTA de red (que formará parte de la RUTA de instalación de los ME COMPARTIDOS) sea solicitada al usuario a la hora de correr la aplicación por primera vez**.**

**De ser necesario, solicitar con tiempo una ruta compartida al encargado del gabinete.**