

Hemos decidido **como ejercicio Práctico Combinado Final** para superar la 3ª Evaluación sin tener que hacer “exámenes” trabajar con una aplicación, un proyecto Java, ya diseñada en parte y comentado en clase. El proyecto ya es Funcional, o casi,... podría mejorarse y podría tener algún error. El proyecto lo componen, principalmente, 2 ficheros: **DBManager**, que implementa una **clase DBManager** que gestiona una Base De Datos en **MySQL** con **JDBC**, y **GestionClientes**, una **aplicación java simple de consola que utiliza la Clase DBManager**. Hay un tercero, también funcional, **GestionClienteGUI**, pruébalo...funciona!!, que **utiliza una interfaz gráfica con Swing**. Es interesante que le echéis un vistazo también para entender cómo funcionan las aplicaciones con interfaces gráficos (GUI).

Al Proyecto le faltan unas cuantas cosas,... **probarlo con el framework JUnit**, **documentarlo**, o **terminar de documentarlo**, y **generar su documentación con Javadoc**, **diseñar unos cuantos diagramas** con el lenguaje de modelado **UML** para describir y **terminar de documentar el proyecto** y, lo más importante, **MODIFICARLO** para que haga unas cuantas cosas más (**añadirle nuevas funcionalidades**) y, de camino, intentar **refactorizarlo** o **limpiarlo** y **eliminar cualquier código “apestoso”** y dejarlo “más limpio”, más claro o entendible. Todo ello utilizando nuestro maravilloso entorno **Eclipse** y sus magníficas prestaciones.

Pero claro, no queremos perder el proyecto de partida y queremos utilizar las posibilidades que nos ofrecen **GitHub** y **Git** para controlar la evolución del proyecto. **Git**, como **sistema de Control de Versiones** y **GitHub** nos van a permitir tener en nuestro/vuestro **repositorio público** una **clonación del proyecto de partida y tantas ramas (versiones) como deseemos**.

Se pide:

- Crear una **cuenta en GitHub** y **comunicársela al profe**, en cuenta se tenga, para que este pueda ver la evolución del proyecto.
- **Crear el proyecto en dicha cuenta** y hacerlo **público** y gestionar la práctica a realizar utilizando las posibilidades que ofrece **GitHub**, **Git** y **Eclipse** en el desarrollo de la misma.
- **Crear las ramas necesarias** para **mantener todas las versiones** desde la inicial (punto de partida o versión v1.0 hasta la final Vn.0 y todas sus posibles versiones previas. **Se propone crear tantas ramas como versiones se quieran tener**.
- **Crear una rama o versión propia** para el proyecto **en su faceta Gráfica**.
- Crear e **incorporar un Paquete** Java para la **clase DBManager** y cualquier otra que se considere en base a la evolución que tenga el proyecto dadas las modificaciones que sufra según se especifica en los próximos párrafos.
- Habrá que **documentar “adecuadamente” el paquete**, las **clases**, **métodos**, etc. con objeto de poder generar una buena documentación con la herramienta **Javadoc**. Habrá que **identificar** claramente al **autor** y las **diferentes versiones**.
- Habrá que generar con el **framework de pruebas JUnit** un **conjunto de pruebas suficiente** para acreditar que el proyecto cumple lo especificado.
- Crear los **Diagramas UML Estructurales** y de **Comportamiento** mínimos que se consideren o se especifiquen (al menos **Diagrama de Clases** y de **Paquetes**, y **Diagrama de Casos de Usos** y **Secuencia**)
- **Implementar algunas de las modificaciones** propuestas a continuación:
 - **Modificar la Clase DBManager** para **utilizar la clase PreparedStatement** en vez de **Statement** con **objeto de mejorar el rendimiento** haciendo los cambios necesarios en el código.
 - **Añadir algún procedimiento almacenado y utilizarlo con la Clase CallableStatement**. **Se puede utilizar una de las opciones de menú que ya se tiene o crear una nueva específica**. Evidentemente, modificando lo necesario.

- Modificar la Clase **DBManager** para que permita manejar cualquier Base de Datos y cualquier Tabla de dicha Base de Datos...
 - Implicará al menos, en una primera modificación-versión, trabajar con cualquier BBDD que tenga una Tabla con 3 campos.
 - Una segunda modificación debería permitir trabajar con cualquier BBDD con cualquier número de Tablas y cualquier número de campos (Esto es para NOTA). Pista... [DatabaseMetadata](#) y [ResultSetMetadata](#).
- Modificar la Aplicación para Añadir varias opciones al menú...
 - Poder crear una Tabla nueva
 - Poder filtrar filas de una Tabla
- Modificar la Aplicación para permitir **VOLCAR** en un fichero una Tabla de una Base de Datos. El contenido del fichero deberá indicar en la primera línea el nombre de la Base de Datos Origen y el de la Tabla Origen, en la segunda las cabeceras de los campos y en las restantes líneas los campos de cada registro por línea.

CONSIDERA QUE SOLO HAY CAMPOS TIPO VARCHAR o ENTEROS EN LAS TABLAS

- Modificar la aplicación para permitir Insertar, Actualizar y Borrar registros a partir de la información contenida en un archivo cuyo contenido y estructura es conocida. Ejemplo de estructura para **INSERTAR**:
 1. Primera línea contendrá **el nombre de la Base de Datos**.
 2. Segunda línea contendrá **el nombre de la Tabla**.
 3. Tercera línea contendrá separados por “,” los **nombres de los campos** o sus **posiciones**.
 4. Cuarta línea y siguientes contendrán **los datos de los campos** en el orden indicado en la tercera línea.
- Ejemplo de estructura para **ACTUALIZAR** :
 1. Primera línea contendrá **el nombre de la Base de Datos**.
 2. Segunda línea contendrá **el nombre de la Tabla**.
 3. Tercera línea contendrá separados por “,” los **nombres de los campos** o sus **posiciones**.
 4. Cuarta línea y siguientes contendrán **el identificador (clave)** del registro a modificar y **los datos de los campos** en el orden indicado en la tercera línea.
- Ejemplo de estructura para **BORRAR** :
 1. Primera línea contendrá **el nombre de la Base de Datos**.
 2. Segunda línea contendrá **el nombre de la Tabla**.
 3. Tercera línea contendrán separados por “,” **los identificadores (clave)** de los registros a borrar.

Son solo ejemplos posibles. Hay muchas maneras de organizar esa información. Tu decides.

NOTAS A TENER EN CUENTA

- Cuantas **más puntos especificados** a realizar se hagan **MÁS NOTA** se obtendrá. No obstante **habrá un mínimo a cumplir** para obtener el 5 o Práctica superada. Este mínimo se irá fijando/concretando en clase conforme se avance y se vea la evolución del alumnado.
- Si es necesario, habrá de instalarse un **SGBD** (MySQL o MariaDB o cualquiera que fuese necesario para el correcto funcionamiento de las App).
- Se podrán **añadir funcionalidades y tomar decisiones** sobre las modificaciones propuestas **para mejorar lo especificado**. Se tendrá en cuenta para mejorar la nota.

ENTREGA**➤ ANTES DEL DOMINGO 29 DE MAYO DE 2022**

- Para la correcta entrega, evaluación y calificación el alumno deberá entregar una **Memoria Técnica Breve de la Práctica realizada**. Esta Memoria debería explicar brevemente lo realizado de forma adecuada. Podrá ser utilizado el típico README.md de GitHub para este cometido, si se considera conveniente.
- Habrá de enviarse un email al profe (pfranciscojurado@iesplayamar.es) comunicando la ENTREGA y adjuntando el pdf de la Memoria Técnica realizada.
- Toda la documentación (documentos) generada deberá estar ubicada en las carpetas del Proyecto en la cuenta de **GitHub** del alumno.

RECUPERACIÓN Y/O MEJORA

- Caso de NO SUPERAR la práctica los mínimos exigidos para considerar superada la 3ª evaluación de BBDD y EEDD. El alumno podrá CONTINUAR durante el periodo de recuperación y mejora dicha práctica. Este periodo, como ya sabéis, corresponde a los días posteriores al 31 de mayo (fecha de la 3ª sesión de evaluación) hasta finalizar el periodo lectivo del curso 24 de junio de 2022.
- A efectos meramente informativo, se comunica que el **22 de junio, miércoles**, se desarrollarán las **Pruebas Finales** (o de recuperación 1ª, 2ª y 3ª evaluación) de **BBDD (3h)** y **EEDD (3h)**. Serán Teórico-Prácticas.
- Aquellos alumnos que **NO hayan superado** la 1ª y 2ª evaluación deberán entregar las actividades propuestas durante el curso o las que se le indiquen al inicio del periodo de recuperación (se hará de forma individualizada) para cada uno de los módulos. De esta forma **podrán evitar tener que presentarse a las pruebas finales** si así lo considera el profesor.
- Aquellos que **NO hayan entregado a fecha 21 de junio** la Práctica Combinada Final propuesta **deberán presentarse a la recuperación de la 3ª evaluación**, al menos.