

Fecha tope de entrega **PROTOTIPO: 14 de MAYO de 2020 a las 23:55 (posible Presentación 15 mayo por videoconferencia)**

Fecha tope de entrega **FINAL: 3 de JUNIO de 2020 a las 23:55 (pendiente de confirmar)**

Posible presentación FINAL a los profesores implicados por videoconferencia los días **4-5 de junio**.

Se valorarán criterios de calidad de código: eficiencia en tiempo y espacio, control de excepciones y aviso al usuario de forma gráfica, código formateado y diseño de ventanas trabajado, usable e intuitivo.

Se pueden añadir campos a las tablas, ventanas y cualquier componente visual así como nuevas funcionalidades (primero cubriendo las aquí descritas)

EXPERIENCIA TURÍSTICA EN CIUDAD ESPAÑOLA

Mediante la recogida de requisitos del grupo correspondiente del ciclo GIAT de Turismo, debéis crear una aplicación gráfica mediante un proyecto de JAVA-FX, con conexión a base de datos, para la gestión de información relativa a elementos para crear experiencias asociadas a actividades a realizar en la ciudad española que os haya correspondido.

Los **requisitos mínimos** serán almacenamiento y gestión de:

- **Actividades** de las que se indicará identificador, imagen, url.... y tipo al que pertenece existiendo los siguientes:
 - **Que visitar:** parque, museo, monumento, ruta...
 - **Donde comer:** restaurante....
 - **Donde dormir:** hoteles
- **Creación de packs:** donde se realizará un presupuesto económico de una selección de actividades, indicando entre otros datos, fechas de validez y creación.

ARQUITECTURA DE LA APLICACIÓN

El proyecto estará estructurado en capas independientes entre sí, por tanto trabajaremos con varios packages, mínimo:

- **MODELO**, se almacenarán las clases base que representaran a: ACTIVIDADES y PACKS
- **VISTA**, contendrá las clases que representan a la interacción con el usuario, es decir ventanas de cada uno de los requisitos para recoger ó mostrar información y sus controllers. Deben llamarse igual y **no debe aparecer "FXMLDocument", cambiar los nombres desde el principio.**
- **DATOS.BDA**, tendrá clases encargadas de las operaciones con la base de datos. Deben recibir y/o enviar objetos de las clases base. Contendrán todo lo relativo a JDBC.
 - una **clase xxxDAO.java** por cada tabla (p.e. ActividadesDAO), que ofrezca los métodos para implementar un CRUD (Create, Read, Update, Delete), es decir las operaciones de mantenimiento de datos de esa tabla.
 - Solo debe existir **una única conexión** y **NO DEBE SER STATIC**. Para ello deberás pasar como parámetro la conexión creada de unas ventanas a otras.
- **DATOS.FICHEROS**, que tendrá clases encargadas de las operaciones sobre ficheros. Deben recibir y/o enviar objetos de las clases base. Los objetos File o Path deben ser recogidos/enviados a la capa vista.
- **otras capas recomendadas**: inicio, imágenes/resources.....

REQUERIMIENTOS TÉCNICOS

El proyecto deberá cumplir los siguientes **aspectos y funcionalidades** según momento de la entrega:

PROTOTIPO

Puntuará un 30% de la nota del proyecto



(PROTOTIPO) Creación de PACKS

- control de datos y formatos
- control de valores not null
- fechas (*DatePicker*), control de valor no anterior a hoy. Indicar el número de días
- **selección de ACTIVIDADES** (implementado en otro fxml e indicará el pack en la parte superior):
 - se mostrarán *todas las actividades* existentes en la base de datos y se permitirá elegir al usuario actividades. Mostrar la información en componentes *TableView* ó *TreeView* ó como queráis (interesante: creación en ejecución de botones/imágenes/checkbox con imagen que representen a cada actividad)
 - se solicitará su precio y el número de plazas, calculando el precio total
 - fecha inicio y fin, indicando número de días totales
- Se mostrará el precio final del pack.
- Confirmación del pack y su conjunto de actividades elegidas, y su inserción en la bda.



(PROTOTIPO) Incorporación de **DOS NUEVOS COMPONENTES javafx** de una librería externa:

- ejemplos de librerías: *Controlsfx*, *jxtras*, *jfoenix*...
- recordar incorporarla a vuestro proyecto usando ruta relativa y guardarla en una carpeta `\lib` dentro del proyecto

PROYECTO FINAL

Puntuará un 70% de la nota del proyecto

✚ PRINCIPAL:

- dará acceso al resto de funcionalidades, mediante: botones con imágenes, menús.... (decidid vosotros)

✚ Mantenimiento de ACTIVIDADES (creación, modificación, borrado)

- control de información y formatos, usar algún tipo de máscara o plantilla de manera que solamente permita introducir unos datos (por ejemplo en url uso de *TextFormatter*)
- control de valores not null
- visualización de imagen
- Al definir su tipo y subtipo, estos valores deberán ser objetos de las clases base, por tanto mostraran la descripción pero se utilizara su id
- cargar la pagina web con la url introducida (componente *webview*)
- Al modificar un dato que el campo cambie de color. Nunca permitir claves primarias.
- Al borrar, comprobar previamente si es referenciada por alguna experiencia, y preguntarle al usuario si está seguro de querer borrar esa actividad y hacerla desaparecer de los packs.

✚ FILTRO sobre ACTIVIDADES mínimo por dos campos:

- Al elegir las actividades de un pack permitir el filtrado
- Mostrar la información en componentes *TableView* ó *TreeView* ó como queráis (interesante: creación en ejecución de botones/imágenes/checkbox según los resultados de búsqueda)
- (OPCIONAL) Investigar para aplicar *Tableview sorting and filtering*

✚ Aspecto visual de las ventanas trabajado, usable e intuitivo.

✚ Todos los **ERRORES** que se generen deben recogerse en la capa de vista y mostrados al usuario mediante formato visual, mínimo Alert

- (OPCIONAL) investigar alertas: clase *Notifications* de la librería *controlsfx*

✚ La aplicación debe tener un **LOGO** que se visualizará como icono de la aplicación

EXTRAS OPCIONALES (solo si todo lo anterior está realizado y comprobado)

Puntuará un 10% extra, si todo lo anterior ha sido realizado

+ Tratamiento de **datos en FICHEROS**, usando *DirectoryChooser*:

- **Exportación** de datos a archivo de texto.
 - Al confirmar la creación de un pack, ofrecer la posibilidad de exportar sus datos incluyendo su lista de actividades y sus precios.
 - Solicitar directorio de almacenamiento, situándolo inicialmente en una carpeta del proyecto
 - Nombre de archivo generarlo usando el id y nombre del pack.
 - Avisad del resultado de la generación del archivo y la ruta y nombre del archivo.

+ Control de acceso:

- Ventana de **login**, control de usuarios mediante contraseña (*PasswordField*) una tabla usuarios. Si los datos son correctos dará acceso a la ventana principal y por tanto a la aplicación, sino avisará del problema.
- **cierre sesión**, cerrando la conexión a base de datos y solicitando de nuevo usuario y contraseña.
- (OPCIONAL) **encriptación de contraseñas** (ejemplo: librería *jasypt*)
- (OPCIONAL) mostrar **imagen** y **nombre** de usuario en todas las ventanas