

FACULTADA DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN  
PROGRAMACIÓN ORIENTADA A OBJETOS  
PROYECTO PRIMER PARCIAL – 2017-I  
**Sistema de Control de Revisiones de Formato de FIEC**

## Problemática:

Cada semestre los estudiantes que finalizan su proyecto de Materia Integradora, Maestría y/o Examen Complexivo, deben presentar de forma impresa el documento final del proyecto utilizando el formato especificado por FIEC.

Al momento el proceso de recepción, revisión, entrega de estos documentos se los lleva de forma manual, es decir el estudiante entrega a la secretaria el documento impreso, la secretaria genera el decreto mediante el cual el documento es asignado a un profesor cuyo rol sea de revisor. El revisor consulta a la secretaria si hay documentos por revisar, si existen documentos por revisar, el revisor firma la hoja de recepción del documento y los retira.

Desde el momento en que se asigna el documento el revisor tiene 7 días para revisarlo. Un revisor puede revisar N veces un documento, es decir que, al finalizar la revisión, este indicará si el documento está aprobado o no, dependiendo si el estudiante cumplió o no con el formato. Una vez que el documento ha sido revisado, este es llevado nuevamente a la oficina de la Secretaría del Sub-Decanato para que se lo devuelvan al estudiante en caso de ser necesaria su corrección o para que le indiquen que el documento fue aprobado. El estudiante debe estar permanentemente consultando en secretaria si hay alguna novedad respecto a la revisión y/o puede recibir una llamada telefónica de la secretaria.

## Solución

Implementar un “Sistema de Control de Revisiones de Formato para FIEC” usando JAVA y Principios de Programación Orientada a Objetos.

El sistema tiene 3 tipos de usuarios: Secretaria, Estudiante y Profesor Revisor

- Estudiante:
  - Solicita revisión de documento físico de proyecto de graduación
  - Retira documento.
- Secretaria:
  - Recpta los documentos físicos de los proyectos de graduación de los estudiantes.
  - Asigna y notifica de una revisión al profesor revisor
  - Entrega el documento físico a revisar
  - Notifica la finalización de una revisión y Entrega el documento físico al estudiante
- Profesor Revisor:
  - Revisa el documento e indica si está aprobado o no.

El sistema debe presentar al usuario sólo las opciones válidas de acuerdo a sus credenciales. El identificador del usuario dentro del sistema es un username asignado por Espol.

## Iniciar Sesión

El usuario inicia sesión con su nombre de usuario y contraseña ESPOL. El archivo usuarios.csv contiene la lista de los usuarios registrados en el sistema y su rol. El sistema compara las credenciales ingresadas por el usuario en el archivo usuarios.csv. Si existe alguna coincidencia de nombre de usuario y contraseña el usuario puede

acceder al sistema, el cual presentará opciones según su rol. Caso contrario, se le presenta un mensaje de error. Y deberá registrarse en el sistema.

### Registro Estudiantes en el sistema

Si un usuario no existe en el sistema, el usuario podrá registrarse. Esta opción está destinada solamente para ESTUDIANTES. La información que se solicitará esta opción es: Nombres Completos, usuario, clave, matrícula y correo electrónico. El sistema creará una entrada en el archivo usuarios.csv y automáticamente le asignará el rol de Estudiante.

### Solicitar Revisión

El estudiante es el único que solicita una revisión, para lo cual deberá ingresar el tipo de documento del que va a solicitar revisión Materia Integradora, Maestría, Examen Complexivo y el título del proyecto. El sistema deberá almacenar esta información y deberá añadir la fecha en que el estudiante realizó la solicitud, en un archivo de texto plano llamado solicitudes.csv, el cual contiene toda la cola de revisiones por realizar.

### Receptar documentos para Revisión

El estudiante lleva su documento impreso y lo entrega a la secretaria. La secretaria debe verificar en el sistema si el estudiante realizó la solicitud de revisión. Es decir, se verifica en el archivo solicitudes.csv si es que existe alguna entrada creada por el estudiante. Si la solicitud existe, entonces cambia el estado de la solicitud en el archivo solicitudes.csv a En-Proceso y se crea un registro nuevo en el archivo revisiones.csv que es donde está la información de todos los documentos que requieren revisión. Cuando se ingresa un registro a revisiones.csv se debe agregar un estado al registro el cual será Por-Asignar.

Si la solicitud no existe, se mostrará un mensaje de error indicando que la solicitud no ha sido creada y no se receptará el documento para revisión.

### Asignar documentos para Revisión

La secretaria ingresa a esta opción e ingresa el número de matrícula del estudiante. Si es la primera revisión que se registra en el sistema entonces el sistema le mostrará a la secretaria un resumen de los profesores y el número de documentos que se encuentran revisando, es decir, cuyo estado en revisiones.csv sea En-Revisión y le solicitará que seleccione el revisor. Si el documento corresponde a una segunda revisión o más, el sistema asignará automáticamente el documento al profesor que lo ha venido revisando. Para ambos escenarios, el sistema cambiará el estado del documento a Retirar, añadirá la fecha en que fue asignado el documento y enviará un correo electrónico al profesor notificándole que tiene una nueva revisión.

### Consultar Revisiones

El profesor revisor ingresa a esta opción e indica si quiere ver las revisiones Pendientes, es decir, en estado En-Revisión o en estado Retirar, y Finalizadas con las resoluciones Aprobadas y Por Corregir(todo esto se consulta usando el archivo revisiones.csv). El sistema le mostrará al usuario un listado de todos los documentos dependiendo de la opción seleccionada, ordenadas desde las más antigua a la más reciente y el tiempo transcurrido desde la asignación. Si el tiempo ha excedido los 7 días de revisión, deberá aparecer en pantalla la palabra Excedido al final de cada registro, esto solamente aplica para revisiones que no se encuentran Aprobadas y cuyo estado sea En-Revisión o Retirar.

El estudiante verá únicamente la información del último documento que envió a revisión, de tal forma que podrá hacer seguimiento del estado del mismo.

### Entregar documentos a Profesor Revisor

La secretaria ingresa a esta opción, el sistema le pide ingresar el número de matrícula del estudiante. El sistema automáticamente agrega la fecha de entrega al revisor y cambia el estado del documento a En-Revision en el archivo revisiones.csv. Para efectos de constancia, la secretaria utiliza un documento físico para registrar la firma del profesor que recibió el documento. A partir de este momento, el profesor tiene 7 días para revisar el documento asignado.

### Finalización Revisión

El profesor ingresa a esta opción y el sistema solicita ingresar el número de matrícula del estudiante cuya revisión ha finalizado y el resultado de la revisión AP para Aprobado y PC para Corrección. En caso de que el estado sea PC, el revisor deberá llenar el formulario correspondiente a la revisión (Se adjunta formato de formulario) indicando las opciones por las cuales el documento no cumple con el formato y alguna observación adicional que quiera realizar. Este formulario deberá almacenarse en un archivo TXT con el nombre form-rev-codigorevision.txt. Adicionalmente, sistema deberá cambiar el estado del registro, en el archivo revisiones.csv, a Revisado.

### Recepción Documento Revisado

La secretaria ingresa a esta opción para indicar que el profesor le ha entregado el documento revisado. Esto lo realiza ingresando el número de matrícula al sistema y confirmando que lo ha recibido. El sistema deberá cambiar el estado de la revisión del documento a Finalizado y enviar un correo electrónico al estudiante indicando que ya puede retirar su documento de secretaría y la resolución final del revisor Aprobado o Por Corregir, si la resolución es Por Corregir el texto del mail deberá mostrar las opciones marcadas en el formulario que requieren revisión. Finalmente, en caso de que la resolución de la revisión sea Por corregir, se vuelve a realizar todo el proceso de nuevo.

## Archivos CSV

Un archivo csv (comma separated values) son un tipo de documento donde cada línea representa un registro de una tabla. Cada línea tiene valores separados por coma, estos valores representan datos correspondientes a las columnas de un registro en una tabla. Los archivos csv que se manejarán son:

- **usuarios.csv:** Tiene la información organizada de la siguiente manera:

| Nombres y Apellidos      | Usuario  | Clave  | Matrícula | Correo Electrónico    | Rol |
|--------------------------|----------|--------|-----------|-----------------------|-----|
| Verónica Carrasco Idrovo | vcarrasc | 123456 |           | vcarrasc@espol.edu.ec | P   |
| Elena Perez Herrera      | eperezhe | 123456 |           | eperezhe@espol.edu.ec | S   |
| Miguel Alvarez Tandazo   | malvarez | 123456 | 200000000 | malvarez@espol.edu.ec | E   |

### Vista archivo CSV

```
Verónica Carrasco Idrovo,vcarrasc,123456,,vcarrasc@espol.edu.ec,P
Elena Perez Herrera,eperezhe,123456,,eperezhe@espol.edu.ec,S
Miguel Alvarez Tandazo,malvarez,123456,200000000,malvarez@espol.edu.ec,E
```

- **solicitudes.csv**

| Codigo | Matrícula | Tipo Documento              | Título                              | Fecha           |                | Estado     |
|--------|-----------|-----------------------------|-------------------------------------|-----------------|----------------|------------|
|        |           |                             |                                     | Inicio          | Fecha Fin      |            |
| 1      | 200000000 | Informe Materia Integradora | Diseño de central recinto Taguaduja | 25/5/2017 13:00 | 29/5/2017 9:00 | Finalizado |
| 2      | 200000000 | Informe Materia Integradora | Diseño de central recinto Taguaduja | 29/5/2017 13:00 |                | Iniciado   |

### Vista archivo CSV

```
1,200000000,"Informe Materia Integradora","Diseño de central recinto Taguaduja","2017/05/25 13:00:00","2017/05/29 09:00:00",Finalizado
2,200000000,"Informe Materia Integradora","Diseño de central recinto Taguaduja","2017/05/29 13:00:00",,Iniciado
```

- **revisiones.csv**

| Códig<br>o | Estudiant<br>e | Revisor       | F.<br>Recepció<br>n | F.<br>Asignació<br>n | F.<br>Entrega<br>Revisor | F. Inicio<br>Revisión | F. Fin<br>Revisión | Resolució<br>n | Estado     |
|------------|----------------|---------------|---------------------|----------------------|--------------------------|-----------------------|--------------------|----------------|------------|
| 1          | 200000000      | vcarrasc<br>o | 29/5/2017 13:00     | 29/5/2017 13:00      | 29/5/2017 13:00          | 29/5/2017 13:00       | 29/5/2017 13:00    | PC             | Finalizado |
| 2          | 200000000      | vcarrasc<br>o | 29/5/2017 13:00     |                      |                          |                       |                    |                | En-Proceso |

```
1,200000000,vcarrasc,"2017/05/29 13:00:00","2017/05/29 13:00:00","2017/05/29 13:00:00","2017/05/29 13:00:00","2017/05/29 13:00:00",PC,Finalizado
2,200000000,vcarrasc,"2017/05/29 13:00:00",,,,,,En-Proceso
```

## Enlaces de Utilidad

<http://docs.oracle.com/javase/8/docs/api/java/util/List.html>

<https://docs.oracle.com/javase/tutorial/essential/io/file.html>

<https://www.mkyong.com/java/javamail-api-sending-email-via-gmail-smtp-example/>

## Consideraciones

- El programa debe ser escrito en lenguaje Java
- Deberán mostrarse en pantalla todos aquellos mensajes que considere necesarios para que el usuario utilice el programa adecuadamente. (Piense siempre en el Usuario)
- Cualquier consideración adicional deberá estar correctamente documentada en el programa.

## Rúbrica (25puntos)

### Funcionalidad (40 %)

- Se valorará la apariencia del proyecto, sin que esto implique que el estudiante deba utilizar recursos adicionales a los aprendidos en clase. (El proyecto es en consola)
- Se verificará el cumplimiento de toda la funcionalidad requerida, así como las correctas validaciones de los datos.
- El proyecto debe ser debidamente probado antes de presentarlo al profesor. Por lo tanto Se espera que el programa no se caiga al ejecutarlo, ni tenga un comportamiento no esperado (Si se cae en la presentación, son 5 puntos menos por cada caída)

### Documentación (10 %)

- El programa fuente deberá presentar un código documentado internamente (un correcto uso de comentarios).
- Diagrama de Clases (relaciones, herencias, no multiplicidad, modificadores de acceso y estático). Esto se puede mandar como avance, en el avance más que todo hacerles notar si les hace falta alguna relación u objeto importante.

### Abstracción y Uso de Objetos (30 %)

- Creación de Clases pertinentes con sus propiedades y métodos
- Correcta interacción de objetos. Que los objetos se comuniquen entre si y no solo sean llamados todos en el main.
- Usar recursos de herencia e interfaces para la solución.

### Modularidad y Encapsulamiento (20 %)

- Dividir el problema usando los métodos correspondientes. No escribir bloques inmensos de código.
- Encapsular correctamente el proyecto. Crear paquetes donde se agrupen clases que se relacionen.
- No olvidar que para acceder a las propiedades de las clases se debe proveer los métodos get y set.

### Penalidades

10 por ciento por no seguir buenas prácticas de programación (convención para nombre de clases, nombre de atributos y métodos, uso de constantes cuando es necesario, o algún fallo grave que se detecte en la forma de programar).

### Modo de Entrega:

- Subir un zip de su proyecto al link de la tarea en el sidweb.
- Subir un archivo en word con la documentación del proyecto: diagramas de clases.