UNIVERSIDAD ICESI

Departamento de TIC
Algoritmos y Programación 1
Tarea Integradora 2 - Unidad 2

En la unidad 2 de nuestro curso hemos trabajado en el análisis de un problema construyendo un modelo con los elementos que intervienen en el problema y especificando los servicios que el programa debe ofrecer, bajo el paradigma de programación orientado a objetos.

Aprendimos a construir las clases que implementan el modelo de la solución del problema, identificando de manera informal los métodos de una clase y clasificarlos en métodos constructores, de consulta y de modificación. Utilizamos una arquitectura para un programa que permita repartir de manera adecuada las responsabilidades entre la interfaz de usuario y el modelo de la solución, y cómo relacionar dichos componentes. Finalmente, hemos aprendido a relacionar todos los conceptos vistos en las dos primeras unidades del curso.

Esta tarea integradora presenta una actividad en la cual se requiere aplicar todos los conocimientos adquiridos en la unidad 2. Por tanto, esta tarea es un instrumento para verificar el cumplimiento de los objetivos que han sido planteados para la unidad 2 descrita en el programa del curso.

Para llevar a cabo este ejercicio es necesario realizar las actividades listadas a continuación:

Actividades

Lleve a cabo las siguientes actividades de cada una de las etapas de desarrollo de software:

1. Análisis del problema (Tabla de especificación y Lista de requerimientos funcionales en el formato visto en la clase de ingeniería de software 1, descárguelo aquí).
2. Diseño de la solución.
    - Elabore un diagrama de clases que modele la solución del problema de acuerdo con las buenas prácticas y los patrones de diseño revisados hasta el momento en el curso. Su diagrama debe incluir el paquete modelo y el de interfaz de usuario.
    - Tabla de trazabilidad entre el análisis y el diseño (Link).
3. Implementación en Java. Incluya en la implementación, los comentarios descriptivos sobre los atributos y métodos de cada clase. Recuerde que todos los artefactos generados en la fase de diseño e implementación deben ser en inglés.
4. Documentación en JavaDoc (Debe entregarse el JavaDoc generado y ubicarlo en la carpeta doc).
5. Usar GitHub como repositorio de código fuente y documentación utilizando la estructura de carpetas aprendida en clase. Recuerde que se debe evidenciar su avance a través de los días en el desarrollo de su tarea.

UNIVERSIDAD ICESI

Departamento de TIC
Algoritmos y Programación 1
Tarea Integradora 2 - Unidad 2

Recuerde que puede encontrar la Rúbrica de la tarea integradora en el siguiente link.

Nota:

- Usted debe entregar la URL de su repositorio GitHub donde se deben encontrar los archivos de codificación en sus respectivos paquetes.
- Tenga en cuenta que su repositorio GitHub debe presentar una estructura base como por ejemplo:

  KnowledgeUnit/
  src/
  bin/
  doc/
- Dentro de los directorios **src/** y **bin/** estarán presentes estos directorios(representando cada uno de sus paquetes):

  ui/
  model/
- El directorio src (source code) contiene sus clases .java dentro del directorio ui/ y model/. Por otro lado, el directorio bin (binary files) contiene los archivos .class en el directorio ui/ y model/. El directorio doc tendrá toda la documentación de análisis y diseño
- Su código debería compilar de acuerdo con lo explicado en la diapositiva 15 de esta presentación: http://tinyurl.com/y3bd9bg2

A continuación, encontrará un enunciado que narra de forma detallada la situación problemática que se espera usted solucione.

Enunciado

Tomado del proyecto de IngeSoft período 2023-1

GreenSQA es una empresa de Tecnología que trabaja con proyectos de Aseguramiento de la Calidad del Software. La empresa desarrolla proyectos para organizaciones que buscan asegurar productos de software con el propósito de cumplir con niveles de alta calidad. Para conocer más sobre nuestro cliente visite el siguiente enlace.

Nuestro cliente describe el problema de la siguiente manera:

Departamento de TIC
Algoritmos y Programación 1
Tarea Integradora 2 - Unidad 2

"La fuga del conocimiento es muy común en las empresas de IT, esto impacta la calidad del software." *Mauricio Aristizábal, Director de tecnología e investigación aplicada.*

Objetivo: "Retener el conocimiento de los empleados antes que roten a otros empleadores."

A continuación, se presenta la descripción del proceso de captura de conocimiento mediante cápsulas generadas por los colaboradores de la empresa GreenSQA. Una cápsula de conocimiento no es más que un texto donde se describen situaciones, elementos o datos importantes del proyecto. Esta descripción del proceso de captura de información muestra las expectativas que tiene la Organización para lograr el objetivo.

Se tendrá una versión piloto del software, por lo que se mencionará topes al registro de las diferentes actividades.

Gestión de los proyectos: En Green al aceptar un proyecto de un cliente se debe almacenar: nombre del proyecto, nombre del cliente, fecha planeada para el inicio del proyecto y fecha planeada para la finalización del proyecto, el valor correspondiente al presupuesto del proyecto y los nombres y números celulares de los gerentes del proyecto tanto por parte de Green como por parte del cliente. Para la versión piloto se contará únicamente con 10 proyectos.

Gestión de las etapas del proyecto: La ejecución de los proyectos se divide en 6 etapas: inicio, análisis, diseño, ejecución, cierre y seguimiento y control del proyecto. Cada etapa tendrá una fecha de inicio y fin (planeada) y una fecha de inicio y fin (real). Adicionalmente se guarda la aprobación del cumplimiento de la etapa. Cuando se crea el proyecto, automáticamente se crean sus 6 etapas, pero solo la etapa de inicio queda activa. Para lograr asignar las fechas planeadas se deberá solicitar al usuario la cantidad de meses que se lleva cada etapa (arreglo de cantidad de meses).

Culminación de una etapa del proyecto: La etapa culmina registrando la aprobación de la misma, la fecha real de finalización y pasa de activa a inactiva. A su vez procede activar la siguiente etapa, con la fecha real de inicio.

Registrar cápsulas de conocimiento: En cada una de las etapas del proyecto, los colaboradores generan cápsulas de conocimiento. Una cápsula tiene un identificador único, una descripción de la situación que desea registrar, un tipo de cápsula (los tipos definidos hasta el momento son técnico, gestión, dominio y experiencias), el nombre y cargo del colaborador y el aprendizaje o lección aprendida con dicha situación.

Para la versión piloto únicamente se ingresarán hasta 50 cápsulas por etapa.

En el texto de la cápsula, tanto en la descripción como en la lección aprendida, se debe marcar con "#" al inicio y al final de la primera aparición de las palabras claves del tema que se esté tratando que el autor considere importante (Ejemplo #Pruebas Funcionales#). Con estos "#" se espera que el sistema extraiga estas palabras clave y las relacione en una característica de la cápsula llamada *"hashtag".* El texto de la cápsula debe incluir los *"hashtag"* de manera obligatoria. Todas las cápsulas registradas quedan en revisión y pueden ser aprobadas con lo cual serán públicas.

Aprobación de las cápsulas: Al aprobarlas se registrará la fecha de aprobación.

Publicación de las cápsulas a la organización: Las cápsulas de interés organizacional que sean aprobadas serán generadas en formato HTML para ser publicadas en la Intranet de la Organización, por lo cual se almacenará la url.

Consultar cápsulas de conocimiento: Las personas de la organización pueden acceder a las cápsulas de conocimiento mediante una cadena de búsqueda sobre los textos de las cápsulas o mediante los *"hashtag"*.

El menú solicitado contendrá las siguientes opciones:

- Crear un proyecto
- Culminar etapa de un proyecto
- Registrar cápsula
- Aprobar cápsula
- Publicar cápsula

- Informar al usuario cuantas de las cápsulas registradas hay por cada tipo de cápsula (técnico, gestión, dominio y experiencias)

- Informar al usuario un listado de lecciones aprendidas correspondientes a las cápsulas registradas en los proyectos para una etapa en particular

- Informar al usuario el nombre del proyecto con más cápsulas registradas

- Informar al usuario si un colaborador (por el nombre) ha registrado cápsulas en algún proyecto.

- Informar al usuario las situaciones y lecciones aprendidas de las cápsulas aprobadas y publicadas, de acuerdo a una cadena de búsqueda dada por él mismo. Esta cadena de búsqueda deberá ser encontrada en los hashtag.

UNIVERSIDAD ICESI

Departamento de TIC
Algoritmos y Programación 1
Tarea Integradora 2 - Unidad 2

La entrega de la tarea integradora se encontrará dividida en dos partes:

Parte 1: Se entregará para el 1 de abril los siguientes componentes:

- Análisis
- Diseño
- Implementación en java de:
  - Crear un proyecto
  - Culminar etapa de un proyecto
  - Registrar cápsula
  - Aprobar cápsula
  - Publicar cápsula
- Javadoc generado para las opciones anteriores

Parte 2: Se entregará completa para 22 de abril

Anexo:

Manejo de Fechas

Se recomienda utilizar Calendar ubicado en java.util y SimpleDateFormat para formatear el despliegue ubicado en java.text

Para Obtener la instancia del Calendar :

```
Calendar calendarTime = Calendar.getInstance();
```

De la instancia del calendar se obtiene la fecha actual, con un formato dado:

```
String timeStamp = new SimpleDateFormat("dd-MM-yyyy").format(calendarTime.getTime());
```

Para sumar 2 meses al mes actual :

```
calendarTime.add(Calendar.MONTH, 2);
```

Para ver mas: https://docs.oracle.com/javase/7/docs/api/java/util/Calendar.html