

Proyecto Programado #3

Gestión de tareas

Introducción

Se desea desarrollar un sistema de gestión de proyectos y tareas. Su trabajo consiste en implementar la aplicación que permita registrar tareas, asignarlas y obtener estadísticas de ellas.

Se debe implementar una Base de Conocimiento (BC) a partir del registro de la actividad de los usuarios en el sistema, la cual será incremental y al salir e ingresar del sistema, los registros se mantendrán.

La información deberá ser almacenada en un medio externo para que al ingresar al sistema la BC se carga en su estado previo y al salir se preserve el estado inicial sumado a la nueva actividad registrada.

Software a desarrollar

Su trabajo consiste en implementar una aplicación de escritorio (consola) para administrar la gestión de tareas, proyectos y usuarios, utilizando el lenguaje Prolog y los principios de la Programación Lógica.

El sistema inicia con un Menú Principal donde se puede ingresar a las opciones del menú o salir.

Opciones Administrativas

Para acceder a estas funcionalidades el usuario deberá ingresar por medio del Menú Principal y se deben habilitar las siguientes funcionalidades (Menú Administrativo):

- Gestión de personas
- Gestión de proyectos
- Gestión de tareas

- Buscar tareas libres
- Recomendar persona
- Asignar a tarea
- Cerrar tarea
- Estadísticas

Gestión de personas

El sistema debe permitir tener registros en su BC de las personas (trabajadoras) registradas en el sistema, por cada persona se guarda el nombre, puesto, costo por tarea, rating y tipos de tareas que realiza (lista). El sistema debe tener la opción para mostrar todas las personas almacenadas en la BC, además, debe permitir incluir una nueva persona en la BC solicitándole al usuario que ingrese cada uno de los datos con los que se registran las personas. Los tipos de tareas permitidas son las siguientes: requerimientos, diseño, desarrollo, qa, fullstack, frontend, backend y administracion. En la funcionalidad de mostrar debe indicarse las tareas en las que ha sido asignado cada usuario, indicando proyecto, nombre de tarea y estado.

Gestión de proyectos

El sistema debe tener registros en su BC de los proyectos registrados en el sistema, por cada proyecto se guarda el nombre, empresa, presupuesto, fecha inicio* y fecha fin*. El sistema debe tener la opción para mostrar todos los proyectos almacenados en la BC, además, debe permitir incluir un nuevo proyecto en la BC solicitándole al usuario que ingrese cada uno de los datos con los que se registran los proyectos. En la funcionalidad de mostrar debe desplegar adicionalmente el costo incurrido en el proyecto (suma de costo de cada tarea asignada según el costo del usuario asignado)

Gestión de tareas

El sistema debe tener registros en su BC de las tareas que se crean en un proyecto, por cada tarea se indica el proyecto, el nombre y tipo de tarea. El sistema debe tener la opción para mostrar todas las tareas almacenadas en la BC, además, debe permitir incluir una nueva tarea en la BC solicitándole al usuario que ingrese cada uno de los datos con que se registran las tareas. Al ingresar una tarea se debe indicar que está en estado pendiente y que no tiene asignada persona, ni fecha de cierre. Al mostrar las tareas se debe indicar adicionalmente su estado, persona asignada y fecha de cierre (estos campos pueden estar vacíos).

Buscar tareas libres

El usuario podrá buscar todas las tareas libres (pendientes) en la BC, se debe mostrar toda la información de la tarea. Además, el usuario podrá indicar un nombre de una persona y el sistema mostrará las tareas libres que la persona podría desarrollar considerando el tipo de la tarea y los tipos de tareas que realiza la persona.

Recomendar

El sistema solicitará un nombre de proyecto y un nombre de tarea y a partir de eso mostrará una lista de las personas recomendadas, de mayor a menor recomendación. El algoritmo de recomendación se calculará de la siguiente forma:

- Desarrollo Previo: por cada tarea asignada del tipo la persona suma 2 ptos.
- Afinidad por proyecto: si la persona tiene alguna tarea del proyecto, suma 5 ptos.
- Rating: se le suma el rating de la persona.
- Tareas abiertas: por cada tarea asignada con estado activa se le resta 3 ptos.

El algoritmo suma todos los puntos y enlista de mayor a menor puntuación las personas. Se mostrará por cada persona: nombre, puntuación y tipo de tareas que realiza.

Asignar tarea

El sistema solicitará un nombre de proyecto, un nombre de tarea y el nombre de la persona que se asigna. Si la tarea no tiene nadie asignado, se le asigna y cambia su estado a activa.

Cerrar tarea

El sistema solicitará un nombre de proyecto, un nombre de tarea, el día, mes y año del cierre. Si la tarea no ha sido cerrada, se le asigna la fecha de cierre y se cambia su estado a finalizada.

Estadísticas

El sistema debe tener las siguientes estadísticas:

- a) Estatus financiero de proyecto: por cada proyecto indica su estado:
Tablas: el presupuesto del proyecto es igual a la suma de los costos de las tareas (según la persona asignada).
Sobre costo: el presupuesto del proyecto es menor a la suma de los costos de las tareas.
Bajo costo: el presupuesto del proyecto es mayor a la suma de los costos de las tareas.
- b) Tipo de cierre de proyecto: por cada proyecto indica su tipo de cierre:
Al día: si la fecha máxima de cierre de una tarea es menor o igual a la fecha de cierre del proyecto.
Tardía: si la fecha máxima de cierre de una tarea es mayor a la fecha de cierre del proyecto.
- c) Cantidad de tareas por persona: nombre de persona y cantidad de tareas asignadas.

Inicio y Salida del sistema

Al iniciar el sistema se carga a la BC, desde un archivo, los hechos que mantienen registros del sistema. Al salir (o en el momento de ejecución a conveniencia) se debe guardar en el archivo los nuevos hechos, generados en la ejecución del programa.

Opcionales

Se darán **2.5 puntos adicionales** al entregar a más tardar el miércoles 18 de octubre a las 11:55:55 PM el Documento de Requerimientos, ver plantilla suministrada en el Tec Digital. Debe subirse en la documentación llamada “Proyecto Programado III (archivos adicionales)” debajo de la carpeta de “Proyectos”.

Se darán **10 puntos adicionales** si se genera un sistema adicional (Java o ASP.NET-C#), que muestre el listado de los títulos, donde adicionalmente se guarde una ruta de imagen que se visualice en el listado. El listado debe obtenerse a través de prolog, por medio de una liga.

Aspectos técnicos

El sistema debe iniciarse con un predicado en Prolog, por lo que la ejecución se hará por medio de consola. El usuario podrá mantenerse “navegando” en entre las diferentes opciones del sistema. En caso de requerir librerías adicionales para compilar y ejecutar el programa, deberán especificarlo en la documentación, ya que de lo contrario se descontarán puntos en la evaluación.

Deberán utilizar el sistema de control de versiones GitHub, el repositorio deberá ser público o incluir al profesor en el control de acceso del mismo. Se utiliza para evaluar la correcta gestión del tiempo y trabajo colaborativo.

Documentación

La documentación es un aspecto de gran importancia en el desarrollo de programas, especialmente en tareas relacionadas con el mantenimiento de los mismos.

Para la documentación interna, deberán incluir comentarios descriptivos para cada función, con sus entradas, salidas, restricciones y **objetivo**.

La documentación externa deberá incluir:

1. Portada.
2. Manual de usuario: **instrucciones de compilación, ejecución y uso**.
3. Pruebas de funcionalidad: incluir *screenshots*.
4. Descripción del problema.
5. Diseño del programa: decisiones de diseño, algoritmos usados.

6. Librerías usadas: manejo entradas-salidas, archivos, etc.
7. Análisis de resultados: objetivos alcanzados, objetivos no alcanzados, y razones por las cuales no se alcanzaron los objetivos (en caso de haberlos).
8. Bitácora (autogenerada en git, commit por usuario incluyendo comentario)

Forma de trabajo

El trabajo se debe realizar en parejas.

Evaluación

La evaluación se va a centrar en dos elementos: programación y documentación.

La tarea tiene un valor de **12.5%** de la nota final, en el rubro de Proyectos.

Desglose de la evaluación de la tarea programada:

1. Documentación interna 2 puntos.
2. Documentación externa 8 puntos.
3. Funcionalidad 80 puntos.
4. Revisión del proyecto (según completitud del proyecto y gestión del tiempo) 5 puntos.
5. Hora de Entrega 5 puntos.

Aspectos administrativos

Debe crear un archivo **.zip** ("PP3_Integrante1_Integrante2.zip") que contenga únicamente un archivo **info.txt** y 2 carpetas llamadas **documentacion** y **programa**, en la primera deberá incluir el documento de *word* o pdf solicitado y en la segunda los archivos y carpetas necesarias para la implementación de este proyecto programado, y/o link en git del repositorio. El archivo **info.txt** debe contener la siguiente información (cualidades):

- a. Nombre del curso
- b. Número de semestre y año lectivo
- c. Nombre de los Estudiantes
- d. Número de carnet de los estudiantes
- e. Número del proyecto programado
- f. Fecha de entrega
- g. Estatus de la entrega (debe ser **CONGRUENTE** con la solución entregada):
[Deplorable | Regular | Buena | MuyBuena | Excelente | Superior]

Entrega

Deberá subir el archivo antes mencionado al TEC Digital en el curso de LENGUAJES DE PROGRAMACIÓN GR 60, en la asignación llamada “P3” debajo del rubro de “Proyectos”. En la evaluación del Proyecto el rubro de “Hora de Entrega” valdrá por 5 puntos de la nota total del proyecto, según la siguiente escala:

- a. Si se entrega antes de las 11:55:55 **PM** del lunes 30 de octubre de 2023, 5 puntos.
- b. Si se entrega antes de las 11:55:55 AM del martes 31 de octubre de 2023, 2.5 puntos.
- c. Si se entrega antes de las 11:55:55 PM del martes 31 de octubre de 2023, 0 puntos.

NO SE ACEPTARÁN trabajos que contengan “commits” posterior a esta fecha.

Todo el contenido de cada proyecto debe ser 100% original y en caso de plagio todos los integrantes del grupo tendrán nota cero.

Todos los miembros del grupo deberán participar de la revisión.