



Tecnológico de Costa Rica

Duan Antonio Espinoza

2019079490

Aarón Piñar Mora

2019063903

Lenguajes de Programación

2 semestre

Proyecto 3

Gestión de tareas

2023

Manual de usuario

Antes de empezar con el programa, es importante instalar SWI-Prolog para que sea posible la ejecución del Gestor de tareas, estos serían los pasos:

1. Ve al sitio web oficial de SWI-Prolog: [SWI-Prolog](https://www.swi-prolog.org).
2. Desplázate hacia abajo hasta la sección "Windows Installer" y descarga el instalador que corresponda a tu versión de Windows (32 bits o 64 bits). Es probable que desees descargar la versión "Stable".
3. Ejecuta el archivo de instalación que descargaste. El asistente de instalación te guiará a través del proceso. Puedes elegir la ubicación de instalación y las características que desees instalar.
4. Una vez que la instalación esté completa, SWI-Prolog estará listo para su uso.



Una vez hecho esto, hay que dirigirse al recién instalado SWI-Prolog. En donde seleccionamos en la esquina superior izquierda la opción "file" y luego "consult" para cargar el archivo, posteriormente de haber sido compilado, se introduce en la consola "?- menu_administrativo." para desplegar el menú del programa.

MENSAJE: A continuación rellene la información que se le solicita para crear una Persona en la base de conocimiento.

- Ingrese el nombre de la persona: Jonatans
- Ingrese el puesto de la persona: programador
- Ingrese el rating de la persona: 10

- Ingrese el costo de la tarea: 12
- Ingrese una tarea (o "fin" para finalizar): fin

----- Los datos ingresados son -----

- ✓ Nombre: Jonatans
- ✓ Puesto: programador
- ✓ Costo : ₡ _9290
- ✓ Rating: 10
- ✓ Tareas: qa:13,backend:12

Registrando Tarea
Guardando datos...

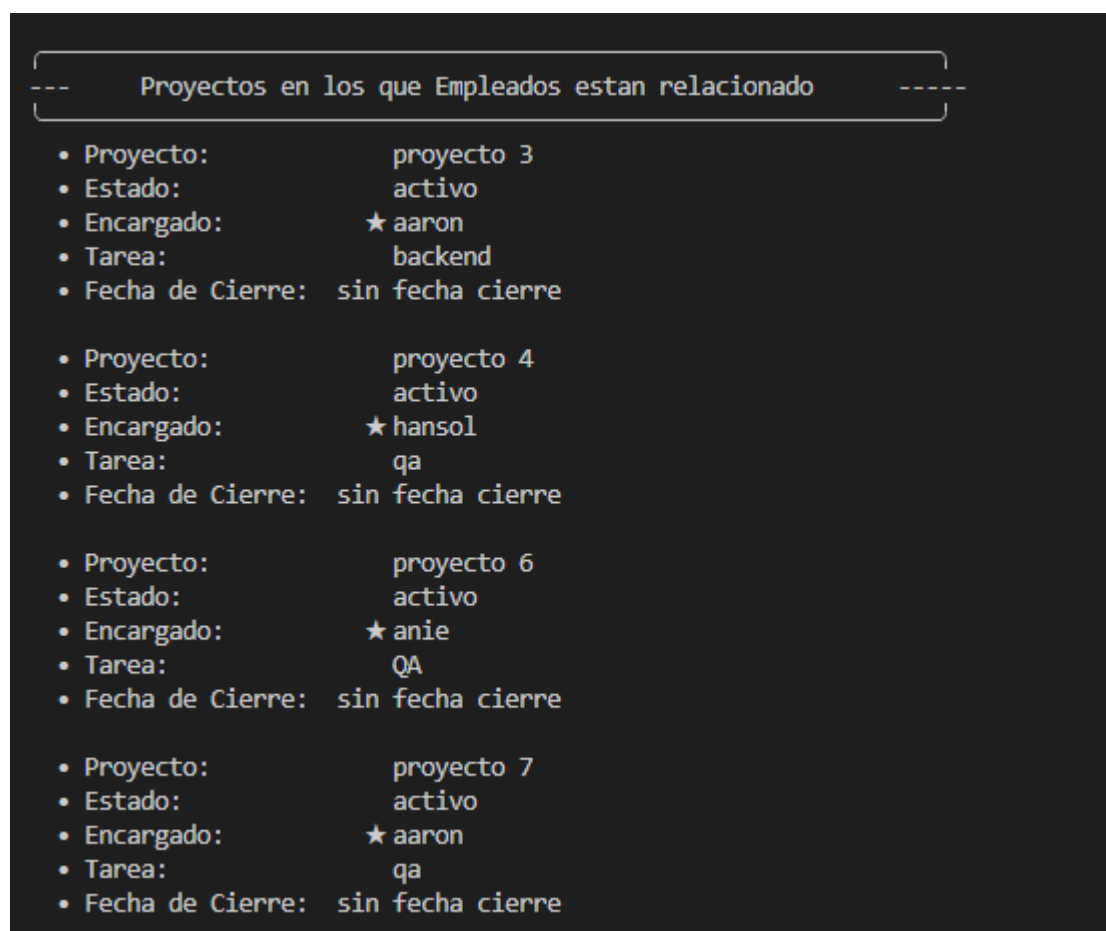
MENÚ GESTION DE PERSONAS

1. Agregar Persona
2. Mostrar Personas
0. Volver

Ingrese una de las Opciones mostradas: █

En la opción 2 “Mostrar personas” de la funcionalidad de gestión de personas, el sistema hará una lectura al archivo de texto y mostrará los datos almacenados en personas.txt, en la opción 3 el usuario regresará a las opciones administrativas

En la opción 2 de las opciones administrativas se desplegará el submenú de gestión de proyectos, se desplegarán las siguientes opciones:



Si el usuario digita la opción 1, el submenú lo llevará a agregar un nuevo proyecto, por medio de la cual se le van a solicitar los siguientes datos: el nombre, empresa, presupuesto, fecha inicio* y fecha fin*

MENSAJE: A continuación rellene la información que se le solicita para crear un Proyecto en la base de conocimiento.

- Ingrese el nombre del Proyecto: proyecto 32
- Ingrese el nombre de la Empresa: TCS
- Ingrese el Presupuesto: 230000
- Ingrese la Fecha de Inicio: 22/02/2020
- Ingrese la Fecha de Fin: 29/02/2020

---- Los datos ingresados son ----

- ✓ Nombre de Proyecto: proyecto 32
- ✓ Empresa: tcs
- ✓ Presupuesto : ¢ 230000
- ✓ Fecha Inicio: 22/02/2020
- ✓ Fecha Fin: 29/02/2020

Registrando proyecto
Guardando datos...

Al seleccionar la opción 2, se mostrarán todos los proyectos registrados en proyectos.txt

--- Proyectos en los que Empleados estan relacionado -----

- Proyecto: proyecto 3
 - Estado: activo
 - Encargado: ★ aaron
 - Tarea: backend
 - Fecha de Cierre: sin fecha cierre
-
- Proyecto: proyecto 4
 - Estado: activo
 - Encargado: ★ hansol
 - Tarea: qa
 - Fecha de Cierre: sin fecha cierre
-
- Proyecto: proyecto 6
 - Estado: activo
 - Encargado: ★ anie
 - Tarea: QA
 - Fecha de Cierre: sin fecha cierre
-
- Proyecto: proyecto 7
 - Estado: activo
 - Encargado: ★ aaron
 - Tarea: qa
 - Fecha de Cierre: sin fecha cierre

En la opción 3 de las opciones administrativas se desplegará el submenú de gestión de tareas, se desplegarán las siguientes opciones

[illegible]

Al seleccionar la opción 1, se mostrarán todas las tareas registradas en tareas.txt

Bienvenido a la Gestión de Tareas

1. Mostrar todas las tareas

2. Agregar una nueva tarea

3. Asignar tarea a persona

4. Cerrar una tarea

5. Buscar tareas libres

6. Salir

Ingrese una opción: 1

MENSAJE: A continuación se mostrará la información de la base de conocimiento registrada con respecto a las Tareas.

Proyecto: proyecto 2

• Estado: pendiente

• Encargado: sin asignar

• Tarea: sin fecha cierre

• Fecha de Cierre:

Proyecto: proyecto 3

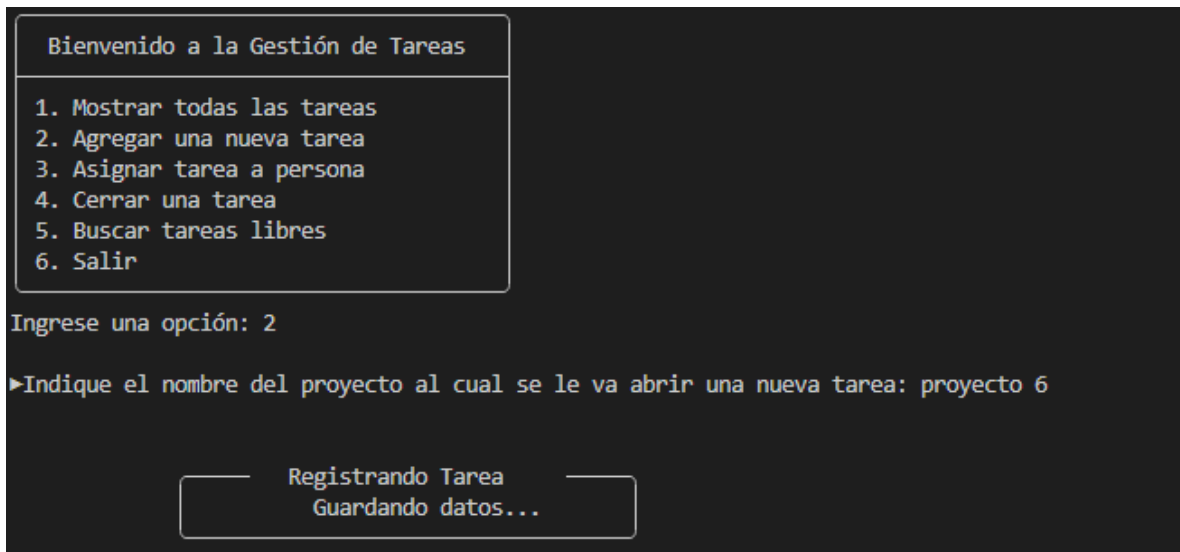
• Estado: activo

• Encargado: aaron

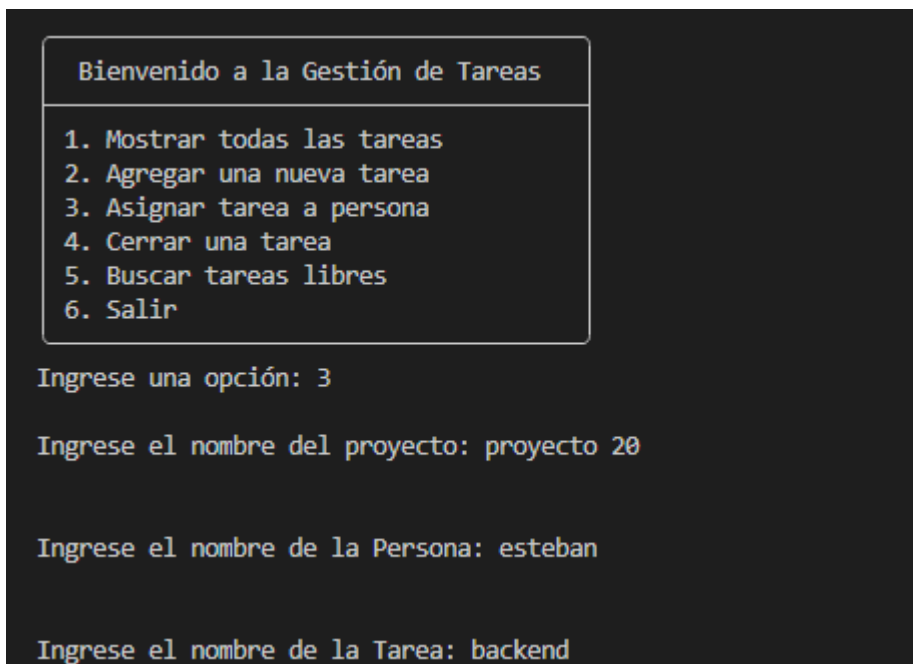
• Tarea: backend

• Fecha de Cierre: sin fecha cierre

Si el usuario digita la opción 2, el submenú lo llevará a agregar una nueva tarea, por medio de la cual se le van a solicitar los siguientes datos: se indica el proyecto, el nombre y tipo de tarea.



Si el usuario selecciona la opción 3, el submenú de la gestión de tareas lo llevará a asignar una nueva tarea, por medio de la cual se le solicitan estos datos: Se solicitará un nombre de proyecto, un nombre de tarea y el nombre de la persona que se asigna.



Si el usuario selecciona la opción 4, el submenú de la gestión de tareas lo llevará a cerrar una tarea, por medio de la cual se le solicitan estos datos: Se solicitará un nombre de proyecto, un nombre de tarea y el día, mes y año del cierre.

>>>adjuntar captura

Si el usuario selecciona la opción 5, el submenú de la gestión de tareas lo llevará a buscar tareas libres (estado pendiente) por lo cual esta va a mostrar toda la información de la

En la captura de pantalla se evidencia el funcionamiento de app.pl en SWI-Prolog, el cual es la funcionalidad que desplegará las opciones que el usuario tendrá disponible para navegar y seleccionar la funcionalidad deseada a través de consola digitando la opción.


```
►Ingrese el costo de la tarea: 12
►Ingrese una tarea (o "fin" para finalizar): fin

----- Los datos ingresados son -----
✓ Nombre: Jonatans
✓ Puesto: programador
✓ Costo : ₡ _9290
✓ Rating: 10
✓ Tareas: qa:13,backend:12

Registrando Tarea
Guardando datos...

MENÚ GESTION DE PERSONAS
1. Agregar Persona
2. Mostrar Personas
0. Volver

Ingrese una de las Opciones mostradas: █
```

Al ser seleccionada la opción 2 mostrará las personas registradas en el archivo “personas.txt”.

--- Proyectos en los que Empleados estan relacionado -----

- ```

• Proyecto: proyecto 3
• Estado: activo
• Encargado: ★ aaron
• Tarea: backend
• Fecha de Cierre: sin fecha cierre

• Proyecto: proyecto 4
• Estado: activo
• Encargado: ★ hansol
• Tarea: qa
• Fecha de Cierre: sin fecha cierre

• Proyecto: proyecto 6
• Estado: activo
• Encargado: ★ anie
• Tarea: QA
• Fecha de Cierre: sin fecha cierre

• Proyecto: proyecto 7
• Estado: activo
• Encargado: ★ aaron
• Tarea: qa
• Fecha de Cierre: sin fecha cierre

```

[illegible]

Se muestra el despliegue de la funcionalidad de gestión de tareas arriba, el submenú mostrará las opciones disponibles

Bienvenido a la Gestión de Tareas

1. Mostrar todas las tareas
2. Agregar una nueva tarea
3. Asignar tarea a persona
4. Cerrar una tarea
5. Buscar tareas libres
6. Salir

Ingrese una opción: 1

MENSAJE: A continuación se mostrará la información de la base de conocimiento registrada con respecto a las Tareas.

Proyecto: proyecto 2

- Estado: pendiente
- Encargado: sin asignar
- Tarea: sin fecha cierre
- Fecha de Cierre:

Proyecto: proyecto 3

- Estado: activo
- Encargado: aaron
- Tarea: backend
- Fecha de Cierre: sin fecha cierre

En la captura se ve como se muestran todas las tareas que se encuentran en tareas.txt

Bienvenido a la Gestión de Tareas

1. Mostrar todas las tareas

2. Agregar una nueva tarea

3. Asignar tarea a persona

4. Cerrar una tarea

5. Buscar tareas libres

6. Salir

Ingrese una opción: 2

►Indique el nombre del proyecto al cual se le va abrir una nueva tarea: proyecto 6

Registrando Tarea

Guardando datos...

Se visualiza que al dirigirse a la opción 2, se agrega una nueva tarea al archivo de texto tareas.txt

Bienvenido a la Gestión de Tareas

1. Mostrar todas las tareas

2. Agregar una nueva tarea

3. Asignar tarea a persona

4. Cerrar una tarea

5. Buscar tareas libres

6. Salir

Ingrese una opción: 2

►Indique el nombre del proyecto al cual se le va abrir una nueva tarea: proyecto 6

Registrando Tarea

Guardando datos...

Al ser seleccionada la opción 3, se visualiza la asignación, en donde modificará el dato seleccionado por el usuario para asignar una “tarea” a una persona, esto se verá en el archivo de texto tareas.txt



```
Bienvenido a la Gestión de Tareas

1. Mostrar todas las tareas
2. Agregar una nueva tarea
3. Asignar tarea a persona
4. Cerrar una tarea
5. Buscar tareas libres
6. Salir

Ingrese una opción: 3

Ingrese el nombre del proyecto: proyecto 20

Ingrese el nombre de la Persona: esteban

Ingrese el nombre de la Tarea: backend
```

Al ser seleccionada la opción 4, se visualiza el cierre de una tarea, en donde modificará el dato seleccionado por el usuario para poner de estado “cerrado” a la tarea asignada, esto se verá en el archivo de texto tareas.txt

Esta funcionalidad muestra todas las tareas pendientes, e incluso puede mostrar las tareas libres para una persona

| MENÚ ADMINISTRATIVO     |
|-------------------------|
| 1. Gestión de Personas  |
| 2. Gestión de Proyectos |
| 3. Gestión de Tareas    |
| 4. Buscar Tareas Libres |
| 5. Recomendar Persona   |
| 6. Asignar a Tarea      |
| 7. Cerrar Tarea         |
| 8. Estadísticas         |
| 0. Salir                |

Ingrese una de las Opciones mostradas: 2

| MENÚ GESTION DE PROYECTOS |
|---------------------------|
| 1. Agregar Proyecto       |
| 2. Mostrar Proyectos      |
| 0. Volver                 |

Ingrese una de las Opciones mostradas: 2

Se muestra el despliegue de la funcionalidad de gestión de proyectos arriba, el submenú mostrará las opciones disponibles

MENSAJE: A continuación rellene la información que se le solicita para crear un Proyecto en la base de conocimiento.

- Ingrese el nombre del Proyecto: proyecto 32
- Ingrese el nombre de la Empresa: TCS
- Ingrese el Presupuesto: 230000
- Ingrese la Fecha de Inicio: 22/02/2020
- Ingrese la Fecha de Fin: 29/02/2020

---- Los datos ingresados son ----

- ✓ Nombre de Proyecto: proyecto 32
- ✓ Empresa: tcs
- ✓ Presupuesto : ₡ 230000
- ✓ Fecha Inicio: 22/02/2020
- ✓ Fecha Fin: 29/02/2020

Registrando proyecto  
Guardando datos...

Se visualiza que se agrega un nuevo proyecto al archivo de texto proyectos.txt

```
--- Proyectos en los que Empleados estan relacionado ----

• Proyecto: proyecto 3
• Estado: activo
• Encargado: ★ aaron
• Tarea: backend
• Fecha de Cierre: sin fecha cierre

• Proyecto: proyecto 4
• Estado: activo
• Encargado: ★ hansol
• Tarea: qa
• Fecha de Cierre: sin fecha cierre

• Proyecto: proyecto 6
• Estado: activo
• Encargado: ★ anie
• Tarea: QA
• Fecha de Cierre: sin fecha cierre

• Proyecto: proyecto 7
• Estado: activo
• Encargado: ★ aaron
• Tarea: qa
• Fecha de Cierre: sin fecha cierre
```

Esta funcionalidad muestra todos los proyectos que están en el archivo de texto proyectos.txt

>>> Al no estar implementada recomendar, no hay capturas disponibles<<<<

Se muestra el despliegue de la funcionalidad de recomendar persona, se le solicita al usuario un nombre de proyecto y un nombre de tarea. A partir de eso mostrará una lista de las personas recomendadas según el algoritmo asignado en el código y los datos en los archivos de texto de tareas y proyectos

>>> Al no estar implementada estadística, no hay capturas disponibles<<<<

Se muestra el despliegue de la funcionalidad de estadísticas, esta funcionalidad mostrará las estadísticas según el algoritmo asignado en el código mediante la lectura de los archivos de texto

## Descripción del problema

El problema que se nos presenta es la creación de un sistema integral de gestión de proyectos y tareas, diseñado para operar en el entorno del lenguaje de programación Prolog. Nuestra tarea central consiste en desarrollar una aplicación capaz de registrar, asignar y proporcionar estadísticas sobre las tareas relacionadas con proyectos.

Para abordar este desafío, se hace necesario establecer una sólida Base de Conocimiento (BC) que actúe como el pilar central de nuestro sistema. La BC tendrá la función de almacenar de manera sistemática y ordenada la actividad de los usuarios dentro del sistema. Un aspecto fundamental de esta BC es su naturaleza incremental, lo que significa que se actualizará de forma continua a medida que los usuarios interactúen con el sistema. Cada acción, tarea o evento registrado por los usuarios se incorporará a esta base de datos de conocimiento, lo que garantiza un seguimiento completo de la actividad del sistema.

Sin embargo, la singularidad de este sistema de gestión de proyectos y tareas radica en su capacidad para preservar el estado de la BC entre sesiones. En otras palabras, cuando un usuario abandona el sistema y luego regresa en sesiones posteriores, la BC debe ser capaz de cargar su estado previo. Esto significa que el sistema retendrá y conservará el conocimiento acumulado en la BC desde el momento en que el usuario anteriormente salió del sistema, y se agregará a este conocimiento toda la nueva actividad registrada en la sesión actual.

Estamos frente a la tarea de implementar un sistema de gestión de proyectos y tareas en Prolog. Para llevar a cabo esta labor, se debe crear una robusta “Base de Conocimiento” que permita la adquisición, asignación y análisis

de tareas. Esta BC será incremental, y lo que es aún más importante, mantendrá la continuidad de la información entre sesiones, garantizando la preservación del estado inicial de la BC y la acumulación de la nueva actividad registrada por los usuarios a lo largo del tiempo. Este enfoque permitirá un seguimiento y análisis efectivos de la evolución de los proyectos y tareas en el sistema.

## Diseño del programa

El programa está diseñado como un sistema de gestión de proyectos y tareas. A continuación, se presentan las decisiones de diseño clave:

1. **Módulos de Funcionalidad:** El programa se divide en tres módulos principales: Proyectos, Tareas y Personas. Cada uno de estos módulos tiene sus propios predicados y se encarga de gestionar la información relacionada. Esto facilita la organización y mantenimiento del código.
2. **Datos Dinámicos:** El uso de hechos dinámicos con predicados **dynamic** permite que el programa mantenga un estado dinámico de la base de conocimientos. Esto significa que los datos, como proyectos, tareas y personas, se pueden cargar y modificar durante la ejecución del programa, lo que brinda flexibilidad y adaptabilidad.
3. **Menú Interactivo:** El programa presenta un menú interactivo que guía al usuario a través de las diferentes acciones disponibles. Los usuarios pueden elegir entre mostrar tareas, agregar tareas, asignar tareas a personas, cerrar tareas, buscar tareas y salir del programa. Esta interfaz amigable facilita la interacción y el uso del sistema.
4. **Carga de Datos:** Antes de realizar cualquier cálculo o recomendación, el programa carga los datos desde archivos externos. Esto garantiza que la base de conocimientos esté actualizada y refleje la información más reciente. Los datos se almacenan en memoria para su posterior manipulación.

## Algoritmos Utilizados:

1. **Módulo de Tareas y Proyectos:**
  - **Carga de Datos:** Para cargar proyectos y tareas desde archivos de texto, el programa utiliza un bucle **repeat** para leer y almacenar los datos en memoria mediante predicados **assert**. Esto garantiza que la base de conocimientos esté actualizada con la información del sistema.
  - **Cálculo de Estatus Financiero:** Para calcular el estatus financiero de un proyecto, el programa suma los costos de todas las tareas relacionadas con ese proyecto. El algoritmo utiliza el predicado

**sum\_list** para calcular el total de costos. Luego, compara este total con el presupuesto asignado para determinar si el proyecto está dentro del presupuesto, por encima o por debajo del presupuesto.

- **Cálculo de Tipo de Cierre:** Para determinar el tipo de cierre de un proyecto, el programa compara la fecha de cierre del proyecto con las fechas máximas de cierre de las tareas asignadas. El algoritmo utiliza **max\_list** para encontrar la fecha máxima de cierre entre todas las tareas. Si la fecha de cierre del proyecto es mayor o igual a la fecha máxima de cierre de tareas, se considera "al día"; de lo contrario, se considera "tardía".

## 2. Módulo de Recomendación de Personas:

- **Cálculo de Puntuación de Personas:** El programa utiliza un algoritmo para calcular la puntuación de cada persona en función de varios factores. Para esto, se consideran:
  - **Tareas Previas:** Se suma un puntaje basado en la cantidad de tareas anteriores realizadas por la persona del mismo tipo.
  - **Afinidad por el Proyecto:** Si la persona tiene alguna tarea en el mismo proyecto, se otorgan puntos.
  - **Rating Personal:** Se utiliza el rating personal de la persona como puntaje adicional.
  - **Tareas Abiertas:** Se restan puntos por cada tarea abierta asignada a la persona.
- **Recomendación de Personas:** Para recomendar personas para un proyecto y una tarea específicos, el programa calcula y almacena las puntuaciones de todas las personas que pueden realizar la tarea. Luego, ordena la lista de personas por puntuación, de mayor a menor, y muestra las personas recomendadas. Esto garantiza que las personas más adecuadas para la tarea estén en la parte superior de la lista.

Este diseño y los algoritmos utilizados permiten que el programa realice tareas de gestión de proyectos, genere estadísticas financieras y recomiende personas para proyectos y tareas según diversos criterios. Proporciona a los usuarios una herramienta eficiente para administrar proyectos y optimizar la asignación de tareas en función de las capacidades y el rendimiento individual. La estructura modular del programa facilita futuras expansiones y modificaciones para adaptarse a las necesidades cambiantes de los usuarios.

## Librerías usadas

1. **library(dialect/sicstus), [read\_line/1]:** Esta librería se utiliza para importar el predicado **read\_line/1**, que permite leer una línea de texto desde la

entrada estándar (teclado) y almacenarla en una variable. Esto es fundamental para interactuar con los usuarios y recibir sus entradas desde el menú interactivo.

2. **dynamic:** Aunque no son librerías en sí, estas directivas **dynamic** se utilizan para declarar los hechos de los predicados como dinámicos. Esto permite que los datos se almacenen en memoria y se actualicen durante la ejecución del programa. En general, facilita la gestión de la base de conocimientos sin necesidad de cargar y descargar datos repetidamente desde archivos.
3. **consult('nombre\_del\_archivo.pl'):** A través de este predicado, el programa puede consultar otros archivos Prolog (en este caso, 'gestionProyectos.pl'). Esta funcionalidad se usa para cargar datos adicionales relacionados con proyectos, lo que permite un mayor modularidad del programa y el uso de múltiples archivos para organizar la lógica.
4. **Predicados personalizados:** Aunque no son librerías en el sentido tradicional, el programa incluye una serie de predicados personalizados, como **read\_file/2**, **write\_file/2**, **eliminar\_archivo/1**, **crear\_archivo/2** y **cambiar\_nombre\_archivo/2**, que se utilizan para la lectura y escritura de archivos, así como para realizar operaciones de archivos, como eliminar, crear y cambiar nombres.
5. **Librerías de manipulación de cadenas:** El programa utiliza una serie de predicados para la manipulación de cadenas, como **split\_string/4**, **downcase\_atom/2**, **atomic\_list\_concat/2** y **string\_concat/2**, que son parte de las librerías de manipulación de cadenas. Estos predicados permiten procesar y dividir las líneas de texto de los archivos y realizar operaciones de manipulación de cadenas para transformar y concatenar datos según sea necesario.

Estas librerías juegan un papel fundamental en la funcionalidad del programa, permitiendo la interacción con los usuarios, la carga y manipulación de datos desde archivos y la realización de operaciones de archivos. En conjunto, proporcionan un entorno efectivo para la gestión de proyectos y tareas, así como la generación de estadísticas y recomendaciones.

## Análisis de resultados

### Opciones Administrativas



|                      |      |
|----------------------|------|
| Gestión de personas  | 100% |
| Gestión de proyectos | 100% |
| Gestión de tareas    | 100% |
| Buscar tareas libres | 100% |
| Recomendar persona   | 35%  |
| Asignar a tarea      | 100% |
| Cerrar tarea         | 100% |
| Estadísticas         | 35%  |

Aunque estadísticas y recomendar se trabajaron, no se pudieron realizar sus implementaciones al programa debido a que se tenían que hacer ciertas modificaciones que llevaban más tiempo de lo esperado y esto terminaría afectando la entrega del proyecto.

A su vez, se dio mayor prioridad a sacar de manera correcta y funcional los 3 gestores para el sistema de gestión de tareas para que el proyecto tuviera un resultado aceptable, de ahí que esas 2 funcionalidades quedaran rezagadas.

## Bitácora y referencias

Link del repositorio github: [Duan-Espinoza/P3\\_Gestion\\_Tareas: Repositorio que contiene el proyecto 3 de lenguajes \(github.com\)](https://github.com/Duan-Espinoza/P3_Gestion_Tareas)

Cómo Programar En Prolog: Tutorial Para Principiantes. (2023, marzo 10). *Revistasuprema.com*.  
<https://revistasuprema.com/como-programar-en-prolog-tutorial-para-principiantes/>

¿Cómo puedo guardar cadenas de texto a una variable en Prolog? (s/f). Stack Overflow en español. Recuperado el 31 de octubre de 2023, de  
<https://es.stackoverflow.com/questions/163887/c%C3%B3mo-puedo-guardar-cadenas-de-texto-a-una-variable-en-prolog>

Conversacional, 1. Prolog es un. (s/f). *Guía de Uso Básico de Prolog*. Uma.es. Recuperado el 31 de octubre de 2023, de <http://lcc.uma.es/~pacog/apuntes/pd/guia.pdf>

De fichero, 1. Escritura de Términos 2. Lectura de Términos 3. Escritura y. Lectura de Caracteres 4. Escritura en Fichero 5. Lectura. (s/f). 5. *Entrada y Salida en PROLOG*. Uva.es. Recuperado el 31 de octubre de 2023, de  
<https://www.infor.uva.es/~calonso/IAI/PracticasProlog/Tema5/Tema%205.%20Entrada%20y%20Salida%20en%20PROLOG.pdf>

*Guía Prolog*. (s/f). Uchile.cl. Recuperado el 31 de octubre de 2023, de  
<https://users.dcc.uchile.cl/~peortega/guiaprolog/leccion3.html>

La programación lógica deductiva, •. Prolog es un Lenguaje de Programación Basado en. (s/f). *Introducción a Prolog Conceptos Básicos (1/2)*. Buap.mx. Recuperado el 31 de octubre de 2023, de [https://www.cs.buap.mx/~iolmos/ia/Sesion4\\_Prolog.pdf](https://www.cs.buap.mx/~iolmos/ia/Sesion4_Prolog.pdf)

*manual*. (s/f). Swi-prolog.org. Recuperado el 31 de octubre de 2023, de [https://www.swi-prolog.org/pldoc/doc\\_for?object=manual](https://www.swi-prolog.org/pldoc/doc_for?object=manual)

*Prolog - COMO ABRO UN ARCHIVO EN PROLOG?* (s/f). Lawebdelprogramador.com. Recuperado el 31 de octubre de 2023, de <https://www.lawebdelprogramador.com/foros/Prolog/528147-COMO-ABRO-UN-ARCHIVO-EN-PROLOG.html>

*Supertutorial de Prolog: aprende Prolog en español*. (2020, agosto 29). Adrianistan.eu.  
<https://blog.adrianistan.eu/supertutorial-prolog>

Wikipedia contributors. (s/f). *Prolog*. Wikipedia, The Free Encyclopedia.  
<https://es.wikipedia.org/w/index.php?title=Prolog&oldid=148164496>