

Universidad Rafael Landívar  
Facultad de Ingeniería  
Licenciatura en Ingeniería en Informática y Sistemas

## **Documentación - Gestor de Actividades Universitarias**

Curso: Programación Avanzada  
Sección: 03

Integrantes:

Mario Rolando Matheu Duque – 1567324

Cindy Nineth Estrada Estrada – 1569225

María Belén Mendoza García - 1517225

Luna Jimena Calderón Escobar - 1638725

Guatemala, agosto 2025

## ÍNDICE

INTRODUCCIÓN: .....	3
DESARROLLO DEL PROYECTO: .....	3
ESTRUCTURA DEL PROYECTO: .....	6

## INTRODUCCIÓN:

El presente documento posee distintos recursos gráficos que buscan ayudar a entender mejor el proyecto. Incluye el pseudocódigo del sistema, imágenes de la interfaz y del proceso de realización del proyecto y ejemplos de cómo usarlo, mostrando las funciones principales. Su objetivo es ofrecer apoyo visual y técnico para comprender el proyecto 'Gestor de Actividades Universitarias'.

El proyecto se basó en lo siguiente:

Crea un sistema para registrar, clasificar y consultar actividades universitarias de un estudiante (clases, tareas, exámenes, reuniones, eventos personales).

Debiendo permitir:

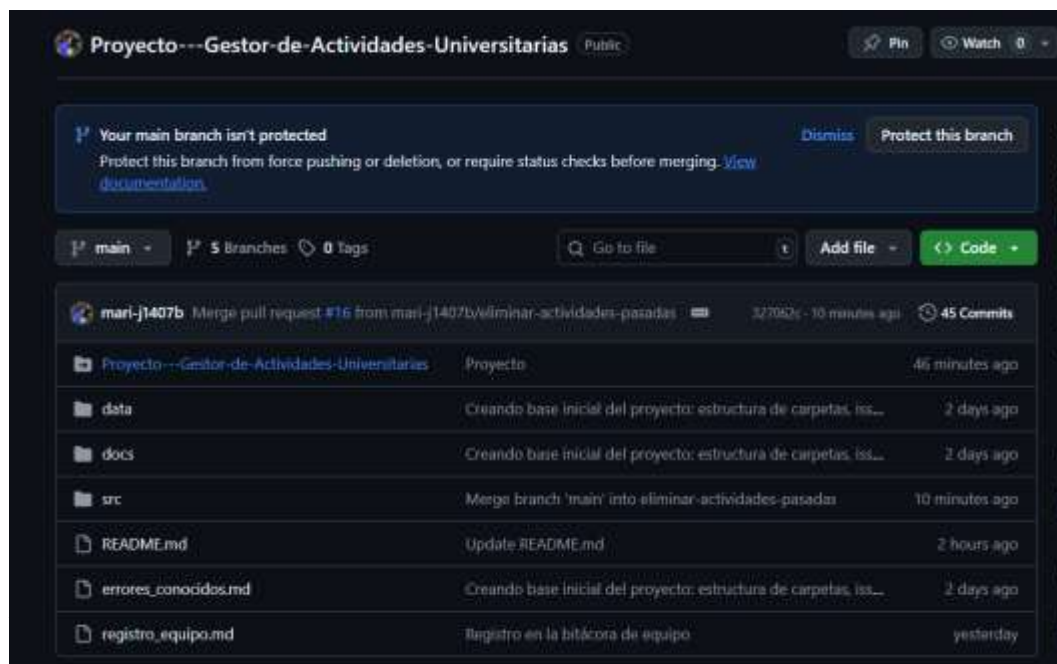
- Registrar y listar nuevas actividades indicando nombre, fecha (día, mes, año) y prioridad.
- Buscar y filtrar actividades por palabras clave.
- Mostrar todas las actividades listadas por orden.
- Eliminar actividades previamente registradas.
- Salir del programa.

## DESARROLLO DEL PROYECTO:

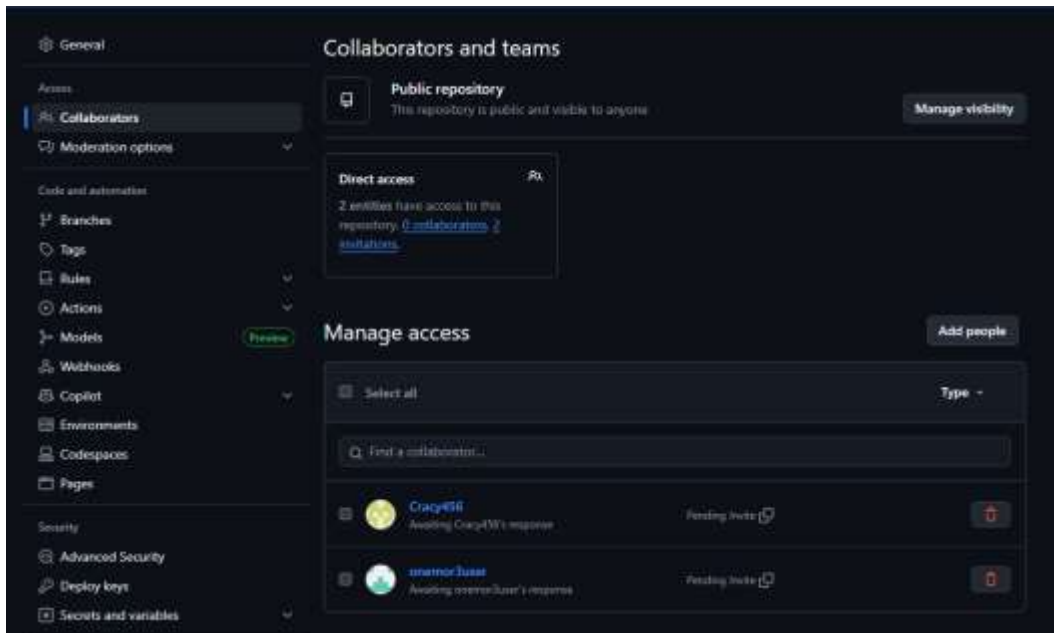
### Creando el repositorio “Gestor de Actividades Universitarias”

Se creó un repositorio en GitHub.com llamado 'Gestor de Actividades Universitarias'. Este repositorio centralizó los archivos del proyecto y permitió el trabajo colaborativo mediante ramas, commits y pull requests.

Repositorio en GitHub: <https://github.com/mari-j1407b/Proyecto---Gestor-de-Actividades-Universitarias.git>



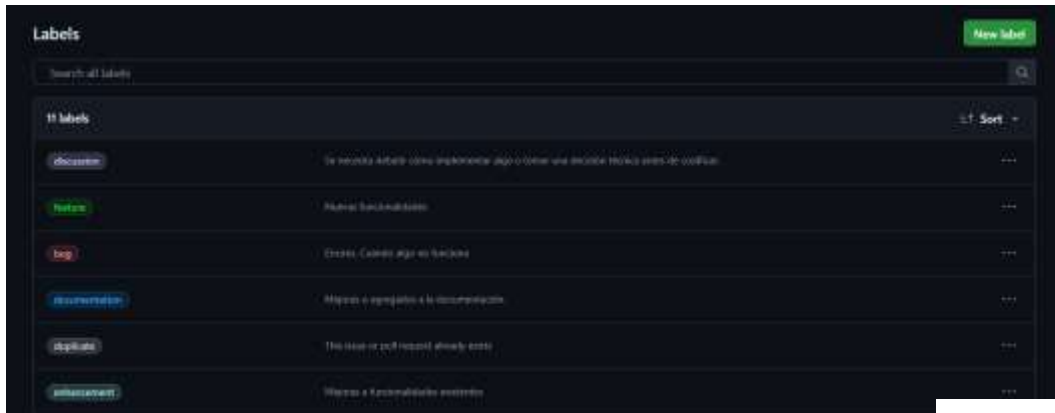
Cada miembro del equipo fue añadido como colaborador en el repositorio para que todos pudieron aportar código, documentación y pruebas.



## Creando “Issues” y sus respectivas etiquetas

Se crearon diferentes 'Issues' para registrar las tareas y mejoras pendientes del proyecto. A estas se les asignaron etiquetas descriptivas para clasificar su prioridad, estado o tipo de tarea (ej. documentación, corrección, mejora).

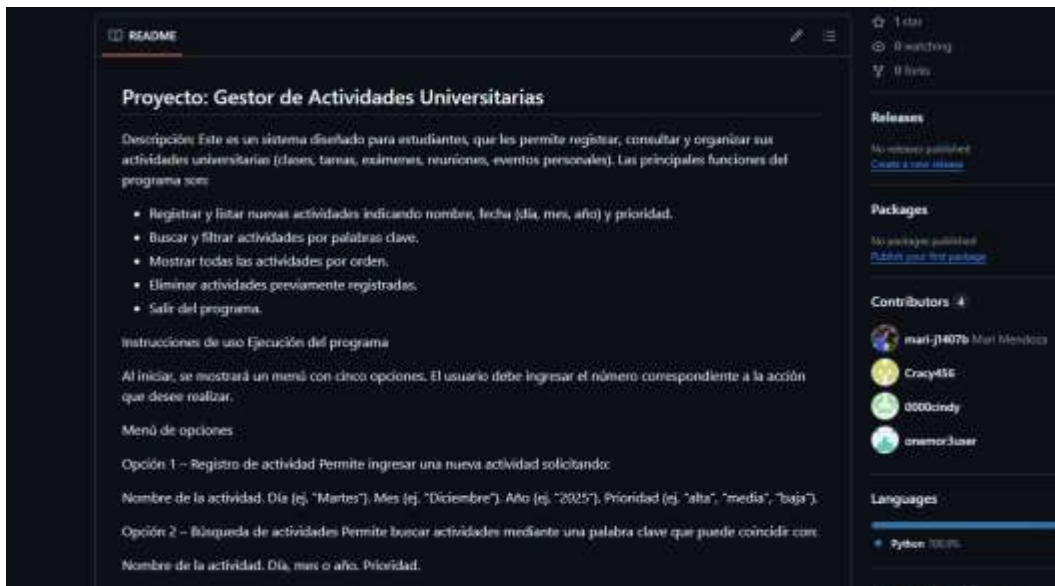




Etiquetas

### Creando el archivo README.md:

Se redactó un archivo README.md para explicar brevemente el propósito del proyecto, cómo usarlo y quiénes eran los integrantes. Este archivo sirvió como punto de partida para cualquiera que accediera al repositorio.



## ESTRUCTURA DEL PROYECTO:

El programa cuenta con un menú principal que ofrece cinco opciones. El usuario debe elegir el número de la opción que desee ejecutar.

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

○ Bienvenido al menu de opciones del gestor de actividades
Opcion 1: Registro de Actividad por fecha y prioridad
Opcion 2: Buscar por palabra clave
Opcion 3: Mostrar todas las actividades por orden
Opcion 4: eliminar actividad
Opcion 5: salir del programa
Elija una opcion valida del menu:
```

### Opción 1 – Registrar una actividad

Permite ingresar una nueva actividad solicitando: nombre de la actividad, día, mes, año y nivel de prioridad. De esta forma, se guarda en la lista de actividades.

Ejemplo registrando actividades en el programa:

```
○ Elijo la opcion 1, registrar una actividad
Ingrese la actividad: Tareas
Ingrese el dia ejemplo (Martes): Lunes
Ingrese el mes ejemplo (Diciembre): Agosto
Ingrese el año ejemplo (2025): 2026
Ingrese la prioridad de la actividad: Entrega rápida
Presione enter para continuar...
```

### Opción 2 – Buscar actividades por palabra clave

Permite localizar actividades filtrando por una palabra clave. La búsqueda puede coincidir con el nombre de la actividad, el día, mes, año o prioridad.

Ejemplo buscando actividades por palabra clave en el programa:

```
○ Se eligio la opcion 2, buscar por palabra clave
Ingrese la palabra la clave: Tarea
Nombre de la actividad tareas dia lunes Mes agosto año 2026 con prioridad entrega rápida
Presione enter para continuar...
```

### Opción 3 – Listar o mostrar actividades

Muestra las actividades registradas en el sistema, ordenadas según el criterio de búsqueda (por mes, año, día o categoría). Facilita una visión general de todas las actividades guardadas.

Ejemplo buscando actividades para que sean listadas en el programa:

```
○ Se eligió la opción 3. Buscar actividades por mes, año, día o categoría
Ingrese el año a buscar: 2026
Nombre de la actividad tarea día lunes mes agosto año 2026 con prioridad entrega rápida
Presione enter para continuar...
```

### Opción 4 – Eliminar actividad

Esta opción lista nuevamente todas las actividades registradas para que posteriormente según la elección del usuario permita eliminar una seleccionada. Para ello, se debe ingresar el día, mes y año correspondientes a la actividad que se desea borrar.

```
Se eligió la opción 4, Eliminar actividad
estas son todas las actividades registradas, ingrese la que quiere eliminar
Nombre de la actividad tareas día lunes Mes agosto año 2026 con prioridad entrega rápida
Ingrese el día a buscar: lunes
Ingrese le mes a buscar: agosto
Ingrese el año a buscar: 2026
Se ha eliminado la actividad
Presione enter para continuar ...
```

### Opción 5 – Salir del programa

Finaliza la ejecución del sistema y cierra el programa.

```
○ se eligio la opcion 5, se cerrara el programa
█
```

## Pseudocódigo del proyecto:

A continuación, se muestra el pseudocódigo del programa, el cual refleja la estructura principal de su funcionamiento.

```
INICIO
# DEFINICIÓN DE LA CLASE PRINCIPAL

Clase Actividades:

    # Esta clase es como una plantilla para crear actividades.

    # Cada actividad tendrá estos 5 datos básicos:

    Constructor(actividad, día, mes, año, prioridad):

        actividad = actividad # El nombre de la tarea (ej: "presentación de historia")

        día = día             # Día de la semana en texto (ej: "Miércoles")

        mes = mes             # Mes del año en texto (ej: "Agosto")

        año = año             # Año numérico (ej: 2023)

        prioridad = prioridad # Nivel de urgencia (ej: "urgente", "normal")

    # Cómo se mostrará cada actividad cuando necesitemos verla

    Método Mostrar():

        Retornar texto formateado con todos los datos

        # Este método hace que al imprimir una actividad se vea ordenado y claro

# LISTA MAESTRA DE ACTIVIDADES

Lista actividades_universitarias = []

# Esta lista servirá para guardar TODAS las actividades que cree el usuario

# -----

# PROGRAMA PRINCIPAL

# -----

Bucle de ejecución continua # Este es el menú que siempre vuelve después de cada acción

    Limpiar pantalla # Para que la pantalla no se llene de texto viejo

    # Mostramos las opciones como en un cajero automático

    Mostrar menú principal con 5 opciones numeradas

    Pedir opción al usuario # Aquí el usuario escribe el número de lo que quiere hacer
```



```
# OPCIÓN 1: AGREGAR

Si opción es 1:

    # Primero pedimos todos los datos necesarios

    Pedir nombre de actividad (en minúsculas para evitar problemas)

    Pedir día (solo texto, no número)

    Pedir mes (texto completo)

    Pedir año (número)

    Pedir prioridad (texto libre)


    # Creamos la actividad nueva con los datos

    nueva_actividad = Nueva Actividad(con todos los datos ingresados)


    # La guardamos en la lista

    actividades_universitarias.agregar(nueva_actividad)

    # Ahora ya queda guardada en el sistema


#OPCIÓN 2: BUSCAR

Sino, si opción es 2:

    # El usuario podrá buscar por palabra clave (día, mes, año, actividad o prioridad)

    Pedir palabra_clave (convertida a minúsculas)


    # Revisamos TODAS las actividades una por una

    Para cada actividad en la lista:

        # Buscamos si la palabra aparece en cualquier campo

        Si la palabra está en el nombre, día, mes, año o prioridad:

            Mostramos esa actividad completa

    # Si no encuentra nada, no muestra nada
```

```
# OPCIÓN 3: Mostrar todas las actividades por orden
Sino, si opción es 3:

    Pedir mes_o_año para filtrar (o puede dejar vacío)

    # Se devolverán solo resultados que coincidan con el filtro
    Para cada actividad:
        Si coincide con el filtro (o si no hay filtro):
            Mostrar actividad

    # Así el usuario puede ver todo o solo partes

# ----- OPCIÓN 4: ELIMINAR -----
Sino, si opción es 4:

    # Primero mostramos todo para que el usuario vea qué hay
    Mostrar "Tus actividades actuales:"

    Para cada actividad:
        Mostrar actividad

    # Pedimos datos EXACTOS para evitar borrar lo equivocado
    Pedir día exacto, mes exacto y año exacto

    # Buscamos la actividad que coincida con los 3 datos
    Si encontramos EXACTAMENTE esa combinación:
        Eliminamos esa actividad de la lista
        Mostrar mensaje de confirmación
```

```
# OPCIÓN 5: SALIR
Sino, si opción es 5:
    Mostrar mensaje de despedida
    Terminar programa # Aquí se cierra todo. Hace que el usuario salga del programa

# OPCIÓN INVÁLIDA
Sino, si opcion es 5:
    Limpiar pantalla
    Mostrar "Se eligió la opción 5, se cerrará el programa"
    Leer entrada_usuario # <<< Esto captura el input vacío
    Terminar bucle # <<< El break para que deje de repetirse el ciclo

Sino:
    Limpiar pantalla
    Mostrar "Opción no válida"

Leer "Presione enter para continuar..."
FIN.
```