



## Laboratorio I

Proyecto integrador

|                          |   |
|--------------------------|---|
| Profesores               | Alejandro Behringer<br>Paula Ithurralde<br>Jonathan Pécora<br>Miguel Silva                                  |
| Alumnos                  | Juan Martin Biedma<br>Santiago Borrajo<br>Gonzalo LLanos  |
| Comisión                 | 1604  |
| Turno                    | Noche   |
| Repositorio del proyecto | <a href="https://github.com/Zasa-san/inspt-laboratorio1">https://github.com/Zasa-san/inspt-laboratorio1</a> |

## Tabla de contenido

|  |    |
|--|----|
| Objetivos.....                               | 2  |
| Alcance del proyecto y público objetivo..... | 2  |
| Características del software.....            | 2  |
| Herramientas.....                            | 3  |
| Perfil de configuración de PSEINT.....       | 3  |
| Menú principal.....                          | 4  |
| Módulo Contactos.....                        | 5  |
| Módulo Calendario.....                       | 10 |
| Módulo Pomodoro.....                         | 14 |

## Objetivos

Desarrollar un software que integre diversas funcionalidades para resolver diferentes necesidades que surgen en entornos de trabajo de oficina, aprovechando al máximo las características disponibles del lenguaje utilizado. Este proyecto involucrará la creación de dos bases de código separadas que contendrán la misma funcionalidad básica: una en PSeInt y otra en C. Aunque ambos códigos ofrecerán la misma funcionalidad, cada uno aprovechará las características propias del lenguaje para implementarlas. Cada módulo será construido por cada uno de los participantes del grupo en colaboración a los demás.

## Alcance del proyecto y público objetivo

Este desarrollo buscará facilitar la centralización de herramientas en una sola plataforma, ahorrando tiempo, mejorando la eficiencia y promoviendo hábitos como la organización y la concentración. Podrá ser utilizado por cualquier persona que se desarrolle en un ámbito de oficina. Es ideal para personas que deben tener interacción con otras personas para la realización de proyectos, los cuales necesiten de reuniones periódicas, cumplir objetivos, etc. También es recomendable para personas que se desempeñen como asistentes o secretarios.

## Características del software

Las características propuestas son una idea general de las soluciones que el software debe ofrecer, pudiendo cambiar a medida que el proyecto se desarrolla para cumplir mejor los objetivos.

1. Calendario con gestión de tareas: Un calendario que muestre el mes completo. Permitirá elegir año, mes, y navegar día a día para visualizar tareas existentes, crear nuevas o editarlas. Este permite gestionar eventos, llevando una agenda diaria de tareas realizadas y por realizar.
2. Agenda: Una agenda de contactos que permita crear, editar y eliminar contactos con una serie de campos básicos. Esta agenda tendrá funcionalidad de búsqueda por campo. Esta facilita la organización y acceso rápido a información de contacto.
3. Pomodoro<sup>1</sup>: Un sistema configurable de módulos pomodoro con un temporizador en pantalla e indicador visual para periodo de trabajo y periodo de descanso. Ayuda a mantener el enfoque en el trabajo con sesiones cronometradas.
4. Menú: Un menú principal para acceder a los módulos.

De ser posible se buscará integrar los módulos de agenda y contactos, por ejemplo permitiendo crear eventos y elegir contactos de la agenda para agregarlos como invitados.

El software contará con datos iniciales incorporados al código para ilustrar su funcionalidad, durante el tiempo de ejecución guardará en memoria los cambios que haga el usuario.

---

<sup>1</sup> La **técnica Pomodoro** es un método para mejorar la administración del tiempo dedicado a una actividad. [Ver más](#).

## Herramientas

Se utilizará git para el control de cambios de código y GitHub para obtener un repositorio remoto compartido para el código y otros insumos del proyecto. Al usar GitHub como repositorio también se podrá aprovechar las herramientas que la plataforma ofrece para la gestión de proyectos.

### Como contribuir

Se debe crear un pull request a main para agregar o modificar código. Solo se verificará que la rama creada esté al día con los últimos cambios que main tenga para poder combinarla.

### Estilo de código

Se debe anteponer a cada algoritmo o subalgoritmo una nomenclatura que refiera al módulo, por ejemplo:

```
SubAlgoritmo pom_pomodoro(argumentos)  
/// algoritmo  
FinSubAlgoritmo
```

Cada integrante del proyecto podrá decidir qué herramientas usar en su entorno de desarrollo para facilitar su tarea. Sin embargo se compartirá una misma configuración de psint y se acordará una fuente para el texto que se mostrará por consola para lograr una visualización consistente para todos los usuarios.

En todo caso se buscará la forma más sencilla que facilite mantener el código limpio, legible y uniforme.

## Perfil de configuración de PSEINT

Se utilizará un perfil específico y común a todos los programadores:

```
desc=inspt laboratorio 1
version=20230211
force_init_vars=0
force_define_vars=0
force_semicolon=0
allow_concatenation=1
enable_string_functions=1
word_operators=1
base_zero_arrays=1
allow_dinamyc_dimensions=1
allow_resize_arrays=1
overload_equal=1
enable_user_functions=1
lazy_syntax=1
coloquial_conditions=1
integer_only_switch=0
deduce_negative_for_step=1
use_nassi_shneiderman=1
use_alternative_io_shapes=0
allow_accents=1
prefer_algoritmo=1
prefer_funcion=1
allow_repeat_while=1
allow_for_each=1
prefer_repeat_while=1
protect_for_counter=0
```

## Menú principal

El menú principal contiene una presentación interactiva que se visualiza al ingresar la primera vez al programa. Luego, al navegar y volver al menú principal no volverá a aparecer.



### **Funciones, subProcesos y subAlgoritmos:**

Algoritmo funcionalidades Oficina: Este es el Algoritmo principal del programa a partir del cual se estructura el mismo.

SubProceso main Pantalla Inicio: Este subproceso muestra una animación de inicio de programa. La animación consiste en una serie de pantallas que se muestran y borran en intervalos de tiempo específicos simulando la realización de una presentación interactiva.

Funcion opc <- main menu: Esta función muestra el menú principal del programa, donde el usuario puede seleccionar entre las diferentes funcionalidades disponibles: gestión de contactos, calendario y temporizador Pomodoro. La función valida la entrada del usuario para asegurarse de que sea una opción válida y devuelve la opción seleccionada.

## **Módulo Contactos**

Este módulo tiene la función de gestión de contactos que permite al usuario agregar, buscar, ver, editar y eliminar contactos en una agenda. Proporciona varias funcionalidades para gestionar la información de los contactos de manera eficiente. En él se puede almacenar nombre, apellido, teléfono, email y dirección de cada persona diferenciadas por un ID único.

### **Funciones, subProcesos y subAlgoritmos:**

SubAlgoritmo con ContactosMain: Este subalgoritmo controla el flujo del módulo de contactos. Permite al usuario navegar por el menú principal de contactos, donde puede elegir entre ver contactos por apellido, buscar contactos por nombre, agregar o cambiar contactos, y listar todos los contactos. También maneja la reordenación de contactos después de ciertas operaciones.

```

-----
/  _  /  _  \  /  |  /  _  /  _  \  /  _  \  /  _  \
//    //    /  |  //    //    //    //    //    \  \
/_    /_    /  |  /_    /_    //    //    //    /_
\  \  \  \  /  |  \  \  \  \  \  \  \  \  \  \  \
-----

```

\*Agenda de contactos\*

Actualmente tiene 26/1000 contactos guardados

Opciones de navegación:

- 1 - Ver por apellido
- 2 - Buscar por nombre o apellido
- 3 - Agregar nuevo contacto
- 4 - Ver listado completo
- 0 - Para salir de la agenda

>

Función con agregarOCambiarContacto: Esta función permite al usuario agregar un nuevo contacto o cambiar la información de un contacto existente. Solicita al usuario que ingrese los detalles del contacto, como nombre, apellido, teléfono, email y dirección. La función valida la entrada del usuario y guarda los cambios en la agenda.

```

-----
Creación de nuevo contacto
-----

```

- 0 - Salir sin guardar
- 1 - Guardar

Nombre:

Apellido:

Telefono:

Email:

Dirección:

Nombre: >

Función con buscarPorNombre: Esta función permite al usuario buscar contactos por nombre o apellido. Muestra una lista de contactos que coinciden con la búsqueda y permite al usuario elegir un contacto para editar o eliminar.



```
-----  
Busqueda por nombre  
-----
```

```
Escriba el nombre y/o apellido a buscar  
Para salir, escriba 0
```

```
>
```

Función con verPorApellido: Esta función permite al usuario ver todos los contactos ordenados por apellido. El usuario puede seleccionar una letra del alfabeto para ver los contactos cuyos apellidos comienzan con esa letra. La función también permite al usuario editar o eliminar un contacto seleccionado.

```
-----  
Vista por apellido  
-----
```

```
Eliga la letra para ver todos los contactos agendados  
Para salir, escriba 0
```

```
>
```

```
-----  
Vista por apellido  
-----
```

```
1) Ramírez Gabriela  
2) Rivas Carmen  
3) Rodríguez Sofía  
Use las teclas numéricas para elegir un contacto  
o 0 para salir
```

```
>
```

Función con listarTodos: Esta función lista todos los contactos en la agenda, paginando los resultados de a 10 si hubiera más que ese número. Permite al usuario navegar entre las páginas de contactos y seleccionar un contacto para editar o eliminar.

-----  
Listado completo  
-----

Página 1

- 0 - Salir
- 1 - Página anterior
- 2 - Página siguiente
- 3 - Elegir contacto de esta página

- 
- 1) Biedm Juan - (ID:26)
  - 2) Bravo Sandra - (ID:14)
  - 3) Castro Valeria - (ID:12)
  - 4) Comanche Sandro - (ID:25)
  - 5) Cruz Silvia - (ID:20)
  - 6) Fernández Tomás - (ID:23)
  - 7) Fernández Mónica - (ID:10)
  - 8) Fernández Ricardo - (ID:5)
  - 9) García Ana - (ID:0)
  - 10) González Jorge - (ID:19)

> |

Función con elegirEnPagina: Esta función permite al usuario elegir un contacto de una página específica de la lista de contactos. Muestra los detalles del contacto seleccionado y ofrece opciones para editar o eliminar el contacto.

-----  
Listado completo  
-----

Use las teclas numéricas para elegir un contacto  
o 0 para salir

- 11) González Renata
- 12) Gutiérrez Emilio
- 13) Gómez Pedro
- 14) Jiménez Álvaro
- 15) Martínez Luis
- 16) Martínez Gustavo
- 17) Morales Tomás
- 18) Pérez Julia
- 19) Ramírez Gabriela
- 20) Rivas Carmen

> |

Función con eliminarContacto: Esta función permite al usuario eliminar un contacto de la agenda. Solicita confirmación antes de borrar el contacto y reordena la agenda después de la eliminación.

```
-----  
Nombre: Pedro  
Apellido: Gómez  
Telefono: 1154782243  
Email: pedro.gomez@example.com  
Dirección: Calle San Juan 300, Buenos Aires  
-----
```

```
1 - Editar este contacto  
2 - Borrar este contacto  
0 - Volver  
> |
```

```
-----  
Borrar el contacto  
-----
```

```
Nombre: Álvaro  
Apellido: Jiménez  
Telefono: 1154782253  
Email:  
Dirección: Av. Belgrano 1200, Buenos Aires  
-----
```

```
¿Desea borrar este contacto?  
1 - Borrar  
0 - Salir sin borrar  
>
```

SubAlgoritmo con ordenarPorApellido: Este subalgoritmo ordena los contactos en la agenda por apellido. Utiliza un algoritmo de ordenación para reorganizar los contactos y actualiza el índice de apellidos para facilitar la búsqueda.

SubAlgoritmo con generarResultados: Este subalgoritmo genera una lista de resultados de búsqueda basados en el nombre o apellido ingresado por el usuario. Calcula un valor de ponderación para determinar la relevancia de cada contacto en los resultados.

Función con obtenerIndice: Esta función devuelve el índice correspondiente a una letra del alfabeto. Es utilizada para organizar y buscar contactos por la primera letra de su apellido.

Función con hacerLinea: Esta función imprime una línea de separación en la pantalla, utilizada para mejorar la legibilidad de la interfaz de usuario.

Función con bienvenida: Esta función imprime un mensaje de bienvenida que se visualiza cuando el usuario accede al módulo de contactos.

SubAlgoritmo con\_MostrarContacto: Este subalgoritmo muestra los detalles de un contacto específico en la pantalla.

Función con\_tituloAgregarOCambiar: Esta función imprime el título de la sección de agregar o cambiar contacto.

Función con\_tituloVerPorApellido: Esta función imprime el título de la sección de ver contactos por apellido.

Función con\_tituloBorrarContacto: Esta función imprime el título de la sección de borrar contacto.

Función con\_tituloListarTodos: Esta función imprime el título de la sección de listar todos los contactos.

Función con\_tituloBuscarPorNombre: Esta función imprime el título de la sección de buscar contactos por nombre.

Función con\_MenuPrincipal: Esta función muestra el menú principal del módulo de contactos y solicita al usuario que elija una opción. Valida la entrada del usuario y devuelve la opción seleccionada.

SubAlgoritmo con\_inicializarIndice: Este subalgoritmo inicializa el índice de apellidos.

Función con\_longitudArreglo: Esta función calcula y devuelve la longitud actual de la agenda, es decir, el número de contactos almacenados.

## Módulo Calendario

Este módulo permite a los usuarios gestionar tareas en un calendario mensual. Los usuarios pueden agregar tareas, marcar tareas como realizadas y asignar participantes a las tareas desde una lista de contactos. El programa está diseñado para ser interactivo, solicitando entradas del usuario para navegar por el calendario y gestionar las tareas del día seleccionado.

### **Funciones, subProcesos y subAlgoritmos:**

SubAlgoritmo Calendario: Este subalgoritmo controla el flujo del módulo. Inicializa las matrices, configura tareas predefinidas para poblar este Demo, maneja la interacción del usuario para seleccionar el año y el mes, renderizar el calendario y gestionar las tareas del día seleccionado.

Bienvenido al calendario. Presiona una tecla para comenzar

 $\gamma$  $\gt |$ 

Subalgoritmo AgregarTarea: Este subalgoritmo se encarga de la carga de una nueva tarea a la matriz Tareas y actualiza el contador de tareas.

Introduce la nueva tarea:

>

Subalgoritmo CalcularDiaInicio: Este subalgoritmo calcula el primer día del mes utilizando la fórmula de Zeller<sup>2</sup>. Devuelve el día de la semana en que comienza el mes para luego ser usado en la renderización del calendario.

Subalgoritmo CalcularDiasMes: Este subalgoritmo determina el número de días de un mes dado y asigna el nombre del mes correspondiente. También maneja los años bisiestos para febrero.

Subalgoritmo cal\_RenderCalendario: Este subalgoritmo renderiza el calendario del mes seleccionado, mostrando los días y resaltando aquellos que ya tienen tareas asignadas.

Mes seleccionado: Noviembre 2024

Calendario de Noviembre 2024

| L    | M     | X    | J  | V     | S    | D  |
|------|-------|------|----|-------|------|----|
|      |       |      |    | 1     | 2    | 3  |
| 4    | [ 5 ] | 6    | 7  | [ 8 ] | 9    | 10 |
| 11   | 12    | 13   | 14 | [15]  | 16   | 17 |
| 18   | 19    | [20] | 21 | 22    | 23   | 24 |
| [25] | 26    | 27   | 28 | 29    | [30] |    |

Introduce el día para ver o agregar tareas (0 para volver a seleccionar el mes):

> |

Subalgoritmo cal\_MenuAccionesDia: Este subalgoritmo maneja las acciones del día seleccionado, permitiendo al usuario agregar tareas, marcar tareas como realizadas y asignar participantes a las tareas desde la lista de contactos del módulo de contactos.

---

<sup>2</sup> Ver más: [https://es.wikipedia.org/wiki/Congruencia\\_de\\_Zeller](https://es.wikipedia.org/wiki/Congruencia_de_Zeller)

Fecha seleccionada: 15 de Noviembre de 2024

Tareas para el día

1. Status de objetivos

Selecciona una opción:

1. Agregar una nueva tarea

2. Volver al mes

3. Marcar tarea como realizada

4. Asignar participantes a una tarea

> |

Selecciona el número de la tarea que deseas marcar como realizada (0 para volver):

>

Fecha seleccionada: 15 de Noviembre de 2024

Tareas para el día

1. Status de objetivos --Realizada--

2. test 1

Selecciona una opción:

1. Agregar una nueva tarea

2. Volver al mes

3. Marcar tarea como realizada

4. Asignar participantes a una tarea

>

Fecha seleccionada: 15 de Noviembre de 2024

Tareas para el día

1. Status de objetivos --Realizada--

2. test 1

Selecciona el número de la tarea a la que deseas asignar participantes (0 para volver):

> |

Función cal\_listaDeContactos: Esta función lista los contactos disponibles y permite al usuario seleccionar uno a partir de la navegación por las páginas de contactos divididas en 10 contactos por cada una y devuelve el ID único del contacto seleccionado.

```
-----  
Listado completo  
-----
```

```
Pagina 1
```

```
0 - Salir  
1 - Página anterior  
2 - Página siguiente  
3 - Elegir contacto de esta página  
-----
```

```
1) Biedm Juan  
2) Bravo Sandra  
3) Castro Valeria  
4) Comanche Sandro  
5) Cruz Silvia  
6) Fernández Tomás  
7) Fernández Mónica  
8) Fernández Ricardo  
9) García Ana  
10) González Jorge  
> |
```

Función cal\_elegirContacto: Esta función permite al usuario elegir un contacto de la lista mostrada, verifica que la selección sea válida y devuelve el ID del contacto seleccionado.

```
Fecha seleccionada: 15 de Noviembre de 2024  
Tareas para el día
```

```
1. Status de objetivos --Realizada--  
2. test 1 - Juan Biedm
```

```
Selecciona una opción:
```

```
1. Agregar una nueva tarea  
2. Volver al mes  
3. Marcar tarea como realizada  
4. Asignar participantes a una tarea  
>
```

## Módulo Pomodoro

Este módulo implementa un temporizador Pomodoro, una técnica de gestión del tiempo que alterna períodos de trabajo (acción) con períodos de descanso (pausa). El objetivo es



mejorar la productividad y la concentración de las personas en el trabajo. Este permite al usuario establecer la duración de los períodos de acción y pausa, y luego muestra una pantalla con un temporizador que cuenta regresivamente. Durante su ejecución, el programa alterna entre los períodos de acción y pausa, mostrando la hora de inicio y la hora estimada de finalización. Además, incluye funciones para construir y mostrar los dígitos del temporizador, así como para calcular y mostrar las horas de inicio y cierre.

### **Funciones, subProcesos y subAlgoritmos:**

#### **SubAlgoritmo Pomodoro:**

Este subalgoritmo es el principal del módulo y controla el flujo del programa. Solicita al usuario que ingrese los tiempos de acción y pausa, construye la pantalla del temporizador y gestiona la alternancia entre los bloques de acción y pausa. También permite al usuario cambiar los valores de los tiempos al finalizar la ronda si así lo desea.

```

>>=====<<
||                                     ||
||               _____           ||
||      _____/_____           ||
||      _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ ||
||      _ // // // // // // // // // // // // // // // // // // // // // ||
||      _ . _ ^ _ // // // // // ^ _ , / \ _ // // \ _ // ||
||      _/_/ ||
||                                     ||
>>=====<<

```

PRESIONE UNA TECLA PARA COMENZAR...

Función pom\_Establecer\_Min: Esta función solicita al usuario que ingrese un valor en minutos y valida que esté dentro de un rango específico (entre 1 y 59 minutos) y lo devuelve. Si el valor ingresado no es válido, la función muestra un mensaje de error y solicita nuevamente el valor hasta que se ingrese un valor correcto.

```

Elija el tiempo de acción del Pomodoro en minutos(entre 1 y 59)
> 2
Elija el tiempo de pausa del Pomodoro en minutos(entre 1 y 59)
> 3

```

SubAlgoritmo pom\_Construir\_Pantalla: Este subalgoritmo construye y muestra la pantalla del temporizador. Inicializa las matrices necesarias para representar los dígitos del cronómetro y los subtítulos, ejecuta el cálculo de hora de inicio y alterna entre los períodos de acción y pausa. Durante cada período, muestra el tiempo restante en la pantalla y

actualiza el temporizador en intervalos regulares. En este se tomó en cuenta el tiempo de construcción de la pantalla para poder cumplimentar el segundo real, dando como resultado un cambio de pantalla cada 740 milisegundos.

----- Hora de inicio 01:58 -- Hora de fin 02:13 -----

```
>>=====<<
||                                     ||
||      _____/_____          ||
||      _ _ _ \ _ _ \ _ _ \ _ _ \ / _ _ \ _ _ \ _ _ \ ||
||      _ // // // // // // // // // // // // // // // ||
||      _ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ ||
||      // ||                                     ||
>>=====<<
```

```
_____
_ | _____ (-) _____
_ // | _ _ \ _ _ \ _ _ \ _ _ \
_ _ _ // // // _ _ // // // //
// | _ _ \ _ _ \ // _ _ // // //
```

|           |         |               |            |
|-----------|---------|---------------|------------|
| .d8888b.  | d888    | 8888888888    | d8888      |
| d88P Y88b | d8888   | 888           | d8P888     |
| 888 888   | 888     | d8b 888       | d8P 888    |
| 888 888   | 888     | Y8P 8888888b. | d8P 888    |
| 888 888   | 888     | °Y88b         | d88 888    |
| 888 888   | 888     | 888           | 8888888888 |
| Y88b d88P | 888     | Y8P Y88b d88P | 888        |
| °Y8888P°  | 8888888 | °Y8888P°      | 888        |

----- Hora de inicio 01:58 -- Hora de fin 02:12 -----

```
>>=====
||                                     ||
||                                     ||
||      _____/_____          ||
||      _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ ||
||      _ // // // // // // // // // // // // // // // // // // // // // // ||
||      _ . _ ^ _ // // // // ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ||
||      / _ /                                     ||
||=====<<
```

```

_____
_ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \ _ _ \
_ // // // // // // // // // // // // // // // // // // // // // //
_ . _ ^ _ // // // // ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _
/_ /      \ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _ ^ _
_____
```

|           |           |           |                     |
|-----------|-----------|-----------|---------------------|
| .d8888b.  | .d8888b.  | 888888888 | .d8888b.            |
| d88P Y88b | d88P Y88b | 888       | d88P Y88b           |
| 888 888   | 888 888   | 888       | 888                 |
| 888 888   | .d88P     | Y8P       | 88888888b.          |
| 888 888   | .od888P°  |           | °Y88b 888P °Y88b    |
| 888 888   | d88P°     | d8b       | 888 888 888         |
| Y88b d88P | 888°      | Y8P       | Y88b d88P Y88b d88P |
| °Y8888P°  | 888888888 |           | °Y8888P° °Y8888P°   |

SubAlgoritmo pom Set Numero: Este subalgoritmo construye los dígitos de los minutos o segundos para mostrarlos en la pantalla del temporizador. Define las representaciones gráficas de los dígitos del 0 al 9 y copia los caracteres de una matriz que se utiliza para construir el número completo. Este devuelve el número dígito a dígito.

SubProceso pom Imprimir Titulo: Este subproceso imprime el título y el subtítulo de la pantalla del temporizador. Define las representaciones gráficas del título y el subtítulo y las muestra en la pantalla.

SubProceso pom Calcular Hora Inicio: Este subproceso calcula la hora de inicio del temporizador a partir de la función HoraActual() que viene en PSEINT. Obtiene la hora

actual, calcula las horas y minutos actuales y convierte estos valores a formato de texto para formar la hora de inicio que se muestra en la pantalla.

SubProceso CalculaHoraCierre: Este subproceso calcula la hora de cierre del temporizador. Obtiene la hora actual, calcula las horas y minutos actuales, y determina la hora de cierre sumando los tiempos de acción y pausa que restan para terminar la ronda completa. Luego, convierte estos valores a formato de texto para formar la hora de cierre que se muestra en la pantalla.

----- Hora de inicio 02:05 -- Hora de fin 02:50 -----

SubProceso pom Pantalla Inicio: Este subproceso muestra una animación de inicio para el módulo Pomodoro. La animación consiste en una serie de pantallas que se muestran y borran en intervalos de tiempo específicos simulando la realización de una presentación interactiva.

SubProceso pom Pantalla Pausa: Este subproceso muestra una pantalla de pausa entre los submódulos de acción y pausa del temporizador Pomodoro. Para seguir es necesario que el usuario toque una tecla, permitiéndole decidir cuando seguir.

PRESIONE UNA TECLA PARA CONTINUAR...

21

