

Lección 1:

Proyecto Final



Tiempo de ejecución: 14 horas
Planteamiento de la sesión

Materiales:



- [Qué es la técnica Pomodoro y cómo usarla para mejorar tu productividad](#)
- [Initial_Solution+-+pomodoro.zip](#)

Es hora de trabajar en un proyecto aún más retador, en donde se apliquen los conceptos de interfaces, bases de datos, visualización de datos y si se quiere, programación orientada a objetos y es la construcción de un aplicativo para trabajar con la técnica pomodoro **Qué es la técnica Pomodoro y cómo usarla para mejorar tu productividad.**

La técnica se basa en establecer una serie de pasos para completar un listado de actividades pendientes.

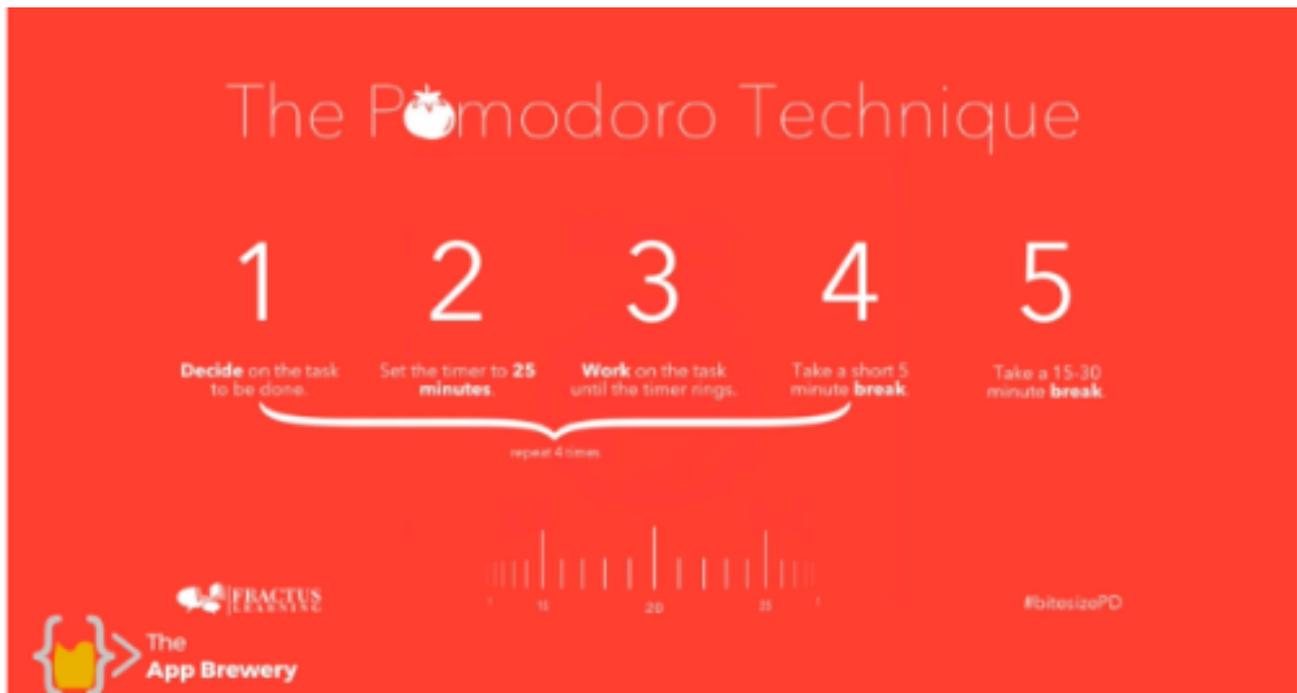
El primer paso consta de realizar un listado de tareas pendientes, lo ideal aquí es que prioricemos esa lista de tareas.

El segundo paso es configurar un temporizador a 25 minutos, tiempo durante el cual vamos a trabajar concentrados en la tarea que más esfuerzo nos demanda, aquí vamos a evitar toda las interrupciones y distracciones posibles.



El siguiente paso es tomar un descanso de 5 minutos cada que se completen los 25 minutos de trabajo, a este bloque o esfuerzo se le conoce como 1 pomodoro.

Ahora debe repetir los pasos anteriores durante 4 pomodoros, al cumplir los pomodoros entonces cambiamos el descanso del final de 5 minutos a un descanso de 15 a 30 minutos.





Al finalizar la iteración completa, volvemos a comenzar con el primer paso y así hasta que logremos completar el listado de tareas propuestas.

Requerimientos

El aplicativo a construir debe implementar estos pasos del pomodoro, utilizando interfaces. La interfaz debe contar con el temporizador de cuenta regresiva, botones para iniciar, cancelar y pausar el pomodoro, debe contar con un indicador del pomodoro que se cursa, es decir si es el primero, o el tercero.

Al finalizar cada pomodoro debe activar el tiempo de descanso e indicar también con un temporizador el tiempo de descanso restante, al completar el cuarto pomodoro, el temporizador debe ajustarse al tiempo más extenso, supongamos que 30 minutos.

Finalmente se debe almacenar el bases de datos cada pomodoro completado con fecha y hora, junto con un botón en la aplicación que despliega una ventana que muestre el total de pomodoros completados por el usuario.

Para la visualización de datos, puede pensar en una gráfica que permita ver el rendimiento de pomodoros completados contra la fecha, puede almacenar también los pomodoros cancelados y presentar algún reporte de pomodoros cancelados VS pomodoros completados.

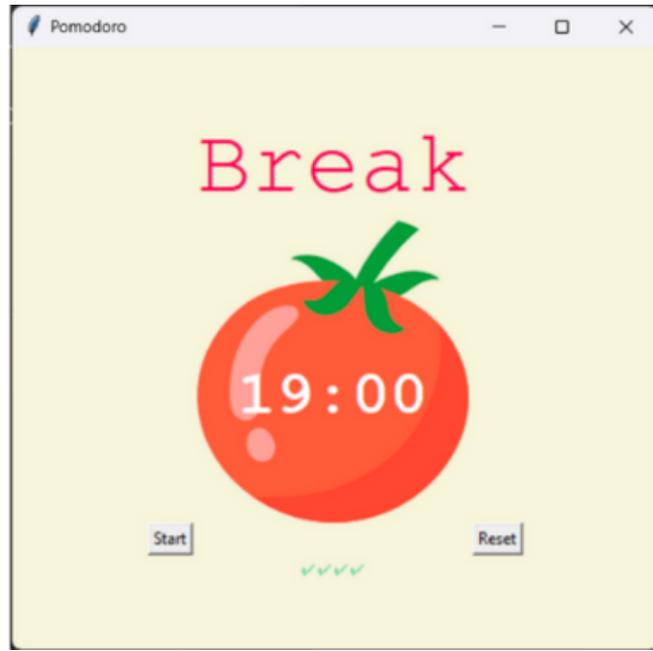




TIC

Aquí proporcionamos a los estudiantes, si se quiere, los recursos de imágenes que se incluyen en la solución inicial del pomodoro.

[Initial_Solution+-+pomodoro.zip](#)



```
def save_tasks_to_file(self):  
    with open("tasks.csv","w",newline="",encoding="utf-8") as file:  
        writer = csv.DictWriter(file, fieldnames=["description","status"])
```

```
        writer.writeheader()  
        for task in self.tasks:  
            writer.writerow(task)
```

```
def load_tasks_from_file(self):  
    try:  
        with open("tasks.csv","r",encoding="utf-8") as file:  
            reader = csv.DictReader(file)  
            self.tasks = [row for row in reader]  
    except FileNotFoundError:  
        # If file doesn't exist, initialize an empty list of tasks  
        self.tasks = []
```

```
def main():  
    root = tk.Tk()  
    app = TaskManagerApp(root)  
    root.mainloop()
```

```
if __name__ ==  
    "__main__":  
    main()
```

