

Análisis Lineal del Código: `validation_logic.py`

El archivo `validation_logic.py` es un módulo especializado y muy enfocado. Su única responsabilidad es contener la lógica de negocio específica para validar si un usuario tiene permiso de acceso en un momento determinado, basándose en reglas de horario. Centralizar esta lógica aquí hace que el resto del código, como la máquina de estados, sea más limpio, ya que solo necesita llamar a estas funciones en lugar de implementar la lógica de fechas y horas directamente.

El script comienza importando la librería `datetime`, que es la herramienta estándar de Python para trabajar con fechas, horas y tiempo. No necesita otras librerías externas o módulos propios, lo que lo hace muy autónomo y fácil de probar.

La primera función, `verificar_horario_trabajador(h_inicio_str, h_fin_str)`, está diseñada para los usuarios de tipo "Trabajador". Recibe dos argumentos de texto: la hora de inicio y la hora de fin del turno del trabajador (ej. "08:00", "17:00"). Primero, se asegura de que los horarios no estén vacíos. Luego, obtiene la fecha y hora actual con `datetime.datetime.now()`. De este objeto, extrae el día de la semana con `.weekday()`, que devuelve un número (Lunes=0, Martes=1, etc.). La primera regla que aplica es verificar si el día actual está entre Lunes y Viernes (índices 0 a 4). Si no es un día laboral, devuelve `False` inmediatamente. Si es un día laboral, convierte las cadenas de texto de los horarios (`h_inicio_str` y `h_fin_str`) a objetos de tipo `time` con `datetime.datetime.strptime(...).time()`. Finalmente, compara la hora actual con el rango de inicio y fin para determinar si el trabajador está dentro de su turno. Toda la lógica está envuelta en un bloque `try...except` para manejar errores si los horarios en la base de datos tuvieran un formato incorrecto (por ejemplo, "9" en lugar de "09:00"), evitando que el programa se caiga.

La segunda función, `verificar_horario_visitante()`, es más simple porque el horario para los visitantes está "hardcodeado" o fijo en el código, en lugar de venir de la base de datos. En este caso, la regla es que los visitantes solo pueden acceder los Miércoles entre las 9:00 AM y las 10:00 AM. La función obtiene el día de la semana actual y comprueba si es Miércoles (índice 2). Si no lo es, devuelve `False`. Si es Miércoles, entonces compara la hora actual con el rango fijo de 9:00 a 10:00. Al igual que la función anterior, devuelve `True` si se cumplen todas las condiciones y `False` en caso contrario.

El bloque final, `if __name__ == '__main__':`, sirve para realizar pruebas rápidas y directas del módulo. Si ejecutas `python validation_logic.py`, el script llamará a ambas funciones de verificación e imprimirá en la consola si, en el momento exacto de la ejecución, un trabajador con un horario de 9 a 5 o un visitante tendrían acceso. Esto es muy útil para depurar la lógica de horarios sin necesidad de ejecutar toda la aplicación.