

Documentación del Sistema de Autocompletado de CFR

1. Visión General del Proyecto

Este proyecto consiste en un conjunto de scripts diseñados para **automatizar la búsqueda y el rellenado de Códigos CFR (Community Fleet Register)** para una lista de buques.

El sistema funciona mediante una arquitectura híbrida:

- **Interfaz y Lógica (R):** El usuario interactúa con los scripts de R. Estos se encargan de leer los datos de entrada (Excel, CSV, TXT), procesar la lista de buques, llamar al buscador y guardar los resultados.
- **Búsqueda y Scraping (Python):** Un script externo de Python (`buscar_cfr.py`) es el encargado de realizar la búsqueda real del CFR en la fuente de datos externa (una página web del gobierno). Devuelve los resultados en formato JSON para que R pueda procesarlos fácilmente.

El objetivo principal es tomar un archivo con nombres de buques, identificar cuáles no tienen un CFR asignado, y rellenar esos campos vacíos automáticamente, modificando el archivo original de forma segura.

2. Prerrequisitos y Configuración Inicial

Antes de usar el sistema, asegúrate de tener todo lo siguiente configurado:

1. **R y RStudio:** Se recomienda tener instalada la última versión de R y RStudio.
2. **Python:** Python 3.x debe estar instalado en el sistema y, lo más importante, **debe estar añadido al PATH del sistema**. Esto permite que R pueda ejecutar comandos de Python desde la consola.
 - Puedes verificarlo abriendo una terminal o CMD y escribiendo `python --version`. Si obtienes una versión, está todo correcto.
3. **Archivos del Proyecto:** Todos los archivos deben estar en la **misma carpeta**:
 - El script principal de R (que contiene las funciones).
 - El script de Python: `buscar_cfr.py`.
 - El archivo de datos que quieres procesar (p.ej., `mi_flota.xlsx`).
4. **Librerías de R:** El script necesita las siguientes librerías de R. Si no las tienes, puedes instalarlas con `install.packages()`:
`install.packages(c("jsonlite", "readxl", "writexl", "openxlsx"))`

3. Guía de Uso Rápido

Para un usuario que solo quiere ejecutar el proceso, los pasos son muy sencillos:

Paso 1: Verificar el Entorno

Abre el script de R en RStudio y ejecuta la función de verificación. Esta función comprobará si Python está instalado y si tiene las librerías necesarias (selenium, webdriver-manager). **Si faltan, intentará instalarlas automáticamente.**

```
# Carga primero el script completo en la memoria de R (puedes usar el botón "Source" en RStudio)
verificar_entorno_python()
```

Si todo va bien, verás mensajes de confirmación (✓). Si hay un error, el script se detendrá con un mensaje explicativo. En caso de error con las librerías de python ejecute:

```
pip install selenium webdriver-manager
```

Paso 2: Procesar tu archivo

Usa la función principal para modificar tu archivo Excel directamente:

```
# Modificar la primera hoja del archivo "mi_flota.xlsx"
completar_cfr_y_modificar_excel(ruta_excel = "mi_flota.xlsx")

# Modificar una hoja específica por su nombre, por ejemplo "Datos 2023"
completar_cfr_y_modificar_excel(ruta_excel = "mi_flota.xlsx", hoja = "Datos 2023")
```

El archivo Excel original será sobrescrito con los CFRs completados, manteniendo intactas las demás hojas y el formato.


4. Documentación Detallada (Para Desarrolladores)

Aquí se describe el propósito y funcionamiento interno de cada componente del sistema.

4.1. buscar_cfr.py (Script de Python - El Buscador)

Este script es el corazón del sistema. Es un **web scraper** automatizado que simula a un usuario navegando en un navegador web para obtener la información requerida.

- **Propósito:** Dado un nombre de buque, navegar al censo de flota pesquera del gobierno, buscar ese buque, extraer su CFR y estado, y devolver esta información en un formato estructurado (JSON).
- **Tecnologías:**
 - **Python:** Lenguaje en el que está escrito.
 - **Selenium:** La librería principal para la automatización del navegador.
 - **WebDriver-Manager:** Una utilidad que descarga y gestiona automáticamente el chromedriver correcto, facilitando enormemente el despliegue.
- **Funcionamiento:**

- **Inicialización:** El script se ejecuta en modo **headless** (sin interfaz gráfica) para mayor eficiencia. Gracias a ChromeDriverManager, configura automáticamente el driver del navegador.
- **Navegación:** Abre la URL del censo de buques, espera a que los elementos carguen y realiza la búsqueda introduciendo el nombre del buque.
- **Filtrado Preciso:** En la página de resultados, filtra los buques para procesar solo aquellos cuyo nombre coincide **exactamente** con el buscado. Esto es clave para evitar ambigüedades.
- **Extracción de Datos:** Para cada resultado coincidente, visita la ficha del buque y extrae los campos "Estado" y "CFR".
- **Manejo de Errores:** Utiliza un bloque try...finally para garantizar que el navegador se cierre siempre, incluso si ocurre un error (p.ej., la web cambió su estructura).
- **Salida de Datos:** Convierte la lista de resultados a una cadena **JSON** y la imprime. Esta salida es la que el script de R captura para su procesamiento.
- **Puntos Clave para Mantenimiento:**
 -  **Punto de Fallo Principal:** Los **selectores CSS** (By.ID, By.CSS_SELECTOR) son el componente más frágil. Si la web del Ministerio cambia su estructura HTML, estos selectores deberán ser actualizados.
 - **URL Fija:** La URL de búsqueda está escrita en el código. Si cambia, debe ser actualizada.

4.2. Funciones de R

`verificar_entorno_python()`

- **Propósito:** Asegurar que el entorno de Python está listo para ser utilizado por R.
- **Funcionamiento:**
 1. Comprueba que python sea un comando ejecutable en el sistema.
 2. Verifica que las librerías selenium y webdriver-manager estén instaladas. Si no lo están, intenta instalarlas automáticamente usando pip.

`obtener_cfr(nombre_buque)`

- **Propósito:** Es el puente entre R y Python. Para un único nombre de buque, obtiene su CFR.
- **Funcionamiento:**
 1. Llama al script buscar_cfr.py pasándole el nombre del buque.
 2. Captura la salida JSON del script de Python.
 3. Parsea el JSON y lo convierte a un data.frame de R.
 4. Aplica una lógica de decisión: si hay un único buque activo, devuelve su CFR; si hay varios, devuelve "consultar manualmente"; si no hay ninguno, devuelve un mensaje apropiado.

`completar_cfr_y_modificar_excel(ruta_excel, hoja, guardar)`

- **Propósito:** Función principal y más robusta, diseñada para trabajar con archivos Excel de forma segura.
- **Funcionamiento:**

1. **Protección de datos:** Lee **todas las hojas** del Excel para no perder información al guardar.
2. **Identificación de columnas:** Busca de forma flexible las columnas de "Buque" y "CodigoCfr".
3. **Procesamiento eficiente:** Itera solo sobre las filas con CFR vacío y utiliza un sistema de **caché** para no buscar dos veces el mismo nombre de buque.
4. **Guardado seguro:** Si guardar = TRUE, escribe todas las hojas (con la procesada ya actualizada) de vuelta al archivo original.