Añadir una funcionalidad avanzada al campo "Cliente" en la sección de carga de documentos (self.cliente\_combo). La idea es que, además de la lista de clientes existentes, haya una opción especial para buscar un cliente por nombre o ID, y que al seleccionarla, el campo de cliente se transforme temporalmente en un cuadro de búsqueda.

Esto implica un cambio dinámico en la interfaz de usuario, donde un QComboBox puede ser "reemplazado" visualmente por un QLineEdit para la búsqueda, y luego volver al QComboBox para la selección final o si se desactiva la búsqueda.

Aquí te presento el plan de trabajo detallado para lograrlo:

**Plan de Trabajo**

**Fase 1: Preparación de la UI (setup\_ui)**

1. **Añadir el campo de búsqueda de cliente temporal:** Crear un QLineEdit (self.cliente\_search\_input) y una etiqueta (self.cliente\_search\_label) que se usarán cuando se active el modo de búsqueda. Estos deben estar inicialmente ocultos.
2. **Añadir la opción "Buscar Cliente por Nombre o ID" al ComboBox:** Insertar esta nueva opción especial en self.cliente\_combo.

**Fase 2: Lógica de Control (métodos del DocumentosModule)**

1. **Manejar la selección del ComboBox (\_handle\_cliente\_combo\_selection\_mode):**
   * Crear un nuevo método que se conectará al currentIndexChanged de self.cliente\_combo.
   * Este método detectará si se seleccionó la opción "Buscar Cliente por Nombre o ID".
   * Si es así, ocultará self.cliente\_combo y mostrará self.cliente\_search\_label y self.cliente\_search\_input.
   * Si se selecciona un cliente real, volverá a ocultar el campo de búsqueda y mostrará el self.cliente\_combo.
2. **Implementar la búsqueda en el QLineEdit (\_perform\_cliente\_search):**
   * Conectar el textChanged de self.cliente\_search\_input a un método que activará la lógica de búsqueda.
   * Este método llamará a una función en clientes\_logic.py para buscar clientes.
   * **Nota importante:** Por ahora, esta fase se centrará en *activar la búsqueda*. La visualización de los resultados y la selección de un cliente de esos resultados será un paso posterior, ya que añade una complejidad significativa a la UI. Asumiremos que la búsqueda se disparará, y en una siguiente iteración definiremos cómo se muestran y seleccionan los resultados.

**Fase 3: Integración y Refinamiento (Conexiones y Comportamiento)**

1. **Conectar señales:** Asegurarse de que el currentIndexChanged del self.cliente\_combo esté conectado al nuevo método de manejo.
2. **Comportamiento inicial:** Asegurarse de que la UI se inicialice correctamente (ComboBox visible, campo de búsqueda oculto).