[Módulo 1. Edición de Productos y RCLVs 1](#_Toc137216423)

[Introducción 1](#_Toc137216424)

[Lectura de una edición 2](#_Toc137216425)

[Guardado de una Edición 3](#_Toc137216426)

[Módulo 2. Edición de Productos 4](#_Toc137216427)

[Edición de Productos 4](#_Toc137216428)

[Módulo 3. Edición de un RCLV 6](#_Toc137216429)

[Aspectos Comunes 6](#_Toc137216430)

# Edición de Productos y RCLVs

## Introducción

### Comentarios

* Sólo se pueden editar registros en los siguientes status:
  1. Status *creado*, sólo por el usuario que creó el registro y durante una hora desde que lo creó.
  2. Status *creadoAprob*, sólo por Revisores.
  3. Status *aprobado*.
* Esos permisos los filtra el middleware *permUserReg*.
* El usuario que crea una edición verá inmediatamente su información en reemplazo de la vigente. Los demás usuarios, no verán la novedad hasta que sea revisada y aprobada.

|  |  |
| --- | --- |
| Entidades | Cómo se accede |
| Productos | Desde *Detalle de Producto* o *ABM Links* |
| RCLV | Desde *Agregar Producto*, *Edición de Prod.* o *Detalle de RCLV* |
| Links | Desde *Detalle de Producto* o *Edición de Producto* |

* Las ediciones:
  1. No tienen status de registro.
  2. Tampoco generan cambios de status en el registro original, independientemente de si son aprobadas o rechazadas.

### Criterio sobre el Valor para los Campos.

* Para los registros originales:
  1. Todos los campos a completar deben tener un valor.
  2. En los casos en que el usuario no está obligado a elegir/completar un valor, el sistema le asigna uno estándar.
* Para los registros de edición, solo se guardan con valor, los campos que difieren con el original.

### Impacto en Tablas.

* La edición se guarda en un registro nuevo en la tabla *prods\_edicion*, *rclvs\_edicion*, según el caso.
* Se guardan estos campos:
  1. *entidad*: la entidad del registro.
  2. *entidad\_id*: el id del registro original al que están vinculados.
  3. Los datos de data-entry que difieren del original.
  4. *editado\_por\_id* y *editado\_en*.

### Acceso

* Para el usuario creador:
  1. Puede acceder cuando el registro está recién creado y no transcurrió una hora desde que se creó. Durante esa hora tiene la exclusividad de uso.
  2. El registro original no se captura.
* Para cualquier usuario no revisor:
  1. Puede acceder cuando el producto está en status *creadoAprob*.
  2. Tiene exclusividad durante una hora desde que accede al registro original.
  3. El registro original se captura.
  4. La vista muestra un timer con la cuenta regresiva del tiempo de exclusividad disponible.

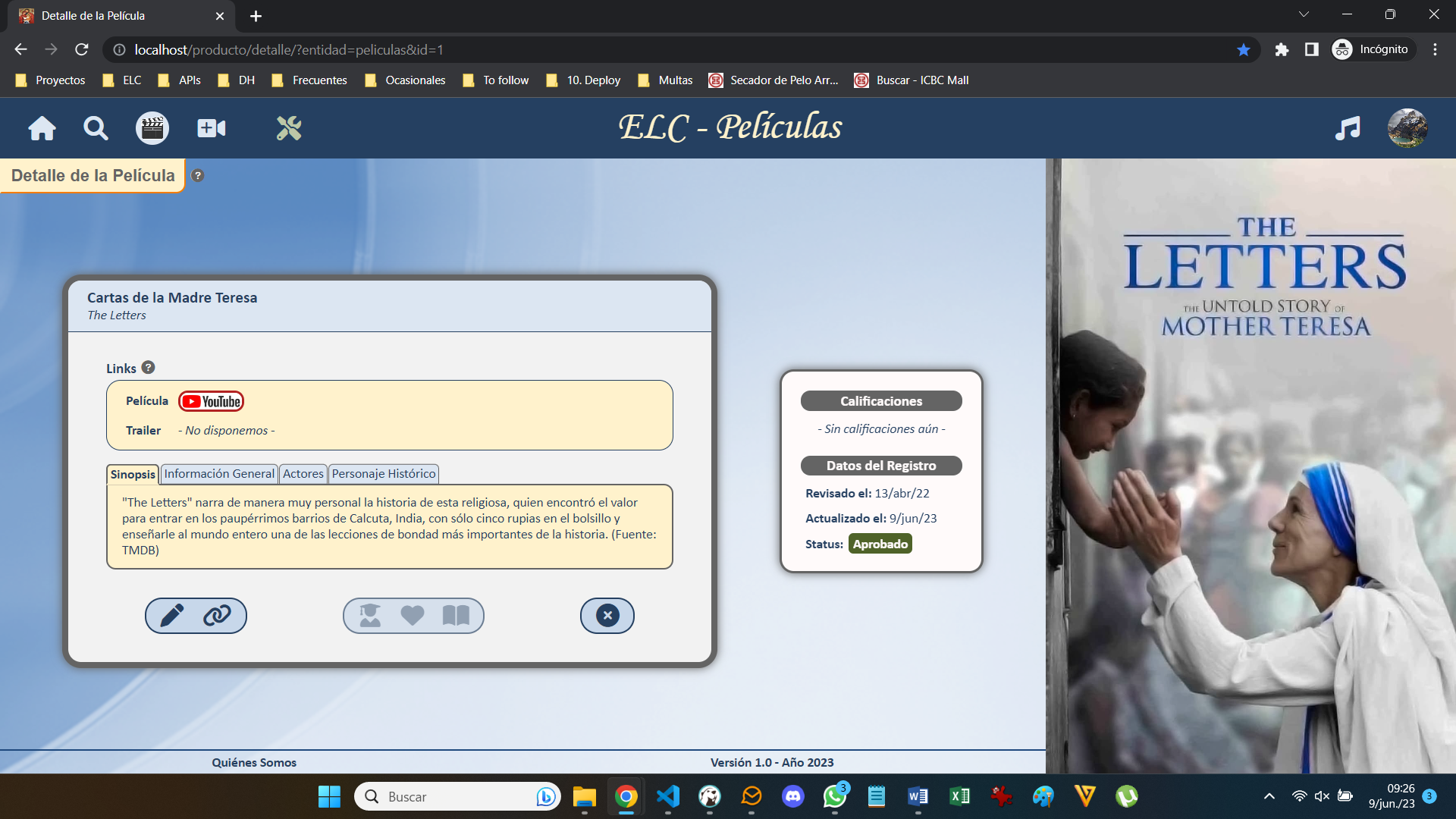
## Campos que se Actualizan

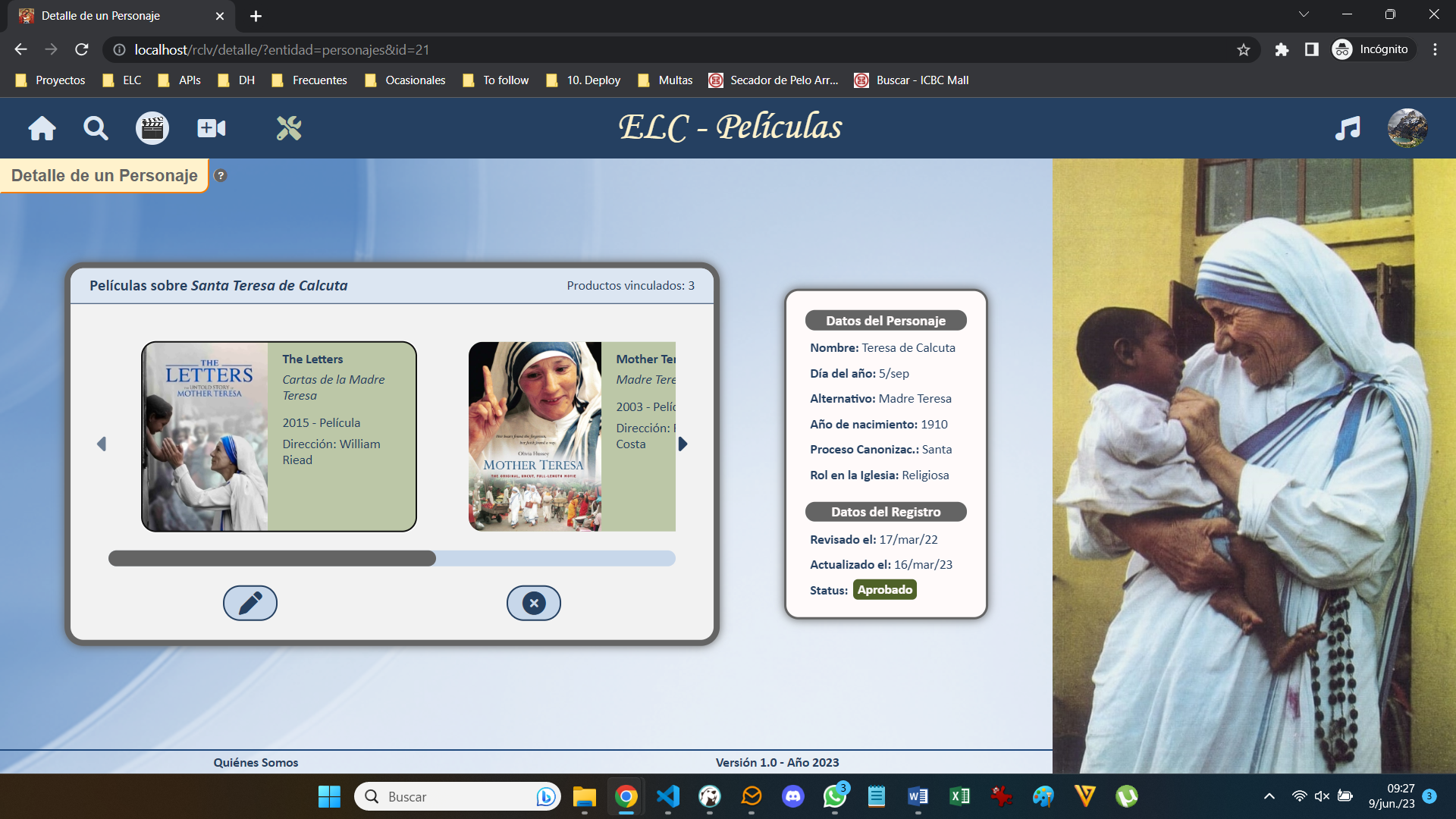
### RCLVs y Productos

* editadoPor\_id: userID
* editadoEn: ahora

## Lectura de una edición

### Vistas





### Pasos

* Se ejecuta la función ***obtieneOriginalEdicion***.
* Se combina el registro original con el editado, para obtener la versión combinada. Se debe tener cuidado de no pisar el id del original con el de la edición.

### Función obtieneOriginalEdicion

* Requiere los siguientes datos: ***entidad***, ***entID***, ***userID***.
* Se obtienen los registros de la versión original y la edición, ambos con todos sus *include* posibles.
  1. La edición se obtiene filtrando por ***campo\_id*** igual a *entID*, y ***editado\_por\_id*** igual a *userID*.
  2. Si no existe, va un objeto vacío.
* A ambos registros original y edición, se les eliminan todos los métodos con valor *null*.
* El valor *null* en *edicion* se asume que se debe a que coincide con el valor del registro *original*.

## Guardado de una Edición

### Pasos

* Para los RCLVs y Links,
  1. Si el original está en *status\_creado* y fue creado por el usuario, actualiza el registro original.
  2. De lo contrario, se ejecuta la función *guardaActEdicCRUD*.
* Para los productos, se ejecuta siempre la función *guardaActEdicCRUD*.

### Función guardaActEdicCRUD

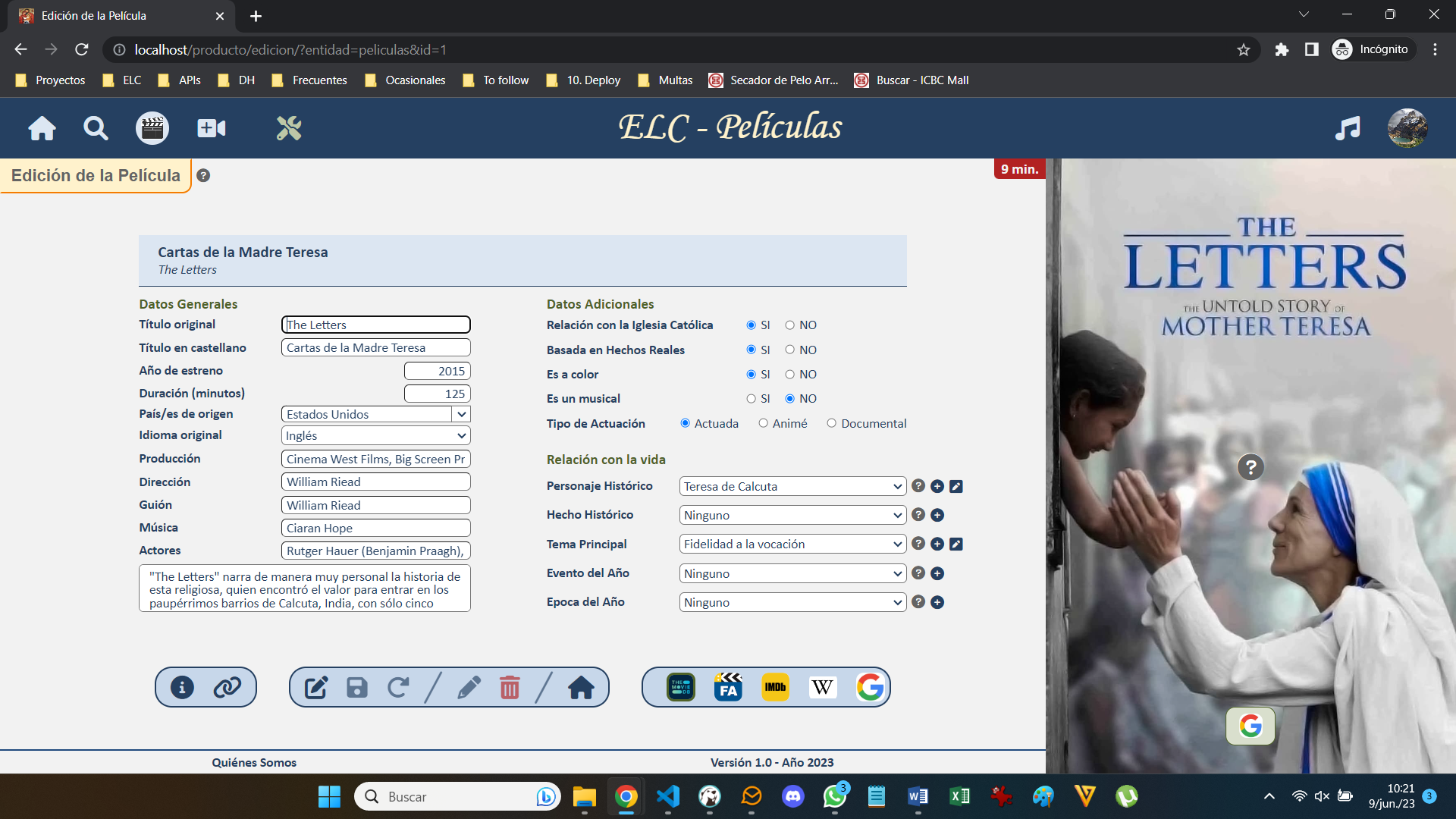
* Se le aplica a *edicion* la función *puleEdicion*.
* Si quedan campos,
  1. Se combina la *edición* con un array que tenga todos los campos con valor *null*: edición = {…camposNull, …edición}.
  2. Si existe la *edición\_id*, se actualiza el registro.
  3. Si no existe la *edición\_id*, se guarda un nuevo registro.
* Para guardar un nuevo registro, se deben obtener estos campos:
  1. ***entidadEdic***: la entidad de la edición.
  2. ***entidad\_id***: el nombre del *campo\_id* que vincula con el registro original. Se averigua por fórmula, a partir de la *entidad*.
  3. ***editado\_por\_id***: se obtiene del *userID*.
  4. ***producto\_id***: el nombre del campo se averigua del original (sólo para los links).

### Función puleEdicion

* Requiere los siguientes datos: ***original****,* ***edicion****,* ***entidad***.
* Se almacena el dato de *edición.id* en la variable *edición\_id*.
* Se descartan los campos que:
  1. No se comparan (ej: el ID).
  2. Iguales al original, tanto los de *dato* como los *include*.
  3. Con valor *null*.
* Si no quedan campos, elimina el registro de la tabla edición.
* Si quedan campos, se le reasigna el dato de *edición\_id*.
* A su vez se crea el objeto *camposNull*, para llevar un registro de los campos que son iguales al original.

# Edición de Productos

## Edición de Productos



### Ficha Técnica

|  |  |
| --- | --- |
| url | /producto/edicion |
| Middlewares aplicados | usuarios/filtro-usAltaTerm  usuarios/filtro-usPenalizaciones  usuarios/filtro-usAptoInput  producto/filtro-entidadValida  producto/filtro-IDvalido  captura/filtro-permUserReg  captura/capturaActivar |
| Métodos en la controladora | 2.1-Prod-RUD → prodDetEdic\_Form  2.1-Prod-RUD → prodEdic\_Guardar |
| Archivo de vista | 2.1-Prod-RUD/PR2-Edicion |
| Archivos de vista include | 2.1-Prod-RUD/Includes   * + PR-Edicion-Avatar   + PR-Edicion-Datos |
| Archivos de formato | 0-Compartido/Main/Flechas  2.0-Familias-CRUD/FM-RUD  2.0-Familias-CRUD/FM1-Prod-2Edicion  2.0-Familias-CRUD/FM1-Prod-0Varios |
| Archivos de front-end | 2.0-Familias-CRUD/FM-Flechas  2.1-Prod-RUD/PR2-Edicion-InputError  2.1-Prod-RUD/PR2-Edicion-vista\_RCLV |
| Métodos en la contr. API | 2.1-Prod-RUD → validaEdicion  2.1-Prod-RUD → obtieneVersionesProd  2.1-Prod-RUD → enviarAReqSession  2.1-Prod-RUD → eliminaEdicG |
| Vista natural anterior | *Detalle de Producto*  *ABM de Links*.  *Agradecimiento*, que aparece cuando se termina de hacer el alta de un producto. |
| Acceso desde el encabezado | - |
| Vista natural posterior | Detalle de Producto |
| Acceso a la vista posterior | Ícono de *información* |

### Impactos

|  |  |
| --- | --- |
| Archivos | * Si se ingresa un nuevo avatar,   + Se guarda el ingresado.   + Si había otro ingresado como edición, se elimina. * Si se elimina la edición, elimina el archivo |
| Tablas | * Se actualiza, guarda o elimina un registro en *prods\_edicion*. |

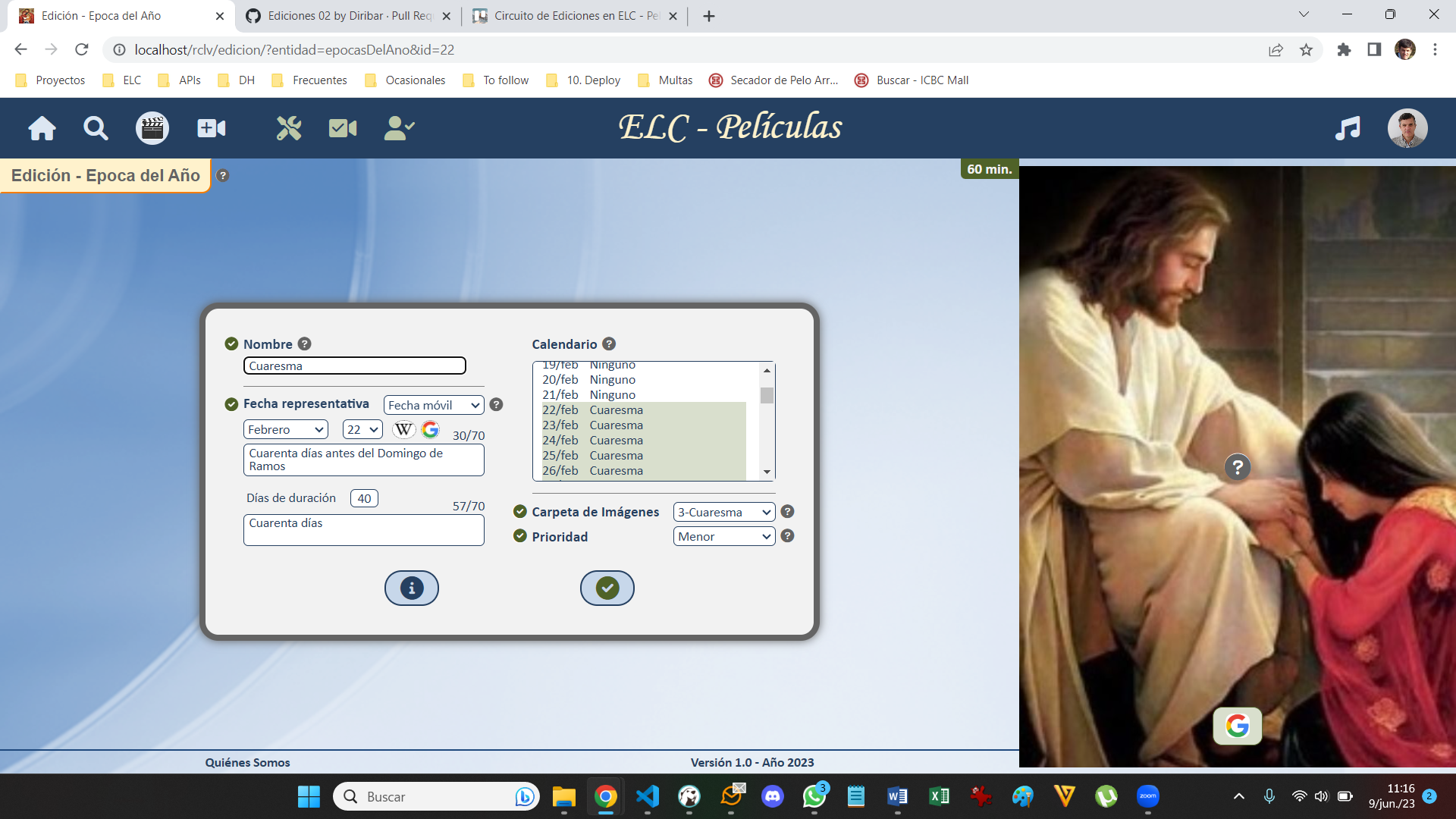
### Dinámica Particular de la Vista

* Se pueden ver tres versiones del registro:
  1. La versión original vigente.
  2. La edición guardada por el usuario. Si no existe, el ícono para ver esa versión está inactivo.
  3. Los cambios que se le quieren hacer a la edición. Si la edición no existe, serán los cambios respecto al original.
* Cada versión tiene sus propios avisos de error.
* Con los botones de abajo a la izquierda se puede:
  1. Ir a la vista de detalle.
  2. Ir a la vista de links.
* Con el grupo de botones de abajo al medio se puede:
  1. Cambiar de versión.
  2. Borrar la versión editada.
  3. Guardar o eliminar los cambios realizados a la versión editada.
* Con el grupo de botones de abajo a la derecha se puede consultar sobre el producto en los correspondientes sitios web.

### Comentarios

* Se tienen 2 fuentes de información:
  1. La información guardada en el registro original (*películas, colecciones, capitulos*).
  2. La información guardada en el registro de *prods\_edicion*, si hubiera uno. Si no lo hay, copia los datos del original.
* A su vez, el usuario puede actualizar o corregir la edición. Es necesario guardar los cambios antes de salir de la vista.
* Se puede editar cualquier campo de data entry, incluso el avatar.
* Si un usuario quiere eliminar su edición, depende del status del registro de la tabla *producto*.
  1. Status anteriores a *aprobado*: no está permitido
  2. Demás status: la puede eliminar.
* Se puede acceder a las vistas de Crear o Editar RCLVs.
  1. Cuando se elige esa opción, se guardan en sesión y cookie del producto los datos de la edición.
  2. Al regresar a la vista de edición, se actualiza el RCLV y en los demás campos se verán los datos que figuraban antes de salir.

# Edición de un RCLV



## Aspectos Comunes

### Ficha Técnica

|  |  |
| --- | --- |
| url | /rclv/edicion |
| Middlewares aplicados | filtrosPorUsuario/usAltaTerm  filtrosPorUsuario/usPenalizaciones  filtrosPorUsuario/usAptoInput  filtrosPorRegistro/entidadValida  filtrosPorRegistro/IDvalido  filtrosPorRegistro/statusCorrecto  filtrosPorRegistro/edicion  filtrosPorRegistro/rclvNoEditable  filtrosPorRegistro/permUserReg |
| *Adicionales para edicionForm*  captura/capturaActivar |
| *Adicionales para edicionGuardar*  captura/capturaInactivar |
| Métodos en la controladora | 2.2-RCLVs-CRUD  altaEdicForm  2.2-RCLVs-CRUD  altaEdicGuardar |
| Archivo de vista | 2.2-RCLVs-CRUD/RCLV1-AltaEdicion |
| Archivos de vista include | * 2.2-RCLVs-CRUD/   + RCLV1-AltaEdicion1-Avatar   + RCLV1-AltaEdicion2-Datos * 2.2-RCLVs-CRUD/Includes   + CMP-Errores   + CMP-TimerContador   + CMP-Cartel |
| Archivos de formato | 0-Compartido/Main/Flechas  2.0-Familias-CRUD/FM-RUD  2.0-Familias-CRUD/FM2-RCLV-AltaEdicion |
| Archivos de front-end | 2.2-RCLVs-CRUD/RCLV-AltaEdicion |
| Métodos en la contr. API | 2.2-RCLVs-CRUD  registrosConEsaFecha  2.2-RCLVs-CRUD  validaSector  2.2-RCLVs-CRUD  prefijos |
| Vista natural anterior | Circuito de Alta de Prod.  DatosAdics  Edición de Producto  Detalle de RCLV |
| Acceso desde el encabezado | - |
| Vista natural posterior | Ídem vista anterior |
| Acceso a la vista posterior | Cualquiera de los íconos de abajo |

### Impactos

|  |  |
| --- | --- |
| Tablas | * Si el RCLV fue creado por el usuario y está en status creado, actualiza el registro ***original*** que corresponda. * De lo contrario, actualiza, guarda o elimina un registro en ***rclvs\_edicion***. |

### Comentarios

* Si el usuario guarda con errores, el sistema elimina el avatar ingresado (eventualmente) y muestra los demás datos ingresados.
* Si lo desea, el usuario puede eliminar su edición.
* Si el original está en *status\_creado* y fue creado por el usuario, actualiza el registro original.
* De lo contrario, se ejecuta la función ***guardaActEdicCRUD*** para guardar la edición en *rclvs\_edicion*.
  1. En esa función, se usa la función *puleEdicion*.
  2. En el registro de edición se guardan solamente los campos con valor agregado, los demás van con el valor estándar *null*.
  3. Si existe la *edición\_id*, se actualiza el registro.
  4. Si no existe la *edición\_id*, se guarda un nuevo registro.