EstructuraTP.txt

Planificación del Release

Listado de Historias

* Emitir Licencia
* Calcular Vigencia de la Licencia
* Calcular Costo de la Licencia
* Imprimir Licencia
* Renovar Licencia
* Emitir una copia
* Listado de licencias que han expirado
* Listado de licencias vigentes por Criterios
* Dar de alta un titular
* Modificar los datos de un titular existente
* Dar de alta un usuario
* Modificar los datos de un usuario

Historias Refinadas

|  |  |
| --- | --- |
| Historia de Usuario | |
| Número: 0 | Nombre: Emitir una licencia |
| Usuario: Administrativo, Administrador | |
| Modificación Historia: 1 | Iteración asignada: 1 |
| Prioridad en negocio: **Alta**  (Alta/Media/Baja) | Puntos estimados: 1 |
| Riesgo en desarrollo: **Bajo**  (Alto/Medio/Bajo) |
| Descripción: |  |
| Un operador podrá emitir una licencia que quedará registrada en el sistema y contiene los siguientes datos:  -Clase de la licencia:   * Clase A: Ciclomotores, motocicletas, y triciclos motorizados. * Clase B: Automóviles y camionetas con acoplado. * Clase C: Camiones sin acoplado y los comprendidos en la clase B. * Clase D: Servicio de transporte de pasajeros, emergencia, seguridad y los comprendidos en las clase B o C, según el caso. * Clase E: Camiones articulados o con acoplado, maquinaria especial no agrícola y los comprendidos en las clases B y C. * Clase F: Automotores especialmente adaptados para discapacitados. * Clase G: Tractores agrícolas y maquinaria especial agrícola. * Las licencias de clases A, B, F, G requieren que el titular tenga, al menos, 17 años de edad. * Las licencias de clases C, D, E son licencias para conductores profesionales, y requieren que el titular tenga, al menos, 21 años de edad. Para poder emitir una licencia de este tipo, se requiere que al titular se le haya otorgado una licencia de clase B, al menos un año antes. Además, no puede otorgarse una licencia profesional por primera vez a personas de más de 65 años.   -Titular asociado.  -Trámite asociado, incluyendo número de trámite, operador que lo realiza y fecha del mismo.  -Observaciones: Texto que indica limitaciones / condiciones del titular de la licencia y requerimientos especiales para conducir, como por ejemplo, llevar anteojos. | |
| Observaciones: |  |

|  |  |
| --- | --- |
| Historia de Usuario | |
| Número: 1 | Nombre: Calcular Vigencia de Licencia |
| Usuario: Sistema | |
| Modificación Historia: 1 | Iteración asignada: 1 |
| Prioridad en negocio: **Alta**  (Alta/Media/Baja) | Puntos estimados: 1 |
| Riesgo en desarrollo: **Medio**  (Alto/Medio/Bajo) |
| Descripción |  |
| Durante la emisión de la licencia, se establece la vigencia de la misma, de acuerdo a la siguiente tabla:  - Menores de 21 años: 1 año la primera vez y 3 años las siguientes  - Hasta 46 años: 5 años  - Hasta 60 años: 4 años  - Hasta 70 años: 3 años  - Mayores de 70 años: 1 año  El día y mes de la fecha de vencimiento deben coincidir con el día y mes de la fecha de nacimiento del titular, respectivamente. La fecha de inicio de vigencia debe ser la fecha del sistema, y no puede cambiarse. | |
| Observaciones: |  |

|  |  |
| --- | --- |
| Historia de Usuario | |
| Número: 2 | Nombre: Calcular Costo de Licencia |
| Usuario: Sistema | |
| Modificación Historia: 1 | Iteración asignada: 1 |
| Prioridad en negocio: **Media**  (Alta/Media/Baja) | Puntos estimados: 1 |
| Riesgo en desarrollo: **Medio**  (Alto/Medio/Bajo) |
| Descripción |  |
| El costo de la licencia varía de acuerdo a la clase y la vigencia en años y no puede ser modificado directamente por un operador:   * A -> {5 años: $40, 4 años: 30, 3 años: $25, 1 año: $20} * B -> {5 años: $40, 4 años: 30, 3 años: $25, 1 año: $20} * C -> {5 años: $47, 4 años: 35, 3 años: $30, 1 año: $23} * D -> {5 años: $59, 4 años: $44, 3 años: $39, 1 año: $29} * E -> {5 años: $59, 4 años: $44, 3 años: $39, 1 año: $29} * F -> {5 años: $59, 4 años: $44, 3 años: $39, 1 año: $29} * G -> {5 años: $40, 4 años: $30, 3 años: $25, 1 año: $20}   A todos estos precios, se le agrega un costo de 8$ por gastos administrativos. | |
| Observaciones: |  |

|  |  |
| --- | --- |
| Historia de Usuario | |
| Número: 3 | Nombre: Imprimir Licencia |
| Usuario: Administrativo, Administrador | |
| Modificación Historia: 1 | Iteración asignada: 1 |
| Prioridad en negocio: **Alta**  (Alta/Media/Baja) | Puntos estimados: 3 |
| Riesgo en desarrollo: **Alta**  (Alto/Medio/Bajo) |
| Descripción |  |
| Se imprime la licencia y se emite un comprobante al interesado para que abone el costo del trámite en Caja | |
| Observaciones: |  |

|  |  |
| --- | --- |
| Historia de Usuario | |
| Número: 4 | Nombre: Renovar licencia |
| Usuario:Administrativo, Administrador | |
| Modificación Historia: 1 | Iteración asignada: 1 |
| Prioridad en negocio: **Alta**  (Alta/Media/Baja) | Puntos estimados: 1 |
| Riesgo en desarrollo: **Medio**  (Alto/Medio/Bajo) |
| Descripción: |  |
| El titular de una licencia puede renovarla, ya sea por expiración de su fecha de vigencia, o por cambios en algunos de los datos de la misma. El costo de renovación por modificación de datos es el mismo que el de emisión del carnet. | |
| Observaciones: | Renovación por cambios que no implica vigencia, clase o datos esenciales como DNI. Renovar licencia por expiración permite modificar datos no esenciales (nunca DNI, clase, etc). |

|  |  |
| --- | --- |
| Historia de Usuario | |
| Número: 5 | Nombre: Emitir una copia |
| Usuario: Administrativo, Administrador | |
| Modificación Historia: 1 | Iteración asignada: 2 |
| Prioridad en negocio: **Baja**  (Alta/Media/Baja) | Puntos estimados: 2 |
| Riesgo en desarrollo: **Medio**  (Alto/Medio/Bajo) |
| Descripción |  |
| Es posible pedir una nueva copia de la licencia (duplicado, triplicado, etc.), por extravío, robo o deterioro del carnet original. El costo de este trámite es de $ 50. Se utilizarán los mismos datos de la licencia original. | |
| Observaciones: | Costo fijo, no cambia. Puede haber infinitas copias pero deben quedar registradas. Se pueden modificar datos no esenciales. |

|  |  |
| --- | --- |
| Historia de Usuario | |
| Número: 6 | Nombre: Listado de licencias que han expirado |
| Usuario: Administrativo, Administrador | |
| Modificación Historia: 1 | Iteración asignada: 2 |
| Prioridad en negocio: **Media**  (Alta/Media/Baja) | Puntos estimados: 1 |
| Riesgo en desarrollo: **Baja**  (Alto/Medio/Bajo) |
| Descripción |  |
| Se emite un listado en base a la fecha de expiración de las mismas | |
| Observaciones: | Filtrarse y ordenarse por el criterio que deseemos. |

|  |  |
| --- | --- |
| Historia de Usuario | |
| Número: 7 | Nombre: Listado de licencias vigentes por criterios |
| Usuario: Administrativo, Administrador | |
| Modificación Historia: 1 | Iteración asignada: 2 |
| Prioridad en negocio: **Alta**  (Alta/Media/Baja) | Puntos estimados: 1 |
| Riesgo en desarrollo: **Bajo**  (Alto/Medio/Bajo) |
| Descripción |  |
| Se deberá un criterio de los siguientes:  Elegir un criterio de entre los siguientes:  -Nombre y apellido  -Grupo sanguíneo  -Factor RH  -Si el titular es donante de órganos  Y mostrar una lista de las licencias vigentes ordenada de acuerdo al criterio elegido | |
| Observaciones: |  |

|  |  |
| --- | --- |
| Historia de Usuario | |
| Número: 8 | Nombre: AM de titular |
| Usuario: Administrativo, Administrador | |
| Modificación Historia: 1 | Iteración asignada: 1 |
| Prioridad en negocio: **Alta**  (Alta/Media/Baja) | Puntos estimados: 1 |
| Riesgo en desarrollo: **Bajo**  (Alto/Medio/Bajo) |
| Descripción |  |
| Un titular es un contribuyente del Municipio con la capacidad de obtener su licencia de conducir. El empleado administrativo comprobará que la documentación presentada sea válida y completa. Los datos requeridos inicialmente son:   * Tipo y Nº de documento * Apellido y Nombre * Fecha de nacimiento * Dirección * Clase solicitada (A-G) * Grupo sanguíneo y factor RH * Donante de órganos (SI/NO) | |
| Observaciones: | Una vez que el contribuyente ha obtenido su carnet es posible modificar sus datos personales. |

|  |  |
| --- | --- |
| Historia de Usuario | |
| Número: 9 | Nombre: AM de usuario |
| Usuario: Administrador | |
| Modificación Historia: 1 | Iteración asignada: 2 |
| Prioridad en negocio: **Baja**  (Alta/Media/Baja) | Puntos estimados: 1 |
| Riesgo en desarrollo: **Baja**  (Alto/Medio/Bajo) |
| Descripción |  |
| Para operar el sistema se debe poder dar de alta a los usuarios administrativos. También deberá existir un superusuario o usuario administrador responsable del mantenimiento total del sistema. | |
| Observaciones: | En cualquier momento es posible modificar los datos de los usuarios encargados de operar el sistema. Para dicha acción se debe contar con privilegios de administrador. |

Estimación de Esfuerzo, Riesgo y Prioridad

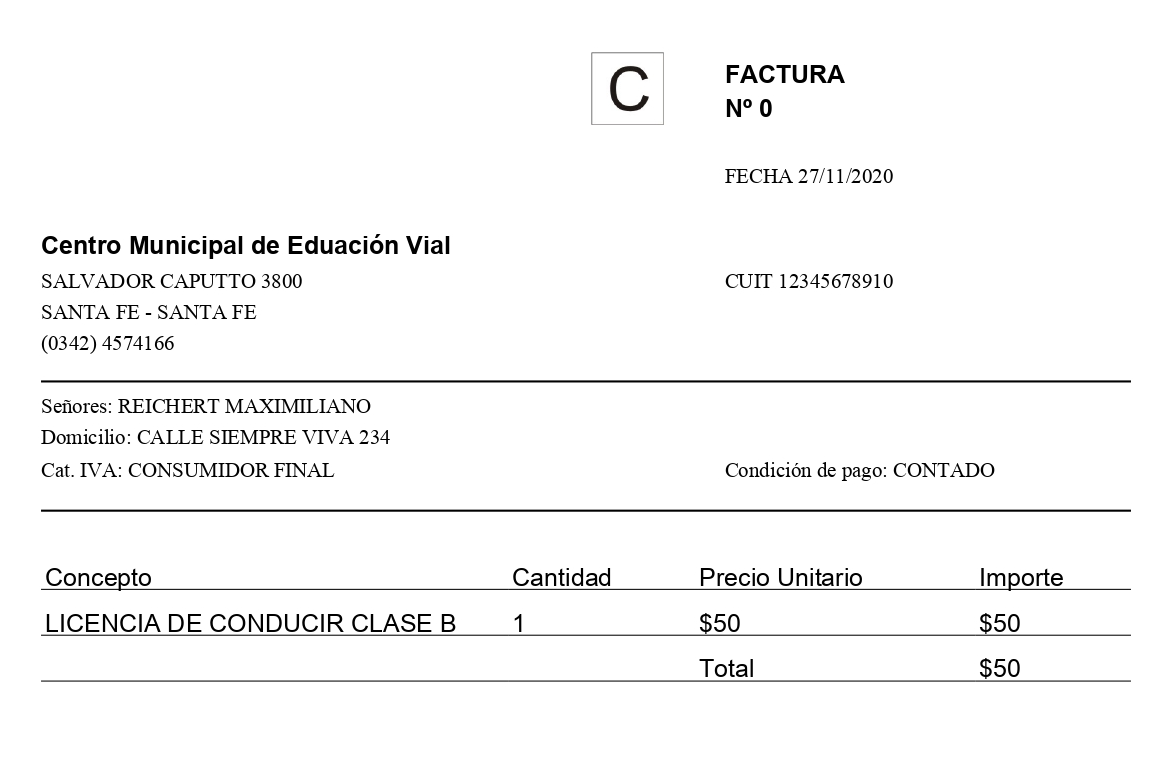
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Número | Nombre | Story Points | Prioridad en negocio | Riesgo en desarrollo |
| 0 | Emitir una licencia | 1 | Alta | Bajo |
| 1 | Calcular vigencia de la licencia | 1 | Alta | Medio |
| 2 | Calcular costo de la licencia | 1 | Media | Medio |
| 3 | Imprimir licencia | 3 | Alta | Alto |
| 4 | Renovar licencia | 1 | Media | Medio |
| 5 | Emitir una copia | 2 | Baja | Medio |
| 6 | Listado de licencias que han expirado | 1 | Media | Bajo |
| 7 | Listado de licencias vigentes | 1 | Baja | Bajo |
| 8 | AM de titular | 1 | Alta | Bajo |
| 9 | AM de usuario | 1 | Baja | Bajo |

Spikes

Spike de licencia



Spike de ticket



Definición de Velocidad y Alcance Metáfora

iteración= 3 semanas.

1 story point= 1 semana.

velocidad= 7 story points por iteración

Número de iteraciones necesarias: total story points/velocidad= 13/7= 2

Fichas CRC

|  |  |
| --- | --- |
| Licencia | |
|  | -Titular  -Tramite  -GestorLicencia |
|

|  |  |
| --- | --- |
| Titular | |
|  | -Licencia  -GestorTitular |
|

|  |  |
| --- | --- |
| Tramite | |
|  | -Usuario |
|

|  |  |
| --- | --- |
| Usuario | |
|  | -GestorUsuario  -Tramite |
|

|  |  |
| --- | --- |
| GestorLicencia | |
| - Dar de alta licencia.  - Modificar licencia.  - Eliminar licencia.  - Recuperar licencia.  - Imprimir licencia.  - Imprimir ticket de pago.  - Calcular vigencia de licencia.  - Calcular costo de licencia. | -Licencia  -UIListadoDeLicenciasQueHanExpirado  -UIListadoDeLicenciasVigentes  -UIRenovarLicencias  -UIEmitirLicencia |
|

|  |  |
| --- | --- |
| GestorTitular | |
| - Crear titular.  - Eliminar titular.  - Modificar titular.  - Recuperar titular. | - Titular  - DAOTitular  - UIAltaTitular |
|

|  |  |
| --- | --- |
| GestorUsuario | |
| - Crear usuario.  - Eliminar usuario.  - Modificar usuario.  - Recuperar usuario.  - Iniciar sesion. | -UIAltaUsuario  -Usuario  -DAOUsuario |
|

|  |  |
| --- | --- |
| GestorTramite | |
| - Crear tramite.  - Eliminar tramite.  - Modificar tramite. | -GestorUsuario  -GestorLicencia |
|

|  |  |
| --- | --- |
| DAOTItular | |
| - Guardar titular.  - Actualizar titular.  - Eliminar titular.  - Recuperar titular. | -Titular  -GestorTitular |
|

|  |  |
| --- | --- |
| DAOLicencia | |
| - Guardar licencia  - Eliminar licencia  - Recuperar licencia  - Listar licencias  - Filtrar licencias | - Licencia  - GestorLicencia |
|

|  |  |
| --- | --- |
| DAOUsuario | |
| - Guardar usuario.  - Eliminar usuario. | - Usuario  - GestorUsuario |
|

Fecha de Inicio del desarrollo: 2/11/2020

Planificación Iteracion 1

Tareas

|  |  |  |  |
| --- | --- | --- | --- |
| Task Card UI Emitir Licencia Programador: José Ignacio Carbone  Fecha: Task Points: 2  Número de historia: 0 | | | |
| Descripción:  Luego de hacer click en el botón “Emitir Licencia” en el menú principal, se muestra una pantalla con los siguientes campos para completar:   * **Tipo de licencia** (Combobox, obligatorio, opciones posibles: “A”, “B”, “C”, “D”, “E”, “F”, “G”). * Clase A: Ciclomotores, motocicletas, y triciclos motorizados. * Clase B: Automóviles y camionetas con acoplado. * Clase C: Camiones sin acoplado y los comprendidos en la clase B. * Clase D: Servicio de transporte de pasajeros, emergencia, seguridad y los comprendidos en las clase B o C, según el caso. * Clase E: Camiones articulados o con acoplado, maquinaria especial no agrícola y los comprendidos en las clases B y C. * Clase F: Automotores especialmente adaptados para discapacitados. * Clase G: Tractores agrícolas y maquinaria especial agrícola. * Las licencias de clases C, D, E son licencias para conductores profesionales, y requieren que el titular tenga, al menos, 21 años de edad. Para poder emitir una licencia de este tipo, se requiere que al titular se le haya otorgado una licencia de clase B, al menos un año antes. Además, no puede otorgarse una licencia profesional por primera vez a personas de más de 65 años. * **Titular asociado** (Objeto Titular, obligatorio, se selecciona ingresando tipo y número de documento y haciendo click en el botón “Buscar Titular”, que se habilita cuando se ha seleccionado el tipo y el número de documento. Si se encuentra el titular en el sistema, se muestra en el campo no editable etiquetado con “Titular”, nombre y apellido del titular seleccionado en color verde. De no ser encontrado el titular, se muestra en el mismo campo no editable “TITULAR NO ENCONTRADO” en color rojo, si se produce un error se muestra en el mismo campo “ERROR DEL SISTEMA AL BUSCAR EL TITULAR” en rojo). * **Observaciones** (String: 128, opcional)   Se debe validar que se haya seleccionado un tipo de licencia y un titular asociado antes de enviar. Observaciones es un campo opcional.  Una vez ingresados los datos, el empleado administrativo hace click en el botón “Emitir Licencia” y se invoca al gestor. (Ver tarea Lógica y Persistencia Emitir Licencia).  De ser exitosa la emisión, se muestra en pantalla un popup con el mensaje de éxito “Licencia emitida correctamente” en color verde.  De lo contrario, si el gestor lanza una excepción, se muestra un pantalla un popup con el mensaje de error en color rojo: “Se ha producido un error al emitir la licencia: ” seguido de la especificación del error, que se encuentra en el mensaje de la excepción. | | | |
|  | | | |
| Notas | | | |
| Fecha | Realizado | A realizar | Comentarios |
| 09/11/2020 | Estructura básica de la interfaz y pruebas unitarias | Implementar las acciones de los botones y validaciones de los campos |  |
| 10/11/2020 | Acciones de los botones y validaciones de los campos, manejo de errores | - |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Task Card Lógica y Persistencia Emitir Licencia Programador: José Ignacio Carbone  Fecha: Task Points: 3  Número de historia: 0 | | | |
| Descripción:  Luego de hacer click en el botón “Emitir Licencia” se llama a gestor titular para emitir la licencia. Antes de emitir la licencia se realizan las siguientes validaciones:   * Para emitir las licencias de clases A, B, F, G se valida que el titular tenga, al menos, 17 años de edad. De no cumplirse, se lanza una EmitirLicenciaException con el mensaje “*El titular debe tener 17 a*ños o más para obtener una licencia de clase $clase” * Para las licencia de clases C, D, E (licencias profesionales) se valida que el titular tenga, al menos, 21 años de edad, y que al menos haya obtenido una licencia de clase B, como mínimo, un año antes, y además, que únicamente si es la primera vez que el titular va a obtener una licencia profesional (clases C, D, E) se valida que la edad del titular sea inferior a 65 años. De no cumplirse, se lanza una EmitirLicenciaException con el mensaje “El titular debe tener 21 años o más para obtener una licencia de clase $clase” o “El titular debe haber obtenido una licencia de clase B, como mínimo, un año antes” o “El titular debe ser menor a 65 años para obtener su primera licencia profesional” según corresponda. * Se verifica que el titular nunca haya obtenido una licencia del mismo tipo (si la obtuvo alguna vez, debe renovarse, no generarse una nueva). De no cumplirse, se lanza una EmitirLicenciaException con el mensaje “El titular ya posee la licencia. Si expiró, debe renovarla”.   Realizadas las validaciones, se crea un objeto licencia y se le asignan tipo de licencia, objeto titular, observaciones, y además un objeto trámite conteniendo el usuario administrativo de la sesión actual al momento de realizarse el trámite, y la fecha al momento de realizarse el trámite, con un número de trámite autogenerado.  Durante la emisión de la licencia, se establece la vigencia de la misma, de acuerdo a la siguiente tabla:  - Menores de 21 años: 1 año la primera vez y 3 años las siguientes  - Hasta 46 años: 5 años  - Hasta 60 años: 4 años  - Hasta 70 años: 3 años  - Mayores de 70 años: 1 año  El día y mes de la fecha de vencimiento deben coincidir con el día y mes de la fecha de nacimiento del titular, respectivamente. La fecha de inicio de vigencia debe ser la fecha del sistema, y no puede cambiarse.  Finalmente, se persiste la licencia en la base de datos. De haber un error, se lanza una EmitirLicenciaException con el mensaje “Error al guardar la licencia en la base de datos. Si el problema persiste, contacte al administrador del sistema”. | | | |
|  | | | |
| Notas | | | |
| Fecha | Realizado | A realizar | Comentarios |
| 11/11/2020 | Estructura básica del gestor y del DAO, pruebas unitarias | Implementación del DAO, implementación de los métodos del gestor |  |
| 12/11/2020 | Implementación de los métodos del gestor | Implementación del DAO |  |
| 13/11/2020 | Implementación del DAO | - |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Task Card Calcular Vigencia de Licencia Programador: Cesar Lencinas  Fecha: Task Points: 3  Número de historia: 1 | | | |
| Descripción:  Método que dada la fecha de nacimiento del titular, se realiza el cálculo de la vigencia, según los criterios indicados (ver abajo).  Como resultado devuelve la fecha de vencimiento.  Criterios:  Durante la emisión de la licencia, se establece la vigencia de la misma, de acuerdo a la siguiente tabla:  - Menores de 21 años: 1 año la primera vez y 3 años las siguientes  - Hasta 46 años: 5 años  - Hasta 60 años: 4 años  - Hasta 70 años: 3 años  - Mayores de 70 años: 1 año  El día y mes de la fecha de vencimiento deben coincidir con el día y mes de la fecha de nacimiento del titular, respectivamente. La fecha de inicio de vigencia debe ser la fecha del sistema, y no puede cambiarse. | | | |
|  | | | |
| Notas | | | |
| Fecha | Realizado | A realizar | Comentarios |
| 11/12/2020 | Implementación del método que calcula la vigencia de acuerdo a los criterios especificados. |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Task Card | | | | Calcular Costo de Licencia | | | Programador:  Francisco Giancarelli |
| Fecha: | | | | Task Points: 1 | | |  |
| Número de historia: 2 | | | |  | | |  |
| Descripción: Se calcula el costo de la licencia según el tipo y período de la licencia, de acuerdo a la siguiente tabla:   * A -> {5 años: $40, 4 años: $30, 3 años: $25, 1 año: $20} * B -> {5 años: $40, 4 años: $30, 3 años: $25, 1 año: $20} * C -> {5 años: $47, 4 años: $35, 3 años: $30, 1 año: $23} * D -> {5 años: $59, 4 años: $44, 3 años: $39, 1 año: $29} * E -> {5 años: $59, 4 años: $44, 3 años: $39, 1 año: $29} * F -> {5 años: $59, 4 años: $44, 3 años: $39, 1 año: $29} * G -> {5 años: $40, 4 años: $30, 3 años: $25, 1 año: $20}   Además se le deben sumar 8$ por gastos administrativos. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 2/11 | Se realizó un método en la clase Licencia que calcula el costo a partir de fecha de vencimiento y otorgamiento y el tipo | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Task Card | | | | UI Alta titular | | | Programador: Cristian Maglianesi |
| Fecha: | | | | Task Points: 2 | | |  |
| Número de historia: 8 | | | |  | | |  |
| Descripción:Luego de hacer click en el botón “Crear titular” en el menú se abre un formulario en donde se requieren los siguientes datos:   * Tipo documento (ComboBox, opciones: le, lc, dni) * Nº de documento (int) * Apellido (String: 15) * Nombre (String: 15) * Fecha de nacimiento (date) * Dirección (String: 30) * Clase solicitada (A-G) (Combobox, obligatorio, opciones posibles: “A”, “B”, “C”, “D”, “E”, “F”, “G”) * Grupo sanguíneo (String: 2) * Factor RH (String: 1) * Donante de órganos (SI/NO) (CheckButton)   El CheckButton de donante de órganos debe estar por defecto en NO. Ambos check deben ser mutuamente excluyentes.  Luego de presionar el botón “Crear” se debe validar que los campos para completar no estén vacíos y que en los comboBox se haya elegido una opción. Si los campos están correctamente completados se debe llamar al gestor de titular para que valide si ya existe el titular. Si el gestor devuelve que el titular existe se debe mostrar un popup con el mensaje “El titular ya existe”, si devuelve que la operación fue exitosa se debe mostrar un popup con el mensaje “El titular se ha creado exitosamente”  Si se presiona el botón “Volver” se debe regresar al menú principal | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Task Card | | | | Lógica y persistencia Alta Titular | | | Programador: Cristian Maglianesi |
| Fecha: | | | | Task Points: 2 | | |  |
| Número de historia: 8 | | | |  | | |  |
| Descripción:Luego de hacer click en el botón “Crear” en la UI de alta titular el gestor debe llamar a la dao de titular para que busque un titular por DNI. Si la dao devuelve un titular se debe enviar un error a la UI de alta titular, si devuelve vacío el gestor devolver a llamar al dao de titular para que cree la entrada en la base de datos. Una vez que se creó la entrada el gestor de titular debe enviar un mensaje a la UI de titular indicando que la operación ha sido exitosa. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Task Card | | | | Logica Modificar titular | | | Programador: Cristian Maglianesi |
| Fecha: | | | | Task Points: 1 | | |  |
| Número de historia: 8 | | | |  | | |  |
| Descripción: Luego de dar click en el botón “Finalizar” de las pantallas “Renovar licencia” o “Emitir copia”, si alguno de los datos modificables (Dirección, clase solicitada o donador de órganos) se modificó, y la modificación es válida, se guardan las modificaciones de los datos. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Task Card | | | | Imprimir Ticket | | | Programador:  Reichert Maximiliano |
| Fecha: | | | | Task Points: 7 | | |  |
| Número de historia: 3 | | | |  | | |  |
| Descripción: Luego de que se emita la licencia o se renueve se llama a una función a la que se le pasa como argumento una instancia de licencia y una instancia de titular y deberá crear una factura en pdf de acuerdo al diseño de Spike Ticket. El número de factura será calculado a través de contar la cantidad de licencias en la base de datos. Una vez creado el pdf se abrirá automáticamente.  En caso de no poder crear o abrir el pdf se debe generar una excepción. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 2/11 | Cálculo del número de factura | | | Colocar en el documento número de factura, tipo de factura, datos de quien emite la factura, quien paga y concepto con su respectivo formato y realizar pruebas | |  | |
| 3/11 | Creación de las pruebas unitarias y mostrar en el documento número y tipo de factura | | | Agregar al documento datos de quien emite la factura, quien paga y concepto con su respectivo formato | |  | |
| 5/11 | Se agregó al pdf datos de quien emite la factura con su respectivo formato | | | Agregar al documento datos de quien paga y concepto con su respectivo formato | |  | |
| 9/11 | Se agregó al documento los datos de quien paga con su respectivo formato | | | Agregar al documento concepto con su respectivo formato | |  | |
| 10/11 | Se agregó al documento el concepto respectivo formato | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | Imprimir Licencia | | | Programador:  Reichert Maximiliano |
| Fecha: | | | | Task Points: 7 | | |  |
| Número de historia: 3 | | | |  | | |  |
| Descripción: Luego de que se emita la licencia o se renueve se llama a una función a la que se le pasa como argumento una instancia de licencia y una instancia de titular. Esta función generará un pdf que contendrá el diseño de la cara anterior y la cara posterior de la licencia, de acuerdo al diseño de Spike Licencia. Una vez creado el pdf se abrirá automáticamente.  En caso de no poder crear o abrir el pdf se debe generar una excepción. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 11/11 | Pruebas unitarias y agregar al documento los marcos de la parte posterior y anterior de la licencia | | | Agregar imágenes y texto de la parte posterior de la licencia e imagenes y texto de la parte anterior y darle formato al texto | |  | |
| 12/11 | Se agregó imágenes de la parte posterior y anterior | | | Agregar texto de la parte posterior y anterior de la licencia y darle formato | |  | |
| 13/11 | Se agregó texto de la parte anterior | | | Agregar texto de la parte posterior y darle formato al texto de ambas caras | |  | |
| 16/11 | Formateo del texto de la cara anterior y reajustar posición del mismo para que coincida con lo determinado por el Spike | | | Agregar texto de la parte anterior y darle formato | |  | |
| 17/11 | Se agregó el texto de la parte anterior | | | Formatear el texto de la parte anterior | |  | |
| 18/11 | Formatear texto de la parte anterior y reajustar posición para que coincida con lo determinado por el Spike | | |  | |  | |
|  |  | | |  | |  | |

Conclusiones parciales de lo estimado

En general el esfuerzo estimado para las tareas fue equivocado, nos llevaron más tiempo del estimado ya que no consideramos cuestiones extras como complicaciones laborales, exámenes parciales y trabajos prácticos de otras asignaturas. Imprimir ticket y licencia se cumplio en tiempo y forma pero esto se logró ya que previamente, a medida que se avanzaba con la planificación se fue investigando la librería a utilizar.

Planificación Iteracion 2

Tareas

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Task Card | | | | Emitir una copia UI | | | Programador:  Franco Friggeri |
| Fecha: | | | | Task Points: 5 | | |  |
| Número de historia: 5 | | | |  | | |  |
| Descripción: Después de hacer click en el botón “Emitir copia” en el menú principal, se muestra una pantalla con un recuadro para ingresar el DNI del titular que quiere realizar la copia. A continuación se muestra por pantalla las licencias que posee ese titular, permitiendo seleccionar una para luego mostrar por pantalla los datos de la misma, permitiendo modificar los mismos excepto los esenciales (dni, clase, grupo sanguíneo y factor, fecha de nacimiento).  Se informa el costo de la copia por pantalla (mismo que al emitir licencia).  Luego con el botón “Emitir Copia” se guardan los datos modificados en caso de haberlos junto con el número de copia de la licencia y se imprime la licencia. Esto debe quedar registrado como una entrada nueva en la base de datos.  Esta pantalla deberá poseer un botón “atrás” en caso de querer volver al menú principal.  Registrar el trámite. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 30/11 | Diseño completo de la interfaz.  Verificación de los campos vacíos | | |  | | Se reutilizó la interfaz de Renovar Licencia UI | |
| 1/12 |  | | |  | |  | |
| 2/12 |  | | |  | |  | |
| 3/12 |  | | |  | |  | |
| 4/12 |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | Modificar Usuario lógica y persistencia | | | Programador: Reichert Maximiliano |
| Fecha: | | | | Task Points: 1 | | |  |
| Número de historia: 8 | | | |  | | |  |
| Descripción: Cuando en la UI de modificar usuario se presiona el botón buscar se le pasa al gestor de usuario un DTO de usuario con tipo de documento y número de documento. Con estos datos se llama al DAO de usuario para que realice la búsqueda del usuario. Si la búsqueda retorna un resultado se completa el DTO de usuario y se lo retorna a la UI. Si la búsqueda retorna null, entonces el usuario no existe y se retorna un DTO de usuario vacío a la UI.  Cuando en la UI de modificar usuario se presiona el botón actualizar se debe le pasa al gestor de usuario un DTO de usuario con los datos completos. El gestor utiliza los datos de dicho DTO para crear una instancia de usuario y pasarsela al DAO de usuario para que actualice los datos. Si la DAO genera una excepción se le retorna un mensaje de error a la UI, de lo contrario se retorna un mensaje que indique que la operación ha sido exitosa. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 27/11 | Creaciones de los métodos de DAO y gestor de usuario y realización de pruebas | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Task Card | | | | Renovar licencia UI | | | Programador:  Franco Friggeri |
| Fecha: | | | | Task Points: 2 | | |  |
| Número de historia: 4 | | | |  | | |  |
| Descripción: Después de hacer click en el botón “Renovar licencia” en el menú principal, se muestra una pantalla con un recuadro para ingresar el DNI del titular que quiere renovar la licencia. A continuación se muestra por pantalla las licencias que posee y puede renovar ese titular (efectuadas las validaciones con respecto a la fecha de expiración), permitiendo seleccionar una para a continuación mostrar por pantalla los datos de la misma, permitiendo modificar los datos excepto los esenciales (dni, clase, fecha de nacimiento, grupo sanguíneo y factor).  Se informa el costo de la renovación por pantalla.  Luego con el botón “Renovar” se renueva la licencia con los datos modificados en caso de haberlos y se actualiza con la nueva fecha de expiración y se imprime la licencia. Esto debe quedar registrado como una fila nueva en la tabla licencia de la base de datos.  Esta pantalla deberá poseer un botón “atrás” en caso de querer volver al menú principal.  Registrar el trámite. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 23/11 | Diseño completo de la interfaz | | | Verificar campos vacíos | |  | |
| 24/11 | Validaciones de los campos | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | Renovar licencia lógica y persistencia. | | | Programador:  Franco Friggeri |
| Fecha: | | | | Task Points: 3 | | |  |
| Número de historia: 4 | | | |  | | |  |
| Descripción: Después de hacer click en el botón “Renovar licencia” en el menú principal, se muestra una pantalla con un recuadro para ingresar el DNI del titular que quiere renovar la licencia. A continuación se muestra por pantalla las licencias que posee y puede renovar ese titular (efectuadas las validaciones con respecto a la fecha de expiración), permitiendo seleccionar una para a continuación mostrar por pantalla los datos de la misma, permitiendo modificar los datos excepto los esenciales (dni, clase, fecha de nacimiento, grupo sanguíneo y factor).  Se informa el costo de la renovación por pantalla.  Luego con el botón “Renovar” se renueva la licencia con los datos modificados en caso de haberlos y se actualiza con la nueva fecha de expiración y se imprime la licencia.Esto debe quedar registrado como una fila nueva en la tabla licencia de la base de datos.  Esta pantalla deberá poseer un botón “atrás” en caso de querer volver al menú principal.  Registrar el trámite. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 25/11 | Obtengo el titular y las licencias asociados a ese titular. | | | Actualizar los datos del titular.  Baja lógica de la licencia vieja.  Guardar nueva licencia. | |  | |
| 26/11 | Actualizar los datos del titular.  Baja lógica de la licencia vieja. | | | Guardar nueva licencia. | |  | |
| 27/11 | Guardar nueva licencia. | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | Emitir una copia, lógica y persistencia | | | Programador:  Franco Friggeri |
| Fecha: | | | | Task Points: 5 | | |  |
| Número de historia: 5 | | | |  | | |  |
| Descripción: Después de hacer click en el botón “Emitir copia” en el menú principal, se muestra una pantalla con un recuadro para ingresar el DNI del titular que quiere realizar la copia. A continuación se muestra por pantalla las licencias que posee ese titular, permitiendo seleccionar una para luego mostrar por pantalla los datos de la misma, permitiendo modificar los mismos excepto los esenciales (dni, clase, grupo sanguíneo y factor, fecha de nacimiento).  Se informa el costo de la copia por pantalla (mismo que al emitir licencia).  Luego con el botón “Finalizar” se guardan los datos modificados en caso de haberlos junto con el número de copia de la licencia y se imprime la licencia. Esto debe quedar registrado como una entrada nueva en la base de datos.  Esta pantalla deberá poseer un botón “atrás” en caso de querer volver al menú principal.  Registrar el trámite. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 7/12 | Obtengo el titular y las licencias asociadas a ese titular.  Actualizar los datos del titular.  Actualizar los datos del titular. | | | Guardar nueva copia. | | Se reutilizo el mismo código de Renovar licencia. Solo se hizo una modificación a nivel lógica para almacenar el número de copia. | |
| 8/12 | Guardar nueva copia | | |  | |  | |
| 9/12 |  | | |  | |  | |
| 10/12 |  | | |  | |  | |
| 11/12 |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |
| --- | --- | --- | --- |
| Task Card Listado de licencias que han expirado Programador: Cesar Lencinas  Fecha: Task Points: 3  Número de historia: 6 | | | |
| Descripción: Se genera un listado por pantalla de las licencias expiradas ordenadas de menor a mayor. Se entiende que la licencia con fecha de expiración menor es la más próxima a la fecha actual del sistema. | | | |
|  | | | |
| Notas | | | |
| Fecha | Realizado | A realizar | Comentarios |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Task Card Listado de licencias vigentes por criterios Programador: Cesar Lencinas  Fecha: Task Points: 4  Número de historia: 7 | | | |
| Descripción: Se genera un listado por pantalla de las licencias vigentes ordenadas de menor a mayor. También se puede seleccionar el checkbox asociado a uno de los criterios. El listado mostrará las primeras 50 licencias, con opción de navegar hasta la última de la lista.  Criterios de ordenamiento:  Elegir un criterio de entre los siguientes:  -Nombre y apellido  -Grupo sanguíneo  -Factor RH  -Si el titular es donante de órganos  Y mostrar una lista de las licencias vigentes ordenada de acuerdo al criterio elegido | | | |
|  | | | |
| Notas | | | |
| Fecha | Realizado | A realizar | Comentarios |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Task Card | | | | Alta Usuario UI | | | Programador: Reichert Maximiliano |
| Fecha: | | | | Task Points: 1 | | |  |
| Número de historia: 9 | | | |  | | |  |
| Descripción:Luego de hacer click en el botón “Agregar Usuario” en el menú se abre un formulario en donde se requieren los siguientes datos:   * Tipo documento (ComboBox, opciones: le, lc, dni) * Nº de documento (int) * Nº de legajo (int) * Apellido (String: 15) * Nombre (String: 15) * Fecha de nacimiento (date) * Dirección (String: 30)   Una vez se oprime el botón “Agregar” se debe validar que los campos para completar no estén vacíos y que en el comboBox se haya elegido una opción. Si los campos no están vacíos se debe llamar al gestor de usuario para realizar la persistencia de dicho usuario. Si los campos están vacíos se debe mostrar un mensaje indicando que los complete.  Si el gestor de usuario indica que el usuario ya existe se debe mostrar un mensaje de error que diga “El usuario ya existe” o si se produjo una excepción se debe mostrar un mensaje que diga “Hubo un problema con la base de datos”. De lo contrario se debe mostrar un mensaje que diga “El usuario ha sido creado con éxito” | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 23/11 | Diseño de la interfaz | | | Comportamiento de los botones y validaciones de campos | |  | |
| 24/11 | Comportamiento de los botones y validaciones de campos | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Task Card | | | | Alta Usuario Lógica y Persistencia | | | Programador: Reichert Maximiliano |
| Fecha: | | | | Task Points: 2 | | |  |
| Número de historia: 9 | | | |  | | |  |
| Descripción:Cuando la UI de Alta usuario llama al gestor de usuario, le pasa como parámetro un DTO de usuario el cual el método del gestor debe convertir en una entidad usuario. Luego debe llamar al DAO de usuario para verificar si el usuario existe. Si existe se debe indicar al UI que dicho usuario ya existe. De lo contrario pedir al DAO de usuario que persista a este usuario. Si no se puede persistir por algún motivo se genera una excepción. | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 25/11 | Pruebas y realización de los métodos de gestor de usuario y dao | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | UI Modificar Usuario | | | Programador: Reichert Maximiliano |
| Fecha: | | | | Task Points: 1 | | |  |
| Número de historia: 8 | | | |  | | |  |
| Descripción:Luego de hacer click en el botón “Modificar Usuario” en el menú se abre un formulario en donde se requieren elegir tipo de documento e ingresar el número de documento   * Tipo documento (ComboBox, opciones: le, lc, dni) * Nº de documento (int)   y se muestran vacíos los campos:   * Nº de legajo (int) * Apellido (String: 15) * Nombre (String: 15) * Fecha de nacimiento (date) * Dirección (String: 30)   Luego de hacer click en el botón “buscar”, se llama al gestor de usuario para que verifique si el usuario existe. Si el gestor retorna un DTO de usuario vacío, el usuario no existe y se mostrará un mensaje que diga “El usuario no existe”. Si existe se completarán automáticamente los campos vacíos y sólo se permitirá modificar Apellido, Nombre y Dirección y se debe habilitar el botón “Actualizar”  Cuando se presiona el botón “Actualizar” se debe validar que los campos modificables no estén vacíos. Si dichos campos están vacíos se debe mostrar un mensaje que diga “Complete todos los campos”. De lo contrario se debe llamar al gestor de usuario para que actualicen los datos. Si se produce una excepción se debe mostrar un mensaje que diga “Hubo un problema con la base de datos”. Si no hubo ningún problema se debe mostrar un mensaje que diga “El usuario ha sido actualizado con éxito” | | | | | | | |
| Notas | | | | | | | |
| Fecha | Realizado | | | A realizar | | Comentarios | |
| 26/11 | Diseño de la pantalla y validaciones de campo | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |
|  |  | | |  | |  | |

Conclusiones parciales de lo estimado

Al igual que en el punto anterior la mayoría de los fallos en la estimación se debió a la no contemplación de complicaciones laborales y responsabilidades con otras asignaturas. En el caso de renovar licencia llevo un poco más del tiempo estimado pero emitir copia llevo mucho menos porque en el desarrollo nos dimos cuenta que podíamos reutilizar mucho código de renovar licencia y en un tiempo mucho menor del pensado estaba listo.

En alta usuario nos llevó más tiempo la interfaz que la lógica, lo contrario a lo planteado en las estimaciones y la modificación del usuario si se llevó en el tiempo estimado porque se pudo reutilizar código de alta usuario.

Hemos errado en casi todas las estimaciones que planteamos en la planificación ya sea porque demoramos más tiempo o mucho menos.

Código Fuente (debe estar ordenado por iteración y por tarea - debe estar comentado)

**Iteración 1**

Calcular Vigencia

Pruebas de Unidad (deben especificarse a que tarea corresponden)

Refactorización (deben especificarse a que tarea corresponden)

INTEGRACIÖN CONTINUA - ESTANDARES

Se estableció como estándar de programación que los nombres de las constantes vayan con mayúscula, los nombres de los métodos y las variables comienzan con mayúscula y los nombres de las clases comienzan con mayúscula. Además en los nombres compuestos la primera letra de las palabras siguientes a la primera deben ser mayúscula.

Para la integración utilizamos un repositorio git en donde determinamos que debíamos separarnos en ramas y hacer un commit cada 3 horas y un push a la rama una vez al día. Cada vez que se terminaba una historia el dueño de la rama era responsable de integrar los cambios a la rama principal.

Conclusiones (deben ser exahustivas respecto al trabajo realizado y la metodología, criticando a favor y en contra de la misma en base a la experienciaadquirida)