**Documento de requerimientos de software**

***[Desarrollo plataforma de gestión de reemplazo para profesores de Duoc]***

***Fecha: [dd/mm/aaa]***

**Tabla de contenido**

Historial de Versiones 3

Información del Proyecto 3

Aprobaciones 3

1. Propósito 4

2. Alcance del producto / Software 4

3. Referencias 4

4. Funcionalidades del producto 5

5. Clases y características de usuarios 5

6. Entorno operativo 5

7. Requerimientos funcionales 6

9.1. (Nombre de la funcionalidad 1) 6

9.2. (Nombre de la funcionalidad 2) 7

9.3. (Nombre de la funcionalidad N) 7

8. Reglas de negocio 8

9. Requerimientos de interfaces externas 9

9.1. Interfaces de usuario 9

9.2. Interfaces de hardware 9

9.3. Interfaces de software 9

9.4. Interfaces de comunicación 9

10. Requerimientos no funcionales 10

11. Otros requerimientos 11

12. Glosario 12

**Historial de Versiones**

| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Información del Proyecto**

| Empresa / Organización |  |
| --- | --- |
| Proyecto | Desarrollo plataforma de gestión de reemplazo para profesores. |
| Fecha de preparación |  |
| Cliente | Rodrigo Meza |
| Patrocinador principal | Ricardo Badilla |
| Gerente / Líder de Proyecto |  |
| Gerente / Líder de Análisis de negocio y requerimientos |  |

**Aprobaciones**

| **Nombre y Apellido** | **Cargo** | **Departamento u Organización** | **Fecha** | **Firma** |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

1. **Propósito**

Desarrollo de plataforma de gestión de reemplazo para profesores de Duoc del area de ingles

Luego se describe cuales componentes o partes del alcance del producto están incluidas en el documento, estableciendo si este documento cubre la totalidad del software, sólo una parte del sistema, subsistema o subgrupo de procesos.

Desarrollar una plataforma web de gestión de reemplazo para profesores de Duoc

1. **Alcance del producto / Software**

Objetivo general: Desarrollar una solución tecnológica integral que facilite la gestión de reemplazos de docentes y el control de horas trabajadas para asegurar una administración eficiente y precisa en el Programa de Inglés de Duoc UC, mejorando la calidad educativa y la precisión en el proceso de pago.

Beneficios: Reducción de tiempo en operaciones al momento de gestionar los reemplazos y recuperación de los docentes del área de inglés en Duoc UC.

1. **Referencias**

1. **Funcionalidades del producto**
2. Creación de perfil de docente
3. Modificar perfiles de docentes
4. Eliminar perfil de docente
5. Agregar horario
6. Modificar horario
7. Eliminar horario
8. Asignar clase a reemplazar
9. Modificar clase a reemplazar
10. Eliminar clase a reemplazar
11. Calculo de horas de cada docente reemplazante
12. Clasificación de horas
13. Contador de bonos
14. Formatos de exportación y descarga
15. Visualización de tablas individuales y generales
16. Registrar clases para recuperación
17. Modificar clases de recuperación
18. Eliminar clases de recuperación
19. Agregar Licencia
20. Modificar Licencia
21. Eliminar Licencia

1. **Clases y características de usuarios**

Existen 2 usuarios que manejan el sistema:

Administrador: Puede hacer todo lo que abarque la plataforma.

Usuario: Solo puede visualizar la plataforma web

1. **Entorno operativo**

En esta sección se describe el entorno operativo en el que se desenvolverá el sistema, software, módulo o grupo de funcionalidades, mencionando aspectos como la plataforma de hardware, versiones de sistema operativo y otros sistemas o componentes con los que debe coexistir.

Se utilizo visual studio code para programar

django 4.2

mysqlclient

python-decouple

django rest framework

Se utiliza dbeaver para gestionar la Base de datos:

MariaDB

Se utiliza GitHub y una extensión de visual llamada LiveShare para el trabajo colaborativo.

1. **Requerimientos funcionales**

| [R-N°] | [Nombre del Requerimiento] | Tipo Requerimiento | Actores Relacionados | [Descripción corta del requerimiento] |
| --- | --- | --- | --- | --- |
| R.1 | Creación de perfil de docente | Funcional | Administrador | Administrador de la página podrá crear perfiles de los profesores con sus horarios y sus características |
| R.2 | Modificar perfiles de docentes | Funcional | Administrador | El administrador podrá modificar los perfiles de los docentes |
| R.3 | Eliminar perfil de docente | Funcional | Administrador | El administrador podrá eliminar un perfil de un docente |
| R.4 | Agregar horario | Funcional | Administrador | El administrador podrá agregar una horario para un profesor |
| R.5 | Modificar horario | Funcional | Administrador | El administrador podrá modificar una horario de un profesor |
| R.6 | Eliminar horario | Funcional | Administrador | El administrador podrá eliminar una horario de un profesor |
| R.7 | Asignar clase a reemplazar | Funcional | Administrador | Con seleccionar la sección se auto completará la jornada de la clase, la sala y el profesor encargado |
| R.8 | Modificar clase a reemplazar | Funcional | Administrador | El administrador podrá modificar la clase a reemplazar para el docente |
| R.9 | Eliminar clase a reemplazar | Funcional | Administrador | El administrador podrá eliminar la clase a reemplazar para el docente |
| R.10 | Calculo de horas de cada docente reemplazante | Funcional | Sistema | Se realizará el cálculo de horas de las jornadas por contrato del profesor + las jornadas de reemplazo realizadas para posteriormente clasificar las horas según tipo de pago |
| R.11 | Clasificación de horas | Funcional | Sistema | El sistema debe clasificar las horas trabajadas como reemplazo para determinar si deben ser pagadas como horas de programación o si corresponden a un bono.  Esta clasificación debe basarse en reglas predefinidas, como la cantidad de horas semanales del docente la cual debe de superar las 40 horas para optar a un bono. |
| R.12 | Contador de bonos | Funcional | Sistema | El sistema debe llevar un contador acumulativo de los bonos que ha recibido cada docente por horas reemplazadas esto seria un registro mensual.(40+ horas = +1 bono) |
| R.13 | Formatos de exportación y descarga | Funcional | Usuario | El sistema debe permitir la descarga de reportes en formato PDF. |
| R.14 | Visualización de tablas individuales y generales | Funcional | Usuario/Administrador | El sistema debe generar tablas que permitan visualizar la cantidad de horas y módulos de reemplazo por docente, tanto a nivel individual como a nivel global (todos los docentes). |
| R.15 | Registrar clases para recuperación | Funcional | Administrador | El administrador debe registrar cuándo un docente ha recuperado una clase perdida, incluyendo detalles como: hora inicio, hora término,jornada, sección, nro. módulos, fecha y horas recuperadas. |
| R.16 | Modificar clases de recuperación | Funcional | Administrador | El administrador podrá modificar las clases de recuperación |
| R.17 | Eliminar clases de recuperación | Funcional | Administrador | El administrador podrá eliminar las clases de recuperación |
| R.18 | Agregar Licencia | Funcional | Administrador | El administrador podrá agregar una licencia y vincularla a un docente. |
| R.19 | Modificar Licencia | Funcional | Administrador | El administrador podrá modificar una licencia. |
| R.20 | Eliminar Licencia | Funcional | Administrador | El administrador podrá eliminar una licencia. |

1. **Creación de perfil de docente**

**Descripción: El administrador podrá crear perfiles de los docentes en el sistema, incluyendo información como nombre, horarios de trabajo, asignaturas y características específicas del docente.**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador ingresa al sistema y selecciona la opción "Crear perfil de docente".**
2. **El sistema presenta un formulario donde el administrador puede ingresar datos del docente (nombres y apellidos).**
3. **El administrador completa el formulario y confirma la creación del perfil.**
4. **El sistema valida los datos ingresados y, si son correctos, crea el perfil del docente en el sistema.**
5. **Si hay algún error o datos inválidos, el sistema muestra un mensaje de error específico (por ejemplo, "El campo nombre o apellido no puede estar vacío").**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir al administrador crear un perfil para cada docente con información básica como nombre completo.**
* **REQ-2: El sistema debe validar que todos los campos requeridos en el formulario (nombre, apellido) estén correctamente completados antes de permitir la creación del perfil.**
* **REQ-3: El sistema debe notificar al administrador si se ingresan datos inválidos (por ejemplo, caracteres no permitidos en el nombre, etc.) y permitir corregir la entrada antes de proceder.**

1. **Modificar perfiles de docentes**

**Descripción: El administrador podrá modificar los perfiles existentes de los docentes.**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona el perfil del docente que desea modificar.**
2. **El sistema presenta el formulario con los datos actuales del docente.**
3. **El administrador realiza los cambios necesarios y confirma la actualización.**
4. **El sistema valida los datos y actualiza el perfil.**
5. **En caso de error, el sistema muestra un mensaje de error indicando el problema (por ejemplo, "El campo 'nombre' no puede estar vacío").**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir al administrador modificar cualquier campo del perfil de un docente.**
* **REQ-2: El sistema debe validar que los campos modificados tengan valores válidos antes de proceder con la actualización.**
* **REQ-3: El sistema debe notificar al administrador cuando se ha realizado una actualización exitosa.**

1. **Eliminar perfil de docente**

**El administrador podrá eliminar un perfil de docente, eliminando todos los registros relacionados.**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona el perfil del docente a eliminar.**
2. **El sistema solicita confirmación antes de proceder con la eliminación.**
3. **El administrador confirma la eliminación.**
4. **El sistema elimina el perfil y muestra un mensaje de éxito.**
5. **Si no se puede eliminar el perfil (por ejemplo, si hay dependencias o registros asociados), el sistema muestra un mensaje de error.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir al administrador eliminar un perfil de docente con confirmación previa.**
* **REQ-2: El sistema debe verificar que no haya registros dependientes antes de permitir la eliminación.**
* **REQ-3: El sistema debe notificar al administrador si la eliminación fue exitosa o si ocurrió un error.**

1. **Agregar horario**

**Descripción: El administrador podrá agregar horarios de trabajo para los docentes, especificando días y horas.**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador podrá agregar un horario a la vez que ingresa el docente, para que el horario asociado al docente.**
2. **El sistema presenta un formulario para ingresar los horarios.**
3. **El administrador ingresa la información y confirma la adición.**
4. **El sistema valida la entrada y agrega el horario.**
5. **El sistema muestra un mensaje de confirmación de la adición exitosa.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir al administrador ingresar y guardar horarios para los docentes.**
* **REQ-2: El sistema debe notificar al administrador si el horario se ha agregado correctamente o si hubo un error.**

1. **Modificar horario**

**Descripción: El administrador podrá modificar los horarios de los docentes.**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona el horario que desea modificar.**
2. **El sistema muestra los detalles del horario actual.**
3. **El administrador edita los campos de horario y confirma la modificación.**
4. **El sistema valida la modificación y actualiza los datos.**
5. **El sistema muestra un mensaje confirmando la modificación exitosa.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir al administrador modificar cualquier campo de los horarios registrados.**
* **REQ-2: El sistema debe validar la no interferencia de nuevos horarios con los existentes.**
* **REQ-3: El sistema debe notificar al administrador cuando la modificación sea exitosa o si hay errores.**

1. **Eliminar horario**

**Descripción: El administrador podrá eliminar horarios previamente asignados a un docente.**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona el horario a eliminar.**
2. **El sistema solicita confirmación antes de proceder con la eliminación.**
3. **El administrador confirma la eliminación.**
4. **El sistema elimina el horario y muestra un mensaje de éxito.**
5. **Si no es posible eliminar el horario (por ejemplo, si hay conflictos con otros horarios), el sistema muestra un mensaje de error.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir al administrador eliminar horarios con confirmación previa.**
* **REQ-2: El sistema debe verificar que no haya conflictos o dependencias antes de permitir la eliminación.**
* **REQ-3: El sistema debe notificar al administrador si la eliminación fue exitosa o si ocurrió un error.**

1. **Asignar clase a reemplazar**

**Descripción: El administrador podrá asignar una clase a reemplazar, completando automáticamente la información relacionada (jornada, sala, docente encargado).**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona la clase a reemplazar y la sección correspondiente.**
2. **El sistema completa automáticamente los detalles de jornada, sala y profesor encargado.**
3. **El administrador revisa la información y confirma la asignación.**
4. **El sistema registra la asignación y muestra un mensaje de confirmación.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir asignar automáticamente jornada, sala y docente a reemplazo basado en la selección de la sección.**
* **REQ-2: El sistema debe validar la disponibilidad del docente y la sala antes de confirmar la asignación.**
* **REQ-3: El sistema debe notificar al administrador sobre la asignación exitosa o cualquier error.**

1. **Modificar clase a reemplazar**

**Descripción: El administrador podrá modificar la clase asignada para ser reemplazada, cambiando detalles como la sección, sala o docente encargado.**

**Prioridad: Media**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona la clase que desea modificar.**
2. **El sistema muestra los detalles de la clase asignada para reemplazo.**
3. **El administrador realiza los cambios necesarios y confirma la modificación.**
4. **El sistema valida y actualiza los detalles de la clase.**
5. **El sistema notifica al administrador que la clase ha sido modificada exitosamente.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir modificar las clases asignadas para reemplazo.**
* **REQ-2: El sistema debe validar que los cambios no generen conflictos en horarios o disponibilidad.**
* **REQ-3: El sistema debe notificar al administrador cuando la modificación sea exitosa.**

1. **Eliminar clase a reemplazar**

**Descripción: El administrador podrá eliminar la asignación de clase a reemplazar para un docente.**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona la clase a eliminar.**
2. **El sistema solicita confirmación antes de proceder con la eliminación.**
3. **El administrador confirma la eliminación.**
4. **El sistema elimina la asignación de clase y notifica la eliminación exitosa.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir eliminar una clase asignada a reemplazo con confirmación previa.**
* **REQ-2: El sistema debe verificar que no haya dependencias o conflictos antes de eliminar la clase.**
* **REQ-3: El sistema debe notificar al administrador que la clase ha sido eliminada.**

1. **Cálculo de horas de cada docente reemplazante**

**Descripción: El sistema calculará automáticamente las horas trabajadas por los docentes reemplazantes, sumando las horas de las jornadas por contrato y las horas de reemplazo realizadas.**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El sistema recopila los datos de las jornadas por contrato y los reemplazos realizados por cada docente.**
2. **El sistema calcula las horas trabajadas en función de las jornadas normales y las horas de reemplazo.**
3. **El sistema genera un reporte con el cálculo total de horas trabajadas.**
4. **El sistema muestra un mensaje de éxito o error dependiendo del resultado del cálculo.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe calcular automáticamente las horas trabajadas de cada docente, combinando las horas de contrato y las de reemplazo.**
* **REQ-2: El sistema debe permitir la actualización de las horas trabajadas cada vez que se ingresa un reemplazo nuevo.**
* **REQ-3: El sistema debe notificar al administrador si hay un error en el cálculo de las horas, como horas duplicadas o no registradas.**

1. **Clasificación de horas**

**Descripción: El sistema debe clasificar las horas trabajadas por los docentes en dos categorías: horas de programación y horas de reemplazo, según reglas predefinidas.**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El sistema recopila las horas trabajadas por el docente.**
2. **El sistema clasifica las horas como horas de programación o de reemplazo en función de las reglas predefinidas (por ejemplo, si un docente supera las 40 horas semanales, las horas adicionales se clasifican como bono).**
3. **El sistema genera un reporte con la clasificación de las horas.**
4. **El sistema notifica al administrador si hubo un error en la clasificación de las horas.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir la clasificación automática de horas trabajadas como horas de programación o reemplazo.**
* **REQ-2: El sistema debe seguir las reglas predefinidas para la clasificación de horas (por ejemplo, bono por horas extras).**
* **REQ-3: El sistema debe permitir la verificación manual y la corrección de la clasificación si es necesario.**

1. **Contador de bonos**

**Descripción: El sistema llevará un contador acumulativo de los bonos otorgados a cada docente por horas de reemplazo trabajadas.  
Prioridad: Alta  
Acciones iniciadoras y comportamiento esperado:**

1. **El sistema recopila las horas de reemplazo trabajadas por cada docente.**
2. **El sistema calcula si el docente ha superado el umbral de horas para recibir un bono (por ejemplo, más de 40 horas semanales).**
3. **El sistema incrementa el contador de bonos cada vez que el docente supera el umbral.**
4. **El sistema genera un reporte con el número total de bonos otorgados a cada docente.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe realizar el seguimiento de los bonos otorgados a cada docente.**
* **REQ-2: El sistema debe actualizar el contador de bonos automáticamente con base en las horas de reemplazo trabajadas.**
* **REQ-3: El sistema debe permitir ver el historial de bonos acumulados por cada docente.**

1. **Formatos de exportación y descarga**

**Descripción: El sistema permitirá a los usuarios descargar reportes en formato PDF para su consulta o impresión.**

**Prioridad: Media**

**Acciones iniciadoras y comportamiento esperado:**

1. **El usuario selecciona el reporte que desea descargar.**
2. **El sistema ofrece la opción de exportar el reporte en formato PDF.**
3. **El usuario confirma la descarga.**
4. **El sistema genera y descarga el archivo PDF con el reporte solicitado.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir exportar los reportes generados en formato PDF.**
* **REQ-2: El sistema debe asegurarse de que el PDF descargado contenga toda la información relevante y esté correctamente formateado.**
* **REQ-3: El sistema debe notificar al usuario si la descarga del archivo PDF fue exitosa.**

1. **Visualización de tablas individuales y generales**

**Descripción: El sistema generará tablas que permiten visualizar la cantidad de horas y módulos de reemplazo por docente, tanto a nivel individual como global.**

**Prioridad: Alta**

**Acciones iniciadoras y comportamiento esperado:**

1. **El usuario selecciona la opción para ver las tablas de horas y módulos de reemplazo.**
2. **El sistema genera una vista con la información de horas trabajadas y módulos reemplazados, filtrada por docente o para todos los docentes.**
3. **El usuario puede consultar o exportar la información.**
4. **El sistema muestra un mensaje de error si no se pueden generar las tablas debido a problemas con los datos.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir ver tablas detalladas de horas trabajadas y módulos de reemplazo para cada docente.**
* **REQ-2: El sistema debe generar tablas tanto a nivel individual como global.**
* **REQ-3: El sistema debe permitir exportar la tabla en formatos descargables, como PDF o Excel.**

1. **Registrar clases para recuperación**

**Descripción: El administrador podrá registrar las clases que un docente ha recuperado, especificando detalles como la hora, jornada, sección y horas recuperadas.**

**Prioridad: Media**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona la opción "Registrar clase de recuperación".**
2. **El sistema presenta un formulario para ingresar los detalles de la clase recuperada.**
3. **El administrador ingresa la información y confirma la acción.**
4. **El sistema registra la clase de recuperación y muestra un mensaje de confirmación.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir al administrador registrar detalles completos de una clase de recuperación (hora de inicio, hora de término, jornada, sección, módulos).**
* **REQ-2: El sistema debe verificar que no haya solapamiento con otras clases al registrar la clase de recuperación.**
* **REQ-3: El sistema debe notificar al administrador si la clase fue registrada correctamente o si hubo un error.**

1. **Modificar clases de recuperación**

**Descripción: El administrador podrá modificar los detalles de las clases de recuperación registradas.**

**Prioridad: Media**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona la clase de recuperación que desea modificar.**
2. **El sistema muestra los detalles actuales de la clase.**
3. **El administrador realiza los cambios necesarios y confirma la modificación.**
4. **El sistema valida los cambios y actualiza la información.**
5. **El sistema muestra un mensaje de éxito o error según el resultado.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir modificar todos los detalles de una clase de recuperación registrada.**
* **REQ-2: El sistema debe validar que no haya conflictos con otras clases al modificar la clase de recuperación.**
* **REQ-3: El sistema debe notificar al administrador cuando la clase de recuperación haya sido modificada correctamente.**

1. **Eliminar clases de recuperación**

**Descripción: El administrador podrá eliminar una clase de recuperación registrada.  
Prioridad: Media  
Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona la clase de recuperación que desea eliminar.**
2. **El sistema solicita confirmación para eliminar la clase de recuperación.**
3. **El administrador confirma la eliminación.**
4. **El sistema elimina la clase de recuperación y muestra un mensaje de confirmación.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir eliminar clases de recuperación con confirmación previa.**
* **REQ-2: El sistema debe verificar que no haya dependencias o registros asociados a la clase antes de eliminarla.**
* **REQ-3: El sistema debe notificar al administrador si la eliminación fue exitosa o si ocurrió algún error.**

1. **Agregar Licencia**

**Descripción: El administrador podrá agregar una licencia y vincularla a un docente, especificando la duración y tipo de licencia.  
Prioridad: Alta  
Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona la opción para agregar una licencia para un docente.**
2. **El sistema presenta un formulario para ingresar detalles de la licencia.**
3. **El administrador ingresa los datos y confirma la adición.**
4. **El sistema valida la información y agrega la licencia al docente.**
5. **El sistema muestra un mensaje de éxito o error, según corresponda.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir agregar licencias a los docentes.**
* **REQ-2: El sistema debe verificar que los datos de la licencia sean válidos antes de guardarlos.**
* **REQ-3: El sistema debe notificar al administrador si la licencia fue agregada correctamente.**

1. **Modificar Licencia**

**Descripción: El administrador podrá modificar los detalles de una licencia previamente agregada para un docente.**

**Prioridad: Media**

**Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona la licencia que desea modificar.**
2. **El sistema muestra los detalles actuales de la licencia.**
3. **El administrador realiza los cambios necesarios y confirma la modificación.**
4. **El sistema valida y actualiza los detalles de la licencia.**
5. **El sistema muestra un mensaje de éxito o error según el resultado.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir modificar las licencias previamente registradas para los docentes.**
* **REQ-2: El sistema debe validar que los cambios realizados sean correctos y no generen conflictos.**
* **REQ-3: El sistema debe notificar al administrador cuando la modificación de la licencia haya sido exitosa.**

1. **Eliminar Licencia**

**Descripción: El administrador podrá eliminar una licencia vinculada a un docente.  
Prioridad: Media  
Acciones iniciadoras y comportamiento esperado:**

1. **El administrador selecciona la licencia que desea eliminar.**
2. **El sistema solicita confirmación para proceder con la eliminación.**
3. **El administrador confirma la eliminación.**
4. **El sistema elimina la licencia y muestra un mensaje de éxito o error.**

**Requerimientos funcionales:**

* **REQ-1: El sistema debe permitir eliminar licencias con confirmación previa.**
* **REQ-2: El sistema debe verificar que no haya dependencias o conflictos antes de permitir la eliminación.**
* **REQ-3: El sistema debe notificar al administrador si la eliminación fue exitosa o si ocurrió un error.**

1. **Reglas de negocio**

Listado de reglas y principios que aplican a todo el conjunto de requerimientos de software contenidos en el documento. Un ejemplo es cuales individuos o roles pueden desempeñar cierta función bajo ciertas circunstancias.

Para hacer cumplir las reglas de negocio, podría ser necesaria la definición de requerimientos funcionales que aplican a todo el sistema, no a una funcionalidad especifica.

Reglas de negocio: Solo el administrador puede usar y ver lo botones de agregar, modificar o eliminar

Solo administrador puede ver la página de docente

1. **Requerimientos de interfaces externas**
2. **Interfaces de usuario**

* Accesible desde cualquier navegador web (Chrome, Firefox, Edge, Opera, Brave, etc).
* Implementadas usando JavaScript, HTML, CSS .

1. **Interfaces de hardware**

Computador con acceso a internet.

Mouse de computadora.

Teclado.

1. **Interfaces de software**

Por el lado del servidor: (falta)

Por el lado del cliente: Corresponde a cualquier sistema operativo que sea capaz de conectarse a internet mediante un navegador web que admita JavaScript y HTML.

1. **Interfaces de comunicación**

(no aplica)

1. **Requerimientos no funcionales**

Los [requerimientos no funcionales](http://www.pmoinformatica.com/2013/01/requerimientos-no-funcionales-porque.html) son los que especifican criterios para evaluar la operación de un servicio de tecnología de información, en contraste con los requerimientos funcionales que especifican los comportamientos específicos.

Para ver algunos ejemplos de cómo se redactan los requerimientos no funcionales, te recomendamos el siguiente enlace:

> [Ejemplos de requerimientos no funcionales de software](http://www.pmoinformatica.com/2015/05/requerimientos-no-funcionales-ejemplos.html)

1. **Otros requerimientos**

Requerimientos no cubiertos en ninguna otra sección del documento de requerimientos de software, por ejemplo: Requerimientos de bases de datos, internacionalización, legales y objetivos de reúso de componentes de software.

1. **Glosario**

Descripción de términos y siglas necesarias para el entendimiento del documento de requerimientos de software.