

Taller  de Ingeniería de Software

Profesores: Raimundo Vega, Cristian Rojas

Ingeniería Civil Informática.

Universidad Austral de Chile

**Avance Análisis de Caso de Uso**

**Integrantes:**

Álvaro Arriagada.

                                                                          Felipe Oyarzun Felipe Rosso.

**Fecha**: 30 de Septiembre de 2015.

1. **Introducción:**

En este documento se presentará el desarrollo de los distintos análisis de casos de uso que estamos realizando con todos los Requerimientos funcionales que obtuvimos a partir de la entrevista con nuestro cliente de la Unidad de integración estudiantil de la escuela San Andres de Tegualda.

1. **Casos de uso**

# 2.1 Requisito funcional: Buscador

**Caso Expandido de uso:**

**Caso de uso:** realizar una búsqueda en la base de datos y obtener información según el filtro.

**Actores:** Usuario, Aplicación web, SGBD.

**Tipo**: Primario, Real.

**Descripción:** un usuario registrado (logged-in) en la aplicación, selecciona el sistema de búsqueda ofrecido por la UI, luego, ingresara algún tipo de filtro (alumno, sesión, profesional) mutuamente excluyentes entre sí, donde luego la UI indicará que ingrese alguna frase de búsqueda según el tipo de filtro selecciona, una vez ingresada la frase, el usuario recibe de vuelta resultados relacionados con su búsqueda dentro de la BD.

**Referencias Cruzadas:**

**Curso normal de los eventos:**

|  |  |
| --- | --- |
| Acción del actor | Respuesta del sistema |
| Este caso de uso comienza cuando el usuario selecciona uno de los filtros. |  |
|  | 2. Detecta que un filtro a sido seleccionado |
|  | 3. Imprime un mensaje especificando los tipos de datos a ingresar para realizar la búsqueda en el espacio designado por la UI. |
| 4.  El usuario lee el mensaje y ingresa alguno de los datos indicados por el sistema. |  |
|  | 5. verifica que el usuario haya ingresado información |
|  | 6.  busca dentro de la base de datos información relacionada con la frase ingresada, |
|  | 7. Ordena los resultados devueltos, y los imprime en el espacio designado por la UI. |
| 8. El Usuario queda satisfecho con su búsqueda. |  |

**Curso alternativo de los eventos:**

|  |  |
| --- | --- |
| **Acción del actor** | **Respuesta del sistema** |
| Este caso de uso comienza cuando el Usuario ingresa una determinada frase después de seleccionar el filtro. |  |
|  | 2. Verifica que el usuario haya ingresado información |
|  | 3. La información ingresada no se encuentra en la BD |
|  | 4. Avisa al usuario  traves de la UI que la información ingresada no tiene datos. |

# 2.2 Requisito funcional: Agregar Usuario.

**Caso Expandido de uso:**

**Caso de uso:** Agregar un usuario al sistema, con todos los datos que lo identifiquen.

**Actores:** Usuario, Aplicación web, SGBD.

**Descripción:** El encargado de ingresar nuevos usuarios, ingresa los datos de éste en el sistema, con sus atributos tales como: Nombre, Apellido, Rut, código de registro en Mineduc.

**Tipo:** Primario- real.

**Referencias Cruzadas:**

**Curso normal de los eventos:**

|  |  |
| --- | --- |
| Acción del actor | Respuesta del sistema |
| 1. El usuario encargado inicia sesión en el programa con su nombre de usuario y contraseña. |  |
|  | 2. El sistema detecta que el usuario ingresa. |
| 3. El usuario encargado selecciona la opción de ingresar un nuevo usuario. |  |
|  | 4. El sistema le pide ingresar los datos (nombre, apellido, rut y número de registro en Mineduc). |
| 5. El usuario encargado ingresa los datos en el sistema. |  |
|  | 6. Registra todos los datos en la base de datos. |
| 7. El usuario queda satisfecho con el registro. |  |

**Curso Alternativos de los eventos:**

|  |  |
| --- | --- |
| Acción del actor | Respuesta del sistema |
| 1. El usuario que quiere ingresar los datos no es el usuario encargado. |  |
|  | 1. El sistema detecta que el usuario no es el encargado de ingresar nuevos usuarios. |
|  | 333333asd   1. El sistema despliega un mensaje diciendo que no puede ingresar nuevos usuarios. |

**Curso Alternativos de los eventos:**

|  |  |
| --- | --- |
| Acción del actor | Respuesta del sistema |
|  | 4. El sistema le pide ingresar los datos (nombre, apellido, rut y número de registro en Mineduc). |
| 515555   1. El usuario encargado no ingresa todos los datos solicitados por el sistema. |  |
|  | 333333asd  6.El sistema pide al usuario ingresar todos los datos solicitados nuevamente. |

**Contrato**

**Nombre**: agregarUsuario()

**Responsabilidad**: Crear un nuevo usuario en el sistema, llama la función del sistema crearFichaUsuario().

**Precondición:**

**Poscondición**: Se crea el objeto usuario por el objeto sistema y luego se asocia usuario al sistema

**Contrato:** IngresarDatosUsuario ( )

**Responsabilidad:** Registrar los datos del usuario en el formulario de registro.

**Precondición:** El objeto usuario debe haber sido instanciado, debe haber iniciado sesión administrador

**Poscondición:** Se modifican los campos del objeto usuario.

# 2.3 Requisito funcional: Mostrar la lista

**Caso Expandido de uso:**

**Caso de uso:** Expone una lista de sesiones recientes mostrando nombre de alumno y nombre de profesional.

**Actores:** Usuario, Aplicación web, SGBD.

**Tipo:** Primario, Real.

**Descripción**: El usuario registrado en la aplicación (logged-in) inicia sesión y aparece desplegada una lista de sesiones recientes en la interfaz indicando el nombre del alumno que fue atendido y el profesional que lo atendió.

**Referencias cruzadas:**

**Curso normal de los eventos:**

|  |  |
| --- | --- |
| Acción del actor | Respuesta del sistema |
| 1. El usuario inicia sesión en el programa con su nombre de usuario y contraseña. |  |
|  | 2. El sistema detecta que el usuario ingresa. |
|  | 3. Utiliza el Buscador para buscar las sesiones recientes. |
|  | 4. Ordena los resultados de las sesiones. |
|  | 5. Despliega las sesiones recientes indicando nombre profesional y alumno atendido. |
| 6. El usuario queda satisfecho con la información. |  |

**Curso alternativo de los eventos:**

|  |  |
| --- | --- |
| **Acción del actor** | **Respuesta del sistema** |
| 1. El usuario inicia sesión con su login y contraseña. |  |
|  | 2. El Sistema verifica los datos del usuario. |
|  | 3. El sistema no tiene sesiones para mostrar. |
|  | 4. Muestra un mensaje diciendo que no hay sesiones para mostrar. |

# 2.4 Requisito funcional: Agregar Alumno.

**Caso de uso:** Agregar los alumnos al sistema.

**Actores:** Usuario, Aplicación web, SGBD.

**Descripción:** El usuario ingresa alumnos al sistema utilizando la UI, para registrar en la base de datos todos los atributos y las fichas clínicas con todos los campos de información.

**Tipo:** Primario- real.

**Curso normal de los eventos:**

**Curso normal de los eventos:**

|  |  |
| --- | --- |
| Acción del actor | Respuesta del sistema |
| 1. El usuario inicia sesión con sus datos de profesional |  |
|  | 2. El sistema detecta los datos ingresados por el usuario. |
| 3. El usuario ingresa un nuevo alumno, con el fin de hacer un registro de ficha clínica del alumno. |  |
|  | 4. El sistema despliega todos los campos de información a rellenar para completar la ficha clínica del alumno. |
| 5. El usuario ingresa datos en los campos de información para rellenar la ficha clínica del alumno |  |
|  | 1. Registra todos los datos del alumno en los campos de ficha clínica. |
| 7. El usuario queda satisfecho con el registro. |  |

**Curso Alternativos de los eventos:**

|  |  |
| --- | --- |
| Acción del actor | Respuesta del sistema |
| 1. El alumno ningún dato para completar su ficha clínica. |  |
|  | 1. El sistema detecta esa falencia. |
|  | 333333asd   1. El sistema despliega un mensaje diciendo que los campos de información están incompletos y no se puede guardar una ficha clínica con tan poca información. |

**Contrato**

**Nombre**: crearPaciente ()

**Responsabilidad**: Crear un nuevo paciente en el sistema, llama la función del sistema crearFichaPaciente ().

**Precondición:**

**Poscondición**: Se crea el objeto paciente por el objeto sistema y luego se asocia paciente al sistema

**Contrato:** IngresarDatos ( )

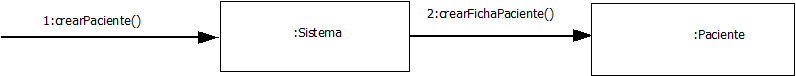
**Responsabilidad:** Registrar los datos del paciente en el formulario de registro.

**Precondición:** El objeto paciente debe haber sido instanciado.

**Poscondición:** Se modifican los campos del objeto Paciente.

**Diagrama de colaboración:**

**Contrato:** crearPaciente()



**Contrato:** IngresarDatos()

