Bitácora Web para el seguimiento y control de los trabajos de titulación del DISC (ERS)

Versión 1.0

# Tipos de Usuarios y Roles

| **Rol** | **Descripción** |
| --- | --- |
| Administrador | Corresponde al administrador general del sistema. Su cuenta debe existir desde el momento de la entrega del sistema. Tiene acceso a todas las funcionalidades del sistema. |
| Estudiante Tesista | Estudiante que está desarrollando su trabajo de titulación. Utiliza la bitácora para reportar su avance semanal |
| Profesor Guía | Tutor de un trabajo de titulación. Un profesor puede guiar varios trabajos de titulación simultáneamente. Utiliza la bitácora para controlar el avance de los trabajos que guía. |
| Secretaria | Secretaria del DISC que se encarga de la parte administrativa del proceso. Es quien registra los trabajos de titulación. |
| Encargado de titulación | Es el académico que está a cargo de coordinar los procesos asociados a la titulación de los estudiantes. |

# Historias de Usuario

|  |  |  |
| --- | --- | --- |
| **HISTORIA DE USUARIO** | | |
| **Id:** ADM-001 | **Usuario:** Administrador | |
| **Nombre Historia:** Administrar usuarios | | |
| **Prioridad en Negocio:** 3 | | **Riesgo de Desarrollo:** |
| **Puntos Estimados:** | | **Iteración Asignada:** |
| **Programador responsable:** | | |
| **Descripción:**  El sistema debe permitir que el administrador cree/actualice/elimine nuevos usuarios del sistema. Al crear un usuario se debe ingresar el correo electrónico y una clave de ingreso y el rol.  Los roles que se pueden asociar son: Estudiante, Profesor, Secretaria, Encargado Titulación | | |
| **Validación:**   * Los estudiantes y secretaria pueden tener asociado un único rol. * Sólo puede existir un encargado de titulación a la vez. * El Encargado de Titulación puede además tener el rol de Profesor * Las eliminaciones se deben manejar sólo mediante un cambio de estado. No se deben borrar registros de la base de datos | | |

|  |  |  |
| --- | --- | --- |
| **HISTORIA DE USUARIO** | | |
| **Id:** ADM-002 | **Usuario:** Secretaria / Encargado de Titulación | |
| **Nombre Historia:** Registrar trabajo de titulación | | |
| **Prioridad en Negocio:** 1 | | **Riesgo de Desarrollo:** |
| **Puntos Estimados:** | | **Iteración Asignada:** |
| **Programador responsable:** | | |
| **Descripción:**  El sistema debe permitir que el usuario cree/actualice/finalice la bitácora de un trabajo de titulación.  Para la creación debe ingresar:   * Título del trabajo * Listado de estudiantes incluyendo por cada uno su rut, nombre y carrera * Profesores guía incluyendo su rut y nombre   Sólo se pueden realizar la siguientes actualizaciones:   * Modificar el título * Eliminar algún estudiante * Modificar el profesor guía   Cuando se da por finalizada la bitácora de debe indicar la causa de la finalización | | |
| **Validación:**   * En cada trabajo de titulación pueden participar entre 1 y 4 estudiantes * Un trabajo de titulación puede tener máximo 2 profesores guías * Las posibles causas de finalización son dos: no continuidad del trabajo o aprobación del término del trabajo | | |

|  |  |  |
| --- | --- | --- |
| **HISTORIA DE USUARIO** | | |
| **Id:** REG-001 | **Usuario:**  Estudiante | |
| **Nombre Historia:** Registrar avance | | |
| **Prioridad en Negocio:** 1 | | **Riesgo de Desarrollo:** |
| **Puntos Estimados:** | | **Iteración Asignada:** |
| **Programador responsable:** | | |
| **Descripción:**  El sistema debe permitir que los estudiantes ingresen sus avances semanales. Cada vez que se registra un avance, el sistema debe registrar la fecha y el nombre del usuario que realizó el registro. Además, de forma opcional, el estudiante podría adjuntar algún archivo como evidencia del avance reportado (Ver HU REG-003). | | |
| **Validación:**   * Se pueden registrar varias entradas en la bitácora en una misma fecha, y cada entrada puede tener su propia evidencia | | |

|  |  |  |
| --- | --- | --- |
| **HISTORIA DE USUARIO** | | |
| **Id:** REG-002 | **Usuario:**  Profesor Guía / Encargado de Titulación | |
| **Nombre Historia:** Revisar avance | | |
| **Prioridad en Negocio:** 1 | | **Riesgo de Desarrollo:** |
| **Puntos Estimados:** | | **Iteración Asignada:** |
| **Programador responsable:** | | |
| **Descripción:**  El sistema debe permitir que el profesor guía revise los avances reportados por los estudiantes.  El sistema debe dar la opción al profesor de registrar comentarios en relación a los avances reportados.  Además, el profesor debe tener la opción de descargar las evidencias asociadas. | | |
| **Validación:**   * El profesor sólo puede revisar los avances de los temas que él/ella guía * El encargado de titulación puede revisar los avances de todos los temas | | |

|  |  |  |
| --- | --- | --- |
| **HISTORIA DE USUARIO** | | |
| **Id:** REG-003 | **Usuario:**  Estudiante | |
| **Nombre Historia:** Registrar evidencia | | |
| **Prioridad en Negocio:** 2 | | **Riesgo de Desarrollo:** |
| **Puntos Estimados:** | | **Iteración Asignada:** |
| **Programador responsable:** | | |
| **Descripción:**  El sistema debe permitir que los estudiantes ingresen algún archivo como evidencia de su avance. Cuando se registra la evidencia, se debe ingresar un nombre que identifique la evidencia, y el sistema debe permitir asociar esta evidencia a alguna actividad reportada previamente. | | |
| **Validación:**   * Los formatos permitidos para el archivo de la evidencia son Word, pdf o jpg | | |

|  |  |  |
| --- | --- | --- |
| **HISTORIA DE USUARIO** | | |
| **Id:** NOT-001 | **Usuario:**  Sistema | |
| **Nombre Historia:** Generar notificaciones | | |
| **Prioridad en Negocio:** 3 | | **Riesgo de Desarrollo:** |
| **Puntos Estimados:** | | **Iteración Asignada:** |
| **Programador responsable:** | | |
| **Descripción:**  El sistema debe generar dos tipos de notificaciones:   1. Notificaciones que se informen al ingresar a la plataforma 2. Notificaciones vía correo electrónico   Las notificaciones que deben generarse son las siguientes:   * Cuando un estudiante registra un avance o ingresa una evidencia, se debe notificar a su profesor guía * Cuando un profesor guía registra un comentario, se debe notificar al estudiante * Cuando la secretaria o encargado de titulación da por finalizada una bitácora, se debe notificar a los estudiantes y profesores asociados | | |
| **Validación:** | | |

|  |  |  |
| --- | --- | --- |
| **HISTORIA DE USUARIO** | | |
| **Id:** REP-001 | **Usuario:**  Estudiante / Profesor Guía / Encargado de Titulación | |
| **Nombre Historia:** Consultar Bitácoras | | |
| **Prioridad en Negocio:** 3 | | **Riesgo de Desarrollo:** |
| **Puntos Estimados:** | | **Iteración Asignada:** |
| **Programador responsable:** | | |
| **Descripción:**  El sistema debe permitir visualizar todos los registros asociados a una bitácora  La información a listar por cada bitácora es:   * Nombre del trabajo de titulación * Estudiante(s) que desarrollan el trabajo * Profesor(es) guía(s) * Todos los registros ingresados, que muestre fecha, quién registró, descripción y si existe alguna evidencia asociada | | |
| **Validación:**   * Un estudiante sólo puede consultar su propia bitácora * Un profesor guía puede consultar cualquier bitácora de los trabajos que guía * El encargado de titulación puede consultar cualquier bitácora registrada en el sistema | | |

# Requerimientos no Funcionales

***REQ NF01: Requerimientos de interfaz.***

El sistema debe minimizar la necesidad de moverse en distintas interfaces de usuario para realizar una tarea.

La interfaz debe ser gráfica, en base a tonos azules y blanco y debe tener siempre a la vista un menú que permita moverse directamente de una opción a otra.

Además, se debe minimizar la digitación de la información.

***REQ NF02: Requerimientos de seguridad***

El acceso al sistema debe ser basado en roles. Esto significa el uso de cuenta y clave de acceso. En base a la cuenta se reconoce el rol del usuario y con ello se despliegan sólo las opciones que están definidas para su rol.

***REQ NF03: Requerimientos de usabilidad***

El sistema debe contar con ayuda en línea, en forma de videos “How to”. Además al finalizar el desarrollo, debe entregarse un manual de usuario que permita que cualquier nuevo usuario pueda realizar las labores de su interés de manera efectiva.

***REQ NF04: Mantenibilidad***

El sistema debe ser desarrollado en base a una arquitectura MVC, teniendo en cuenta los conceptos de cohesión y acoplamiento. El código debe ser desarrollado en base a un estándar para la nominación de métodos, variables y constantes.

Los nombres utilizados deben ser mnemotécnicos.

El código que no se pueda entender por sí mismo, debe contar con los comentarios que permitan entender la lógica y funcionalidad implementada.

***REQ NF05: Facilidades para la instalación del producto***

El sistema debe contar con una guía de instalación que indique claramente qué se debe realizar para que el sistema quede en funcionamiento.

***REQ NF06: Disponibilidad***

El sistema debe estar disponible vía web, sobre cualquier sistema operativo, utilizando cualquier explorador de internet 24 horas los 7 días a la semana, durante todo el año.

***REQ NF07: Performance***

El sistema debe soportar 10 usuarios concurrentes. Los tiempos de respuesta del sistema deben ser menores a 30 seg