



Departamento de Informática
Universidad Técnica Federico Santa María



Entregable III

Análisis y Diseño de Software / Fundamentos de Ingeniería de Software

Integrantes:

Nombre	Email	Teléfono
Mario Perez	mario.perez.12@sansano.usm.cl	95542873
Belén González	belen.gonzalez.12@sansano.usm.cl	98423118
Andrea Rodriguez	andrea.rodriguez.12@sansano.usm.cl	67796474

1- Listado de requerimientos

Funcional:

- 1.- (id: RF01) El sistema debe proveer un acceso diferenciado con cuatro niveles usuario: [Obligatorio]
- 2.- (id: RF02) El sistema debe permitir al usuario Administrador y Director de Laboratorio el acceso completo a la página. [Obligatorio]
- 3.- (id: RF03) El usuario Director de Laboratorio debe poder crear reportes de otros usuarios. [Obligatorio]
- 4.- (id: RF04) El usuario Director de Laboratorio debe poder crear eventos del laboratorio. [Obligatorio]
- 5.- (id: RF05) El sistema debe permitir al usuario Director de Laboratorio y Administrador poder agregar usuarios. [Obligatorio]
- 6.- (id: RF06) El usuario Director de Laboratorio debe poder validar la información de otros usuarios. [Obligatorio]
- 7.- (id: RF07) Usuarios del Laboratorio deben poder navegar en una intranet privada. [Obligatorio]
- 8.- (id: RF08) Usuarios no registrados deben tener acceso sólo a la información pública en la página web. [Obligatorio]
- 9.- (id: RF09) El sistema debe proveer de un registro para los usuarios. [Obligatorio]
- 10.- (id: RF10) El usuario debe poder ver una ficha personal donde tendrá que ingresar de manera obligatoria los campos que allí se le soliciten, como su nombre por ejemplo (nombre de usuario no es lo mismo que el campo de nombre). [Obligatorio]
- 11.- (id: RF11) El sistema debe proveer al usuario la opción de almacenar fotografías. [Obligatorio]
- 12.- (id: RF12) El sistema debe proveer al usuario la opción de almacenar PDF, la cual pueden ser de diferentes tipos: Publicaciones en revistas científicas , Publicaciones en capítulos de libros, Asistencia a Congresos y Tesis de pregrado y postgrado. [Obligatorio]
- 13.- (id: RF13) El sistema debe proveer al usuario la opción de almacenar videos. [Obligatorio]

- 14.- (id: RF14) El usuario debe poder escoger opciones para ingresar el Cargo en el Laboratorio (Estudiante de pregrado, Estudiante de Postgrado, Trabajador, Director (Jefe de Laboratorio), Colaborador (nacional o internacional), otros)[Obligatorio]
- 15.- (id: RF15) El usuario debe poder escoger opciones para ingresar el área de investigación (Biotecnología vegetal, Compuestos bioactivos, Microbiología, Biotecnología Ambiental, otros) [Obligatorio]
- 16.- (id: RF16) El sistema automáticamente, subiendo un pdf de un artículo científico, debe reconocer los autores, título, año, revista, keywords del paper (tipo Mendeley) y cuando haya algún campo vacío o erróneo debe ser modificable [Obligatorio]
17. (id: RF17) Que el sistema haga un link entre los usuarios, sus áreas de expertise, artículos científicos, la participación a congresos y las cooperaciones que tenga con otros integrantes del laboratorio y que en la ficha de cada uno se muestre con quién tiene más colaboración (basándose en los keywords de los paper por ejemplo) [Obligatorio]
18. (id: RF18) El usuario debe poder buscar artículos científicos, tesis, capítulos de libros, congresos etc y debe poder refinar la búsqueda [Obligatorio]
19. (id: RF19) Sistema debe notificar al director o administrador, cuando el usuario se registre, y notificar de vuelta al usuario cuando el director o administrador acepte o rechace la solicitud de registro.[Obligatorio]
20. (id: RF20) El usuario debe poder recuperar su contraseña. [Obligatorio]
21. (id: RF21) Usuario debe mandar una solicitud al profesor que quiere como guía para su tesis, y este decide si aceptar o rechazar. [Obligatorio]
22. (id: RF22) El sistema debe proveer un foro entre el profesor guía y su estudiante. [Obligatorio]

No Funcional

1. (id: RNF01) La página web debe ser de fácil manejo (nivel usuario) y con una interfaz amigable. (asoc RF01)
2. (id: RNF02) Se debe poder acceder al sistema desde el PC y móvil.(asoc RF01)
3. (id: RNF03) La página debería estar en Español e Inglés. (asoc RF01)

2- Casos de uso

Caso de uso breve

1°-Caso de uso breve: "Comentar tesis"

El **usuario alumno** sube un avance de tesis. El SISTEMA envía una notificación al **usuario profesor** sobre documento enviado. El **usuario profesor** revisa el documento. El **usuario profesor** envía documento con comentarios asociados. El SISTEMA envía notificación al usuario sobre documento enviado.

2°-Caso de uso breve: "Generar reporte de Usuario"

El **director** ingresa en sus sesión de laboratorio. El **director** ingresa en la sección de reportes de usuario. El **director** selecciona generar un nuevo reporte. El **director** selecciona un usuario y genera el reporte. El SISTEMA recopila la información del usuario. El SISTEMA retorna la información en formato pdf. El SISTEMA regresa mensaje de éxito.

3°-Caso de uso breve: "Crear evento de laboratorio"

El **director** ingresa en su sesión de laboratorio. El **director** selecciona la opción de crear un nuevo evento. El **director** ingresa los datos relativos al evento. El **director** selecciona guardar los datos ingresados. El SISTEMA almacena el evento y retorna mensaje de éxito.

4°-Caso de uso breve: "Iniciar sesión"

El **usuario** ingresa al portal web del laboratorio. El **usuario** se dirige a la zona de login. El **usuario** ingresa su nombre de usuario, contraseña y tipo de usuario. El SISTEMA confirma que los datos ingresados por el usuario sean correctos. El SISTEMA redirige al usuario a la página principal con uso diferenciado.

5°-Caso de uso breve: "Modificar ficha personal"

El **usuario** ingresa en su ficha personal. El **usuario** edita o completa los campos que se encuentren vacíos con sus datos. El **usuario** guarda los cambios realizados. El SISTEMA almacena la información ingresada y retorna un mensaje de confirmacion.

6°-Caso de uso breve: "Identificar campos de PDF"

El **usuario** sube un archivo PDF a la página. El SISTEMA recibe el archivo. El SISTEMA examina el archivo en búsqueda de campos de identificación. El SISTEMA retorna los campos de identificación de paper.

7°-Caso de uso breve: "Buscar documento"

El **usuario** inicia una búsqueda de un documento en la página ingresando palabras clave, el SISTEMA retorna todos los documentos que contengan las palabras clave ingresadas en el buscador, el **usuario** selecciona distintos filtros de visualización, el sistema filtra los resultados según los filtros y retorna los documentos.

8°-Caso de uso breve: "Registrar usuario"

El **usuario** se registra en la página, el SISTEMA retorna un mensaje de confirmación de solicitud de registro y envía notificación al **director** y **administrador**, el **director** o **administrador** acepta o rechaza la solicitud de registro, el SISTEMA envía notificación de aceptación o rechazo de solicitud al **usuario**, el SISTEMA registra o elimina al usuario.

9°-caso de uso breve: "Recuperar contraseña"

El **usuario** selecciona opción para recuperar contraseña. El SISTEMA solicita al usuario ingresar el correo electrónico de registro. El **usuario** ingresa su correo electrónico. El SISTEMA confirma que el correo electrónico ingresado coincide con correo electrónico de registro. El SISTEMA envía mensaje al correo electrónico que contiene la contraseña asociada a este.

10°-caso de uso breve: "Seleccionar profesor guía de tesis"

El **usuario estudiante** selecciona opción que muestra a los profesores del departamento. El SISTEMA provee lista con profesores. El **usuario estudiante** elige al profesor que desea como guía de tesis. El SISTEMA envía solicitud de "Aceptar/Rechazar Tesis" al usuario-profesor. El **usuario profesor** recibe solicitud. El **usuario profesor** envía respuesta de solicitud. El SISTEMA recibe la respuesta. El SISTEMA reenvía respuesta al usuario-estudiante. El **usuario estudiante** recibe respuesta del usuario-profesor.

Casos de Uso Extendidos

I.- Nombre del caso: Iniciar Sesión

Actores: Usuario(primario)

Precondiciones:

- Usuario está registrado en la página web

Postcondiciones:

- Sesión iniciada

Curso Normal:

- 1) Usuario ingresa a la página web del laboratorio.
- 2) Usuario ingresa a la pestaña de login.
- 3) Usuario ingresa nombre de usuario y contraseña.
- 4) Sistema confirma que los datos ingresados por usuario sean correctos.
- 5) Sistema redirige al usuario a la página principal del laboratorio con uso diferenciado.

Curso Alternativo:

3.a) Usuario ingresa nombre de usuario o contraseña inválidos.

3.a.1) Usuario ingresa datos erróneos.

3.a.2) Sistema notifica al usuario datos erróneos.

4.a) Falla en la conexión al Sistema.

4.a.1) Sistema intenta conexión a base de datos.

4.a.1.a) Sistema logra conexión.

4.a.1.b) Sistema no logra conexión.

4.a.1.b.1) Sistema envía a usuario mensaje de conexión no disponible

II-Nombre del caso: Registrar usuario.

Actores: Usuario (primario), director(primario),
administrador(primario).

Precondiciones:

-El usuario no debe poseer ya una cuenta registrada a su nombre.

-El usuario debe pertenecer al laboratorio de investigación.

Postcondiciones:

-Usuario queda registrado en la página.

-Usuario puede seguir completando el formulario de registro.

Curso normal:

1)usuario ingresa a la página web del laboratorio.

2)usuario ingresa en la sección de registro de cuentas.

3)usuario ingresa los datos pedidos en los distintos campos del formulario de registro (nombres, apellidos, correo y contraseñas en un principio).

4)sistema verifica que los datos ingresados sean correctos.

5)sistema envía mail de solicitud de registro al director y administrador y notifica la solicitud en espera al usuario.

6)sistema retorna mensaje de confirmación de solicitud.

7)director o administrador acepta registro del usuario.

8)sistema registra al usuario.

9)sistema retorna notificación de confirmación de registro al usuario.

Cursos alternativos:

3.a)Usuario no ingresa correctamente los datos en los campos del formulario.

3.a.1.a)usuario ingresa datos erróneos, repetidos o deja campos en blanco.

3.a.1.a.1)sistema verifica que los datos sean correctos.

3.a.1.a.2)sistema retorna mensaje de campos erróneos.

7.a)director o administrador rechaza solicitud de registro.

7.a.1)sistema envía notificación de rechazo de solicitud al usuario.

7.a.2)sistema almacena los datos del registro por 3 meses.

8.a)Falla conexión con el sistema.

8.a.1)sistema envía mensaje de error de conexión.

III.- Nombre del caso: Recuperar Contraseña

Actores: Usuario(primario)

Precondiciones:

- Usuario está registrado en la página web.

Postcondiciones:

- Contraseña modificada.

Curso Normal:

- 1) Usuario selecciona opción para recuperar su contraseña.
- 2) Sistema solicita al usuario ingresar correo electrónico con el que se registró.
- 3) Usuario ingresa su correo electrónico.
- 4) Sistema verifica que el correo electrónico ingresado coincida con el de registro
- 5) Sistema envía mensaje al correo electrónico que contiene contraseña asociada a este.

Curso Alternativo:

3.a) Usuario ingresa datos inválidos o correo electrónico no registrado en el Sistema.

3.a.1) Usuario ingresa correo electrónico inválido.

3.a.2) Sistema envía mensaje de datos inválidos y solicita al usuario que intente de nuevo.

4.a) Falla en la conexión con el Sistema

4.a.1) Sistema intenta conexión.

4.a.1.a) Sistema se conecta..

4.a.1.b) Sistema no se conecta.

4.a.1.b.1) Sistema envía a usuario mensaje de conexión no disponible.

4.a.1.b.2) Sistema ofrece al usuario informar a mantención sobre el error.

4.a.1.b.3) Sistema solicita al usuario que intente conexión más tarde

IV-nombre del caso: Comentar Proyecto

Actores: Usuario(Primario), Profesor (secundario)

Precondiciones:

-El usuario se encuentra logueado en su cuenta del laboratorio.

-El usuario ya realizó una solicitud aceptada con éxito y por tanto tiene un Profesor guía.

Postcondiciones:

-El usuario revisa documento corregido.

Curso normal:

- 1) Usuario ingresa en su ficha personal.
- 2) Usuario selecciona la opción "avance de Proyecto".
- 3) Usuario sube un documento (.doc) o ingresa un link (google drive)
- 4) Usuario envía la información al Profesor.
- 5) Sistema envía una notificación al Profesor de documento enviado.
- 6) Sistema retorna mensaje de éxito en el envío.
- 7) Profesor revisa el documento.
- 8) Profesor escribe comentarios y adjunta el documento o link.
- 9) Profesor envía comentarios a usuario.

10) Sistema envía una notificación al Usuario de documento enviado.

11) Sistema retorna mensaje de éxito en el envío.

Curso alternativos:

3.a) Usuario sube un documento de tipo erróneo

3.a.1) Sistema retorna mensaje de error en el tipo archivo y solicita cambiarlo

3.a.2) Sistema no permite el envío hasta cambiar el archivo.

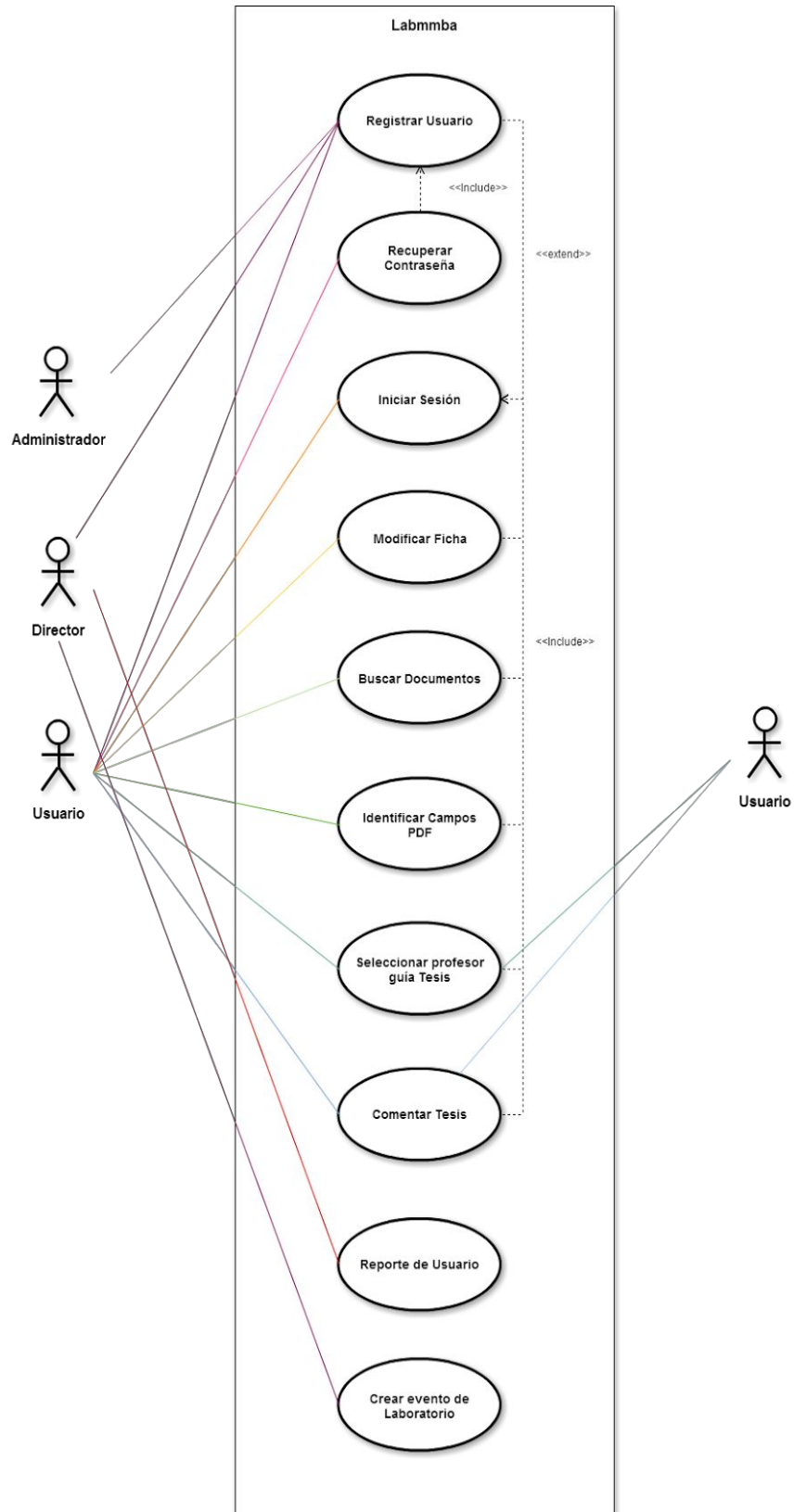
5.a) Sistema falla en enviar notificación

5.a.1) Sistema retorna mensaje de error en el envío.

5.a.1) Sistema solicita intentarlo nuevamente.

Diagrama de caso de uso

A modo general el administrador y el director también son usuarios, por ende cumplen las mismas funcionalidades del mismo. También se distinguen en el diagrama 2 tipos de usuario, dentro del usuario común, que serían: usuario alumno y usuario profesor o tutor que participan en los casos de uso: Seleccionar profesor guía Tesis, Comentar Tesis. Nótese además que es posible cambiar el rol de usuario profesor o tutor por usuario cotutor o que el alumno quede secundario cuando éste comenta de vuelta.



3- Estimación inicial de *tamaño* del sistema

Como SP se utilizará un día ideal de trabajo de 8 horas aplicada a sólo una persona inexperta

Funcional:

1.- El sistema debe proveer un acceso diferenciado con cuatro niveles usuario: La convergencia de nuestras estimaciones concluyó que para hacer esta parte es necesario tener conocimiento, obviamente, de rails, crear la base de datos y finalizado eso recién crear el sistema diferenciado. Considerando a una sola persona inexperta, concluimos que en un día ideal de trabajo le tomaría alrededor de una semana manejar de manera eficaz rails (no de manera experta, ojo ahí, sólo conocería los aspectos básicos de rails), y otra semana más para implementar la base de datos, esto es, pensar de mejor manera los requerimientos que le pide el cliente (que ya están previamente definidos y conversados) y finalmente 5 días para realizar los accesos diferenciados y sus respectivos login de usuario, considerando la dificultad que estos presentan por la cantidad de roles de usuario. Añadiremos 2 días más por posibles errores que puedan ocurrir en el proceso. (7 SP)

2.- El sistema debe permitir al usuario Administrador y Director de Laboratorio el acceso completo a la información. Con lo anterior ya hecho es posible que la persona lo haga en 4 ó 5 días, pues ya sabría como manejar rails, al crear el acceso diferenciado solo le faltaría otorgar ciertos permisos y añadiremos 1 día más para solucionar los problemas que se presenten en el camino. (6 SP)

3.- El usuario Director de Laboratorio debe poder crear reportes de otros usuarios. Los reportes se generarán a partir del contenido que tengan los usuarios en la Base de Datos, por ende primero hay que llenar la Base de Datos con los archivos que hayan subido los usuarios para generar el reporte y crear la aplicación necesaria que entregue dichos documentos, esto puede tardar 4 ó 5 días para generar una buena aplicación y un día más para la solución de errores. (6 SP)

4.- El usuario Director de Laboratorio debe poder crear eventos del laboratorio. No sé habló mucho al respecto de este tema con el cliente, y el poco tiempo que tuvimos para realizar este informe y entre nuestras otras materias, desconocemos un poco esta función. Pero asumimos mientras que al igual que Facebook sea un evento donde el Director de Laboratorio cree un comunicado y llegue como notificación a los otros usuarios, donde podrán ver de qué se trata el evento, pero sin poner asistencia. Sino más bien un mensaje, el cual se verá en una

página destinada a eventos de Laboratorio. Para crear esta función se necesitará alrededor de 2 días pues se necesitará agregar un comentario y mostrarlo por la página y además si así lo desean nuestro clientes podemos añadir alguna función de subir alguna foto o documento si así lo o algún filtro de eventos, con ello una persona en un día ideal de trabajo le tomaría 3 días más para su implementación y otro día por posibles fallos. (2 SP)

5.- El sistema debe permitir al usuario Director de Laboratorio y Administrador poder agregar usuarios. Ya tenemos la Base de Datos y tenemos al director agregado, ahora para la funcionalidad de esta aplicación pensamos que en un día ideal de trabajo tomaría alrededor de 5 ó 6 días pues consideramos que es necesario dedicar más tiempo para el correcto funcionamiento y no hayan fallas inesperadas que perjudiquen a la página. (6 SP)

6.- El usuario Director de Laboratorio debe poder validar la información de otros usuarios. En base a la similitud de la aplicación anterior tomamos como base de lo que haya creado anteriormente esté bien y simplemente le haga ajustes. En base a qué tan parecido serán asumimos que son 2 día de trabajo ideal. (2 SP)

7.- Usuarios del Laboratorio deben poder navegar en una intranet privada. En base al acceso diferenciado, y teniendo la Base de datos lista, asumimos que 8 días por lo bajo, ya que la intranet lo es todo. Dejaremos 4 días mas de pruebas por errores, también por lo bajo. Cabe destacar que en esta parte hay más requerimientos involucrados (12 SP)

8.- Usuarios no registrados deben tener acceso sólo a la información visible en la página web. En base a que lo anterior esté bien creado solo falta restringir ciertos permisos a estos usuarios 2 días estaría bien para que quede bien implementando, pero entendiendo los riesgos que esto pueda acarrear le añadiremos 2 días para cerciorarnos de su buen funcionamiento. (4 SP)

9.- El sistema debe proveer de un registro para los usuarios. Esto fue considerado en la parte inicial del tema en el cual englobamos los accesos diferenciados, entendiendo que nos costaba trabajo imaginarnos los accesos diferenciados sin tener usuarios logueados. Con lo cual sumimos 2 días de trabajo, a lo mucho. (2 SP)

10.-El usuario debe poder ver una ficha personal donde tendrá que ingresar de manera obligatoria los campos que allí se le soliciten, como su nombre por ejemplo (nombre de usuario no es lo mismo que el campo de nombre). Como ya está la base de datos creada simplemente creamos la aplicación solicitada, para ello necesitamos 1 día ideal de trabajo sólo para implementar el nombre y algunas funcionalidades, ya que las demás los tenemos especificados abajo. (1 SP)

11.-El sistema debe proveer al usuario la opción de almacenar fotografías. Almacenaremos la fotos dentro de la Base de Datos, e implementaremos el sistema para que

pida las fotos a los usuarios, ello tomará 3 días ideales de trabajo para implementar su correcto funcionamiento. (3 SP)

12.- **El sistema debe proveer al usuario la opción de almacenar PDF, la cual pueden ser de diferentes tipos: Publicaciones en revistas científicas, Publicaciones en capítulos de libros, Asistencia a Congresos y Tesis de pregrado y postgrado.** Como hemos creado la base de datos, solo debemos implementar el que suba los archivos, pero esto debemos hacerlo en las diferentes partes de la página donde lo solicitan, es decir, Publicaciones en revistas científicas, Publicaciones en capítulos de libros, Asistencia a Congresos y Tesis de pregrado y postgrado, por ello creemos que en un día ideal de trabajo tomará aproximadamente 4 días incluyendo los posibles errores. (4 SP)

13.- **El sistema debe proveer al usuario la opción de almacenar videos.** (Pese a que no se habló mucho de esto y en vista del poco tiempo que hemos tenido esta semana para afinar el proyecto, estos videos pueden ser subidos a la plataforma o direcciones de youtube.) Justamente para lo anterior en base a que es similar el proceso de almacenar los pdf creemos que en un día ideal de trabajo nos tomará aproximadamente 3 días, más un día de error, pues como la persona es inexperta tiene que encontrar la manera más óptima su implementación. (4 SP)

14.- **El usuario debe poder escoger opciones para ingresar el Cargo en el Laboratorio (Estudiante de pregrado, Estudiante de Postgrado, Trabajador, Director (Jefe de Laboratorio), Colaborador (nacional o internacional), otros).** Esto está almacenado en la Base de Datos, el valor que escoja el usuario será True, mientras todos los demás False. Ello tomará alrededor de 2 días hábiles. (2 SP)

15.- **El usuario debe poder escoger opciones para ingresar el área de investigación (Biotecnología vegetal, Compuestos bioactivos, Microbiología, Biotecnología Ambiental, otros).** Es lo mismo que lo anterior, pero ahora asociado a un área. Por lo tanto 2 días hábiles. (2SP)

16.- **El sistema automáticamente, subiendo un pdf de un artículo científico, debe reconocer los autores, título, año, revista, keywords del paper (tipo Mendeley) y cuando haya algún campo vacío o erróneo debe ser modificable.** Para captar el funcionamiento de la API necesitamos averiguar sobre este, para ello serán 2 días para investigar y 2 para implementar, más 2 días más para checkear errores, y solucionar posibles impedimentos que vayan surgiendo en el camino. (6 SP)

17. **Que el sistema haga un link entre los usuarios, sus áreas de expertise, artículos científicos, la participación a congresos y las cooperaciones que tenga con otros integrantes del laboratorio y que en la ficha de cada uno se muestre con quién tiene más colaboración (basándose en los keywords de los paper por ejemplo).** Al tener la base de datos creada solo necesitamos hacer un join con las conexiones correspondientes que tengan

estos usuarios para así crear los link necesarios entre ambos, o estrechamente que cada usuario coloque de manera manual con quien colaboró para ciertos proyectos o asociaciones. Se ocuparán 1 día para averiguar cuál dicho proceso es más conveniente. 1 días más para averiguar cómo implementarlo y 3 día de implementación. (5 SP)

18. **El usuario debe poder buscar artículos científicos, tesis, capítulos de libros, congresos etc y debe poder refinar la búsqueda.** Cómo es posible usar palabras claves como Mendeley para encontrar pdf necesitamos conocer más sobre cómo implementar la búsqueda y captar filtros. Para ello se usará 4 días ideales de trabajo. (4 SP)

19. **Sistema debe notificar al director o administrador, cuando el usuario se registre, y notificar de vuelta al usuario cuando el director o administrador acepte o rechace la solicitud de registro.** Para crear esta función es necesario averiguar cómo hacer este procedimiento 2 días averiguaremos sobre cómo se podrá implementar y usaremos 3 días más para implementarlo, más otro día para verificar que todo funcione correctamente. (5 SP)

20. **El usuario debe poder recuperar su contraseña.** Este proceso es un tanto complicado también, y le dedicaremos 2 días para averiguar su implementación, 4 días más para implementarla y otro día más para verificar si es que hay algún error. (6 SP)

21. **Usuario debe mandar una solicitud al profesor que quiere como guía para su tesis, y este decide si aceptar o rechazar.** Se utilizarán 3 días para su implementación y otro para verificar errores. (4 SP)

22. **El sistema debe proveer un foro entre el profesor guía y su estudiante.** Básicamente es mandar mensajes entre el profesor guía y el estudiante, encontramos que es un requerimiento difícil lo cual tomaría 2 días para averiguar, 4 para su implementación y 2 para verificar errores. (8 SP)

No Funcional

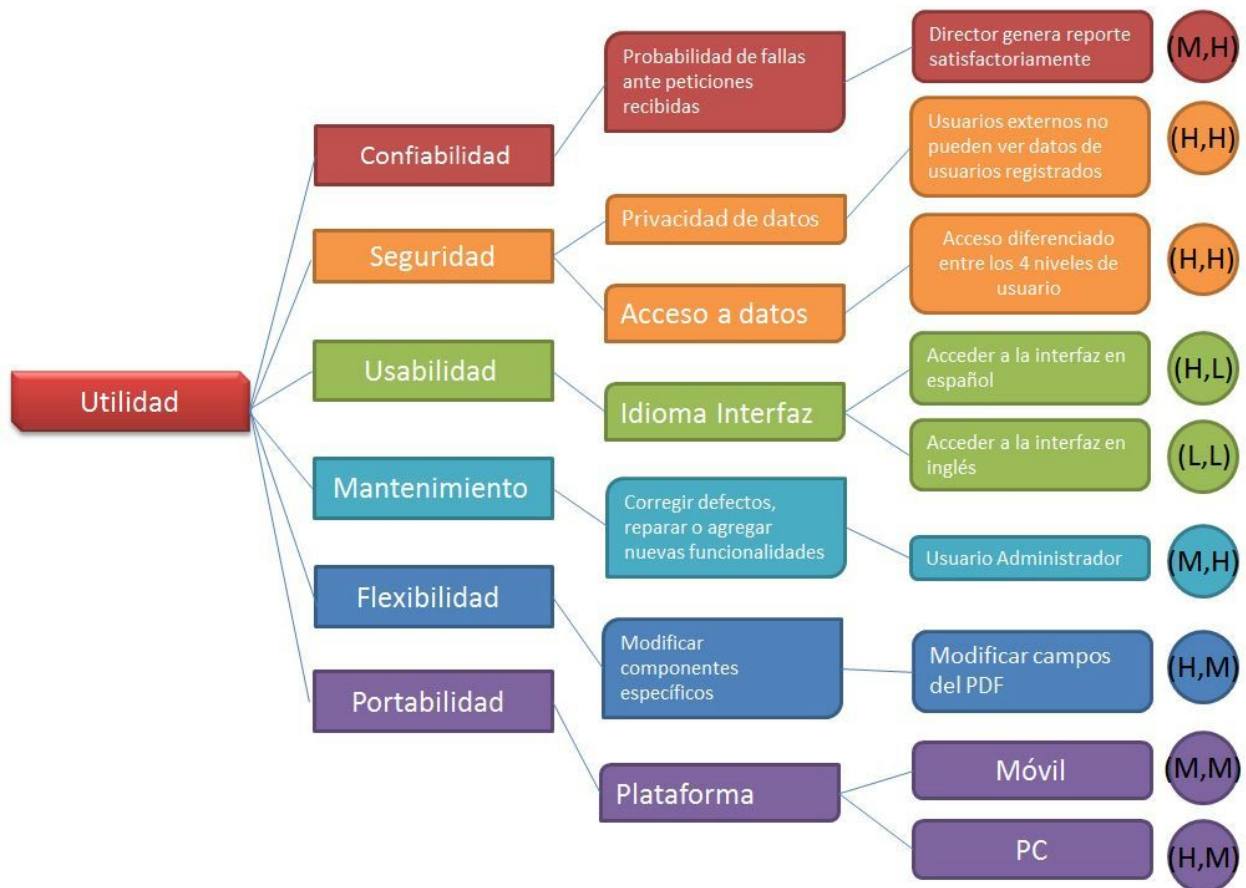
1. **La página web debe ser de fácil manejo (nivel usuario) y con una interfaz amigable.** Esto se puede hacer a medida que vamos implementado los requerimientos, y al finalizar adornarla y agregarle aspectos atractivos y modificar algunas cosas ya implementadas, esto sería alrededor de 5 días, pues también habría que modificar cosas que ya se hicieron para que la página quede más amigable. (5 SP)

2. **Se debe poder acceder al sistema desde diferentes dispositivos.** Este es un tema que hay que averiguar, ya que no se sabe que cosas habrá que implementar nuevamente, se estima que se necesitan 7 días por lo bajo para entender cómo implementarlo. (7 SP)

3. **La página debería estar en Español e Inglés.** Necesitamos averiguar esta funcionalidad, o simplemente encontrar alguna api para ello, estimamos 2 días para averiguar su funcionalidad y 4 para implementarla. (6 SP)

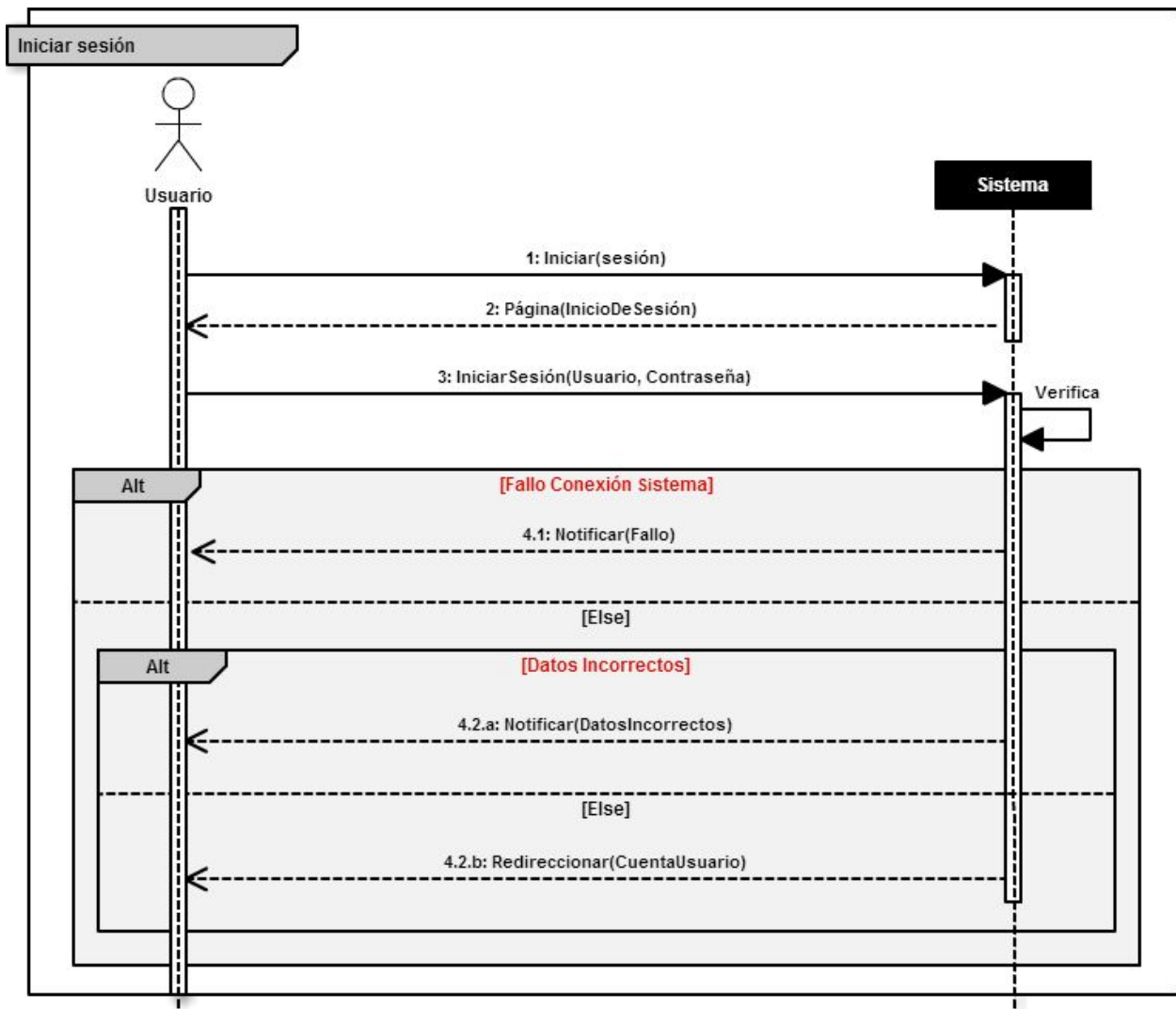
Esto nos da un total de 119 SP. Hay que tener en cuenta que en esta estimación se considera a una persona inexperta en el tema, claramente los SP bajan si se considera un equipo de trabajo completo y con más experiencia.

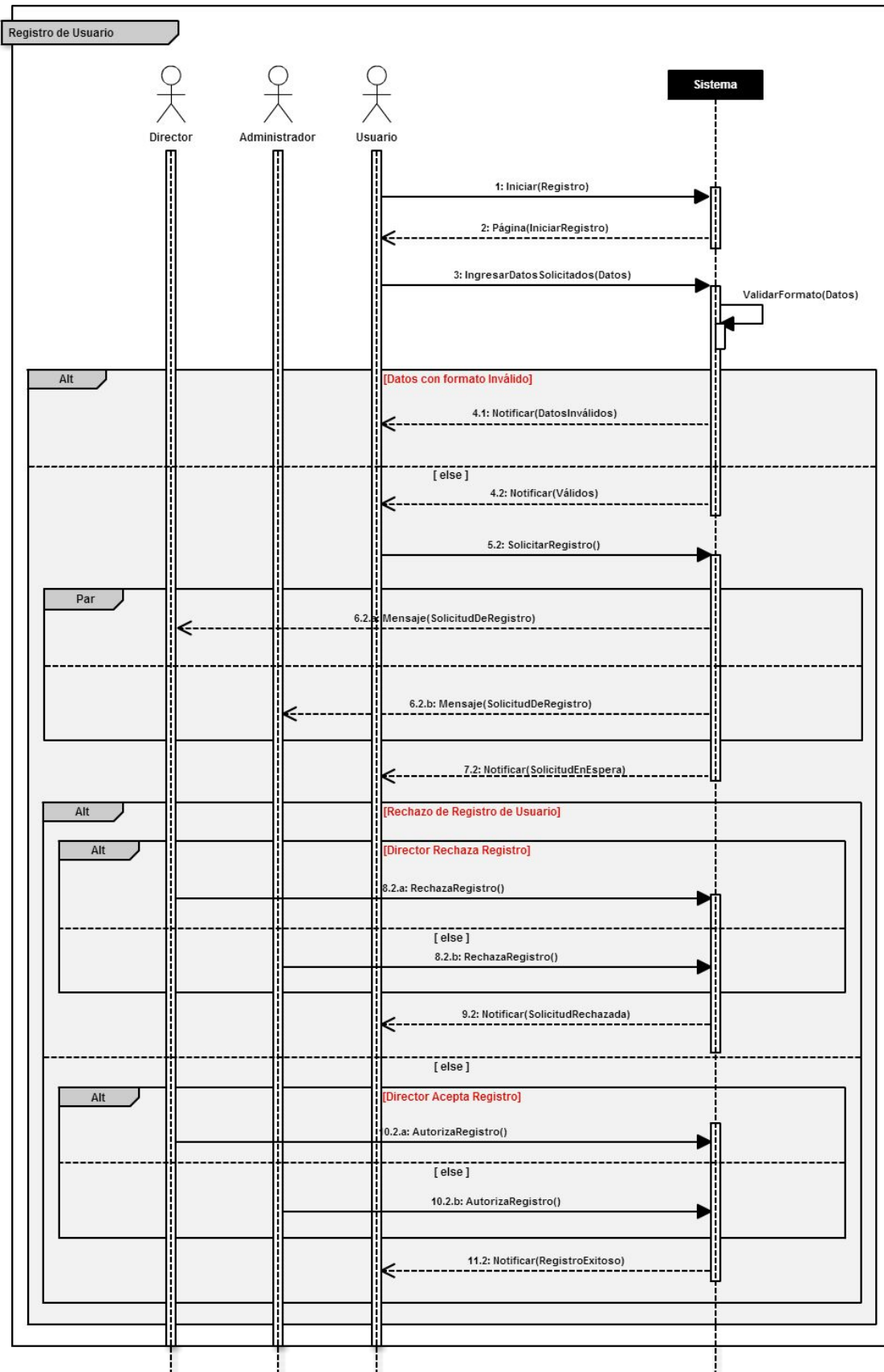
4- Árbol de utilidad



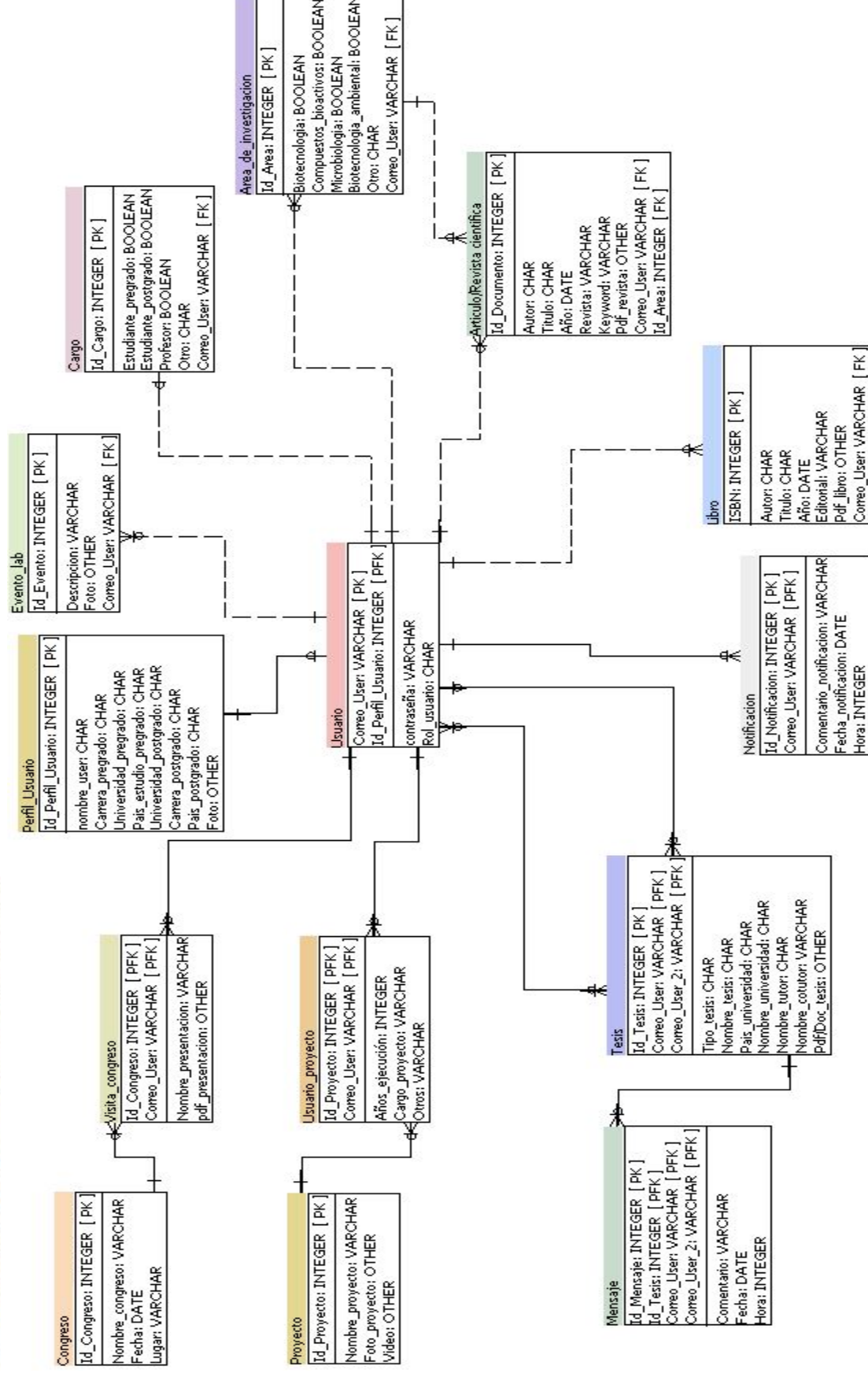
El árbol de utilidad posee los atributos de calidad más necesarios para el correcto funcionamiento de la página web, los cuales proveen a grandes rasgos **Atributos de calidad en operación, Atributos de calidad en desarrollo, Atributos de calidad en implementación**, basados en las reuniones con el cliente. Además se considera que el administrador será la persona encargada de manejar la aplicación a futuro y los posibles requerimientos que esta necesite.

5- Diagramas de secuencia del sistema

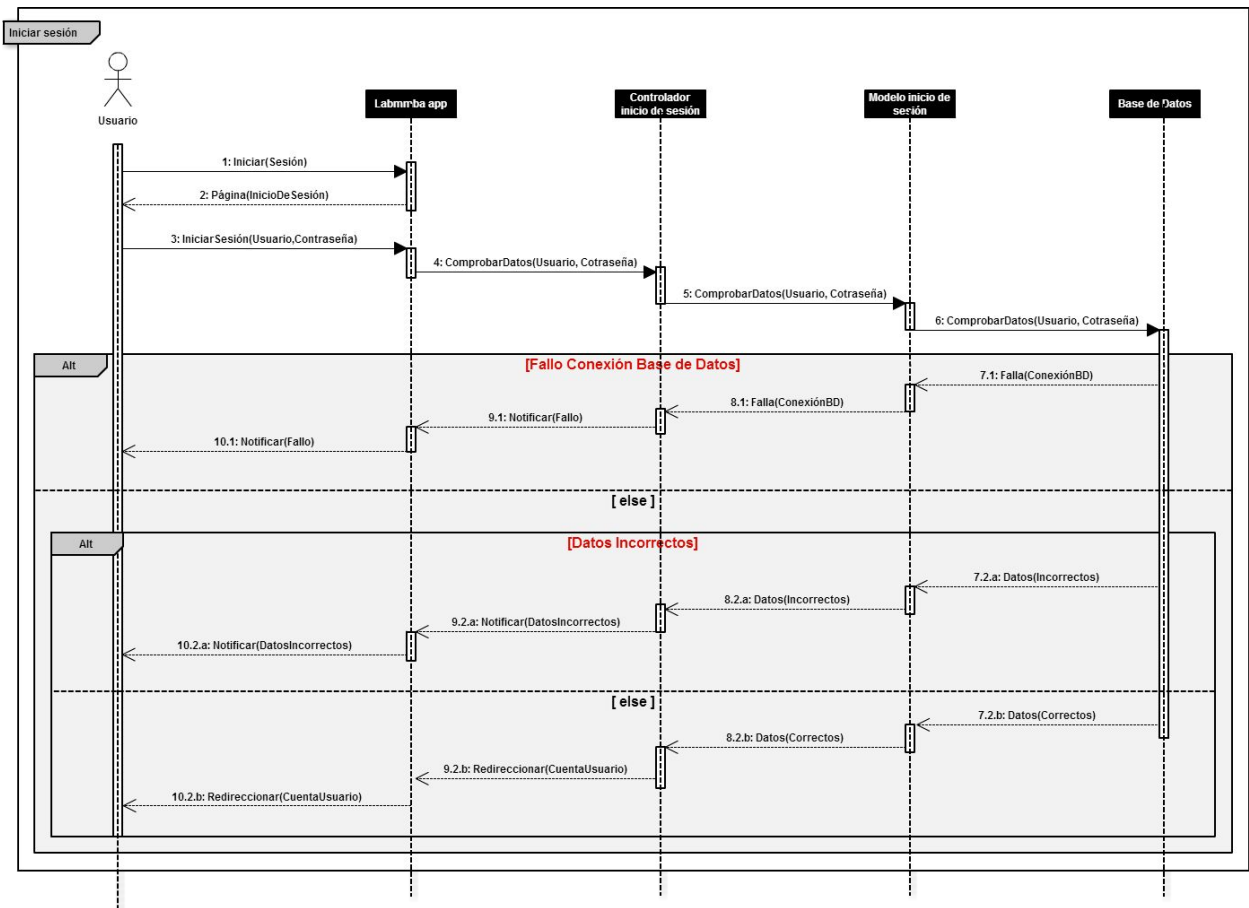


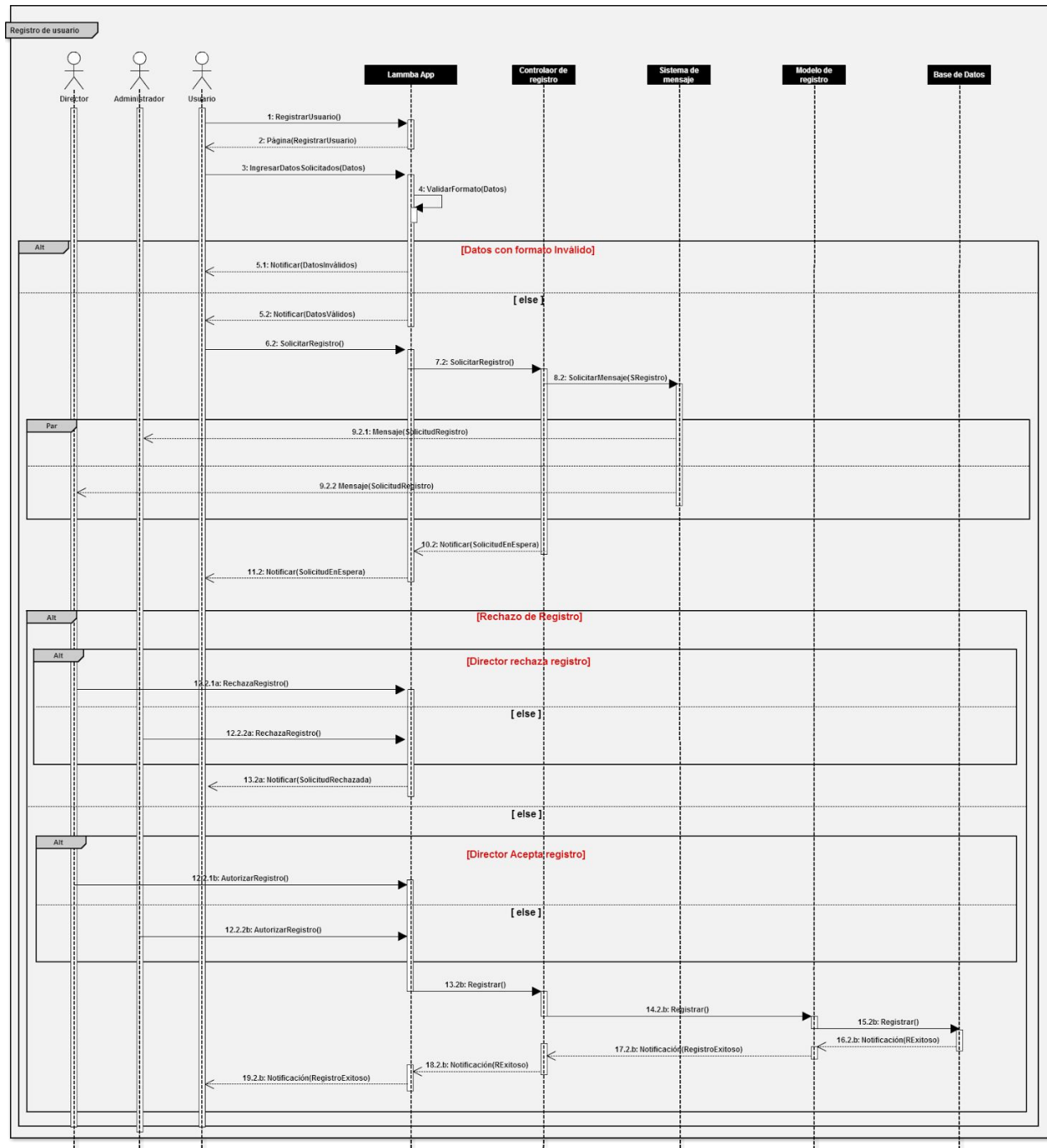


6- Modelo (ER) de la base de datos



7- Diagrama de secuencia de componentes del sistema

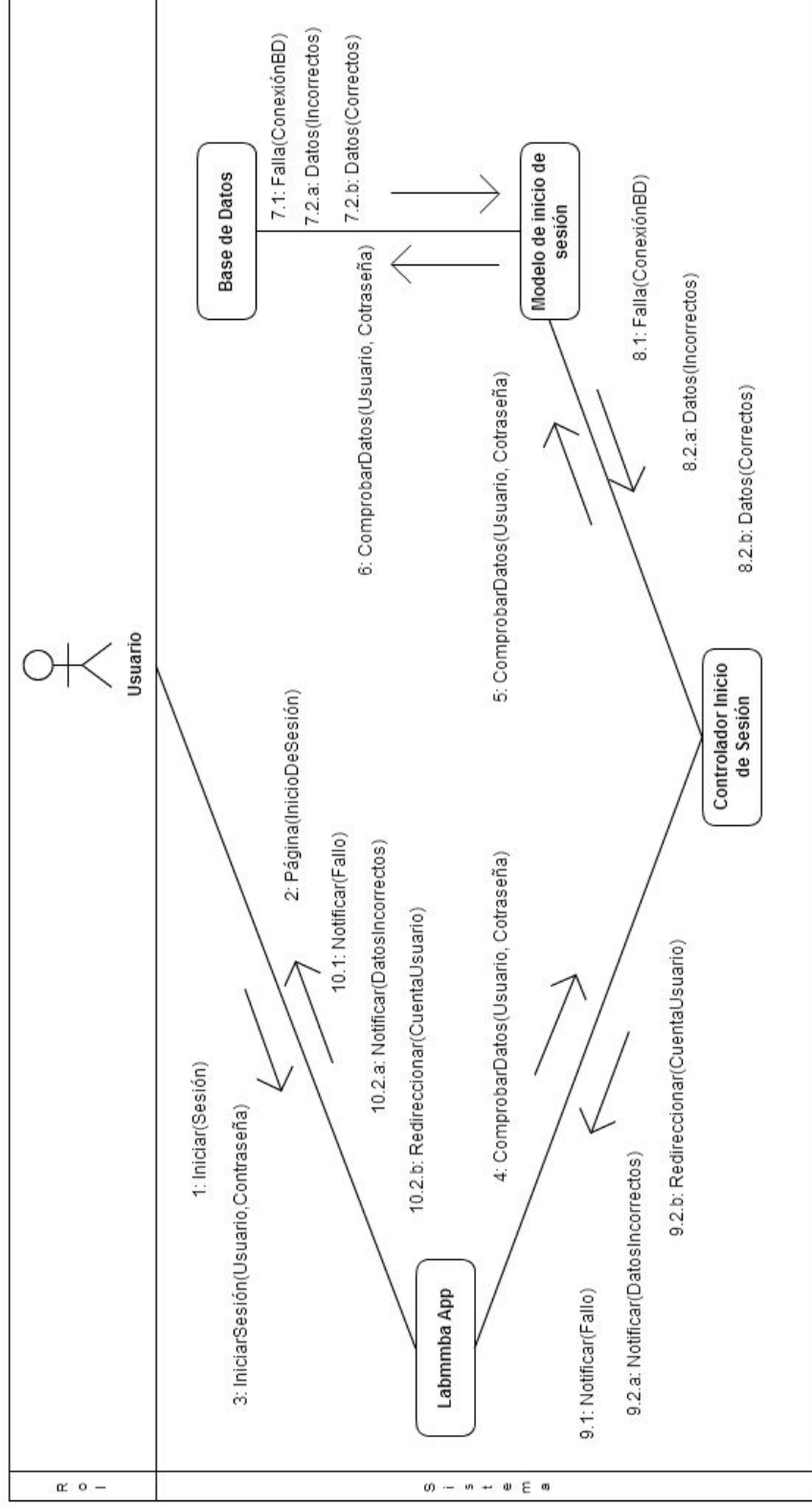




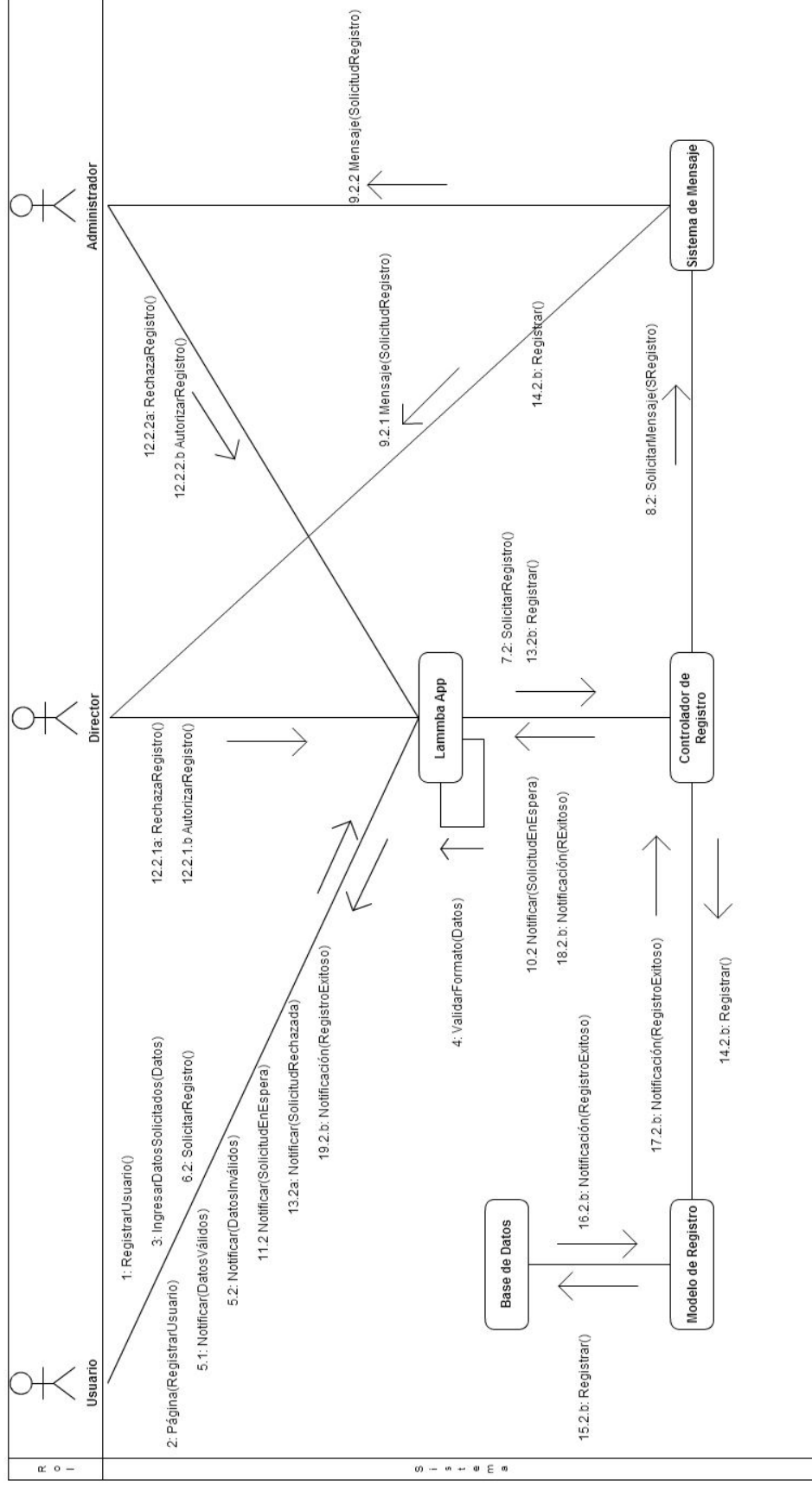
Nota Importante [Rechazo de Registro] se efectúa para rechazar el registro del usuario, dentro de los cuales existen 2 casos alternativos de rechazo ya sea por el director [Director rechaza registro] o el administrador [else: Administrador rechaza registro]

8- Diagrama de comunicación

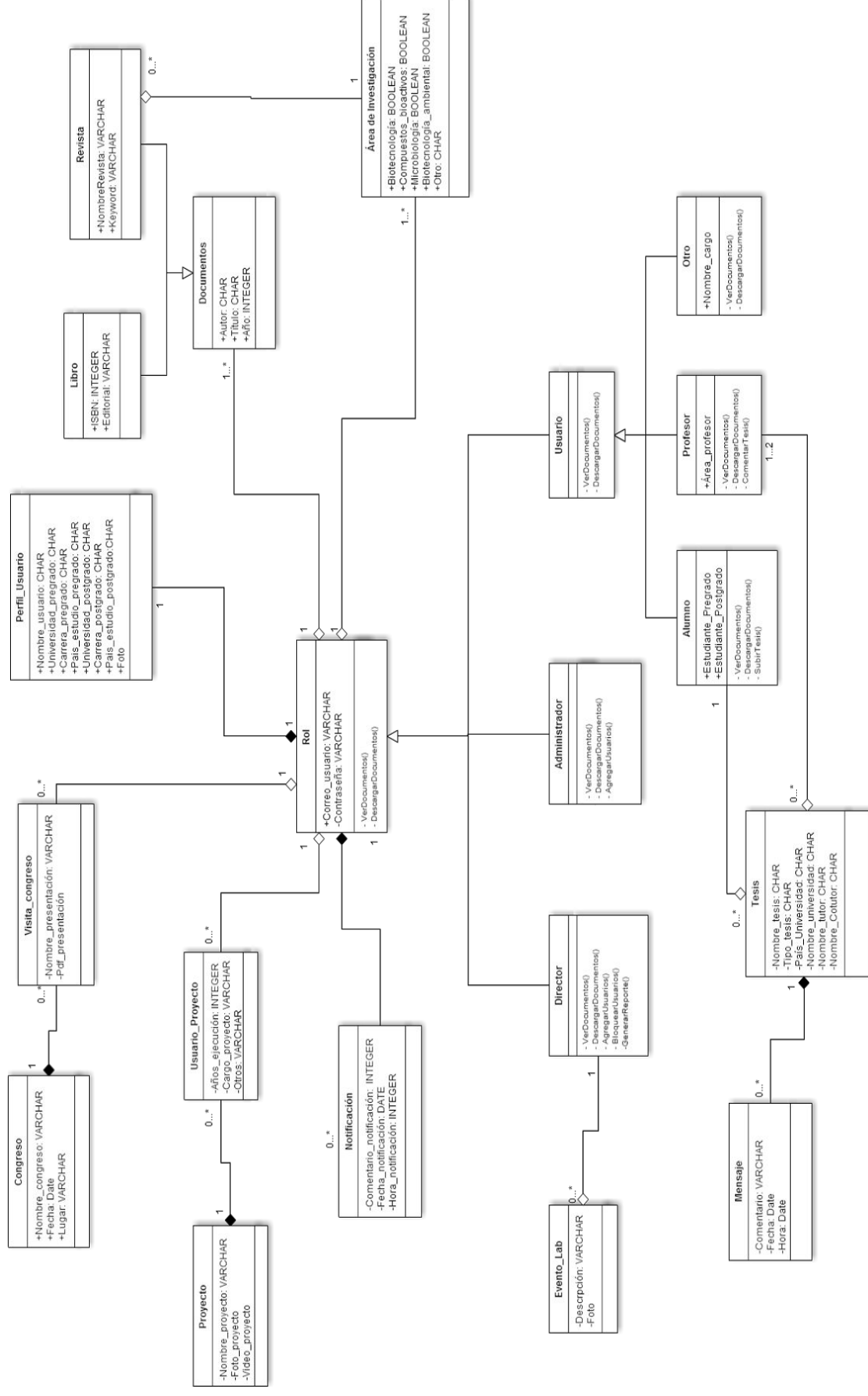
Iniciar sesión



Registro de usuario



9- Diagrama de clases



10- Bosquejo MVC

Vista

- congress
- books
- includes
- layouts
- login
- magazines
- proyectos
- register
- role
- tesisPostgrade
- tesisPregrade
- user
- userProfile
- adminControlPanel

Controlador

- CongressController
- BooksController
- MagazinesController
- ProjectController
- RegisterController
- RoleController
- UserController
- TesisPostgradeController
- TesisPregradeController
- UserProfileController
- AdminControlPanelController

Modelo

- Congress
- Books
- Magazines
- Proyectos
- Role
- User
- UserRole
- TesisPostgrade
- TesisPregrade

- UserProfile
- AdminControlPanel

Explicación del mvc para el cliente

Modelo: Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación. Envía a la “vista” aquella parte de la información que en cada momento se le solicita para que sea mostrada. Las peticiones de acceso o manipulación de información llegan al “modelo” a través del “controlador”.

Controlador: Responde a eventos (usualmente acciones del usuario) e invoca peticiones al “modelo” cuando se hace alguna solicitud sobre la información, por tanto se podría decir que el “controlador” hace de intermediario entre la “vista” y el “modelo”.

Vista: Presenta el “modelo” en un formato adecuado para interactuar (*interfaz de usuario*) por tanto requiere de dicho “modelo” la información que debe representar como salida.

11- Comentarios sobre el reporte de revisión I

A modo general las observaciones fueron de gran utilidad para mejorar aspectos del entregable 2, ya que, por ejemplo, en las observaciones hechas en los requerimientos el simple cambio de ciertas palabras hacían más entendible la información que deseábamos entregar. Lo mismo sucede para las observaciones en los casos de uso breves. En cuanto a los casos de uso extendido, no fuimos cuidadosos, y se nos olvidaron ciertos detalles que ya fueron mejorado, lo mismo va para los requerimientos no funcionales.

Cabe mencionar que habían otras observaciones que se escapaban del caso de nuestro documento y nos generaron cierta extrañeza como por ejemplo “Se debe verificar con mayor detalle las estimaciones”, con lo cual a nuestro parecer ya estaban más que verificadas y detalladas, siempre considerando que se trataba de una persona totalmente inexperta y que cuesta mucho tiempo solucionar errores, más que implementar un requerimiento, además en cada comentario se hacía la relación con los otros requerimientos. En cuanto a los diagramas no supimos cómo utilizar la herramienta, pero eso ya fue solucionado y ambos fueron rehechos; y en referencia al modelo E-R fue un detalle que se nos escapó, sin embargo los Boolean son necesarios pues son checkbox el cual usuario debe seleccionar, y fue el método que nosotros consideramos más óptimo para el caso.

12- Comentarios sobre el reporte de revisión II

Encontramos que se puso demasiado énfasis en el contenido del Informe I, en el cual reciclaron los mismos comentarios del reporte de revisión I, siendo que estos errores ya fueron justificados en los comentarios del informe, por lo cual creímos innecesario volverlos a explicar. Según la revisión del presente informe, considerando los nuevos items, encontramos que tienen razón en cuanto al diagrama 1 al faltarle la fecha que comunique a “lammba app” al usuario. En cuanto a los errores encontrados en el diagrama 2:

- Los pasos 1 y 2 se refieren a que dentro de la aplicación se despliega la página donde se encuentra el registro de usuario, pero entiendo que el nombre causara confusión, el cual será modificado.
- El validarFormato(Datos) lo habíamos considerado como que la misma aplicación verifica rápidamente si falta un campo, si lo llevásemos al controlador, también deberíamos llevarlo al modelo para que implementara la “lógica de negocios”, en cambio nosotros lo dejamos en la vista para que el código php se encargara de verificar eso, pero si es que se quería ver que aquellos datos eran verídicos, el director o adminitrador revisarían eso.
- Por último, si se guardaran directamente en la Base de Datos, estos ya estarían registrados, por lo cual si el director o el administrador rechazara el registro sería más engorroso eliminarlo de la Base de Datos y hacer los accesos restringidos para que todavía no fuesen usuarios validados, para evitar aquello decidimos guardar aquellos datos en una memoria caché, y de esa manera puedan ver los datos.