

SUBSECRETARÍA DE EDUCACIÓN SUPERIOR

DIRECCIÓN GENERAL DE EDUCACIÓN SUPERIOR  
TECNOLÓGICA



SECRETARÍA DE  
EDUCACIÓN PÚBLICA



**SEP**

**TRABAJO PROFESIONAL**

**COMO REQUISITO PARA OBTENER EL TITULO DE:**

**INGENIERO EN SISTEMAS  
COMPUTACIONALES**

**QUE PRESENTA:**

**VÍCTOR ESTEBAN GONZÁLEZ PALOMEQUE**

**CON EL TEMA:**

**“HERRAMIENTA CASE PARA LA  
GENERACIÓN DEL DOCUMENTO DE  
ESPECIFICACIÓN DE REQUISITOS EN  
PROYECTOS DE SOFTWARE”**

**MEDIANTE:**

**T. I.**

**(TITULACIÓN INTEGRAL)**

TUXTLA GUTIERREZ, CHIAPAS

OCTUBRE 2014

## Índice

|  |    |
|--|----|
| 1. Introducción.....   | 5  |
| 2. Planteamiento del problema.....   | 7  |
| 3. Estado del arte .....   | 8  |
| 3.1 OSRMT (Open SourceRequirements Management Tool) .....  | 8  |
| 3.2 CALIBER-RM .....   | 8  |
| 3.3 DOORS .....  | 8  |
| 3.4 REQUISITEPRO .....   | 8  |
| 3.5 REM .....  | 9  |
| 4. Justificación.....  | 10 |
| 5. Objetivos.....  | 11 |
| 5.1 Objetivo General .....   | 11 |
| 5.2 Objetivos específicos .....  | 11 |
| 6. Caracterización del área en que se participó .....  | 12 |
| 6.1 Datos generales de la organización.....  | 12 |
| 6.2 Misión.....  | 13 |
| 6.3 Visión .....   | 13 |
| 6.4 Organigrama.....   | 13 |
| 6.5 Croquis de ubicación.....  | 14 |
| 6.6 Datos generales del área en donde participó.....   | 14 |
| 7. Problemas a resolver. ....  | 16 |
| 8. Alcances y limitaciones .....   | 17 |
| 9. Fundamento Teórico .....  | 18 |
| 9.1 Ingeniería de Requisitos .....   | 18 |
| 9.1.1 Procesos de Ingeniería de Requisitos.....  | 19 |
| 9.2 Documento de Especificación de Requisitos .....  | 20 |
| 9.2.1 Objetivos de la Especificación de Requisitos del Software .....                                    | 21 |
| 9.2.2 Características de una Especificación de Requisitos del Software.....                              | 22 |
| 9.2.3 Formato de la Especificación de Requisitos del Software definida en el<br>IEEE/ANSI 830-1998. .... | 23 |
| 9.2.4 Usuarios de la Especificación de Requisitos del Software .....                                     | 24 |
| 9.3 Modelos de desarrollo.....   | 24 |

|   |    |
|---|----|
| 9.3.1 El modelo de desarrollo en cascada .....                      | 24 |
| 9.4 Herramienta CASE .....  | 26 |
| 9.5 Tecnologías .....   | 27 |
| 9.5.1 Bases de datos.....   | 27 |
| 9.5.2 Sistema Manejador de Base de Datos. ....                      | 27 |
| Extensible MarkupLanguage.....                                      | 28 |
| 9.6 Modelos.....  | 30 |
| 9.6.1 UML .....   | 30 |
| 10. Procedimiento y descripción de las actividades realizadas ..... | 31 |
| 10.1 Análisis y definiciones de requerimientos.....                 | 32 |
| 10.1.1 Arquitectura del sistema.....                                | 32 |
| 10.1.2 Funciones del sistema.....                                   | 34 |
| 10.1.3 Usuarios principales.....                                    | 36 |
| 10.2 Diseño del sistema y software.....                             | 37 |
| 10.2.1 Aplicación web.....  | 37 |
| 10.2.2 Base de datos.....   | 38 |
| 10.2.3 Interfaces de Usuario.....                                   | 38 |
| 10.3 Desarrollo.....  | 38 |
| 10.3.1 Base de datos.....   | 38 |
| 10.3.2 Aplicación web.....  | 39 |
| 10.4 Pruebas.....   | 39 |
| 10.4.1 Registro de usuarios.....                                    | 40 |
| 10.4.2 Login de Usuario .....                                       | 42 |
| 10.4.3 Crear nuevo documento (proyecto).....                        | 43 |
| 10.4.4 Captura de documento.....                                    | 45 |
| 10.4.4.1 Datos generales del proyecto .....                         | 45 |
| 10.4.4.2 Introducción .....   | 45 |
| 10.4.5. Generación de documento PDF .....                           | 55 |
| 11. Resultados, planos, gráficos, prototipos y programas .....      | 57 |
| 11.1 Diagrama de casos de uso.....                                  | 57 |
| 11.2 Plantillas de caso de uso .....                                | 58 |
| 11.2.1 Casos de uso Alumno .....                                    | 58 |

|   |    |
|---|----|
| 11.2.2 Casos de uso Maestro .....                 | 63 |
| 11.2 Diagrama Entidad-Relación .....              | 67 |
| 11.2.1 Diseño entidad relación corte 1.....       | 68 |
| 11.2.2 Diseño entidad relación corte 2.....       | 69 |
| 11.2.3 Diseño entidad relación corte 3.....       | 70 |
| 11.2.4 Diseño entidad relación corte 4.....       | 71 |
| 11.2.5 Diseño entidad relación corte 5.....       | 71 |
| 11.3 Diagrama de bloque de la plataforma web..... | 72 |
| 11.4 Diagrama de despliegue.....                  | 77 |
| 12. Conclusiones y recomendaciones .....          | 78 |
| 13. Referencias.....                              | 79 |

## **1. Introducción**

En la actualidad, el uso de herramientas que faciliten o que automaticen un proceso, se ha vuelto cada vez más necesario en diferentes ámbitos tanto empresariales como educativos. Asimismo, la necesidad de conseguir información mediante la práctica de los diferentes procesos que se llevan a cabo en una institución o empresa ha venido derivando en que debe existir en la mayoría de lo posible retroalimentación que lleve al aprendizaje de nuevos conocimientos. Esto principalmente se ve reflejado en el ámbito educativo.

Resulta así, que la retroalimentación viene a ser fundamental al momento de querer lograr un mayor entendimiento tanto teórico como práctico de los diferentes temas que se enseñan y las diferentes materias y asignaturas que existen. Es por esto que el uso de herramientas interactivas que fomenten la retroalimentación viene a ser muy útil para el correcto aprendizaje.

La carrera de Ingeniería en Sistemas Computacionales que se imparte en el Instituto Tecnológico de Tuxtla Gutiérrez en su retícula 2010 cuenta con diferentes asignaturas cuya finalidad es enseñar teórica y prácticamente las diferentes etapas del ciclo de vida de desarrollo de software. Estas explican tanto la teoría de los diferentes paradigmas y modelos, así como la elaboración y la aplicación de estos.

Las unidades II y III de la materia Fundamentos de Ing. De Software cubren la disciplina de la Ingeniería de Requisitos, teniendo como producto principal una especificación de Requisitos que los alumnos deben elaborar. Dicha especificación es un documento que contiene toda la información referente a las funciones que va a realizar el sistema, así como sus restricciones.

Se espera que el documento que genera el alumno contenga todos los elementos necesarios para una especificación de requisitos que sea completa,

y de acuerdo a un estándar, para que el alumno se familiarice con las exigencias reales que va a enfrentar en su vida profesional.

## **2. Planteamiento del problema**

La herramienta que se utiliza actualmente para elaborar el documento de Especificación de Requisitos es un editor de texto de uso general como Word.

Debido a que un editor de textos no contiene ayuda sobre lo que se va capturando en él, los trabajos que elaboran los estudiantes muchas veces no cumplen con las expectativas del profesor de la materia, porque manejan estilos diferentes en formato y muchas veces no se comprende bien lo que se espera en cada punto del documento. Así también dichos trabajos muchas veces no están apegados a un estándar tal como el 830 de la IEEE. Todo esto da como resultado un documento poco profesional.

Tomando en cuenta lo anterior, surge la necesidad de disponer de un software que sirva como herramienta de laboratorio de Ingeniería de Software, para obtener resultados de mejor calidad.

### **3. Estado del arte**

Hasta hace poco tiempo las herramientas para la gestión de requisitos de software se limitaban a editores de texto, los cuales hacían de esta tarea una labor tediosa y confusa. Actualmente, se cuenta con múltiples opciones, tras una búsqueda en diversos artículos sobre herramientas CASE que sirven de apoyo a la Ingeniería de Requisitos y que tienen como característica generar alguna clase de documentación sobre los requisitos tratados en un proyecto se encontraron entre las más utilizadas las que se mencionan a continuación:

#### **3.1 OSRMT (Open SourceRequirements Management Tool)**

Esta es una herramienta muy general para la gestión de requisitos, es mencionada porque es capaz de generar documentación de los requisitos tratados, pero dicha documentación no es la Especificación de Requisitos, es solo una referencia. La documentación que genera puede ser exportada en formato XML y HTML

#### **3.2 CALIBER-RM**

Al igual que OSRMT está diseñado para capturar y manejar requerimientos. De manera que estos se puedan cumplir, pero nos encontramos nuevamente con la limitante de que la documentación que este genera solo abarca información sobre la clasificación de los requisitos y sus posibles cambios, tiene la capacidad de exportar estos documentos en formato Word.

#### **3.3 DOORS**

DOORS (DynamicObjectOrientedRequirementsSystem) es una base de datos que permite el almacenamiento estructurado y la administración de requisitos. La ventaja de esta herramienta es que precisamente utiliza bases de datos, y facilita al usuario el análisis de los requisitos, pero al igual que CALIBER-RM, la documentación se enfoca en describir de manera simple los requisitos, y da la posibilidad de exportar el documento en un formato Word.

#### **3.4 REQUISITEPRO**

Esta es quizá la herramienta más “amigable” en cuanto al manejo de requisitos, y la gestión de ellos, ofrece la posibilidad de generar documentación más

detallada que las herramientas mencionadas anteriormente pero es de difícil acceso para un estudiante, pues la licencia ronda casi los dos mil seiscientos dólares.

### 3.5 REM

REM (REquirements Management) es una herramienta experimental gratuita de Gestión de Requisitos diseñada para soportar la fase de Ingeniería de Requisitos de un proyecto de desarrollo software. Permite exportar información de requisitos en un formato HTML.

Tabla 3.1 Tabla Comparativa de las herramientas existentes.

| CARACTERÍSTICAS   | HERRAMIENTAS UTILIZADAS                           |  |   |   |   |
|---|---|--|---|---|---|
|   | CALIBER-RM  | DOORS                                  | REQUISITEPRO                                      | REM   | OSRMT   |
| <b>Facilidad para importación/exportar requisitos</b>                     | Si (formato Word)                                 | Si (formato Word)                      | Si (formato Word y CSV)                           | Si (formato HTML)                                 | Si (formato XML y HTML)                           |
| <b>Posibilidad de clasificar requisitos</b>                               | Sí, definen tipos de requisitos                   | Si, se definen requisitos como objetos | Si, se definen tipos de requisitos                | Si, requisitos estándar                           | Si, se definen tipos de requisitos                |
| <b>Posibilidad de personalizar el modelo de requisitos</b>                | No  | No                                     | No  | No  | No  |
| <b>Soporte selección y reutilización de requisitos en otros proyectos</b> | No. Sólo copiar y pegar dentro del mismo proyecto | No. Sólo copiar y pegar módulos        | No. Sólo copiar y pegar dentro del mismo proyecto | No. Sólo copiar y pegar dentro del mismo proyecto | No. Sólo copiar y pegar dentro del mismo proyecto |
| <b>Manejo de versiones</b>  | Si  | Si                                     | Si  | No  | No  |
| <b>Control de acceso</b>  | Si, grupos y usuarios                             | Si, grupos y usuarios                  | Si, grupos y usuarios                             | No  | Si, grupos y usuarios                             |
| <b>Práctica y sencilla de utilizar. Interfaz amigable</b>                 | Alta  | Baja                                   | Alta  | Alta  | Alta  |
| <b>Licenciamiento libre</b>   | No  | No                                     | No  | Si  | Si  |
| <b>Facilidad para generación de modelo de diseño</b>                      | No  | No                                     | No  | No  | No  |

## 4. Justificación

La implementación de una herramienta case para la generación del documento de especificación de requisitos conlleva múltiples beneficios que se pueden apreciar principalmente en tres áreas.

El alumno se ve beneficiado ya que al contar con una herramienta de soporte para su trabajo permitiría que este elabore un trabajo ordenado y de mayor calidad. Así también permitiría que los trabajos se lleven a cabo con una mayor organización y control sobre quién realiza cada parte del documento ya que por las características del sistema se requiere un acceso único a cada miembro del proyecto. Pero el principal beneficio para el alumno es que no irá a “ciegas” realizando el trabajo sino que tendrá una guía para poder realizar un trabajo de calidad.

También para el profesor esta herramienta es beneficiosa ya que le permitirá observar de mejor manera el trabajo de cada uno de sus alumnos y saber quién está o no trabajando en cada parte del documento. Así también el sistema brinda la facilidad de hacer revisiones directamente en la plataforma y realizar las observaciones pertinentes. Por último el profesor será capaz de elegir los elementos que contendrá la plantilla sobre la que trabajarán los alumnos de acuerdo a los temas vistos en clase.

La calidad del trabajo se ve beneficiada ya que este estará más completo porque la herramienta guiará en los puntos que debe contener y la ayuda para completarlos. El trabajo es uniforme como consecuencia de que se respeta un estándar (IEEE 830).

## **5. Objetivos**

### **5.1 Objetivo General**

Desarrollar una herramienta de apoyo al proceso de Ingeniería de Requisitos, enfocada a la elaboración y generación del documento de Especificación de Requisitos, que sirva para el desarrollo de prácticas para las materias de Ing. de Software de la carrera de I.S.C.

### **5.2 Objetivos específicos**

- Obtener el diseño de la arquitectura del sistema.
- Diseñar las diferentes plantillas de especificación de requisitos, a partir del estándar 830-IEEE.
- Diseñar las interfaces de usuario
- Elaborar la información de ayuda para el usuario e integrarla a la herramienta
- Desarrollar e implementar la base de datos del sistema
- Desarrollar e implementar la herramienta y realizar las pruebas pertinentes generando una especificación de requisitos como prueba piloto

## **6. Caracterización del área en que se participó**

### **6.1 Datos generales de la organización**

En la década de los 70's, se incorpora el estado de Chiapas al movimiento educativo nacional extensión educativa, por intervención del Gobierno del Estado de Chiapas ante la federación.

Esta gestión dio origen a la creación del Instituto Tecnológico Regional de Tuxtla Gutiérrez (ITRTG) hoy Instituto Tecnológico de Tuxtla Gutiérrez (ITTG).

El día 23 de agosto de 1971 el Gobernador del Estado, Dr. Manuel Velasco Suárez, colocó la primera piedra de lo que muy pronto sería el Centro Educativo de nivel medio superior más importante de la entidad.

El día 22 de octubre de 1972, con una infraestructura de 2 edificios con 8 aulas, 2 laboratorios y un edificio para talleres abre sus puertas el Instituto Tecnológico de Tuxtla Gutiérrez con las carreras de Técnico en Máquinas de Combustión Interna, Electricidad, Laboratorista Químico y Máquinas y Herramientas.

En el año 1974 dio inicio la modalidad en el nivel superior, ofreciendo las carreras de Ingeniería Industrial en Producción y Bioquímica en Productos Naturales.

En 1980 se amplió la oferta educativa al incorporarse las carreras de Ingeniería Industrial Eléctrica e Ingeniería Industrial Química.

En 1987 se abre la carrera de Ingeniería en Electrónica y se liquidan en 1989 las carreras del sistema abierto del nivel medio superior y en el nivel superior se reorientó la oferta en la carrera de Ingeniería Industrial Eléctrica y se inicia también Ingeniería Mecánica.

En 1991 surge la licenciatura en Ingeniería en Sistemas Computacionales.

Desde 1997 el Instituto Tecnológico de Tuxtla Gutiérrez ofrece la Especialización en Ingeniería Ambiental como primer programa de postgrado.

## 6.2 Misión

Formar de manera integral profesionales de excelencia en el campo de la ciencia y la tecnología con actitud emprendedora, respeto al medio ambiente y apego a los valores éticos.

## 6.3 Visión

Ser una Institución de excelencia en la educación superior tecnológica del Sureste, comprometida con el desarrollo socioeconómico sustentable de la región.

## 6.4 Organigrama

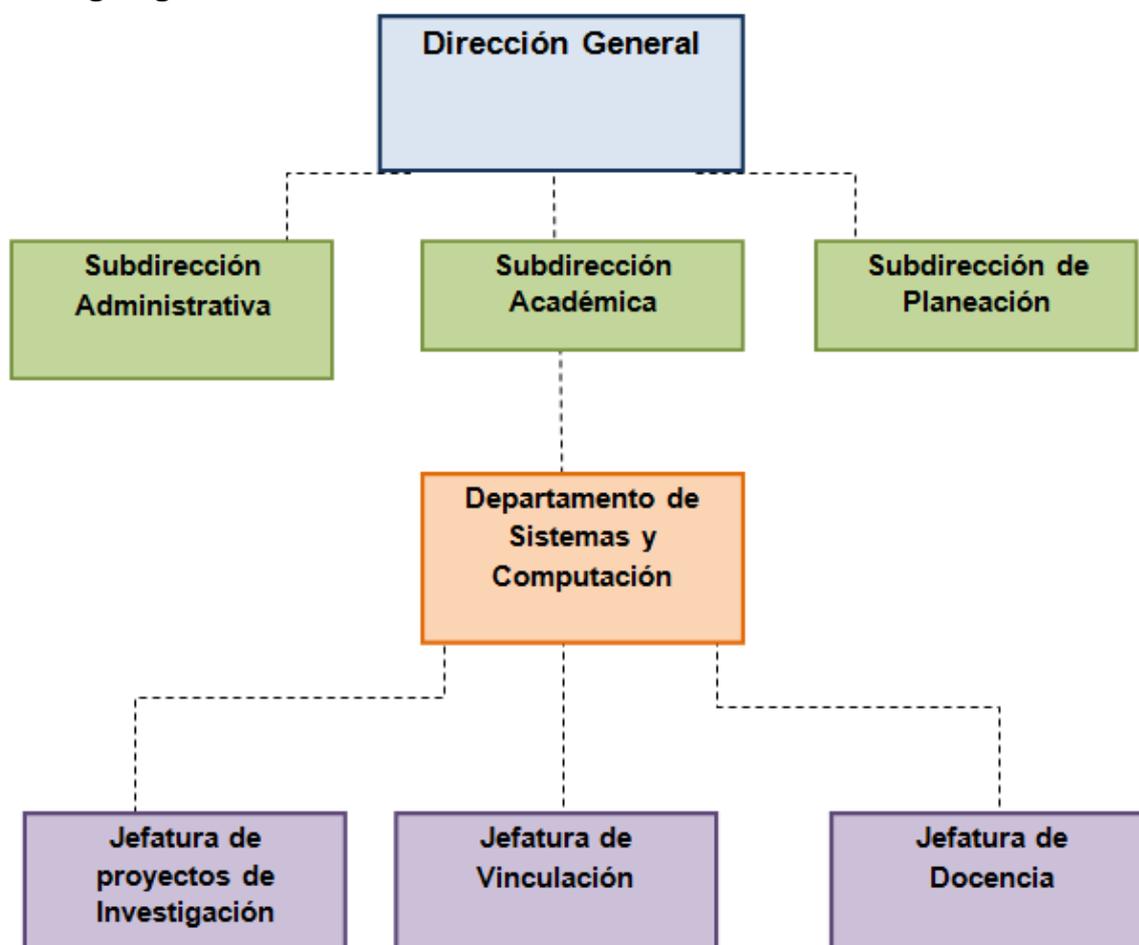


Ilustración 6.4 Organigrama de la Institución

## 6.5 Croquis de ubicación

Ubicación del Instituto Tecnológico de Tuxtla Gutiérrez

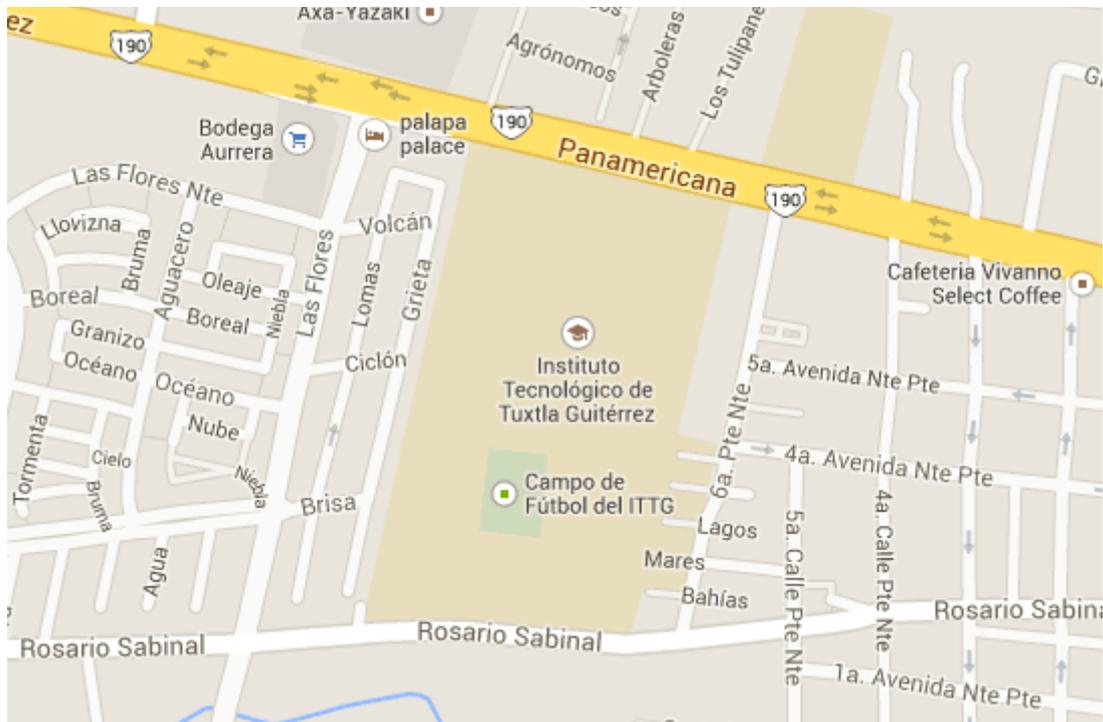


Imagen 6.5 Ubicación del ITTG

El Instituto Tecnológico de Tuxtla Gutiérrez se ubica en la ciudad de Tuxtla Gutiérrez, Chiapas, en la carretera Panamericana km. 1080, C.P. 29000.

## 6.6 Datos generales del área en donde participó

El objetivo que tiene la carrera de Ingeniería en Sistemas Computacionales es formar profesionales capaces de diseñar y desarrollar sistemas de software que les permitan propiciar el fortalecimiento de la tecnología nacional; administrar proyectos de desarrollo de software, especificar y evaluar configuraciones de sistemas de cómputo en todo tipo de organizaciones donde se utilicen sistemas computacionales.

Las actividades que se llevan a cabo en la línea de investigación de Tecnologías de la Información y Base de Datos del departamento de Sistemas Computacionales son las siguientes:

- Esta línea se encarga de promover la investigación en el área de sistemas computacionales
- Crea tecnologías basadas en la información y base de datos
- Apoya a las diversas asignaturas impartidas en la carrera de Sistemas Computacionales con tecnología y herramientas de buena calidad.

## 7. Problemas a resolver.

- Desarrollar el módulo para la captura de requisitos mediante una interfaz amigable y poco tediosa
- Implementar métodos o técnicas de exportación de un texto sin formato a un formato PDF.
- Desarrollar un algoritmo para la maquetación del documento generado en formato PDF.
- Desarrollar un módulo dentro de la herramienta que permita controlar los permisos de los usuarios que interactúan con esta.
- Lograr además de la captura de texto, capturar imágenes que complementen la descripción de los requisitos y las secciones del documento.
- Desarrollar el módulo que permita realizar observaciones por parte del profesor y editarlas
- Implementar clases o métodos que permitan la validación de los formularios
- Desarrollar un módulo que permita gestionar tanto los grupos como los equipos de trabajo por parte de nuestros dos usuarios principales.
- Realizar una prueba piloto que tenga como resultado un documento de especificación de requisitos realizado con la herramienta
- Montar la herramienta en un servidor para futuras pruebas

## 8. Alcances y limitaciones

La herramienta está diseñada para usarse como apoyo a las materias de ingeniería de software en el área de sistemas computacionales del ITTG.

La herramienta es una guía que reúne y concentra una serie de prácticas recomendadas para una buena elaboración del ERS, sin embargo algunos de los puntos que contiene pueden o no ser requeridos para un proyecto en específico, esto dependerá del contexto y el criterio de los que desarrollan dicho proyecto.

La herramienta tiene como funcionalidad principal capturar requisitos y exportarlos a un formato PDF ya que ese fue el formato definido dentro de los requisitos. La herramienta NO puede exportar a otro formato.

La herramienta NO “genera automáticamente” una ERS, da al usuario una serie de pautas y pasos a seguir para realizarla y automatiza el proceso de generación de esta en el sentido de la maquetación y las partes del documento. El usuario debe seguir llenando los campos del documento de acuerdo a su criterio.

La herramienta permite gestionar grupos por parte del maestro y equipos de trabajo por parte del alumno para trabajar en el documento de especificación de requisitos.

La herramienta NO puede ser usada por un usuario no registrado en el sistema.

La herramienta permite dar alta nuevos usuarios por parte de los maestros. Los permisos dentro de la herramienta están fijos.

La herramienta actualmente no permite la concurrencia de usuarios en la misma sección del documento al mismo tiempo.

## **9. Fundamento Teórico**

### **9.1 Ingeniería de Requisitos**

Ingeniería de Requisitos tiene que ver con aquellas actividades en pos de entender exactamente las necesidades de los usuarios de un Sistema de Software y traducir tales necesidades a un conjunto de sentencias precisas, no ambiguas, las cuales serán usadas para el desarrollo del Sistema [Loucopoulos, 1995].

El ciclo de vida de desarrollo de software es considerado como una secuencia de procesos. Un Proceso es una serie de pasos que incluye actividades, restricciones y recursos para producir un determinado resultado esperado.

Los procesos son importantes porque imponen consistencia y estructura sobre un conjunto de actividades, manteniendo un nivel de consistencia y calidad en los productos o servicios que son producidos [Lawrence , 2002].

Cada proceso puede ser escrito en una variedad de formas utilizando texto, gráficos o una combinación de ambos. Además, todo proceso debe ser adaptado para la situación especial en que será utilizado.

Antes de planificar un proyecto de software, se deberían establecer sus objetivos y su ámbito de aplicación, considerando soluciones alternativas e identificando tanto dificultades técnicas como de gestión. El desarrollador de software y el cliente deben reunirse para definir los objetivos del proyecto y su ámbito. En muchos casos, esta actividad comienza como parte del proceso de ingeniería del sistema y continúa como el primer paso en el análisis de los requisitos del software [Pressman, 1998].

Los objetivos identifican las metas generales del proyecto sin considerar como se conseguirán. Una vez que se han comprendido los objetivos y el ámbito del proyecto, se consideran las soluciones alternativas [Pressman,

1998]. Además, los objetivos establecidos permiten posteriormente formular los requisitos del Sistema de Software.

El análisis de estos requisitos es una de las tareas más importantes en el ciclo de vida del desarrollo de software, puesto que en ella se determinan los “planos” de la nueva aplicación.

En cualquier proyecto software los requisitos son las necesidades del producto que se debe desarrollar. Por ello, en la fase de análisis de requisitos se deben identificar claramente estas necesidades y documentarlas. Como resultado de esta fase se debe producir un documento de especificación de requisitos en el que se describa lo que el futuro sistema debe hacer. Por tanto, no se trata simplemente de una actividad de análisis, sino también de síntesis.

### **9.1.1 Procesos de Ingeniería de Requisitos**

#### ***9.1.1.1 Elicitación de Requisitos***

En la mayoría de los casos, al comenzar un proyecto de software, el analista conoce muy poco acerca del problema a resolver. La elicitación de requisitos es el proceso que consiste en adquirir todo el conocimiento relevante, necesario para producir un modelo de requisitos (especificación) de un dominio de problema [Thomas , 2005].

El analista debe realizar la especificación de requisitos y subsecuentemente su validación con el usuario, solamente después de comprender la naturaleza, características y límites de un problema.

#### ***9.1.1.2 Especificación de Requisitos***

Una especificación puede ser vista como un contrato entre usuarios y desarrolladores de software, el cual define el comportamiento funcional del artefacto de software (y otras propiedades tales como performance, confiabilidad, seguridad, etc.), sin mostrar cómo será obtenido ese comportamiento [Thomas , 2005].

Es importante ver la especificación como un proceso complejo que requiere “feedback” desde el analista al usuario y viceversa. La especificación de requisitos es el proceso central de ingeniería de requisitos.

### **9.1.1.3 Validación de Requisitos**

El proceso de validación certifica la corrección del modelo/especificación de requisitos contra las intenciones de los usuarios (por lo que la participación de los usuarios es crucial). Validar, luego de la fase de especificación de requisitos, puede ayudar a evitar costosas correcciones después del desarrollo [Thomas , 2005].

Otra definición [Loucopoulos , 1995], indica que la validación establece y justifica la convicción del analista y del usuario, de que el modelo de requisitos especifica una solución de software la cual es correcta para las necesidades del usuario.

## **9.2 Documento de Especificación de Requisitos**

El documento de requisitos del software, también conocido como Especificación de Requisitos del Software o ERS (SRS por sus siglas en inglés), es la declaración oficial de qué deben implementar los desarrolladores del sistema. Debe incluir tanto los requerimientos del usuario para el sistema como una especificación detallada de los requerimientos del sistema. En

algunos casos, los dos tipos de requerimientos se pueden integrar en una única descripción. En otros, los requerimientos del usuario definen una introducción a la especificación de los requerimientos del sistema [Sommerville , 2005].

Este documento es el resultado de analizar los requisitos; el análisis de requisitos se puede definir como el proceso del estudio de las necesidades de los usuarios para llegar a una definición de los requisitos del sistema, hardware o software, así como el proceso de estudio y refinamiento de dichos requisitos, definición proporcionada por el IEEE [Piattini, 1996], es decir, definir los flujos de información, las estructuras primarias de datos, las características funcionales del sistema, los requerimientos de rendimiento y las restricciones impuestas por el cliente, incorporar los criterios globales de validación [Pressman, 1998].

En la determinación de los requisitos no sólo deben actuar los analistas, es muy importante la participación de los propios usuarios, porque son éstos los que mejor conocen el sistema que se va a automatizar. Analista y cliente se

deben poner de acuerdo en las necesidades del nuevo sistema, ya que el cliente no suele entender el proceso de diseño y desarrollo del software como para redactar una especificación de requisitos software y los analistas no suelen entender completamente el problema del cliente, debido a que no dominan su área de trabajo.

Varios autores y organizaciones grandes, como el Departamento de Defensa de los Estados Unidos y el IEEE, han definido estándares para los documentos de requerimientos. En general este documento debe constar de [Gegovia, 2005]:

- **Introducción:** Describe los fines y objetivos del software a desarrollar.
- **Descripción de la Información:** Descripción detallada del problema que el software debe resolver. Estarán documentados el flujo y las estructuras de la información, así como el hardware, el software, y las interfaces hombre-máquina.
- **Descripción Funcional.** Proporciona una descripción de cada función requerida para resolver el problema.
- **Descripción del Comportamiento.** Describe el comportamiento del sistema software mediante un diagrama de estados, con una especificación de eventos y acciones.
- **Criterios de Validación.** Incorpora las clases de prueba que se deben realizar al software, los límites de rendimiento y la respuesta esperada.
- **Bibliografía y Apéndices.** En los apéndices se presentan las tablas de datos, la descripción detallada de los algoritmos, los diagramas, los gráficos, y otro material de interés.

### **9.2.1 Objetivos de la Especificación de Requisitos del Software**

Los principales objetivos que se identifican en la especificación de requisitos software son [Chalmeta, 1999]:

1. Ayudar a los clientes a describir claramente lo que se desea obtener mediante un determinado software: El cliente debe participar activamente en la especificación de requisitos, ya que éste tiene una visión mucho más detallada

de los procesos que se llevan a cabo. Asimismo, el cliente se siente partícipe del propio desarrollo.

2. Ayudar a los desarrolladores a entender qué quiere exactamente el cliente: En muchas ocasiones el cliente no sabe exactamente qué es lo que quiere. La ERS permite al cliente definir todos los requisitos que desea y al mismo tiempo los desarrolladores tienen una base fija en la que trabajar. Si no se realiza una buena especificación de requisitos, los costes de desarrollo pueden incrementarse considerablemente, ya que se deben hacer cambios durante la creación de la aplicación.

3. Servir de base para desarrollos de estándares de ERS particulares para cada organización: Cada entidad puede desarrollar sus propios estándares para definir sus necesidades.

### **9.2.2 Características de una Especificación de Requisitos del Software**

Las características deseables para una buena especificación de requisitos software que se indican en el IEEE son las siguientes [Chalmeta, 1999][Piattini, 1996]:

- **Correcta:** Implica que el sistema implementado será el sistema deseado.
- **No ambigua:** Un documento es no ambiguo si y solo si cada requisito tiene una única interpretación.
- **Completa:** Incluye todos los requisitos significativos del software, existe una definición de respuestas a todas las posibles entradas, tanto válidas como inválidas, en todas las posibles situaciones, cumple con el estándar utilizado y aparecen etiquetadas todas las figuras, tablas, diagramas, etc., así como definidos todos los términos y unidades de medida empleados.
- **Verificable:** Un requisito se dice que es verificable si existe algún proceso no excesivamente costoso por el cual una persona o una máquina puedan chequear que el software satisface dicho requerimiento.
- **Consistente:** Una ERS es consistente si y sólo si ningún conjunto de requisitos descritos en ella son contradictorios o entran en conflicto.

- Clasificada: No todos los requisitos son igual de importantes. Los requisitos pueden clasificarse por diversos criterios, lo ideal es el establecimiento de prioridades, de modo que la implementación de un requisito de menor prioridad no emplee excesivos recursos.
- Modificable: Una ERS es modificable si cualquier cambio puede realizarse de manera fácil, completa y consistente.
- Explorable: Una ERS es explorable si el origen de cada requerimiento es claro tanto hacia atrás (origen que puede ser
- un documento, una persona etc.) como hacia delante (componentes del sistema que realizan dicho requisito).
- Utilizable durante las tareas de mantenimiento y uso: La ERS actúa a modo de plano de la aplicación, permitiendo incluso modificaciones que no requieran un cambio en el diseño.

9.2.3 Formato de la Especificación de Requisitos del Software definida en el IEEE/ANSI 830-1998.

La siguiente figura muestra la estructura de la ERS propuesta por el IEEE en su estándar 830 [IEEE, 1998] [upm, 2000].

|          |  |
|----------|--|
| <b>1</b> | <b>Introducción</b>                    |
| 1.1      | Propósito                              |
| 1.2      | Ámbito del Sistema                     |
| 1.3      | Definiciones, Acrónimos y Abreviaturas |
| 1.4      | Referencias                            |
| 1.5      | Visión general del documento           |
| <b>2</b> | <b>Descripción General</b>             |
| 2.1      | Perspectiva del Producto               |
| 2.2      | Funciones del Producto                 |
| 2.3      | Características de los usuarios        |
| 2.4      | Restricciones                          |
| 2.5      | Suposiciones y Dependencias            |
| 2.6      | Requisitos Futuros                     |
| <b>3</b> | <b>Requisitos Específicos</b>          |
| 3.1      | Interfaces Externas                    |
| 3.2      | Funciones                              |
| 3.3      | Requisitos de Rendimiento              |
| 3.4      | Restricciones de Diseño                |
| 3.5      | Atributos del Sistema                  |
| 3.6      | Otros Requisitos                       |
| <b>4</b> | <b>Apéndices</b>                       |
| <b>5</b> | <b>Índice</b>                          |

Imagen 9.2.3 1 Estándar IEEE-830

## 9.2.4 Usuarios de la Especificación de Requisitos del Software

- Clientes y Usuarios.
- Analistas.
- Administradores
- Desarrolladores
- Ingenieros de sistemas.
- Ingenieros probadores de sistemas.
- Ingenieros encargados del mantenimiento del sistema.

## 9.3 Modelos de desarrollo

### 9.3.1 El modelo de desarrollo en cascada.

El primer modelo de proceso de desarrollo de software que se publicó se derivó de procesos de ingeniería de sistemas más generales (Royce, 1970). Este modelo se muestra en la ilustración 1. Debido a la cascada de una fase a otra,

dicho modelo se conoce como modelo en cascada o como ciclo de vida del software. Las principales etapas de este modelo se transforman en actividades fundamentales de desarrollo:

1. Análisis y definición de requerimientos. Los servicios, restricciones y metas del sistema, se definen a partir de las consultas con los usuarios. Entonces se definen en detalle y sirven como una especificación del sistema.
2. Diseño del sistema y del software. El proceso de diseño del sistema divide los requerimientos en sistemas hardware o software. Establece una arquitectura completa del sistema. El sistema de software identifica y describe las abstracciones fundamentales del software y sus relaciones.
3. Implementación y prueba de unidades. Durante esta etapa, el diseño del software se lleva a cabo como un conjunto o unidades de programas. La prueba de unidades implica verificar que cada una cumpla su especificación.
4. Integración y prueba del sistema. Los programas o las unidades individuales de programas se integran y prueban como un sistema completo para asegurar que se cumplan los requerimientos del software. Después de las pruebas, el sistema de software se entrega al cliente.
5. Funcionamiento y mantenimiento. Por lo general (aunque no necesariamente), ésta es la fase más larga del ciclo de vida. El sistema se instala y se pone en funcionamiento práctico. El mantenimiento implica corregir errores no descubiertos en las etapas anteriores del ciclo de vida, mejorar la implementación de las unidades del sistema y resaltar los servicios una vez que se descubren nuevos requerimientos.[Sommerville, 2005]

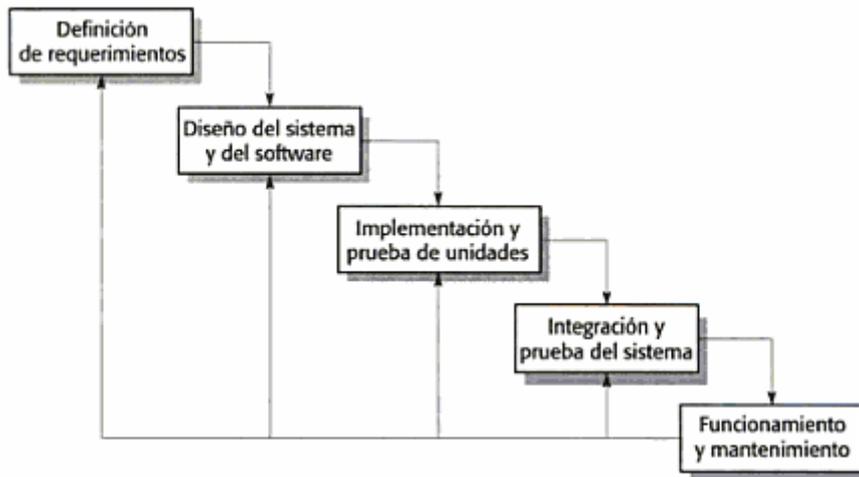


Imagen 9.3.1 Modelo de desarrollo en cascada

#### 9.4 Herramienta CASE

Se puede definir a las Herramientas CASE como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del Ciclo de Vida de desarrollo de un Software. Como es sabido, los estados en el Ciclo de Vida de desarrollo de un Software son: Investigación Preliminar, Análisis, Diseño, Implementación e Instalación [INEI, 1999].

CASE se define también como:

- Conjunto de métodos, utilidades y técnicas que facilitan la automatización del ciclo de vida del desarrollo de sistemas de información, completamente o en alguna de sus fases.
- La sigla genérica para una serie de programas y una filosofía de desarrollo de software que ayuda a automatizar el ciclo de vida de desarrollo de los sistemas.
- Una innovación en la organización, un concepto avanzado en la evolución de tecnología con un potencial efecto profundo en la organización. Se puede ver al CASE como la unión de las herramientas

automáticas de software y las metodologías de desarrollo de software formales.

## **9.5 Tecnologías**

### **9.5.1 Bases de datos.**

Es un conjunto de datos que es utilizado por los sistemas de aplicación de alguna empresa dada. El término de empresa es simplemente un término genérico para la identificación de cualquier organización independiente de tipo comercial, técnico, científico u otro [Flores, 2005].

Los tres componentes principales de un sistema de base de datos son el hardware, el software DBMS y los datos a manejar, así como el personal encargado del manejo del sistema.

- Campo:

Es la unidad más pequeña a la cual uno puede referirse en un programa. Desde el punto de vista del programador representa una característica de un individuo u objeto. Este puede ser de diferentes tipos de acuerdo a la información que va a contener, es decir, texto, Fecha, Entero, caracteres.

- Registro:

Colección de campos de iguales o de diferentes tipos.

- Administrador de base de datos (DBA):

Es la persona o equipo de personas profesionales responsables del control y manejo del sistema de base de datos, generalmente tiene experiencia en DBMS, diseño de bases de datos, Sistemas operativos, comunicación de datos, hardware y programación.

### **9.5.2 Sistema Manejador de Base de Datos.**

Un sistema de manejador de bases de datos (DBMS) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a ellos, es básicamente un sistema computarizado para llevar registros, es decir, un sistema cuya finalidad general es almacenar información y permitir a los usuarios realizar una variedad de operaciones sobre dichos registros, por

ejemplo: insertar, borrar, modificar, agregar, consultar. La colección de datos se denomina base de datos (BD) [Flores, 2005].

El objetivo primordial de un DBMS es proporcionar que a su vez sea conveniente y eficiente para ser utilizado al extraer o almacenar información en la BD.

Los sistemas de bases de datos están diseñados para gestionar grandes bloques de información, que implica tanto la definición de estructuras para el almacenamiento como de mecanismos para la gestión de la información.

### **Extensible MarkupLanguage.**

XML, siglas en inglés de eXtensibleMarkupLanguage, es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C). Deriva del lenguaje SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML) para estructurar documentos grandes. A diferencia de otros lenguajes, XML da soporte a bases de datos, siendo útil cuando varias aplicaciones se deben comunicar entre sí o integrar información [w3c, 2012].

XML no ha nacido sólo para su aplicación en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable.

XML es una tecnología sencilla que tiene a su alrededor otras que la complementan y la hacen mucho más grande y con unas posibilidades mucho mayores. Tiene un papel muy importante en la actualidad ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil.

### ***Ventajas de XML***

- Es extensible: Después de diseñado y puesto en producción, es posible extender XML con la adición de nuevas etiquetas, de modo que se pueda continuar utilizando sin complicación alguna.
- El analizador es un componente estándar, no es necesario crear un analizador específico para cada versión de lenguaje XML. Esto posibilita

el empleo de cualquiera de los analizadores disponibles. De esta manera se evitan bugs y se acelera el desarrollo de aplicaciones.

- Si un tercero decide usar un documento creado en XML, es sencillo entender su estructura y procesarla. Mejora la compatibilidad entre aplicaciones. Podemos comunicar aplicaciones de distintas plataformas, sin que importe el origen de los datos, es decir, podríamos tener una aplicación en Linux con una base de datos Postgres y comunicarla con otra aplicación en Windows y Base de Datos MS-SQL Server.
- Transformamos datos en información, pues se le añade un significado concreto y los asociamos a un contexto, con lo cual tenemos flexibilidad para estructurar documentos.
- PHP: PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante.
- HTML: HTML, siglas de HyperTextMarkupLanguage («lenguaje de marcas de hipertexto»), hace referencia al lenguaje de marcado para la elaboración de páginas web. Es un estándar que, en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, etc.
- JavaScript: JavaScript es un lenguaje de programación interpretado, se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas aunque existe una forma

de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

- **HTML2PDF:** HTML2PDF es un conversor de HTML a PDF escrito en PHP4 (utilizando FPDF), y PHP5 (utilizando TCPDF). Permite la conversión de HTML 4.01 válido en formato PDF, y se distribuye bajo licencia LGPL.

Esta biblioteca se ha hecho para ayudar en la creación de PDF.

- **CKEditor :**CKEditor es un editor de texto HTML/ WYSIWYG de código abierto que provee a la web del poder de las aplicaciones de escritorio al estilo de editores como Microsoft Word, sin la necesidad de instalar ningún componente en la computadora del cliente.

## 9.6 Modelos

### 9.6.1 UML

Lenguaje Unificado de Modelado (LUM o UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar,

especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados.

Es importante remarcar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional o RUP), pero no especifica en sí mismo qué metodología o proceso usar.

UML no puede compararse con la programación estructurada, pues UML significa Lenguaje Unificado de Modelado, no es programación, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, la programación orientada a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos. [Fowler y Scott, 1999]

## **10. Procedimiento y descripción de las actividades realizadas**

En esta parte se explicarán y detallarán cada una de las actividades realizadas para llevar a cabo el proyecto. Como marco de referencia tomamos el modelo de desarrollo en cascada ya que es el que se acopla en mayor medida con las actividades que se programaron.

## 10.1 Análisis y definiciones de requerimientos.

Para poder llevar a cabo el análisis nos apoyamos de técnicas como la lluvia de ideas y pláticas con nuestro asesor para definir cuáles eran los componentes que se iban a entregar, el alcance de la herramienta y las tecnologías para desarrollarlo.

En esta etapa se analizaron y definieron cada uno de los requerimientos del sistema. Como resultado de esta etapa, se obtuvo el documento de especificación de requisitos ERS de la Herramienta CASE para la generación de la Especificación de Requisitos en proyectos de Software.

Las principales partes que se definieron dentro de esta etapa fueron las siguientes.

### 10.1.1 Arquitectura del sistema.

Al ser parte de un sistema integrado principalmente por dos módulos que pretenden resolver una problemática dentro del laboratorio de ingeniería de software se obtiene que la vista global del sistema es la siguiente.

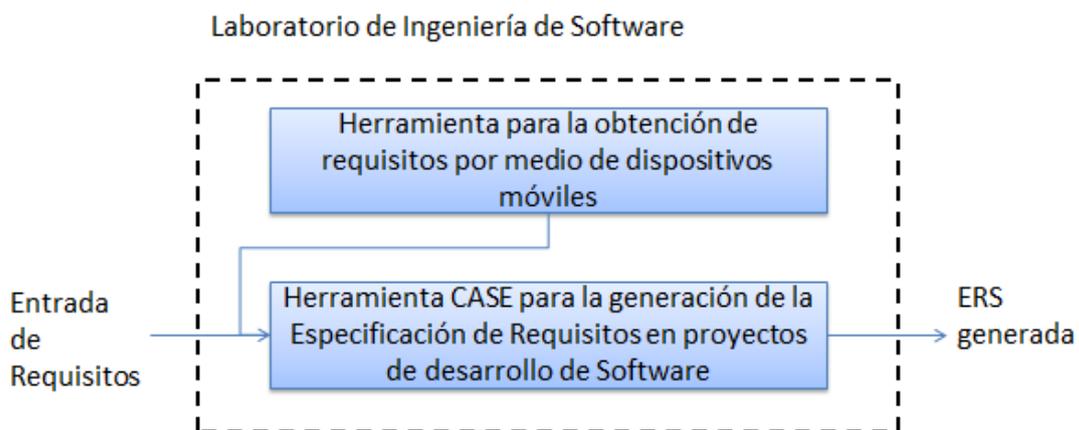


Imagen 10.1.1 Arquitectura Conceptual del Sistema

De manera más específica y enfocándonos a la Herramienta Case para la generación de la Especificación de Requisitos en proyectos de desarrollo de Software tenemos el siguiente diagrama.

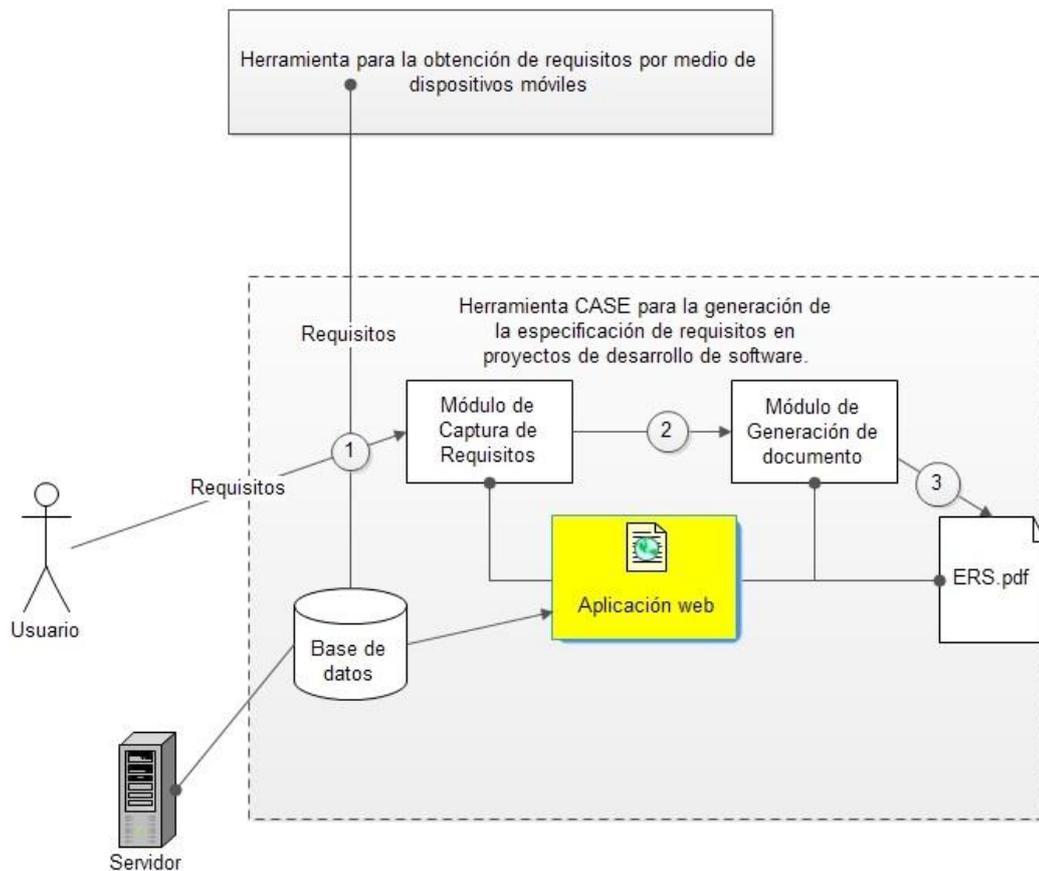


Imagen 10.1.1 b Arquitectura detallada del sistema

Del cual podemos definir lo siguiente:

1. Es un sistema que tiene como entrada requisitos y la salida es un documento de especificación de requisitos en un formato pdf
2. La aplicación cuenta con dos módulos principales uno de captura de requisitos y otro de generación de documento.
3. La aplicación está montada en un servidor mismo que contiene la base de datos del sistema.
4. El proceso de captura de requisitos es lineal. El usuario accede a la aplicación dónde captura requisitos en el módulo de captura (1), estos

son almacenados en la base de datos y posteriormente rescatados por el módulo de generación de documento (2) para generar el ERS. El resultado final es el documento de especificación de requisitos, ERS en formato pdf (3).

### **10.1.2 Funciones del sistema.**

Las funciones del sistema se refieren a los requisitos funcionales de la herramienta.

También en esta etapa se definieron las principales funciones del sistema, el equivalente a los requisitos funcionales. Las principales funciones quedaron definidas de la siguiente forma de acuerdo a los actores definidos posteriormente.

#### **Funciones de alumno**

- **Crear un documento nuevo:**  
Al crear un documento nuevo se crea automáticamente un equipo de trabajo para ese documento, cuyo único miembro por el momento, es el alumno que crea el documento.
- **Ver lista de documentos:**  
En esta lista se muestra el título de los documentos, así como su fecha de creación y las demás funciones de la que dispone el alumno que creó el documento
- **Ver observaciones:**  
Proporciona una lista con las observaciones que se han realizado acerca de un documento en específico seleccionado de la lista de documentos, esta lista muestra un título de la observación, la sección del documento a la que se refiere dicha observación, y la fecha en la que se realizó la observación. Proporciona la opción de leer a detalle la observación.
- **Leer observación:**  
Dentro de la lista de observaciones existe la opción de leer a detalle la descripción de la observación realizada por el maestro en dicho documento.
- **Ver equipo de trabajo:** despliega en pantalla la lista de miembros que integran el equipo de trabajo para ese documento, es decir, los miembros que pueden editar el documento.

- **Agregar miembro de equipo:** Despliega en pantalla una lista de los alumnos inscritos en el mismo grupo que el alumno que ha iniciado sesión, dándole la opción de seleccionar a través de un cuadro de selección (checkbox) a los alumnos que desea agregar al equipo de trabajo (esta acción la puede llevar a cabo cualquier alumno que sea miembro del equipo de trabajo)
- **Eliminar miembro de equipo:** Elimina a un miembro del equipo (esta acción solo puede llevarla a cabo el jefe de equipo, es decir, el alumno que crea el proyecto)
- **Generar documento pdf:** Esta función genera de manera automática en una nueva ventana un documento en formato pdf, tomando como datos de entrada la información capturada y guardada por cualquiera de los miembros del equipo de trabajo en cada uno de los formularios de las secciones del documento que proporciona la herramienta. La herramienta integra en el documento únicamente aquellas secciones que no estén vacías, si una sección está vacía, se descarta, y no se integra al documento, dándole la oportunidad a los usuarios de usar o no, cualquiera de las secciones de acuerdo al formato IEEE-830, pues dependiendo el proyecto que se está documentando, dependerá si se requiere el uso de cualquiera de estas secciones.
- **Editar documento:** muestra en pantalla un menú principal, tanto en el centro de la pantalla, como debajo del encabezado, con las secciones del documento de ERS, en los cuales el alumno puede acceder a los formularios para capturar, o editar la información capturada con anterioridad.
- **Borrar documento:** Elimina el documento seleccionado, así como todos sus registros e información almacenada (Esta función solo puede ser ejecutada el jefe de equipo, es decir, por el alumno creador del documento).

## Funciones del maestro

- **Crear grupo:** Con esta función el maestro crea un grupo de trabajo a los que los alumnos pertenecientes a este grupo podrán inscribirse para poder usar la herramienta, el maestro, debe proporcionar un nombre de grupo, el semestre, el periodo de estudio, y el año del periodo, para identificar al grupo, y que los alumnos puedan saber que grupo escoger

a la hora de registrarse, el maestro deberá registrar una contraseña para poder acceder a este grupo.

- **Editar grupo:** El maestro podrá realizar cambios en cualquiera de los campos mencionados en la función anterior.
- **Ver grupos:** Despliega en pantalla una lista con los grupos creados por el maestro, esta lista no muestra la contraseña pero incluye la cantidad total de alumnos inscritos al grupo.
- **Ver lista de alumnos:** Despliega una lista con el nombre completo, número de control de los alumnos inscritos e incluye un campo con la cantidad de documentos en las que el alumno participa como miembro de equipo de trabajo.
- **Ver equipos:** Muestra una lista con los proyectos existentes en el grupo de alumnos seleccionados, esta lista incluye el título del documento, así como el nombre de todos los miembros participantes, y las funciones de ver el documento para realizar observaciones.
- **Ver documento y realizar observaciones\*** :El maestro podrá ver en pantalla (por definirse si será en una pantalla nueva o en la misma pantalla en la que podrá realizar las observaciones) el documento generado en pdf con la información que los alumnos han capturado hasta el momento de ejecutar esta acción, para su lectura, y captura de observaciones sobre dicho documento.

A partir de estas funciones se definieron los casos de uso que quedarán plasmados en el diagrama de casos de uso.

### 10.1.3 Usuarios principales.

Para nuestro sistema se definieron dos tipos de usuario principales, maestro y alumno. En caso de ser aplicada para un contexto más general, es decir, para

la elaboración de documentos de especificación de requisitos en un carácter más empresarial estos dos actores podrían cambiar a “Analista de requisitos”

- Alumno:

Este usuario es un estudiante de ingeniería en sistemas computacionales con conocimientos técnicos en el manejo de múltiples programas y aplicaciones.

El alumno debe tener nociones y conocimientos de cómo se elabora un documento de ERS. Su función principal es capturar, modificar y elaborar el documento ERS.

- Profesor:

Este usuario es el titular de la materia de Ingeniería de Software, con experiencia en el campo de la Ingeniería de Software y el desarrollo de documentos de ERS. Su función principal es revisar, guiar y hacer observaciones del documento ERS elaborado por los alumnos

## **10.2 Diseño del sistema y software**

En esta etapa se diseñaron los diferentes componentes que formarían parte del sistema basándonos en el uso de herramientas modelos y diagramas para poder definirlos correctamente. Dentro de ellos podemos encontrar, la aplicación web, la base de datos y las interfaces de usuario.

### **10.2.1 Aplicación web**

Para el diseño de nuestra aplicación web, utilizamos herramientas de maquetación y de estilo como son HTML y CSS.

La navegación entre los distintos formularios quedó definida en el organigrama de bloque de la plataforma web en la sección 11.3 de este documento.

### **10.2.2 Base de datos**

Para diseñar la base de datos utilizamos herramientas de maquetación como MySQLWorkbench en dónde podíamos definir las tablas y las relaciones entre estas. Se diseñaron y definieron la cardinalidad de las relaciones, las tablas padre, las llaves primarias, las llaves foráneas y los atributos de cada una de las tablas.

Una vez definidas las tablas y las relaciones lo plasmamos en el diagrama Entidad-Relación el cual se puede apreciar en la sección 11.2 de este documento.

Cada una de ellas representa una parte del documento de especificación de requisitos y su contenido particular se explicará más adelante

### **10.2.3 Interfaces de Usuario.**

La interfaz general de la aplicación la realizamos entregando prototipos elaborados en GUI DESIGNER, una aplicación que simula la interacción del usuario con la aplicación mediante pantallas no funcionales que contienen elementos generales.

Los elementos que utilizamos fueron:

- Formularios
- Botones
- Listas
- Checkbox

## **10.3 Desarrollo.**

### **10.3.1 Base de datos.**

Una vez concluida la etapa de diseño procedimos a implementar las funciones, las clases y las tablas. Para poder implementar la base de datos utilizamos el sistema manejador de base de datos MySQL debido a que es multiplataforma y de licencia libre. Una de las características principales para elegir este DBMS

es que se puede integrar fácilmente con un servidor Web como Apache que también es de licencia libre.

Dentro de MySQL se creó una nueva base de datos a la cual se le agregaron tablas que servirían para guardar datos de los usuarios y de los documentos creados por estos.

### **10.3.2 Aplicación web.**

De acuerdo a los prototipos realizados en la fase anterior se buscó un entorno de desarrollo que permitiera implementarlas. Utilizamos Macromedia Dreamweaver para desarrollar los formularios de la herramienta. Dentro de los formularios teníamos que encontrar la forma de capturar imágenes y texto de una manera sencilla e intuitiva para el usuario, después de investigar encontramos la herramienta CKEditor la cual puede ser integrada en formularios web HTML /PHP y permite esta función. La configuración por defecto de CKEditores demasiado y tuvo que adaptarse y configurarse en conjunto con la herramienta KCUploader que permite la integración de imágenes dentro del editor.

Se crearon menús intuitivos para facilitar la navegación dentro de la herramienta. (pendiente con funciones)

Se integró la ayuda a cada sección y subsección del documento con la intención de facilitar la comprensión y el llenado de cada una de estas.

### **10.4 Pruebas**

Se procedió a hacer una prueba completa del sistema abarcando desde el registro de usuarios hasta la generación del documento en PDF. Para esta prueba no se abordó cada uno de los puntos del documento ERS a detalle, sino sólo un breve llenado de cada uno de los campos para probar la funcionalidad de la herramienta. Algunos de los campos por consecuencia están vacíos dentro de esta prueba.

Para esta primera ronda de pruebas, se utilizaron datos de entrada correspondientes al ERS del proyecto “Herramienta CASE para la obtención de la Especificación de Requisitos en proyectos de desarrollo de Software”

#### 10.4.1 Registro de usuarios.

La primera prueba funcional que se le hizo a la herramienta fue el registro de nuevos usuarios. Para esto tomamos los datos de entrada de uno de nosotros y los capturamos en la herramienta como se puede apreciar en la siguiente captura:

Generación de la Especificación de Requisitos en proyectos de desarrollo de Software

Inicio Iniciar Sesión Registrarse Contacto

REGISTRO DE USUARIO

BIENVENIDO

Por favor introduzca sus datos para registrarse

Nombre(s)\* Dorian

Ap. Paterno\* Garcia

Ap. Materno\* Castillo

No. de Control\* 09270818

Correo\* doriangc@hotmail.com

Contraseña\* \*\*\*\*\*

Verif. Contraseña\* \*\*\*\*\*

Maestro/Asesor\* Jose Suarez Vazquez

Correo Alternativo Ejemplo@email.com

Restablecer Siguiente

Captura 10.4.1 Registro de usuarios

##### 10.4.1.1 Registrar grupo

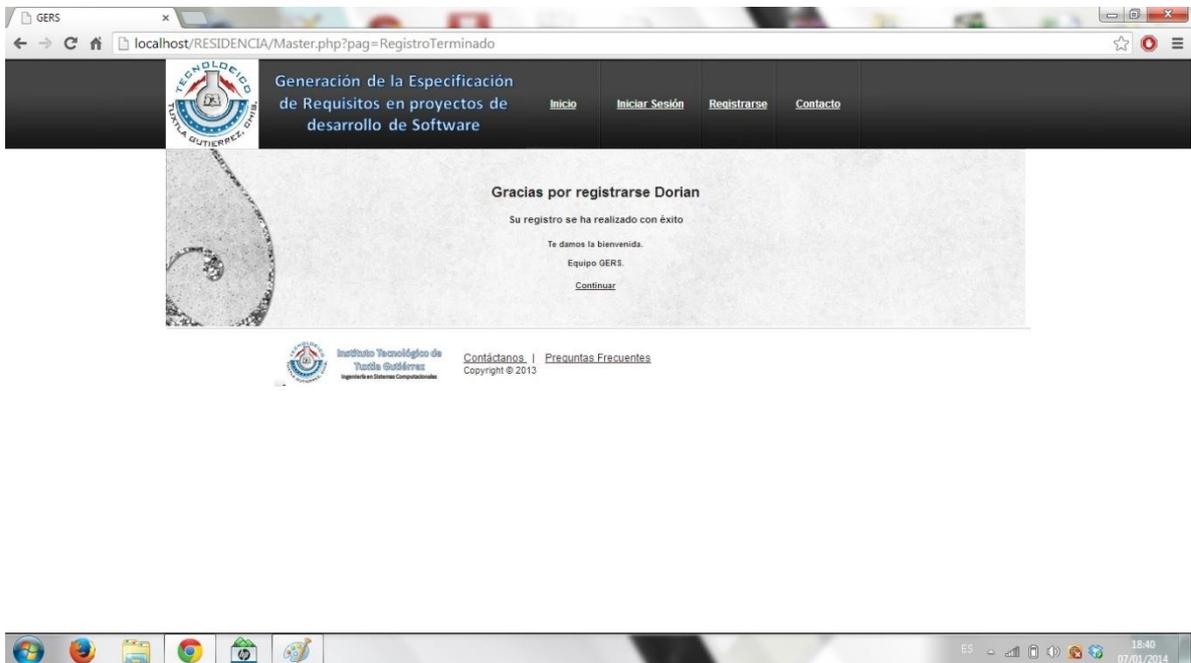
Después de llenar los campos de manera exitosa se procedió a dar de alta al alumno dentro de un grupo.



Captura 10.4.1.1 Registrar grupo

### 10.4.1.2 Mensaje de bienvenida

Una vez concluido el registro, la herramienta nos muestra un mensaje de bienvenida que valida que nuestro registro fue exitoso.



Captura 10.4.1.2 Mensaje de bienvenida

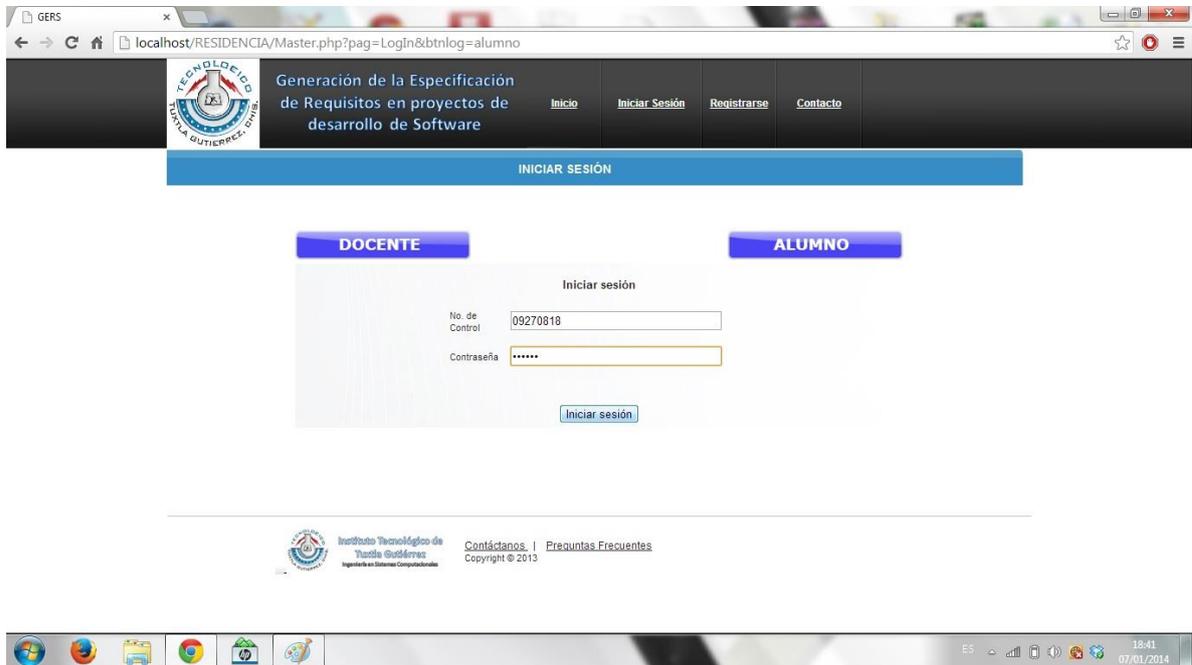
### **10.4.2 Login de Usuario**

El siguiente paso para llevar a cabo las pruebas, fue hacer login con la cuenta que acabamos de registrar.

Hay que considerar que el tipo de usuario considerado para esta prueba es el tipo "Alumno"

### 10.4.2.1 Login Alumno

Se ingresaron los datos de acceso dentro de la herramienta para ingresar con nuestra cuenta.

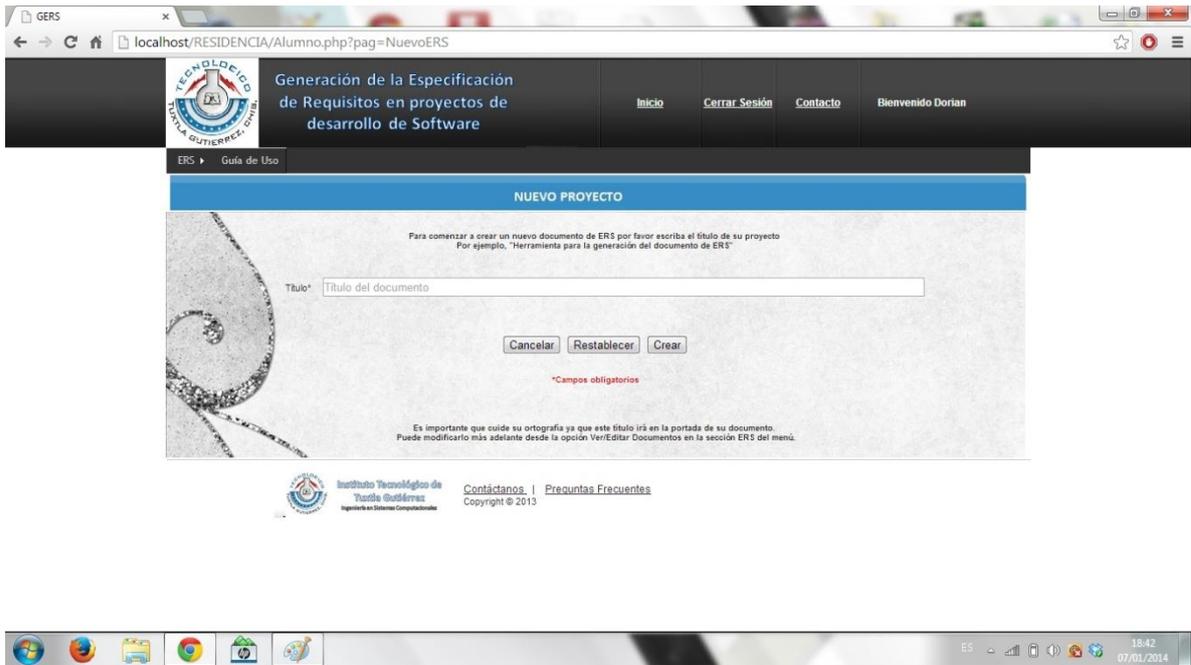


Captura 10.4.2.1 Login Alumno

Una vez que estuvimos dentro de la herramienta, se puede confirmar que el módulo para registrar nuevos usuarios y la base de datos que los contiene y actualiza funciona de manera correcta.

### 10.4.3 Crear nuevo documento (proyecto)

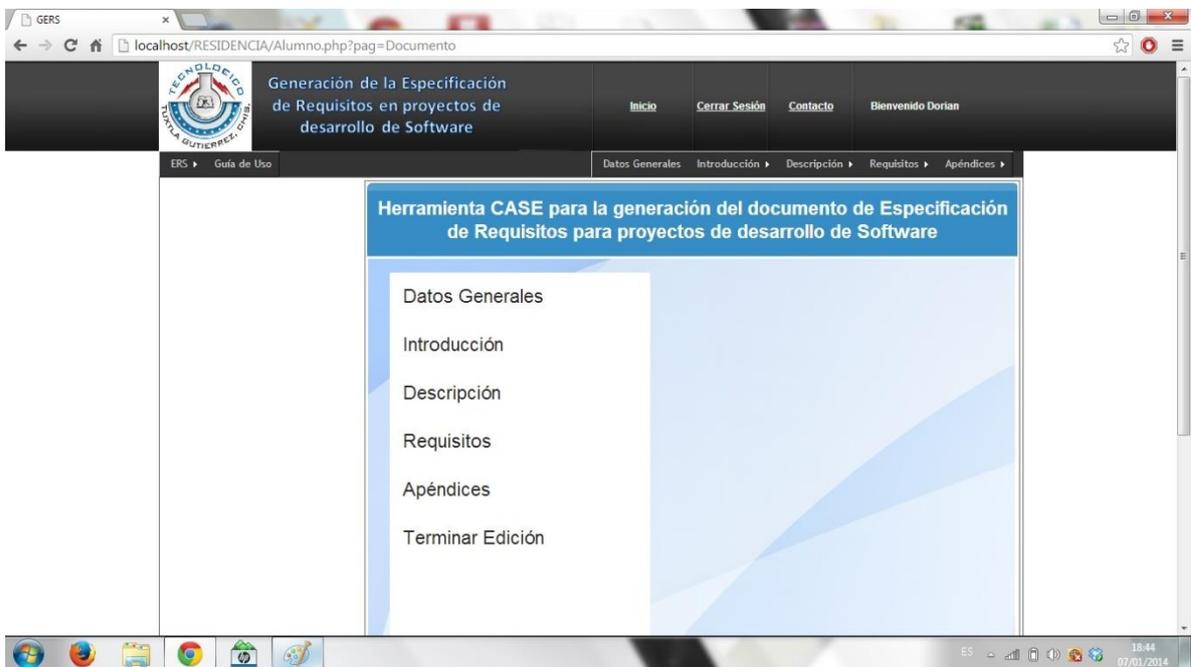
La siguiente etapa de pruebas que continúa después del Login del usuario, es probar el módulo para la creación de un nuevo documento. Para esto creamos un nuevo documento ingresando los datos que nos pedía la herramienta y validamos que el documento se hubiera creado con éxito.



Captura 10.4.3 Crear nuevo documento

### 10.4.3.1 Pantalla principal de nuevo documento

Una vez que le pusimos nombre al proyecto, la herramienta despliega la pantalla principal de nuevo documento.



Captura 10.4.3.1 Pantalla principal de nuevo documento 1

#### 10.4.4 Captura de documento

La siguiente etapa en la prueba piloto del sistema, es la captura de todas y cada una de las secciones del documento, para esto tomamos datos previamente recolectados para la creación de un ERS y fuimos llenando cada uno de los formularios. Casi todos los formularios ocupan la misma estructura, sin embargo existen algunos que contienen tablas especiales las cuales se mencionarán más adelante.

##### 10.4.4.1 Datos generales del proyecto

En esta imagen se aprecia el llenado del apartado “Datos generales del proyecto”



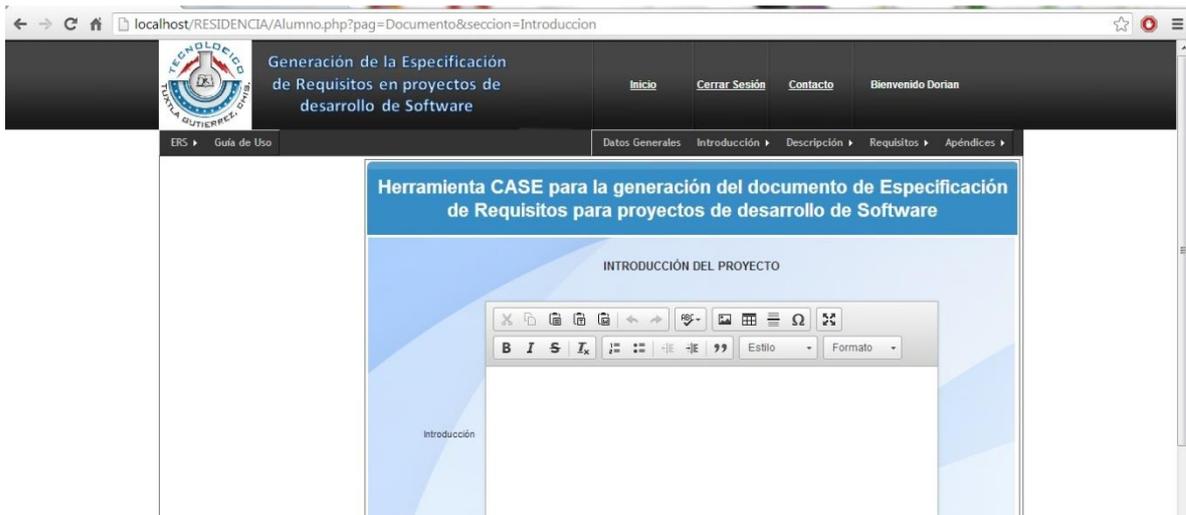
The screenshot shows a web browser window with the URL `localhost/RESIDENCIA/Alumno.php?pag=Documento&seccion=DatosGenerales`. The page header includes the logo of the Instituto Tecnológico de Tuxtla Gutiérrez and the title 'Generación de la Especificación de Requisitos en proyectos de desarrollo de Software'. The main content area is titled 'Herramienta CASE para la generación del documento de Especificación de Requisitos para proyectos de desarrollo de Software' and contains a form for 'DATOS GENERALES DEL PROYECTO'. The form includes the following fields:

- Titulo:** Herramienta CASE para la generación del documento de Especificación de Requisitos para
- Palabras Clave:** Herramienta CASE, software, ERS, IEEE-830, Requisitos, Especificación de Requisitos
- Sinopsis:** Este documento plasma cada uno de los requisitos acordados, tanto por el cliente como por el desarrollador, para el proyecto titulado "Herramienta CASE para la generación del documento de Especificación de Requisitos para proyectos de desarrollo de Software"
- Cliente:** Instituto Tecnológico de Tuxtla Gutiérrez

Captura 10.4.4.1 Datos generales del proyecto

##### 10.4.4.2 Introducción

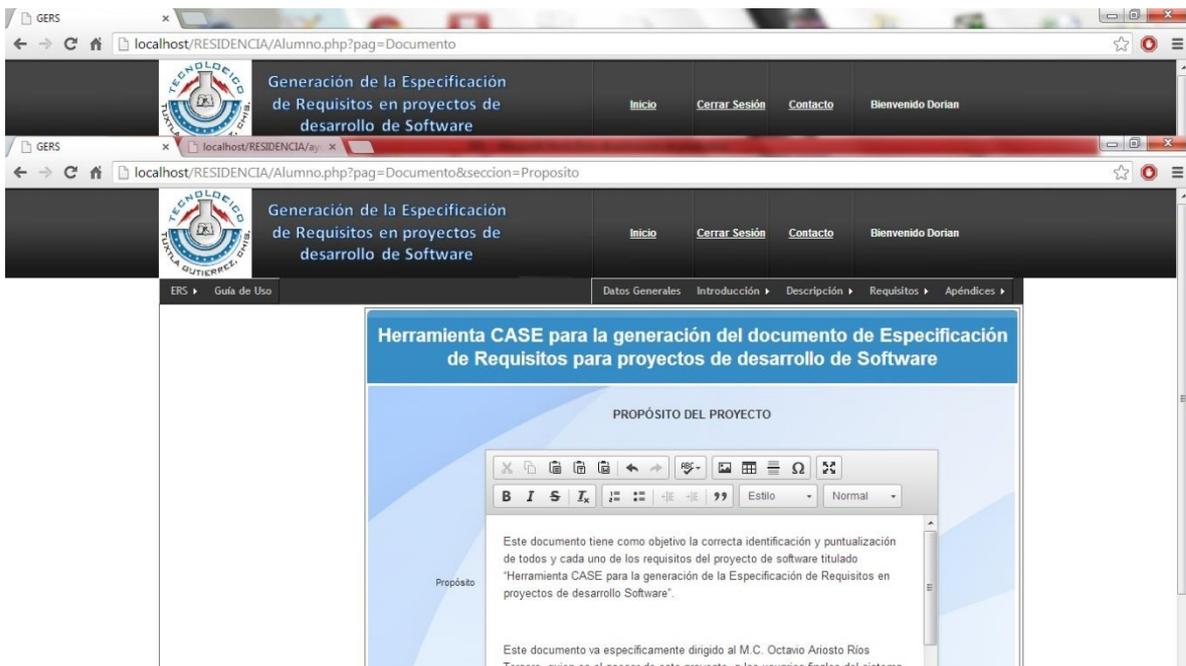
En esta imagen se aprecia el llenado del apartado “Introducción”



Captura 10.4.4.2 Introducción

### 10.4.4.3 Propósito

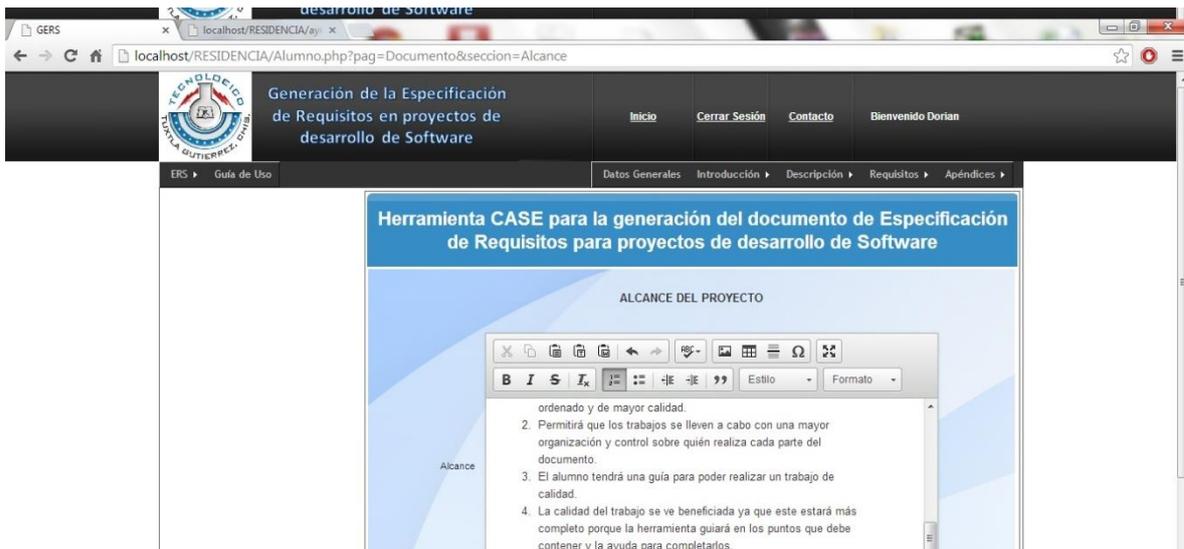
En esta imagen se aprecia el llenado del apartado “Propósito”



Captura 10.4.4.3 Propósito

### 10.4.4.4 Alcance

En esta imagen se aprecia el llenado del apartado “Alcance”



Captura 10.4.4.4 Alcance

#### 10.4.4.5 Personal involucrado

La sección del documento relacionada con el personal tiene una tabla especial que consta de dos secciones, la sección para agregar personal y la sección para ver la lista del personal involucrado. Así como esta tabla existen otras como la de usuarios, requisitos, definiciones y referencias que abordan el mismo tipo de llenado. Para no ser redundante en las imágenes, sólo se mostrará en esta sección las dos partes que la componen y el llenado de las mismas.

##### 10.4.4.5.1 Agregar personal

En esta imagen se aprecia el llenado de la subsección de “Agregar personal” de la sección “Personal”



Captura 10.4.4.5.1 Agregar personal

### 10.4.4.5.2 Lista de personal

En esta imagen se aprecia la subsección “ Lista de personal” dentro de la sección “Personal”



Captura 10.4.4.5.2 Lista de personal

### 10.4.4.6 Definiciones, Acrónimos y Abreviaturas

En esta imagen se aprecia el llenado del apartado “Definiciones, Acrónimos y Abreviaturas”



Captura 10.4.4.6 Definiciones, Acrónimos y abreviaturas

### 10.4.4.7 Referencias

En esta imagen se aprecia el llenado del apartado “Referencias”



Captura 10.4.4.7 Referencias

### 10.4.4.8 Resumen

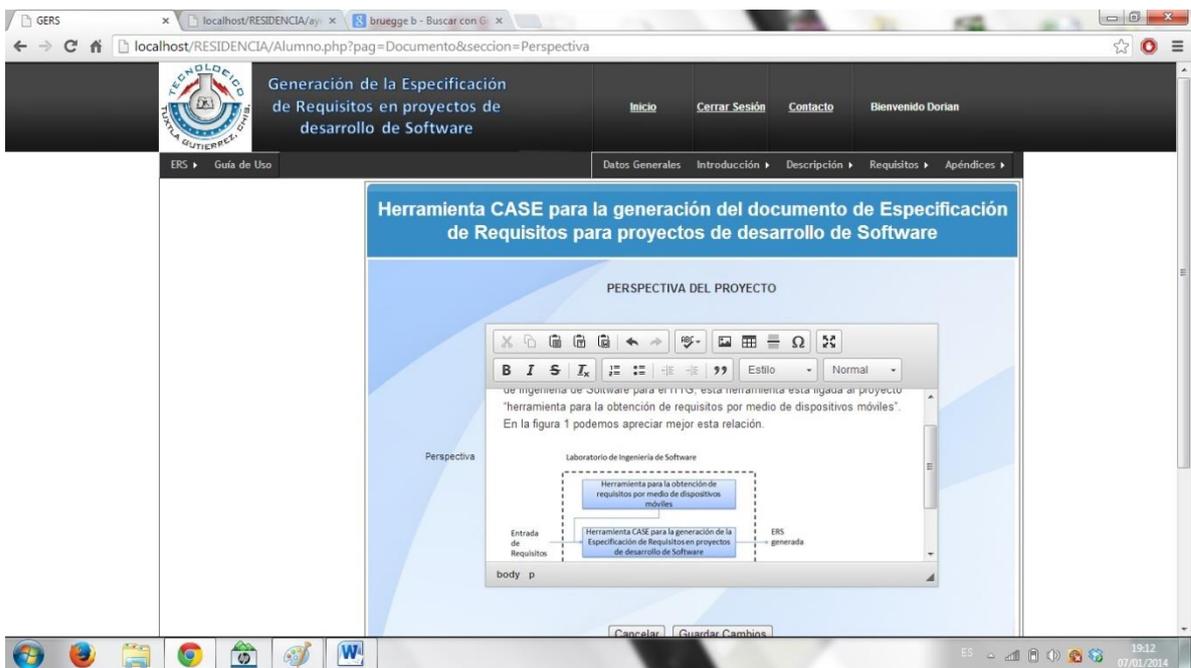
En esta imagen se aprecia el llenado del apartado “Resumen”



Captura 10.4.4.8 Resumen

### 10.4.4.9 Perspectiva del producto

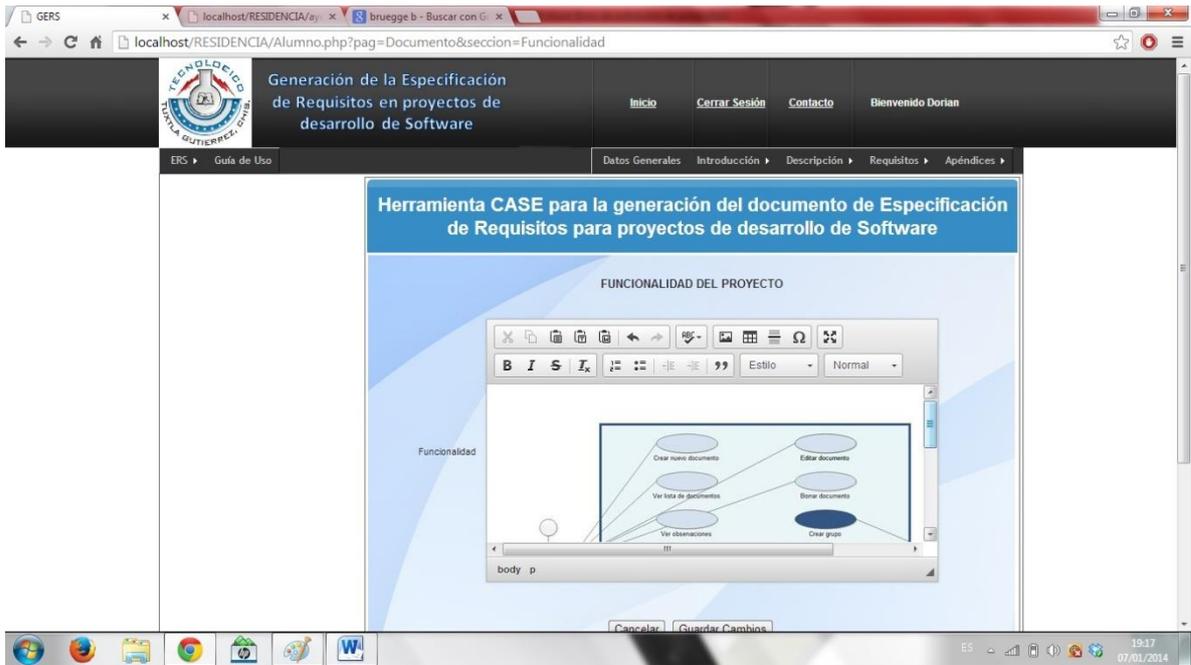
En esta imagen se aprecia el llenado del apartado “Perspectiva del producto”



Captura 10.4.4.9 Perspectiva del producto

### 10.4.4.10 Funcionalidad

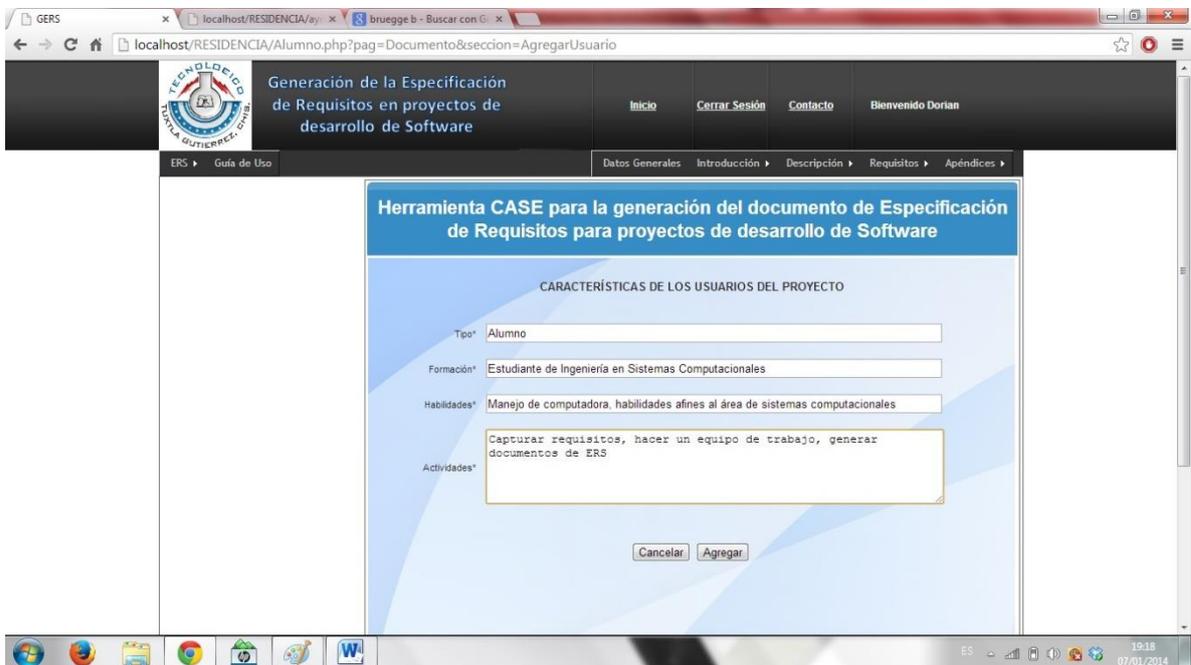
En esta imagen se aprecia el llenado del apartado “Funcionalidad”



Captura 10.4.4.10 Funcionalidad

### 10.4.4.11 Características de los usuarios

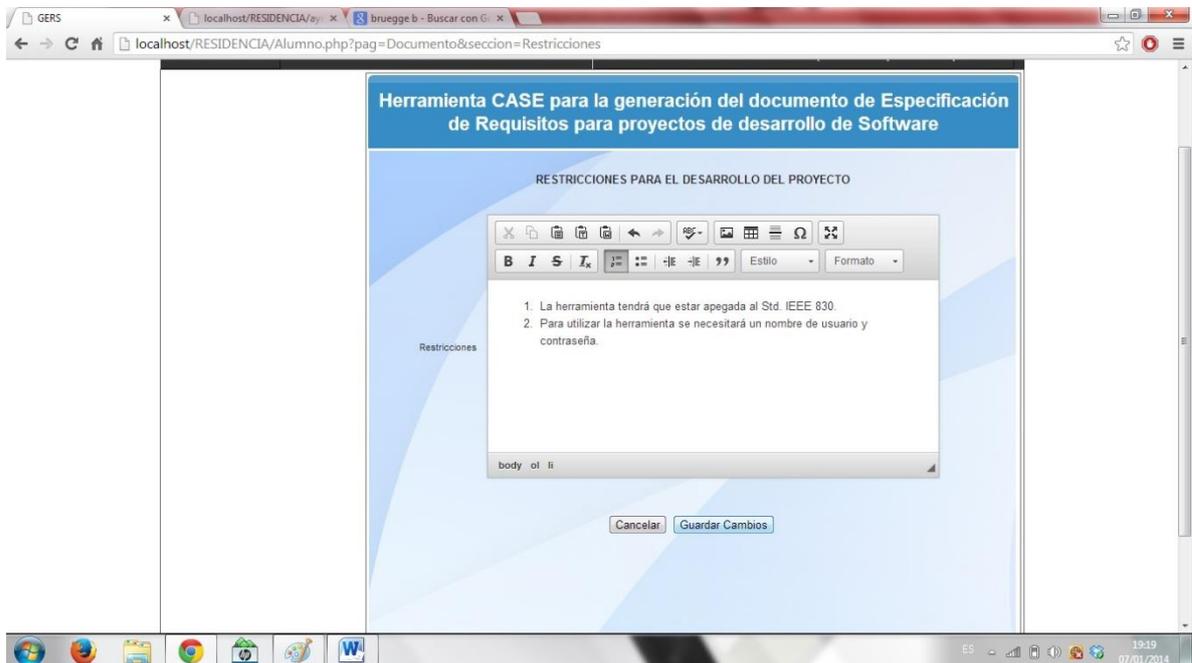
En esta imagen se aprecia el llenado del apartado “Características de los usuarios”



Captura 10.4.4.11 Características de los usuarios

#### 10.4.4.12 Restricciones

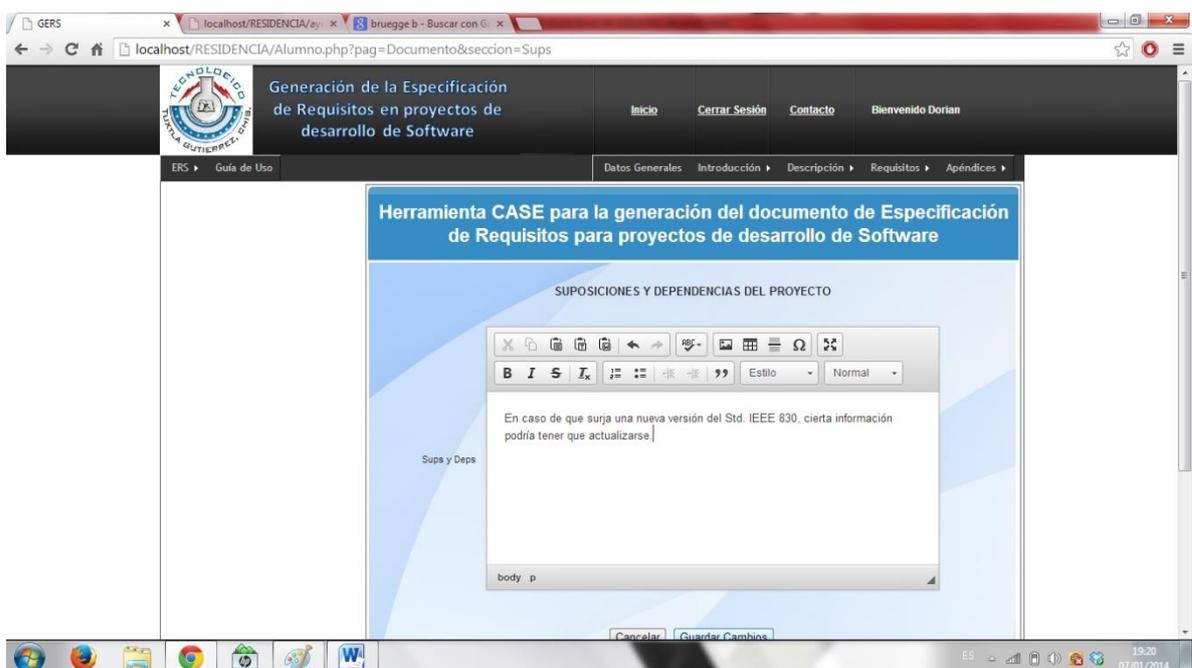
En esta imagen se aprecia el llenado del apartado “Restricciones”



Captura 10.4.4.12 Restricciones

#### 10.4.4.13 Suposiciones y dependencias

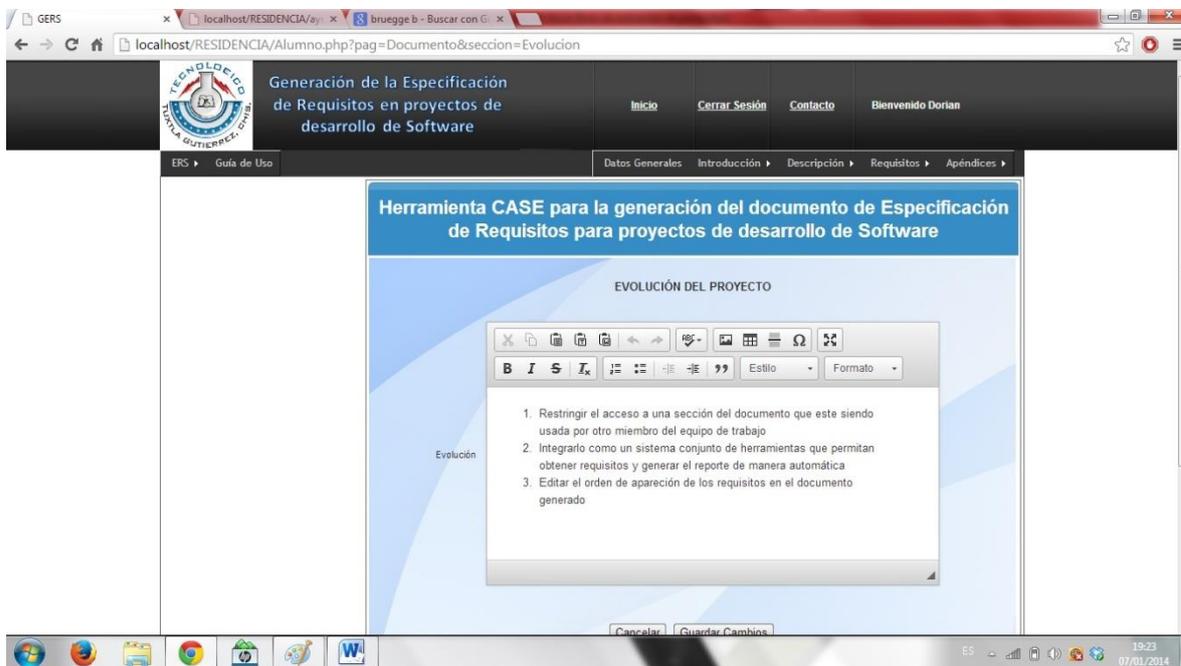
En esta imagen se aprecia el llenado del apartado “Suposiciones y dependencias”



Captura 10.4.4.13 Suposiciones y dependencias

#### 10.4.4.14 Evolución del proyecto

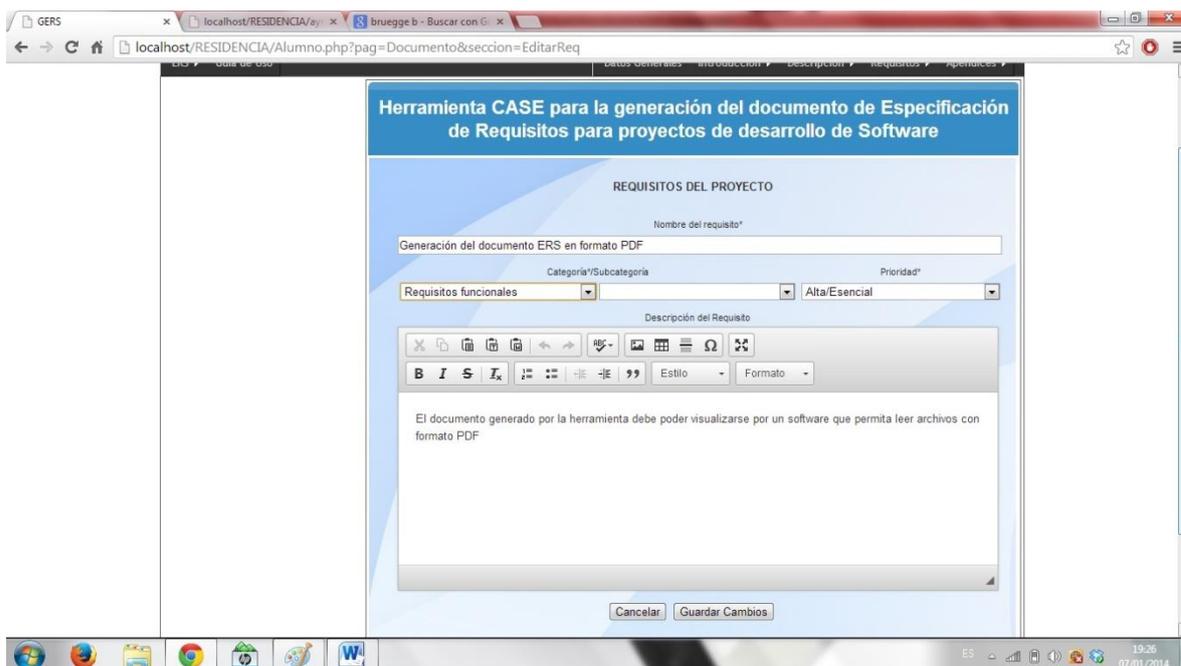
En esta imagen se aprecia el llenado del apartado “Evolución del proyecto”



Captura 10.4.4.14 Evolución del proyecto

#### 10.4.4.15 Requisitos del proyecto

En esta imagen se aprecia el llenado del apartado “Requisitos”

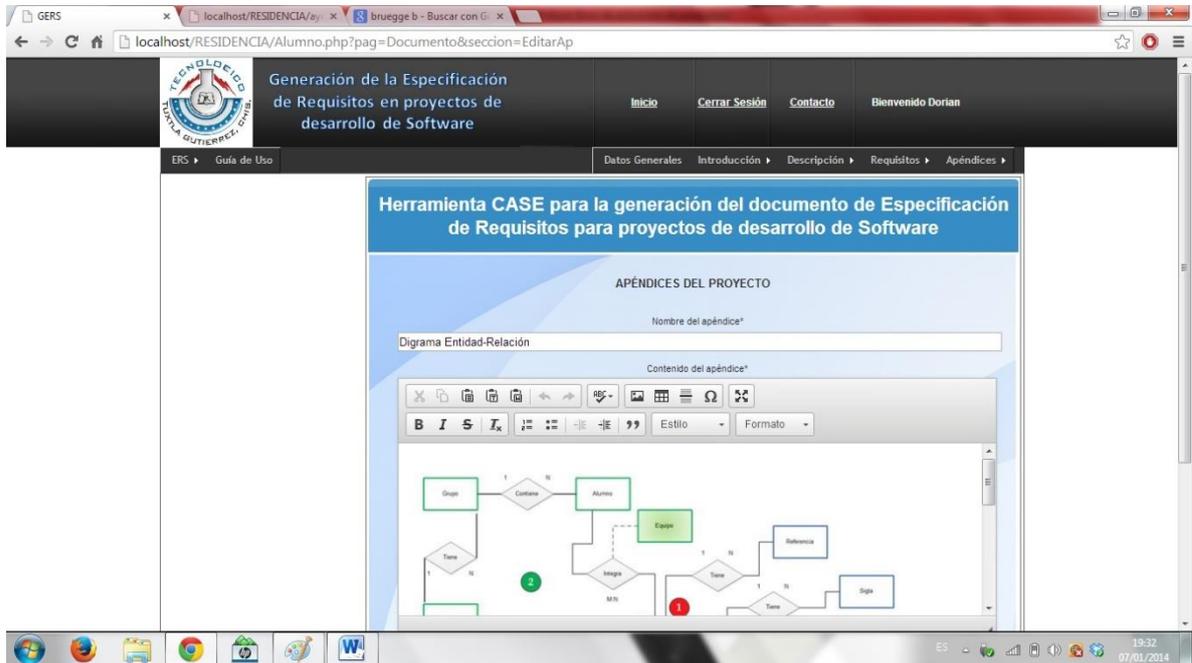


Captura 10.4.4.15 Requisitos del proyecto



#### 10.4.4.16 Apéndices

En esta imagen se aprecia el llenado del apartado “Requisitos”

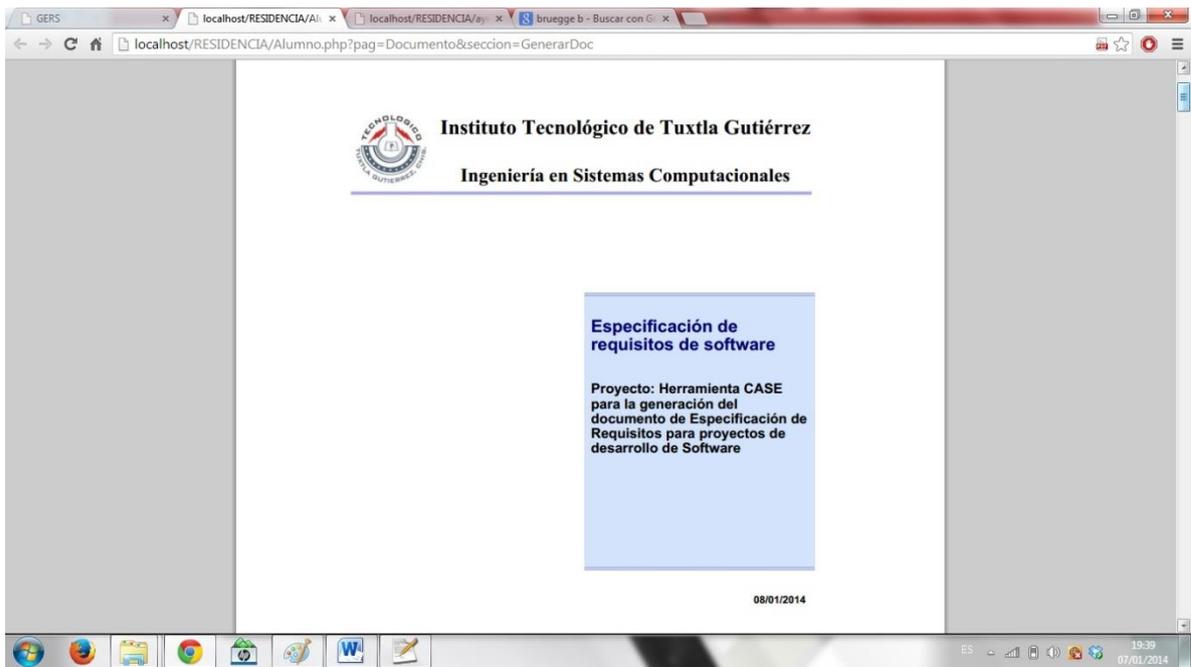


Captura 10.4.4.16 Apéndices

#### 10.4.5. Generación de documento PDF

El último paso para concluir esta prueba piloto de la funcionalidad de la herramienta era generar el documento de especificación de requisitos en formato PDF y ver y analizar los resultados.

Para facilitar la lectura la siguiente imagen muestra la portada del documento generado. Los demás puntos y resultados de la prueba están contenidos en la sección 11 de este documento.



Captura 10.4.5 Documento generado

## 11. Resultados, planos, gráficos, prototipos y programas

### 11.1 Diagrama de casos de uso

El diagrama de casos de uso, es una herramienta útil para determinar cómo interactúan los usuarios con la herramienta. Se pueden apreciar las funciones principales que llevan a cabo cada uno de ellos.

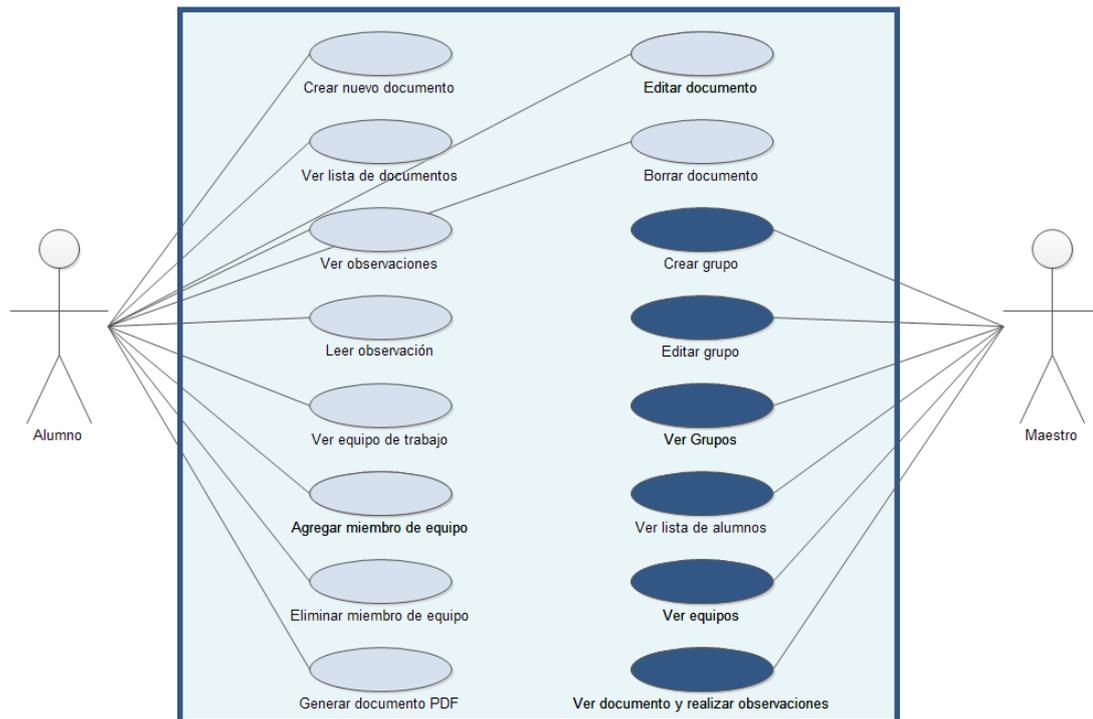


Figura 11.1 Diagrama de Casos de Uso

## 11.2 Plantillas de caso de uso

### 11.2.1 Casos de uso Alumno

|                                |   |   |
|--------------------------------|---|---|
| <b>NOMBRE DEL CASO DE USO:</b> | Crear nuevo documento   |   |
| <b>ACTOR:</b>                  | Alumno  |   |
| <b>PROPÓSITO:</b>              | El alumno crea un nuevo documento ERS para trabajar en su captura     |   |
| <b>CÓMO SE ACTIVA:</b>         | El Alumno activa la función dentro del sistema                        |   |
| <b>FLUJO DE EVENTOS:</b>       |   |   |
| <b>No</b>                      | <b>ACTOR</b>  | <b>SISTEMA</b>  |
| 1                              |   | Pide al Alumno que asigne un nombre para el nuevo documento a crearse         |
| 2                              | Introduce el nombre del documento.                                    |   |
| 3                              |   | Crea automáticamente un equipo de trabajo correspondiente al nuevo documento. |
| Condición de salida:           | Cuando el Alumno visualiza en pantalla el nombre del nuevo documento. |   |
| Requerimientos especiales:     | El nombre del documento no puede estar vacío.                         |   |

|                                |  |   |
|--------------------------------|--|---|
| <b>NOMBRE DEL CASO DE USO:</b> | Ver lista de documentos  |   |
| <b>ACTOR:</b>                  | Alumno   |   |
| <b>PROPÓSITO:</b>              | El Alumno pueda visualizar una lista con todos los documentos en los que está trabajando                       |   |
| <b>CÓMO SE ACTIVA:</b>         | El Alumno activa la función dentro del sistema   |   |
| <b>FLUJO DE EVENTOS:</b>       |  |   |
| <b>No</b>                      | <b>ACTOR</b>   | <b>SISTEMA</b>  |
| 1                              |  | Muestra una lista con los documentos que correspondientes a ese Alumno. En esta se aprecian el título del documento y la fecha en la que fue creado |
| 2                              | Elige un documento de la lista   |   |
| Condición de salida:           | Cuando el alumno accede a alguno de los documentos para modificarlo  |   |
| Requerimientos especiales:     | El alumno tiene que estar asignado a algún equipo de trabajo, de lo contrario no habrá documentos en la lista. |   |

|                                |   |   |
|--------------------------------|---|---|
| <b>NOMBRE DEL CASO DE USO:</b> | Ver observaciones.  |   |
| <b>ACTOR:</b>                  | Alumno.   |   |
| <b>PROPÓSITO:</b>              | El Alumno pueda visualizar la lista de los documentos que recibieron observaciones.     |   |
| <b>CÓMO SE ACTIVA:</b>         | El Alumno activa la función dentro del sistema.   |   |
| <b>FLUJO DE EVENTOS:</b>       |   |   |
| <b>No</b>                      | <b>ACTOR</b>  | <b>SISTEMA</b>  |
| 1                              |   | Proporciona una lista con las observaciones que se han realizado acerca de un documento en específico seleccionado de la lista de documentos. |
| 2                              | Visualiza la lista de observaciones en pantalla.  |   |
| Condición de salida:           | Cuando el Alumno visualiza en su pantalla la lista de observaciones de manera adecuada. |   |
| Requerimientos especiales:     | El alumno tuvo que haber recibido observaciones o la lista aparecerá vacía.             |   |

|                                |  |   |
|--------------------------------|--|---|
| <b>NOMBRE DEL CASO DE USO:</b> | Leer observación.  |   |
| <b>ACTOR:</b>                  | Alumno.  |   |
| <b>PROPÓSITO:</b>              | El Alumno pueda visualiza una de las observaciones a detalle.                  |   |
| <b>CÓMO SE ACTIVA:</b>         | El Alumno elige la opción dentro del apartado "observaciones"                  |   |
| <b>FLUJO DE EVENTOS:</b>       |  |   |
| <b>No</b>                      | <b>ACTOR</b>   | <b>SISTEMA</b>  |
| 1                              |  | Muestra a detalle la descripción de la observación realizada por el maestro en dicho documento. |
| 2                              | Lee el nombre y la descripción de la observación elegida.                      |   |
| Condición de salida:           | Cuando el Alumno visualiza en pantalla el contenido de la observación deseada. |   |
| Requerimientos especiales:     | Ninguna.   |   |

|                                |  |  |
|--------------------------------|--|--|
| <b>NOMBRE DEL CASO DE USO:</b> | Ver equipo de trabajo.   |  |
| <b>ACTOR:</b>                  | Alumno.  |  |
| <b>PROPÓSITO:</b>              | El Alumno pueda visualizar los miembros de su equipo.  |  |
| <b>CÓMO SE ACTIVA:</b>         | El Alumno activa la función dentro del sistema.  |  |
| <b>FLUJO DE EVENTOS:</b>       |  |  |
| <b>No</b>                      | <b>ACTOR</b>   | <b>SISTEMA</b>   |
| 1                              |  | Muestra una lista de miembros que integran el equipo de trabajo para ese documento, es decir, los miembros que pueden editar el documento. |
| 2                              | Consulta quiénes pueden modificar el documento.  |  |
| Condición de salida:           | Cuando el alumno visualiza en pantalla la lista de integrantes que están en su equipo  |  |
| Requerimientos especiales:     | Para mostrar más de un miembro en el actual equipo de trabajo el alumno debe agregarlos mediante la función “agregar miembro de equipo”, en caso contrario sólo se mostrará el alumno que creó el documento. |  |

|                                |  |   |
|--------------------------------|--|---|
| <b>NOMBRE DEL CASO DE USO:</b> | Agregar miembro de equipo.   |   |
| <b>ACTOR:</b>                  | Alumno.  |   |
| <b>PROPÓSITO:</b>              | El Alumno pueda agregar miembros al equipo de trabajo.   |   |
| <b>CÓMO SE ACTIVA:</b>         | El Alumno activa la función dentro del sistema.  |   |
| <b>FLUJO DE EVENTOS:</b>       |  |   |
| <b>No</b>                      | <b>ACTOR</b>   | <b>SISTEMA</b>  |
| 1                              |  | Muestra una lista de miembros que integran el equipo de trabajo para ese documento. |
| 2                              | Activa la opción “agregar miembro de equipo”.  |   |
| 3                              |  | Despliega un <i>checkbox</i> con los alumnos que puede agregar al equipo.           |
| 4                              | El alumno elige que alumnos se agregarán al equipo.  |   |
| Condición de salida:           | Cuando el alumno termina de agregar miembros al equipo de manera exitosa.                                |   |
| Requerimientos especiales:     | Sólo los alumnos del mismo grupo se pueden agregar al equipo y sólo el jefe del equipo puede agregarlos. |   |

|                                |                             |  |
|--------------------------------|-----------------------------|--|
| <b>NOMBRE DEL CASO DE USO:</b> | Eliminar miembro de equipo. |  |
| <b>ACTOR:</b>                  | Alumno.                     |  |

|                            |  |   |
|----------------------------|--|---|
| <b>PROPÓSITO:</b>          | El Alumno pueda eliminar miembros de su equipo de trabajo.       |   |
| <b>CÓMO SE ACTIVA:</b>     | El Alumno activa la función dentro del sistema.                  |   |
| <b>FLUJO DE EVENTOS:</b>   |  |   |
| <b>No</b>                  | <b>ACTOR</b>   | <b>SISTEMA</b>  |
| 1                          |  | Muestra una lista de miembros que integran el equipo de trabajo para ese documento. |
| 2                          | Activa la opción “eliminar miembros”                             |   |
| 3                          |  | Despliega un <i>checkbox</i> con los alumnos que puede eliminar del equipo.         |
| 4                          | El alumno elige que alumnos serán eliminados.                    |   |
| Condición de salida:       | Cuando el alumno elimina exitosamente los alumnos seleccionados. |   |
| Requerimientos especiales: | Sólo el jefe del equipo puede eliminar miembros.                 |   |

|                                |   |  |
|--------------------------------|---|--|
| <b>NOMBRE DEL CASO DE USO:</b> | Generar documento PDF   |  |
| <b>ACTOR:</b>                  | Alumno.   |  |
| <b>PROPÓSITO:</b>              | El Alumno pueda generar el documento ERS en formato PDF de manera automática                          |  |
| <b>CÓMO SE ACTIVA:</b>         | El Alumno activa la función dentro del sistema.   |  |
| <b>FLUJO DE EVENTOS:</b>       |   |  |
| <b>No</b>                      | <b>ACTOR</b>  | <b>SISTEMA</b>   |
| 1                              | Activa la opción de guardar los cambios dentro del documento  |  |
| 2                              |   | Guarda los cambios hechos por el alumno  |
| 3                              | Elige la opción “generar pdf”   |  |
| 4                              |   | Despliega en una nueva ventana el documento en formato PDF con los cambios realizados hasta el momento |
| Condición de salida:           | Cuando el alumno visualiza el documento PDF de manera exitosa en la pantalla                          |  |
| Requerimientos especiales:     | Sólo se muestran las secciones que han sido capturadas por el usuario, las que están vacías se omiten |  |
| <b>NOMBRE DEL CASO DE USO:</b> | Editar documento  |  |
| <b>ACTOR:</b>                  | Alumno.   |  |
| <b>PROPÓSITO:</b>              | El Alumno pueda capturar y editar su documento ERS  |  |
| <b>CÓMO SE ACTIVA:</b>         | El Alumno crea un nuevo documento para trabajar   |  |
| <b>FLUJO DE EVENTOS:</b>       |   |  |

| No                         | ACTOR  | SISTEMA   |
|----------------------------|--|---|
| 1                          | Crea un nuevo documento  |   |
| 2                          |  | Despliega la pantalla de nuevo documento con las secciones que lo conforman |
| 3                          | Captura las secciones del documento y edita los cambios que considere adecuados. |   |
| 4                          | Guarda los cambios realizados  |   |
| Condición de salida:       |  | Cuando el sistema guarda los cambios realizados a petición del alumno       |
| Requerimientos especiales: |  | Ninguno   |

| <b>NOMBRE DEL CASO DE USO:</b> | Eliminar documento.  |  |
|--------------------------------|--|--|
| <b>ACTOR:</b>                  | Alumno.  |  |
| <b>PROPÓSITO:</b>              | El Alumno pueda eliminar el documento de la herramienta y su contenido de manera permanente. |  |
| <b>CÓMO SE ACTIVA:</b>         | El Alumno activa la función dentro del sistema.  |  |
| <b>FLUJO DE EVENTOS:</b>       |  |  |
| No                             | ACTOR  | SISTEMA  |
| 1                              |  | Muestra una lista de documentos en los que el alumno tiene participación.      |
| 2                              | Activa la opción "eliminar documentos"   |  |
| 3                              |  | Despliega un <i>checkbox</i> con los documentos que puede eliminar del equipo. |
| 4                              | El alumno elige los documentos serán eliminados.   |  |
| Condición de salida:           |  | Cuando el alumno elimina exitosamente los documentos seleccionados             |
| Requerimientos especiales:     |  | Sólo el jefe del equipo puede eliminar documentos.                             |

| <b>NOMBRE DEL CASO DE USO:</b> | Eliminar documento.  |   |
|--------------------------------|--|---|
| <b>ACTOR:</b>                  | Alumno.  |   |
| <b>PROPÓSITO:</b>              | El Alumno pueda eliminar el documento de la herramienta y su contenido de manera permanente. |   |
| <b>CÓMO SE ACTIVA:</b>         | El Alumno activa la función dentro del sistema.  |   |
| <b>FLUJO DE EVENTOS:</b>       |  |   |
| No                             | ACTOR  | SISTEMA   |
| 1                              |  | Muestra una lista de documentos en los que el alumno tiene participación. |

|                            |  |  |
|----------------------------|--|--|
| 2                          | Activa la opción “eliminar documentos”                             |  |
| 3                          |  | Despliega un <i>checkbox</i> con los documentos que puede eliminar del equipo. |
| 4                          | El alumno elige los documentos serán eliminados.                   |  |
| Condición de salida:       | Cuando el alumno elimina exitosamente los documentos seleccionados |  |
| Requerimientos especiales: | Sólo el jefe del equipo puede eliminar documentos.                 |  |

### 11.2.2 Casos de uso Maestro

|                                |  |  |
|--------------------------------|--|--|
| <b>NOMBRE DEL CASO DE USO:</b> | Crear grupo  |  |
| <b>ACTOR:</b>                  | Maestro  |  |
| <b>PROPÓSITO:</b>              | El maestro pueda crear un grupo de trabajo a los que los alumnos puedan inscribirse para poder usar la herramienta.                |  |
| <b>CÓMO SE ACTIVA:</b>         | El Maestro activa la función dentro del sistema.   |  |
| <b>FLUJO DE EVENTOS:</b>       |  |  |
| <b>No</b>                      | <b>ACTOR</b>   | <b>SISTEMA</b>   |
| 1                              |  | Pide al maestro una serie de datos para poder crear el grupo |
| 2                              | Proporciona el nombre de grupo, el semestre, el periodo de estudio, el año del periodo y una contraseña para identificar al grupo. |  |
| 3                              |  | Agrega el nuevo grupo a la base de datos                     |
| Condición de salida:           | Cuando el grupo se agrega con éxito.   |  |
| Requerimientos especiales:     | Ninguna  |  |
| <b>NOMBRE DEL CASO DE USO:</b> | Editar grupo   |  |
| <b>ACTOR:</b>                  | Maestro  |  |
| <b>PROPÓSITO:</b>              | El pueda cambiar los datos de los grupos creados   |  |
| <b>CÓMO SE ACTIVA:</b>         | El Maestro activa la función dentro del sistema.   |  |
| <b>FLUJO DE EVENTOS:</b>       |  |  |
| <b>No</b>                      | <b>ACTOR</b>   | <b>SISTEMA</b>   |
| 1                              |  | Despliega una lista con todos los grupos creados             |
| 2                              | Identifica el grupo que desea modificar  |  |
| 3                              |  | Despliega los datos del grupo seleccionado                   |

|                            |  |  |
|----------------------------|--|--|
| 4                          | Modifica los datos del grupo que considere pertinentes           |  |
| Condición de salida:       | Cuando los cambios en el grupo se actualizan en la base de datos |  |
| Requerimientos especiales: | Ninguna  |  |

|                            |   |  |
|----------------------------|---|--|
| NOMBRE DEL CASO DE USO:    | Ver grupos  |  |
| ACTOR:                     | Maestro   |  |
| PROPÓSITO:                 | El Maestro pueda visualizar la lista completa de los grupos creados |  |
| CÓMO SE ACTIVA:            | El Maestro activa la función dentro del sistema.                    |  |
| <b>FLUJO DE EVENTOS:</b>   |   |  |
| <b>No</b>                  | <b>ACTOR</b>  | <b>SISTEMA</b>   |
| 1                          | Accede al apartado grupos.  |  |
| 2                          |   | Da la opción de visualizar la lista de grupos.   |
| 3                          | Activa la función "ver grupos"                                      |  |
| 4                          |   | Despliega una lista con todos los grupos mostrando la cantidad de alumnos en cada uno, pero sin mostrar la contraseña. |
| Condición de salida:       | Cuando el maestro visualiza la lista de grupos en la pantalla       |  |
| Requerimientos especiales: | Ninguna   |  |

|                          |   |   |
|--------------------------|---|---|
| NOMBRE DEL CASO DE USO:  | Ver equipos   |   |
| ACTOR:                   | Maestro   |   |
| PROPÓSITO:               | El Maestro pueda visualizar la lista completa de los grupos creados |   |
| CÓMO SE ACTIVA:          | El Maestro activa la función dentro del sistema.                    |   |
| <b>FLUJO DE EVENTOS:</b> |   |   |
| <b>No</b>                | <b>ACTOR</b>  | <b>SISTEMA</b>  |
| 1                        | Accede al apartado grupos   |   |
| 2                        |   | Pide que elija el grupo al cual acceder   |
| 3                        | Elige el grupo y activa la función "ver equipos"                    |   |
| 4                        |   | Muestra una lista con los proyectos existentes en el grupo de alumnos seleccionados, esta lista incluye el título del documento, así como el nombre de todos los miembros |

|                            |  |
|----------------------------|--|
|                            | participantes.   |
| Condición de salida:       | Cuando el maestro visualiza la lista de proyectos en la pantalla |
| Requerimientos especiales: | Ninguna  |

|                            |   |  |
|----------------------------|---|--|
| NOMBRE DEL CASO DE USO:    | Ver documento y realizar observaciones  |  |
| ACTOR:                     | Maestro   |  |
| PROPÓSITO:                 | El Maestro pueda visualizar el documento hecho por los alumnos y realizar observaciones al respecto         |  |
| CÓMO SE ACTIVA:            | El Maestro activa la función dentro del sistema.  |  |
| <b>FLUJO DE EVENTOS:</b>   |   |  |
| <b>No</b>                  | <b>ACTOR</b>  | <b>SISTEMA</b>   |
| 1                          |   | Muestra en pantalla del pdf del documento consultado   |
| 2                          | Accede a las secciones del documento que desea revisar  |  |
| 3                          |   | Muestra una sección en la cual puede realizar observaciones especificando el asunto, la sección del documento y el contenido de la observación |
| 4                          | Inserta los datos de la observación y continúa agregando observaciones                                      |  |
| 5                          |   | Da la opción de editar las observaciones realizadas  |
| 6                          | Edita las observaciones en caso de requerirlo y guarda los cambios realizados                               |  |
| Condición de salida:       | Cuando el maestro está conforme con las observaciones realizadas y el sistema guarda los cambios realizados |  |
| Requerimientos especiales: | Ninguna   |  |

## 11.2 Diagrama Entidad-Relación

El diseño Entidad-Relación nos sirve para identificar las diferentes tablas contenidas dentro de la base de datos y como se relacionan entre sí. Sirve también para identificar los atributos de cada una de las tablas y cuáles son tanto sus llaves identificadoras como las llaves que heredan de otras tablas.

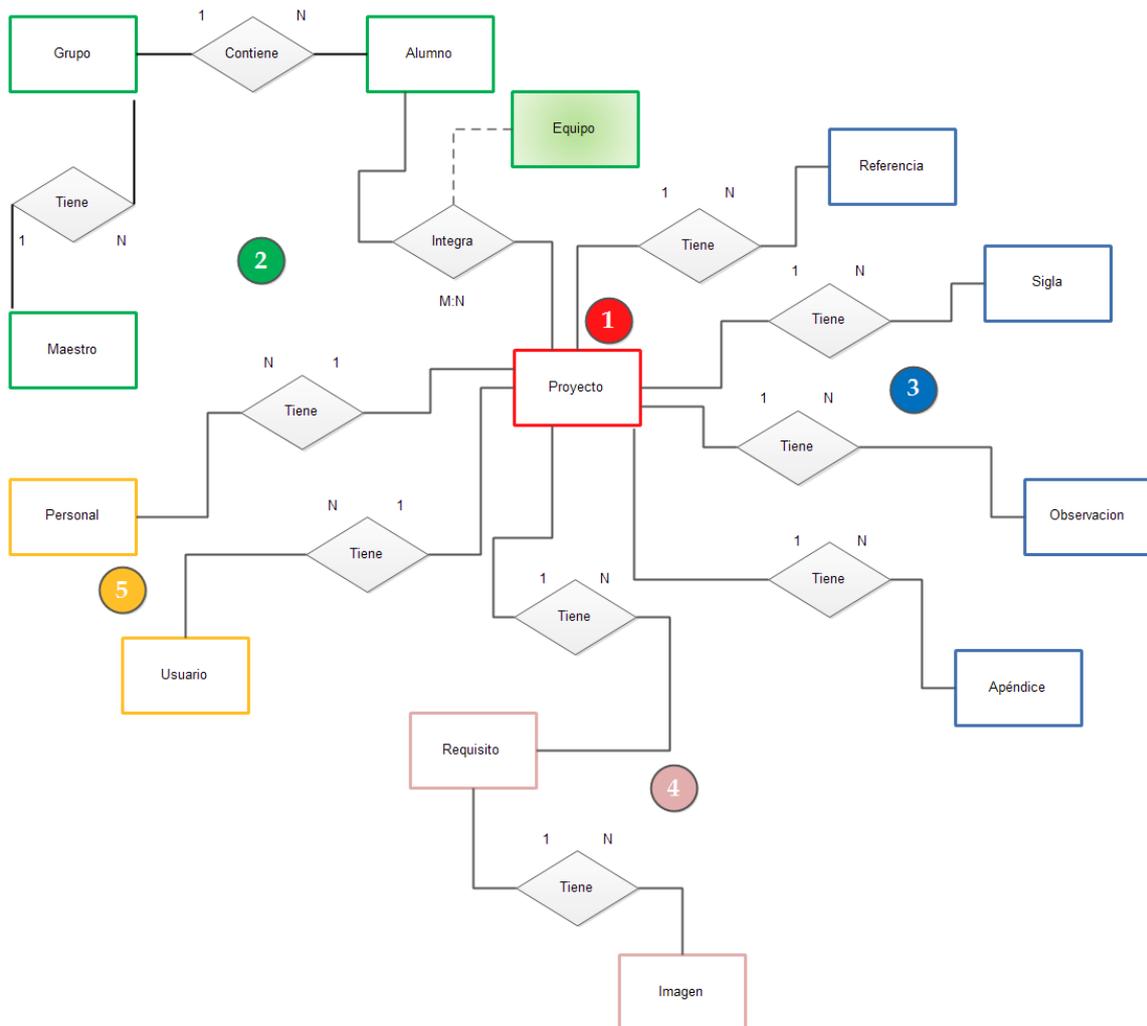


Ilustración 11.2 Diagrama ER de la base de datos

En los siguientes diagramas, se pueden apreciar a detalle cada uno de los cortes, la división es de acuerdo a su función dentro del documento y queda explicada en la explicación de las tablas contenida en la sección anterior del documento.

### 11.2.1 Diseño entidad relación corte 1

1

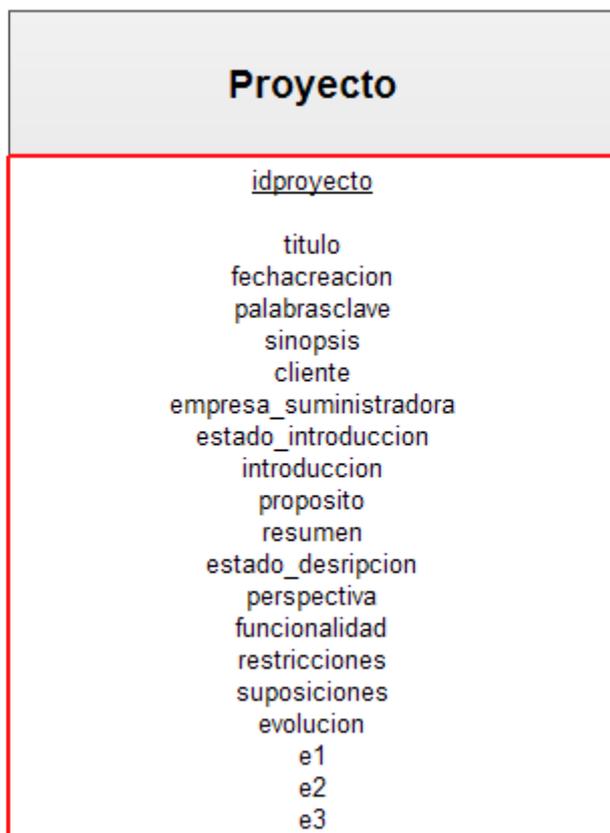


Ilustración 11.2.1 Diagrama ER Corte 1

Este corte muestra los atributos de la tabla principal "proyecto" la cual hereda su identificador a las demás tablas.

## 11.2.2 Diseño entidad relación corte 2

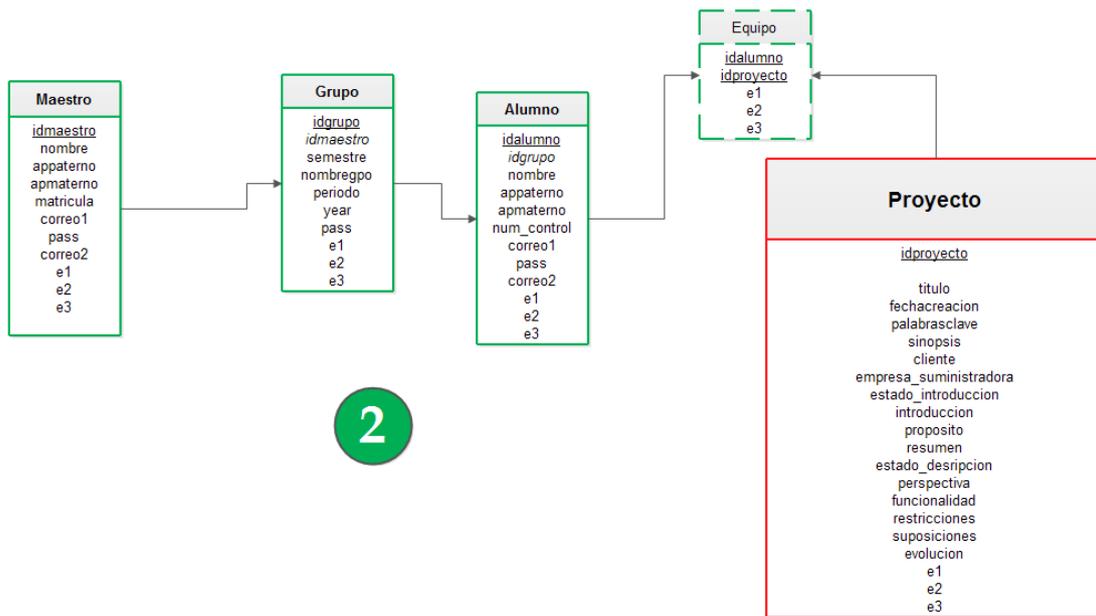


Ilustración 11.2.2 Diagrama ER Corte 2

Este corte muestra la relación de las tablas Maestro, Grupo y Alumno. Estas tablas se relacionan entre sí y la última se relaciona con cardinalidad M:N a la tabla principal Proyecto creando una tabla adicional Equipo.

### 11.2.3 Diseño entidad relación corte 3

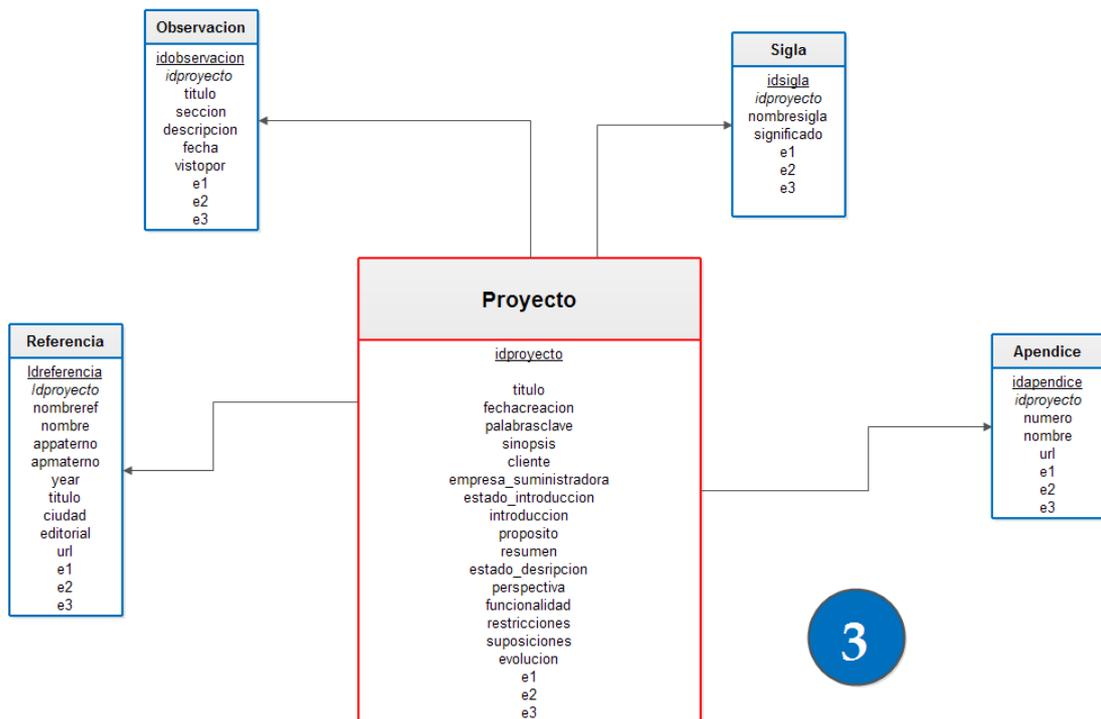


Ilustración 11.2.3 Diagrama ER Corte 3

Este corte muestra las relaciones de las tablas Referencia, Observacion, Sigla y Apendice con la tabla principal proyecto, todas se relacionan de la forma 1:N y heredan la clave *idproyecto*

### 11.2.4 Diseño entidad relación corte 4

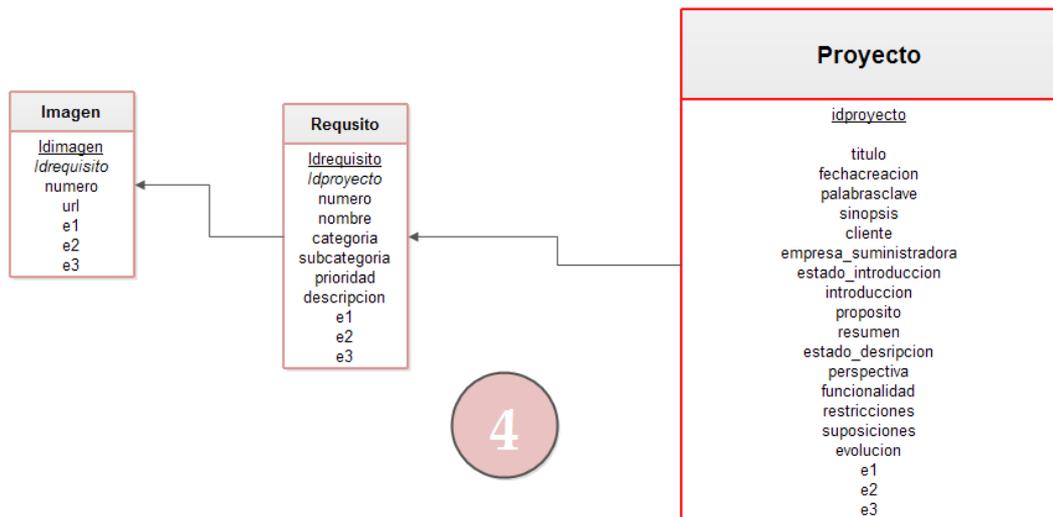


Ilustración 11.2.4 1 Diagrama ER Corte 4

Este corte muestra la relación de la tabla requisito, con la tabla proyecto. Es una relación 1:N dónde Requisito hereda la clave *ldproyecto* de la tabla proyecto.

### 11.2.5 Diseño entidad relación corte 5

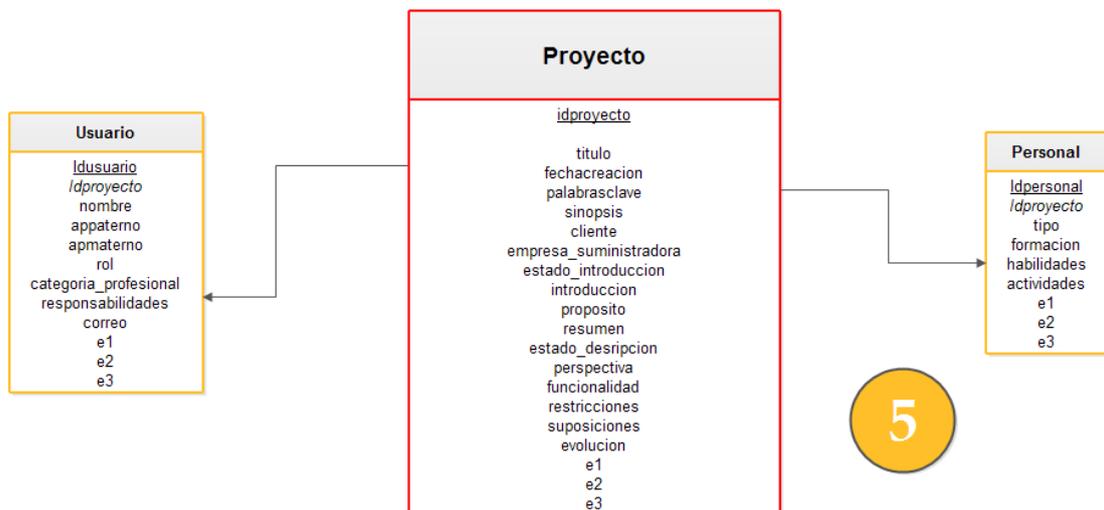


Ilustración 11.2.5 Diagrama ER Corte 5



**Registrar información del alumno:** Despliega en pantalla un formulario de registro, en la que los alumnos podrán inscribirse a un grupo de trabajo, para poder utilizar la herramienta.

**Registrarse en un grupo:** Se muestra después de que el alumno registra sus datos, y en esta sección aparece un pequeño formulario en la que el alumno selecciona el grupo a inscribirse y proporciona la contraseña del grupo para poder hacerlo.

**Información de contacto:** Muestra a los usuarios las maneras en que pueden contactarse para cualquier duda o comentario (correo electrónico, teléfono)

**Maestro:** Al iniciar sesión se muestra su pantalla principal

**Página principal maestro:** Despliega un mensaje de bienvenida, y algunos consejos para utilizar la herramienta.

**Grupos:** Muestra el menú de opciones sobre las acciones que puede realizar el maestro respecto a sus grupos de trabajo.

**Guía de uso:** Muestra la guía de uso de la herramienta para el maestro.

**Ver/editar grupos:** Muestra una lista de los grupos creados por el maestro, la lista incluye el nombre, semestre, año, y la cantidad de alumnos inscritos, así como las opciones de ver lista de alumnos, ver los equipos de cada grupo, y editar un grupo existente.

**Crear grupo nuevo:** Formulario con los datos para crear un grupo nuevo.

**Ver lista:** Muestra la lista de los alumnos inscritos al grupo seleccionado, muestra también la cantidad de documentos en las que el alumno es miembro.

**Ver equipos:** Muestra los equipos de trabajo dentro del grupo seleccionado. Da la opción de revisar el documento de cada equipo y realizar observaciones así como también ver o editar las observaciones realizadas con anterioridad.

**Editar grupo:** Formulario para cambiar los datos de un grupo seleccionado.

**Hacer observaciones:** Muestra el documento generado en pdf con los datos que los miembros del equipo han capturado hasta el momento, en un costado se encuentra un formulario para ir agregando observaciones conforme se está leyendo el documento.

**Editar observaciones:** Lista de observaciones realizadas a un documento seleccionado, con la opción de abrirlas para modificar su contenido o simplemente consultar lo que se escribió.

**Alumno:** muestra la página principal del alumno.

**Página principal alumno:** Muestra un mensaje de bienvenida al alumno, así como también unos consejos para utilizar la herramienta.

**Guía de uso:** Muestra la guía de uso de la herramienta para el alumno.

**Documentos:** Muestra el menú de opciones que el alumno puede realizar con un documento, ya sea la creación de uno nuevo, o la edición de uno existente.

**Crear Nuevo documento:** Muestra un campo obligatorio en el que el alumno deberá ingresar el título del documento a crear.

**Página principal de edición del documento:** Muestra un menú para acceder a cada sección del documento a editar.

**Ver/Editar documentos:** Muestra una lista con todos los documentos creados por el alumno, y da la opción de ver las observaciones, el equipo, generar el documento en pdf, editar el documento, y en caso de ser el jefe de equipo, también le dará la opción de eliminar el documento.

**Ver lista de observaciones:** Muestra la lista de observaciones realizadas al documento seleccionado.

**Leer observación:** Muestra la descripción de la observación seleccionada en la lista de observaciones.

**Ver equipo:** Muestra una lista con el nombre de los miembros que forman parte del equipo de trabajo, si es el jefe de equipo, da la opción de agregar más miembros o eliminar miembros.

**Lista de miembros del equipo:** lista con el nombre y rango de cada integrante del equipo de trabajo.

**Agregar miembro:** Lista con todos los alumnos inscritos al grupo para que el jefe de grupo seleccione a los que desea agregar al equipo de trabajo del documento.

**Eliminar miembro:** Elimina a un miembro seleccionado del equipo de trabajo del documento.

**Generar documento pdf:** Muestra en pantalla el documento generado en formato pdf, con la información capturada y guardada por todos los miembros del equipo de trabajo del documento.

**Eliminar documento:** Elimina todos los registros relacionados con el documento seleccionado, solo el jefe de equipo puede eliminar el documento.

**Editar documento:** Muestra el menú de edición de documento.

**Datos generales:** Formulario con los datos generales del proyecto.

**Introducción:** despliega un menú con las subsecciones editables que integran la introducción del documento.

**Introducción (submenú):** Formulario de captura así como la ayuda de la introducción del documento.

**Propósito:** Formulario de captura así como la ayuda y propósito del documento.

**Alcance:** Formulario de captura así como la ayuda del alcance del documento.

**Resumen:** Formulario de captura así como la ayuda del resumen del documento.

**Personal:** Muestra el submenú para agregar o editar el personal del documento.

**Ver/Editar personal:** Muestra una lista con el personal agregado y da la opción de seleccionar a alguno para editar sus datos.

**Agregar personal:** Formulario de captura así como la ayuda para agregar un nuevo personal al documento.

**Referencias:** Muestra un submenú para agregar o editar referencias del documento.

**Ver o editar referencias:** Muestra una lista con las referencias agregadas y da la opción de seleccionar a alguna para editar sus datos.

**Agregar referencia:** Formulario de captura así como la ayuda para agregar una nueva referencia al documento.

**Definiciones, acrónimos y abreviaturas:** Muestra un submenú para agregar o editar definiciones, acrónimos o abreviaturas del documento.

**Ver o editar Definiciones, acrónimos y abreviaturas:** Muestra una lista con las definiciones, acrónimos o abreviaturas agregadas y da la opción de seleccionar a alguna para editar sus datos.

**Agregar Definiciones, acrónimos y abreviaturas:** Formulario de captura así como la ayuda para agregar nuevas definiciones, acrónimos o abreviaturas al documento.

**Descripción:** despliega un menú con las subsecciones editables que integran la descripción general del documento.

**Perspectiva:** Formulario de captura así como la ayuda de la perspectiva del documento.

**Funcionalidad:** Formulario de captura así como la ayuda de la funcionalidad del documento.

**Restricciones:** Formulario de captura así como la ayuda de las restricciones del documento.

**Usuario:** Muestra un submenú para agregar o editar usuarios del documento.

**Ver o editar usuarios:** Muestra una lista con los usuarios agregados y da la opción de seleccionar a alguno para editar sus datos.

**Agregar usuario:** Formulario de captura así como la ayuda para agregar un nuevo usuario al documento.

**Suposiciones y dependencias:** Formulario de captura así como la ayuda de las suposiciones y dependencias del documento.

**Evolución:** Formulario de captura así como la ayuda de la evolución del documento.

**Requisitos:** Muestra un submenú para agregar o editar requisitos del documento.

**Ver o editar requisitos:** Muestra una lista con los requisitos agregados y da la opción de seleccionar a alguno para editar sus datos.

**Agregar requisito:** Formulario de captura así como la ayuda para agregar un nuevo requisito al documento.

**Apéndices:** Muestra un submenú para agregar o editar apéndices del documento.

**Ver o editar apéndices:** Muestra una lista con los apéndices agregados y da la opción de seleccionar a alguno para editar sus datos.

**Agregar apéndice:** Formulario de captura así como la ayuda para agregar un nuevo apéndice al documento.

**Terminar edición:** Finaliza la edición del documento y regresa al usuario a la lista de documentos creados.

## 11.4 Diagrama de despliegue

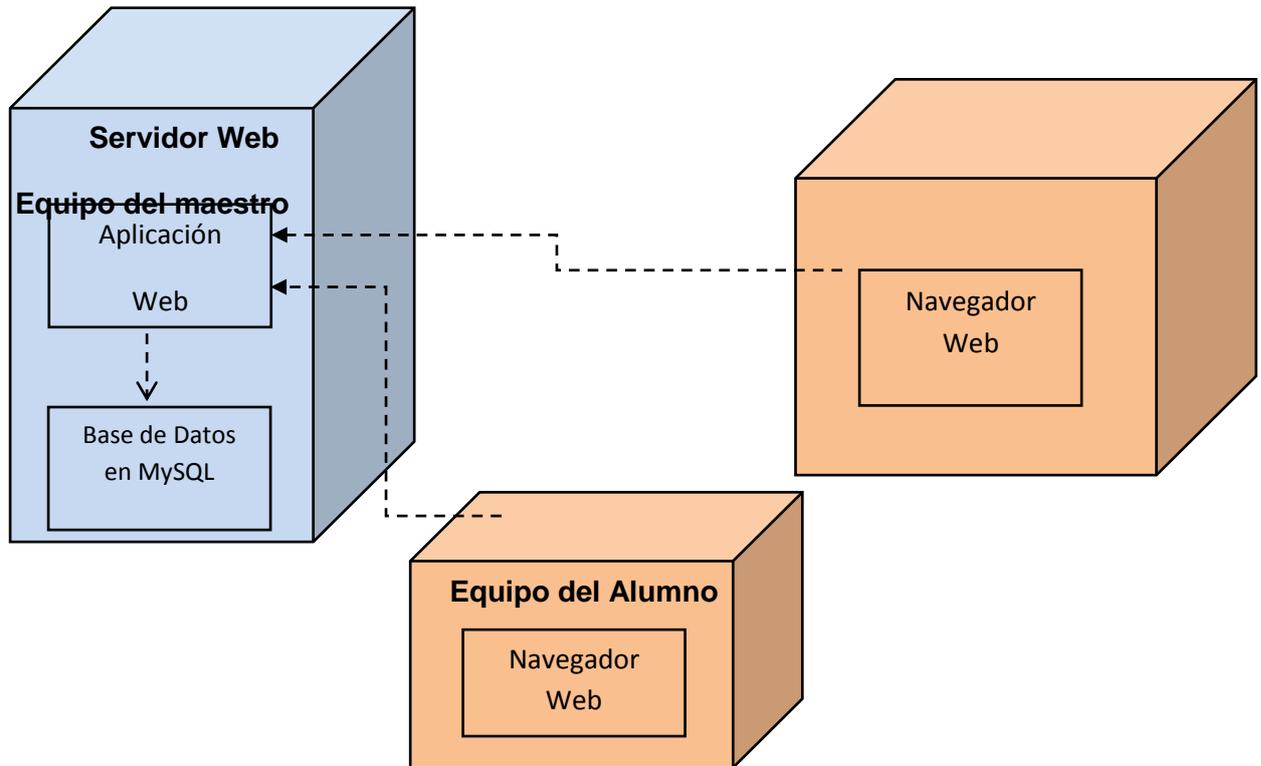


Ilustración 11.4 Diagrama de despliegue 1

El diagrama de despliegue sirve para identificar cada uno de los componentes de hardware del sistema,

## **12. Conclusiones y recomendaciones**

Las conclusiones a las que podemos llegar en el desarrollo de esta herramienta son satisfactorias. La herramienta servirá como apoyo al área de ingeniería de software utilizándola en el laboratorio por los alumnos. La utilización de esta herramienta facilitará la comprensión y la correcta elaboración de los ERS en el futuro.

Así también ayudará a la gestión e integración de grupos y equipos de trabajo, facilitando la elaboración del documento de especificación de requisitos y el control por parte del maestro encargado de impartir la asignatura.

Se espera que la herramienta se utilice y se adapte como un estándar de desarrollo para futuros trabajos y proyectos de desarrollo de Software de la institución.

Se recomienda que los usuarios del sistema lean la guía para un correcto funcionamiento de la aplicación y el estándar IEEE-830 para la comprensión de las diferentes partes del documento de especificación de requisitos

### 13. Referencias

[Chalmeta, 1999]: Chalmeta R. "ADSI II. 2º Boletín de transparencias". UJI, 1999.

[Flores, 2005]: Flores, J. "Sistema de Administración de Red (S.A.R)". Versión 1. Tesis Ing. Electrónica y Telecomunicaciones, Univ. Autónoma del Estado de Hidalgo, Facultad de Ciencias Básicas e Ingeniería, 2005.

[Fowler y Scott, 1999] "UML Gota a Gota" Pearson Educación, 1999

[Gegovia]: Gegovia, F. "Introducción a la ingeniería del software: Modelos de desarrollo de programas". 1ª ed. Delta Publicaciones Universitarias. Madrid, España, 2005.

[ [INEI, 1999]: Instituto Nacional de Estadística e Informática (INEI). "Herramientas CASE". 875-99-OI-OTDETI-INEI, 1999.

[INTECO, 2009]: Instituto Nacional de Tecnologías de la Comunicación (INTECO). "Ingeniería del Software: Metodologías y Ciclos de Vida". 1ª ed. Laboratorio Nacional de Calidad del Software, Marzo de 2009.

[Lawrence , 2002]: Lawrence, S. "Ingeniería de Software, Teoría y Práctica". 1ª ed. Editorial Prentice Hall. ISBN 987-9460-71-5. Pp. 52-54, 2002

[Loucopoulos , 1995]: Loucopoulos P. "System Requirements Engineering". McGraw-Hill International series in Software Engineering, ISBN 0-07-707843-8, 1995.

[Piattini, 1996]: Piattini Mario G. Análisis y diseño detallado de aplicaciones informáticas de gestión. 1ª ed. RA-MA Editorial, Madrid, 1996.

[Pressman, 1998] Pressman R. "Ingeniería del Software, Un Enfoque Práctico". 4ª ed. Editorial McGraw-Hill. ISBN 84-481-1186-9, Pp. 40, 1998.

[Thomas , 2005]: Thomas, P. "Definición de un Proceso de Elicitación de Objetivos". Tesis Mag. Ing. de Software, Univ. Nacional de la Plata, Fac. de Informática. Pp 111, 2005.

[Sommerville , 2005]: Sommerville, I. "Ingeniería de Software". 7ª ed. Editorial Prentice Hall.Pp. 123, 2005.

[upm, 2000]: Web de la universidad politécnica de Madrid, aparecen algunos artículos sobre la IEEE 830 y sobre análisis de requisitos en general. <http://www.upm.es>

[w3c, 2012]: Web de la comunidad internacional encargada de desarrollar normas abiertas para el desarrollo en la web. <http://www.w3.org/XML/>