



PONTIFICIA **UNIVERSIDAD CATÓLICA** DEL PERÚ

Esta obra ha sido publicada bajo la licencia Creative Commons
Reconocimiento-No comercial-Compartir bajo la misma licencia 2.5 Perú.

Para ver una copia de dicha licencia, visite
<http://creativecommons.org/licenses/by-nc-sa/2.5/pe/>



PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA



ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE INFORMACIÓN
PARA LA GESTIÓN Y EVALUACIÓN DEL IMPACTO EN LA SOCIEDAD DE
PROYECTOS DE RESPONSABILIDAD SOCIAL UNIVERSITARIA

Tesis para optar por el Título de Ingeniero Informático

Presentada por:

Nelly Elizabeth Trevejo Zamora

LIMA – PERÚ

2008

RESUMEN

La responsabilidad social universitaria es un tema de importancia para la mayoría de universidades del Perú. El aprendizaje que se logra fuera de las aulas contribuye en gran medida a la formación de los estudiantes pues ponen en práctica sus conocimientos y habilidades en la solución de problemas sociales y a la vez brindan a la comunidad herramientas para alcanzar un desarrollo sostenible. Es por ello que se presta especial atención en la búsqueda de herramientas que faciliten la labor que se realiza en los proyectos de responsabilidad social y que a la vez sean fuente de retroalimentación para futuros proyectos que se realicen en este ámbito.

Un proyecto de responsabilidad social universitario tendrá éxito si es que se logra que tanto los alumnos como la comunidad se beneficien. Por esta razón se requiere contar con información organizada y herramientas que permitan controlar los procesos dentro de un proyecto de responsabilidad social, el avance del mismo y el cumplimiento de objetivos.

En todo proyecto se busca que las personas encargadas de tomar decisiones y organizar las actividades puedan disponer de información relevante que les permita continuar con su labor a pesar de los constantes cambios en los miembros del proyecto y; en caso no se cumplan los objetivos, planificar acciones correctivas.

Las universidades que promueven estos proyectos también tienen la necesidad de monitorizar los proyectos y analizar en qué medida impactan en el aprendizaje de los alumnos y en la comunidad con la finalidad de realizar mejoras a futuro y poder llevar una adecuada gestión de los proyectos.

El presente proyecto de fin de carrera propone realizar la implementación de un sistema de información para la gestión y evaluación del impacto en la sociedad de proyectos de responsabilidad social universitaria. Este sistema permitirá a los usuarios contar con toda la información del proyecto y será también una herramienta útil para la organización de los procesos y actividades que se desarrollan, así como la evaluación de los avances y el impacto que puedan tener en el desarrollo de la sociedad.

DEDICATORIA

*A los dos tesoros más preciados que
tengo: Mayrita y Marycielito, a mi mamita Lola 
a quién recuerdo a cada instante de mi vida,
y al ser a quién más amo, Jesús.*



AGRADECIMIENTO

A mis padres, quiénes con esfuerzo y dedicación me apoyaron a lo largo de mi carrera. A mis hermanas Mayra y Marycielo, quiénes siempre están conmigo apoyándome y aconsejándome. A mi mamita Esther, quién siempre está pendiente de mi y quién muchas veces se desveló conmigo a lo largo de estos años. A mi mamita Lola, quién me han llenado de amor, cariño y ternura y me ha enseñado el sentido de la humildad. A mi asesor, el Ingeniero Oscar Maldonado, quién ha sabido guiarme en esta difícil labor.

Finalmente, quiero agradecer a Dios, por todo lo que me ha dado y por el amor que me tiene y me demuestra a cada instante.

INDICE GENERAL

Introducción.....	1
1. Capítulo 1: Generalidades.....	3
1.1. Definición del Problema.....	3
1.2. Marco conceptual del problema.....	4
1.2.1. Responsabilidad Social Universitaria (RSU).....	5
1.2.2. Programas de RSU.....	6
1.2.3. Proyectos de RSU.....	7
1.3. Plan del proyecto de fin de carrera.....	14
1.3.1. Métodos y Procedimientos.....	14
1.3.2. Planificación.....	22
1.4. Estado del Arte.....	33
1.4.1. Aplicaciones para la Gestión y Planificación de Proyectos.....	33
1.4.2. Aplicaciones para el almacenamiento y organización de la documentación.....	35
1.4.3. Aplicaciones para el manejo de historial de atenciones clínicas.....	37
1.5. Descripción y sustentación de la solución.....	38
1.5.1. Módulos de seguridad y configuración de parámetros.....	39
1.5.2. Módulo de gestión de documentos.....	39
1.5.3. Módulo de programación, monitoreo y auditoría de proyectos de RSU.....	39
1.5.4. Módulo de atención a la comunidad.....	40
1.5.5. Módulo de evaluación de impacto.....	40
1.5.6. Módulo de reportes.....	40
2. Capítulo 2: Análisis.....	40
2.1. Definición de la metodología de la solución.....	42
2.1.1. Fases del desarrollo de software según RUP.....	42
2.1.2. Disciplinas bajo las cuales se desarrolla el ciclo de vida por cada iteración.....	43
2.1.3. Elementos de la Metodología RUP.....	45
2.2. Identificación de los requerimientos.....	46
2.2.1. Requerimientos Funcionales.....	46
2.2.2. Requerimientos no funcionales.....	48

2.3.	Análisis de la Solución.....	49
2.3.1.	Necesidades del usuario.....	50
2.3.2.	Evaluación de viabilidad.....	50
2.3.3.	Consideraciones sobre el Sistema.....	52
2.3.4.	Análisis técnico y económico.....	53
2.3.5.	Definición del sistema.....	56
3.	Capítulo 3: Diseño.....	61
3.1.	Arquitectura de la Solución.....	61
3.1.1.	Vista lógica.....	63
3.1.2.	Vista de despliegue.....	65
3.2.	Diseño de la Interfaz Gráfica.....	66
3.3.	Consideraciones del Diseño Propuesto.....	70
4.	Capítulo 4: Construcción.....	68
4.1.	Construcción.....	71
4.1.1.	Herramientas para la implementación.....	71
4.1.2.	Patrones de programación.....	73
4.2.	Pruebas.....	75
4.2.1.	Planteamiento de casos de prueba.....	75
4.2.2.	Criterios de aceptación o rechazo.....	77
4.2.3.	Reporte de ejecución de pruebas.....	85
5.	Capítulo 5: Observaciones, conclusiones y recomendaciones	87
5.1.	Observaciones.....	87
5.2.	Conclusiones.....	88
5.3.	Recomendaciones y trabajos futuros.....	89
	Bibliografía.....	90

INDICE DE FIGURAS

Figura 1.1. Comunidad de Aprendizaje [3].....	6
Figura 1.2. Estudio, Planificación y Organización del Proyecto.....	11
Figura 1.3. Ejecución de Actividades	13
Figura 1.4. Procesos Básicos según PMBOK[19].....	14
Figura 1.5. Tareas de la Etapa de Concepción	24
Figura 1.6. Tareas de la Etapa de Elaboración	25
Figura 1.7.1. Tareas de la Etapa de Construcción – Iteración 1.....	26
Figura 1.7.2. Tareas de la Etapa de Construcción – Iteración 2.....	27
Figura 1.7.3. Tareas de la Etapa de Construcción – Iteración 3.....	27
Figura 1.7.4. Tareas de la Etapa de Construcción – Iteración 4.....	28
Figura 1.7.5. Tareas de la Etapa de Construcción – Iteración 5.....	28
Figura 1.8. Tareas de la Etapa de Transición.....	29
Figura 1.9.1. Diagrama de Gantt – Etapas de Concepción y Elaboración.....	30
Figura 1.9.2. Diagrama de Gantt – Etapa de Construcción correspondiente a las iteraciones 1, 2 y 3.....	31
Figura 1.9.3. Diagrama de Gantt – Etapa de Construcción correspondiente a las iteraciones 4 y 5.....	32
Figura 1.9.4. Diagrama de Gantt – Etapa de Transición.....	32
Figura 2.1. Fases de la metodología RUP[20].....	43
Figura 2.2. Fases e iteraciones de la metodología RUP[20].....	44
Figura 2.3. Paquetes del Sistema.....	57
Figura 2.4. Diagrama de clases.....	59
Figura 3.1. Diagrama de componentes.....	64
Figura 3.2. Diagrama de despliegue.....	65
Figura 3.3. Distribución de la Interfaz Gráfica.....	67
Figura 3.4. Pantalla del Sistema de Información.....	69
Figura 4.1. Menú de acceso a las herramientas proporcionadas por la aplicación VertrigoServ.....	73
Figura 4.2. Estructura de clases del patrón MVC.....	75

INDICE DE TABLAS

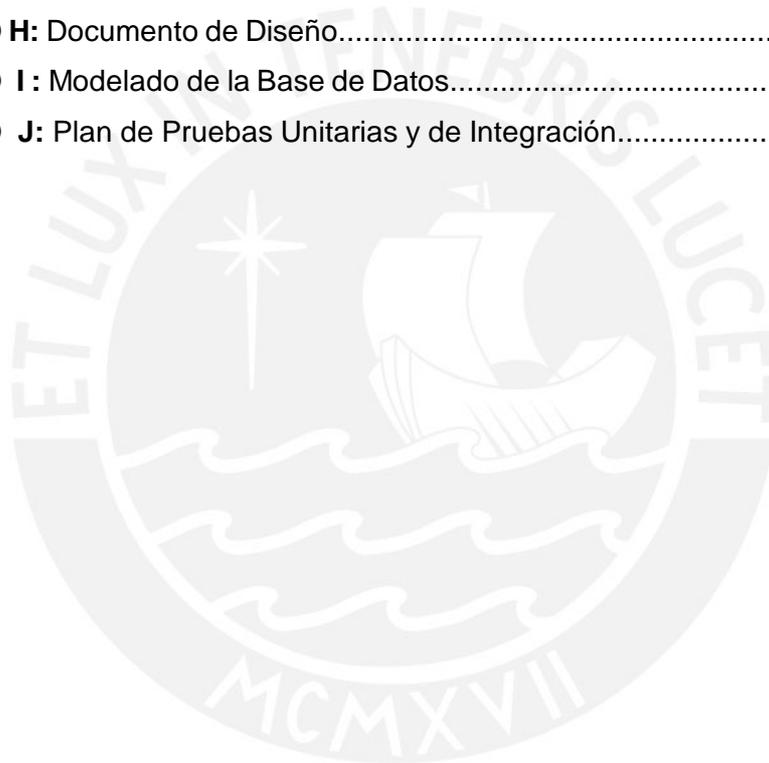
Tabla 1.1. Tareas del Proyecto.....	16
Tabla 1.2. Tareas a realizarse durante el desarrollo.....	23
Tabla 2.1. Presupuesto para el desarrollo del sistema.....	55



INDICE DE ANEXOS

(Ver sección de anexos)

ANEXO A: Indicadores para proyectos de desarrollo.....	2
ANEXO B: Catálogo de Requisitos.....	5
ANEXO C: Documento de Visión.....	15
ANEXO D: Plan de Proyecto.....	31
ANEXO E: Especificación de Requisitos de Software.....	41
ANEXO F: Documento de Análisis.....	171
ANEXO G: Documento de Arquitectura.....	201
ANEXO H: Documento de Diseño.....	211
ANEXO I: Modelado de la Base de Datos.....	217
ANEXO J: Plan de Pruebas Unitarias y de Integración.....	219



INTRODUCCIÓN

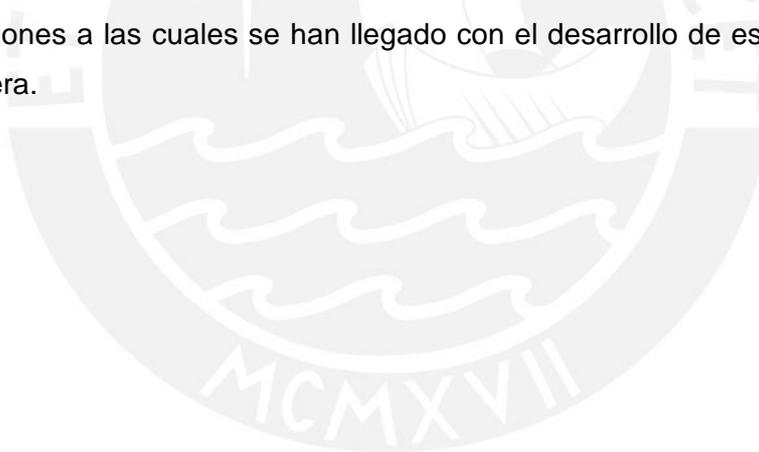
Uno de los problemas más sobresalientes de los proyectos de responsabilidad social universitaria es que no se cuenta con información oportuna sobre los avances del proyecto, las decisiones tomadas, el calendario de actividades, entre otros. Dicha información es vital para el avance normal de los proyectos, pues al haber un constante cambio en los miembros de los equipos, la información se pierde y se comienzan a elaborar tareas y actividades desde cero.

Por otro lado, el hecho de no tener bien definidos los objetivos y asegurarse que todas las actividades que se realicen apunten al cumplimiento de los mismos conlleva al fracaso del proyecto, lo cual afecta a los alumnos universitarios, quienes no logran un aprendizaje continuo dado que no saben cómo focalizar sus conocimientos, y a los miembros de la comunidad, quienes no alcanzan un desarrollo sostenible puesto que se les brinda diversas herramientas sin tener un objetivo definido.

A pesar de lo expuesto, en el Perú no se cuenta con herramientas que brinden una solución a estos problemas, por el contrario, los proyectos de responsabilidad social se realizan sin considerar fuentes de retroalimentación que les permitan mejorar los procesos y alcanzar el cumplimiento de los fines para los cuales se han creado estos proyectos.

Por lo anteriormente expuesto, es necesario contar con una herramienta que permita a los miembros del proyecto tomar decisiones que apunten al logro de los objetivos y que les facilite las labores que realizan en la gestión de proyectos de responsabilidad social universitaria. Por ello, el presente trabajo de tesis tiene como finalidad brindar una herramienta que solucione los problemas descritos y que sea viable de acuerdo a las condiciones en las que la mayoría de proyectos de responsabilidad social se desarrolla.

El contenido del presente trabajo está dividido en cinco capítulos. En el primer capítulo se define el problema que se busca resolver, la solución que se plantea, y de qué manera se llevará a cabo el proyecto planteado. En el segundo capítulo, se realiza el análisis de la solución propuesta, lo cual involucra la metodología utilizada, requerimientos del usuario y la viabilidad de la solución. El tercer capítulo abarca el diseño de la solución, esto es, la arquitectura elegida para el desarrollo. El cuarto capítulo trata sobre la construcción de la solución, es decir, el desarrollo del producto, las herramientas empleadas, así como las pruebas realizadas. Y por último, el quinto capítulo contiene las observaciones, recomendaciones y conclusiones a las cuales se han llegado con el desarrollo de este proyecto de fin de carrera.



1. Capítulo 1: Generalidades

En este capítulo se describirá el contexto en el cual se desarrolla el problema así como el detalle del mismo. De esta manera, el lector podrá comprender la situación real y posteriormente la solución propuesta.

Así mismo, se detallará la planificación de las tareas y actividades a realizarse durante el desarrollo del presente proyecto de fin de carrera con la finalidad de cumplir con el alcance planteado para el mismo.

1.1. Definición del Problema

En la actualidad, las universidades peruanas están dando especial énfasis al tema de Responsabilidad Social Universitaria (RSU), puesto que éste se ha presentado a lo largo de los últimos años como un desafío, dado que este concepto implica que el compromiso de las universidades no sólo sea formar profesionales de alto nivel, sino también personas comprometidas con el desarrollo de su país y capaces de articular los conocimientos adquiridos con el desarrollo participativo de la comunidad. Para ello, se desarrollan proyectos de RSU, los cuales tienen por finalidad brindar servicios, dar asistencia, educar e informar a la población en diversos temas de interés, previo análisis de su situación.

Los alumnos universitarios que participan en los proyectos de RSU no cuentan con equipos de cómputo designados exclusivamente para el proyecto pues las zonas en las que estos se realizan son variables de acuerdo al tipo de actividad a realizarse y la disponibilidad de ambientes. Así mismo, los alumnos universitarios que participan de algún proyecto realizan esta labor de acuerdo a su disponibilidad de tiempo, por lo que cambian constantemente. Como consecuencia, los nuevos miembros del equipo no cuentan con información actualizada sobre los avances del proyecto y no existen registros de las atenciones y talleres brindados. Adicionalmente, existe también una mala administración de los recursos del proyecto, constituidos por las personas, la documentación, el material empleado y los gastos realizados. Así mismo, a lo largo de los proyectos se realizan evaluaciones y encuestas para medir el impacto de los mismos en la sociedad y de esta manera evaluar su progreso. Sin embargo, no hay un registro de los indicadores utilizados, ni tampoco un análisis detallado de los mismos. Por ello, muchas veces el aporte que se trata de dar a la comunidad es repetitivo o inconsistente con las actividades realizadas previamente. Todo esto afecta directamente a la comunidad puesto que no hay un aporte real a su desarrollo, e indirectamente a los miembros del equipo, quienes pierden tiempo en tareas realizadas previamente.

Según el análisis realizado, luego de entrevistas y encuestas a participantes de diversos proyectos de RSU, no existe una verdadera fuente de retroalimentación, es por eso que se requiere un producto de software que permita centralizar la información de los proyectos y automatizar los registros de atención al público, con la finalidad de contar con toda la información necesaria para que los nuevos miembros puedan continuar teniendo como base el trabajo realizado. Dicha información debe estar disponible en todo momento y desde cualquier lugar pues, como se mencionó anteriormente, se debe tener en cuenta que los alumnos universitarios que participan en los proyectos de responsabilidad social trabajan en los mismos de acuerdo a su disponibilidad de tiempo.

1.2. Marco conceptual del problema

En esta sección se explicarán los conocimientos mínimos requeridos para la comprensión del problema, cuya solución se plantea en el presente trabajo de fin de carrera.

1.2.1. Responsabilidad Social Universitaria (RSU)

La Responsabilidad Social (RS) es un enfoque que promueve visiones y actitudes nuevas frente a los problemas sociales, promueve el compromiso con la sociedad, de manera tal que los ciudadanos no sean observadores de los problemas que atañen al país sino agentes activos en el cambio que plantean programas de desarrollo sostenible. Se puede decir que la RS es la movilización de recursos, destrezas y conocimientos, recursos materiales, humanos y económicos, y redes de relaciones personales, para solucionar problemas que afectan a toda la sociedad y que es de interés de todos resolver. [1] (Mónica Jiménez de Barros, 2001)

Conociendo la definición de RS, se puede hablar de la Responsabilidad Social Universitaria (RSU). RSU es un enfoque que promueve la participación de la Universidad dentro del campo de la RS. La Universidad no solo tiene la responsabilidad de formar a los alumnos en el campo académico y de investigación, sino que también debe contribuir con el desarrollo social elevando los niveles de vida de la población y, de esta manera, participar en la transformación de la sociedad.

Para cumplir con esta nueva meta que se trazan las universidades, es necesario conseguir la integración de la formación académica y de investigación con la responsabilidad social, es por ello que se plantean principios y objetivos que conducen a buenas prácticas de RS.

La RSU considera que la formación profesional y humana de los universitarios debe basarse en el aprendizaje a través de la participación en proyectos de desarrollo social, de esta manera, ellos son directamente beneficiados dado que ponen en práctica sus conocimientos y requerirán de investigación académica para dar solución a los problemas. Por lo tanto, se creará una comunidad de aprendizaje, que consiste en llevar el proceso de enseñanza y aprendizaje fuera de la universidad, permitiendo la formación de nuevos espacios de aprendizaje. Por lo tanto, actores no universitarios y situaciones no académicas pueden construir nuevos métodos de aprendizaje para todos los involucrados. [2](François Vallaey)

En la figura 1.1. se muestran los componentes que intervienen en la creación de comunidades de aprendizaje, los cuales son el voluntariado, la investigación y la formación académica. Dichos componentes son esenciales para conseguir una buena formación académica a través de proyectos de RSU que sean beneficiosos para los alumnos universitarios y para la comunidad.

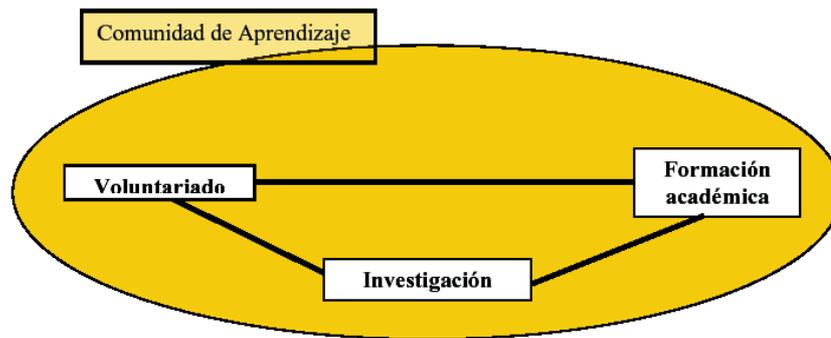


Figura 1.1. Comunidad de Aprendizaje [3]

1.2.2. Programas de RSU

Los Programas de RSU son proyectos que buscan contribuir con el desarrollo sostenible de una determinada población o asociación. Estos proyectos siguen el enfoque de RSU y pueden ser de dos tipos:

- Proyectos que contribuyen con la solución de un problema social
- Proyectos que contribuyen con el desarrollo de determinada zona geográfica.

A. Indicadores para proyectos de RSU

Los indicadores para proyectos de RSU son indicadores para la medición del cumplimiento de ciertos principios, que permiten asegurar la finalidad que debe cumplir todo proyecto de responsabilidad social universitaria orientado al desarrollo sostenible .

Según François Vallaëys, en el documento de Indicadores de desarrollo en comunidades de aprendizaje [4], se tienen definidos indicadores para la evaluación de proyectos de desarrollo que permiten cumplir con los principios y objetivos de un proyecto de responsabilidad social universitaria orientado al desarrollo sostenible. Dichos indicadores se encuentran detallados en el Anexo A.

1.2.3. Proyectos de RSU

Los proyectos de RSU son proyectos que tienen como finalidad lograr el desarrollo sostenible de la comunidad. A continuación se definen los principales puntos asociados a este tipo de proyectos.

A. Objetivo Principal

El objetivo principal de los proyectos de desarrollo sostenible es contribuir con el desarrollo de la comunidad a través de campañas de asistencia, talleres y charlas educativas, en los cuales alumnos de diferentes especialidades y universidades de Lima ponen en práctica sus conocimientos para dar solución a los problemas sociales que se presentan. De esta manera se brinda a la comunidad herramientas necesarias para que puedan iniciar un plan de mejora continua a largo plazo.

B. Participación Social

Los proyectos de desarrollo sostenible involucran la participación no solo de la población, sino también de municipios, universidades y otras instituciones con la finalidad de formar equipos de trabajo interdisciplinarios para atacar los principales problemas encontrados:

- Violencia
- Abandono de menores
- Madres solteras
- Violación Sexual
- Desnutrición

- Presencia de enfermedades causadas por ausencia de condiciones mínimas de higiene
- Discriminación
- Drogadicción

C. Actores

Actores son todas aquellas personas involucradas en las diferentes etapas de los proyectos de desarrollo sostenible. Cada uno de ellos tiene roles específicos e intervienen en diferentes momentos; sin embargo, es necesario que todos se involucren en la misma medida para el logro de los objetivos.

Existen comúnmente los siguientes actores:

- Docentes de las Universidades: Actor cuya función es apoyar en el campo del aprendizaje a lo largo del proyecto, verificar si es que están aplicando sus conocimientos de forma correcta y en beneficio de la población, analizar el impacto académico y el impacto social del proyecto realizado.
- Alumnos universitarios: Actor cuya función es organizar campañas, talleres y charlas educativas, bajo la tutoría de un docente, aplicando sus conocimientos para la solución de los problemas encontrados. Otra de sus funciones es analizar el impacto social y el cumplimiento de los objetivos trazados.
- Municipios: Actor cuya función es brindar las facilidades necesarias para el normal desarrollo de los proyectos como son: formar comités dentro de la población, incentivar a la población a participar en los proyectos, coordinar con los alumnos fechas para encuestas y evaluación del avance del proyecto.
- Miembros del Comité Organizacional: Actor cuya función es proveer los ambientes necesarios para la realización de las actividades a realizarse, organizar a la población durante la actividad para evitar el desorden, informar a los vecinos sobre los beneficios del proyecto, solicitar

información sobre los objetivos del proyecto y sobre el alcance de los mismos.

- Población Objetiva: Actor cuya función es participar de las actividades realizadas y colaborar con las encuestas y evaluaciones realizadas. Dependiendo del lugar, la población podría estar dada por los miembros de una comunidad en un asentamiento humano, los residentes de algún albergue, entre otros.

D. **Actividades de los Proyectos de RSU**

Son Actividades de este tipo de proyectos todas aquellas reuniones de coordinación, capacitaciones a los miembros del proyecto, campañas de asistencia, talleres y charlas educativas que se encuentran registradas dentro del cronograma de actividades de un proyecto de desarrollo sostenible.

Todas las actividades tienen objetivos específicos que deben tener concordancia con el objetivo principal del proyecto. Adicionalmente, se definen participantes específicos de acuerdo al tipo de actividad y a los objetivos del mismo.

E. **Servicios brindados a la comunidad**

Un servicio es un conjunto de actividades que buscan responder a una o más necesidades, en este caso son necesidades encontradas en la población.

Los servicios que actualmente se brindan durante la ejecución de la mayoría de proyectos son los siguientes:

- Psicología
- Nutrición
- Medicina General
- Odontología

Sin embargo, los proyectos de desarrollo sostenible tienen como finalidad que se involucren otras áreas para poder atender los problemas de la población.

F. Fases de los Proyectos de RSU y Documentación Asociada

En la figura 1.2. se muestra el proceso que se sigue en cada una de las fases de los proyectos de desarrollo sostenible.

Como se ve, en este tipo de proyectos participan activamente la universidad, los alumnos, el municipio y los miembros de la comunidad, cada uno con una labor específica.

➤ Estudio de la Situación y la Problemática

En esta etapa se realiza un profundo análisis a cargo de alumnos de diferentes especialidades de la universidad con el apoyo de la municipalidad de los principales problemas por los que atraviesa la población de las zonas o lugares donde se pretende realizar proyectos de desarrollo sostenible. Como resultado de este análisis, se sabe de qué manera podrá contribuir cada carrera profesional con el desarrollo de esta población. Posteriormente, se elabora el reporte del estudio realizado el cual sirve como fuente principal para el planteamiento de objetivos.

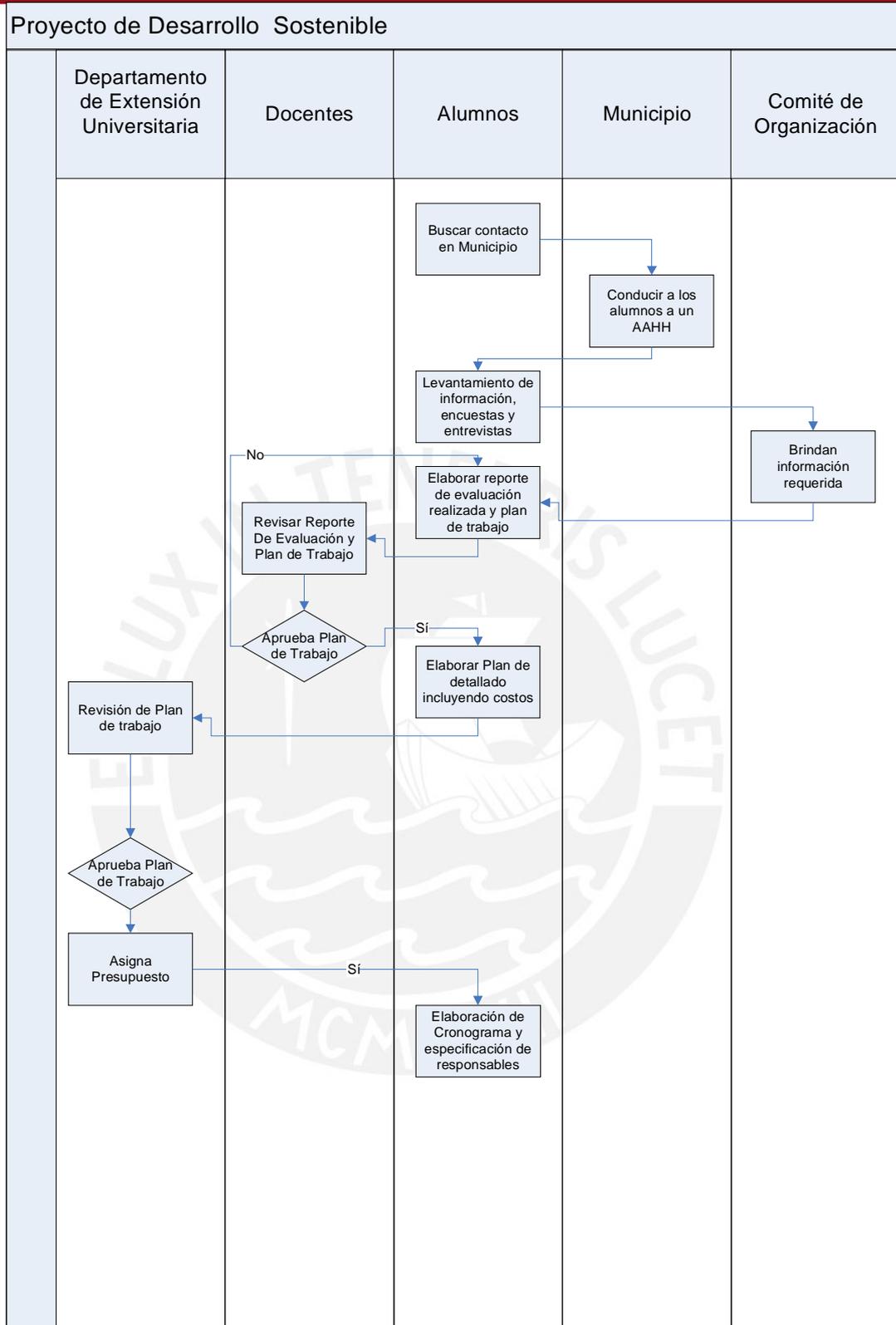


Figura 1.2. Estudio, Planificación y Organización del Proyecto

➤ **Planificación de Proyectos de RSU**

En esta etapa se realiza la planificación del proyecto que involucra las siguientes tareas:

- Determinación de los objetivos generales del proyecto
- Elección de los responsables del proyecto,
- Elaboración del Cronograma de actividades
- Sustentación de Costes del proyecto

Todos los acuerdos tomados están en las actas de reunión, las cuales se difunden a todos los participantes del proyecto. El listado de responsables, el cronograma de actividades y los costes del proyecto son enviados junto con la definición del proyecto, en el cual se incluyen los objetivos del mismo, los organismos encargados de proveer los recursos económicos y los permisos necesarios para la ejecución del proyecto. Una vez aceptado el proyecto se continua con la siguiente etapa.

➤ **Organización de Proyectos de RSU**

Los responsables del proyecto teniendo en cuenta el cronograma de actividades realizarán las siguientes tareas:

- Definición de los objetivos específicos del proyecto
 - Asignación de responsables y participantes en cada actividad
- Asignación de recursos para cada actividad

Todos los acuerdos tomados están en las actas de reunión, las cuales se difunden a todos los participantes del proyecto.

➤ **Ejecución de Proyectos de RSU**

En esta etapa se llevan a cabo las actividades según el cronograma establecido por los participantes del proyecto. Como se muestra en la figura 1.3 , la ejecución de los proyectos de RSU involucran las siguientes tareas:

- Preparación del tema en caso de ser taller, charla o capacitación
- Elaboración del material a utilizarse
- Revisión del material
- Realización de la actividad

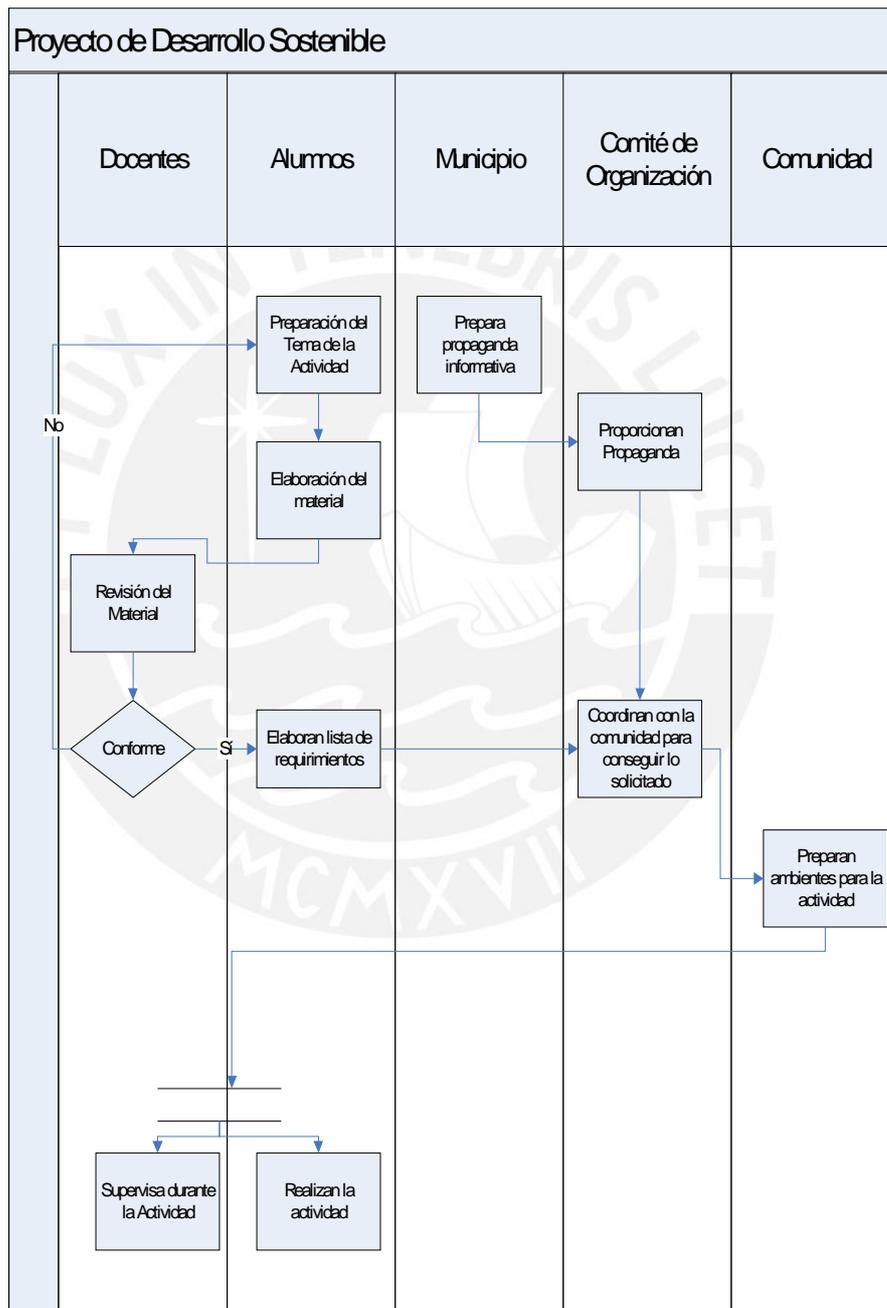


Figura 1.3. Ejecución de Actividades

➤ Control y Evaluación de Proyectos de RSU

En esta etapa, que es paralela a las otras etapas, se llevan a cabo las siguientes tareas:

- Control del cumplimiento de las fechas establecidas
- Control del cumplimiento de responsabilidades asignadas
- Planteamiento de indicadores para medir todos los objetivos establecidos.
- Realización de encuestas y evaluaciones a todos los participantes del proyecto.

1.3. Plan del proyecto de fin de carrera

Para la realización del proyecto de fin de carrera es necesario definir una metodología de gestión a seguir, asegurando de esta manera el cumplimiento de los objetivos trazados al inicio del proyecto y un orden en su desarrollo.

1.3.1. Métodos y Procedimientos

Este proyecto se desarrollará bajo el enfoque de la Guía del PMBOK [19] desarrollada por el Project Management Institute (PMI). El PMBOK reconoce 5 procesos básicos y 9 áreas de conocimiento comunes a casi todos los proyectos.

Los cinco grupos de procesos básicos se muestran en la figura 1.4.



Figura 1.4. Procesos Básicos según PMBOK[19]

Los procesos se traslapan e interactúan y son descritos en términos de: Entradas (documentos, planes, diseños, etc.), herramientas y técnicas (mecanismos aplicados a las entradas) y salidas (documentos, productos, etc.).

Las áreas del conocimiento mencionadas en el PMBOK que se utilizarán para el desarrollo de este proyecto son:

1. Gestión de la Integración de Proyectos
2. Gestión del Alcance en Proyectos
3. Gestión del Tiempo en Proyectos
4. Gestión de la Calidad en Proyectos
5. Gestión del Riesgo en Proyectos



A continuación se indican las tareas de gestión a realizar durante el desarrollo del presente proyecto de fin de carrera agrupadas de acuerdo a los procesos y áreas del conocimiento:

Procesos de un Área de Conocimiento	Grupos de Procesos de Dirección de Proyectos			
	Grupo de Procesos de Iniciación	Grupo de Procesos de Planificación	Grupo de Procesos de Ejecución	Grupo de Procesos de Seguimiento y Control
Gestión de Integración del Proyecto	Desarrollo del enunciado del proyecto	Desarrollo del plan de gestión del proyecto	Gestión de la ejecución del proyecto	Control del proyecto
				Control integral de cambios
Gestión de Alcance del Proyecto		Planificación del alcance		Verificación del alcance
		Definición del alcance		
		Generación de la Estructura de Descomposición del Trabajo (WBS)		Control del alcance
Gestión del Tiempo del Proyecto		Definición de actividades		Control del cronograma
		Secuencia de Actividades		
		Estimación de recursos de actividades		
		Estimación de duración de actividades		
		Desarrollo del cronograma		
Gestión de la Calidad del Proyecto		Planificación de la calidad		Control de la calidad
Gestión de los Riesgos del Proyecto		Identificación de riesgos		Control de Riesgos
		Análisis Cualitativo de Riesgos		
		Planificación de respuestas a los riesgos		

Tabla 1.1. Tareas del Proyecto

Los siguientes puntos describen los aspectos más importantes de los procesos a seguir para la gestión del presente trabajo de tesis con un enfoque práctico enmarcado desde las áreas de conocimiento del PMBOK.

A. **Gestión de la integración:**

La gestión de la integración se encarga de la coordinación de la superposición e interacción de los procesos separados que existen a pesar de la definición de interfaces, tales como definición de costos o riesgos.

La principal tarea dentro de la Gestión de la Integración es el desarrollo del Plan de Proyecto, asegurando la correcta ejecución de éste. Coordina los procesos de planificación, implementación y control de todas las áreas de conocimiento dentro del proyecto.

La gestión de la integración abarca distintos puntos los cuales serán tratados en el presente proyecto de la siguiente manera:

- Diagrama de estructura del proyecto: El proyecto será dividido en distintas fases definiendo de forma clara y precisa cada una de las partes del proyecto. El detalle de la planificación del proyecto será detallado más adelante.
- Dirección y gestión de la ejecución del proyecto: El desarrollo se basa en el plan de tesis y se llevará a cabo de acuerdo a la planificación. Deberá seguirse las normas definidas de acuerdo a la metodología elegida para el desarrollo, la cuál será descrita en el siguiente capítulo.
- Control del proyecto: El control se llevará a cabo durante las reuniones hechas con el asesor de tesis, en las cuales se comprobará el correcto avance del proyecto y se tomarán las medidas correctivas necesarias para el ajuste del proyecto si es que fuese necesario

- Control integral de cambios: El control se llevará a cabo durante la ejecución debido a que pueden aparecer imprevistos que impliquen cambios en el proyecto. Estos cambios deben ser controlados para medir el impacto sobre el proyecto.

B. Gestión del alcance:

La gestión de alcance asegura que el proyecto esté claramente definido y estructurado. También se encarga de asegurar que los resultados obtenidos correspondan a los requerimientos.

La gestión del alcance abarca distintos puntos los cuales serán tratados en el proyecto de la siguiente manera:

- Planificación del alcance: En el plan de tesis se define el alcance del proyecto el cual será usado como un documento que administre la planificación del alcance. Se verifica que todo lo que abarque el proyecto esté incluido en la planificación.
- Definición del alcance: Se indicará los límites del proyecto, los cuales permitirán llevar un control adecuado sobre el tiempo empleado y los resultados esperados.
- Generación de la Estructura de Descomposición del Trabajo (WBS): Se definirán todos los productos que se requieran para el desarrollo del presente proyecto, con el fin de organizar y establecer el alcance total del mismo. En el punto de Planificación, que será detallado más adelante, se muestra cómo quedan agrupados estos productos en cada una de las fases del proyecto mediante WBS y se verifica que cumplan lo indicado en el alcance.
- Verificación del alcance: El asesor de tesis verificará el avance del proyecto, según lo indicado en la planificación, si es que cumple con los alcances definidos.

- Control del alcance: Se verificará que el alcance sea el mismo que ha sido definido en Definición del alcance y en caso exista algún cambio se deberá actualizar la documentación correspondiente, verificar el impacto ocasionado al proyecto y actualizar el calendario del proyecto.

C. Gestión del tiempo:

La gestión del tiempo permite llevar un control adecuado del cumplimiento de las fechas. La falta de cuidado o errores en la programación del desarrollo, conduce al no cumplimiento de los plazos del proyecto y a retrasos.

La gestión del tiempo abarca distintos puntos los cuales serán tratados en el proyecto de la siguiente manera:

- Definición de actividades: En este punto se definirán todas las actividades a realizar para el cumplimiento de los objetivos del proyecto, asignándoles una duración de acuerdo a su complejidad. Se debe tener en cuenta que a lo largo del proyecto pueden aparecer nuevas tareas que no fueron consideradas, sin embargo, son fundamentales para el desarrollo del proyecto y por ello deben ser agregadas.
- Secuencia de actividades: En este punto se establece la secuencia de las tareas, teniendo en cuenta las relaciones de dependencia entre ellas, como es el caso de las tareas predecesoras, o pueden ser realizadas en paralelo. Tal como se menciona en el punto anterior, existe la posibilidad que aparezcan nuevas actividades lo que ocasionaría que pueda haber un cambio en la secuencia y dependencia de las mismas.
- Estimación de recursos de actividades: Para la realización de las tareas se requieren recursos como herramientas para el análisis, software para el desarrollo, etc. Los recursos utilizados son estimados puesto que siempre cabe la posibilidad de cambios en las tareas.

- Estimación de la duración de cada actividad: La duración de las actividades se muestran en el punto de Planificación (1.3.2.), la duración es estimada puesto que siempre cabe la posibilidad de existir cambios en las tareas.
- Desarrollo del cronograma: Se llevarán a cabo las tareas según lo establecido en el cronograma, el cual se basa en la planificación del proyecto.
- Control del cronograma: El control de las actividades se realizará en las reuniones con el asesor de tesis y se ajustará la planificación y las actividades en caso haya algún cambio en el desarrollo del proyecto.

D. **Gestión de la calidad:**

La gestión de la calidad identifica los estándares de calidad relevantes del proyecto. En este punto se controla permanentemente el cumplimiento de los requerimientos de calidad basándose en la evaluación de los resultados obtenidos.

La gestión de la calidad abarca distintos puntos los cuales serán tratados en el proyecto de la siguiente manera:

- Planificación de la calidad: Se desarrollará un plan de pruebas que verifique la calidad del proyecto. El plan de pruebas con el que se evaluará el proyecto abarca el cumplimiento de objetivos del proyecto.
- Control de la calidad: Se garantizará la calidad del proyecto efectuando las pruebas que se definirán durante la Planificación de la calidad. Estas pruebas cumplirán con lo señalado en cada una de las partes que compongan el plan de pruebas y se guardarán los resultados para una posterior revisión tal como se indica en el punto siguiente. El control de la calidad del proyecto se realizará a través de las pruebas mencionadas y los resultados serán evaluados continuamente en las reuniones con el asesor de tesis. Se mantendrá un registro de los controles efectuados el cual será evaluado en conjunto con el asesor de tesis para de esta manera tener un control claro y garantizar que las pruebas se efectúen de manera

satisfactoria y cumplan con su objetivo de medir la calidad del proyecto de fin de carrera.

E. Gestión del riesgo:

La gestión de riesgos asegura que se identifiquen y analicen los riesgos del proyecto y que se tomen medidas necesarias para reducirlos. Durante el desarrollo del proyecto se controlan la probabilidad de suceso y el impacto de los riesgos individuales, así como también la efectividad de las medidas.

La gestión del riesgo abarca distintos puntos los cuales serán tratados en el proyecto de la siguiente manera:

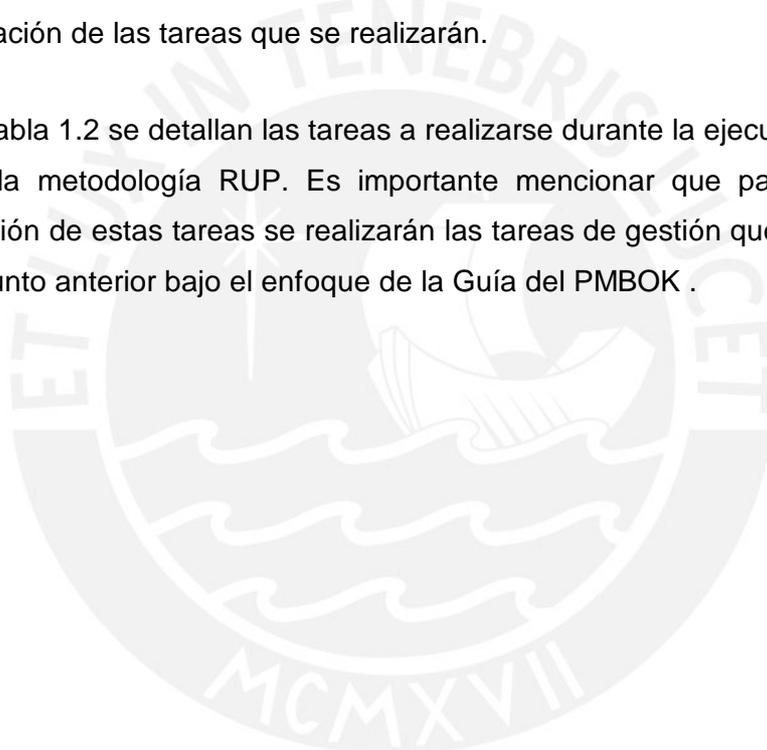
- **Identificación de riesgos:** Los cambios que impacten gravemente en el desarrollo del proyecto son los principales riesgos que afectan al proyecto. Estos serán identificados durante la planificación.
- **Análisis cualitativo de riesgos:** Se debe analizar la posibilidad de demora en la curva de aprendizaje de las herramientas y la tecnología a utilizar, así como la necesidad de investigar nuevos temas y desarrollar soluciones a problemas que vayan surgiendo que demanden más tiempo del previsto.
- **Planificación de respuestas a los riesgos:** Se tienen que realizar ajustes en los tiempos planificados a causa de retrasos que puedan presentarse y elaborar un plan de contingencia para poder recuperar el tiempo que se pueda perder.
- **Control de riesgos:** Esto se efectuará en las reuniones realizadas con el asesor de tesis comprobando que los tiempos definidos en la planificación no se vean muy afectados.

1.3.2. Planificación

Para realizar la planificación de las tareas y actividades involucradas en el desarrollo del presente proyecto de tesis es necesario considerar que éste involucra la realización de un Sistema de Información. Por ello, se utilizará metodología RUP para tener un mejor control de todas las tareas, actividades, técnicas y productos asociados al ciclo de vida del Software.

La metodología empleada será explicada a detalle en el Capítulo 2; sin embargo, es necesario mencionarlo en este punto pues de las fases que propone la metodología, y que serán utilizadas en le presente proyecto, depende la planificación de las tareas que se realizarán.

En la Tabla 1.2 se detallan las tareas a realizarse durante la ejecución del proyecto según la metodología RUP. Es importante mencionar que paralelamente a la realización de estas tareas se realizarán las tareas de gestión que se mencionaron en el punto anterior bajo el enfoque de la Guía del PMBOK .



	Concepción	Elaboración	Construcción	Transición
Modelamiento de Negocio	Definición del producto(*) Trabajo de investigación Documentación de resultados			
Requerimientos	Elaboración de: Catalogo de requisitos, Documento de visión, Analizar la viabilidad de los requerimientos y Especificación de Requisitos de Software			
Análisis y Diseño	Elaboración de Documento de análisis, diagramas de diseño.	Elaboración de Documentación de Diseño Final, Modelado de la Base de Datos, Documento de Arquitectura Final y Prototipo de Interfaz Gráfica		
Implementación			Implementación de los módulos del sistema de información	
Pruebas		Elaboración de Plan de Pruebas de Integración y Unitarias	Revisión de Plan de Pruebas de Integración y Unitarias	
Configuración y administración del cambio		Elaboración de: Documento de configuración de cambios.	Elaboración de: Documento de configuración de cambios.	

Tabla 1.2. Tareas a realizarse durante el desarrollo

A. **Descomposición de Trabajo**

A continuación se detallan las tareas a realizar durante el proyecto, según la metodología RUP. Para fines de este proyecto se ha elegido algunos artefactos de acuerdo a cada fase, los cuales permitirán documentar y desarrollar el proyecto de forma organizada.

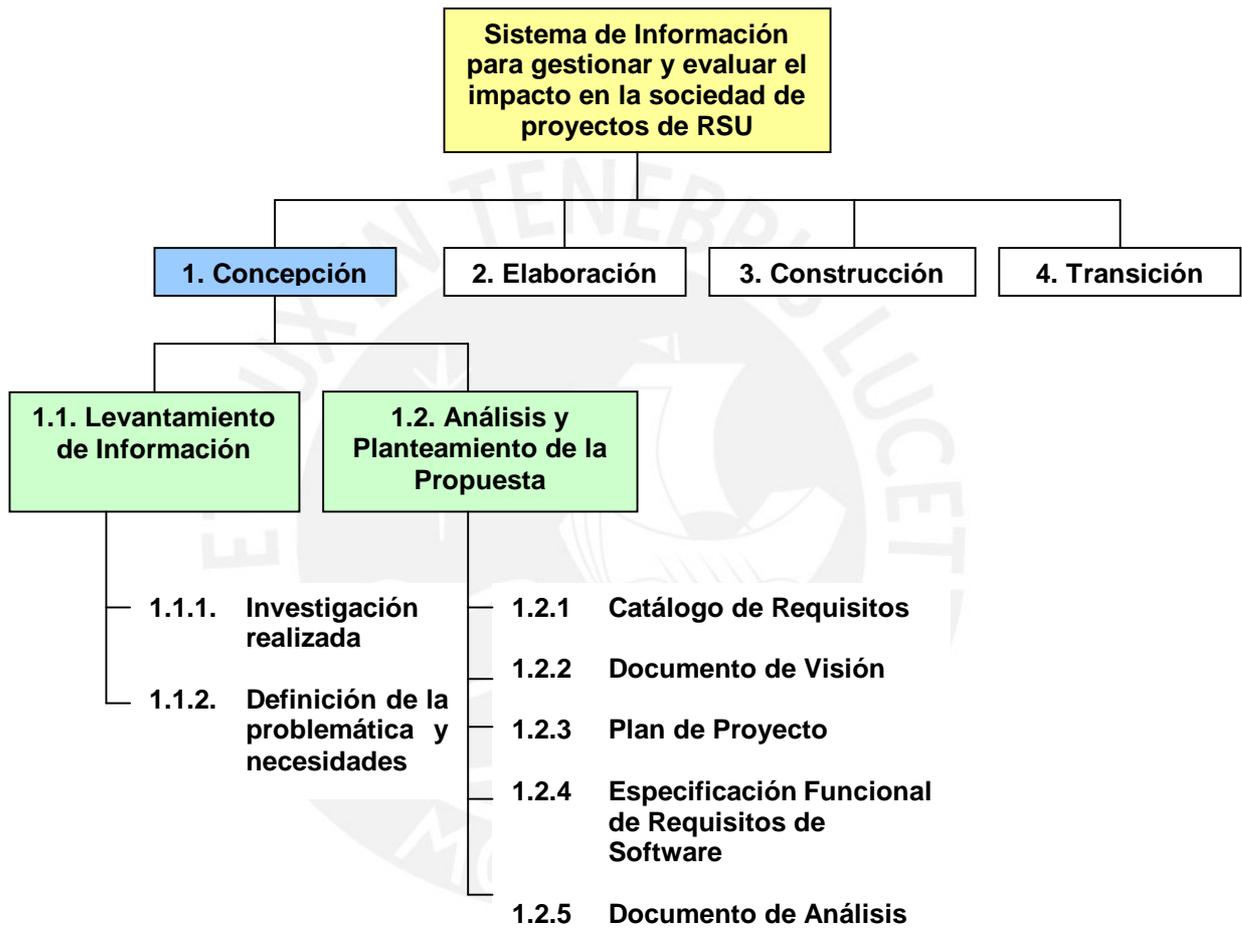


Figura 1.5. Tareas de la Etapa de Concepción

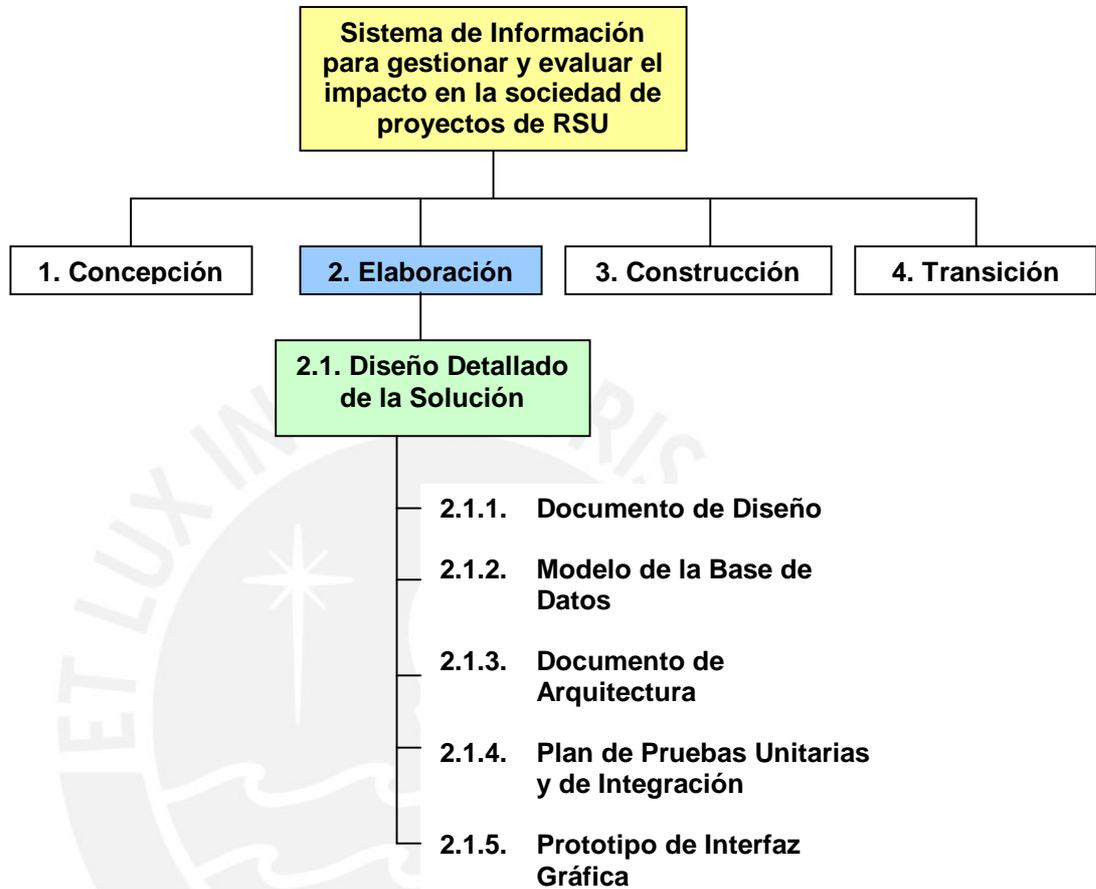


Figura 1.6. Tareas de la Etapa de Elaboración

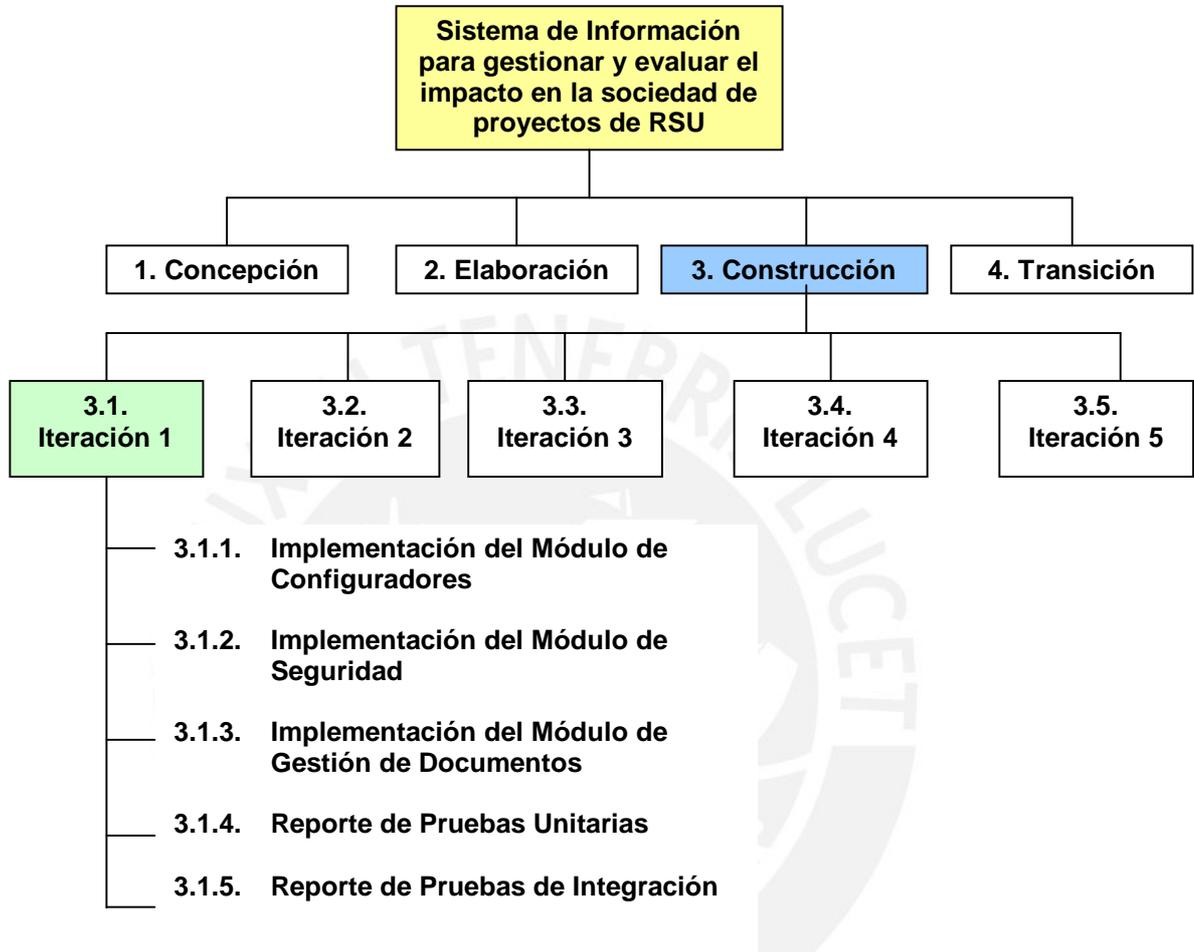


Figura 1.7.1. Tareas de la Etapa de Construcción – Iteración 1

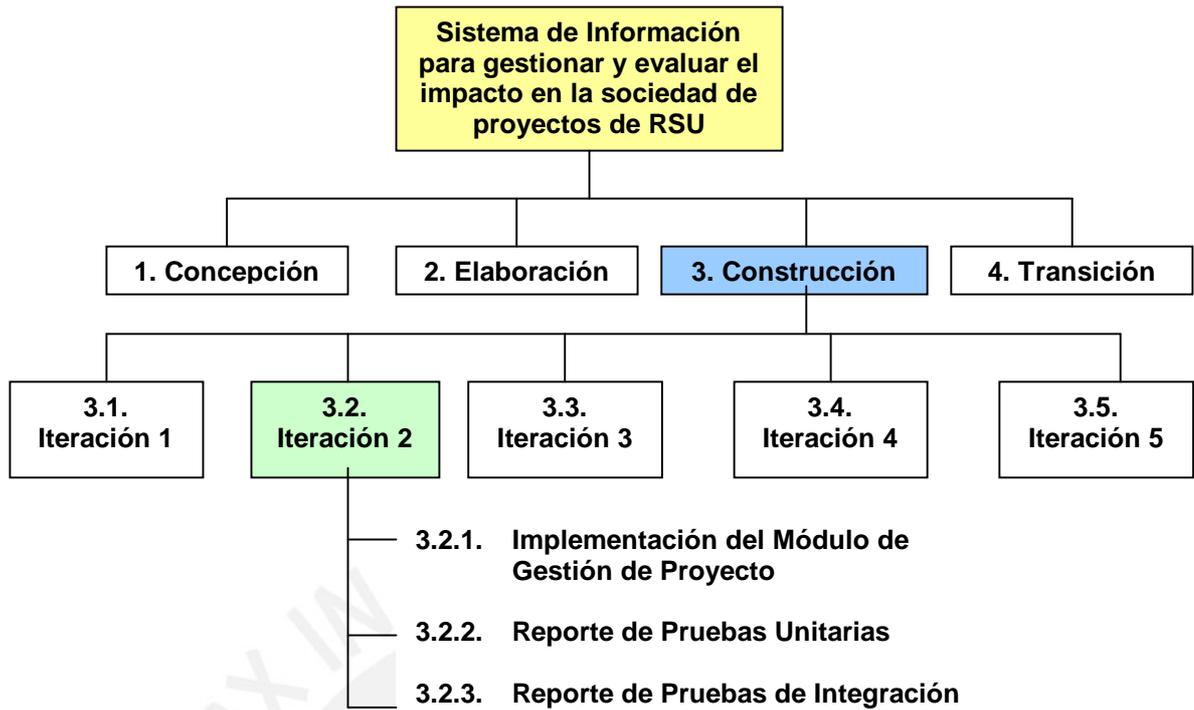


Figura 1.7.2. Tareas de la Etapa de Construcción – Iteración 2

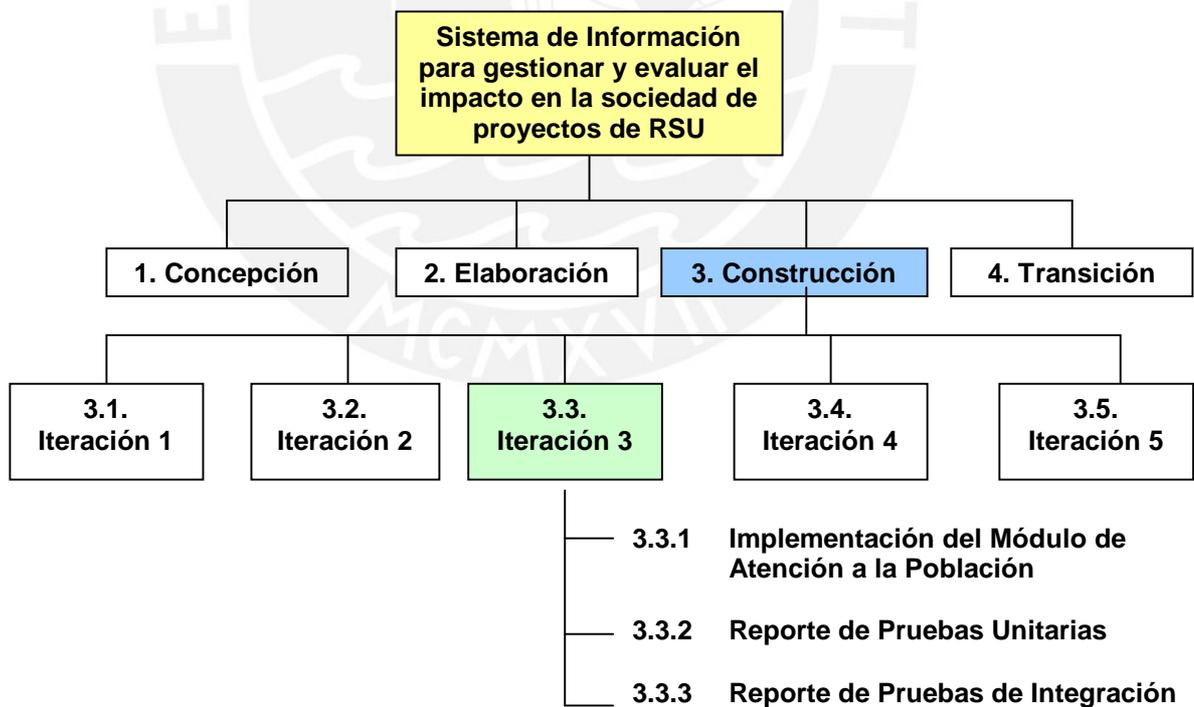


Figura 1.7.3. Tareas de la Etapa de Construcción – Iteración 3

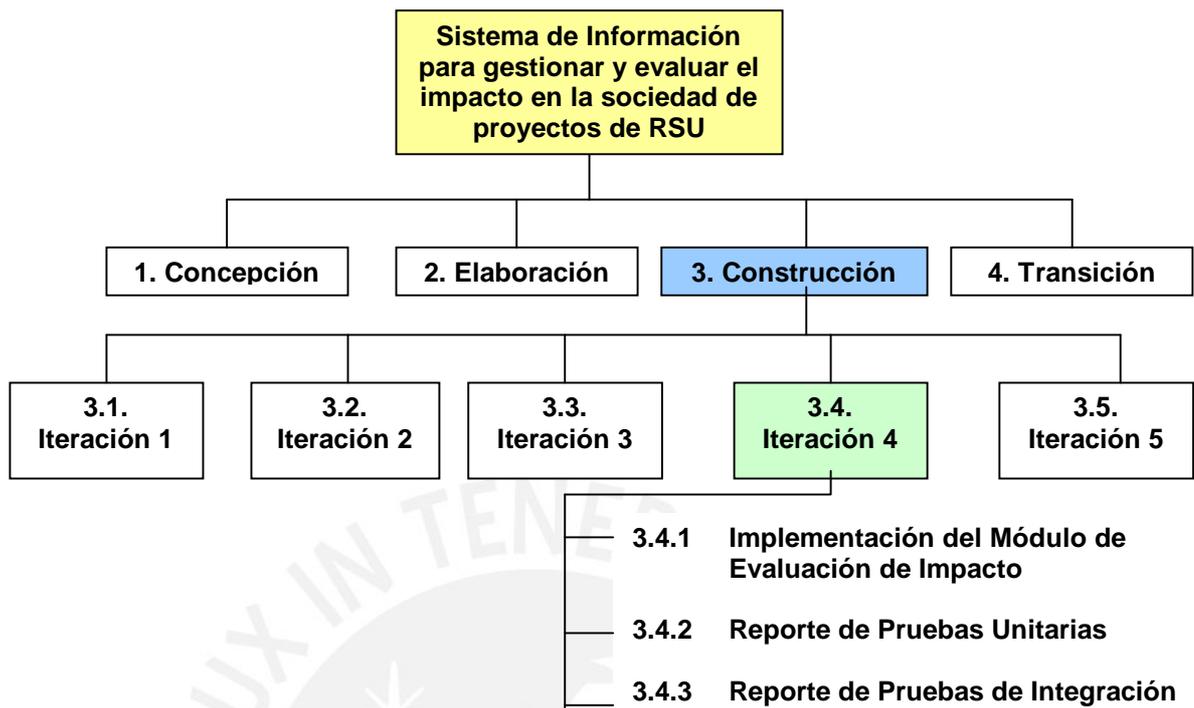


Figura 1.7.4. Tareas de la Etapa de Construcción – Iteración 4

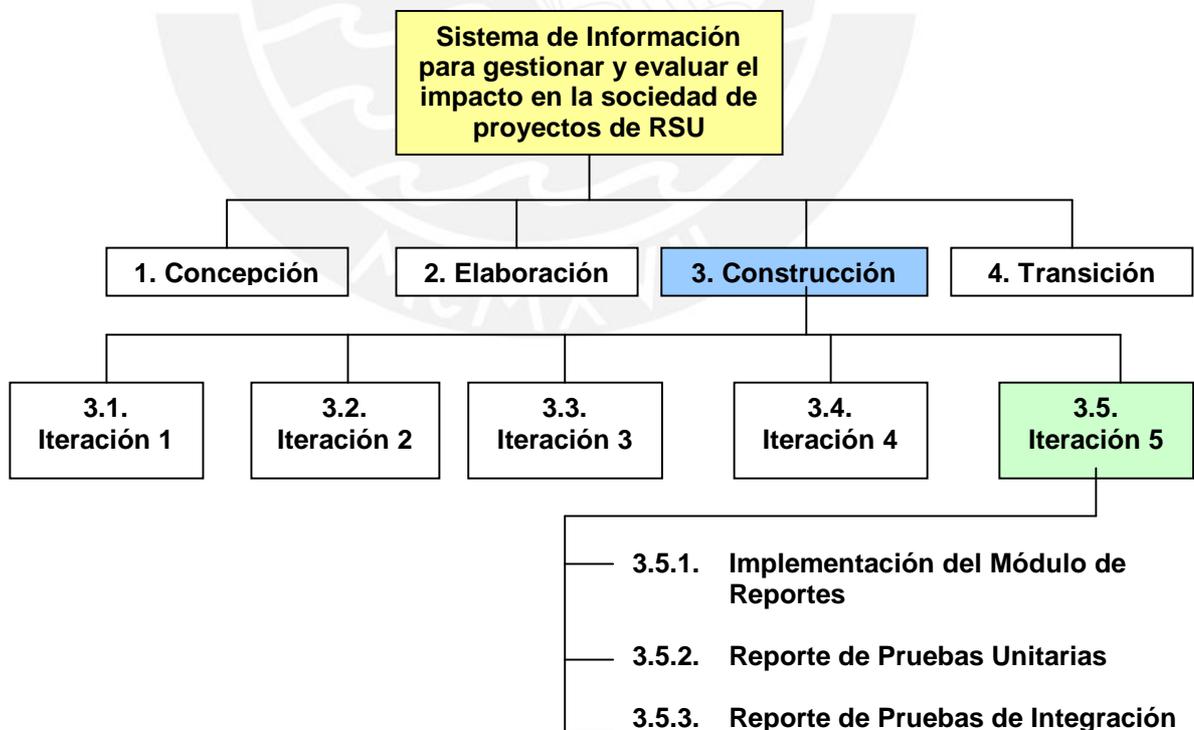


Figura 1.7.5. Tareas de la Etapa de Construcción – Iteración 5

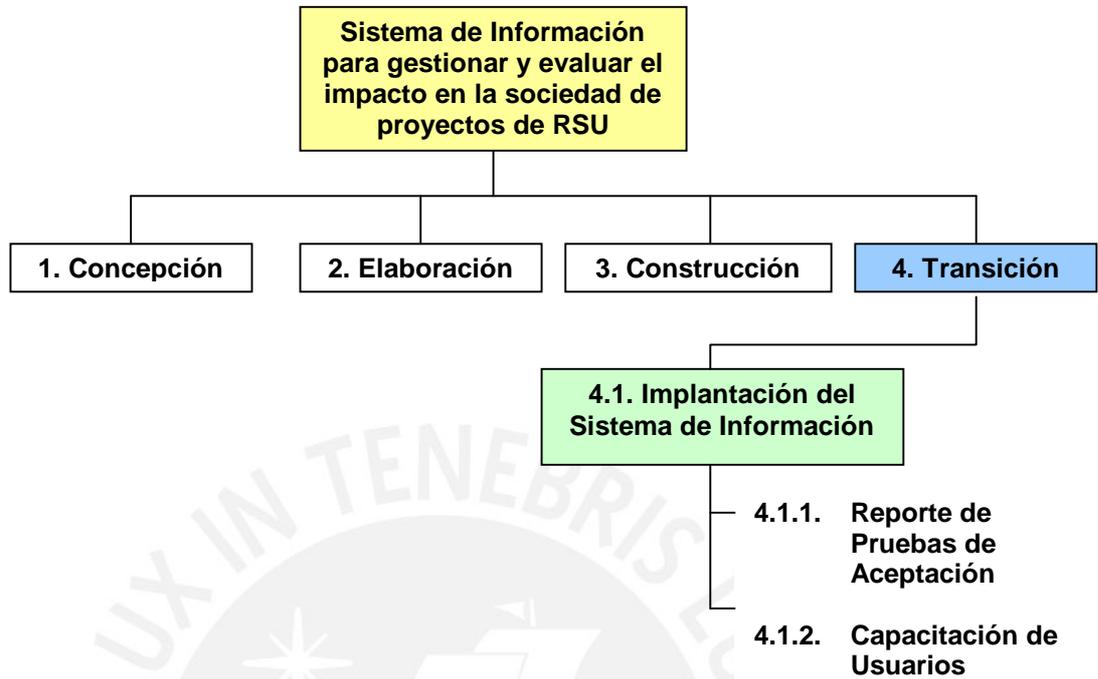
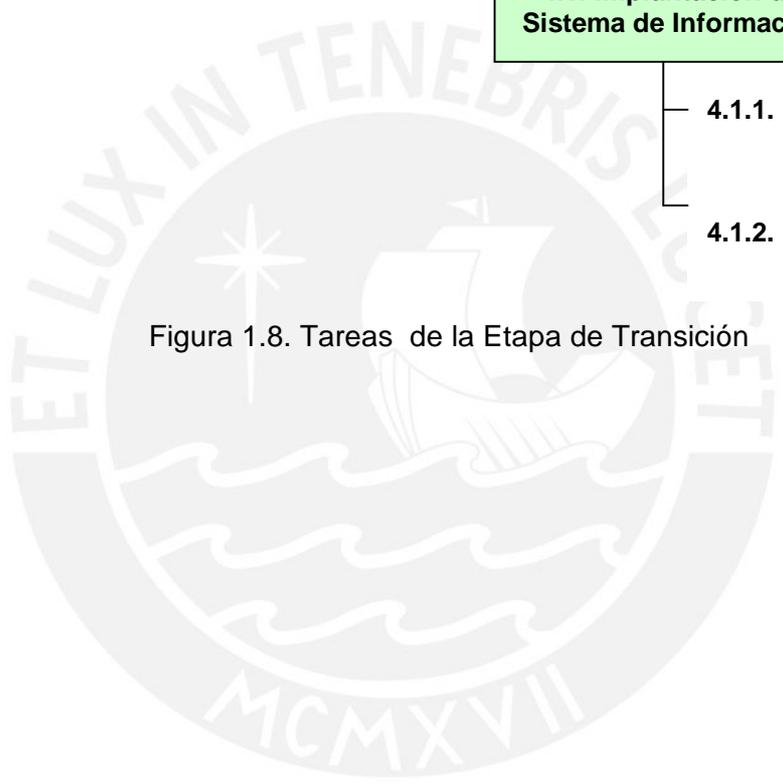


Figura 1.8. Tareas de la Etapa de Transición



B. Diagrama de Gantt

Id	Nombre de tarea	Predecesoras	Duración
1	Proyecto de Fin de Carrera		1050 horas
2	Sistema de Información para la Gestión y Evaluación del Impacto en la Sociedad de Proyectos de RSU		1050 horas
3	1. Concepción		234 horas
4	1.1. Levantamiento de información		103 horas
5	1.1.1. Investigación realizada		48 horas
6	Realización de encuestas y entrevistas - Manejo actual		40 horas
7	Comparación de proyectos de RSU	6	8 horas
8	1.1.2. Definición de la problemática y necesidades	7	55 horas
9	Realización de encuestas y entrevistas		40 horas
10	Definición de módulos	9	15 horas
11	1.2. Análisis y Planteamiento de la Propuesta	4	131 horas
12	1.2.1. Catalogo de requisitos	8	27 horas
13	Listar requerimientos funcionales por módulo		16 horas
14	Reunión para validar los requerimientos	13	6 horas
15	Priorizar requerimientos de acuerdo a necesidades	14	5 horas
16	1.2.2. Documento de visión	12	23 horas
17	Detallar el problema		8 horas
18	Planteamiento de Solución	17	15 horas
19	1.2.3. Plan de Proyecto	16	16 horas
20	Definición de iteraciones y tareas		10 horas
21	Definición de costos	20	6 horas
22	1.2.4. Especificación de Requisitos de Software	19	35 horas
23	Definición de paquetes y casos de uso		32 horas
24	Definición de requerimientos tecnológicos	23	3 horas
25	1.2.5. Documento de análisis	22	30 horas
26	Definición de las clases de análisis		18 horas
27	Elaboración del Diccionario de datos	26	8 horas
28	Realización de diagramas de estado	27	4 horas
29	2. Elaboración	3	220 horas
30	2.1. Diseño Detallado de la Solución		220 horas
31	2.1.1. Documento de Diseño		25 horas
32	Definición de clases de diseño		20 horas
33	Realización del documento de diseño	32	5 horas
34	2.1.2. Modelado de la Base de Datos	31	20 horas
35	Modelamiento de la Base de datos		15 horas
36	Creación de Base de datos	35	5 horas
37	2.1.3. Documento de Arquitectura	34	40 horas
38	Investigación sobre arquitecturas		15 horas
39	Definición de la arquitectura	38	20 horas
40	Realización del documento	39	5 horas
41	2.1.4. Plan de Pruebas de Integración y Unitarias	37	25 horas
42	Definición de pruebas unitarias		15 horas
43	Definición de pruebas de integración	42	10 horas
44	2.1.5. Prototipo de Interfaz Gráfica	41	110 horas
45	Diseño de pantallas		90 horas
46	Elaboración de mantenimiento guía	45	20 horas

Figura 1.9.1. Diagrama de Gantt – Etapas de Concepción y Elaboración

47		3. Construcción	29	572 horas
48		3.1. Iteración 1		114 horas
49		3.1.1. Implementación del Módulo de Configuradores		42 horas
50		Construcción del módulo		36 horas
51		Realización de pruebas unitarias	50	2 horas
52		Corrección de fallos	51	2 horas
53		Realización de pruebas de regresión	52	2 horas
54		3.1.2. Implementación del Módulo de Seguridad	49	46 horas
55		Construcción del módulo		40 horas
56		Realización de pruebas unitarias	55	2 horas
57		Corrección de fallos	56	2 horas
58		Realización de pruebas de regresión	57	2 horas
59		3.1.3. Implementación del Módulo de Gestión de Documentos	54	22 horas
60		Construcción del módulo		16 horas
61		Realización de pruebas unitarias	60	2 horas
62		Corrección de fallos	61	2 horas
63		Realización de pruebas de regresión	62	2 horas
64		3.1.4. Reporte de Pruebas Unitarias	59	2 horas
65		Documentación de resultados de pruebas unitarias		2 horas
66		3.1.5. Reporte de Pruebas de Integración	65	2 horas
67		Documentación de resultados de pruebas de integración		2 horas
68		3.2. Iteración 2	48	137 horas
69		3.2.1. Implementación del Módulo de Gestión de Proyecto		129 horas
70		Construcción del módulo		120 horas
71		Realización de pruebas unitarias	70	3 horas
72		Corrección de fallos	71	4 horas
73		Realización de pruebas de regresión	72	2 horas
74		3.2.2. Reporte de Pruebas Unitarias	69	4 horas
75		Documentación de resultados de pruebas unitarias		4 horas
76		3.2.3. Reporte de Pruebas de Integración	75	4 horas
77		Documentación de resultados de pruebas de integración		4 horas
78		3.3. Iteración 3	68	91 horas
79		3.3.1. Implementación del Módulo de Atenciones a la Población		84 horas
80		Construcción del módulo		75 horas
81		Realización de pruebas unitarias	80	3 horas
82		Corrección de fallos	81	4 horas
83		Realización de pruebas de regresión	82	2 horas
84		3.3.2. Reporte de Pruebas Unitarias	83	3 horas
85		Documentación de resultados de pruebas unitarias		3 horas
86		3.3.3. Reporte de Pruebas de Integración	85	4 horas
87		Documentación de resultados de pruebas de integración		4 horas

Figura 1.9.2. Diagrama de Gantt – Etapa de Construcción correspondiente a las iteraciones 1, 2 y 3

88		3.4. Iteración 4	78	156 horas
89		3.4.1. Implementación del Módulo de Evaluación de Indicadores	87	149 horas
90		Construcción del módulo		140 horas
91		Realización de pruebas unitarias	90	3 horas
92		Corrección de fallos	91	4 horas
93		Realización de pruebas de regresión	92	2 horas
94		3.4.2. Reporte de Pruebas Unitarias	93	3 horas
95		Documentación de resultados de pruebas unitarias		3 horas
96		3.4.3. Reporte de Pruebas de Integración	95	4 horas
97		Documentación de resultados de pruebas de integración		4 horas
98		3.5. Iteración 5	88	74 horas
99		3.5.1. Implementación del Módulo de Reportes		67 horas
100		Construcción del módulo		60 horas
101		Realización de pruebas unitarias	100	2 horas
102		Corrección de fallos	101	3 horas
103		Realización de pruebas de regresión	102	2 horas
104		3.5.2. Reporte de Pruebas Unitarias	99	3 horas
105		Documentación de resultados de pruebas unitarias		3 horas
106		3.5.3. Reporte de Pruebas de Integración	105	4 horas
107		Documentación de resultados de pruebas de integración		4 horas

Figura 1.9.3. Diagrama de Gantt – Etapa de Construcción correspondiente a las iteraciones 4 y 5

108		4. Transición	47	24 horas
109		4.1. Implantación del Sistema de Información		24 horas
110		4.1.1. Reporte de Pruebas de Aceptación		8 horas
111		Realización de pruebas de aceptación		5 horas
112		Documentación de las pruebas	111	3 horas
113		4.1.2. Capacitación de Usuarios	110	16 horas
114		Capacitación de usuarios		16 horas

Figura 1.9.4. Diagrama de Gantt – Etapa de Transición

1.4. Estado del Arte

Hoy en día no hay sistemas de información diseñados exclusivamente para la gestión de proyectos de RSU ni la documentación asociada a los mismos. Sin embargo, en el mercado hay aplicaciones para la gestión de todo tipo de proyectos y también herramientas para el almacenamiento y organización de la documentación.

1.4.1. Aplicaciones para la Gestión y Planificación de Proyectos

Una de las herramientas gráficas más conocidas para la planificación y gestión de proyectos es el diagrama de Gantt, el cual tiene como objetivo mostrar las diferentes fases, tareas y actividades programadas como parte de un proyecto, el tiempo que se dedican los recursos a cada tarea o actividad y el inicio y fin de una unidad mínima de trabajo en el tiempo.

Para una mejor comprensión del sistema a implementarse se verán las aplicaciones existentes en el mercado que emplean la herramienta mencionada para la gestión de proyectos, estas herramientas pueden monitorizar varios proyectos a la vez, pues permiten la creación de nuevos proyectos, basándose en las principales tareas y recursos que tienen en común. Sin embargo, una gran deficiencia que presentan estas aplicaciones es que no se puede medir la productividad y los indicadores del cumplimiento de objetivos, como se verá a continuación.

A. Gantt Project

Gantt Project es un software libre y fácil de usar que provee funcionalidades para la planificación y gestión de proyectos. Está programado en Java y se ejecuta en entornos Windows y Linux. [10]

Características del Gantt Project:

- Permite definir una jerarquía de tareas y su dependencia.
- Permite elaborar el Diagrama de Gantt con el listado de tareas y las asignaciones de recursos por tareas.

- Importa y exporta archivos MS Project.
- Genera Tablas PERT
- Genera archivos XML
- Permite generar archivos en formatos HTML y PDF

B. KPlato

Kplato es un software libre que provee funcionalidades robustas para la planificación y gestión de proyectos en KDE, el cual es un escritorio e infraestructura de desarrollo para sistemas Unix y, en particular, Linux. [11]

Características del KPlato:

- Permite elaborar el Diagrama de Gantt con el listado de tareas y las asignaciones de recursos por tareas.
- Muestra una vista contable en la que se muestra el coste previsto con la fecha final y la periodicidad configurables.
- Muestra una vista de recursos con asignaciones de tareas por recursos.
- Organiza las tareas siguiendo la estructura de descomposición de trabajo (WBS)
- Organiza los recursos siguiendo la estructura de descomposición de recursos (RBS)

Características no incluidas del KPlato:

- Los recursos son asignados a las tareas a pesar de que tengan ya tareas asociadas, lo cual puede ocasionar errores al realizar la planificación.

C. Planner

Planner es una aplicación libre para Linux que provee herramientas para el planeamiento de proyectos. [12]

Características del Planner:

- Permite definir las tareas con fechas de inicio/fin y duraciones.
- Permite definir dependencias entre tareas y subtareas
- Permite manejar el calendario de acuerdo a vacaciones y días feriados
- Permite asignar recursos a las tareas
- La planificación del proyecto se almacena en archivos XML.
- Está incluido en la mayoría de distribuciones de Linux.

D. **Microsoft Project**

Microsoft Project es la solución de Microsoft Office para la Planificación y Gestión de Proyectos. [13]

Características del Microsoft Project:

- Permite definir las tareas con dependencias, fechas de inicio/fin y duraciones.
- Permite evaluar el rendimiento, visualizar tendencias, administrar riesgos e identificar carencias.
- Puede realizar un seguimiento de la disponibilidad, la utilización y el progreso del trabajo mediante horarios.
- Permite administrar una cartera de proyectos y programas.
- Permite administrar subproyectos y las dependencias existentes entre ellos de una manera coordinada.

1.4.2. **Aplicaciones para el almacenamiento y organización de la documentación.**

En el mercado existen aplicaciones para el almacenamiento y organización de la documentación, a continuación se detallan algunos tipos.

A. **Servicios de almacenamiento de documentos**

Son aplicaciones que permiten almacenar y organizar la información de manera similar a como lo haría un disco duro, permitiendo el acceso a los documentos desde cualquier lugar a través de Internet.

Estos servicios pueden ser gratuitos o pueden tener un costo dependiendo de la capacidad de almacenamiento y los privilegios del usuario. Ejemplos de estas aplicaciones son: Box.net y el 4Shared [14]

Características de los Servicios de Almacenamiento de Documentos:

- Permiten acceder a los archivos desde cualquier computador.
- Permiten compartir archivos convirtiéndolos en públicos, esto generará un URL de donde se pueden descargar los archivos.
- La interfaz de la aplicación, en la mayoría de casos, contiene publicidad.
- Permite llevar el control sobre las descargas a través de un historial del número de descargas que se hacen a los archivos compartidos como públicos.

B. **Sistema de almacenamiento y gestión de documentos para el desarrollo de bibliotecas digitales**

Es un sistema multiplataforma en Java para almacenar y gestionar documentos de forma independiente de la base de datos y del sistema operativo, propuesto por Salvador Otón, José M. Gutiérrez, Javier Macías, Roberto Barchino (Departamento de Ciencias de la Computación, Universidad de Alcalá). [15]

Proporciona un mecanismo de acceso con seguridad, gestión de privilegios y categorización de documentos.

Características del Sistema:

- Permite almacenar y gestionar documentos.

- Permite almacenar todo tipo de documentos sin importar el formato que pueda tener.
- El sistema es ampliable mediante nuevas capas que añaden servicios a los ya disponibles.
- Permite a los usuarios otorgar y revocar permisos sobre los documentos.
- Incorpora un mecanismo de categorización de documentos similar a la organización de las carpetas de un disco.
- Incluye un sistema de control de versiones para poder seguir la evolución de los documentos en el tiempo.

1.4.3. Aplicaciones para el manejo de historial de atenciones clínicas

En el mercado existen muchas aplicaciones para el manejo de historial de atenciones, principalmente historias clínicas. El crecimiento continuo del volumen almacenado, los riesgos de pérdida y deterioro hacen que sea prioritario en la Informática médica la implementación de historias clínicas informatizadas.

A. CHAMAN

Entre las soluciones informáticas para la salud que CHAMAN proporciona, se encuentra la historia clínica electrónica cuyo módulo central está compuesto por la gestión de pacientes, episodios y actuaciones. [16]

Características de la Historia Clínica Electrónica:

- Permite el mantenimiento de los pacientes.
- Permite el registro de antecedentes del paciente.
- Permite el mantenimiento de episodios.
- Permite el mantenimiento de diagnósticos.
- Permite listar los episodios por paciente.
- Permite asociar los diagnósticos con los episodios.

1.5. Descripción y sustentación de la solución

Como proyecto de fin de carrera se desarrollará un sistema de información para la gestión de proyectos de responsabilidad social universitaria y la evaluación del impacto en la comunidad de los mismos.

El sistema de información propuesto será una herramienta de gran utilidad para los miembros del proyecto, puesto que contarán con información actualizada sobre los avances del proyecto, los registros de atenciones, encuestas y evaluaciones realizadas. Adicionalmente, el sistema permitirá realizar una eficiente planificación, organización y ejecución de los proyectos de RSU.

El sistema de información propuesto facilitará el trabajo de los participantes de los proyectos, debido a que contarán con toda la información necesaria en todo momento y lugar. Así mismo, esta solución brindará la posibilidad de poder realizar las actividades de los proyectos de RSU en diversos lugares sin necesidad de contar con equipos propios y con software especializado para este tipo de proyectos.

El sistema de información propuesto también beneficiará a la comunidad, pues se podrá monitorizar a través de indicadores el impacto del proyecto y el cumplimiento de sus objetivos, ofreciendo de esta manera un aporte real al desarrollo de su comunidad.

El sistema propuesto contendrá los siguientes módulos principales:

- Módulos de seguridad y configuración de parámetros
- Módulo de gestión de documentos.
- Módulo de programación, monitoreo y auditoría de proyectos de RSU.
- Módulo de atención a la comunidad.
- Módulo de evaluación de impacto.
- Módulo de reportes.

1.5.1. Módulos de seguridad y configuración de parámetros

El sistema brindará a los usuarios la posibilidad de crear perfiles de accesos de acuerdo al rol de los participantes y a los proyectos en los cuales se encuentren inscritos.

Asimismo, el módulo de configuración de parámetros brindará la posibilidad de tener un sistema flexible que se ajuste a los posibles cambios de los proyectos.

1.5.2. Módulo de gestión de documentos

El sistema permitirá organizar, almacenar y recuperar toda la documentación de los proyectos. De esta manera, los miembros del proyecto podrán acceder a ellos en todo momento y desde cualquier lugar.

1.5.3. Módulo de programación, monitoreo y auditoría de proyectos de RSU.

El sistema propuesto permitirá la programación de actividades según el calendario de los proyectos, a través de una agenda, en la cual se podrá registrar los datos de la actividad, personas involucradas, roles y funciones, gastos realizados, fecha y tiempo asignado. A diferencia de las herramientas de software para el apoyo a la gestión de proyectos, por ejemplo, MS Project, este módulo está integrado al sistema permitiendo considerar factores como servicios brindados, problemas atacados, cruces de recursos y zonas en las que se realizan los proyectos. De esta manera se advierte al usuario en caso se esté realizando otra actividad en la misma zona en el día y hora indicados o un recurso esté involucrado en actividades paralelas.

Asimismo, permitirá realizar el monitoreo de las actividades durante la realización de las mismas, permitiendo controlar y evaluar la asistencia y participación de los miembros del equipo, así como eventualidades que podrían ocurrir. De la misma manera, se permitirá realizar la auditoría de las actividades realizadas para evaluar el correcto desempeño de las labores establecidas y realizar las observaciones necesarias.

1.5.4. Módulo de atención a la comunidad

El sistema permitirá llevar un control de las atenciones personalizadas realizadas a los miembros de la comunidad durante las actividades a través de un historial de atenciones. Se manejarán tres tipos de historial de acuerdo al servicio brindado, los cuales serán :

- Historial de atención de Medicina General
- Historial de atención de Psicología
- Historial de atención de Odontología

1.5.5. Módulo de evaluación de impacto

El sistema propuesto permitirá el mantenimiento de indicadores asociados a los objetivos del Proyecto de RSU, los cuáles serán medidos a través de encuestas y evaluaciones realizadas a los miembros del proyecto y a la población. Dichas encuestas y evaluaciones serán registradas en el sistema.

El sistema no incluirá la realización de encuestas y evaluaciones en línea, los resultados obtenidos serán ingresados al sistema por los miembros del equipo encargados de la realización de dicha encuesta o evaluación. Esto se debe a que la gran mayoría de miembros de la comunidad podría no tener acceso a computadoras conectadas a Internet y por lo tanto las encuestas o evaluaciones que pudiesen ser realizadas por aquellos que sí tienen acceso, ya sea desde su hogar o desde su trabajo, no representaría a la mayoría y por lo tanto no sería un buen indicador para medir el impacto en la sociedad.

1.5.6. Módulo de reportes

El sistema utilizará la información almacenada para la generación de reportes de gastos, participación y desarrollo de las actividades, medición de indicadores, los cuales permitirán verificar el real aporte de dicho proyecto al desarrollo de la comunidad y al aprendizaje de los participantes, así como el cumplimiento de objetivos.

Este Sistema de Información unifica muchas de las ventajas, que independientemente ofrecen las herramientas descritas anteriormente. Provee a los miembros del proyecto la flexibilidad de poder consultar información requerida desde cualquier lugar y en cualquier momento a través de Internet, además contribuye en gran medida al cumplimiento de objetivos de todo tipo de proyectos de Responsabilidad Social, debido a que es adaptable a la realidad de los mismos.



2. Capítulo 2: Análisis

En este capítulo se detallará el análisis realizado partiendo de las necesidades encontradas. Para ello se describirá la metodología de la solución empleada y, posteriormente, se evaluará la viabilidad de la solución propuesta a través de un análisis técnico y económico

2.1. Definición de la metodología de la solución

El sistema de información, como ha sido detallado en el Capítulo 1, está dividido en módulos, por lo que el desarrollo se realizará por etapas iteradas. Cada iteración requiere un control adecuado de las actividades y tareas asociadas, con esa finalidad se ha elegido la Metodología Rational Unified Process (RUP), pues provee herramientas para la administración de los requerimientos, verificación continua de la calidad, análisis de riesgo, control de cambios a través de un desarrollo iterativo y tiene un enfoque basado en modelos por lo que utiliza lenguaje UML.

La principal ventaja de utilizar la metodología RUP es el desarrollo iterativo, pues el conocimiento y la experiencia adquiridos en cada iteración es aplicable a las iteraciones sucesivas, obteniendo de esta manera una mejora continua en el desarrollo del producto.

Al tratarse de un proyecto a largo plazo es necesario dar especial énfasis a la etapa de análisis, dado que de ello dependerá el éxito en la etapa de implementación. Para realizar un buen análisis es necesario contar con una metodología que proporcione las herramientas necesarias y un control adecuado para esta fase. La metodología RUP permite mitigar riesgos futuros pues, como se detallará en los siguientes puntos, provee una estructura que permite organizar y documentar las etapas de desarrollo del software dando énfasis en las etapas de análisis y diseño.

2.1.1. Fases del desarrollo de software según RUP

RUP divide en 4 fases el desarrollo del software, como se muestra en la figura 2.1:

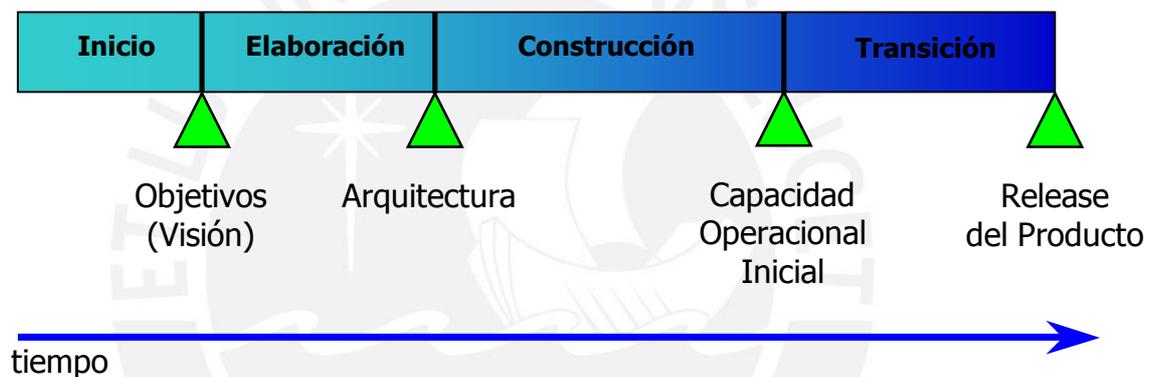


Figura 2.1. Fases de la metodología RUP[20]

- **Concepción:** En esta fase se definirá la visión, el alcance del proyecto en función al análisis de los requerimientos del sistema. Se elaborarán los siguientes artefactos: Documento de visión, Plan de proyecto, Catálogo de requisitos, Especificación de requerimientos de software y Documento de análisis
- **Elaboración:** En esta fase se definirá el diseño y la arquitectura óptima para el desarrollo del sistema. Se elaborarán los siguientes artefactos: Documento de diseño, Documento de arquitectura, Prototipo de la solución y Plan de pruebas unitarias y de integración.

- **Construcción:** En esta fase se construirá el sistema y se realizarán las pruebas necesarias para garantizar su correcto funcionamiento y el cumplimiento de los requerimientos. Se construirán las versiones 1.0, 2.0, 3.0, Beta y Final
- **Transición:** Se pondrá a disposición de los usuarios la versión final del sistema de información para la realización de las pruebas finales. Se manejarán criterios de aceptación, los cuales deberán ser aprobados por los usuarios.

2.1.2. Disciplinas bajo las cuales se desarrolla el ciclo de vida por cada iteración:

Las disciplinas bajo las cuáles se desarrolla el ciclo de vida de cada iteración están asociadas a las fases del desarrollo de software como se muestra en la figura 2.2.

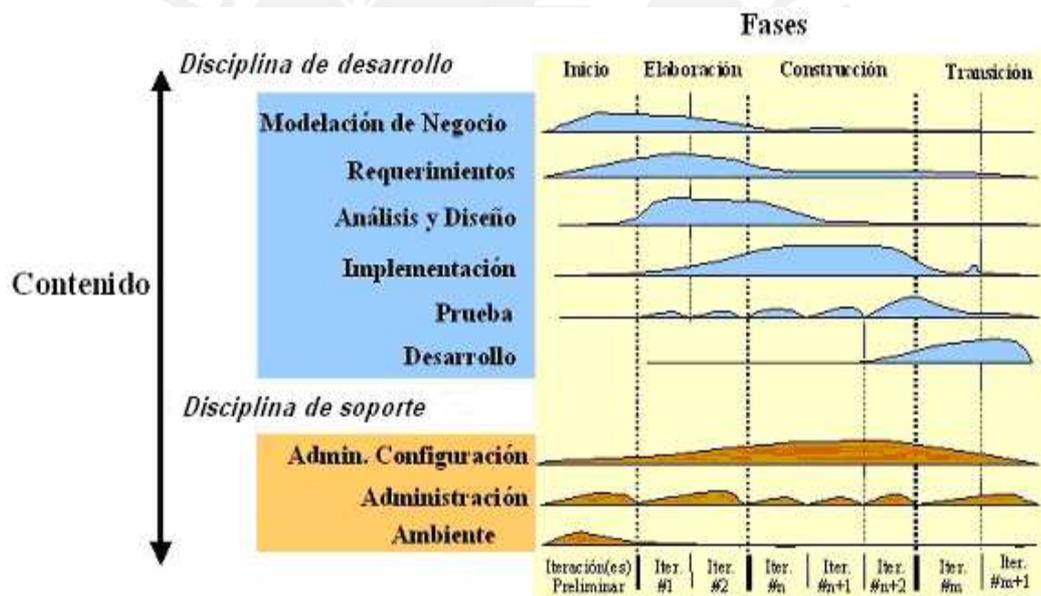


Figura 2.2. Fases e iteraciones de la metodología RUP[20]

A. Disciplina de Desarrollo

- **Modelación de Negocio:** Se realizará el levantamiento de toda la información necesaria para poder definir los requerimientos funcionales del sistema.

- **Requerimientos:** Con la información recopilada según el punto anterior se elaborará una lista de exigencias mínimas y deseables para el Sistema a implementarse, analizando la viabilidad y los costos.
- **Análisis y Diseño:** En función a los requerimientos obtenidos se diseñará la arquitectura de software y el prototipo de pantallas, los cuáles serán evaluados para analizar el cumplimiento con los requerimientos.
- **Implementación:** Se implementará el Sistema de información teniendo en cuenta cierto número de iteraciones en las cuáles se implementarán los módulos del Sistema de Información de acuerdo a un diagrama de precedencias elaborado.
- **Pruebas:** Las pruebas se realizarán para la revisión de la calidad de los módulos implementados en cada iteración.

B. Disciplina de Soporte

- **Administración de la configuración del cambio:** Para el sistema a implementarse se llevará un estricto control de cambios mediante el manejo de versiones y evaluación del impacto del cambio.
- **Administración del proyecto:** Para el sistema a implementarse se asignarán horas a cada tarea y se controlará el avance realizado.

2.1.3. Elementos de la Metodología RUP

Los elementos de la metodología RUP que serán empleados para el desarrollo del Sistema de Información son:

- **Actividades**

Son los procesos que se llegan a determinar en cada iteración. Todas las actividades a realizarse están definidas en el diagrama de Gantt, indicadas en el punto 1.3.2.2.

- **Artefactos**

Resultado parcial o final que es producido y usado durante el proyecto. Son las entradas y salidas de las actividades. [20] Un artefacto puede ser un documento, un modelo, o un elemento de modelo. Todos los artefactos que se elaborarán se encuentran definidos en el diagrama de Gantt y el WBS, indicados en el punto 1.3.2.

2.2. Identificación de los requerimientos

Luego de entrevistas y encuestas realizadas a diferentes participantes de diversos tipos de proyectos de RSU, se ha obtenido valiosa información sobre la situación real de los proyectos y se ha tratado, en lo posible, de proponer una solución general que sea la adecuada para todos los proyectos. Con este fin, la solución deberá ser lo suficientemente flexible y configurable para que se ajuste a las necesidades de cada uno de ellos.

El análisis de las necesidades y problemas encontrados durante el estudio de la situación real de los proyectos de RSU puede ser revisado en el Documento de visión (véase Anexo C). Así mismo, se ha elaborado el Catálogo de Requisitos (véase Anexo B), en el cual se listan todos los requerimientos que debe cumplir el sistema a desarrollarse. Dichos requerimientos están asociados a las necesidades de todos los actores de los diferentes proyectos de Responsabilidad Social Universitaria y se les ha asignado un nivel de prioridad, con la finalidad de priorizar todos aquellos que sean los más importantes y solucionen los problemas encontrados, para luego comenzar con aquellos requerimientos que puedan darle valor agregado al producto. Así mismo, existen requerimientos exigibles y deseables, dicha clasificación se ha realizado considerando las necesidades de aquellos que utilizarán el sistema.

2.2.1. Requerimientos Funcionales

El sistema de información a desarrollarse contará con un módulo de gestión de documentos, el cuál permitirá la creación de directorios y subdirectorios para organizar la documentación de un proyecto de Responsabilidad Social Universitaria,

de tal manera que los alumnos que participan en un proyecto de este tipo tengan a su disposición todos los documentos y puedan revisarlos desde cualquier lugar y en cualquier momento, esto soluciona el problema de la falta de información sobre el proyecto, dado que sin necesidad de consultar con otras personas, el alumno podrá revisar las actas de reunión, los permisos solicitados, los informes presentados, entre otros documentos, como si tuviese toda la información guardada en el disco duro de la computadora desde la que accede al sistema. Con los permisos necesarios, de acuerdo al perfil que tenga el usuario, éste podrá además de revisar los documentos, eliminarlos y guardar otros documentos o nuevas versiones.

Otro problema que afecta mucho el desarrollo del proyecto es la falta de conocimiento de los nuevos miembros del proyecto sobre las actividades realizadas y las que están pendientes, así como los problemas que tiene la comunidad, por ello el sistema contará con el módulo de gestión del proyecto, el cuál permitirá llevar un registro de todas las actividades del proyecto, problemas encontrados, servicios brindados, miembros del equipo y los roles de cada uno de ellos, entre otros registros importantes para controlar el correcto desarrollo del proyecto, esto permitirá a los miembros del proyecto tener el cronograma de actividades en línea con toda la información asociada que puedan requerir y conocer también el organigrama del proyecto con información detallada, lo que les permitirá saber a quién recurrir para despejar las dudas que puedan tener sobre algún punto específico.

Es importante resaltar que el módulo de gestión del proyecto, brindará también las herramientas necesarias para asegurar el correcto desarrollo de las actividades del proyecto, dado que el Jefe de proyecto podrá asignar tareas específicas a cada participante. Durante la actividad, el Jefe de proyecto o responsable de la actividad, podrá registrar la asistencia de los participantes.

Al finalizar la actividad el Jefe de proyecto y los profesores responsables podrán auditar las actividades realizadas a través del sistema haciendo observaciones sobre el cumplimiento de las tareas asignadas a los participantes, revisando el cumplimiento de los tiempos asignados, los retrasos al inicio y fin de la actividad y las observaciones anotadas durante el desarrollo de la misma. Después de terminada la auditoría podrá agregar la actividad a un grupo de acuerdo a los patrones en común que pueda encontrar con sucesos anteriores en otras actividades, lo que le permitirá al Jefe de proyecto llevar un control de los

imprevistos o sucesos, recomendaciones a futuro y ver la frecuencia con la que ocurren para poder tener planes de contingencia en las actividades sucesivas. Como se ve, este módulo solucionará problemas latentes en las actividades como su desordenado desarrollo, el incumplimiento de los fines para los que ha sido realizada, la falta de control antes, durante y después de su realización y sobre todo la falta de una fuente de retroalimentación para su continua mejora hacia el logro de los objetivos.

Como se ha mencionado en el capítulo anterior, además de talleres educativos y charlas, también se da asistencia a la población a través de campañas, en las cuales los alumnos brindan atención a la población en los servicios de psicología, odontología y medicina general, aquí surgía otro problema pues no se lleva un registro de las atenciones realizadas, por lo que no se brinda un aporte real a la comunidad, al no contar con información sobre los casos atendidos, frecuencia con la que ocurren, soluciones dadas, patrones encontrados, entre otros. Por ello, el sistema contará con el módulo de atención, el cuál permitirá registrar todas las atenciones en un Historial de atenciones que tendrá un tipo de ficha diferente dependiendo del servicio brindado.

Como en todo proyecto, es necesario definir objetivos y cumplirlos a lo largo del proyecto; sin embargo, difícilmente se podrán cumplir estos objetivos si es que todos los involucrados no apuntan hacia el mismo fin común. Es por ello, que el sistema deberá tener un módulo de evaluación, a través del cual todos los involucrados puedan conocer si es que realmente se están cumpliendo los objetivos. En este módulo, se podrán registrar indicadores para medir los objetivos y a través de encuestas y evaluaciones, que podrán ser creados con un asistente dentro del sistema, se podrá verificar si es que realmente se están cumpliendo los objetivos y en qué medida. Adicionalmente, se podrá registrar acciones correctivas para poder alcanzar los objetivos propuestos.

2.2.2. Requerimientos no funcionales

Debido a que las actividades se llevan a cabo en lugares cercanos o en los mismos asentamientos humanos, parroquias, centros comunales, colegios públicos, entre otros; el Sistema será una aplicación Web, pues no siempre se manejan los mismos equipos, además el proyecto no cuenta con equipos propios, por lo que se piensa

tener durante la actividad una cabina de Internet a disposición del proyecto, desde la cual puedan registrar las atenciones y observaciones necesarias. Además, los miembros del proyecto no tienen horario fijo de dedicación al proyecto, por lo que sería útil para ellos poder actualizar información sobre el proyecto o consultarla desde sus universidades o sus casas. El sistema deberá ser restringido, sólo deberán poder acceder al sistema aquellos que cuenten con un usuario y una contraseña. Como en todos los proyectos, los miembros cumplen funciones específicas y diferentes, por ello se les asignará un perfil, de manera que solo puedan realizar acciones que le correspondan de acuerdo a los privilegios asociados a su perfil. Adicionalmente, el sistema podrá manejar varios proyectos, por lo que es necesario asegurarse de que los usuarios solo puedan visualizar la información de los proyectos en los cuales participan.

El manejo del módulo de documentación será muy sencillo debido a que su funcionalidad es muy similar a la que se acostumbra ver cuando se guarda documentos en directorios en la computadora, se consultan las propiedades de los documentos, las propiedades de los directorios y el espacio libre que se tiene para seguir guardando documentos, es así como trabajará este módulo.

El sistema será intuitivo y sólo requiere tener conocimientos básicos de Internet, además, el sistema brindará al usuario alertas, que no le impedirán seguir con su labor, pero que serán útiles en caso se confunda o no tome en cuenta algunas condiciones como cruces de fechas, cruces en las labores de los participantes, no considerar algún indicador, entre otros; que finalmente se dejarán a criterio del usuario, pues pueden darse esos casos en el proyecto.

2.3. Análisis de la Solución

En esta fase se han identificado las necesidades de los usuarios y en función a los requerimientos y otros factores se propone una solución adecuada cuya implementación conduzca a la solución del problema.

2.3.1. Necesidades del usuario

Luego de entrevistas y encuestas realizadas a diferentes participantes de diversos tipos de proyectos de RSU, podemos reconocer las principales necesidades:

- Disponibilidad de toda la documentación del proyecto actualizada.
- Acceso a la información sobre actividades realizadas.
- Llevar un control adecuado de las actividades a realizarse según el cronograma.
- Registrar las atenciones realizadas.
- Evaluar el avance el proyecto.
- Disponibilidad del sistema las 24 horas.

Dichas necesidades se encuentran más detalladas en el Documento de Visión (véase Anexo C).

Como se ha visto en el punto anterior, estas necesidades están cubiertas en su totalidad por los requerimientos del sistema, pues el sistema contará con cinco módulos principales para cubrir todas las necesidades y poder solucionar los problemas encontrados.

2.3.2. Evaluación de viabilidad

Para el análisis de la viabilidad del sistema se ha considerado varios factores importantes tanto del lado del usuario, como del lado del desarrollo del sistema.

Actualmente, el equipo de trabajo no cuenta con computadoras destinadas al proyecto y debido a que éste no tiene fines de lucro, se sostiene a través de entidades que donan los materiales o a través de las respectivas direcciones de responsabilidad social de sus universidades, además los lugares donde se realizan las actividades no son fijos.

Considerando esto, se ha optado por alquilar una cabina de Internet cercana al lugar donde se lleve a cabo la actividad, lo cual sería lo más factible para el proyecto, es por ello que el sistema a desarrollarse será una aplicación Web.

Otro factor importante que se debe considerar es el factor económico en el desarrollo del sistema, por lo que las herramientas a emplearse son de código abierto, pues el proyecto no cuenta con suficiente dinero para obtener licencias para desarrollar el sistema.

2.3.3. Consideraciones sobre el Sistema

Los requerimientos del sistema se han elaborado considerando las necesidades del usuario y también el factor económico, pues no se realizará una fuerte inversión en el sistema. Por ello se busca que éste sea útil y que el alcance cubra las necesidades más urgentes de un proyecto de Responsabilidad Social Universitaria, por lo que detalles como la interacción con el sistema y el diseño gráfico del mismo no son tan importantes como la funcionalidad que éste tenga y no son parte de los requerimientos del sistema.

Teniendo en cuenta esto, el desarrollo del sistema se centrará en su funcionalidad para cumplir con el alcance en las fechas establecidas, éste no deberá ser mayor a los 5 meses, pero sí deberá cumplir con los siguientes atributos de calidad:

- **Performance:** El tiempo de acceso al sistema no deberá ser mayor a 5 segundos.
- **Disponibilidad:** El sistema estará disponible las 24 horas del día, los 7 días de la semana.
- **Utilidad:** El sistema será apropiado para el uso de los distintos perfiles, de manera que sirvan como soporte de las labores de todas personas que interactúen con él.
- **Usabilidad:** El sistema será fácil de usar, se le proporcionará al usuario mensajes de error y de advertencia.
- **Mantenibilidad:** El sistema estará diseñado para un fácil mantenimiento. Las funciones internas contarán con la documentación necesaria para realizar los cambios oportunos sin mayores complicaciones.

- **Escalabilidad:** El sistema estará diseñado de tal manera que se puedan agregar nuevas funcionalidades sin que esto afecte de manera crítica el código.
- **Seguridad:** El acceso al sistema estará restringido a usuarios que cuenten con una contraseña.
- **Validación de la información:** La información registrada en el sistema será validada para evitar el ingreso de datos erróneos.

El sistema será desarrollado en cinco iteraciones, por lo que se presentará una versión al finalizar cada iteración, como hay cierta independencia entre los módulos, el sistema puede operar desde la versión 1.0. Los requerimientos del sistema están clasificados en exigibles y deseables, por lo tanto las versiones entregadas cumplirán por lo menos con todos los requerimientos exigibles. El detalle del contenido de las versiones se encuentra especificado en el Plan de proyecto (véase Anexo D).

A continuación se detallan los actores involucrados en todo tipo de proyectos de RSU que interactuarán directamente con el sistema a desarrollarse, considerando sus necesidades y sus funciones dentro del proyecto se definirán los perfiles que serán creados para asignar los privilegios y accesos en el manejo del sistema.

- **Usuario:** Persona que ingresa al sistema a través de un login y un password.
- **Jefe de proyecto:** Persona encargada de la administración de los miembros del equipo y de la gestión del proyecto.
- **Administrador de sistema:** Persona encargada de la creación de proyectos, perfiles y usuarios.
- **Administrador de documentos:** Persona encargada del manejo de la documentación, creación de directorios y organización de los documentos.

- **Administrador de actividades:** Persona encargada del manejo del cronograma de actividades y la administración de las tareas asociadas.
- **Miembro del equipo:** Toda persona que participa en el proyecto en las diferentes actividades que se realizan.
- **Comité de evaluación o Auditores:** Personas encargadas de evaluar el proyecto y analizar el cumplimiento de los objetivos que se plantearon al inicio del mismo.
- **Profesor :** Persona encargada de supervisar la labor de los miembros del equipo.
- **Auspiciador:** Persona que brinda ayuda económica al proyecto y que requiere información de las actividades que se realizan.

2.3.4. Análisis técnico y económico

Para la elección de las herramientas a utilizarse para la implementación del sistema se han considerado los siguientes criterios:

- Curva de aprendizaje: Se requiere herramientas cuyo aprendizaje sea sencillo, para evitar retrasos en los tiempos establecidos.
- Lenguaje multiplataforma
- La aplicación gráfica debe ser independiente del navegador
- Herramientas libres, de fácil acceso
- Base de datos rápida en la lectura

Adicionalmente, se debe considerar que no se cuenta con computadoras muy potentes para la ejecución, por lo que las herramientas deben ser ligeras.

La seguridad no es un factor crítico para el desarrollo del sistema, pues no se maneja información confidencial y las alteraciones que se puedan realizar no ponen en riesgo el éxito del proyecto de RSU.

Considerando todos estos factores, se ha optado por la utilización del lenguaje de programación PHP, cuyas siglas significan "PHP Hypertext Pre-processor" (inicialmente PHP Tools, o, Personal Home Page Tools) y el manejador de base de datos MySQL.

PHP presenta las siguientes ventajas:

- Es un lenguaje multiplataforma sencillo de aprender.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- Es una alternativa de fácil acceso pues es software libre.
- Permite las técnicas de Programación Orientada a Objetos.
- No requiere ser ejecutado en una máquina muy potente, un 486 con 16 Mb de RAM y con 500Mb de disco puede ser suficiente.

Una de las desventajas de PHP es que trabaja del lado del servidor, por lo tanto, a medida que aumentan las solicitudes es más ineficiente; sin embargo, hay que tener en cuenta que los participantes de algún proyecto no son más de 70 personas y por lo general en las actividades los que interactuarán con el sistema serán 10 personas en promedio, pues existen múltiples tareas operativas que se deben realizar y solo se tendrá disponibles aproximadamente 10 equipos. Por lo explicado anteriormente, no hay riesgo de que el sistema se vuelva ineficiente, a pesar de manejar varios proyectos.

MySQL es un manejador de base de datos libre muy veloz en la realización de operaciones y de gran rendimiento, su bajo consumo lo hace ideal para equipos de escasos recursos.

Los requerimientos de instalación y configuración para poder trabajar con las herramientas elegidas se explicarán a detalle en el Capítulo 4: Construcción.

Como el software que se utilizará es de código abierto, no se incurrirá en gastos por licencias de software; sin embargo, se requiere un servidor de aplicaciones, un servidor de base de datos y un dominio, pues para el caso de un proyecto de RSU no se necesitan correos electrónicos propios.

A continuación se presenta el detalle del presupuesto estimado:

<u>Presupuesto</u>		
Costo por Hora: S/. 10		
<u>Trabajo</u>		
Fases	Horas	Costos (S/.)
Concepción	250	2500
Elaboración	280	2800
Construcción - Iteración 1	100	1000
Construcción - Iteración 2	90	900
Construcción - Iteración 3	80	800
Construcción - Iteración 4	90	900
Construcción - Iteración 5	70	700
Total	960	9600
TOTAL TRABAJO:		S/.9 600,00
<u>Otros</u>		
Servicios de mantenimiento y soporte(*):		400
Dominio y servidores de aplicación y base de datos(*)		200
TOTAL OTROS:		S/.600.00
TOTAL:		S/.10 200

Tabla 2.1. Presupuesto para el desarrollo del sistema

(*)Pagos que deberán ser efectuados anualmente.

2.3.5. Definición del sistema

En este punto se explicará la definición del sistema, esto servirá como base para su diseño, pues permitirá modelar y definir una arquitectura que soporte todos los requerimientos funcionales del sistema.

El sistema cuenta con cinco módulos principales que son:

- Módulo de gestión de documentos.
- Módulo de programación, monitoreo y auditoría de proyectos de RSU.
- Módulo de atención a la comunidad.
- Módulo de evaluación de impacto.
- Módulo de reportes.

Se ha optado por dividir el sistema en estos módulos debido a que realizan funcionalidades independientes entre ellos, a excepción del módulo de reportes. Como los módulos trabajan de forma separada, cada módulo se implementará en una iteración diferente, como se detalla en el Plan de proyecto (véase Anexo D).

Adicionalmente, el sistema cuenta con un módulo de soporte que es el módulo de seguridad, el cual permitirá la creación de usuarios, perfiles y proyectos. De este módulo dependen todos los demás módulos, pues los demás necesitan de la creación de proyectos, debido a que toda funcionalidad brindada por el sistema está asociada a un proyecto específico. Así mismo, el módulo de reportes depende de la información registrada en los otros módulos para emitir los reportes.

En la figura 2.3. se muestran los paquetes con los que contará el sistema.

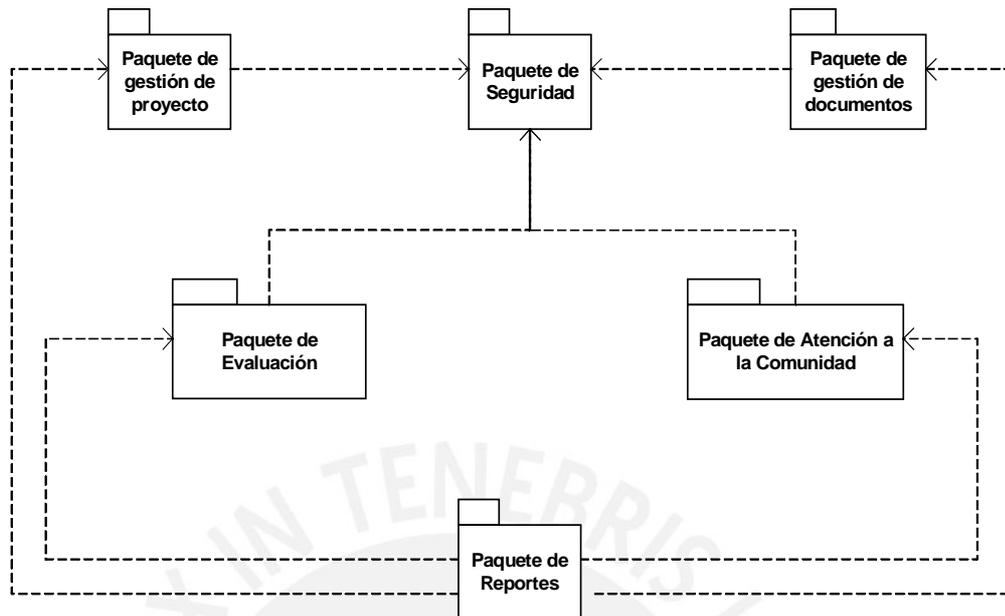


Figura 2.3. Paquetes del Sistema

A continuación se detallan cada uno de los paquetes:

- **Paquete de seguridad**

Este paquete contiene todos los casos de uso que corresponden a la administración de perfiles y usuarios. Aquí se incluyen casos de usos que permitan el mantenimiento de perfiles y mantenimiento de usuarios, así como el cambio de contraseña para ingresar al sistema.

- **Paquete de gestión de documentos**

Este paquete contiene todos los casos de uso que corresponden a la administración y organización de los documentos. Aquí se incluyen casos de uso que permitan crear y modificar directorios y subdirectorios, guardar y recuperar documentos.

- **Paquete Gestión del Proyecto**

Este paquete contiene todos los casos de uso que corresponden a la administración de actividades que se realizan en el proyecto (participantes,

tareas, roles, fecha, duración, zonas) y miembros del proyecto. Aquí se incluyen casos de uso que permitan registrar miembros del equipo, roles, actividades, auspiciadores; así como organizar las actividades y enviar recordatorios.

Dentro de este paquete se encuentra el mantenimiento de proyectos, que será accedido solo por los usuarios con el perfil adecuado para crear y modificar proyectos.

- **Paquete de Atención a la Comunidad**

Este paquete contiene todos los casos de uso que corresponden al registro del historial de atención de los miembros de la comunidad. Aquí se incluyen casos de uso que permitan almacenar y consultar información útil sobre las atenciones realizadas a los miembros de la comunidad.

- **Paquete de Evaluación del Proyecto**

Este paquete contiene todos los casos de uso que corresponden a la evaluación del proyecto. Aquí se incluyen casos de uso que permitan registrar objetivos, indicadores y asociarlos, así mismo se debe permitir la creación de encuestas y evaluaciones, cuyos resultados servirán para aplicar acciones correctivas.

- **Paquete de Reportes**

Este paquete contiene todos los casos de uso que corresponden a la elaboración de reportes solicitados.

La descripción y especificación de los casos de uso se encuentran detallados en la Especificación de Requisitos de Software (véase Anexo E) y en el Documento de análisis se encuentran los diagramas de clases de cada paquete y el diccionario de datos con la descripción de todas las clases (véase Anexo F).

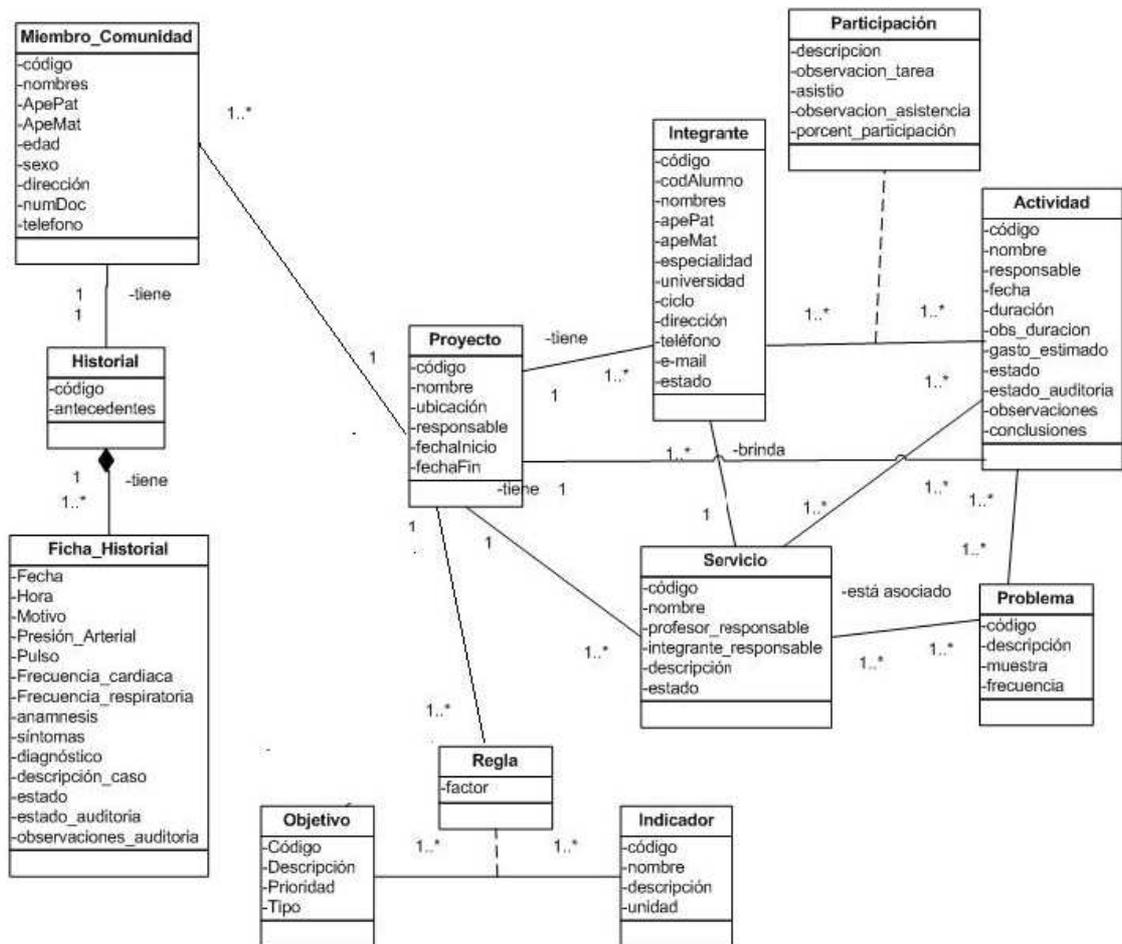


Figura 2.4. Diagrama de clases

Las clases mostradas en la figura 2.4. permitirán modelar los principales procesos que se llevan a cabo y cumplen con los requerimientos exigibles del sistema. Los principales requerimientos giran en torno al proyecto, los miembros del equipo, las actividades, los miembros de la comunidad, los servicios que se brindan, los objetivos e indicadores del proyecto y el historial de atenciones de los miembros de la comunidad, es por ello que conforman las principales clases del sistema a diseñarse, adicionalmente se tienen otras clases que permiten cumplir con todos los requerimientos del sistema, tanto los exigibles como los deseables.

Estas clases exhiben los atributos y asociaciones que deben tener todos los objetos involucrados en la realidad de los proyectos de RSU. Dichas clases se han definido teniendo en cuenta que deben cumplir con todos los requerimientos que están

detallados en el catálogo de requisitos, es por ello que después de elaborarse la lista de exigencias, se detalló los casos de uso asociados a cada requerimiento, posteriormente se realizó la especificación de los casos de uso y se definieron las clases necesarias por paquete para que cumplan con el comportamiento requerido para el sistema según lo indicado en la Especificación de requisitos de software (véase Anexo E).



3. Capítulo 3: Diseño

En este capítulo se detallará el diseño de la solución propuesta y la arquitectura empleada. Así mismo, se explicarán los criterios del diseño de la interfaz gráfica de usuario.

3.1. Arquitectura de la Solución

En este capítulo se presenta una visión general de la arquitectura del sistema, por lo que se abarcará diferentes aspectos del sistema. Toda la información técnica sobre la arquitectura empleada se encuentra en el Documento de Arquitectura (véase Anexo G).

El análisis y diseño es plasmado a través de un conjunto de vistas que permiten mostrar el aspecto funcional y el aspecto técnico propuesto para el sistema. Para la definición de la arquitectura de la solución se ha considerado los diagramas de clases de diseño del sistema, los cuales están basados en las especificaciones de RUP y pueden ser revisados en el Documento de Diseño (véase Anexo H).

Existen algunos requerimientos y necesidades clave para el sistema que influyen sobre la arquitectura. Estos son los siguientes:

- El sistema debe asegurar la protección completa de los datos y el control del acceso no autorizado. El acceso estará sujeto a la identificación del usuario y su respectiva contraseña.
- Todos los requerimientos y performance del sistema, descritos en el Documento de Visión (véase Anexo C) , deben ser considerados durante el desarrollo de la arquitectura.
- Facilitar el trabajo en la etapa de construcción brindando una estructura para el desarrollo del sistema , de esta manera se podrán identificar las partes que lo conforman.
- Proveer una base que facilite los mantenimientos, ampliaciones o modificaciones que en el futuro podría requerir el sistema.

Se han empleado cinco vistas para presentar la arquitectura del sistema, las cuales son: vista de casos de uso, vista lógica, vista de procesos, vista de despliegue y vista de implementación. Estas vistas han sido especificadas basándose en lenguaje UML.

- La vista de casos de uso se encuentra organizada por los siguientes paquetes: paquete de seguridad, paquete de gestión de documentos, paquete de gestión del proyecto, paquete de atención, paquete de evaluación y paquete de reportes.
- La vista lógica contiene los siguientes componentes: pantalla principal, mantenimientos, procesos del negocio, consultas y base de datos.
- La vista de procesos contiene las clases que se utilizarán para el desarrollo del sistema.

- La vista de despliegue muestra la representación física de la implementación, es decir describe cada nodo y cómo se encuentran conectados en la red.
- La vista de implementación contiene las siguientes capas model, controller, view.

Se ha optado por explicar en esta sección la vista lógica y la vista de despliegue debido a su importancia en el cumplimiento de los requerimientos funcionales y no funcionales explicados a detalle en el Capítulo 2.

3.1.1. Vista lógica

Con la finalidad de agrupar las clases del sistema de acuerdo a su funcionalidad, se utiliza la vista lógica. Dicha vista permitirá separar las diferentes funcionalidades en componentes lo que permitirá ordenar la implementación para hacer más fácil el desarrollo del sistema y su mantenimiento a futuro. Los componentes definen una arquitectura que permite cumplir con todos los requerimientos funcionales del sistema pues las clases están basadas en la definición del sistema realizado en el capítulo anterior.

Para el sistema a desarrollarse se proponen los componentes que se muestran en la Figura 3.1. y cuya organización en capas se detalla a continuación:

- **Capa de presentación**

Esta capa está conformada por el componente GUI que contiene todas las clases de interfaz de usuario, para el sistema a desarrollarse será interfaz gráfica Web.

Las clases de este componente permitirán mostrar páginas que servirán para que el usuario pueda interactuar con el sistema.

- **Capa de lógica**

Esta capa está conformada por los componentes BE y BL. Estos componentes poseen todas las clases que contienen la lógica de la solución y las clases que definen el negocio.

Esta capa es el nexo entre la capa de presentación y la capa de almacenamiento.

- **Capa de almacenamiento**

Esta capa está conformada por el componente DAO, dicho componente contiene la clase de conexión a la base de datos y las clases para el almacenamiento de la información de los procesos del negocio que se realizan a través de la solución.

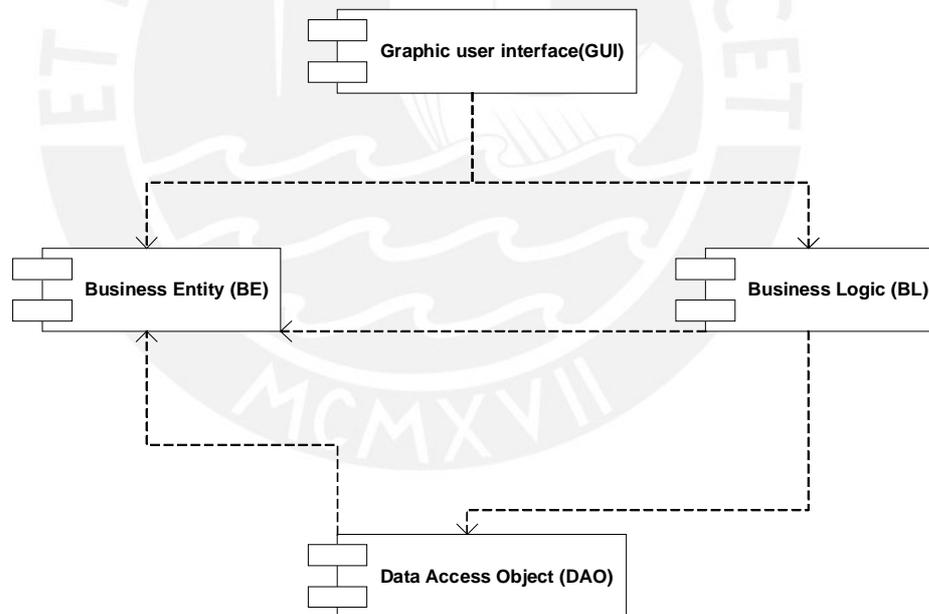


Figura 3.1. Diagrama de componentes

El detalle de cada uno de los componentes y sus dependencias se encuentra detallado en el punto 5 del Documento de Arquitectura (véase Anexo G).

3.1.2. Vista de despliegue

En la vista de despliegue se muestra de manera gráfica los nodos que conforman el sistema, además indica la localización de las tareas en los nodos físicos. Cada nodo representa un recurso de ejecución como una computadora o servidor y se encuentra conectado con otros mediante un enlace de comunicación.

En la figura 3.2. se muestra el diagrama de despliegue el cual se detalla a continuación:

- **Cliente**
En este nodo se implementa el componente GUI a través del cual el usuario interactuará con el sistema. Los usuarios acceden al sistema desde computadoras personales a través de un browser.
- **Servidor de Aplicaciones**
En este nodo reside toda la lógica del negocio.
- **Servidor de Base de datos**
Este nodo representa al único servidor de base de datos, el cual almacenará los datos operacionales de todos los procesos que se realicen en todas las estaciones clientes de la empresa.

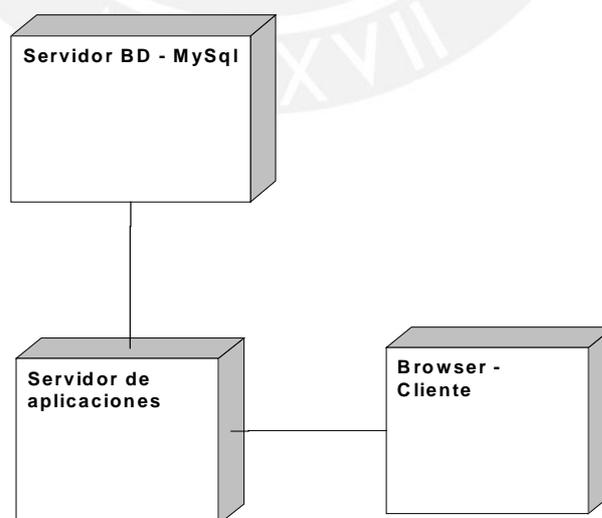


Figura 3.2. Diagrama de despliegue

El detalle de cada uno de los nodos del diagrama de despliegue y sus dependencias se encuentra detallado en el punto 7 del Documento de Arquitectura (véase Anexo G).

La arquitectura seleccionada facilitará la implementación del sistema, así como su mantenimiento, pues los componentes permiten separar la lógica del negocio de la interfaz gráfica y del acceso a la base de datos, por lo que a futuro será mucho más fácil agregar nuevas funcionalidades, migrar a otra base de datos o cambiar la interfaz gráfica, dado que estos cambios solo afectarán a un componente. Adicionalmente, la ubicación de los componentes en los diferentes nodos, permiten tener un sistema seguro y ligero.

3.2. Diseño de la Interfaz Gráfica

El sistema a desarrollarse será una aplicación Web y la interacción con el usuario se realizará por medio de teclado y ratón. Deberá además ser diseñada teniendo siempre en cuenta el hecho de que está orientada a una comunidad de usuarios con conocimientos básicos en computación.

Para el diseño de la interfaz gráfica se ha considerado también el factor económico, pues no se realizará una fuerte inversión en el sistema, además los equipos no siempre serán potentes. Por ello se busca que se prioricen las necesidades funcionales de un proyecto de Responsabilidad Social Universitaria, por lo que la interacción con el sistema y el diseño gráfico del mismo serán, en lo posible, muy sencillos, es decir, no se busca una aplicación recargada de animaciones, sino una aplicación que cuente con todas las funcionalidades mínimas para la solución de los problemas que se presentan. Un diseño gráfico con animaciones sería mucho más costoso que el que se propone y tomaría más tiempo.

El ingreso al sistema será a través de un usuario y contraseña, indicando el nombre del proyecto, por ello la primera página contará con lo necesario para cumplir con este requerimiento, después de validado el ingreso se mostrará un encabezado con el nombre del proyecto y un menú principal al lado izquierdo, el cuál contendrá los módulos a los cuáles se puede acceder de acuerdo al perfil del usuario. Al lado superior derecho, debajo del nombre del proyecto, se mostrará un submenú de acuerdo a lo indicado en el menú principal y debajo de éste el cuerpo de la

aplicación que le permitirá al usuario realizar registros, transacciones, consultas, etc.

En la figura 3.3. se muestra la plantilla principal del diseño de la interfaz gráfica.

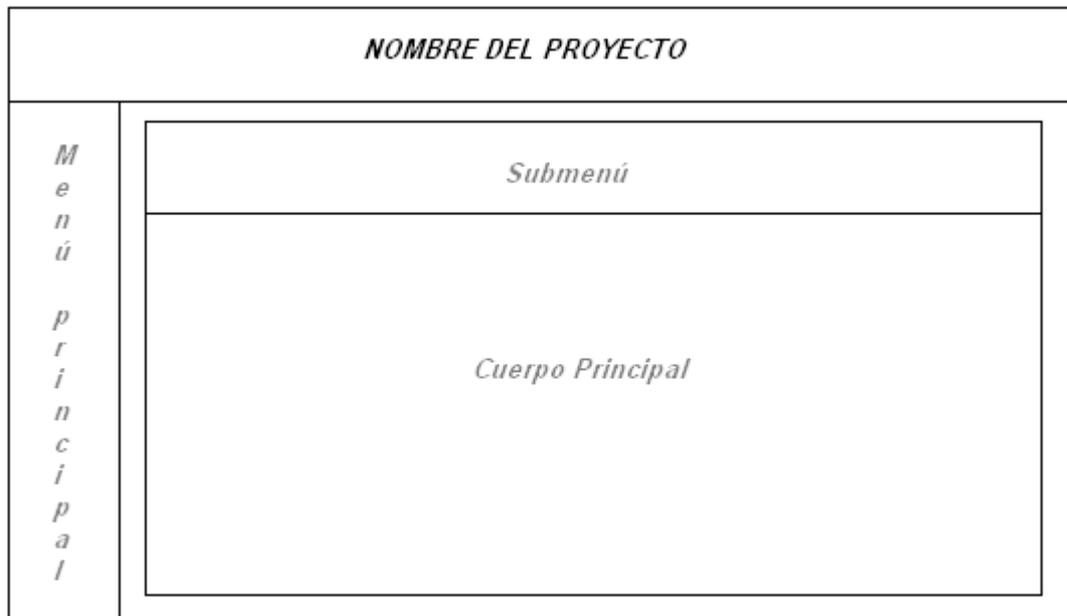


Figura 3.3. Distribución de la Interfaz Gráfica

La distribución anterior permite un manejo intuitivo y sobre todo es de fácil navegabilidad pues el usuario siempre tendrá a la vista el menú principal y el submenú, es decir, sabrá en todo momento donde se encuentra y será muy fácil poder cambiar de página si lo desea.

El usuario deberá tener alguna forma de conocer el resultado de sus acciones, mediante mensajes de confirmación o variaciones en pantalla. Adicionalmente, el sistema manejará en ciertos casos que el usuario considera necesario la utilización de alarmas del sistema para verificar cruces, temas pendientes, posibles errores al momento de registrar actividades o encuestas.

En la figura 3.4. se muestra una de las pantallas del sistema, la cual cumple con lo establecido anteriormente.

MI PRIMER PROYECTO

Módulo de Gestión de Proyecto

[Programar Actividad](#) [Monitorear Actividad](#) [Auditar Actividad](#) [Consultar Actividades](#)

[Servicios](#) [Problemas](#) [Roles](#) [Integrantes](#) [Auspiciadores](#) [Materiales](#) [Organigrama](#) [Grupos Aud.](#)

Seguridad

Documentos

Gestión

Atención

Evaluación

Reportes

Configurador

Proyectos

Lista de Integrantes

Codigo	Nombre	Apellido Paterno	Apellido Materno	Direccion	Telefono	Correo Electronico	Estado	Acciones
1	LORENA	WOOLS	JIMENEZ	JR. ANCASH 399, LIMA CERCADO	6933852	lwools@uch.edu.pe	ACTIVO	Ver detalle Modificar Delete
2	SILVANA	NUNEZ	MENDOZA	AV. LA PERLA CANTO REY	96830069	snunez@usmp.edu.pe	ACTIVO	Ver detalle Modificar Delete
3	NELLY	TREVEJO	ZAMORA	OLLANTAYTAMBO 343	5266578	nelly.trejejo@pucp.edu.pe	ACTIVO	Ver detalle Modificar Delete
4	RAUL	ALVAREZ	TANAKA	ZAA 456	96382269	RALVAREZ@PUCP.EDU.PE	ACTIVO	Ver detalle Modificar Delete
5	KARINA	LIMACHI	ROJAS	AV. LAS VIOLETAS 1236	12345698	kilimachi@usmp.edu.pe	ACTIVO	Ver detalle Modificar Delete
6	MICHEL	MONRROY	AA	ollantaytambo 343	9938324785	allb@pucp.edu.pe	ACTIVO	Ver detalle Modificar Delete
7	JUAN	SOLARI	CHILLCE	OLLANTAYTAMBO 343	5266578	JSOLARI@C.EDU.PE	ACTIVO	Ver detalle Modificar Delete
8	LEILA	GONZALEZ	VALDIVIA	AA 345		lgonzal@pucp.edu.pe	ACTIVO	Ver detalle Modificar Delete
9	FERNANDO	ALBA	ALBA	AAA 123	526548	falba@hotmail.com	ACTIVO	Ver detalle Modificar Delete
10	lucia	salazar	torres	ASAD 12234	45345	STORRES@PUCP.EDU.PE	ACTIVO	Ver detalle Modificar Delete

- [Nuevo Integrante](#)

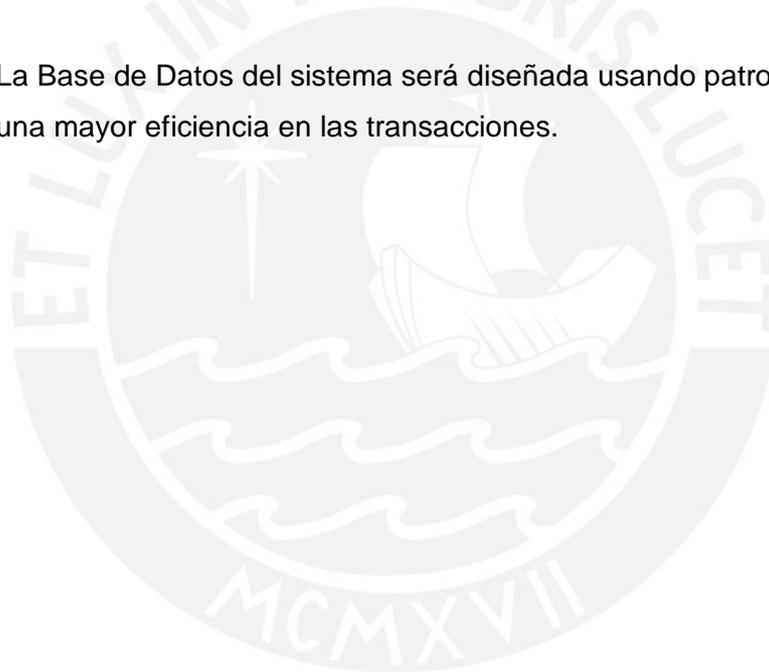
Usuarios no registrados como miembros

Figura 3.4. Pantalla del Sistema de Información

3.3. Consideraciones del Diseño Propuesto

A continuación se listan las consideraciones contempladas en el diseño del Sistema de información propuesto:

- El sistema debe permitir el acceso concurrente de varios usuarios, de modo que puedan realizar transacciones simultáneas.
- La utilización del sistema de información no debe demandar más recursos que los que tendría una computadora como la especificada dentro de los requisitos de hardware mínimos para las computadoras clientes definidas en el Documento de Visión (véase Anexo C).
- La Base de Datos del sistema será diseñada usando patrones que permitan una mayor eficiencia en las transacciones.



4. Capítulo 4: Construcción

En este capítulo se detallarán las decisiones tomadas con respecto a la construcción de la solución propuesta. Así mismo, se explicará la estrategia de pruebas utilizada, los criterios de aceptación y los resultados de las pruebas realizadas.

4.1. Construcción

A continuación se detallará todo lo concerniente a la construcción del Sistema de Información propuesto. En un primer momento se describirán las herramientas empleadas para la implementación y luego los patrones de programación utilizados.

4.1.1. Herramientas para la implementación

Como se explicó en el Capítulo 2, se han considerado diversos criterios para la elección de las herramientas a utilizarse para la implementación del sistema, entre ellos está la necesidad de utilizar herramientas de código libre y un motor de base de datos rápido y ligero. Otro factor fundamental que debe ser considerado es el hecho de que no se cuenta con computadoras muy potentes durante el desarrollo

de las actividades de un proyecto de RSU, por lo que las herramientas no deben consumir muchos ciclos de procesador.

Considerando todos los factores mencionados, se ha optado por la utilización del lenguaje de programación PHP, cuyas siglas significan "PHP Hypertext Pre-processor" (inicialmente PHP Tools, o, Personal Home Page Tools) y el manejador de base de datos MySQL.

PHP es un lenguaje multiplataforma sencillo de aprender que destaca su conectividad con el motor de base de datos MySQL, a pesar de no ser un lenguaje orientado a objetos, permite las técnicas de programación orientada a objetos. Además no requiere ser ejecutado en una máquina muy potente, una computadora 486 con 16 Mb de RAM y con 500Mb de disco puede ser suficiente para ejecutar aplicaciones en PHP versión 5 [21]. La desventaja que presenta PHP es que a medida que aumentan las solicitudes es más ineficiente; sin embargo, se sabe que en promedio 10 personas utilizarán el sistema al mismo tiempo por lo que este riesgo queda eliminado.

MySQL es un manejador de base de datos libre muy veloz en la realización de operaciones y de gran rendimiento, su bajo consumo lo hace ideal para equipos de escasos recursos.

Para poder trabajar con el lenguaje PHP y el motor de base de datos MySQL es necesario contar con un servidor Web, para el sistema a desarrollarse se utilizará servidor http Apache, que es un software de código abierto y multiplataforma.

Para la instalación de todas las herramientas necesarias se ha utilizado VertrigoServ, que es una potente aplicación que permite instalar Apache 2.0.59, PHP 5.2.3 y MySQL 5.0.41 sin necesidad de configuraciones y procesos complejos. En la Figura 4.1., se muestra el menú que proporciona esta aplicación y desde el cual se puede acceder a todas las herramientas que brinda.

Además de las herramientas anteriormente mencionadas, esta aplicación proporciona la herramienta PHPMyAdmin 2.10.2 que es un programa que permite conectarse con una base de datos y administrarla en un entorno de trabajo Web. La base de datos del sistema será administrada a través de este programa.

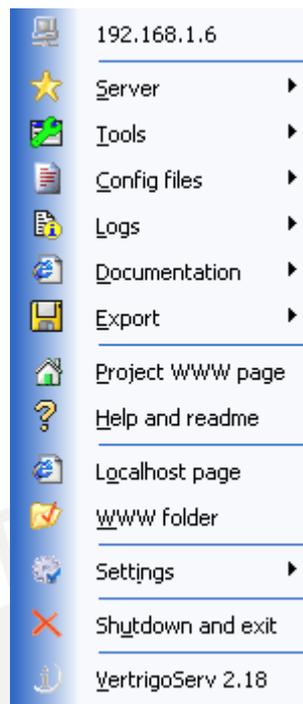


Figura 4.1. Menú de acceso a las herramientas proporcionadas por la aplicación VertrigoServ

4.1.2. Patrones de programación

Un patrón de programación es una solución de diseño de software a un problema, aceptada como correcta, a la que se ha dado un nombre y que puede ser aplicada en otros contextos. Es una abstracción de una solución de diseño a muy alto nivel.[22]

Para la arquitectura técnica del sistema se ha utilizado el Patrón MVC (Model View Controller) y el Patrón DAO(Data Access Object).

A. Patrón MVC(Model View Controller)

El patrón MVC ayuda a dividir la aplicación en capas lógicas, independientes unas de otras, las cuales dividen de manera eficaz la implementación, con lo cual el

comportamiento de cada una de las capas es independiente a lo que pueda hacer la otra.

El patrón MVC divide al sistema en tres partes, como se muestra en la Figura 4.2., y que se describen a continuación:

- **Model:** Modelo o capa de negocio. Encapsula los datos y la funcionalidad de la aplicación detrás del proceso de negocio. En esta capa se tiene que controlar la interacción con la base de datos, realizar cálculos referidos al proceso de negocio, procesar la información ingresada o solicitada por el usuario. Encapsula la data y el comportamiento de manera independiente a como se muestra en la capa de presentación.
- **View:** Vista o capa de presentación. Despliega la información contenida en el modelo, sin importarle como fue obtenida, se puede indicar que muestre los mismos resultados de manera distinta, dependiendo del tipo de cliente. En esta capa la única preocupación es el diseño de las vistas sin importar cómo se procesa la información ni cómo se obtiene.
- **Controller:** Vista o capa de Control. Mediante esta capa se realiza la conexión lógica del usuario con el negocio, toma decisiones respecto a la presentación que se usará, cómo se mostrará (brindar acceso a ciertas vistas). Cuando la aplicación recibe una petición decide cómo debe de manejarla y qué información debe de ser retornada.

Las características más importantes de este patrón son:

- Maneja la vista a través de componentes gráficos en distintos tipos de clientes
- Arquitectura orientada a eventos
- Los componentes de interfaz de usuario (UI) son independientes de su presentación.
- Manejo de la validación y errores de manera simplificada y estándar.

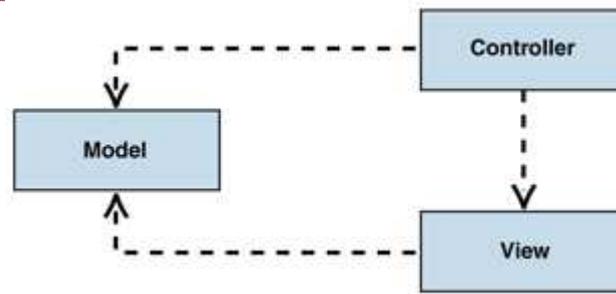


Figura 4.2. Estructura de clases del patrón MVC

B. Patrón DAO(Data Access Object)

Se encarga de la persistencia y acceso a la información. Las características más importantes de este patrón son:

- Implementa la persistencia de la información.
- Desacopla la capa de persistencia de negocio del resto de la aplicación.
- Abstrae y encapsula los accesos, gestiona las conexiones y obtiene los datos almacenados a la fuente de datos, ocultando la implementación completamente del resto de las capas de negocio.
- Separa el código de PHP de SQL.
- Soporta objetos de agrupación como Map, Collections, List, objetos simples y datos primitivos para el manejo de la data.

4.2. Pruebas

A continuación se detallará todo lo concerniente a la realización de las pruebas, que tienen por finalidad la aceptación del producto final. En un primer momento se describirán los tipos de prueba a realizarse y los criterios de aceptación o rechazo para luego presentar los resultados obtenidos.

4.2.1. Planteamiento de casos de prueba

En esta sección se pretende mostrar los casos de prueba que se van a realizar al sistema con la finalidad de localizar defectos en cómo la aplicación satisface los requerimientos solicitados por el usuario. Se ha desarrollado un plan de pruebas

para la administración del proceso y las estrategias en la ejecución de las mismas (ver Anexo J).

Las consideraciones principales para la estrategia de pruebas se basan en las técnicas que se utilizarán y el criterio para saber cuándo una determinada prueba termina. Además de las consideraciones proporcionadas, para cada prueba se debe tener en cuenta que ésta se ejecutará con una base de datos controlada y en un ambiente seguro.

Se utilizarán los siguientes tipos de prueba:

- **Pruebas Unitarias**

Las pruebas unitarias se realizan a una determinada unidad del sistema para poder probar su funcionalidad y se reporta las fallas que tenga. Con las pruebas unitarias se verifica que cada unidad cumpla con las especificaciones deseadas. Las unidades a probar son los módulos del sistema

- **Pruebas de integración**

Las pruebas de integración se realizarán al sistema utilizando la estrategia incremental, se irá integrando módulo a módulo, para poder así probar el sistema de forma parcial y se elaborará el reporte de fallas que se tengan para tomar las medidas de corrección necesarias. El objetivo principal de las pruebas de integración y a nivel del sistema es de verificar que se cumple con todos los requerimientos especificados en su versión actual.

- **Pruebas de regresión**

Las pruebas de regresión son aquellas pruebas selectivas sobre los reportes de incidentes de pruebas generados a partir de las pruebas unitarias y de integración. El objetivo es solucionar dichos reportes de incidentes y realizar nuevamente las pruebas.

4.2.2. Criterios de aceptación o rechazo

El criterio de aceptación o rechazo está ligado directamente con el Catálogo de requisitos y la Especificación de Requerimientos de Software, sobre la base de estos documentos se verificará el cumplimiento de los requerimientos. Así mismo, se han definido las condiciones que deben cumplirse en las pruebas para que sean aceptadas, dichas especificaciones pueden ser revisadas en el punto 5 del Plan de pruebas unitarias y de integración (ver Anexo J).

Las principales pruebas que se realizarán son las siguientes:

Módulo de Seguridad
1. Verificar que el usuario ha ingresado mal su código de usuario o su contraseña de acceso. Ingresar en los campos Código de Usuario y Contraseña datos incorrectos.
2. Verificar que el usuario ha sido desactivado del sistema por un número frecuente de intentos fallidos de contraseña. Ingresar erróneamente un número de contraseñas incorrectas, ingresando con un usuario registrado del sistema.
3. Verificar que el usuario no puede ingresar al sistema desde otra computadora si su sesión está activa. Ingresar el código de usuario y la contraseña correctamente.
4. Si los casos anteriores no se han dado, el usuario puede ingresar sin ningún problema al sistema, siempre que haya elegido el tipo de usuario correcto. Ingresar el código de usuario y la contraseña correctamente.
5. Verificar el tipo de usuario , si es administrador podrá tener accesos solamente a la creación de perfiles, configuración de parámetros, creación de nuevos proyectos y de usuarios. Si es usuario normal, se mostrará la lista de proyectos en los cuáles participa y deberá entrar en la sesión de alguno de ellos, y dependiendo del perfil asignado tendrá acceso solo a ciertas opciones del Menú.
6. Ingresar caracteres no alfanuméricos en los campos: Código de

usuario y contraseña.
7. Consultar la información de un usuario haciendo doble clic sobre el registro del usuario.
8. El usuario debe ingresar todos los datos indicados por la aplicación para registrar un usuario. Registrar un usuario dejando vacíos algunos campos obligatorios.
9. Probar que se ha registrado un usuario en el sistema. Ingresar todos los datos indicados para el registro del usuario y registrar el usuario.
10. Probar que se ha modificado un usuario en el sistema. Ingresar todos los datos indicados para modificación del usuario y modificar el usuario.
11. Probar que se ha eliminado un usuario en el sistema. Seleccionar uno o más usuarios para la eliminación.
12. Consultar la información de un proyecto haciendo doble clic sobre el registro del proyecto.
13. El usuario debe ingresar todos los datos indicados por la aplicación para registrar un proyecto. Registrar un proyecto dejando vacíos algunos campos obligatorios.
14. Probar que se ha registrado un proyecto en el sistema. Ingresar todos los datos indicados para el registro del proyecto y registrar el proyecto.
15. Ingresar cualquier tipo de caracteres excepto letras y espacio en blanco en el campo nombre.
16. Indicar solo un proyecto para la modificación. No seleccionar ningún proyecto o seleccionar más de un proyecto para su modificación.
17. Probar que se ha modificado un proyecto en el sistema. Ingresar todos los datos indicados para modificación del proyecto y modificar el proyecto.
18. Probar que se ha eliminado un proyecto en el sistema. Seleccionar uno o más proyectos para la eliminación.
19. Realizar una búsqueda de perfil. Ingresar los datos indicados para la búsqueda del perfil, según el criterio en que se quiera buscar.
20. Consultar la información de un perfil haciendo doble clic sobre el registro del perfil.
21. El usuario debe ingresar todos los datos indicados por la aplicación para registrar un perfil. Registrar un perfil dejando vacíos algunos campos obligatorios.
22. Probar que se ha registrado un perfil en el sistema. Ingresar todos los datos indicados para el registro del perfil y registrar el perfil.
23. Probar que se ha modificado un perfil en el sistema. Ingresar todos los datos indicados para modificación del perfil y modificar el perfil.

<p>24. Asignar a un usuario un perfil por proyecto en el que participa y probar que éste pueda ingresar a la sesión de dicho proyecto teniendo acceso solo a aquellas funcionalidades permitidas por el perfil asignado para ese usuario en dicho proyecto.</p>
<p>25. Probar que el usuario ha ingresado su contraseña actual incorrecta. El usuario debe ingresar su contraseña actual incorrecta</p>
<p>26. Probar que el usuario ha ingresado como contraseña nueva una cadena nula. El usuario debe ingresar una contraseña nueva nula.</p>
<p>27. Probar que el usuario ha ingresado una contraseña que no tiene ocho caracteres. El usuario debe ingresar una contraseña que no tenga ocho caracteres.</p>
<p>28. Probar que el usuario ha ingresado una contraseña nueva y la confirmación de la nueva contraseña de manera diferente. El usuario debe ingresar una contraseña nueva y la confirmación de la nueva contraseña de forma diferente.</p>
<p>29. Probar que el usuario ha cambiado su contraseña de forma correcta. El usuario debe ingresar su actual contraseña correcta, una contraseña nueva y su confirmación, ambos deben ser iguales, y guardar los cambios realizados.</p>

Módulo de Gestión de documentos
<p>1. Crear un directorio y asignarle un nombre con caracteres inválidos o un nombre ya utilizado en el mismo nivel.</p>
<p>2. Cambiar de nombre un directorio y asignarle un nombre con caracteres inválidos o un nombre ya utilizado en el mismo nivel.</p>
<p>3. Eliminar directorio cuando este contiene documentos.</p>
<p>4. Subir o mover un documento que tiene el mismo nombre de uno ya existente al mismo directorio.</p>
<p>5. Subir un documento que es de mayor tamaño al espacio disponible</p>

Módulo de Gestión del proyecto
<p>1. Consultar la información de un integrante haciendo doble clic sobre el registro del integrante.</p>
<p>2. El usuario debe ingresar todos los datos indicados por la aplicación para registrar un integrante. Registrar un integrante dejando vacíos algunos campos obligatorios.</p>
<p>3. Probar que se ha registrado un integrante en el sistema. Ingresar</p>

<p>todos los datos indicados para el registro del integrante y registrar el integrante.</p>
<p>4. Ingresar cualquier tipo de caracteres excepto letras y espacio en blanco en el campo nombre.</p>
<p>5. Probar que se ha modificado un integrante en el sistema. Ingresar todos los datos indicados para modificación del integrante y modificar el integrante.</p>
<p>6. Verificar que se listen todos los usuarios del proyecto que no son aún registrados como integrantes y permitir registrarlos.</p>
<p>7. Probar que se ha eliminado un integrante en el sistema. Seleccionar uno o más integrantes para la eliminación.</p>
<p>8. Ingresar caracteres no numéricos en los campos número de documento y código de estudiante.</p>
<p>9. Realizar una búsqueda de roles. Ingresar los datos indicados para la búsqueda del rol, según el criterio en que se quiera buscar.</p>
<p>10. Consultar la información de un rol haciendo doble clic sobre el registro del rol.</p>
<p>11. El usuario debe ingresar todos los datos indicados por la aplicación para registrar un rol. Registrar un rol dejando vacíos algunos campos obligatorios.</p>
<p>12. Probar que se ha registrado un rol en el sistema. Ingresar todos los datos indicados para el registro del rol y registrar el rol.</p>
<p>13. Indicar solo un rol para la modificación. No seleccionar ningún rol o seleccionar más de un rol para su modificación.</p>
<p>14. Probar que se ha modificado un rol en el sistema. Ingresar todos los datos indicados para modificación del rol y modificar el rol.</p>
<p>15. Probar que se ha eliminado un rol en el sistema. Seleccionar uno o más roles para la eliminación.</p>
<p>16. Realizar una búsqueda de problemas. Ingresar los datos indicados para la búsqueda del problema, según el criterio en que se quiera buscar.</p>
<p>17. Consultar la información de un problema haciendo doble clic sobre el registro del problema.</p>
<p>18. El usuario debe ingresar todos los datos indicados por la aplicación para registrar un problema. Registrar un problema dejando vacíos algunos campos obligatorios.</p>
<p>19. Probar que se ha registrado un problema en el sistema. Ingresar todos los datos indicados para el registro del problema y registrar el problema.</p>
<p>20. Indicar solo un problema para la modificación. No seleccionar ningún problema o seleccionar más de un problema para su modificación.</p>

21. Probar que se ha modificado un problema en el sistema. Ingresar todos los datos indicados para modificación del problema y modificar el problema.
22. Indicar los problemas para la eliminación. No seleccionar ningún problema.
23. Probar que se ha eliminado un problema en el sistema. Seleccionar uno o más problemas para la eliminación.
24. Realizar una búsqueda de servicios. Ingresar los datos indicados para la búsqueda del servicio, según el criterio en que se quiera buscar.
25. Consultar la información de un servicio haciendo doble clic sobre el registro del servicio.
26. El usuario debe ingresar todos los datos indicados por la aplicación para registrar un servicio. Registrar un servicio dejando vacíos algunos campos obligatorios.
27. Probar que se ha registrado un servicio en el sistema. Ingresar todos los datos indicados para el registro del servicio y registrar el servicio.
28. Indicar solo un servicio para la modificación. No seleccionar ningún servicio o seleccionar más de un servicio para su modificación.
29. Probar que se ha modificado un servicio en el sistema. Ingresar todos los datos indicados para modificación del servicio y modificar el servicio.
30. Indicar los servicios para la eliminación. No seleccionar ningún servicio.
31. Probar que se ha eliminado un servicio en el sistema. Seleccionar uno o más servicios para la eliminación.
32. Realizar una búsqueda de actividades. Ingresar los datos indicados para la búsqueda de la actividad, según el criterio en que se quiera buscar.
33. Consultar la información de una actividad haciendo doble clic sobre el registro de la actividad.
34. El usuario debe ingresar todos los datos indicados por la aplicación para registrar una actividad. Registrar una actividad dejando vacíos algunos campos obligatorios.
35. Probar que se ha registrado una actividad en el sistema. Ingresar todos los datos indicados para el registro de la actividad y registrar la actividad.
36. Indicar solo una actividad para la modificación. No seleccionar ninguna actividad o seleccionar más de una actividad para su modificación.
37. Probar que se ha modificado una actividad en el sistema. Ingresar

<p>todos los datos indicados para modificación de la actividad y modificar la actividad.</p>
<p>38. Indicar las actividades para la eliminación. No seleccionar ninguna actividad.</p>
<p>39. Probar que se ha eliminado una actividad en el sistema. Seleccionar una o más actividades para la eliminación.</p>
<p>40. Realizar una búsqueda de actividades. Ingresar los datos indicados para la búsqueda de la actividad, según el criterio en que se quiera buscar.</p>
<p>41. Auditar la información de una actividad haciendo doble clic sobre el registro de la actividad.</p>
<p>42. El usuario debe ingresar todos los datos indicados por la aplicación para auditar una actividad. Auditar una actividad dejando vacíos algunos campos obligatorios.</p>
<p>43. Probar que se ha auditado una actividad en el sistema. Ingresar todos los datos indicados para auditar la actividad y auditar la actividad.</p>
<p>44. Indicar solo una actividad para auditarla. No seleccionar ninguna actividad o seleccionar más de una actividad para ser auditada.</p>

Módulo de Atención a la comunidad
<p>1. Realizar una búsqueda de personas. Ingresar los datos indicados para la búsqueda de la persona, según el criterio en que se quiera buscar.</p>
<p>2. Consultar la información de una persona haciendo doble clic sobre el registro de la persona.</p>
<p>3. El usuario debe ingresar todos los datos indicados por la aplicación para registrar una persona. Registrar una persona dejando vacíos algunos campos obligatorios.</p>
<p>4. Probar que se ha registrado una persona en el sistema. Ingresar todos los datos indicados para el registro de la persona y registrar la persona.</p>
<p>5. Ingresar cualquier tipo de caracteres excepto letras y espacio en blanco en el campo nombre.</p>
<p>6. Indicar solo una persona para la modificación. No seleccionar ninguna persona o seleccionar más de una persona para su modificación.</p>
<p>7. Probar que se ha modificado una persona en el sistema. Ingresar todos los datos indicados para modificación de la persona y modificar la persona.</p>

8. Indicar las persona para la eliminación. No seleccionar ninguna persona.
9. Probar que se ha eliminado una persona en el sistema. Seleccionar una o más personas para la eliminación.
10. Ingresar caracteres no numéricos en el campo número de documento.
11. Realizar una búsqueda de personas. Ingresar los datos indicados para la búsqueda de la persona, según el criterio en que se quiera buscar.
12. El usuario debe ingresar todos los datos indicados por la aplicación para registrar una ficha. Registrar una ficha dejando vacíos algunos campos obligatorios.
13. Probar que se ha registrado una ficha en el sistema. Ingresar todos los datos indicados para el registro de la ficha y registrar la ficha.
14. El usuario debe ingresar todos los datos indicados por la aplicación para registrar una receta o una orden de actividades médicas. Registrar dejando vacíos algunos campos obligatorios.
15. Probar que se ha registrado una receta o una orden de actividades médicas en el sistema. Ingresar todos los datos indicados para el registro.
16. Probar que se ha modificado una receta o una orden de actividades médicas en el sistema. Ingresar todos los datos indicados para modificación.
17. Realizar una búsqueda de personas. Ingresar los datos indicados para la búsqueda de la persona, según el criterio en que se quiera buscar.

Módulo de Evaluación
1. Realizar una búsqueda de objetivos. Ingresar los datos indicados para la búsqueda del objetivo, según el criterio en que se quiera buscar.
2. Consultar la información de un objetivo haciendo doble clic sobre el registro del objetivo.
3. El usuario debe ingresar todos los datos indicados por la aplicación para registrar un objetivo. Registrar un objetivo dejando vacíos algunos campos obligatorios.
4. Probar que se ha registrado un objetivo en el sistema. Ingresar todos los datos indicados para el registro del objetivo y registrar el objetivo.
5. Indicar solo un objetivo para la modificación. No seleccionar ningún

objetivo o seleccionar más de un objetivo para su modificación.
6. Probar que se ha modificado un objetivo en el sistema. Ingresar todos los datos indicados para modificación del objetivo y modificar el objetivo.
7. Indicar los objetivos para la eliminación. No seleccionar ningún objetivo.
8. Probar que se ha eliminado un objetivo en el sistema. Seleccionar uno o más objetivos para la eliminación.
9. Realizar una búsqueda de indicadores. Ingresar los datos indicados para la búsqueda del indicador, según el criterio en que se quiera buscar.
10. Consultar la información de un indicador haciendo doble clic sobre el registro del indicador.
11. El usuario debe ingresar todos los datos indicados por la aplicación para registrar un indicador. Registrar un indicador dejando vacíos algunos campos obligatorios.
12. Probar que se ha registrado un indicador en el sistema. Ingresar todos los datos indicados para el registro del indicador y registrar el indicador.
13. Indicar solo un indicador para la modificación. No seleccionar ningún indicador o seleccionar más de un indicador para su modificación.
14. Probar que se ha modificado un indicador en el sistema. Ingresar todos los datos indicados para modificación del indicador y modificar el indicador.
15. Indicar los indicadores para la eliminación. No seleccionar ningún indicador.
16. Probar que se ha eliminado un indicador en el sistema. Seleccionar uno o más indicadores para la eliminación.
17. El usuario debe ingresar todos los datos indicados por la aplicación para registrar una encuesta o evaluación. Registrar un indicador dejando vacíos algunos campos obligatorios.
18. Probar que se ha registrado una encuesta o evaluación en el sistema. Ingresar todos los datos indicados para el registro una encuesta o evaluación y registrar la encuesta o evaluación.
19. El usuario debe ingresar todos los datos indicados por la aplicación para registrar resultados. Registrar un resultado dejando vacíos algunos campos obligatorios.
20. Probar que se ha registrado un resultado en el sistema. Ingresar todos los datos indicados para el registro de un resultado y registrar resultado.

El catálogo de pruebas unitarias puede ser revisado a detalle en el punto 11 del Plan de pruebas unitarias y de integración (ver Anexo J) y el catálogo de pruebas de integración en el punto 12 del mismo documento.

Además, se cuenta con criterios para la suspensión como que la unidad a la que se va a realizar la prueba no esté terminada, (cuando una unidad presente errores de compilación y ejecución, o su implementación está incompleta) o cuando algún reporte de incidente de pruebas impida continuar el resto de las pruebas.

Las pruebas podrán ser reiniciadas cuando se agrega alguna nueva unidad o cuando se desea agregar una funcionalidad nueva a una unidad.

4.2.3. Reporte de ejecución de pruebas

En esta sección se presentará el resultado de la evaluación de los casos de pruebas unitarias, de integración y de regresión, para ello se han realizado las pruebas indicadas en el Plan de pruebas unitarias y de integración (ver Anexo J).

Para cada caso de prueba se ha definido un ítem de verificación con la finalidad de registrar si es que la prueba se realizó exitosamente (ÉXITO) o si es que ocurrió una falla en la prueba (FALLO).

A. Evaluación de resultados de pruebas unitarias

En función al análisis de los resultados indicado en el punto 11 del Plan de pruebas unitarias y de integración (ver Anexo J) se puede concluir que las pruebas unitarias fueron exitosas en un 94.8%. Debido a las fallas encontradas se tuvieron que realizar las respectivas pruebas de regresión para verificar que la falla haya sido corregida.

B. Evaluación de resultados de pruebas de integración

En función al análisis de los resultados indicado en el punto 12 del Plan de pruebas unitarias y de integración (ver Anexo J) se puede concluir que las pruebas de

integración fueron exitosas en un 86%. Debido a las fallas encontradas se tuvieron que realizar las respectivas pruebas de regresión para verificar que la falla haya sido corregida.

Luego de realizadas las pruebas de integración se puede concluir que debido a la construcción incremental del sistema de información es más probable encontrar fallas al momento de integrar un nuevo módulo al sistema; sin embargo luego de las pruebas realizadas y de la corrección de las fallas encontradas se puede asegurar el correcto funcionamiento del sistema.



5. Capítulo 5: Observaciones, conclusiones y recomendaciones

En este capítulo se detallarán las observaciones y conclusiones del presente proyecto de fin de carrera. Así mismo, se incluirán recomendaciones y sugerencias para futuras mejoras o nuevos proyectos que mantengan relación con el tema del proyecto desarrollado.

5.1. Observaciones

La mayoría de proyectos de desarrollo de software presentan errores debido a una mala gestión de los requerimientos de software. Es por ello que en el desarrollo del sistema de información se ha priorizado la etapa de análisis, dado que los errores que pudiesen haber en esta etapa no son tan costosos como los que se dan si es que los errores aparecen en la etapa de construcción.

Las etapas de análisis y diseño del sistema de información tomaron poco más del 55% del tiempo de desarrollo del proyecto, esto permitió definir de manera clara y precisa las funcionalidades con el usuario y modelar el sistema de manera óptima por lo que la etapa de construcción se llevó a cabo de manera rápida y los cambios

realizados a pedido del usuario fueron mínimos. Esto permitió mitigar los riesgos que pudiesen aparecer y se redujo la posibilidad de incremento de costos en la etapa de implementación.

5.2. Conclusiones

Después del trabajo realizado se ha concluido lo siguiente:

- La metodología elegida permitió llevar el control de todo el ciclo de vida del producto de manera organizada permitiendo alcanzar los hitos sin presentar mayores retrasos y sobre todo cumplir con los atributos de calidad planteados.
- La inversión de tiempo en las etapas de análisis y diseño redujo riesgos en la implementación y pérdidas de tiempo y costos, debido a que una buena definición de las funcionalidades y la aceptación del usuario del prototipo propuesto aseguran cambios mínimos y la aceptación del producto final.
- La arquitectura elegida permitió el desarrollo del sistema de manera rápida y ordenada. Así mismo, se logró independizar la interfaz gráfica, la lógica de negocio y los datos, con ello se obtuvo un sistema flexible y fácil de mantener. Si a futuro se requiere cambiar el motor de base de datos o el diseño de la interfaz de usuario, este cambio originaría un mínimo impacto en los componentes de la lógica de negocio.
- La selección de las herramientas para la implementación ahorraron tiempo en el inicio de la etapa de construcción, pues el desarrollador ya no debe preocuparse por la configuración de archivos.
- La documentación del proyecto (anexos) permiten a las personas que deseen continuar con el desarrollo del sistema, agregando funcionalidades o mejorando las existentes, hacerlo de manera fácil y adecuada. Así mismo, si el sistema pasa por una auditoría a futuro, la documentación existente servirá de apoyo para el mejor entendimiento del sistema.

5.3. Recomendaciones y trabajos futuros

El sistema de información desarrollado podrá ser utilizado por todas aquellas personas involucradas en algún proyecto de RSU que deseen contar con los beneficios que la herramienta les proporcionaría; para ello, el interesado deberá solicitar la creación de una cuenta administradora, dicha cuenta les permitirá acceder a todas las funcionalidades del sistema.

El otorgamiento de la cuenta de acceso está sujeto a una evaluación para asegurar la correcta utilización del sistema, es por ello que se recomienda a los interesados proporcionar la información necesaria sobre el o los proyectos para los cuáles se requiere la herramienta.

Si bien el uso del sistema es intuitivo, podría ser necesario consultar algún tipo de información sobre la funcionalidad del mismo, por ello es necesario que el usuario ante cualquier duda o eventualidad solicite el servicio de soporte con el que cuenta el sistema.

El sistema de información ha sido desarrollado pensando en la ampliación de funcionalidades que podrían dar soporte a nuevos servicios que se brinden en un proyecto de responsabilidad social. Así mismo, el sistema podrá hacer interfaz con otros sistemas de información que puedan proporcionarle información o funcionalidades útiles como ubicación de asentamientos humanos, detalle de censos realizados, etc.

Los reportes que se generan a través del sistema son reportes mínimos requeridos por el usuario; sin embargo, se podrían agregar nuevos reportes que permitan al usuario tomar decisiones.

Haciendo uso de herramientas de inteligencia de negocios se podría crear un datamart que permita centralizar la información de todos los proyectos de RSU, permitiendo dar un mayor aporte a la comunidad a través de información importante que pudiera obtenerse del mismo. De esta manera, se puede analizar mejor la situación real del país y su desarrollo sostenible.

Bibliografía

- [1] Mónica Jiménez de Barros, “Responsabilidad Social Universitaria”, El Mercurio, Martes 15 de Mayo.
- [2] François Vallaey, *“Breve marco teórico de responsabilidad social universitaria”*, Perú: Pontificia Universidad Católica del Perú.
- [3] François Vallaey, *“El voluntariado solidario: Ventajas y Peligros”*, México: Universidad de las Américas Puebla.
Disponible <http://www.udlap.mx/rsu/pdf/1/EIVoluntariadoSolidario.pdf>. Consultado 22/09/2007
- [4] François Vallaey, *“Indicadores para proyectos de desarrollo en comunidades de aprendizaje”*, México: Universidad de las Américas Puebla. Disponible <http://www.udlap.mx/rsu/pdf/1/IndicadoresParaProyectosdeDesarrollo.pdf>. Consultado 22/05/2007
- [5] María M. Schwalb, *Emilio García, Virginia de Soldevilla, “Buenas prácticas de responsabilidad Social Empresarial” (1ro ed)*. Perú: Universidad del Pacífico, 2005
- [6] *“Pobreza y desarrollo en el Perú” (2006)*. Informe anual 2005 – 2006
- [7] Luis Carrizo, *“Gestión Social del Conocimiento”*, México: Universidad de las Américas Puebla.
Disponible <http://www.udlap.mx/rsu/pdf/3/GestionSocialdelConocimiento.pdf>. Consultado 27/05/2007
- [9] François Vallaey y Luis Carrizo, *“Hacia la construcción de indicadores de responsabilidad social universitaria”*, Colombia: Universidad de Antioquía.
Disponible http://controlinterno.udea.edu.co/docs/proyrsu/doc_interes/IndicadoresResponsabilidadSocialUniversitaria.pdf. Consultado 10/11/2008.
- [10] <http://ganttproject.biz/>. Consultado 18/06/2007.

[11] <http://www.koffice.org/kplato/>. Consultado 18/06/2007.

[12] <http://www.simpleprojectmanagement.com/planner/home.html> . Consultado 18/06/2007.

[13] <http://office.microsoft.com/es-es/epmsolution/HA101655113082.aspx>. Consultado 18/06/2007.

[14] <http://www.box.net/> y <http://www.4shared.com/> . Consultado 18/06/2007.

[15] Salvador Otón, José M. Gutiérrez, Javier Macías y Roberto Barchino, “*Sistema de almacenamiento y gestión de documentos para el desarrollo de bibliotecas digitales*”, Madrid: Universidad de Alcalá, Departamento de Ciencias de la Computación.

[16] <http://www.costaisa.com/defaultChaman.htm> . Consultado 18/06/2007.

[17] <http://www.reynox.com.ar/sap/metodologia.php>. Consultado 19/09/2007.

[18] Jeffrey K. Pinto, David I., “*The Frontiers of Project Management Research*”, *Project Management Institute*, 2003

[19] “*PMI, A guide to the Project Management Body of Knowledge*” (3ra ed.), *Project Management Institute*, 2003

[20] Patricio Letelier, “*Introducción a Rational Unified Process*”, España: *Universidad Politécnica de Valencia, Departamento de Sistemas Informáticos y Computación*.

[21] <http://www.php.net/> . Consultado 12/01/2008.