Universidad Católica Andrés Bello.

Facultad de Ingeniería.

Escuela Ingeniería Informática.

# Desarrollo de una herramienta de apoyo a una adaptación metodológica basada en SCRUM y eXtreme Programming para el manejo de proyectos de software de la empresa TRASCEND.

**Este jurado; una vez realizado el examen del presente trabajo ha evaluado su contenido con el resultado:** González Stephanie (Dieciséis puntos).
Revilla Andrea (Dieciocho puntos).

JURADO EXAMINADOR

**Firma:** **Nombre:** Cardoso Lucia.

**Firma:** **Nombre:** Campos Rodolfo.

**Firma:** **Nombre:** Magurno Carlo.

| | |
|---|---|
| REALIZADO POR | González S. Stephanie A. |
| | Revilla P. Andrea. |
| TUTOR | Ing. Magurno C. Carlo S. |
| FECHA | 23 de Octubre de 2008. |

## Apéndice A. Historias de Usuario

| Historia de Usuario | |
|---|---|
| **Número:**1 | **Iteración:** 1 |
| **Nombre de Historia:** Estudiar los procesos de SCRUM y XP | |
| **Prioridad:** 1 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** estudio de XP y SCRUM realizando la comparación de sus prácticas para poder escoger las que serán utilizadas en la adaptación metodológica. | |
| **Observaciones:** | |

**A.1 – Historia de Usuario Estudio de SCRUM y XP.**
Elaboración Propia.

| Historia de Usuario | |
|---|---|
| **Número:**2 | **Iteración:** 1 |
| **Nombre de Historia:** Selección de prácticas | |
| **Prioridad:**2 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** seleccionar las mejores prácticas de desarrollo de software que serán utilizadas para realizar la adaptación. | |
| **Observaciones:** | |

**A.2 –Historia de Usuario Selección de Prácticas.**
Elaboración Propia.

| Historia de Usuario | |
|---|---|
| **Número:** 3 | **Iteración:** 1 |
| **Nombre de Historia:** Adaptación | |
| **Prioridad:** 3 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Con las prácticas seleccionadas se realizará la adaptación, definiendo cada práctica y escribiendo el proceso que cumplirá la metodología. | |
| **Observaciones:** | |

**A.3 – Historia de Usuario Adaptación**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:** 4 | **Iteración:** 2 |
| **Nombre de Historia:** Diseño de Interfaz de Usuario | |
| **Prioridad:** 1 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Diseño de las pantallas de interfaz de usuario que se utilizarán para los distintos módulos (peticiones, defectos, pruebas, equipos de trabajo, planificación y entregas). Se escogerá formato para los formularios, botones, menús, navegación, etc. | |
| **Observaciones:** | |

**A.4 – Historia de Usuario Interfaz de Usuario**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**5 | **Iteración:** 3 |
| **Nombre de Historia:** Listar Proyecto | |
| **Prioridad:** 1 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de proyecto. La herramienta de apoyo debe listar los proyectos de forma de poder hacer seguimiento a los existentes. | |
| **Observaciones:** | |

**A.5 – Historia de Usuario Listar Proyecto**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**6 | **Iteración:** 3 |
| **Nombre de Historia:** Agregar Proyecto | |
| **Prioridad:** 2 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de proyecto. El usuario debe ser capaz de ingresar los datos principales de los proyectos en un formulario. Los datos que debe tener son: nombre, descripción. Estos deben ser grabados en la base. | |
| **Observaciones:** | |

**A.6 – Historia de Usuario Agregar Proyecto**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**7 | **Iteración:** 3 |
| **Nombre de Historia:** Editar Proyecto | |
| **Prioridad:** 3 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de proyecto. La herramienta de apoyo debe contar con la opción de editar la información de los proyectos. | |
| **Observaciones:** | |

A.7 – Historia de Usuario Editar Proyecto
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**8 | **Iteración:** 3 |
| **Nombre de Historia:** Eliminar Proyecto | |
| **Prioridad:** 4 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de proyecto. El usuario debe ser capaz de eliminar los proyectos existentes. | |
| **Observaciones:** | |

A.8 – Historia de Usuario Eliminar Proyecto
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**9 | **Iteración:** 3 |
| **Nombre de Historia:** Listar Petición | |
| **Prioridad:** 5 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de peticiones. La herramienta de apoyo debe presentar una lista con todas las peticiones que existan de un proyecto. | |
| **Observaciones:** | |

**A.9 – Historia de Usuario Listar Petición**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**10 | **Iteración:** 3 |
| **Nombre de Historia:** Ingresar Petición | |
| **Prioridad:** 6 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de peticiones o requerimientos. El usuario debe ser capaz de ingresar los requerimientos pertenecientes al Product Backlog en un formulario. Los datos que debe tener son: descripción, tiempo estimado, prioridad, grado de complejidad, status. Estos deben ser grabados en la base. | |
| **Observaciones:** | |

**A.10 – Historia de Usuario Ingresar Petición**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**11 | **Iteración:** 3 |
| **Nombre de Historia:** Editar Petición | |
| **Prioridad:** 7 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de peticiones o requerimientos. El usuario debe ser capaz de modificar los datos (descripción, tiempo estimado, prioridad, grado de complejidad, status) de los requerimientos existentes. Todos los cambios deben ser grabados en la base de datos. | |
| **Observaciones:** | |

**A.11 – Historia de Usuario Editar Petición**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**12 | **Iteración:** 3 |
| **Nombre de Historia:** Eliminar Petición | |
| **Prioridad:** 8 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de peticiones o requerimientos. El usuario debe ser capaz de eliminar los requerimientos existentes. | |
| **Observaciones:** | |

**A.12 – Historia de Usuario Eliminar Petición**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**13 | **Iteración:**3 |
| **Nombre de Historia:** Listar Defecto | |
| **Prioridad:** 9 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de defectos.<br>La herramienta de apoyo debe listar todos los defectos existentes de un proyecto específico. | |
| **Observaciones:** | |

**A.13 – Historia de Usuario Listar Defecto**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**14 | **Iteración:** 3 |
| **Nombre de Historia:** Ingresar Defecto | |
| **Prioridad:** 10 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de defectos. El usuario debe ser capaz de ingresar nuevos defectos que surjan durante el desarrollo de un proyecto. | |
| **Observaciones:** | |

**A.14 – Historia de Usuario Ingresar Defecto**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**15 | **Iteración:** 3 |
| **Nombre de Historia:** Modificar Defecto | |
| **Prioridad:** 11 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de defectos.<br>El usuario debe tener la opción de modificar los datos defectos existentes. Todos los cambios deben ser grabados en la base de datos. | |
| **Observaciones:** | |

A.15 – Historia de Usuario Modificar Defecto
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:** 16 | **Iteración:** 3 |
| **Nombre de Historia:** Eliminar Defecto | |
| **Prioridad:**12 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de defectos.<br>El usuario debe tener la posibilidad de eliminar los defectos existentes. | |
| **Observaciones:** | |

A.16 – Historia de Usuario Eliminar Defecto
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**17 | **Iteración:** 3 |
| **Nombre de Historia:** Listar Prueba | |
| **Prioridad:** 13 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de pruebas.<br>La herramienta de apoyo debe listar todas las pruebas existentes de un proyecto. | |
| **Observaciones:** | |

A.17 – Historia de Usuario Listar Prueba
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**18 | **Iteración:** 3 |
| **Nombre de Historia:** Ingresar Prueba | |
| **Prioridad:** 14 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de pruebas. El usuario debe ser capaz de ingresar las pruebas realizadas durante el desarrollo. Los datos ingresados deben ser: requerimiento/tarea asociada, nombre y status. Los datos deben ser grabados en la base de datos. | |
| **Observaciones:** | |

A.18 – Historia de Usuario Ingresar Prueba
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**19 | **Iteración:** 3 |
| **Nombre de Historia:** Modificar Prueba | |
| **Prioridad:** 15 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de pruebas. El usuario debe ser capaz de modificar las pruebas existentes. Todos los cambios deben ser grabados en la base de datos. | |
| **Observaciones:** | |

**A.19 – Historia de Usuario Modificar Prueba**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**20 | **Iteración:** 3 |
| **Nombre de Historia:** Eliminar Prueba | |
| **Prioridad:** 16 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de pruebas. El usuario debe ser capaz de eliminar las pruebas existentes. | |
| **Observaciones:** | |

**A.20 – Historia de Usuario Eliminar Prueba**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**21 | **Iteración:** 3 |
| **Nombre de Historia:** Listar Equipo | |
| **Prioridad:** 17 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de equipos de trabajo. La herramienta de apoyo debe listar todos los integrantes del equipo de trabajo de un proyecto. | |
| **Observaciones:** | |

**A.21 – Historia de Usuario Listar Equipo**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**22 | **Iteración:** 3 |
| **Nombre de Historia:** Agregar Integrante | |
| **Prioridad:** 18 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de equipos de trabajo. Formulario que brinde la posibilidad al usuario de ingresar un nuevo integrante al equipo. Los datos que debe presentar son: nombre y apellido de la persona. Los datos deben ser grabados en la base de datos. También debe ser capaz de asignar las personas ya existentes a un proyecto | |
| **Observaciones:** | |

**A.22 – Historia de Usuario Agregar Integrante**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**23 | **Iteración:** 3 |
| **Nombre de Historia:** Eliminar Integrante | |
| **Prioridad:** 19 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de equipos de trabajo. El usuario debe ser capaz de eliminar un integrante del equipo del proyecto. | |
| **Observaciones:** | |

A.23 – Historia de Usuario Eliminar Integrante
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**24 | **Iteración:** 3 |
| **Nombre de Historia:** Editar Persona. | |
| **Prioridad:** 20 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de equipos de trabajo. El usuario debe ser capaz de editar los datos de las personas existentes. | |
| **Observaciones:** | |

A.24 – Historia de Usuario Editar Persona
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**25 | **Iteración:** 3 |
| **Nombre de Historia:** Eliminar Persona | |
| **Prioridad:** 21 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de manejo de equipos de trabajo. El usuario debe ser capaz de eliminar las personas existentes de la base de datos. | |
| **Observaciones:** | |

A.25 – Historia de Usuario Eliminar Persona
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**26 | **Iteración:** 4 |
| **Nombre de Historia:** Listar Iteraciones | |
| **Prioridad:** 1 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Se debe poder ver todas las iteraciones por proyecto con su descripción más importante. | |
| **Observaciones:** | |

A.26 – Historia de Usuario Listar Iteraciones
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**27 | **Iteración:** 4 |
| **Nombre de Historia:** Ingresar iteración. | |
| **Prioridad:** 2 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. En la herramienta de apoyo se debe poder ingresar nuevas iteraciones asociadas a los proyectos. | |
| **Observaciones:** | |

A.27 – Historia de Usuario Ingresar iteración
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**28 | **Iteración:** 4 |
| **Nombre de Historia:** Editar Iteración | |
| **Prioridad:** 3 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Para cada iteración el usuario podrá editar los datos de las iteraciones ya agregadas. Solo se podrán editar sino han sido iniciadas. | |
| **Observaciones:** | |

A.28 – Historia de Usuario Editar Iteración
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**29 | **Iteración:** 4 |
| **Nombre de Historia:** Eliminar Iteración | |
| **Prioridad:** 4 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Las iteraciones no culminadas podrán ser eliminadas. | |
| **Observaciones:** | |

**A.29 – Historia de Usuario Eliminar Iteración**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**30 | **Iteración:** 4 |
| **Nombre de Historia:** Cambiar status de iteración | |
| **Prioridad:** 5 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Para cada iteración se deberá llevar el control del status en el que se encuentra: sin iniciar, iniciada, culminada. | |
| **Observaciones:** | |

**A.30 – Historia de Usuario Cambiar status de iteración**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**31 | **Iteración:** 4 |
| **Nombre de Historia:** Asignar Peticiones | |
| **Prioridad:** 6 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Para cada iteración se debe poder asociar los requerimientos que van a ser desarrollados, para mantener el control de su status. | |
| **Observaciones:** | |

**A.31 – Historia de Usuario Asignar Peticiones**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**32 | **Iteración:** 5 |
| **Nombre de Historia:** Editar peticiones | |
| **Prioridad:** 7 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Las peticiones de cada una de las iteraciones deben poder ser editadas. | |
| **Observaciones:** | |

**A.32 – Historia de Usuario Editar Peticiones**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**33 | **Iteración:** 4 |
| **Nombre de Historia:** Cambiar status de petición | |
| **Prioridad:** 8 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Para cada uno de los requerimientos asociados a una iteración que van a ser desarrollados, se debe tener la opción de cambiar el status: sin iniciar, iniciado, finalizado. | |
| **Observaciones:** | |

**A.33 – Historia de Usuario Cambiar status Petición**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**34 | **Iteración:** 4 |
| **Nombre de Historia:** Listar Tarea/ Pruebas Funcionales por Petición | |
| **Prioridad:** 9 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Cada uno de los requerimientos va a ser dividido en las distintas tareas que lo componen, este módulo debe brindar la posibilidad al usuario de visualizar todas las tareas de un requerimiento, además de las pruebas funcionales asociadas a la petición. | |
| **Observaciones:** | |

**A.34 – Historia de Usuario Listar tarea/Pruebas Funcionales por Petición**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**35 | **Iteración:** 4 |
| **Nombre de Historia:** Agregar tarea | |
| **Prioridad:** 10 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Cada uno de los requerimientos va a ser dividido en las distintas tareas que lo componen, este módulo debe brindar la posibilidad al usuario de agregar las tareas de un requerimiento. | |
| **Observaciones:** | |

**A.35 – Historia de Usuario Listar tarea**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**36 | **Iteración:** 4 |
| **Nombre de Historia:** Editar tarea | |
| **Prioridad:** 11 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Para cada una de las tareas agregadas se debe presentar la opción de editar los datos de la misma. | |
| **Observaciones:** | |

**A.36 – Historia de Usuario Editar tarea**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**37 | **Iteración:** 4 |
| **Nombre de Historia:** Eliminar tarea | |
| **Prioridad:** 12 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Se debe presentar al usuario la opción de eliminar las tareas que no han sido iniciadas. | |
| **Observaciones:** | |

**A.37 – Historia de Usuario Eliminar tarea**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**38 | **Iteración:** 4 |
| **Nombre de Historia:** Cambiar status de tarea | |
| **Prioridad:** 13 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Cada una de las tareas debe tener un campo para llevar el control del status de la misma, y la opción de cambiarlo cuando se inicie o se culmine la misma. | |
| **Observaciones:** | |

**A.38 – Historia de Usuario Cambiar status tarea**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**39 | **Iteración:** 4 |
| **Nombre de Historia:** Ingresar prueba unitaria | |
| **Prioridad:** 14 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Las tareas de cada requerimiento van a tener asociadas una o varias pruebas unitarias, por lo que debe ser posible ingresarlas. | |
| **Observaciones:** | |

**A.39 – Historia de Usuario Ingresar prueba unitaria**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**40 | **Iteración:** 4 |
| **Nombre de Historia:** Editar prueba unitaria | |
| **Prioridad:** 15 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Para cada una de las pruebas unitarias asociadas a las tareas deben presentar la opción de editar sus datos. | |
| **Observaciones:** | |

**A.40 – Historia de Usuario Editar prueba unitaria**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**41 | **Iteración:** 4 |
| **Nombre de Historia:** Eliminar prueba unitaria | |
| **Prioridad:** 16 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Para cada una de las pruebas unitarias asociadas a las tareas deben presentar la opción de eliminar. | |
| **Observaciones:** | |

A.41 – Historia de Usuario Eliminar prueba unitaria
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**42 | **Iteración:** 4 |
| **Nombre de Historia:** Ingresar/editar horas por tarea | |
| **Prioridad:** 17 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Para cada una de las tareas asociadas a la interacción se debe llevar un control de las horas utilizadas o ejecutadas para su realización, por lo que debe ser posible ingresar y editar las horas por cada tarea. | |
| **Observaciones:** | |

A.42 – Historia de Usuario Ingresar/Editar horas por tarea
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**43 | **Iteración:** 4 |
| **Nombre de Historia:** Listar Horas por Tarea | |
| **Prioridad:** 18 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de planificación. Para cada una de las tareas asociadas a la interacción se debe llevar un control de las horas utilizadas o ejecutadas para su realización, por lo que debe ser posible visualizar las horas por cada tarea. | |
| **Observaciones:** | |

**A.43 – Historia de Usuario Listar Horas por Tarea**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**44 | **Iteración:** 4 |
| **Nombre de Historia:** Listar Entregas | |
| **Prioridad:** 19 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de Entregas. Todas las entregas realizadas por cada producto, deben ser mostradas de forma ordenada. | |
| **Observaciones:** | |

**A.44 – Historia de Usuario Listar Entrega**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**45 | **Iteración:** 4 |
| **Nombre de Historia:** Crear Entrega | |
| **Prioridad:** 20 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de Entrega. Al finalizar cada iteración debe ser posible, crear la entrega de los requerimientos finalizados. | |
| **Observaciones:** | |

**A.45- Historia de Usuario Crear Entrega**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**46 | **Iteración:** 4 |
| **Nombre de Historia:** Editar Entrega | |
| **Prioridad:** 21 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de Entrega. Para cada entrega creada debe ser posible editar los datos asociados de la misma. | |
| **Observaciones:** | |

**A.46 – Historia de Usuario Editar Entrega**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**47 | **Iteración:** 4 |
| **Nombre de Historia:** Eliminar Entrega | |
| **Prioridad:** 22 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de Entrega. El usuario debe tener la opción de eliminar las entregas creadas por cada proyecto. | |
| **Observaciones:** | |

**A.47 – Historia de Usuario Eliminar Entrega**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**48 | **Iteración:** 4 |
| **Nombre de Historia:** Listar Peticiones por entrega | |
| **Prioridad:** 23 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de Entrega. La herramienta debe brindarle al usuario la opción de visualizar las peticiones asociadas a una entrega. | |
| **Observaciones:** | |

**A.48 – Historia de Usuario Listar Peticiones por Entrega**
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**49 | **Iteración:** 4 |
| **Nombre de Historia:** Listar reportes | |
| **Prioridad:** 24 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de Reportes. La herramienta debe brindarle al usuario la opción de visualizar la lista de los distintos reportes que existen. | |
| **Observaciones:** | |

A.49 – Historia de Usuario Reportes-Listar reportes
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**50 | **Iteración:** 4 |
| **Nombre de Historia:** Mostrar reporte como tabla | |
| **Prioridad:** 25 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de Reportes. La herramienta debe brindarle al usuario la opción de visualizar los reportes como una tabla con los datos necesarios. | |
| **Observaciones:** | |

A.50 – Historia de Usuario Reportes-Mostrar reporte como tabla
Elaboración Propia

| Historia de Usuario | |
|---|---|
| **Número:**51 | **Iteración:** 4 |
| **Nombre de Historia:** Mostrar reporte como gráfico | |
| **Prioridad:** 26 | |
| **Responsable:** Stephanie González-Andrea Revilla | |
| **Descripción:** Módulo de Reportes. La herramienta debe brindarle al usuario la opción de visualizar los reportes de forma gráfica. Dependiendo de la información a visualizar se debe escoger un grafico que facilite su entendimiento. | |
| **Observaciones:** | |

**A.51 – Historia de Usuario Reportes-Cumplimiento de tareas por persona**
Elaboración Propia

Nombre de archivo: Apéndice A-Historias-de-Usuario
Directorio: C:\Documents and Settings\diseno.SOINCOPY\Mis documentos
Plantilla: C:\Documents and Settings\diseno.SOINCOPY\Datos de programa\Microsoft\Plantillas\Normal.dotm
Título:
Asunto:
Autor: Stephanie Gonzalez
Palabras clave:
Comentarios:
Fecha de creación: 11/09/2008 12:28:00
Cambio número: 12
Guardado el: 23/09/2008 18:06:00
Guardado por: Stephanie Gonzalez
Tiempo de edición: 59 minutos
Impreso el: 22/10/2008 16:24:00
Última impresión completa
Número de páginas: 26
Número de palabras: 3.236 (aprox.)
Número de caracteres: 17.803 (aprox.)

## Apéndice B. Comparación de XP-SCRUM

| | XP | | SCRUM | |
|---|---|---|---|---|
| Roles | Cliente | *Personas o empresa que genere la idea del producto.<br>* Debe definir qué actividades son más importantes y cuáles deben ser desarrolladas primero. | Dueño del Producto | *Responsable de representar los intereses de todos los involucrados.<br>*Debe definir los requerimientos iniciales del proyecto y prioritizarlos frecuentemente. |
| | Entrenador | *Supervisar el proceso general del proyecto.<br>*Debe conocer las aplicaciones de XP a fondo. | Scrum *Master* | * Responsable del proceso de Scrum.<br>*Asegura que todos sigan sus reglas y prácticas. |
| | Jefe | *Dueño o persona encargada de la empresa que realiza el proyecto. | - | - |
| | Probador | *Ayudar al cliente a escoger y escribir pruebas funcionales.<br>*Correr todas las pruebas regularmente. | - | - |
| | Programador | *Definir la arquitectura.<br>*Diseñar el Sistema.<br>*Escribir pruebas y código. | Equipo | * Desarrollan la funcionalidad.<br>* Auto-gerenciados, auto-organizados y multifuncionales. |
| | Rastreador | *Hacer estimados del proyecto.<br>*Verificar el funcionamiento general del proyecto llevando el récord de los resultados de las pruebas funcionales y de los defectos. | - | - |

**B.1 Comparación de Roles XP-SCRUM**
Elaboración Propia.

| | XP | | SCRUM | |
|---|---|---|---|---|
| Esquema General | Planificación | *Desarrolladores definen a profundidad las tareas que deben ser realizadas para cada historia de usuario. | Planificación | *Crear/modificar *product backlog* *Crear el *release* y *sprint backlog.* *Estimaciones |
| | Diseño | *Se establecen las formas más simples de realizar el desarrollo, los estándares de codificación y metáforas | - | - |
| | Desarrollo | * Desarrollo de cada una de las tareas, probando e integrando el código constantemente. | Desarrollo | * Desarrollo de cada una de las tareas. *El equipo determina como construir la funcionalidad, modificando el enfoque diariamente a medida que encuentra nuevas complejidades, dificultades y sorpresas. |
| | Pruebas | * Se corren las pruebas de cada módulo y se hace cualquier corrección que sea necesaria. *Luego de tener el incremento listo. El entrenador se encarga de hacer una revisión de la iteración, verificando las prácticas cumplidas. | Revisión | *Se presenta el incremento del producto al *Product Owner.* *Se hace una revisión del funcionamiento general del *Sprint.* |

**B.2 Comparación del Esquema General XP-SCRUM**

Elaboración Propia.

| | XP | | SCRUM | |
|---|---|---|---|---|
| Prácticas | Iteraciones | * Duran de 2 a 4 semanas.<br>* Durante cada iteración se realizarán una cierta cantidad de Historias de Usuario para un total de puntos definidos. | *Sprint* | *Iteración de 30 días consecutivos.<br>*Se realizan los requerimientos escogidos del *product backlog*. |
| | - | - | Reunión de Planificación | *El *Product Owner* presenta al equipo los requerimientos del *Product Backlog* de mayor prioridad.<br>*El equipo planea el *Sprint*. |
| | Metáforas | * Ayudan a entender todos los aspectos de la arquitectura del sistema.<br>*Para identificar las entidades técnicas se usarán palabras sacadas de la metáfora escogida | - | - |
| | Diseño Simple | *El sistema será el diseño más simple que le de soporte a la funcionalidad actual. | - | - |
| | *Test Driven Development* (TDD) | * Técnica evolutiva de desarrollo que consiste en escribir las pruebas automatizadas de la unidad antes de escribir el código. | - | - |
| | Integración Continua | * Se realizará integración y pruebas de ésta varias veces al día.<br>* Si falla alguna prueba las personas realizando la integración en ese momento | - | - |

| | | | | |
|---|---|---|---|---|
| | | deben solucionar el problema.<br>* Al final de cada iteración se realiza una integración con el último *release*, y se corren las pruebas. | | |
| | Reuniones de pie | *Reuniones cortas realizadas con todos los integrantes del equipo.<br>*Se revisa el progreso, las metas del día, y los inconvenientes presentados. | Reuniones diarias | * El equipo se reúne durante 15 minutos<br>* Se sincroniza el trabajo de todos los miembros del equipo diariamente. |
| | Programación en Parejas | * Colaboración de programadores.<br>* Las parejas programarán juntas (en una computadora). | - | - |
| | Pruebas Continuas | *Todos los programadores hacen pruebas de código todo el tiempo.<br>* Estas pruebas deben funcionar en su totalidad.<br>*Normalmente se trabaja con pruebas unitarias y funcionales. | - | - |
| | Estándares de códigos | * Los programadores deben escribir el código siguiendo reglas, y bajo comunicación constante.<br>* Para identificar las entidades técnicas se usarán palabras sacadas de la metáfora escogida. | - | - |
| | - | - | Reuniones de Revisión | *El equipo presenta al *Product Owner*, lo que ha |

| | | | | |
|---|---|---|---|---|
| | | | | desarrollado durante el *Sprint*. |
| | Revisión Final de XP | *Revisar el proceso general de la iteración, verificando las prácticas de XP cumplidas y las que faltaron por cumplir, para mejorar en la siguiente iteración. | Reunión de Retrospectiva | *Revisar el proceso de desarrollo para hacerlo más efectivo y agradable para el próximo *Sprint*. |

**B.3 Comparación de Prácticas XP-SCRUM**

Elaboración Propia.

| | XP | | SCRUM | |
|---|---|---|---|---|
| Artefactos | Historias de Usuario | *Muestran la idea general de cada uno de los requerimientos del proyecto.<br>* Son escritas en fichas. | *Product Backlog* | * Lista de requerimientos.<br>* Es dinámico, sus cambios deben ser administrados constantemente.<br>*Se puede registrar el nombre del item, tiempo estimado inicial, factor de complejidad, sprint en el cual es desarrollado. |
| | - | - | *Release Backlog* | *Contiene los requerimientos escogidos para la iteración. |
| | - | - | *Sprint Backlog* | *Contiene las tareas que el equipo define para completar los requerimientos seleccionados. |
| | Casos de Pruebas Funcionales | * Los clientes escriben pruebas para cada historia de usuario.<br>* No siempre funcionan en su totalidad, a medida que se va avanzando el porcentaje de fallas va disminuyendo. | - | - |
| | Incremento Funcional | *Ejecutable que se le entrega al cliente al final de cada iteración, totalmente probado y funcional. | Incremento del producto | *Debe ser un ejecutable cuyo código esté bien escrito, estructurado y probado. |

**8.4 Comparación de Artefactos XP-SCRUM**

Elaboración Propia.

## Apéndice C. Adaptación Metodológica

Esquema ..... 104

Proceso ..... 105

Roles ..... 108

*Dueño del Producto* ..... 108

*Gerente de Proyecto* ..... 109

*Equipo* ..... 109

Flujo de Trabajo ..... 112

*Actividades de Planificación* ..... 112

*Creación o actualización de la Lista de Requerimientos (Product Backlog)* ..... 112

*Reunión de Planificación* ..... 113

Primera Parte – Elegir requerimientos ..... 113

Segunda Parte - Creación de lista de Tareas (Sprint Backlog) ..... 114

Boceto inicial del documento de arquitectura del software ..... 114

Actividades de Desarrollo ..... 115

*Reunión diaria* ..... 115

*Creación del Prototipo de Interfaz* ..... 115

*Test Driven Development (TDD)* ..... 116

*Integración Continua* ..... 117

*Control de versiones* ..... 117

*Revisión de Código* ..... 118

*Reunión de Revisión* ..... 119

*Reunión de Retrospectiva* ..... 119

Artefactos ..... 119

*Lista de Requerimientos (Product Backlog)* ..... 119

*Documento de especificación de requerimientos del software* ..... 120

*Lista de Funcionalidades (Release Backlog)* ..... 120

*Lista de Tareas (Sprint Backlog)* ..... 121

*Documento de Arquitectura del software* ..... 121

*Prototipo de Interfaz* ..... 122

*Pruebas* ..... 122

*Incremento funcional (Release)* ..... 122

## Esquema

Gran parte de la complejidad del desarrollo de software radica en la ambigüedad de los requerimientos del mismo y en los constantes cambios que éstos sufren. La intención de este esquema es manejar esta complejidad, mediante un conjunto de simples prácticas y reglas que componen un proceso de control empírico.

Los procesos de control definidos son aquellos que repetidamente generan productos que cumplen con los requerimientos solicitados. Cuando el proceso es demasiado complicado, como el de desarrollo de software, los procesos empíricos son la mejor opción.

Estos se basan en tres fundamentos:

- Visibilidad. Aquellos aspectos del proceso que afectan el producto deben ser visibles para todos aquellos que lo controlen.
- Inspección. Todos los aspectos del proceso deben ser inspeccionados con la frecuencia necesaria para que las variaciones inaceptables puedan ser detectadas.
- Adaptación. Si después de la inspección se determina que uno o más aspectos están fuera de los límites de aceptación y el producto resultante puede ser inaceptable, se deben hacer los ajustes necesarios al proceso o al producto en cuestión.

Este esquema basa todas estas prácticas en un proceso iterativo e incremental. El resultado de cada iteración es un incremento totalmente funcional de potencial entrega del producto.

Al comienzo de cada iteración, guiada por la lista de requerimientos, el equipo revisa lo que debe hacer. Luego selecciona los requerimientos que cree que puede convertir en un incremento para el término de la iteración y éste realiza su mejor esfuerzo durante el resto de la misma. Al final de la iteración el equipo

presenta el incremento de funcionalidad construido para que aquellos interesados en el producto puedan inspeccionar su funcionalidad.

El equipo determina colectivamente como construir la funcionalidad, modificando el enfoque diariamente a medida que encuentra nuevas complejidades, dificultades y sorpresas. Decide lo que se tiene que hacer y selecciona la mejor manera de hacerlo.

### Proceso

Un proyecto empieza con una visión. El Dueño del Producto es el responsable de transmitir esta visión de una manera que maximice el Retorno de Inversión (ROI) para el cliente, formulando un plan que incluya la lista con los requerimientos que se deben cumplir (Product Backlog).

Durante la primera reunión el Dueño del Producto, explica su visión y le presenta al Gerente de Proyecto la Lista de Requerimientos (Product Backlog), ambos se encargan de asignarle a cada requerimiento su nivel de importancia y nivel de dificultad, dándole prioridad a aquellos que generen mayor valor para la empresa.

Todo el trabajo es hecho en distintas iteraciones. Cada iteración consta de 15 a 30 días consecutivos y se inician con una reunión de planificación, donde el Dueño del Producto, el Gerente de Proyecto y el equipo se reúnen, y en colaboración definen lo que será hecho en la próxima iteración.

Esta reunión consta de dos partes. La primera consiste en que el Dueño del Producto presente al equipo los requerimientos de la lista (Product Backlog). Cuando el equipo tiene toda la información necesaria, selecciona todos los requerimientos que creen que pueden transformar en un incremento completamente funcional del producto al finalizar la iteración.

Durante esta etapa de la reunión también se tiene que establecer la idea inicial de la arquitectura del software, estudiando todos los requerimientos y las necesidades del Dueño del Proyecto, planteando así una idea global del proyecto.

Esta idea global luego será definida en la primera versión del documento de arquitectura escrito por una persona asignada de los integrantes del equipo.

En la segunda parte de la reunión, el equipo planea la iteración junto al Gerente de Proyecto, dividiendo los requerimientos escogidos en las tareas que se deben cumplir para completarlos, y asignándole una persona encargada de resolver cada una de éstas. Luego realizan un estimado del tiempo que les tardará en completar cada tarea, el Gerente de Proyecto debe verificar que las estimaciones estén acordes con la dificultad que fue asignada al requerimiento y reflejada en la Lista (Product Backlog).

Todas las tareas que compongan este plan son colocadas en una Lista de Tareas (Sprint Backlog), junto con la persona encargada y el tiempo estimado.

El Dueño del Producto es responsable de hacer cualquier cambio en la Lista de requerimientos (Product Backlog) actualizándolo, cambiando prioridades o reflejando nuevos requerimientos que surjan en el proceso. Cualquier cambio que se realice será explicado en el comienzo de la siguiente iteración.

No se debe alterar la planificación de la iteración que esté en curso. Si en el transcurso de ésta surgen necesidades de mayor prioridad e importancia para el Dueño del Producto, la iteración puede ser terminada y una nueva iniciada, aunque esto debe evitarse en lo posible.

Al finalizar la segunda parte de la reunión de planificación, teniendo lista la Lista de Tareas (Sprint Backlog), comienzan las actividades de desarrollo. Durante esta etapa se realizan todas las tareas necesarias para obtener el incremento funcional.

Cada día, durante el desarrollo, el equipo se reúne con el Gerente de Proyecto durante 15 minutos aproximadamente. El propósito de esta reunión es sincronizar el trabajo de todos los miembros del equipo diariamente, verificando cual es el avance en las tareas asignadas, lo que tienen planificado hacer durante ese día, si han tenido algún inconveniente que les impida o dificulte la culminación de una tarea y programar cualquier reunión que necesiten para obtener la ayuda necesaria para avanzar.

En el primero o segundo día del desarrollo el equipo deberá presentarle al Dueño del Producto, un prototipo de la interfaz que se utilizará para el incremento. Con esto se podrá verificar que se esté tomando en consideración las funcionalidades esperadas, y que al dueño del producto le agrade visualmente el incremento que le será entregado.

Cada integrante del equipo es responsable de escribir las pruebas, código, y documentación de las tareas que le fueron asignadas, realizando su integración a medida que complete cada una de ellas, verificando que funcionen correctamente resolviendo cualquier error que se le presente.

Para el desarrollo de cada incremento funcional el equipo utilizará Test Driven Development (TDD), por lo que cada integrante comenzará por escribir pruebas automatizadas para los métodos que luego va a implementar, con la finalidad de completar cada tarea.

El equipo realizará integración continua. A medida que se vayan completando las tareas, éstas deben ser integradas, corriendo todas las pruebas existentes hasta el momento para verificar que no se generó ningún problema.

Se utilizarán herramientas que permiten el control o manejo de varias versiones del proyecto. Éstas también son beneficiosas en el momento de la integración, ya que monitorean cualquier conflicto que se pueda generar entre la versión existente y los cambios que se están tratando de introducir, generando

una alerta y mostrando los lugares en los que se produjo el conflicto, de manera que el programador los pueda solucionar.

Para llevar un control general del proyecto y mantener los estándares de codificación se deberá hacer revisión de código, asegurándose de que esté bien estructurado.

Luego de completadas todas las tareas, se realiza una última integración, corriendo todas las pruebas que fueron diseñadas para éstas, verificando que todo funcione correctamente.

Al finalizar el Sprint se lleva a cabo una reunión de revisión. En ésta el equipo presenta al Dueño del Producto, y a los interesados que quieran asistir, lo que han desarrollado durante la iteración, incluyendo la lista con las funcionalidades del incremento. Éste revisa el trabajo entregado y da su opinión. Si algún requerimiento no pudo ser realizado, se puede incluir en cualquiera de las iteraciones siguientes.

Luego de la revisión, y antes de la próxima reunión de planificación, el Gerente de Proyecto sostiene una reunión de retrospectiva con el equipo, donde los alienta a revisar su proceso de desarrollo para hacerlo más efectivo y agradable para la próxima iteración.

## Roles

### *Dueño del Producto*

El Dueño del Producto es el responsable de representar los intereses de todos aquellos que tengan cierto interés en el proyecto y el sistema resultante, definiendo los requerimientos iniciales del proyecto, los objetivos de retorno de inversión (ROI) y el plan de entregas.

También es responsable de realizar y usar la Lista de Requerimientos (Product Backlog) para asegurar que las funcionalidades de mayor valor sean cubiertas primero. Esto lo logra revisando

frecuentemente la prioridad reflejada en la Lista, y modificándolo de manera que asegure que los requerimientos de mayor valor sean trabajados en la próxima iteración.

### *Gerente de Proyecto*

Es responsable de manejar el proceso general, de saber sus prácticas, reglas, productos, terminología y como aplicar todo correctamente.

Debe mantener comunicación continua con el Dueño del Producto, ayudándolo a entender mejor los aspectos del desarrollo, para que éste lo pueda dirigir, sin perjudicar el flujo de trabajo natural del proceso.

Éste asiste al Dueño del Producto durante la primera reunión para verificar que la Lista de Requerimientos (Product Backlog) refleje lo requerido para agregar valor, pero tomando en consideración los aspectos de desarrollo.

Es responsable de dirigir la creación de la lista de funcionalidades que es entregada al final de cada iteración junto con el incremento funcional.

También debe mantener el balance entre la responsabilidad y el bienestar del equipo, enfocándose siempre en obtener los mejores resultados para el proyecto. Por lo que debe tomar en cuenta las diferentes formas de facilitar la creatividad y mejorar la productividad de los integrantes del equipo.

### *Equipo*

El equipo de trabajo está conformado, preferiblemente, por un grupo pequeño de personas, de manera que se haga más fácil manejar el proyecto y que se puedan cumplir y llevar a cabo las actividades organizadamente.

Es responsable de manejar y realizar las actividades del desarrollo. Siendo así un equipo auto-gerenciado, auto-organizado y multifuncional.

Los integrantes del equipo están encargados de averiguar cómo convertir la Lista de Requerimientos (Product Backlog) en un incremento de funcionalidad en cada iteración. Por lo que deben llegar a un acuerdo con el Dueño del Producto de los requerimientos que son capaces de desarrollar durante una iteración.

Se encargan de dividir los requerimientos, siguiendo las especificaciones anotadas, con la ayuda del Gerente de Proyecto, en las distintas tareas que los conforman y que deben ser cubiertas para cumplir con la funcionalidad deseada. Para cada tarea realizan estimaciones del tiempo que piensan que les tomará desarrollarla.

Dentro del equipo se pueden definir distintos roles, para llevar un mejor control y seguimiento de las tareas que se están realizando. Estos roles pueden ser rotados entre los integrantes del equipo durante las distintas iteraciones, tomando en cuenta su nivel de experiencia. Algunos de estos son:

- Probador:

  Verifica que todos los integrantes del equipo estén escribiendo pruebas antes de escribir el código funcional, y los ayuda en caso de ser nuevos ante esta práctica, por lo que es recomendable que la persona que asuma este cargo tenga experiencia.

  Se encarga de correr todas las pruebas escritas continuamente para verificar que no se haya producido algún error durante la integración. Debe hacer llegar los resultados de las pruebas a todos los integrantes del equipo.

  También se encarga de escribir pruebas funcionales, tomando en consideración las especificaciones dadas por el Dueño del Producto.

- Verificador:

Se encarga de revisar las estimaciones hechas por el equipo, comparándolas con el progreso real del proyecto, y mostrarlo al Gerente del Proyecto de manera que se pueda hacer seguimiento y recomendaciones para cumplir con lo planteado, o decidir si se tiene que tomar alguna otra medida.

Si al finalizar una iteración las estimaciones fueron muy altas o muy bajas, debe mostrarle esto al equipo y hacer las recomendaciones pertinentes para mejorar en la siguiente iteración.

- Encargado de Interfaz:

Éste realiza el prototipo de interfaz que se muestra al Dueño del Producto al comienzo de la iteración.

Debe hacer el diseño general de la interfaz, tomando en cuenta los requerimientos que se quieren cumplir, y las necesidades planteadas durante la reunión de planificación.

Este rol puede se asignado a mas de una persona, dependiendo del número de integrantes del equipo.

- Arquitecto:

Encargado de escribir el documento de arquitectura del software, debe considerar todo lo discutido durante la reunión de planificación, por lo que es recomendable que realice anotaciones de lo acordado durante ésta.

Realiza los cambios correspondientes que vayan surgiendo a medida que se desarrolla el producto. Manteniendo así el documento al día. Y asesora a los demás integrantes en caso de que surja alguna duda especifica de como se integra la tarea a la idea global del proyecto.

Estos roles pueden ser cambiados, o se pueden agregar nuevos roles, dependiendo de las necesidades del proyecto y del tamaño del equipo. También se puede asignar más de un rol a una persona.

## Flujo de Trabajo

### *Actividades de Planificación*

*Creación o actualización de la Lista de Requerimientos (Product Backlog)*

Comenzando la reunión el Dueño del Producto le presenta al Gerente de Proyecto, la visión del proyecto y el valor que espera que éste agregue a su organización.

Luego de que el Gerente de Proyecto maneje la visión general del proyecto y el camino que se desea tomar, comienzan la elaboración de la lista de requerimientos funcionales y no funcionales (Product Backlog), con los que debe cumplir el producto. A cada requerimiento le es dado un nivel de importancia que establece su prioridad según el valor que generen a la organización, y también se le asigna un nivel de dificultad con lo que el Gerente de Proyecto podrá ayudar al equipo a la hora de realizar las estimaciones de tiempo sobre las tareas de cada requerimiento.

Esta lista debe ser actualizada constantemente a medida que avance el proyecto, surjan nuevos requerimientos o cambien las prioridades de éstos. El Dueño del Producto podrá hacer cualquier cambio que desee, los cuales serán explicados en la siguiente reunión de planificación, siempre y cuando no afecten la iteración en curso.

*Reunión de Planificación*

La reunión de planificación es llevada a cabo al comienzo de cada iteración y luego de que el Dueño del Producto y el Gerente de Proyecto tengan definida la Lista de Requerimientos (Product Backlog); ésta consta de dos partes, la primera se basa en elegir los requerimientos y la segunda en crear la Lista de Tareas (Sprint Backlog).

*Primera Parte – Elegir requerimientos*

El Dueño del Producto y Gerente de Proyecto se reúnen con el Equipo para presentarles la visión del proyecto y la Lista de Requerimientos (Product Backlog) previamente elaborada, asegurando que todos los puntos queden cubiertos y claros para los miembros del equipo.

Al discutir cada requerimiento de la Lista, se van detallando las especificaciones que deben ser cubiertas para poder cumplir con lo planteado. Todas estas especificaciones son anotadas por el equipo para utilizarlas en la segunda parte de la reunión al definir las tareas.

Luego de aclarar todas las dudas, el equipo de trabajo llega a un acuerdo con el Dueño del Producto de que requerimientos pueden ser convertidos en un incremento funcional para el final de la iteración. Con los requerimientos escogidos se crea la Lista de Funcionalidades.

Para elegir los requerimientos que van a ser elaborados, el equipo debe tomar en cuenta la dificultad y la prioridad que tiene cada uno de éstos, siempre enfocados en el objetivo de que tienen que ser completados en el lapso de tiempo de la iteración.

Al tener claros los requerimientos, se debe definir el boceto inicial de la arquitectura del proyecto, generando una idea global de cómo se va a desarrollar, y como será conformado estableciendo las

relaciones entre los distintos requerimientos. Esta idea global luego será definida en la primera versión del documento de arquitectura del software.

*Segunda Parte - Creación de lista de Tareas (Sprint Backlog)*

Al quedar definidos los requerimientos que se van a cumplir durante la iteración, el Equipo y el Gerente de Proyecto se reúnen y desglosan cada uno de estos requerimientos en las tareas que deben ser realizadas para cumplir el objetivo de la iteración.

A cada tarea se asigna un responsable, y éste realiza una estimación del tiempo que cree necesario para poderla cumplir, el Gerente de Proyecto debe tomar en cuenta la dificultad que se le asigno al requerimiento y guiar a la persona si es necesario.

Al terminar las estimaciones se obtiene una lista de tareas, que contiene las personas encargadas de realizarlas y el tiempo estimado en el cual serán completadas.

*Boceto inicial del documento de arquitectura del software*

Un integrante del equipo debe generar la primera versión del documento de arquitectura del software para el proyecto. Escribiendo de forma sencilla y fácil de entender todo lo acordado, sobre este aspecto, durante la reunión de planificación.

La creación de este documento debe hacerse de manera rápida, de forma que esté listo al comenzar la primera iteración.

Este documento es modificado a medida que se realiza el desarrollo o antes de comenzar cada iteración.

## Actividades de Desarrollo

### *Reunión diaria*

Durante el desarrollo el Gerente de Proyecto realiza reuniones diarias con el equipo de unos 15 minutos aproximadamente. Éstas se llevan a cabo al comienzo del día y no deben ser muy largas ya que están programadas para llevar el control del progreso del proyecto, y no para resolver problemas individuales.

Cada miembro del equipo le informa al Gerente de Proyecto el progreso en las tareas de las cuales es responsables, los objetivos a cumplir en el transcurso del día y de haberlos, los obstáculos y dificultades encontradas.

Si algún integrante presenta problemas en los que necesita ayuda, el Gerente de Proyecto convocará una reunión, luego de finalizar la reunión diaria, con los miembros necesarios para resolver dichas dificultades, permitiendo así que los demás integrantes comiencen con sus objetivos planteados.

### *Creación del Prototipo de Interfaz*

En el primer o segundo día del desarrollo, el equipo debe diseñar y presentarle al Dueño del Producto un prototipo de la interfaz que se utilizará para el incremento funcional que será entregado al finalizar la iteración. Con esta entrega se realizará una verificación inicial de que se estén tomando en consideración las funcionalidades que se esperan para los requerimientos escogidos.

Este prototipo debe cubrir las funcionalidades de los requerimientos que fueron escogidos y se debe realizar en paralelo con las tareas que le fueron asignadas, de forma de no retrasar la iteración.

*Test Driven Development (TDD)*

Durante el desarrollo el equipo utilizará Test Driven Development (TDD), siendo ésta una práctica que consiste en escribir pruebas antes de escribir el código funcional, tomando en cuenta los diferentes métodos que serán implementados luego.

En esta técnica de desarrollo se cumplen las siguientes etapas:

- Escribir una prueba: los programadores escriben una prueba para la tarea que están desarrollando en ese momento.
- Compilar: luego de que la prueba ha sido escrita, es compilada para asegurarse de que no existan errores. Debido a que no se ha escrito código del método que se está desarrollando, la compilación fallará.
- Corregir los errores de compilación escribiendo suficiente código para que la prueba compile: el programador escribe suficiente código del método que está implementando para que corra sin problemas.
- Correr la prueba y verla fallar: Luego de que se escribe suficiente código para poder compilar, se corre la prueba, debido a que sólo se escribió una pequeña parte del código con la finalidad de que compilara, la prueba fallará.
- Escribir más código para que pase la prueba: Luego de ver fallar la prueba, el programador escribe más código para completar la funcionalidad del método que está implementando.
- Correr la prueba y verla pasar: al terminar de escribir el código, el programador corre la prueba, la que debería pasar sin problemas.

- Refactorizar el código: cuando la prueba pase sin problemas, procederá a refactorizar con la finalidad de asegurarse que el código que escribió no presente repeticiones y esté lo mas simplificado posible.
- Comenzar con la siguiente prueba: al finalizar la refactorización, comienza de nuevo en la primera etapa, escribiendo una prueba para la siguiente funcionalidad.

Las personas encargadas de escribir estas pruebas son los programadores, que deben tener claras las especificaciones y los requerimientos de las tareas que están realizando para esta iteración.

### *Integración Continua*

La integración continua es un proceso en el que se une el código escrito por los distintos integrantes del equipo. Este proceso permite verificar que los cambios realizados no producen errores.

A medida que se van terminando las tareas deben ser integradas, verificando siempre que no se produzcan errores de compilación y que el código pase todas las pruebas escritas hasta el momento sin ningún problema.

Para controlar la integración continua, y los posibles errores que pueden surgir cuando un grupo de personas está realizando modificaciones y uniones bajo un mismo proyecto, se debe utilizar control de versiones.

### *Control de versiones*

El control de versiones permite mantener un historial del código fuente, teniendo así un control sobre todos los cambios realizados.

Los sistemas encargados de esto permiten a más de un programador editar sobre el mismo código fuente simultáneamente.

Cada programador tendrá una copia local de la última versión del proyecto, trabajando sobre ésta, al finalizar su trabajo éste realizará "commit", subiendo así los cambios al controlador de versiones. Si se presentan conflictos entre su trabajo y los cambios que realizó otro programador sobre ese código fuente, el sistema alertará y mostrará los lugares donde se está produciendo el problema, de manera que se puedan hacer comparaciones y solucionar el conflicto.

Esta forma de manejar las distintas versiones y de alertar ante los conflictos, ayuda a los programadores durante la integración del código.

### *Revisión de Código*

Durante la revisión de código se revisa todo lo escrito por algún miembro del equipo, asegurando que esté bien estructurado. Para esta práctica se pueden utilizar herramientas de control, facilitando la revisión.

Si se encuentra algún error, éste será corregido antes de comenzar con los objetivos planteados, evitando generar problemas futuros.

## Actividades de Revisión

### *Reunión de Revisión*

Esta reunión es llevada a cabo al final de cada iteración. En ella, el equipo, junto al Gerente de Proyecto, le presentan el incremento funcional desarrollado durante la iteración y la lista de funcionalidades al Dueño del Producto, donde éste da sus críticas y observaciones al respecto.

### *Reunión de Retrospectiva*

Ésta, al igual que la reunión de entrega, es llevada a cabo al final de cada iteración. En ella el Gerente de Proyecto se reúne con el Equipo para discutir todo el proceso del ciclo culminado. Cada persona da su opinión de lo que se debería mejorar para hacer el proceso más efectivo y agradable para todos.

El Gerente de Proyecto deberá tomar en consideración todas las opiniones para la siguiente iteración, con el fin de mejorar el rendimiento del equipo y evitar repetir situaciones que retrasen o no ayuden al proceso.

## Artefactos

### *Lista de Requerimientos (Product Backlog)*

Lista dinámica con los requerimientos funcionales y no funcionales del producto. Sus cambios deben ser administrados constantemente para identificar que necesita el producto para ser apropiado, competitivo y útil.

En ésta podemos registrar el nombre del ítem, su nivel de importancia o prioridad ante los demás requerimientos y su factor de complejidad. También se debe registrar la iteración en el cual el ítem es desarrollado.

Todos los requerimientos reflejados en la Lista, son agregados por el Dueño del Producto, manteniendo siempre en cuenta la visión general del proyecto y asegurándose de que cumplan un valor para la empresa.

Para establecer la prioridad y el factor de complejidad, el Dueño del Producto es asesorado por el Gerente de Proyecto, de forma tal que se consideren tanto los aspectos importantes para la empresa como los aspectos del desarrollo en sí.

### *Documento de especificación de requerimientos del software*

Cada requerimiento de la Lista tiene especificaciones que definen lo que el Dueño del Producto realmente quiere o necesita. Estas especificaciones son discutidas durante la primera parte de la reunión de planificación.

El Equipo anota cada una de estas especificaciones y luego las revisa para hacer la Lista de Tareas.

### *Lista de Funcionalidades (Release Backlog)*

Los requerimientos escogidos para la iteración son escritos en la lista de funcionalidades.

Con esta lista se verificará que los requerimientos planteados fueron cumplidos. y será actualizada en el caso de que no se cumpla algún requerimiento. También le permite al Dueño del Producto saber que funcionalidades tiene el incremento que le fue entregado.

*Lista de Tareas (Sprint Backlog)*

Contiene las tareas que el equipo define para transformar los requerimientos seleccionados para la iteración en un incremento funcional. Estas tareas deben ser divididas de manera tal que tome entre 4 y 16 horas terminar cada una de ellas.

Esta lista solo puede ser manipulada por los integrantes del equipo. En la primera columna se reflejaran las tareas que se van a realizar, en la segunda el nombre del encargado de esta tarea y luego una columna para cada día de la iteración.

En la intersección de la tarea y el primer día de la iteración, se colocará el tiempo total estimado que se tardará en completar la tarea. Durante cada día de la iteración se irá colocando el número de horas implementadas en la tarea.

*Documento de Arquitectura del software*

Documento que define la arquitectura del proyecto. Éste será una idea global de las tareas que se deben llevar a cabo para asegurar que se cumplan todos los requerimientos.

Esta arquitectura ira evolucionando a medida que se vaya desarrollando el proyecto, creciendo de manera incremental, y así cubrir cualquier cambio que se haya encontrado durante el desarrollo.

El documento debe ser escrito de manera sencilla y fácil de leer; facilitando el agregar cosas nuevas o realizar cambios a medida que vayan surgiendo.

### *Prototipo de Interfaz*

Al comienzo de las actividades de desarrollo se mostrará al Dueño del Producto un prototipo de la interfaz que se utilizará para el producto.

Este prototipo cumplirá con los requerimientos planteados durante la reunión de planificación, mostrando todas las funcionalidades que serán desarrolladas y el diseño de la interfaz (colores, letra, distribución, navegación, etc.).

### *Pruebas*

Una de las prácticas que el Equipo utilizará es TDD, ésta consiste en escribir pruebas antes de escribir el código funcional.

Se escribirán pruebas para cada uno de los métodos que luego serán implementados, tomando en cuenta los requerimientos y las necesidades que se tienen que cumplir para cada tarea.

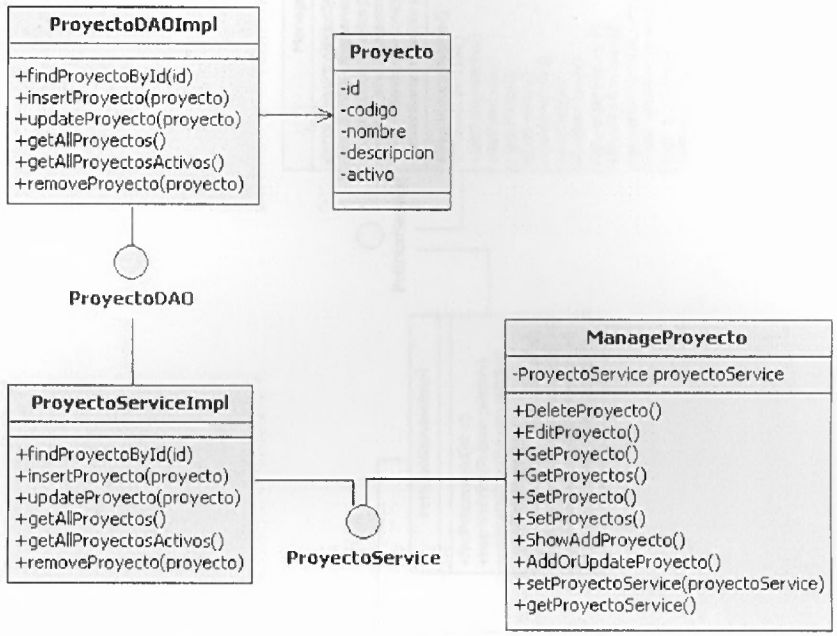
Éstas serán automatizadas, de manera que se puedan correr cada vez que se realiza integración de código, para comprobar que no se ha generado ningún problema. Con esto se verificará que el código compile y funcione de manera correcta, además se escribirán pruebas funcionales con las que se verá si los resultados son los esperados por el Dueño del Proyecto.

### *Incremento funcional (Release)*

Al final de cada iteración se entregará al Dueño del Producto un incremento del producto, que debe cumplir con todos los requerimientos que fueron acordados durante la reunión de planificación.

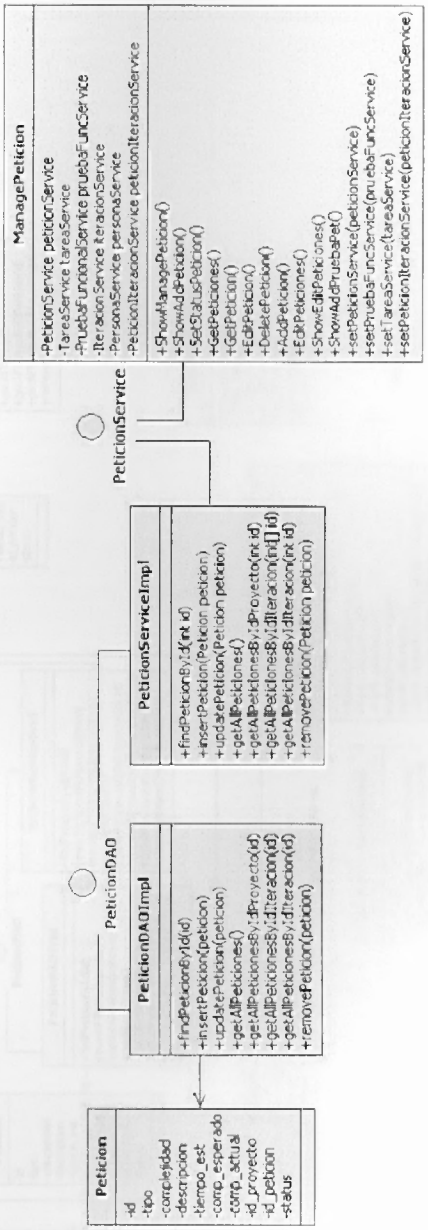
Este incremento del producto debe ser funcional, probado, bien estructurado, y con su operatividad para el lado del usuario totalmente documentada, así sea en archivos de ayuda o en documentación de usuario.

## Apéndice D. Diagramas de Clases

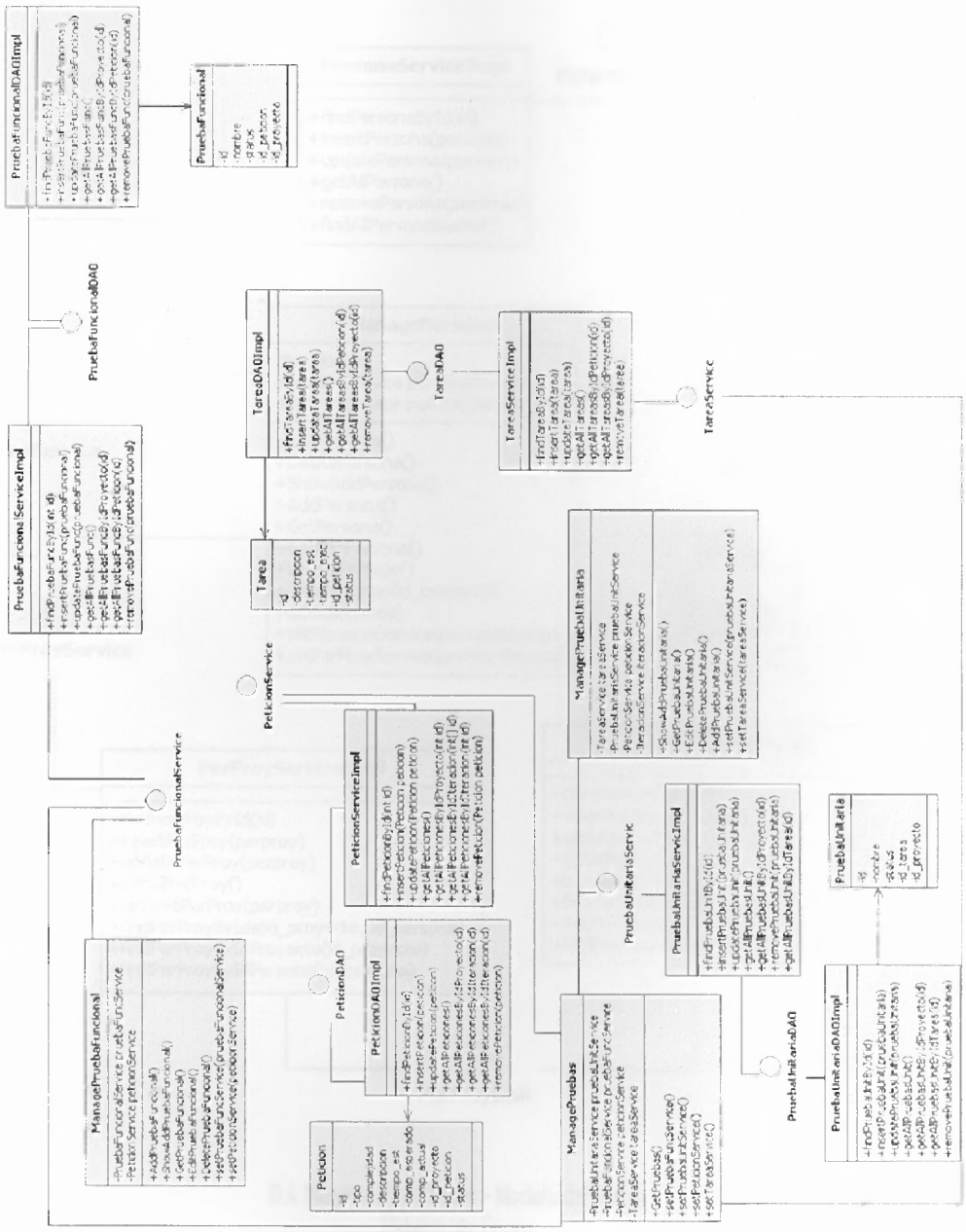



D.1 Diagrama de Clases - Modulo de Proyecto

Elaboración Propia

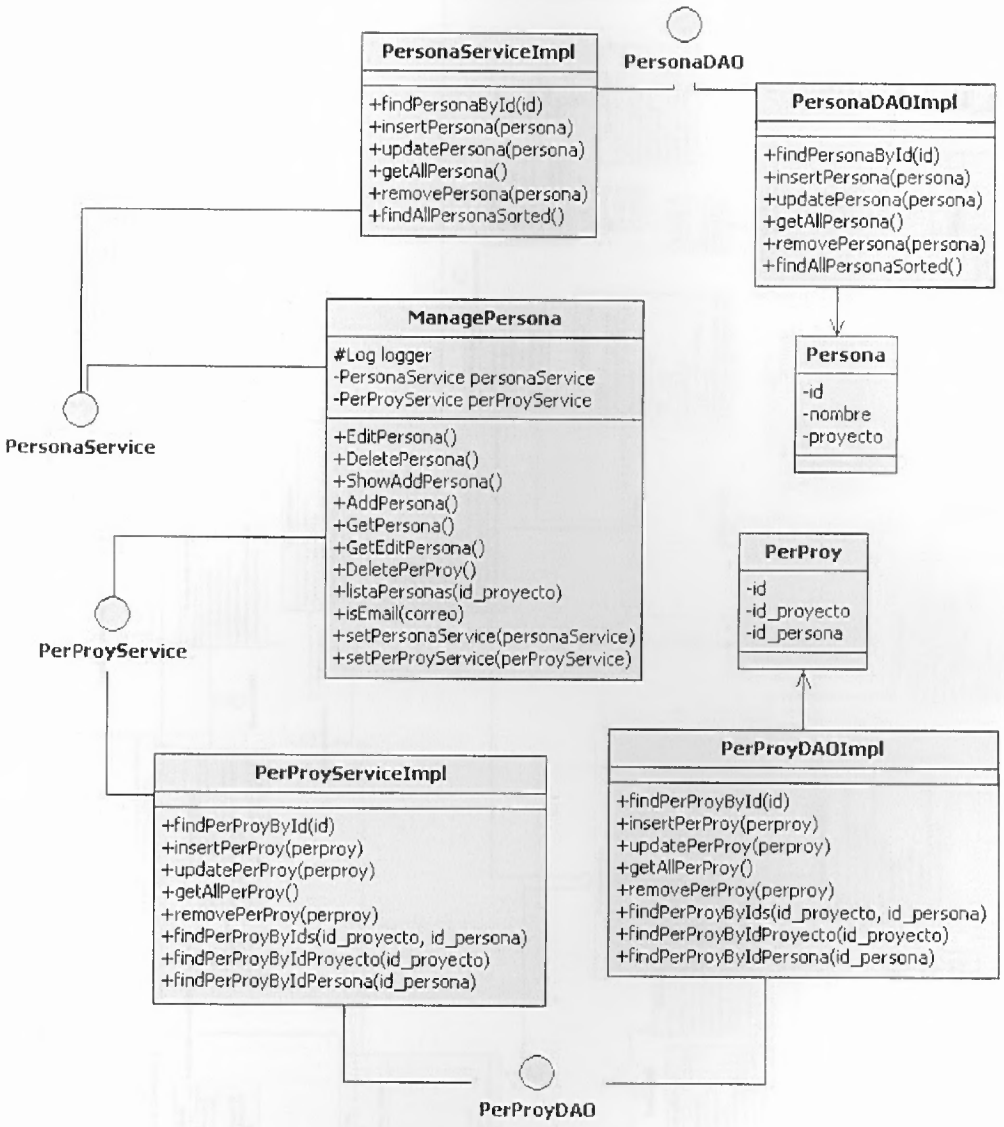



D.2 Diagrama de Clases - Modulo de Peticiones
Elaboración Propia



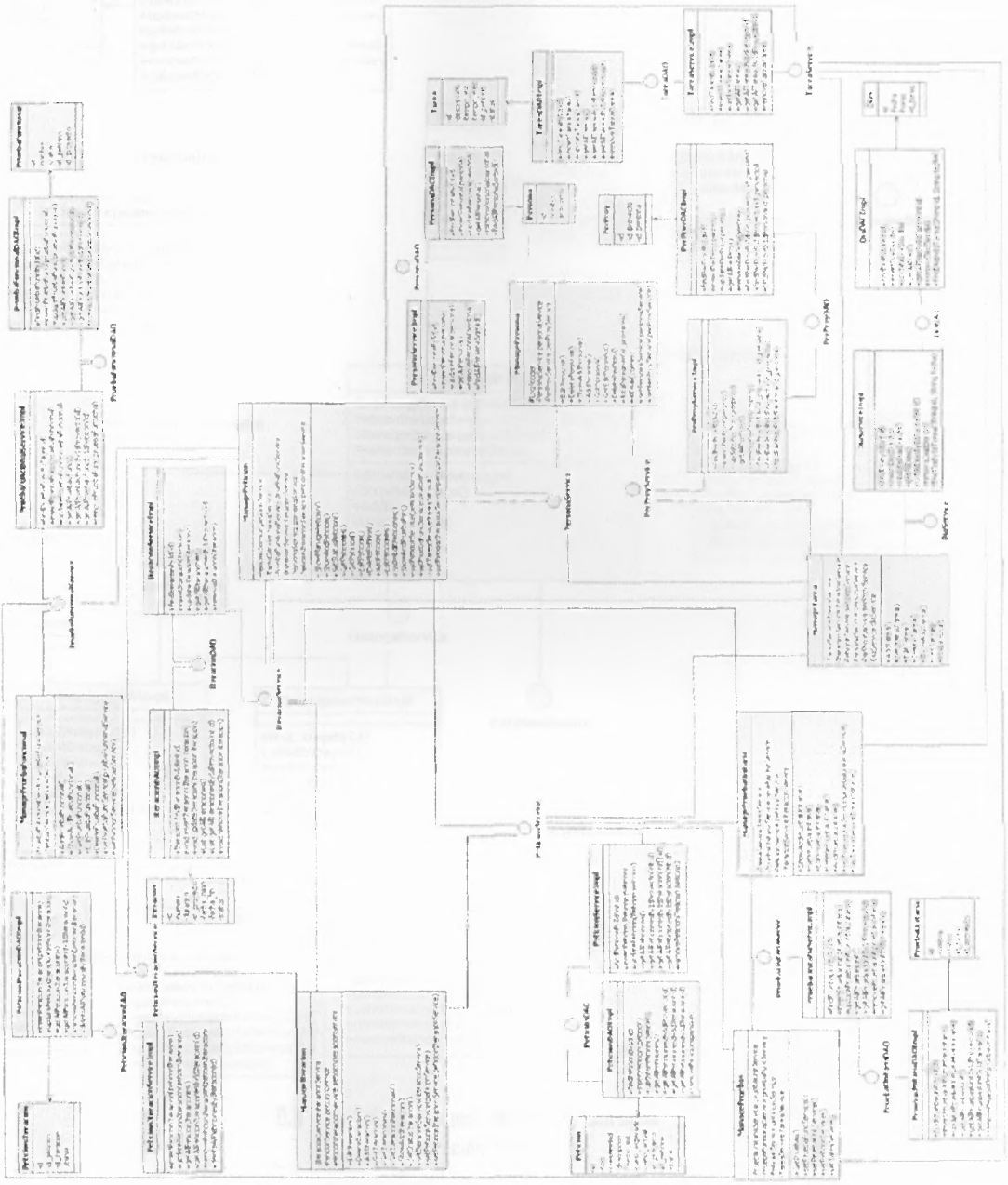

**D.3 Diagrama de Clases - Modulo de Pruebas**

Elaboración Propia

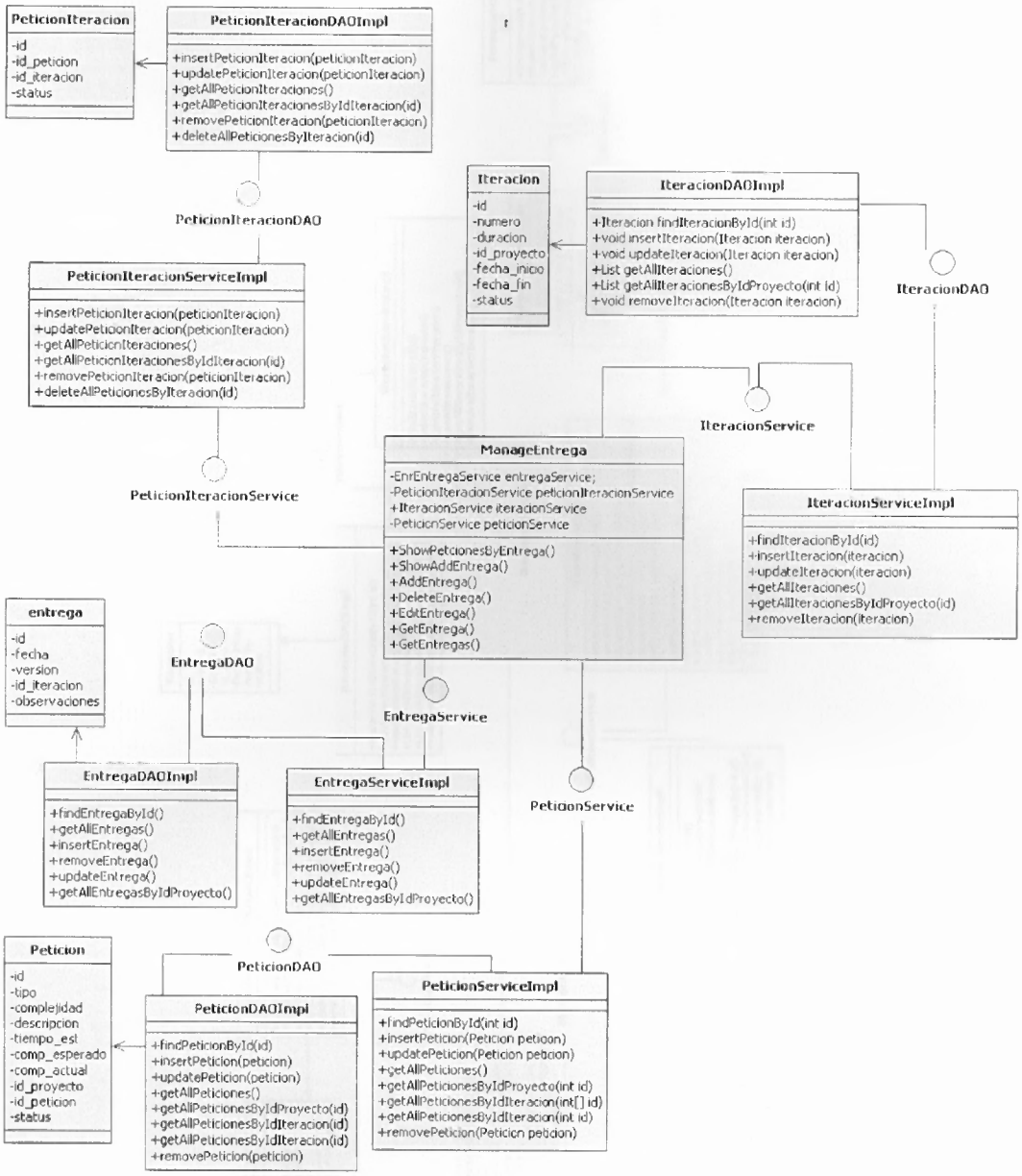



0.4 Diagrama de Clases - Modulo de Equipo

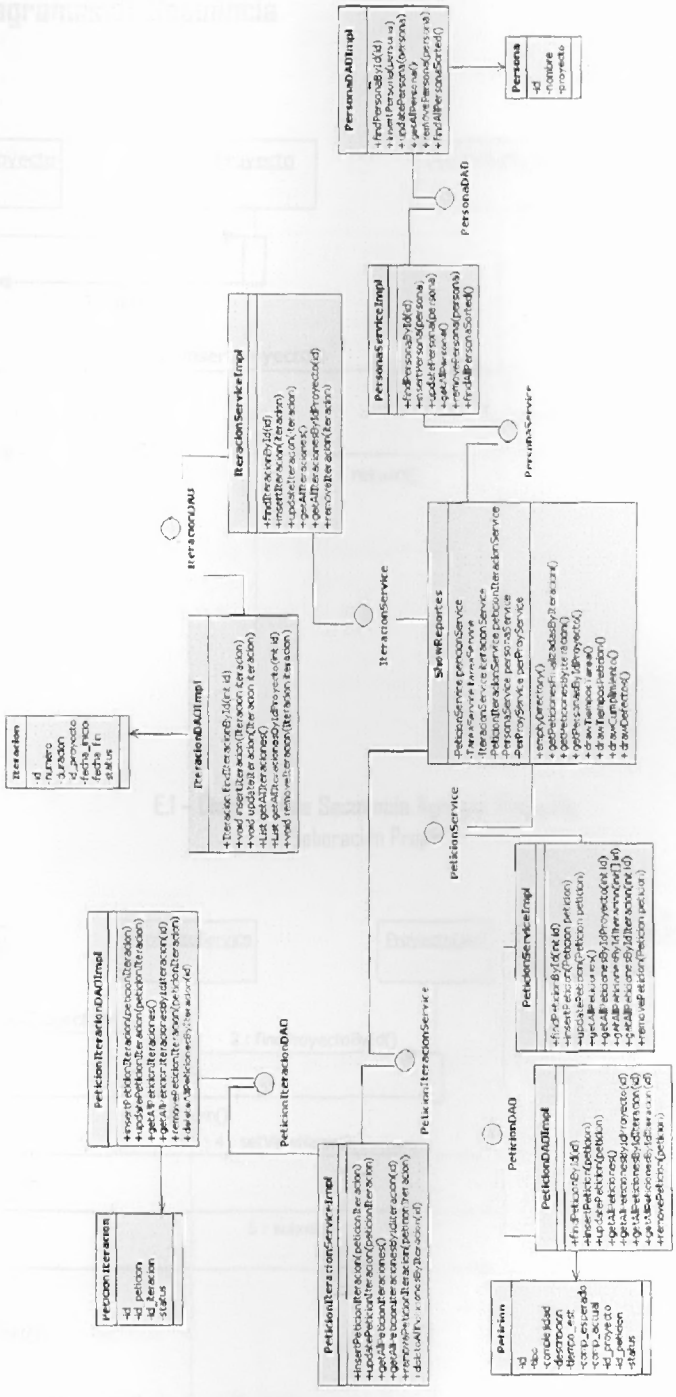
Elaboración Propia



**D.5 Diagrama de Clases - Modulo de Planificación**
Elaboración Propia

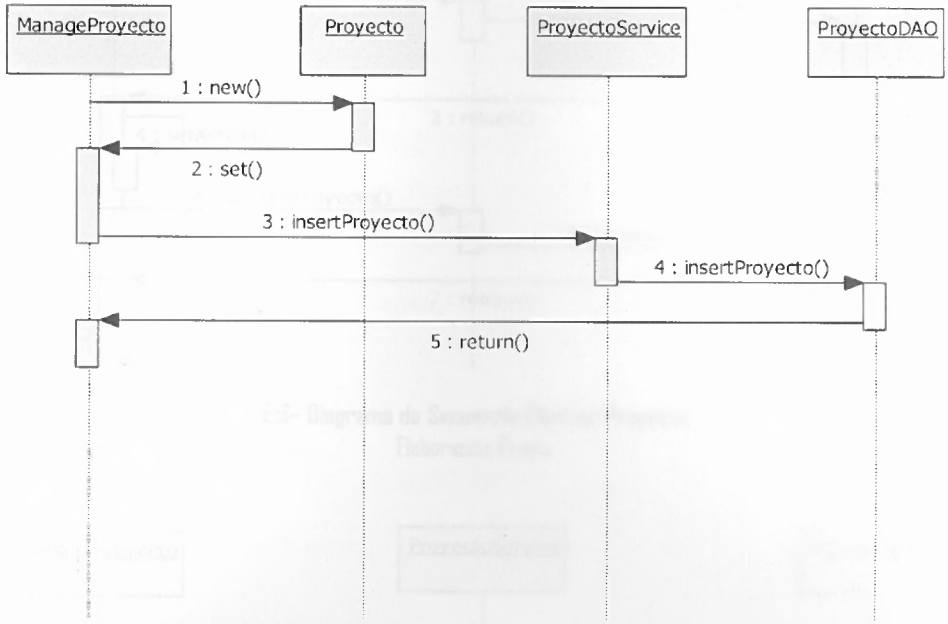



D.6 Diagrama de Clases - Modulo de Entregas
Elaboración Propia

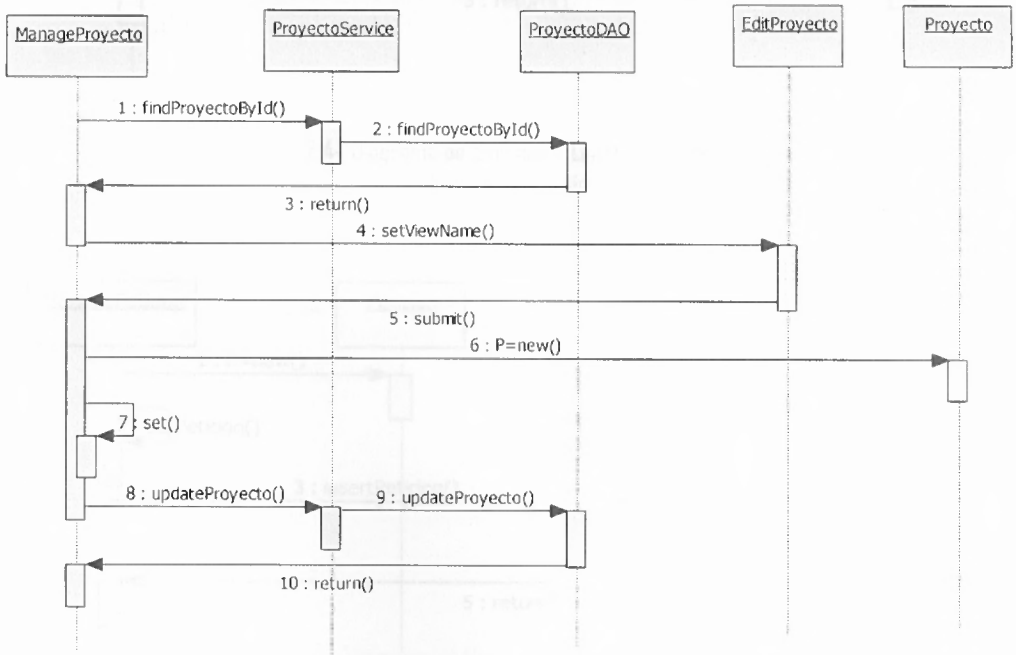


0.7 Diagrama de Clases - Modulo de Reportes
Elaboración Propia

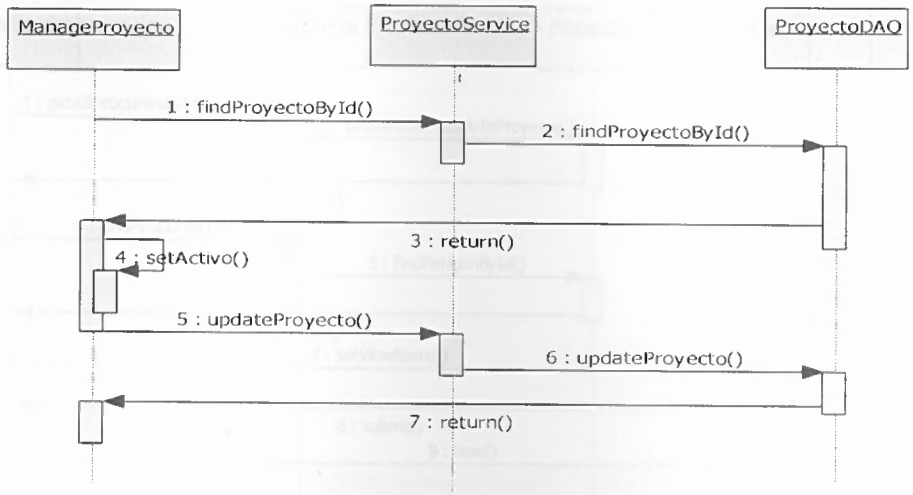
## Apéndice E. Diagramas de Secuencia



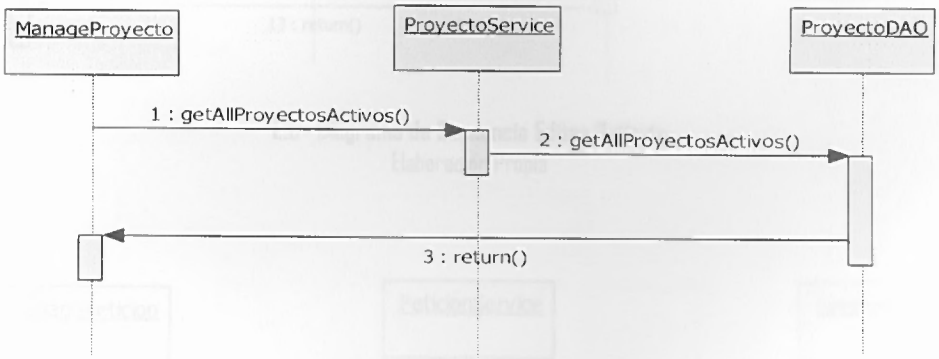

**E.1 – Diagrama de Secuencia Agregar Proyecto**
Elaboración Propia



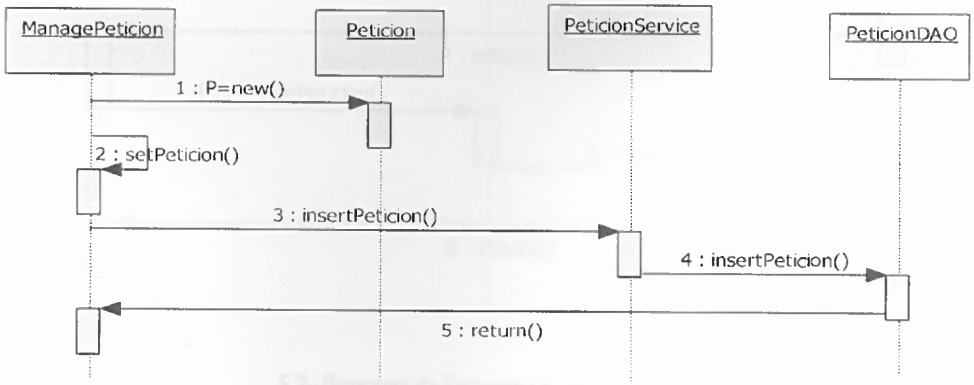

**E.2– Diagrama de Secuencia Editar Proyecto**
Elaboración Propia



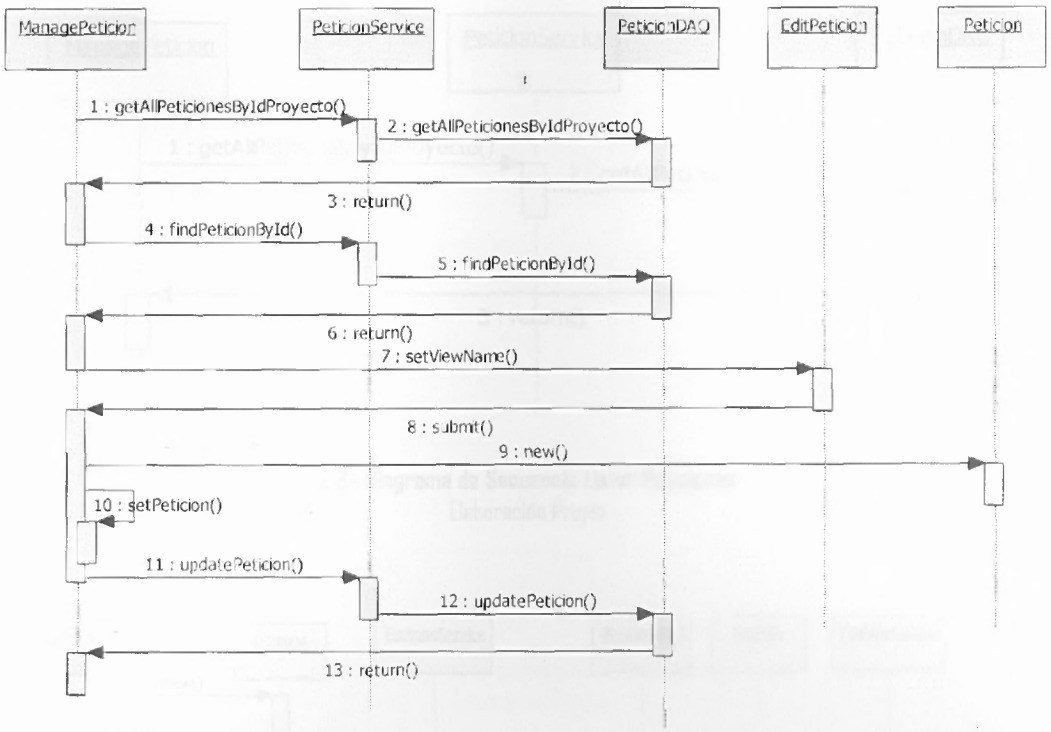

**E.3- Diagrama de Secuencia Eliminar Proyecto**
Elaboración Propia



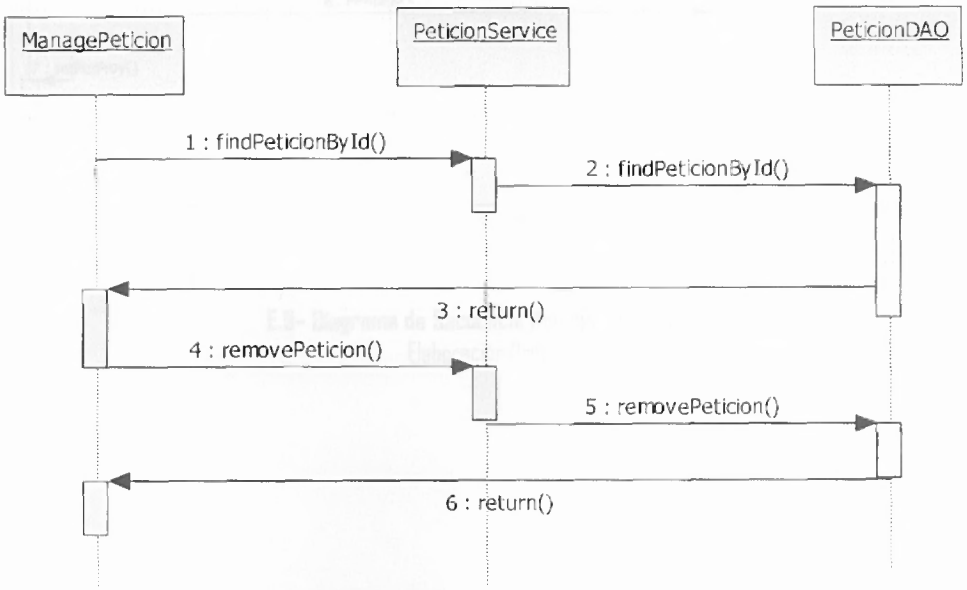

**E.4- Diagrama de Secuencia Listar Proyecto**
Elaboración Propia



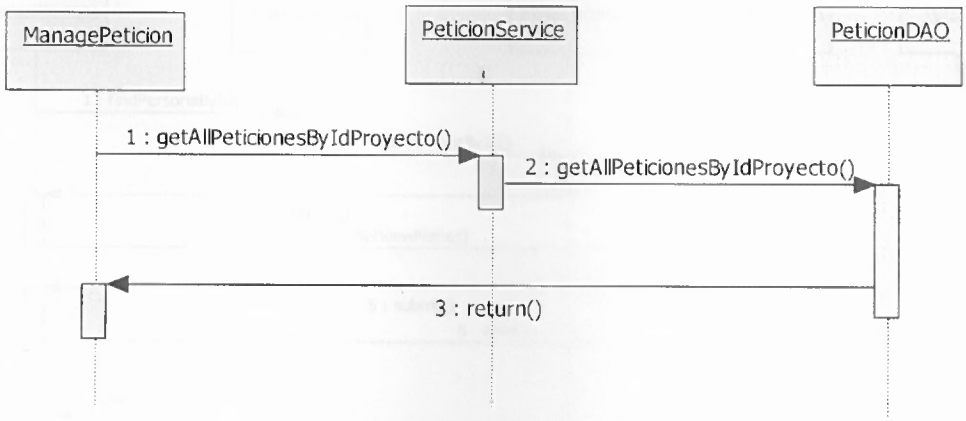

**E.5- Diagrama de Secuencia Ingresar Petición**
Elaboración Propia



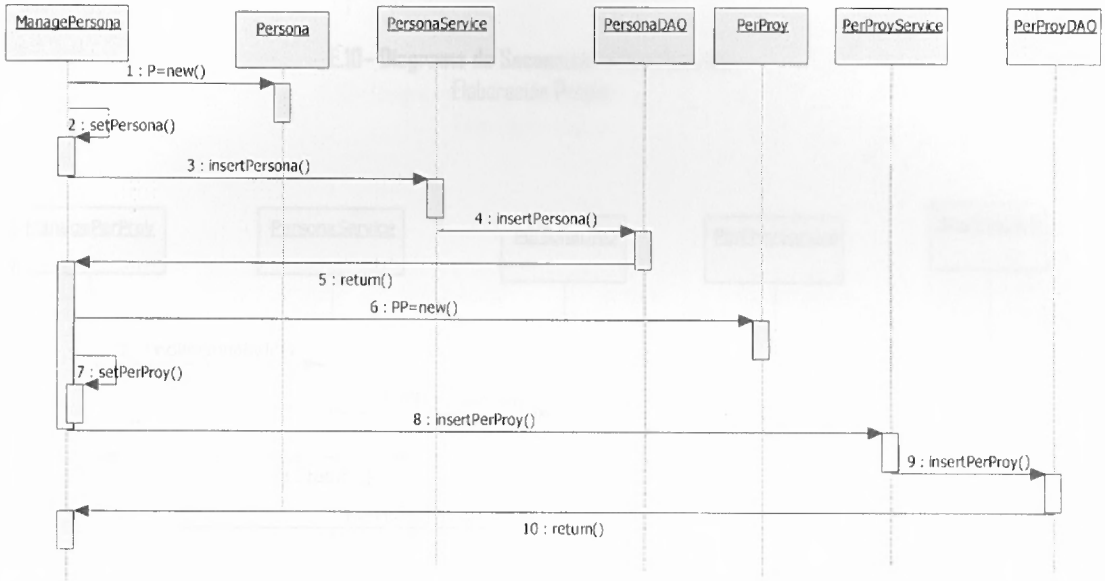

**E.6- Diagrama de Secuencia Editar Petición**
Elaboración Propia



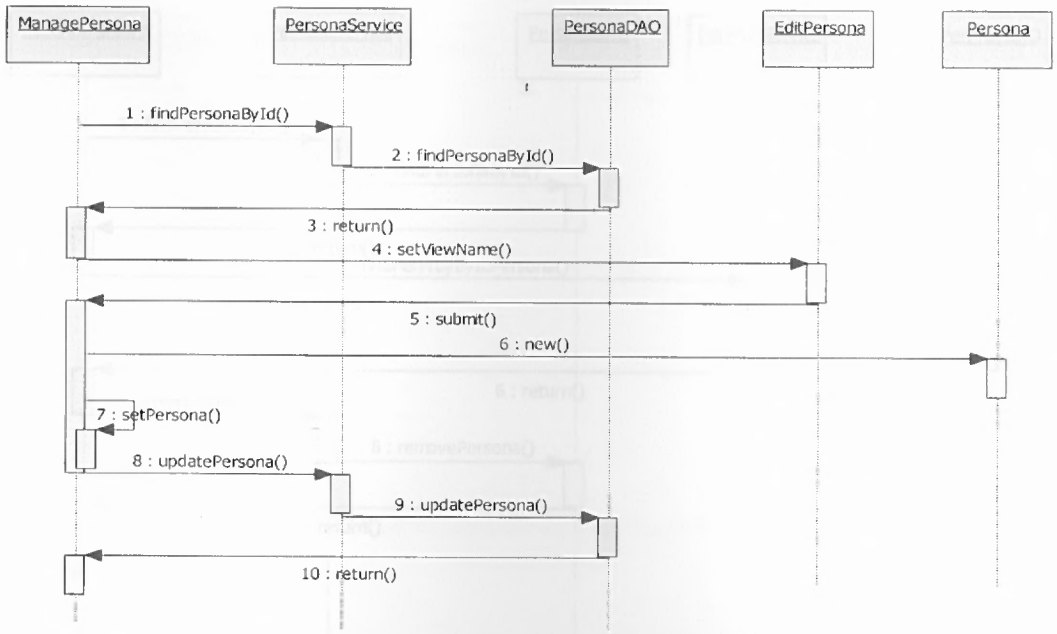

**E.7- Diagrama de Secuencia Eliminar Petición**
Elaboración Propia



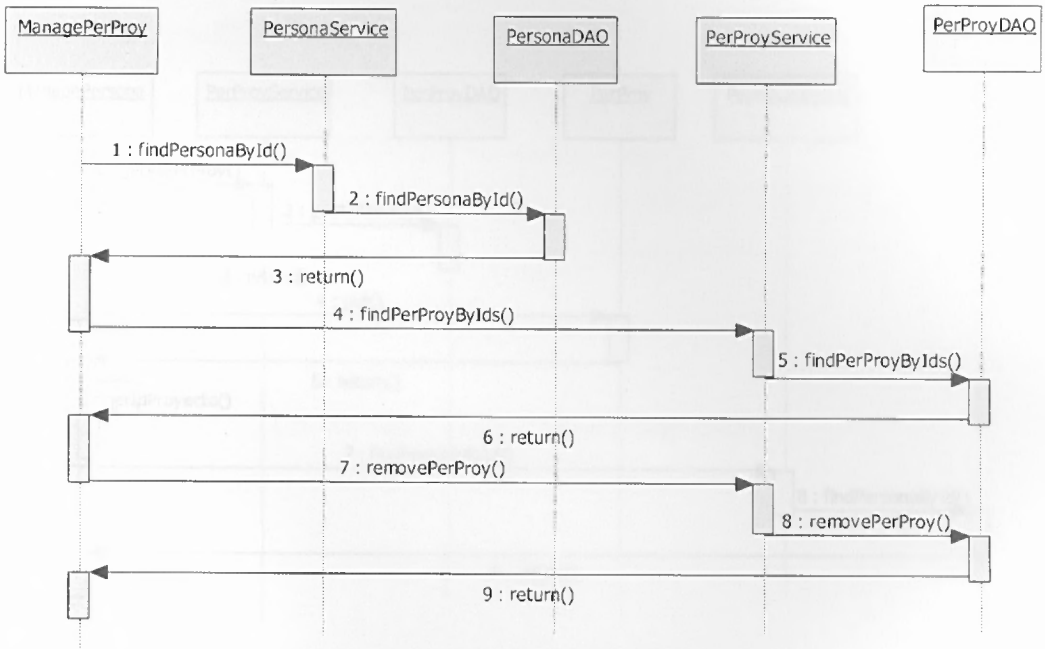

**E.8- Diagrama de Secuencia Listar Peticiones**
Elaboración Propia



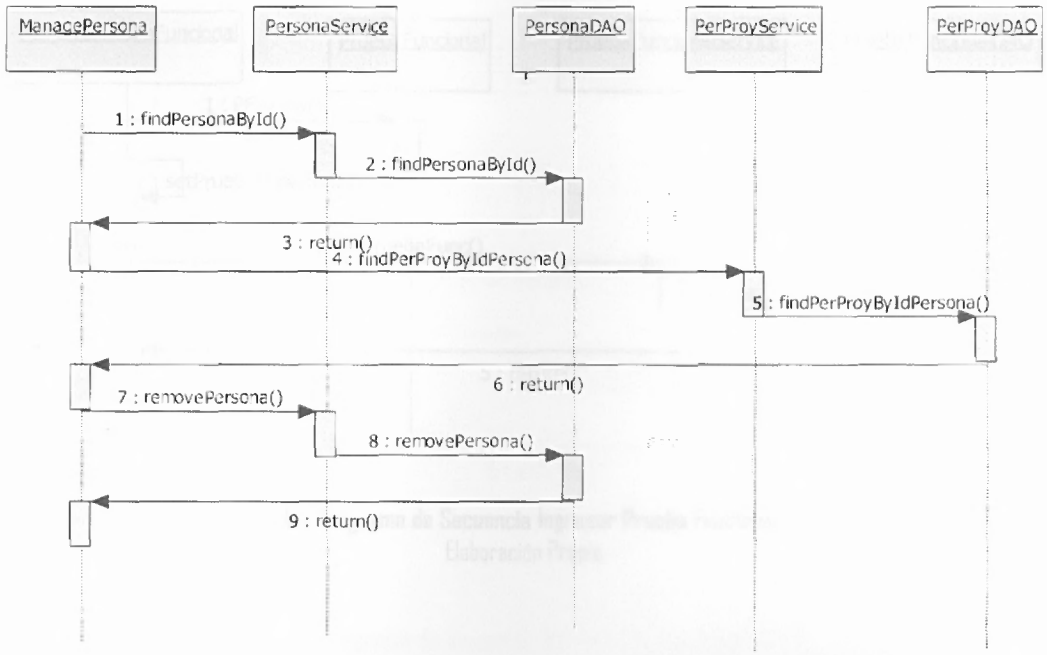

**E.9- Diagrama de Secuencia Agregar Integrante**
Elaboración Propia



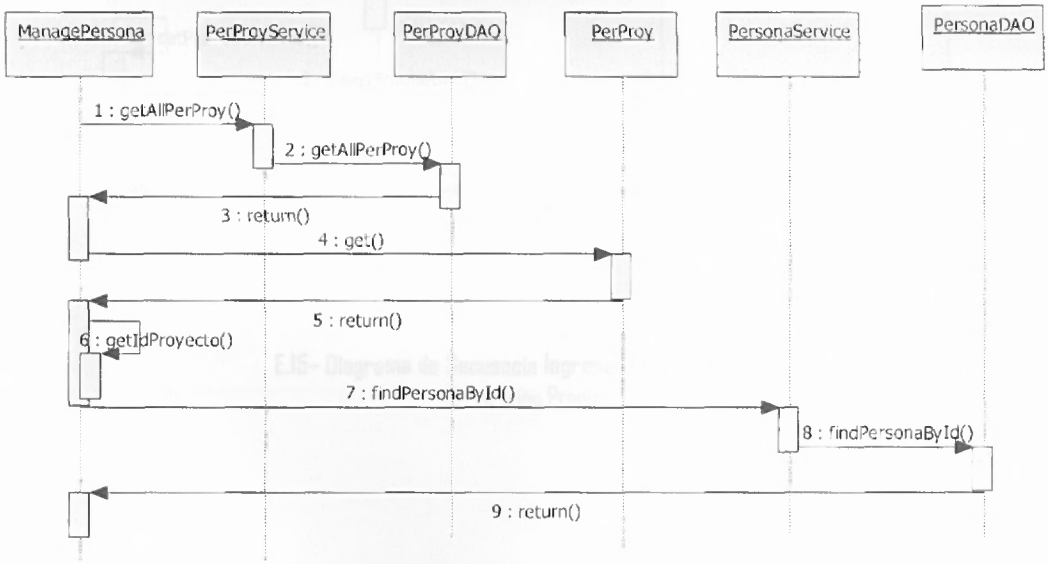

**E.10- Diagrama de Secuencia Editar Persona**
Elaboración Propia



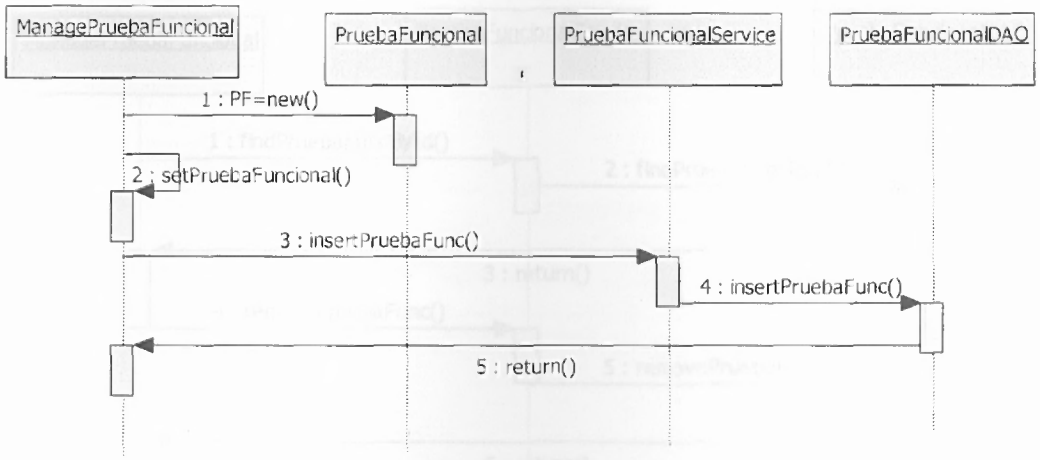

**E.11- Diagrama de Secuencia Eliminar Integrante**
Elaboración Propia



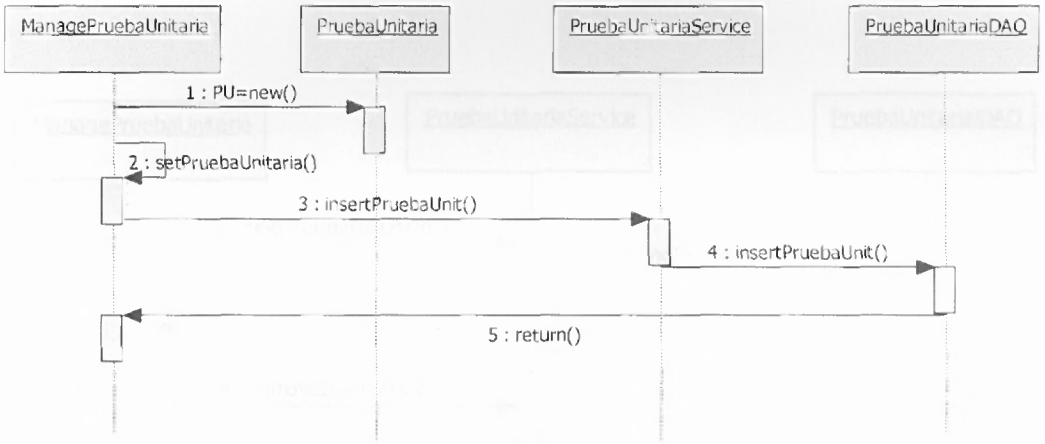

**E.12- Diagrama de Secuencia Eliminar Persona**
Elaboración Propia



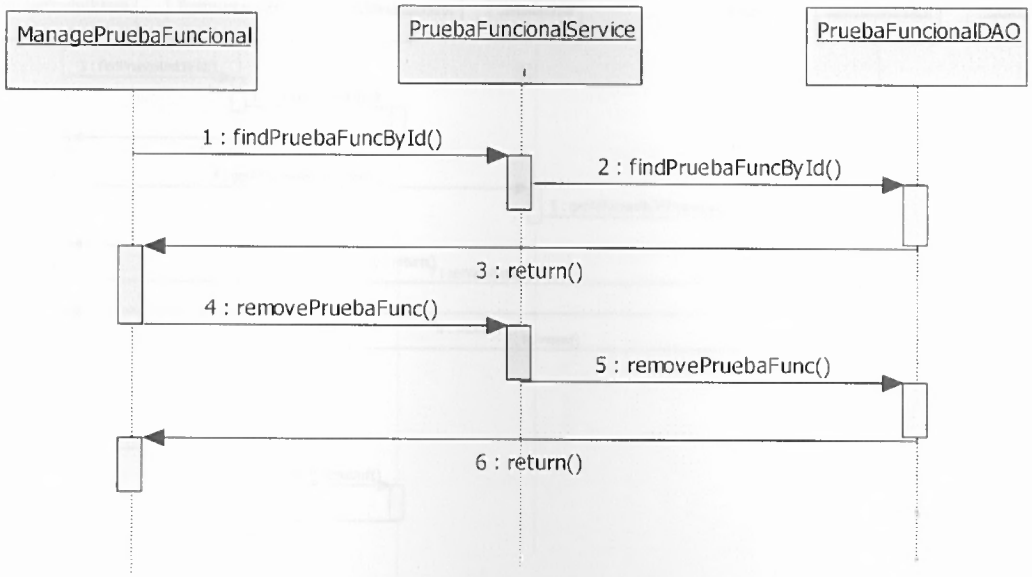

**E.13- Diagrama de Secuencia Listar Equipo**
Elaboración Propia



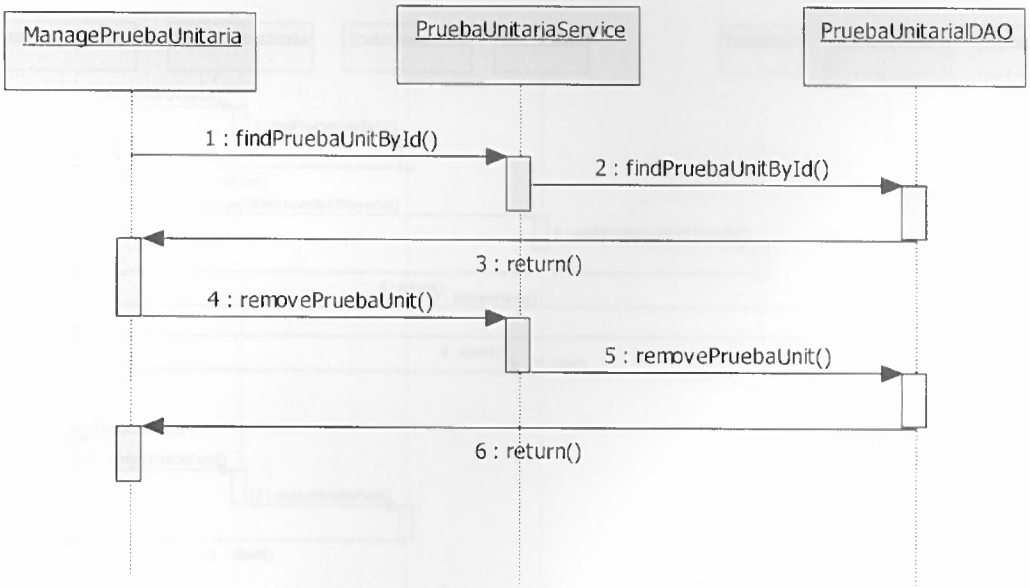

**E.14- Diagrama de Secuencia Ingresar Prueba Funcional**
Elaboración Propia



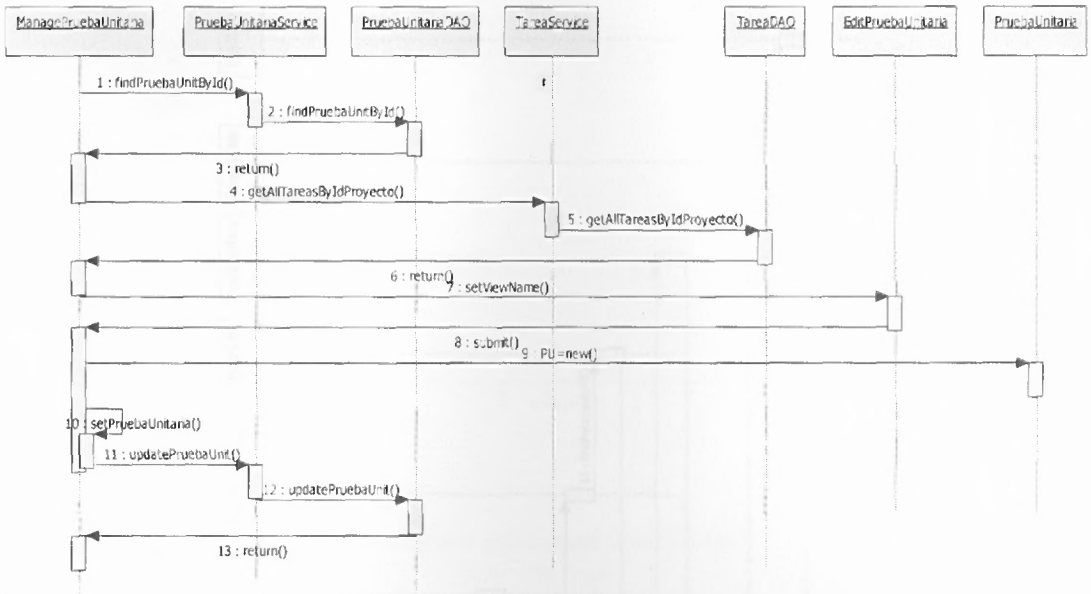

**E.15- Diagrama de Secuencia Ingresar Prueba Unitaria**
Elaboración Propia



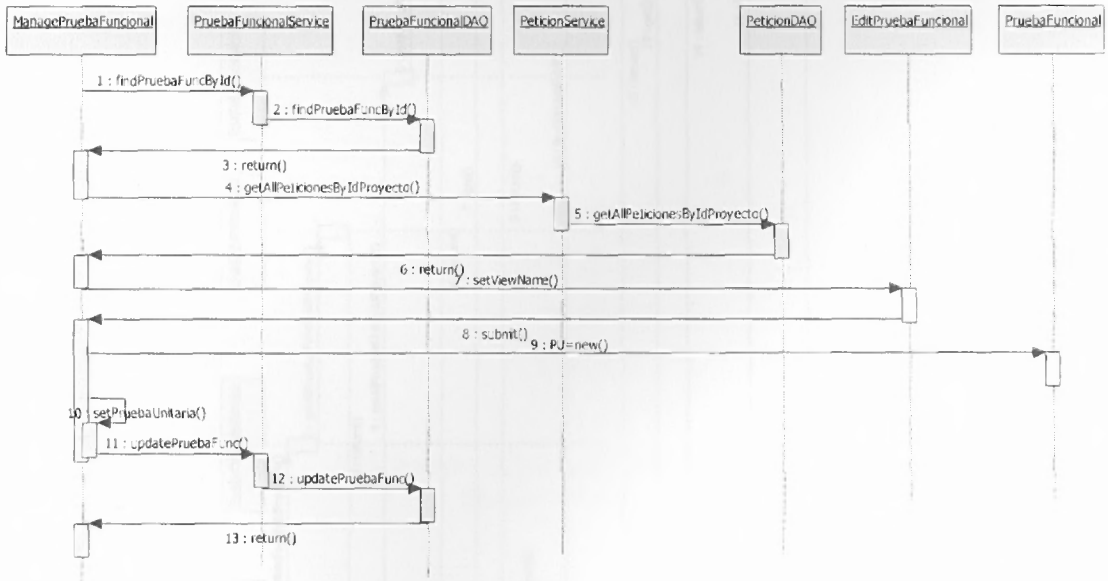

E.16- Diagrama de Secuencia Eliminar Prueba Funcional
Elaboración Propia



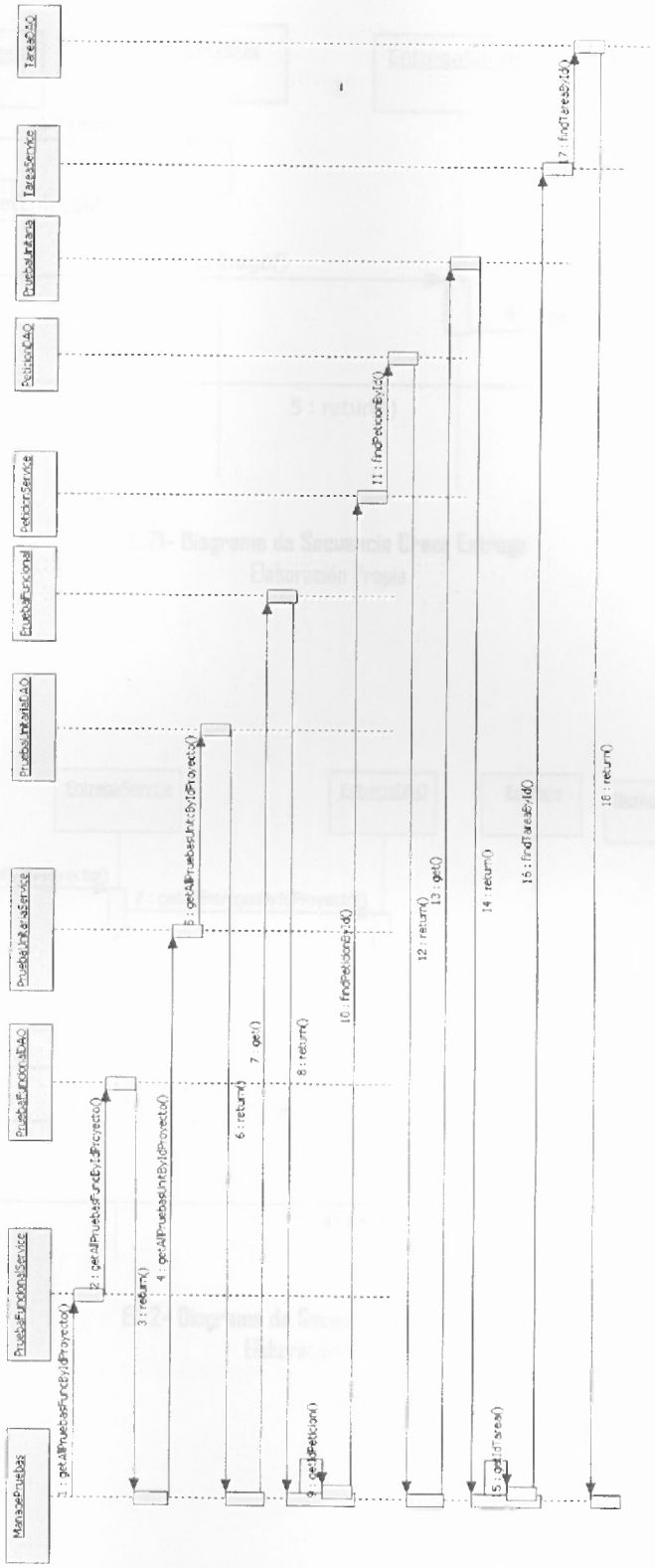

E.17- Diagrama de Secuencia Eliminar Prueba Unitaria
Elaboración Propia



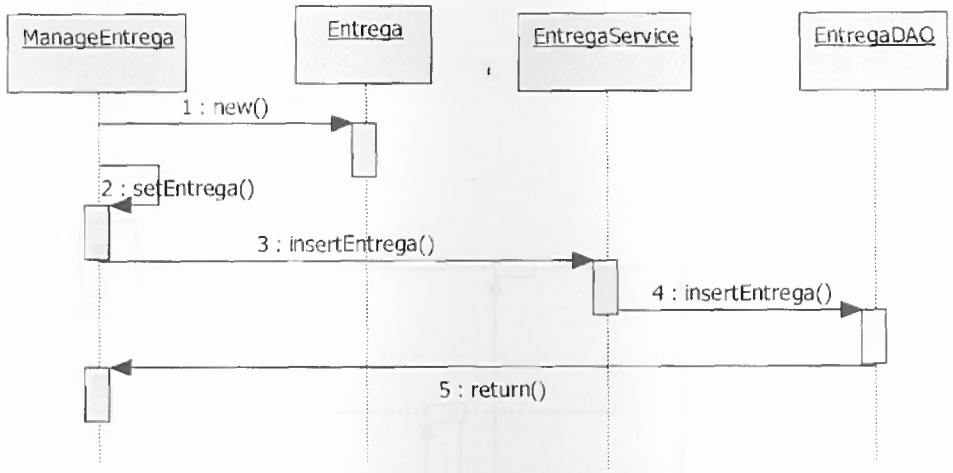

**E.18- Diagrama de Secuencia Modificar Prueba Unitaria**
Elaboración Propia



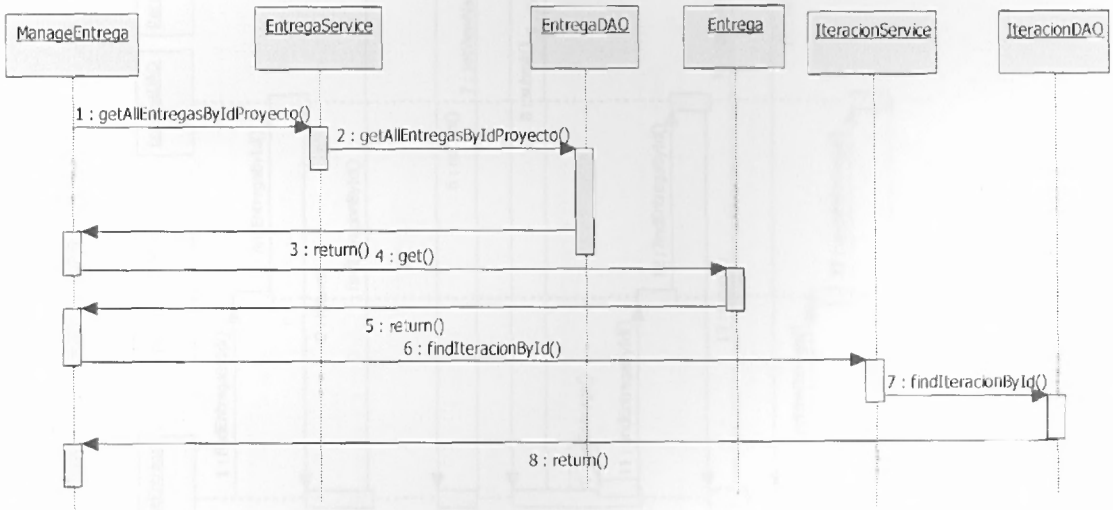

**E.19- Diagrama de Secuencia Modificar Prueba Funcional**
Elaboración Propia



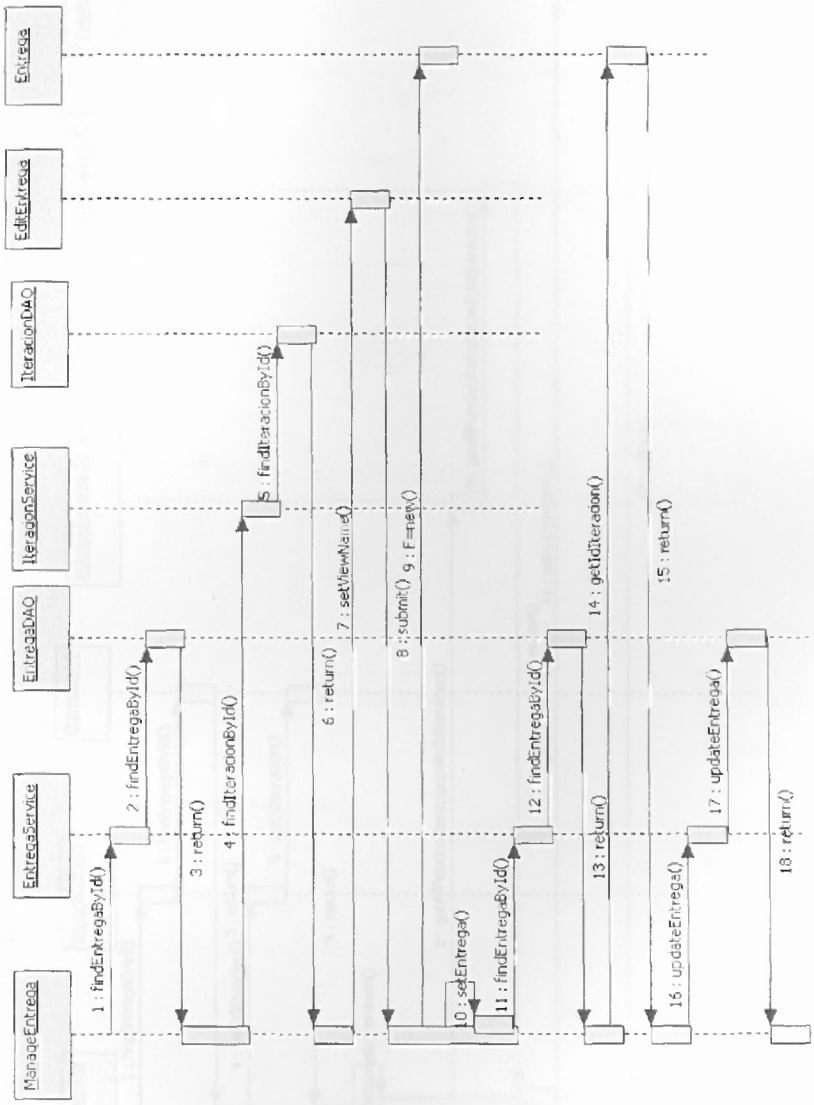

E.20- Diagrama de Secuencia Listar Pruebas
Elaboración Propia




**E.21- Diagrama de Secuencia Crear Entrega**
Elaboración Propia

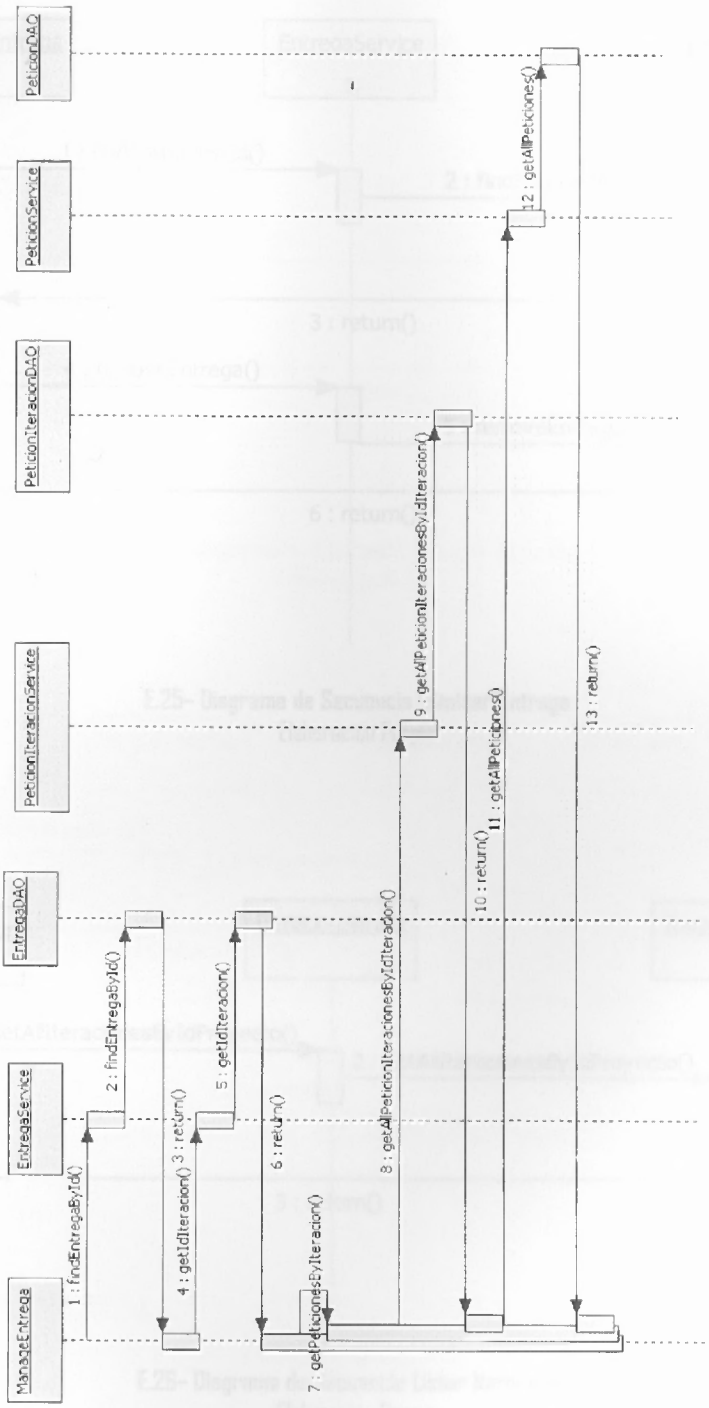



**E.22- Diagrama de Secuencia Listar Entrega**
Elaboración Propia

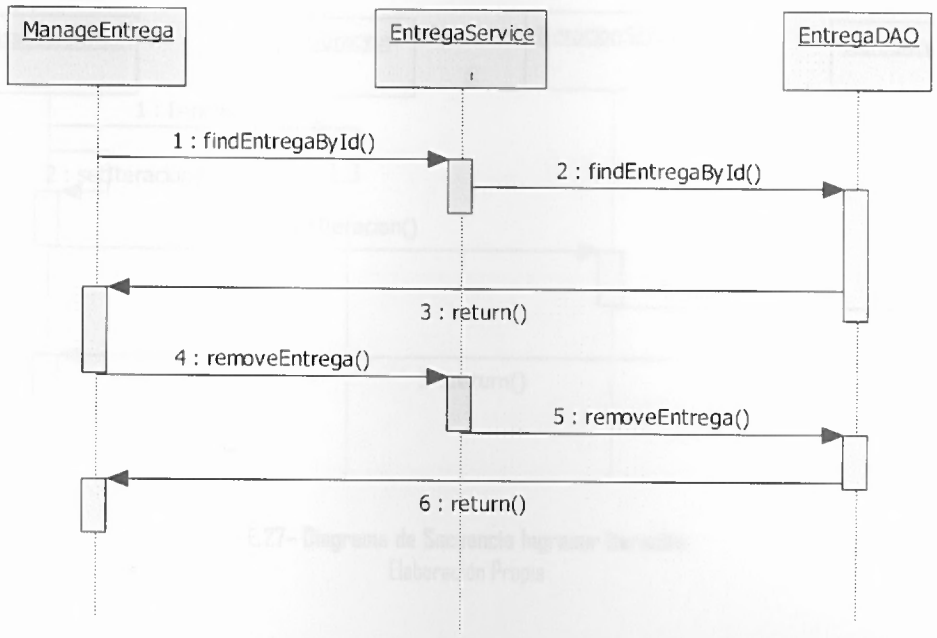



**E.23- Diagrama de Secuencia Editar Entrega**

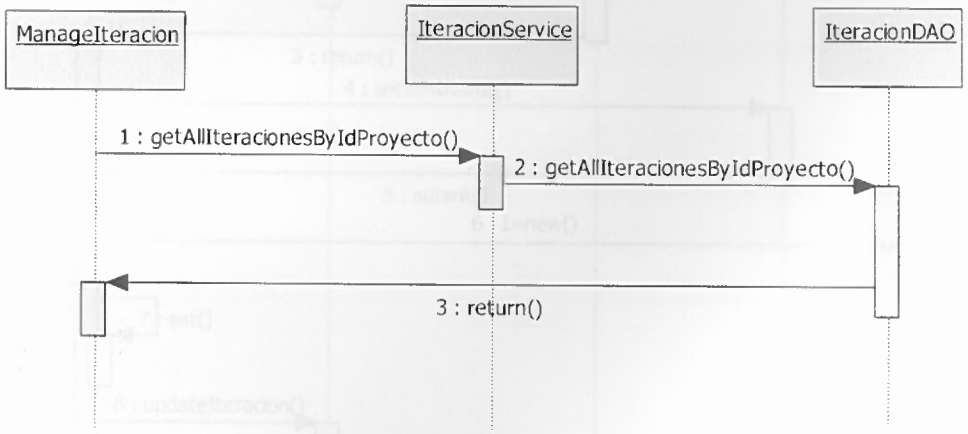
Elaboración Propia



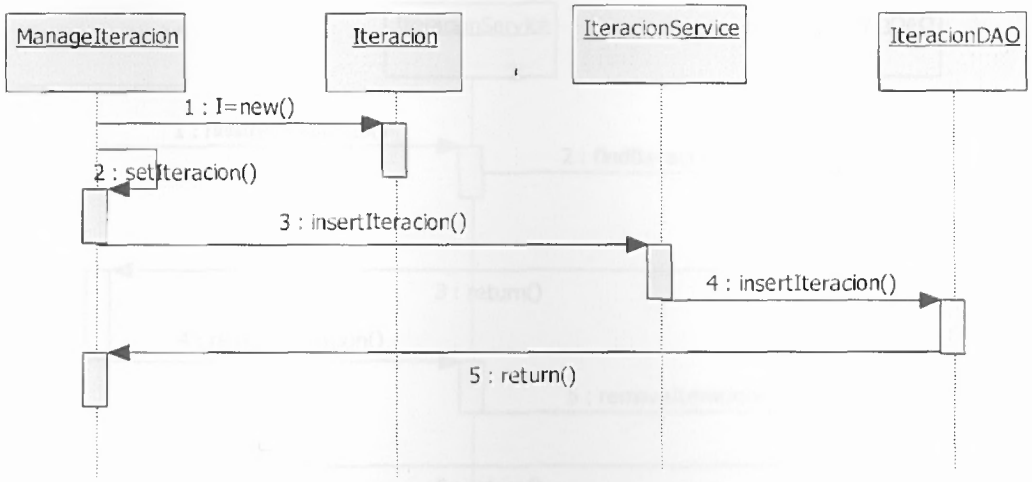

**E.24- Diagrama de Secuencia Listar Peticiones por Entrega**
Elaboración Propia



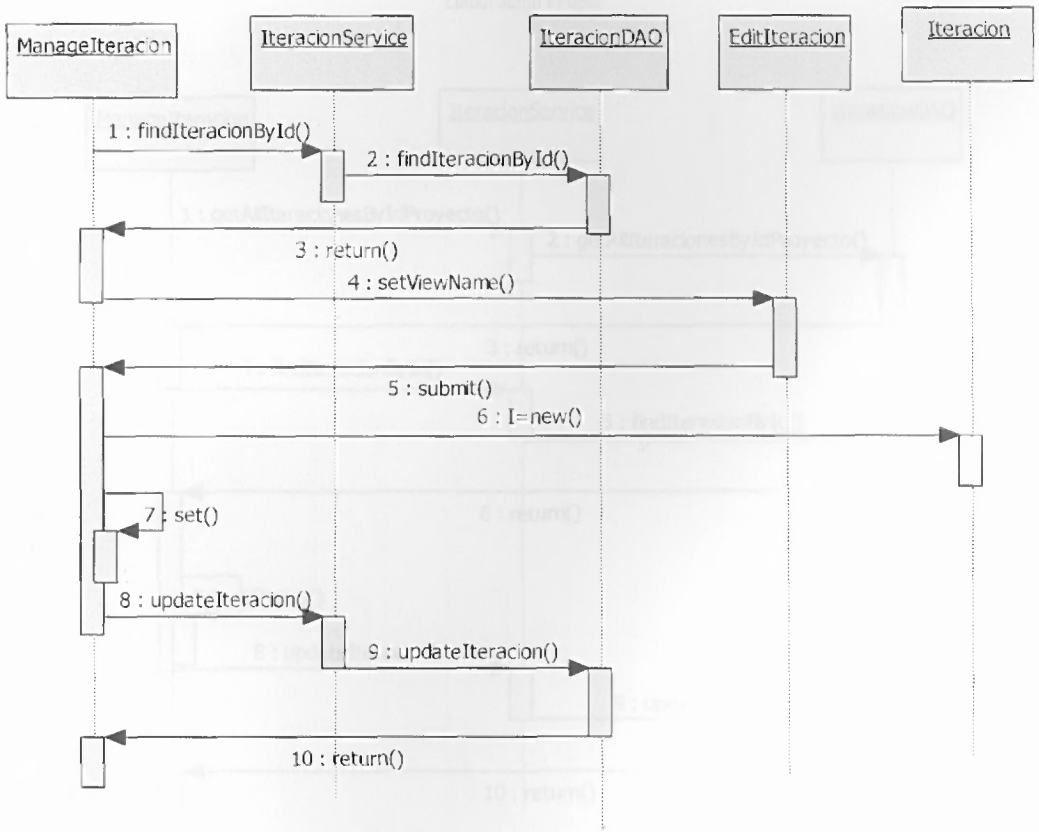

**E.25- Diagrama de Secuencia Eliminar Entrega**
Elaboración Propia



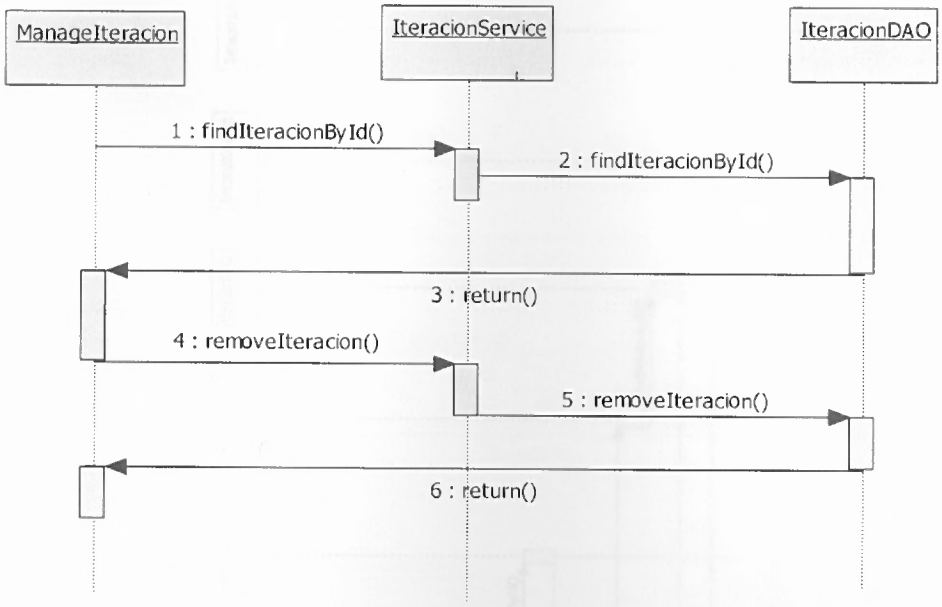

**E.26- Diagrama de Secuencia Listar Iteraciones**
Elaboración Propia



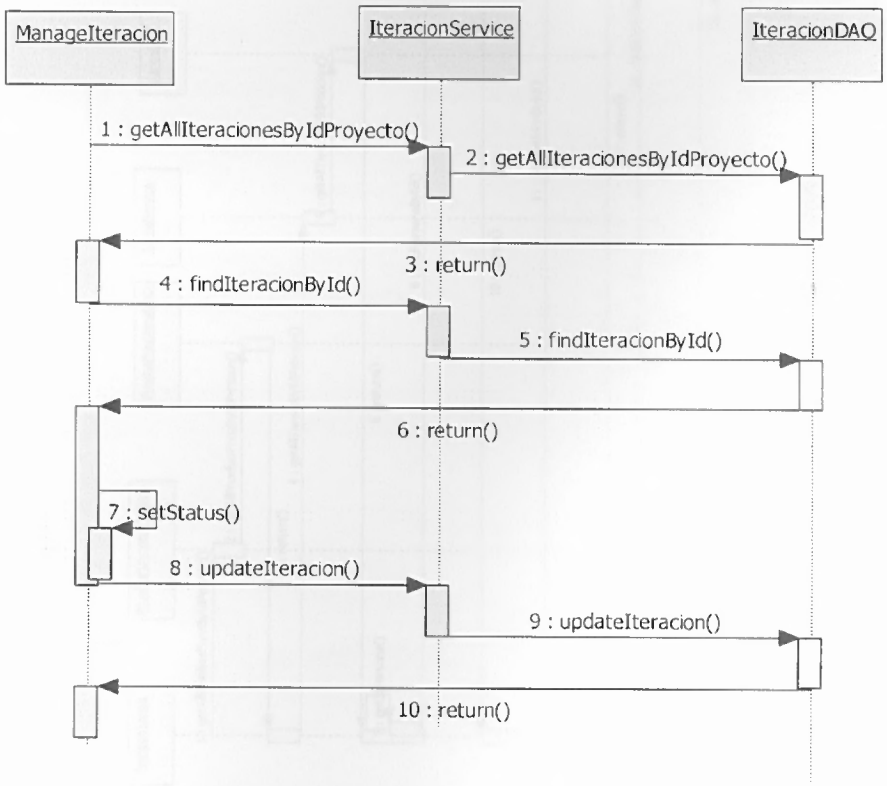

**E.27- Diagrama de Secuencia Ingresar Iteración**
Elaboración Propia



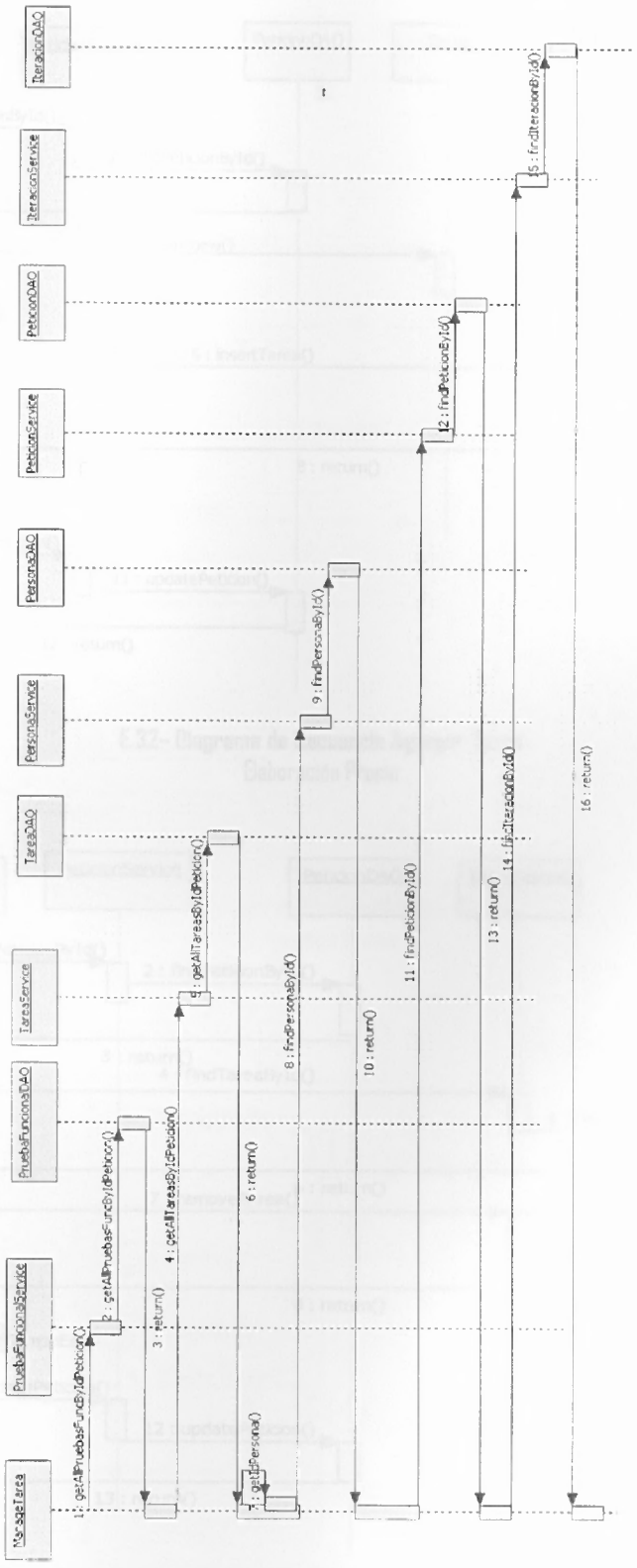

**E.28- Diagrama de Secuencia Editar Iteración**
Elaboración Propia



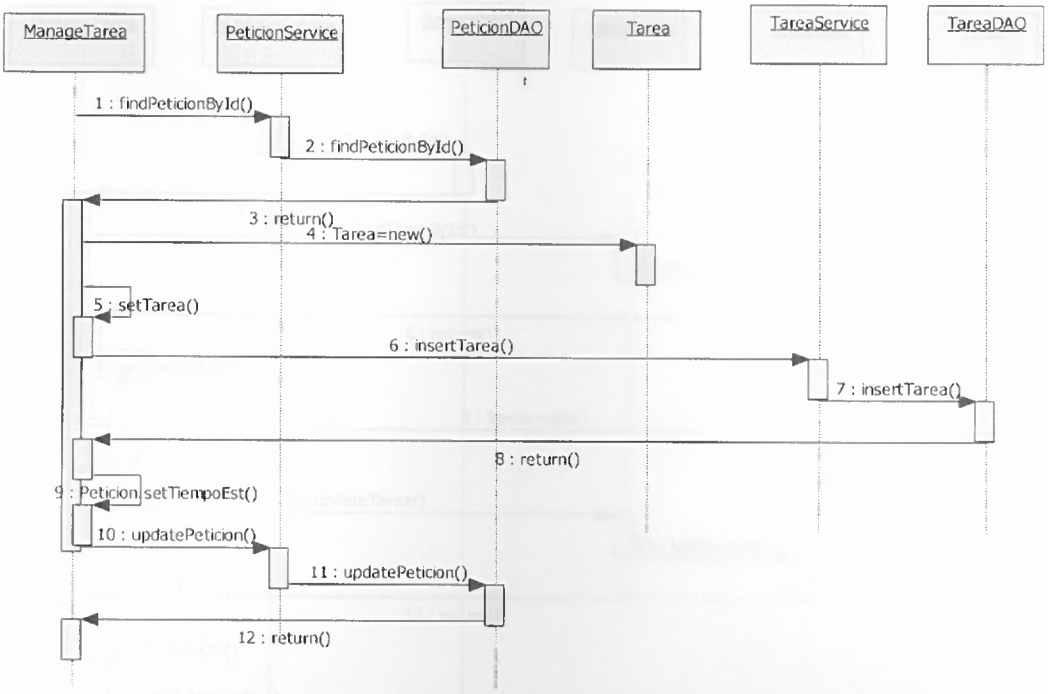

**E.29- Diagrama de Secuencia Eliminar Iteración**
Elaboración Propia



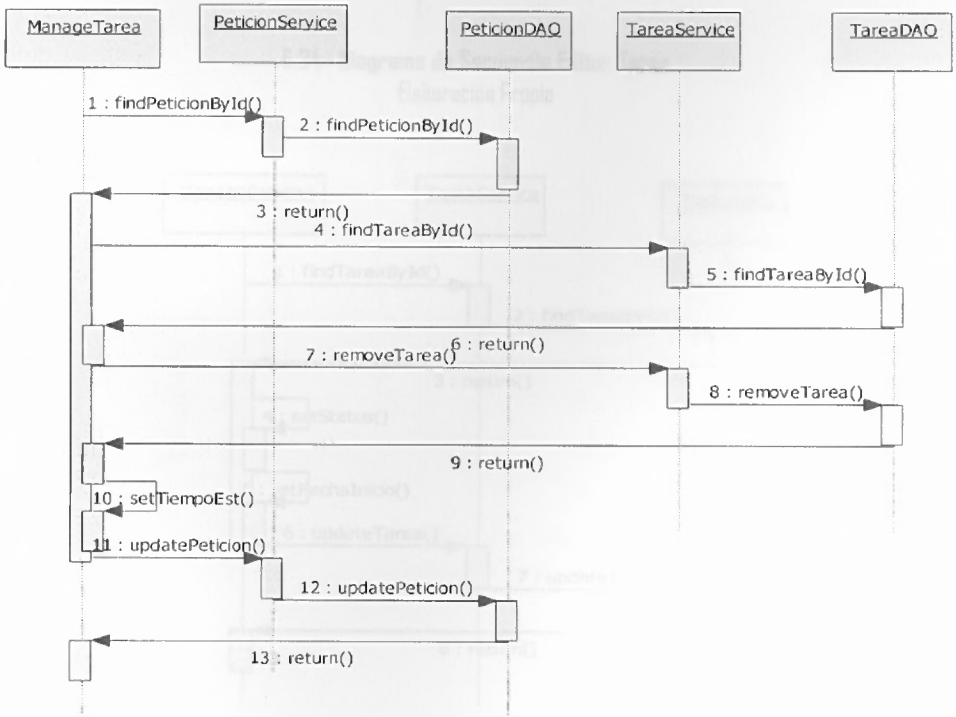

**E.30- Diagrama de Secuencia Cambiar Status Iteración**
Elaboración Propia



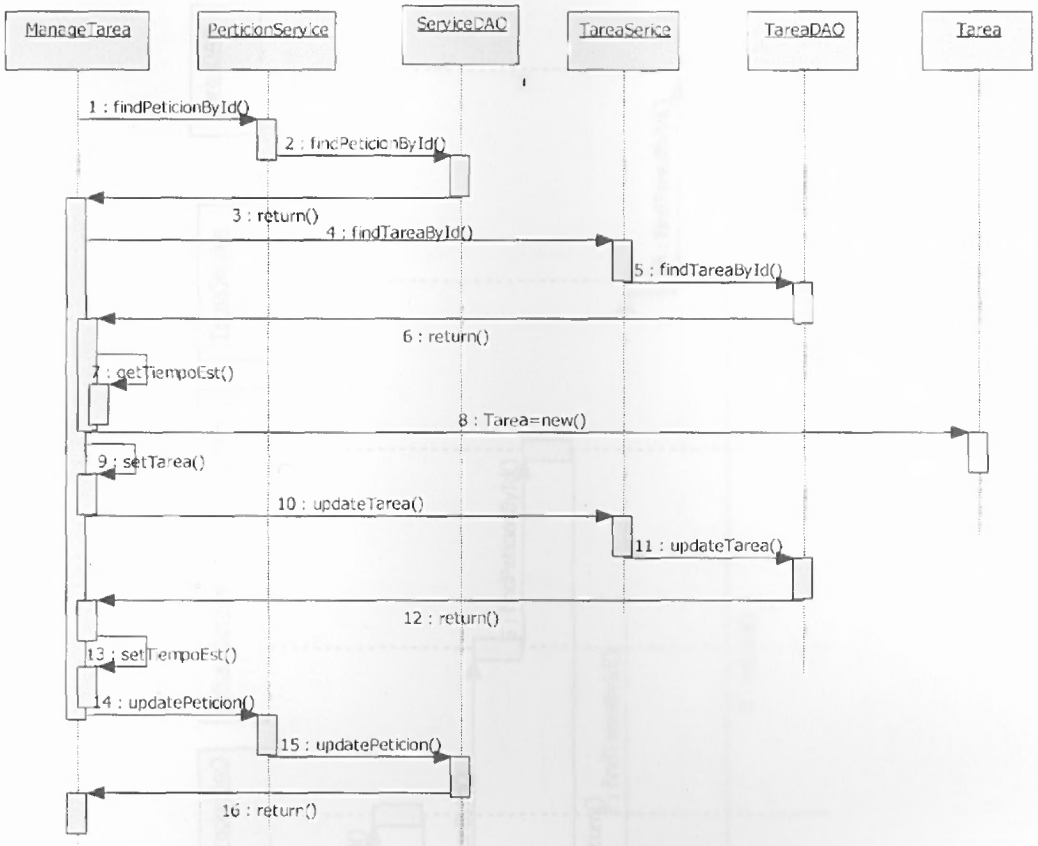

E.31- Diagrama de Secuencia Listar Tarea/ Prueba Funcional por Petición
Elaboración Propia



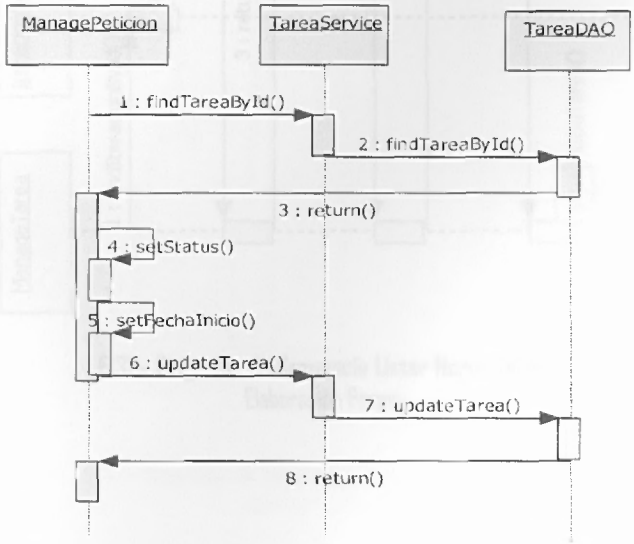

**E.32- Diagrama de Secuencia Agregar Tarea**
Elaboración Propia



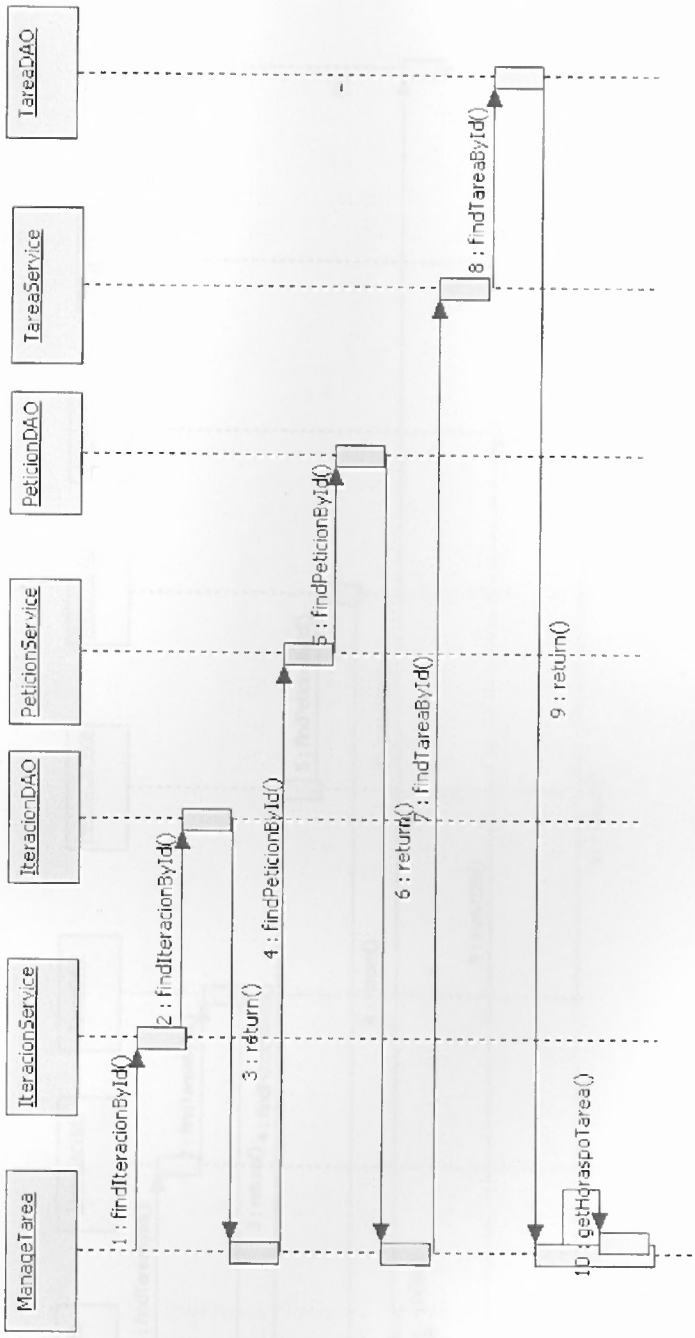

**E.33- Diagrama de Secuencia Eliminar Tarea**
Elaboración Propia



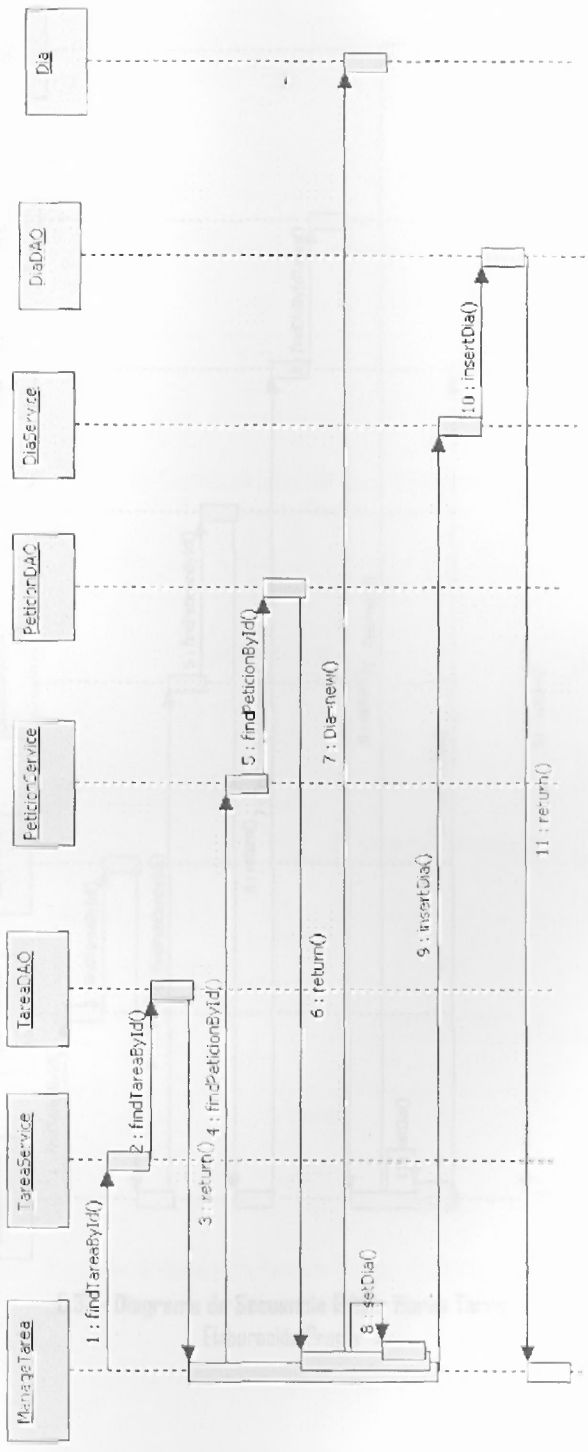

**E.34- Diagrama de Secuencia Editar Tarea**
Elaboración Propia



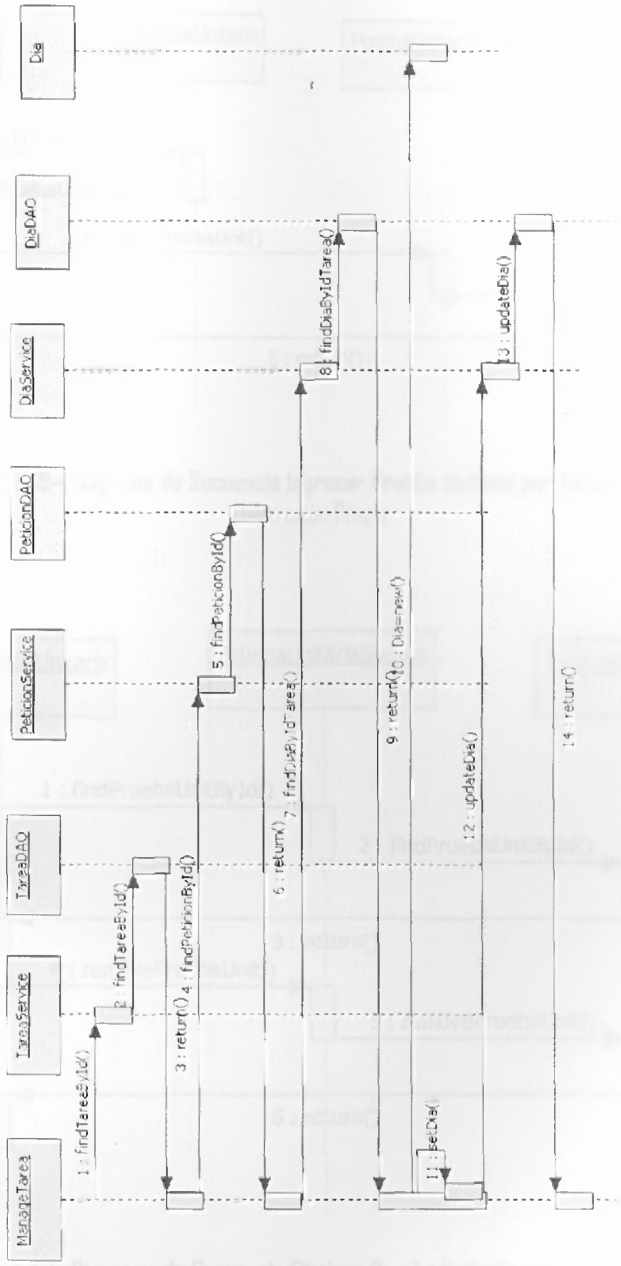

**E.35- Diagrama de Secuencia Cambiar Status Tarea**
Elaboración Propia



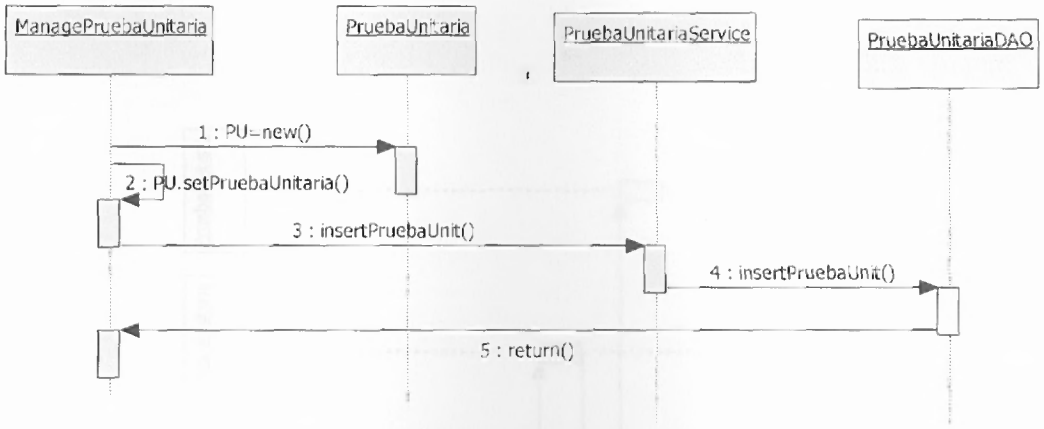

**E.36- Diagrama de Secuencia Listar Horas Tarea**
Elaboración Propia



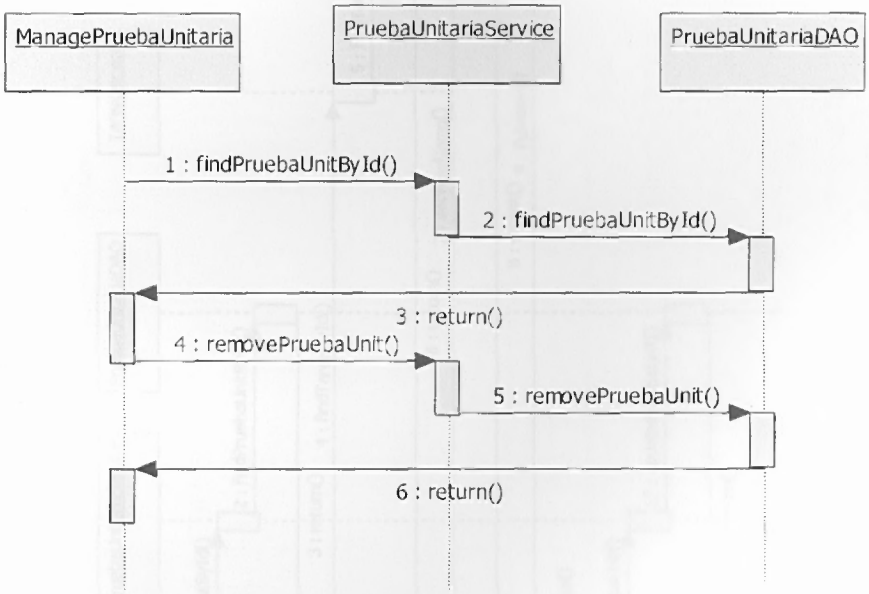

E.37- Diagrama de Secuencia Ingresar Horas Tarea
Elaboración Propia



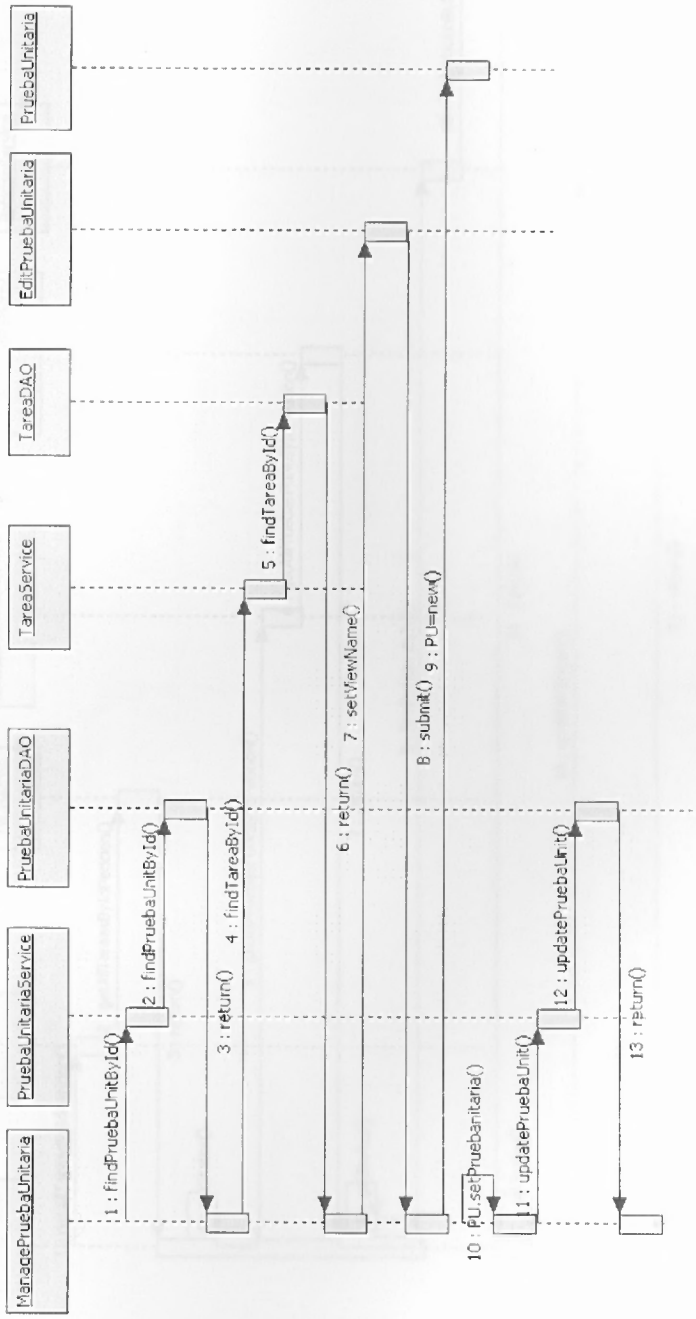

E.38- Diagrama de Secuencia Editar Horas Tarea
Elaboración Propia



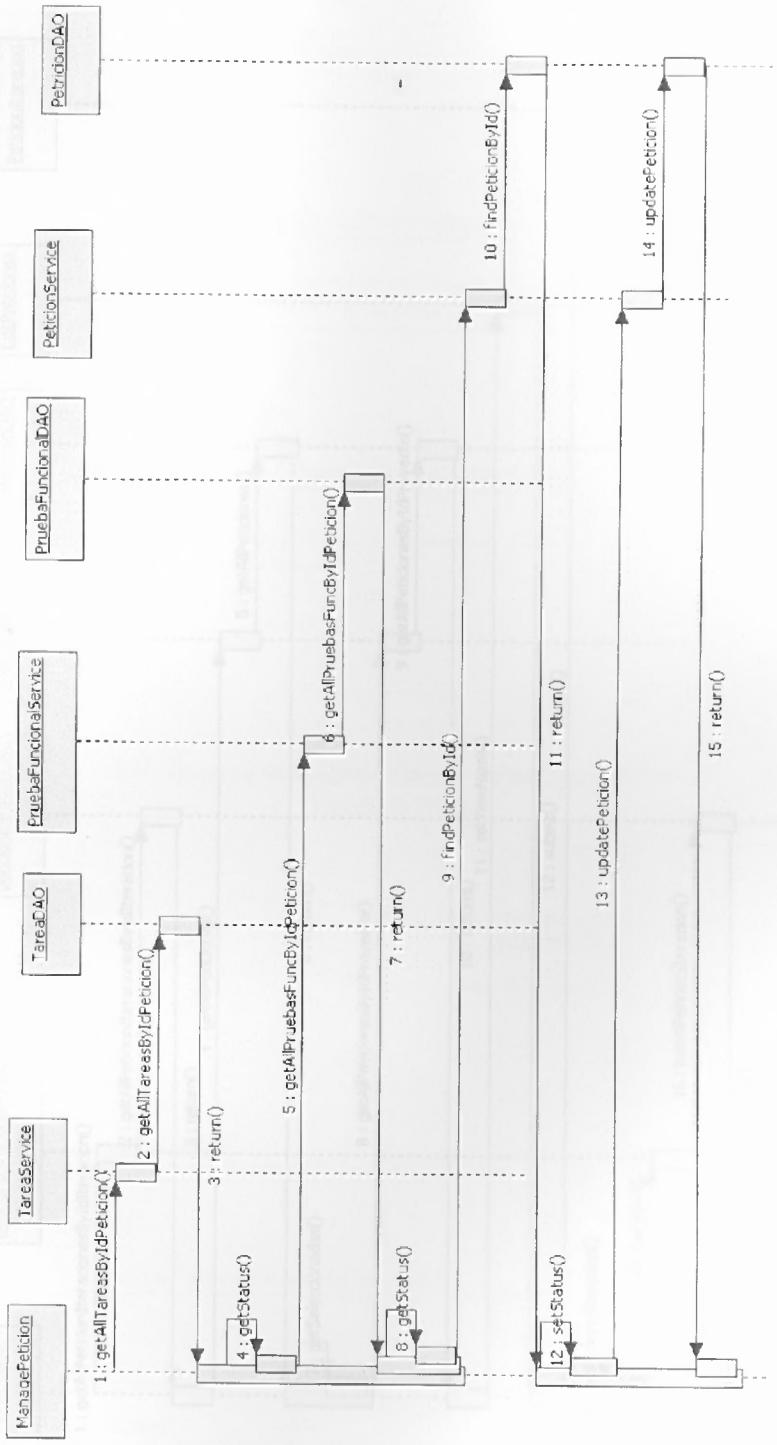

E.39- Diagrama de Secuencia Ingresar Prueba Unitaria por Tarea
Elaboración Propia



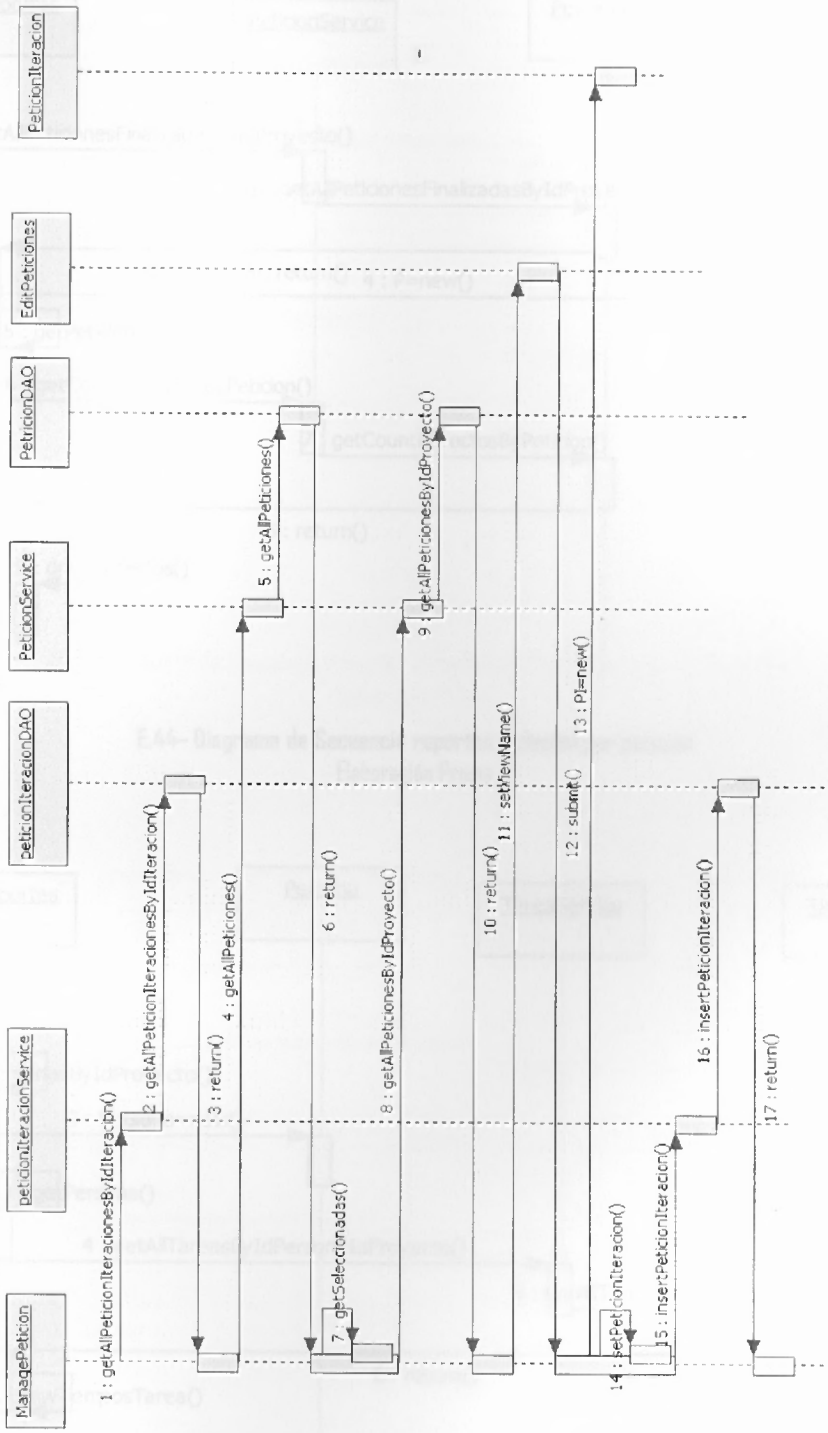

E.40- Diagrama de Secuencia Eliminar Prueba Unitaria por Tarea
Elaboración Propia



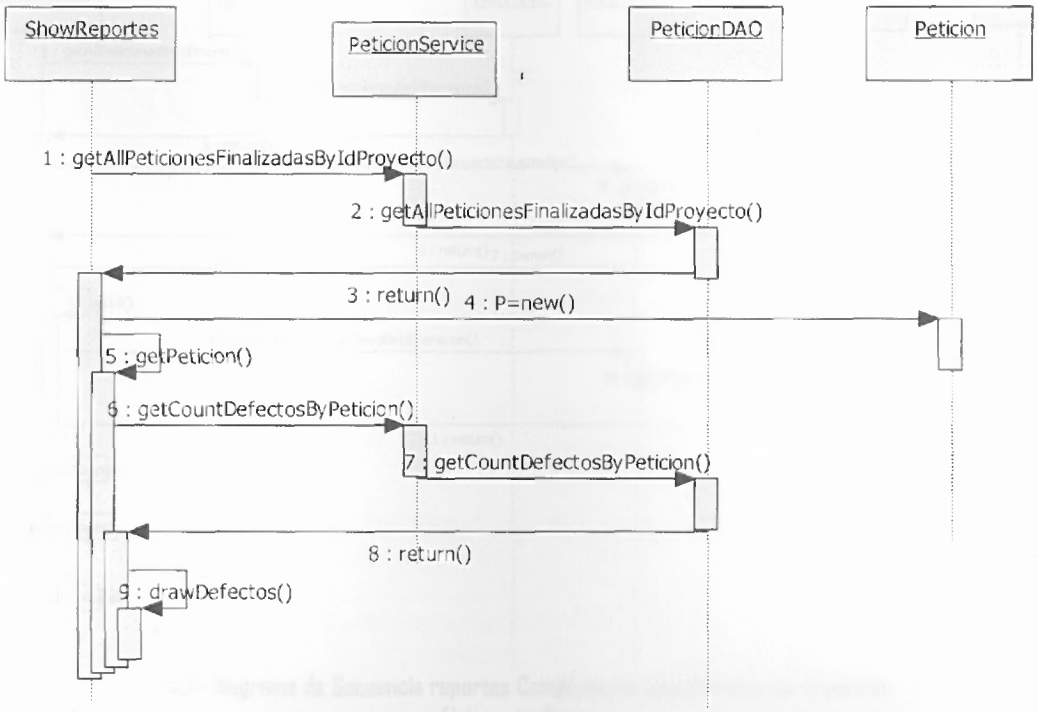

E.41- Diagrama de Secuencia Editar Prueba Unitaria por Tarea
Elaboración Propia



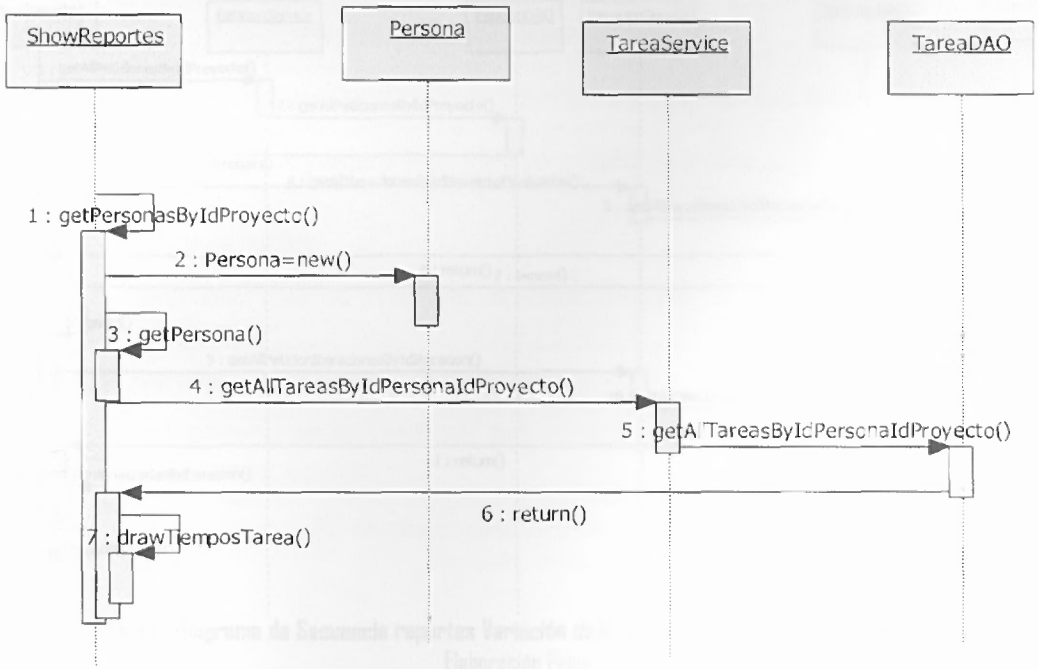

E.42- Diagrama de Secuencia Cambiar Status Peticiones
Elaboración Propia



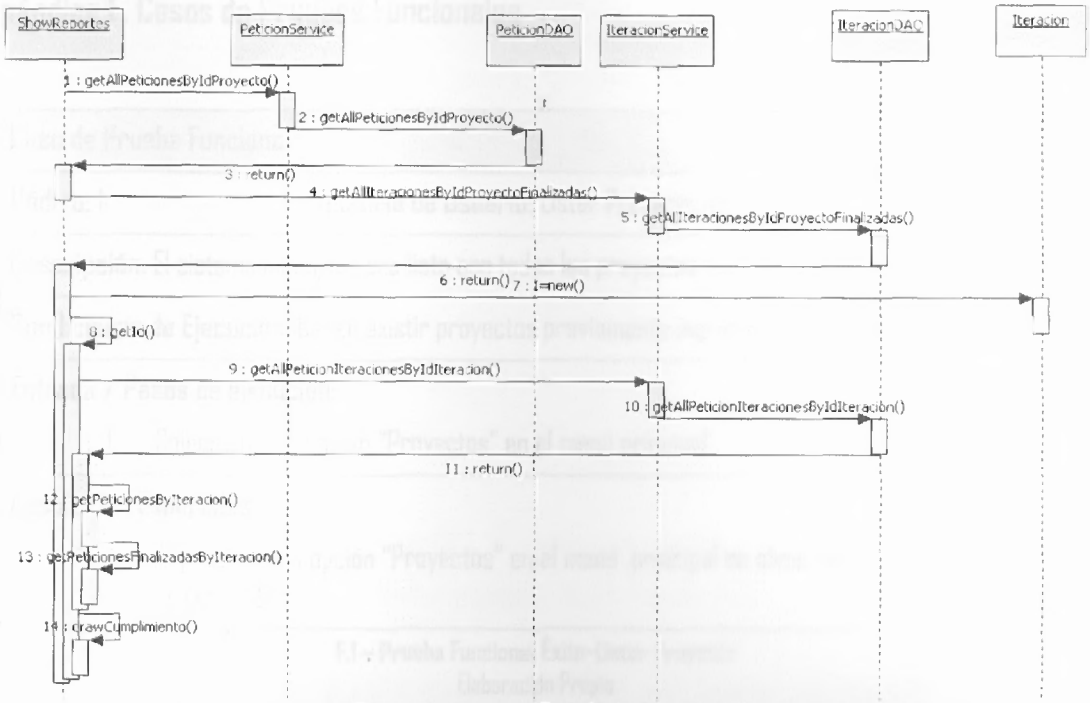

**E.43- Diagrama de Secuencia Asignar/Editar Peticiones**
Elaboración Propia



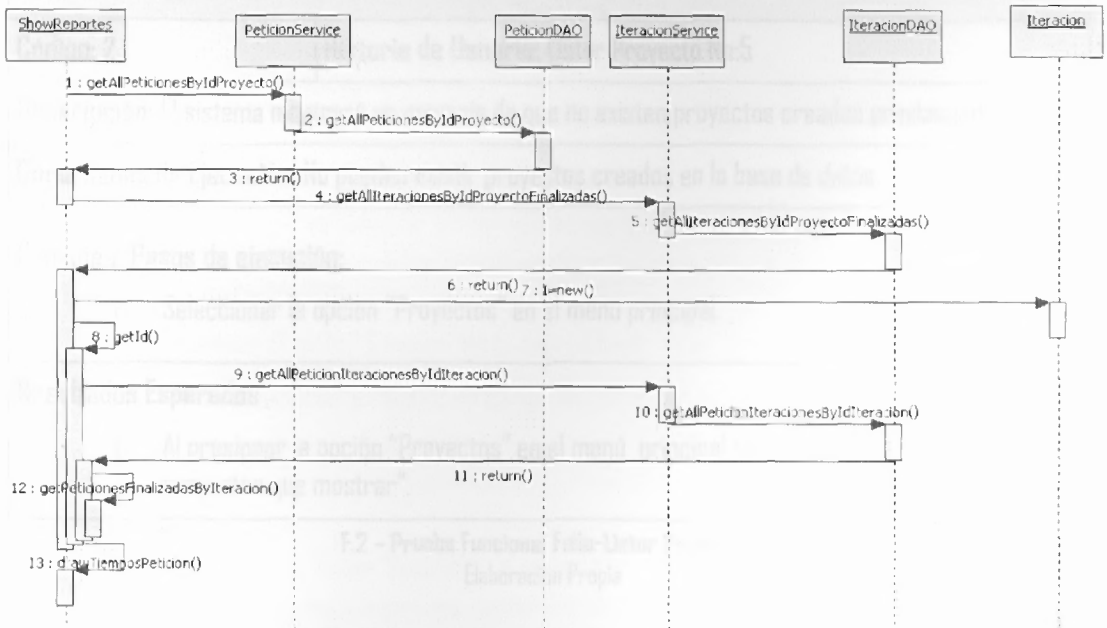

**E.44- Diagrama de Secuencia reportes Defectos por petición**
Elaboración Propia




**E.45- Diagrama de Secuencia reportes Cumplimiento de tareas por persona**
Elaboración Propia




E.46- Diagrama de Secuencia reportes Cumplimiento de peticiones por iteración
Elaboración Propia




E.47- Diagrama de Secuencia reportes Variación de la estimación de tiempo por petición
Elaboración Propia

## Apéndice F. Casos de Pruebas Funcionales

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 1 | **Historia de Usuario:** Listar Proyecto No:5 |
| **Descripción:** El sistema mostrará una lista con todos los proyectos existentes en la base de datos. | |
| **Condiciones de Ejecución:** Deben existir proyectos previamente ingresados en la base de datos. | |
| **Entrada / Pasos de ejecución:**<br>1. Seleccionar la opción "Proyectos" en el menú principal. | |
| **Resultados Esperados**<br>1. Al presionar la opción "Proyectos" en el menú principal se observará por pantalla la lista de proyectos. | |

**F.1 – Prueba Funcional Éxito-Listar Proyecto**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 2 | **Historia de Usuario:** Listar Proyecto No:5 |
| **Descripción:** El sistema mostrará un mensaje de que no existen proyectos creados previamente. | |
| **Condiciones de Ejecución:** No pueden existir proyectos creados en la base de datos. | |
| **Entrada / Pasos de ejecución:**<br>1. Seleccionar la opción "Proyectos" en el menú principal. | |
| **Resultados Esperados**<br>1. Al presionar la opción "Proyectos" en el menú principal se observará el mensaje: "No hay proyectos que mostrar". | |

**F.2 – Prueba Funcional Fallo-Listar Proyecto**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 3 | **Historia de Usuario:** Agregar Proyecto No:6 |
| **Descripción:** El sistema ingresará en la base de datos el nuevo proyecto. | |
| **Condiciones de Ejecución:** Se deben llenar todos los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Llenar los campos del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", los datos serán ingresados en la BD.<br>2. El sistema re direccionará a la pantalla de Listar Proyectos, y se verá el proyecto con los datos ingresados.<br>3. El sistema mostrará un mensaje de éxito: "Operación Exitosa", en la parte de arriba de la tabla. | |

**F.3 – Prueba Funcional Éxito-Agregar Proyecto**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 4 | **Historia de Usuario:** Agregar Proyecto No:6 |
| **Descripción:** El sistema validará que falta por llenar algún campo obligatorio y mostrará un mensaje de error. | |
| **Condiciones:** Se debe dejar en blanco al menos uno de los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Llenar el formulario dejando en blanco el campo código.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", el sistema se mantendrá en la página de Agregar Proyecto, con los campos que fueron llenados.<br>2. El sistema mostrará el mensaje: "Debe llenar los campos obligatorios". | |

**F.4 – Prueba Funcional Fallo-Agregar Proyecto**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 5 | **Historia de Usuario:** Editar Proyecto No:7 |
| **Descripción:** El sistema editará los valores de la base de datos del proyecto. | |
| **Condiciones:** Se deben llenar todos los campos obligatorios del formulario | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar", en la columna de opciones, del proyecto que se quiere editar.<br>2. Editar el campo "Nombre" del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", los datos cambiados serán editados en la base de datos.<br>2. El sistema re direccionará a la pantalla de Listar Proyectos, y se verá el proyecto con los datos editados.<br>3. El sistema mostrará un mensaje de éxito: "Operación Exitosa", en la parte de arriba de la tabla. | |

**F.5 – Prueba Funcional Éxito-Editar Proyecto**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 6 | **Historia de Usuario:** Editar Proyecto No:7 |
| **Descripción:** El sistema no editará los valores de la base de datos del proyecto. | |
| **Condiciones:** Se debe dejar en blanco al menos uno de los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar", del proyecto que se quiere editar.<br>2. Dejar en blanco el campo "Nombre" del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema se mantendrá en la página de Editar Proyecto, con los cambios en los campos.<br>2. Se mostrará el mensaje: "Debe llenar los campos obligatorios". | |

**F.6 – Prueba Funcional Fallo-Editar Proyecto**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 7 | **Historia de Usuario:** Eliminar Proyecto No:8 |
| **Descripción:** El sistema eliminará el proyecto de la lista. | |
| **Condiciones de Ejecución:** El proyecto a eliminar debe existir en la base de datos. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Eliminar", del proyecto que se quiere eliminar.<br>2. El sistema mostrará una ventana de alerta.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", el sistema eliminará el proyecto.<br>2. Se re direccionará a la página Listar Proyectos.<br>3. El sistema mostrará el mensaje: "Operación Exitosa". | |

**F.7 – Prueba Funcional Éxito-Eliminar Proyecto**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 8 | **Historia de Usuario:** Ingresar Petición No:10 |
| **Descripción:** El sistema ingresará en la base de datos la nueva petición. | |
| **Condiciones:** Debe existir el proyecto al que se le agregará la petición. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Llenar los campos obligatorios del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", los datos serán ingresados en la BD.<br>2. El sistema re direccionará a la pantalla de Listar Peticiones, y se verá la petición con los datos ingresados.<br>3. El sistema mostrará un mensaje de éxito: "Operación Exitosa". | |

**F.8 – Prueba Funcional Éxito-Ingresar Petición**
Elaboración Propia

| **Caso de Prueba Funcional** | |
|---|---|
| **Código:** 9 | **Historia de Usuario:** Ingresar Petición No:10 |
| **Descripción:** El sistema no ingresará los de datos. | |
| **Condiciones de Ejecución:** Debe existir el proyecto al que se le agregará la petición. Se deben dejar en blanco alguno de los campos obligatorios del formulario. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Dejar en blanco el campo "Descripción".<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema se mantendrá en a la pantalla de Agregar Peticiones, con los valores en los campos.<br>2. El sistema mostrará un mensaje: "Debe llenar todos los campos obligatorios". | |

**F.9 – Prueba Funcional Fallo-Ingresar Petición**
Elaboración Propia

| **Caso de Prueba Funcional** | |
|---|---|
| **Código:** 10 | **Historia de Usuario:** Listar Peticiones No:9 |
| **Descripción:** El sistema listará todas las peticiones de un proyecto. | |
| **Condiciones:** Debe existir el proyecto y peticiones ingresadas en la base de datos. | |
| **Entrada / Pasos de ejecución:**<br>Opción 1: 1. Seleccionar "Proyecto" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Seleccionar "Peticiones" en el menú principal.<br>Opción 2: 1. Seleccionar "Peticiones" en el menú principal.<br>2. Seleccionar el proyecto del *select* en la parte superior de la pantalla. | |
| **Resultados Esperados**<br>1. El sistema mostrará las peticiones existentes para el proyecto. | |

**F.10 – Prueba Funcional Éxito-Listar Peticiones**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 11 | **Historia de Usuario:** Listar Peticiones No:9 |
| **Descripción:** El sistema mostrará un mensaje de que no existen peticiones. | |
| **Condiciones de Ejecución:** Debe existir el proyecto al cual pertenecen las peticiones a listar. No Deben existir peticiones en la base de datos. | |
| **Entrada / Pasos de ejecución:**<br>Opción 1: 1. Seleccionar "Proyecto" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Seleccionar "Peticiones" en el menú principal.<br>Opción 2: 1. Seleccionar "Peticiones" en el menú principal.<br>2. Seleccionar el proyecto del *select* en la parte superior de la pantalla. | |
| **Resultados Esperados**<br>1. Se mostrará el mensaje: "No existen Peticiones para el Proyecto". | |

F.11 – Prueba Funcional Fallo-Listar Peticiones
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 12 | **Historia de Usuario:** Editar Petición No:11 |
| **Descripción:** El sistema editará los valores de petición de la base de datos. | |
| **Condiciones:** Debe existir la petición. Llenar los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar", de la petición que se quiere editar.<br>2. Editar el campo "Descripción" del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Los datos serán ingresados en la base de datos.<br>2. Se re direccionará a la pantalla de Listar Peticiones, y se verá la petición con los datos editados.<br>3. El sistema mostrará un mensaje de éxito: "Operación Exitosa". | |

F.12 – Prueba Funcional Éxito-Editar Petición
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 13 | **Historia de Usuario:** Editar Petición No:11 |
| **Descripción:** El sistema mostrará un mensaje de que no ha sido posible editar. | |
| **Condiciones de Ejecución:** La petición a editar debe existir en la BD. Se debe dejar en blanco alguno de los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar", de la petición que se quiere editar.<br>2. Dejar en blanco el campo "Descripción" del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", el sistema validará los datos y se mantiene en la pantalla Editar Petición, con los campos editados.<br>2. El sistema mostrará un mensaje: "Debe llenar todos los campos obligatorios". | |

**F.13 – Prueba Funcional Fallo-Editar Petición**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 14 | **Historia de Usuario:** Eliminar Petición No:12 |
| **Descripción:** El sistema eliminará la petición. | |
| **Condiciones de Ejecución:** El proyecto y la petición a eliminar deben existir. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Eliminar", de la petición que se quiere eliminar.<br>2. El sistema mostrará una ventana de validación: "¿Esta seguro que desea eliminar la petición?".<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", el sistema eliminará la petición.<br>2. El sistema re direccionará a la pantalla listar peticiones y mostrará un mensaje: "Operación Exitosa". | |

**F.14 – Prueba Funcional Éxito-Eliminar Petición**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 15 | **Historia de Usuario:** Ingresar Defecto No:14 |
| **Descripción:** El sistema ingresará en la base de datos el nuevo defecto. | |
| **Condiciones:** Se deben llenar todos los campos obligatorios del formulario. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Llenar los campos obligatorios del formulario. ("Tipo": defecto)<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", los datos serán ingresados en la BD.<br>2. Se re direccionará a la pantalla de Listar Peticiones, y se verá el defecto con los datos ingresados.<br>3. El sistema mostrará un mensaje de éxito: "Operación Exitosa". | |

**F.15 – Prueba Funcional Éxito-Ingresar Defecto**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 16 | **Historia de Usuario:** Listar Pruebas No:17 |
| **Descripción:** El sistema listará todas las pruebas de un proyecto. | |
| **Condiciones:** Debe existir el proyecto y pruebas ingresadas en la base de datos. | |
| **Entrada / Pasos de ejecución:**<br>Opción 1: 1. Seleccionar "Proyecto" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Seleccionar "Pruebas" en el menú principal.<br>Opción 2: 1. Seleccionar "Pruebas" en el menú principal.<br>2. Seleccionar el proyecto del *select* en la parte superior de la pantalla. | |
| **Resultados Esperados**<br>1. El sistema mostrará las pruebas existentes para el proyecto. | |

**F.16 – Prueba Funcional Éxito-Listar Pruebas**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 17 | **Historia de Usuario:** Listar Pruebas No:17 |
| **Descripción:** El sistema mostrará un mensaje de que no existen pruebas. | |
| **Condiciones:** No Deben existir pruebas ingresadas en la BD para el proyecto. | |
| **Entrada / Pasos de ejecución:**<br>Opción 1: 1. Seleccionar "Proyecto" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Seleccionar "Pruebas" en el menú principal.<br>Opción 2: 1. Seleccionar "Pruebas" en el menú principal.<br>2. Seleccionar el proyecto del *select* en la parte superior de la pantalla. | |
| **Resultados Esperados**<br>1. El sistema mostrará el mensaje: "No existen pruebas ingresadas". | |

F.17 – Prueba Funcional Fallo-Listar Pruebas
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 18 | **Historia de Usuario:** Ingresar Prueba No:18 |
| **Descripción:** El sistema ingresará en la base de datos la nueva prueba funcional. | |
| **Condiciones:** Se deben llenar todos los campos obligatorios del formulario. Debe existir la petición a la que se va a asociar la prueba. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Llenar los campos obligatorios del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", los datos serán ingresados en la BD.<br>2. Se re direccionará a la pantalla de Listar Pruebas, y se verá la prueba con los datos ingresados.<br>3. El sistema mostrará un mensaje de éxito: "Operación Exitosa". | |

F.18 – Prueba Funcional Éxito-Ingresar Pruebas Funcionales
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 15 | **Historia de Usuario:** Ingresar Prueba No:18 |
| **Descripción:** El sistema ingresará en la base de datos la nueva prueba unitaria. | |
| **Condiciones:** Se deben llenar todos los campos obligatorios del formulario. Debe existir la tarea a la que se va a asociar la prueba. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Llenar los campos obligatorios del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", los datos serán ingresados en la BD.<br>2. Se re direccionará a Listar Pruebas, y se verá la prueba ingresada.<br>3. El sistema mostrará un mensaje de éxito: "Operación Exitosa". | |

**F.19 – Prueba Funcional Éxito-Ingresar Pruebas Unitarias**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 20 | **Historia de Usuario:** Modificar Prueba No:19 |
| **Descripción:** El sistema editará los datos de la prueba seleccionada. | |
| **Condiciones:** Se deben llenar todos los campos obligatorios del formulario. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar", de la prueba unitaria o funcional a ser editada.<br>2. Editar los campos del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el *link* "Editar", los datos serán editados en la BD.<br>2. Se re direccionará a Listar Pruebas, se verá la prueba editada.<br>3. El sistema mostrará un mensaje: "Operación Exitosa". | |

**F.20 – Prueba Funcional Éxito-Modificar Pruebas**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 21 | **Historia de Usuario:** Modificar Prueba No:19 |
| **Descripción:** El sistema no editará los datos de la prueba seleccionada. | |
| **Condiciones:** Se debe dejar en blanco al menos uno de los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar" de la prueba unitaria o funcional a ser editada.<br>2. Dejar en blanco el campo "Nombre" del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el *link* "Aceptar", validará los datos y se mantendrá en la ventana de modificación.<br>3. El sistema mostrará un mensaje: "Debe llenar los campos obligatorios". | |

**F.21 – Prueba Funcional Fallo-Modificar Pruebas**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 22 | **Historia de Usuario:** Eliminar Prueba No:20 |
| **Descripción:** El sistema eliminará la prueba. | |
| **Condiciones:** El proyecto y la prueba a eliminar deben existir. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Eliminar", de la prueba que se quiere eliminar.<br>2. Se mostrará una ventana de validación: "¿Está seguro que desea eliminar la prueba?".<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", el sistema eliminará la prueba.<br>2. El sistema re direccionará a la pantalla listar pruebas y mostrará un mensaje: "Operación Exitosa". | |

**F.22 – Prueba Funcional Éxito-Eliminar Pruebas**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 23 | **Historia de Usuario:** Listar Equipo No:21 |
| **Descripción:** El sistema listará todas las personas asignadas a un proyecto. | |
| **Condiciones:** Debe existir el proyecto y personas ingresadas en la base de datos, además deben estar asignadas al proyecto. | |
| **Entrada / Pasos de ejecución:**<br>Opción 1: 1. Seleccionar "Proyecto" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Seleccionar "Equipo" en el menú principal.<br>Opción 2: 1. Seleccionar "Equipo" en el menú principal.<br>2. Seleccionar el proyecto del *select* en la parte superior de la pantalla | |
| **Resultados Esperados**<br>1. El sistema mostrará las personas existentes para el proyecto. | |

**F.23 – Prueba Funcional Éxito-Listar Equipo**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 24 | **Historia de Usuario:** Listar Equipo No:21 |
| **Descripción:** El sistema mostrará un mensaje de que no existen personas asignadas | |
| **Condiciones:** No deben existir personas asignadas al proyecto. | |
| **Entrada / Pasos de ejecución:**<br>Opción 1: 1. Seleccionar "Proyecto" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Seleccionar "Equipo" en el menú principal.<br>Opción 2: 1. Seleccionar "Equipo" en el menú principal.<br>2. Seleccionar el proyecto del *select* en la parte superior de la pantalla. | |
| **Resultados Esperados**<br>1. El sistema mostrará el mensaje: "No existen personas asignadas". | |

**F.24 – Prueba Funcional Fallo-Listar Equipo**
Elaboración Propia

| **Caso de Prueba Funcional** | |
|---|---|
| **Código:** 25 | **Historia de Usuario:** Agregar Integrante No:22 |
| **Descripción:** El sistema asignará dos personas al proyecto una existente y una nueva. | |
| **Condiciones de Ejecución:** Debe existir el proyecto al que se le agregará la persona. Se deben llenar todos los campos obligatorios del formulario y seleccionar una persona existente. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla de Equipo.<br>2. Seleccionar una persona existente (haciendo *click* en el *checkbox*).<br>3. Presionar el *link* "Agregar Nuevo".<br>4. Llenar los campos obligatorios del formulario.<br>5. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", los datos serán ingresados en la base de datos y se asignarán ambas personas al proyecto.<br>2. El sistema re direccionará a la pantalla de Listar Equipo, y se verá la prueba con los datos ingresados.<br>3. El sistema mostrará un mensaje de éxito: "Operación Exitosa", en la parte de arriba de la tabla. | |

F.25 – Prueba Funcional Éxito-Agregar Integrante
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 26 | **Historia de Usuario:** Eliminar Integrante No:23 |
| **Descripción:** El sistema quitará la asignación de la persona al proyecto. | |
| **Condiciones de Ejecución:** La persona debe estar asignada al proyecto. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Eliminar", de la persona que se desea retirar del proyecto.<br>2. El sistema mostrará una ventana de validación: "¿Está seguro que desea eliminar esta persona del proyecto?".<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema eliminará la asignación de la persona al proyecto. Se mostrará un mensaje: "Operación Exitosa". | |

**F.26 – Prueba Funcional Éxito-Eliminar Integrante**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 27 | **Historia de Usuario:** Editar Persona No:24 |
| **Descripción:** El sistema editará los datos de la persona seleccionada. | |
| **Condiciones:** Se deben llenar todos los campos obligatorios del formulario. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar", de la persona a ser editada.<br>2. Editar los campos del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Los datos serán editados en la BD.<br>2. Se re direccionará a la pantalla de Listar Equipo, y se verá la persona con los datos editados.<br>4. El sistema mostrará un mensaje de éxito: "Operación Exitosa". | |

**F.27 – Prueba Funcional Éxito-Editar Persona**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 28 | **Historia de Usuario:** Eliminar Persona No:25 |
| **Descripción:** El sistema eliminará la persona de la persona base de datos. | |
| **Condiciones de Ejecución:** La persona no puede estar asignada a ningún proyecto. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la tabla de equipo.<br>2. Presionar el *link* "Eliminar", de la persona que se quiere eliminar.<br>3. El sistema mostrará una ventana de validación: "¿Está seguro que desea eliminar esta persona?".<br>4. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", el sistema eliminará la persona.<br>2. El sistema mostrará el mensaje: "Operación Exitosa". | |

**F.28 – Prueba Funcional Éxito-Eliminar Persona**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 29 | **Historia de Usuario:** Listar Entregas No:44 |
| **Descripción:** El sistema Listará las entregas por proyecto. | |
| **Condiciones de Ejecución:** Deben existir entregas en la base de datos para el proyecto. | |
| **Entrada / Pasos de ejecución:**<br>Opción 1: 1. Seleccionar "Proyecto" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Seleccionar "Entregas" en el menú principal.<br>Opción 2: 1. Seleccionar "Entregas" en el menú principal.<br>2. Seleccionar el proyecto del *select* en la parte superior de la pantalla. | |
| **Resultados Esperados**<br>1. El sistema mostrará las Entregas por proyecto. | |

**F.29 – Prueba Funcional Éxito-Listar Entrega**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 30 | **Historia de Usuario:** Listar Entregas No: 44 |
| **Descripción:** El sistema no listará las entregas por proyecto. | |
| **Condiciones:** No deben existir entregas en la base de datos para el proyecto. | |
| **Entrada / Pasos de ejecución:**<br>Opción 1:<br>1. Seleccionar "Proyecto" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Seleccionar "Entregas" en el menú principal.<br>Opción 2:<br>1. Seleccionar "Entregas" en el menú principal.<br>2. Seleccionar el proyecto del *select* en la parte superior de la pantalla. | |
| **Resultados Esperados**<br>1. El sistema mostrará un mensaje: "No existen entregas de este proyecto". | |

**F.30 – Prueba Funcional Fallo-Listar Entrega**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 31 | **Historia de Usuario:** Crear Entregas No:45 |
| **Descripción:** El sistema agregará la entrega al proyecto. | |
| **Condiciones:** Debe existir el proyecto. Llenar los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Llenar los campos obligatorios del formulario.<br>4. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Se re direccionará a la pantalla de listar Entregas y se verá la nueva entrega ingresada.<br>2. El sistema mostrará un mensaje: "Operación Exitosa". | |

**F.31 – Prueba Funcional Éxito-Crear Entrega**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 32 | **Historia de Usuario:** Crear Entregas No:45 |
| **Descripción:** El sistema no agregará la entrega. | |
| **Condiciones:** Debe existir el proyecto. Dejar en blanco al menos uno los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Dejar en blanco un campo obligatorio.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema mostrará un mensaje: "Debe llenar todos los campos obligatorios". | |

**F.32 – Prueba Funcional Fallo-Crear Entrega**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 33 | **Historia de Usuario:** Editar Entregas No:46 |
| **Descripción:** El sistema editará la entrega. | |
| **Condiciones:** Debe existir el proyecto. Llenar los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Seleccionar el Proyecto.<br>2. Presionar el *link* "Editar" de la entrega a editar.<br>3. Editar los campos del formulario.<br>4. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Se re direccionará a la pantalla de listar Entregas y se verá la entrega editada.<br>2. El sistema mostrará un mensaje: "Operación Exitosa". | |

**F.33 – Prueba Funcional Éxito-Editar Entrega**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 34 | **Historia de Usuario:** Editar Entregas No:46 |
| **Descripción:** El sistema no editará la entrega. | |
| **Condiciones:** Debe existir el proyecto. Dejar en blanco al menos uno de los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar" de la entrega a editar.<br>2. Dejar en blanco uno de los campos obligatorios.<br>1. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema mostrará un mensaje: "Debe llenar los campos obligatorios". | |

**F.34 – Prueba Funcional Fallo-Editar Entrega**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 35 | **Historia de Usuario:** Eliminar Entregas No:47 |
| **Descripción:** El sistema eliminará la entrega. | |
| **Condiciones:** Debe existir la entrega. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Eliminar" de la entrega a que se desea borrar.<br>2. El sistema mostrará una ventana de validación: "¿Está seguro que desea eliminar esta entrega?".<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", el sistema eliminará la entrega.<br>2. El sistema mostrará el mensaje: "Operación Exitosa". | |

**F.35 – Prueba Funcional Éxito-Eliminar Entrega**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 36 | **Historia de Usuario:** Listar Iteraciones No:26 |
| **Descripción:** El sistema listara las iteraciones del proyecto escogido. | |
| **Condiciones:** Debe existir al menos una iteración creada en la base de datos. | |
| **Entrada / Pasos de ejecución:**<br>Opción 1: 1. Seleccionar "Proyecto" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Seleccionar "Planificación" en el menú principal.<br>Opción 2: 1. Seleccionar "Planificación" en el menú principal.<br>2. Seleccionar el proyecto del *select* en la parte superior de la pantalla. | |
| **Resultados Esperados**<br>1. El sistema listara las iteraciones existentes para el proyecto. | |

**F.36 – Prueba Funcional Éxito-Listar Iteraciones**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 37 | **Historia de Usuario:** Listar Iteraciones No:26 |
| **Descripción:** El sistema mensaje de fallo. | |
| **Condiciones:** El proyecto escogido no debe tener iteraciones creadas. | |
| **Entrada / Pasos de ejecución:**<br>Opción 1: 1. Seleccionar "Proyecto" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Seleccionar "Planificación" en el menú principal.<br>Opción 2: 1. Seleccionar "Planificación" en el menú principal.<br>2. Seleccionar el proyecto del *select* en la parte superior de la pantalla. | |
| **Resultados Esperados**<br>1. El sistema mostrara el mensaje: "no existen iteraciones para el proyecto". | |

**F.37 – Prueba Funcional Fallo-Listar Iteraciones**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 38 | **Historia de Usuario:** Ingresar Iteración No:27 |
| **Descripción:** El sistema ingresara la nueva iteración en la BD. | |
| **Condiciones:** Debe haber un proyecto seleccionado. Se deben llenar todos los campos obligatorios del formulario. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Llenar todos los campos obligatorios del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Se re direccionará a la pantalla de listar Iteraciones y se verá la nueva iteración ingresada.<br>2. El sistema mostrará un mensaje: "Operación Exitosa". | |

**F.38 – Prueba Funcional Éxito-Ingresar Iteración**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 39 | **Historia de Usuario:** Ingresar Iteración No:27 |
| **Descripción:** El sistema mostrará un mensaje de error. | |
| **Condiciones:** Debe haber un proyecto seleccionado. Se deben dejar en blanco al menos uno de los campos obligatorios del formulario. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" en la parte inferior de la tabla.<br>2. Dejar en blanco el campo "Duración".<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema mostrará un mensaje: "Debe llenar todos los campos obligatorios". | |

**F.39 – Prueba Funcional Fallo-Ingresar Iteración**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 40 | **Historia de Usuario:** Editar Iteración No:28 |
| **Descripción:** El sistema editara los datos de la iteración en la BD. | |
| **Condiciones:** Debe haber un proyecto seleccionado. Se deben llenar todos los campos obligatorios del formulario. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar" de la iteración que se desea.<br>2. Llenar todos los campos obligatorios del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Se re direccionará a la pantalla de listar Iteraciones y se verá la iteración editada.<br>2. El sistema mostrará un mensaje: "Operación Exitosa". | |

**F.40 – Prueba Funcional Éxito-Editar Iteración**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 41 | **Historia de Usuario:** Editar Iteración No:28 |
| **Descripción:** El sistema editara los datos de la iteración en la BD. | |
| **Condiciones:** Se deben dejar en blanco al menos un campo obligatorio. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar" de la iteración que se desea.<br>2. Dejar en blanco el campo "Duración".<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema mostrará un mensaje: "Debe llenar los campos obligatorios". | |

**F.41 – Prueba Funcional Fallo-Editar Iteración**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 42 | **Historia de Usuario:** Eliminar Iteración No:25 |
| **Descripción:** El sistema eliminará la iteración. | |
| **Condiciones:** Debe existir la entrega. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Eliminar" de la iteración a que se desea borrar.<br>2. El sistema mostrará una ventana de validación: "¿Está seguro que desea eliminar esta iteración?".<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. Al presionar el botón "Aceptar", el sistema eliminará la iteración.<br>2. El sistema mostrará el mensaje: "Operación Exitosa". | |

F.42 – Prueba Funcional Éxito-Eliminar Iteración
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 43 | **Historia de Usuario:** Cambiar Status Iteración No:30 |
| **Descripción:** El sistema cambiara el status de la iteración. | |
| **Condiciones:** Debe existir la iteración. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* del status de la iteración.<br>2. El sistema mostrará una ventana de validación: "¿Está seguro que desea iniciar/culminar esta iteración?".<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema cambiara el status de la iteración.<br>2. El sistema mostrará el mensaje: "Operación Exitosa". | |

F.43 – Prueba Funcional Éxito-Cambiar status Iteración
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 44 | **Historia de Usuario:** Asignar/Editar Peticiones No:31 y 32 |
| **Descripción:** El sistema asociara peticiones a la iteración. | |
| **Condiciones:** Debe existir la iteración, deben existir peticiones creadas. | |
| **Entrada / Pasos de ejecución:**<br>1. Seleccionar "Planificación" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Presionar el *link* del número de la iteración.<br>4. Presionar el botón "Editar" en la parte de abajo de la tabla.<br>5. Seleccionar el *checkbox* de las peticiones a asociar.<br>6. Presionar el botón "Guardar". | |
| **Resultados Esperados**<br>1. El sistema asociara la petición a la iteración.<br>2. El sistema mostrará el mensaje: "Operación Exitosa". | |

**F.44 – Prueba Funcional Éxito-Asignar/Editar Peticiones**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 45 | **Historia de Usuario:** Listar Tarea/ Pruebas Funcionales por Petición No:34 |
| **Descripción:** El sistema listará las tareas y pruebas funcionales por petición. | |
| **Condiciones:** Deben existir peticiones asociadas a la iteración. | |
| **Entrada / Pasos de ejecución:**<br>1. Seleccionar "Planificación" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Presionar el *link* del número de la iteración.<br>4. Presionar el *link* del nombre de la petición. | |
| **Resultados Esperados**<br>1. El sistema mostrará las tareas y pruebas funcionales asociadas a la iteración. | |

**F.45 – Prueba Funcional Éxito-Listar Tareas/Pruebas Funcionales por Peticiones**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 46 | **Historia de Usuario:** Agregar Tarea No:35 |
| **Descripción:** El sistema agregará una tarea a la petición. | |
| **Condiciones:** Deben existir la petición. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" de la parte de abajo de la tabla.<br>2. Llenar todos los campos obligatorios del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema ingresará la tarea en la base de datos.<br>2. Se re direccionará a la pantalla de Listar Tareas, y mostrará un mensaje "Operación exitosa". | |

**F.46 – Prueba Funcional Éxito-Agregar Tarea**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 47 | **Historia de Usuario:** Agregar Tarea No:35 |
| **Descripción:** El sistema no agregará la tarea a la petición. | |
| **Condiciones:** Deben existir la petición. Dejar en blanco al menos un campo obligatorio. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el botón "Agregar" de la parte de abajo de la tabla.<br>2. Dejar en blanco el campo "Tiempo Estimado" del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El Sistema mostrará el mensaje "Debe llenar los campos obligatorios". | |

**F.47 – Prueba Funcional Fallo-Agregar Tarea**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 48 | **Historia de Usuario:** Editar Tarea No:36 |
| **Descripción:** El sistema editará los datos de la tarea. | |
| **Condiciones:** Deben existir la tarea. Llenar todos los campos obligatorios. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar" de la tarea que se desea editar.<br>2. Llenar todos los campos obligatorios del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema editará los datos de la tarea en la base de datos.<br>2. Se re direccionará a la pantalla de Listar Tareas, y mostrará un mensaje "Operación exitosa". | |

**F.48 – Prueba Funcional Éxito-Editar Tarea**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 49 | **Historia de Usuario:** Editar Tarea No:36 |
| **Descripción:** El sistema no editará los datos de la tarea. | |
| **Condiciones:** Deben existir la tarea. Dejar en blanco al menos un campo obligatorio. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Editar" de la tarea que se desea editar.<br>2. Dejar en blanco el campo "Tiempo Estimado" del formulario.<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema mostrará un mensaje "Debe llenar los campos obligatorios". | |

**F.49 – Prueba Funcional Fallo-Editar Tarea**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 50 | **Historia de Usuario:** Eliminar Tarea No:37 |
| **Descripción:** El sistema eliminará la tarea de la base de datos. | |
| **Condiciones:** Deben existir la tarea. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* "Eliminar" de la tarea que se desea eliminar.<br>2. El sistema mostrará una ventana de validación: "¿Está seguro que desea eliminar la tarea?".<br>3. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema eliminará la tarea.<br>2. Se mostrará un mensaje "Operación exitosa". | |

**F.50 – Prueba Funcional Éxito-Eliminar Tarea**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 51 | **Historia de Usuario:** Cambiar status de petición No:33 |
| **Descripción:** Se cambiará el status de la petición: sin Iniciar- En curso -Finalizada. | |
| **Condiciones:** Debe existir la petición. Debe estar iniciada la Iteración. Deben estar finalizadas las tareas y pruebas asignadas. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* del número de la iteración (Módulo de Planificación)<br>2. El sistema mostrará las peticiones por iteración.<br>3. Presionar el *link* Iniciar/Finalizar de la columna status de la petición que se desee.<br>4. El sistema mostrará una ventana de validación: "¿Está seguro que desea iniciar/finalizar la petición?".<br>5. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema cambiará el status de la petición. | |

**F.51 – Prueba Funcional Éxito-Cambiar status Petición**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 52 | **Historia de Usuario:** Cambiar status Tarea No:38 |
| **Descripción:** Se cambiará el status de la Tarea: sin Iniciar- En curso -Finalizada. | |
| **Condiciones:** Debe existir la tarea. Debe estar iniciada la petición a la que pertenece. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar el *link* del número de la iteración.<br>2. Presionar el *link* del nombre de la petición.<br>3. Presionar el *link* Iniciar/Finalizar de la columna status de la tarea que se desee.<br>4. El sistema mostrará una ventana de validación: "¿Está seguro que desea iniciar/finalizar la tarea?".<br>5. Presionar el botón "Aceptar". | |
| **Resultados Esperados**<br>1. El sistema cambiará el status de la tarea. | |

**F.52 – Prueba Funcional Éxito-Cambiar status Tarea**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 53 | **Historia de Usuario:** Listar Horas por Tarea No:43 |
| **Descripción:** Se listará las horas trabajadas de la tarea escogida. | |
| **Condiciones:** Debe existir la tarea. | |
| **Entrada / Pasos de ejecución:**<br>1. En la pantalla de listar tareas por petición, presionar el *link* del nombre la tarea que se desee. | |
| **Resultados Esperados**<br>1. El sistema mostrará una tabla con las horas trabajadas de cada tarea por día tarea. | |

**F.53 – Prueba Funcional Éxito-Listar horas por Tarea**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 54 | **Historia de Usuario:** Ingresar/editar horas por tarea No:42 |
| **Descripción:** Se ingresarán/editarán las horas trabajadas de la tarea escogida. | |
| **Condiciones:** Debe existir la tarea. | |
| **Entrada / Pasos de ejecución:**<br>1. En la pantalla de listar tareas por petición, presionar el *link* del nombre la tarea que se desee.<br>2. Ingresar/Editar las horas trabajadas del día.<br>3. Presionar el *link* guardar. | |
| **Resultados Esperados**<br>1. El sistema ingresará/editara las horas trabajadas del día en la base de datos. | |

**F.54 – Prueba Funcional Éxito-Ingresar/Editar horas por Tarea**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 55 | **Historia de Usuario:** Listar Peticiones por entrega No:48 |
| **Descripción:** El Sistema mostrará las peticiones asociadas a una entrega. | |
| **Condiciones:** Debe existir la entrega. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar "Entregas" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Presionar el *link* de la versión de la entrega que se desee. | |
| **Resultados Esperados**<br>1. El sistema mostrará las peticiones asociadas a la entrega. | |

**F.55 – Prueba Funcional Éxito-Listar peticiones por Entrega**
Elaboración Propia

| Caso de Prueba Funcional | |
|---|---|
| **Código:** 56 | **Historia de Usuario:** Reportes No:49-51 |
| **Descripción:** El Sistema mostrará las tablas y gráficos para cada uno de los reportes. | |
| **Condiciones:** Se debe seleccionar un proyecto. El proyecto debe tener datos agregados. | |
| **Entrada / Pasos de ejecución:**<br>1. Presionar "Reportes" en el menú principal.<br>2. Seleccionar el proyecto.<br>3. Presionar el *link* del reporte que se desee ver.<br>4. Presionar el link mostrar gráfico/mostrar Tabla. | |
| **Resultados Esperados**<br>1. El sistema mostrará las tablas y gráficos correspondientes al reporte escogido. | |

**F.56 – Prueba Funcional Éxito-Visualizar Reportes**
Elaboración Propia

Shinto

# Manual de Usuario

# Índice


Introducción 4

Proyectos 5

- Agregar Proyecto 5
- Editar Proyecto 5
- Eliminar Proyecto 5

Equipo 6

- Editar Personas de un Proyecto 6
- Eliminar una Persona de un Proyecto 6
- Asignar Personas a un Proyecto 6
- Editar/Eliminar Personas no asociadas al Proyecto 7

Peticiones 8

- Agregar Peticiones a un proyecto 8
- Editar Peticiones de un proyecto 8
- Eliminar Peticiones de un proyecto 8

Pruebas 9

- Agregar Pruebas Funcionales asociadas a una Petición 9
- Agregar Pruebas Unitarias asociadas a una Tarea 9
- Editar datos de Pruebas Funcionales/Unitarias 10
- Eliminar Pruebas Funcionales/Unitarias 10

Planificación 10

- Crear Iteración para un proyecto 10

Editar Iteración de un proyecto 11

Eliminar Iteración de un proyecto 11

Iniciar/Finalizar Iteración de un proyecto 11

Asociar Peticiones a una Iteración 11

Iniciar/Finalizar Petición asociada a una Iteración 12

Crear Tareas asociadas a una petición 13

Editar/Eliminar Tareas de una petición 13

Iniciar/Finalizar Tarea 14

Ingresar/Editar horas por Tarea 14

Entregas 15

Crear Entrega para un proyecto 15

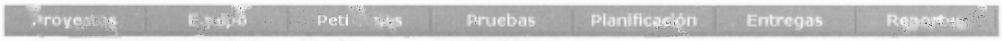
Editar/Eliminar Entrega de un proyecto 15

Visualizar Peticiones asociadas a la Entrega 16

Reportes 16

## Introducción

**Shinto** es una herramienta para el apoyo en la planificación y seguimiento del desarrollo de proyectos. Para la navegación principal cuenta con un menú en la parte superior de la pantalla.




**Proyectos:** manejo de la información básica de los proyectos.

**Equipo:** manejo de los integrantes de los equipos de los proyectos.

**Peticiones:** manejo de las peticiones de un proyecto.

**Pruebas:** manejo la información de pruebas funcionales y unitarias de proyectos.

**Planificación:** manejo la información asociada a las iteraciones.

**Entregas:** manejo de la información de las entregas.

**Reportes:** presenta reportes para el monitoreo del progreso de los proyectos.

Para todas las operaciones de los siguientes módulos se debe seleccionar primero un proyecto, puede hacerlo desde el menú desplegable que está en la parte superior derecha o ir al módulo de proyectos y seleccionar el *checkbox* correspondiente.

## Equipo

### Editar Personas de un Proyecto

1. Presionar el enlace Edit correspondiente a la persona que se desee editar.
2. Editar los datos del formulario.
3. Presionar el botón aceptar.

*Condiciones:* Se deben llenar todos los campos obligatorios. (Éstos presentan un * a su lado).

### Eliminar una Persona de un Proyecto

1. Presionar el enlace Elim correspondiente la persona que se desee eliminar.
2. Presionar el botón aceptar en la ventana de alerta.

### Asignar Personas a un Proyecto

1. Presionar el botón agregar.

2. Seleccionar el *checkbox* correspondiente a la persona que quiere asignar y/o presionar el enlace Agregar Nuevo si ésta no se encuentra en la lista.
3. Éste desplegara un formulario donde puede ingresar los datos básicos de la nueva persona.
4. Presionar el botón aceptar.

*Condiciones:* Se debe seleccionar o agregar al menos una persona. Al agregar una nueva persona se deben llenar todos los campos obligatorios. (Éstos presentan un * a su lado).

### Editar/Eliminar Personas no asociadas al Proyecto

1. En la pantalla de agregar: presionar el enlace Edit o Elim.
2. La opción de *editar* es igual al punto: **Editar Personas de un Proyecto.**
3. En la opción de *eliminar* aparecerá una ventana de alerta donde debe presionar el botón aceptar.

*Condiciones:* Al *editar* se deben llenar todos los campos obligatorios. (Éstos presentan un * a su lado).

Al *eliminar* una persona ésta no puede estar asociada a ningún proyecto.

# Peticiones

## Agregar Peticiones a un proyecto

1. Presionar el botón agregar.
2. Llenar el formulario.
3. Presionar el botón aceptar.

*Condiciones:* Se deben llenar todos los campos obligatorios. (Éstos presentan un * a su lado).

## Editar Peticiones de un proyecto

1. Presionar el enlace Edit correspondiente a la petición que desee editar.
2. Editar los datos del formulario.
3. Presionar el botón aceptar.

*Condiciones:* La opción de edición sólo estará disponible para aquellas peticiones cuyo estatus sea *Por Iniciar*. Se deben llenar todos los campos obligatorios. (Éstos presentan un * a su lado).

## Eliminar Peticiones de un proyecto

1. Presionar el enlace Elim correspondiente a la petición que desee eliminar.
2. Presionar el botón aceptar en la ventana de alerta.

*Condiciones:* Sólo podrán ser eliminadas aquellas peticiones que no hayan sido iniciadas.

## Pruebas

### Agregar Pruebas Funcionales asociadas a una Petición

1. Presionar el botón agregar de la tabla de Pruebas Funcionales.
2. Llenar el formulario.
3. Presionar el botón aceptar.

*Condiciones:* Sólo se podrán agregar pruebas funcionales si existen peticiones asociadas al proyecto seleccionado. Se deben llenar todos los campos obligatorios del formulario. (Éstos presentan un * a su lado).

### Agregar Pruebas Unitarias asociadas a una Tarea

1. Presionar el botón agregar de la tabla de Pruebas Unitarias.
2. Llenar el formulario.

3. Presionar el botón aceptar.

*Condiciones:* Sólo se podrán agregar pruebas unitarias si existen tareas asociadas al proyecto seleccionado. Se deben llenar todos los campos obligatorios del formulario. (Éstos presentan un * a su lado).

### Editar datos de Pruebas Funcionales/Unitarias

1. Presionar el enlace Edit correspondiente a la prueba que se desee editar.
2. Editar los datos del formulario.
3. Presionar el botón aceptar.

*Condiciones:* Se deben llenar todos los campos obligatorios. (Éstos presentan un * a su lado).

### Eliminar Pruebas Funcionales/Unitarias

1. Presionar el enlace Elim correspondiente la prueba que se desee eliminar.
2. Presionar el botón aceptar en la ventana de alerta.

## Planificación

### Crear Iteración para un proyecto

1. Presionar el botón agregar.
2. Llenar el formulario.
3. Presionar el botón aceptar.

*Condiciones:* Se deben llenar todos los campos obligatorios del formulario. (Éstos presentan un * a su lado).

### Editar Iteración de un proyecto

1. Presionar el enlace Edit correspondiente a la iteración que se desee editar.
2. Editar los datos del formulario.
3. Presionar el botón aceptar.

*Condiciones:* Sólo podrán ser editadas aquellas iteraciones que no hayan sido iniciadas. Se deben llenar todos los campos obligatorios. (Éstos presentan un * a su lado).

### Eliminar Iteración de un proyecto

1. Presionar el enlace Elim correspondiente la iteración que se desee eliminar.
2. Presionar el botón aceptar en la ventana de alerta.

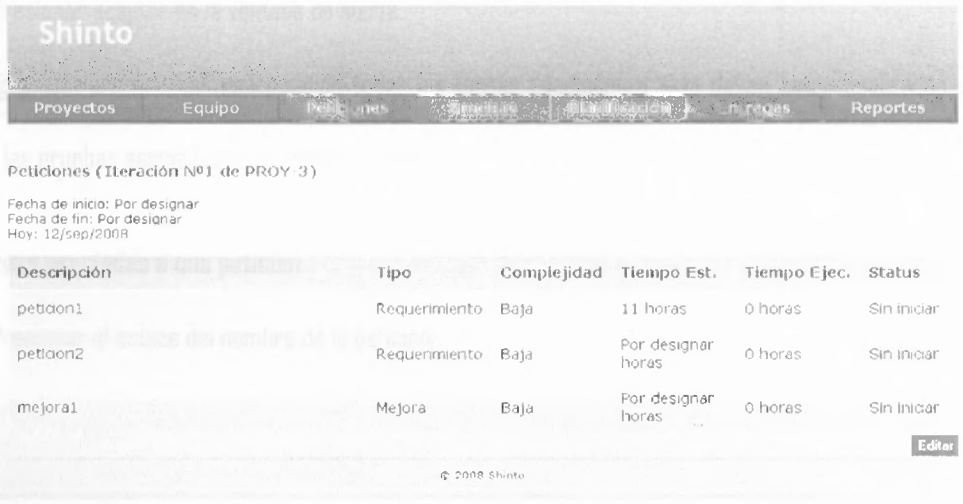
*Condiciones:* Sólo podrán ser eliminadas aquellas iteraciones que no hayan sido iniciadas.

### Iniciar/Finalizar Iteración de un proyecto

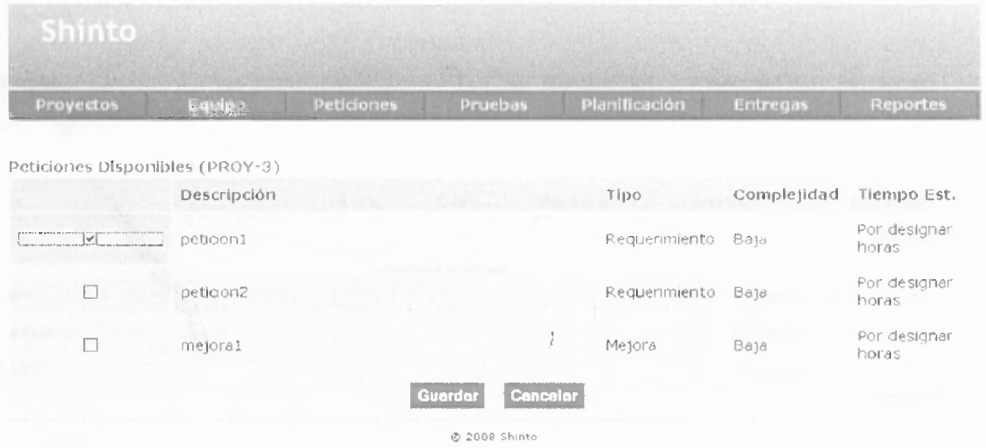
1. Presionar el enlace (Iniciar)/ (Finalizar) correspondiente a la iteración cuyo estatus desee modificar.
2. Presionar aceptar en la ventana de alerta.

## Asociar Peticiones a una Iteración

1. Presionar el enlace del número de la iteración.
2. Presionar el botón Editar en la tabla de peticiones.




3. Seleccionar el *checkbox* de las peticiones que desee asociar a la iteración.




4. Presionar el botón Guardar.

*Condiciones:* Luego de iniciada/finalizada la iteración no se podrá editar sus peticiones asociadas.

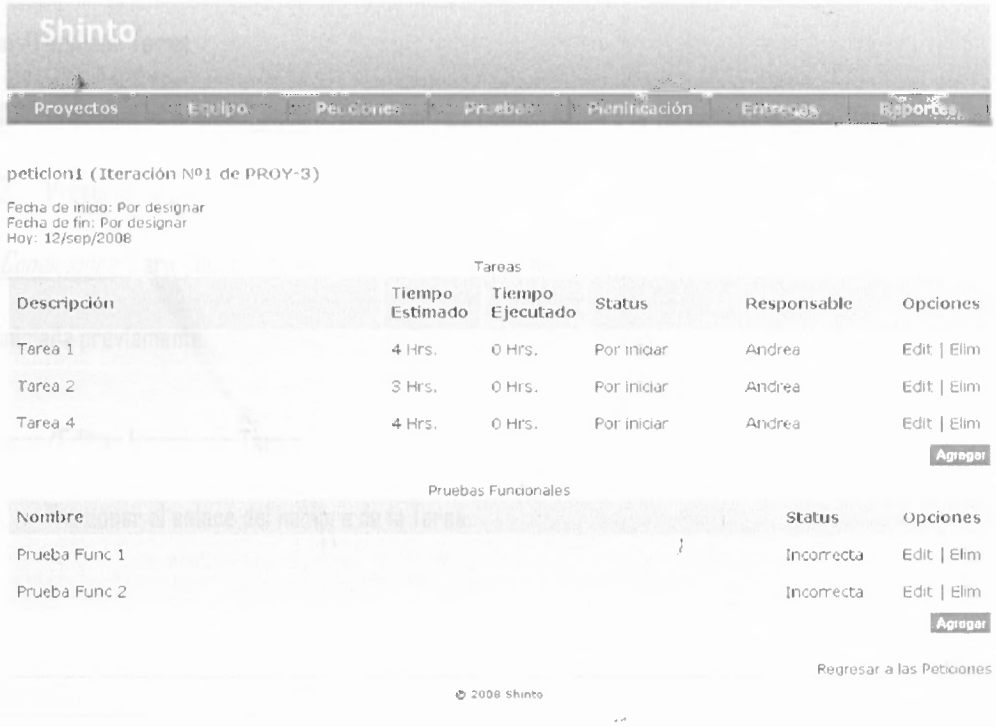
### Iniciar/Finalizar Petición asociada a una Iteración

1. Presionar el enlace (Iniciar)/ (Finalizar) correspondiente la petición cuyo estatus desee modificar.
2. Presionar aceptar en la ventana de alerta.

*Condiciones*: Para finalizar una petición todas las tareas asociadas a ésta deben estar finalizadas y todas las pruebas asociadas deben estar correctas.

### Crear Tareas asociadas a una petición

1. Presionar el enlace del nombre de la petición.




2. Presionar el botón Agregar en la tabla de Tareas.
3. Llenar el formulario.
4. Presionar el botón aceptar.

*Condiciones:* Sólo se podrán asociar nuevas tareas a aquellas peticiones que no hayan sido finalizadas.

Se deben llenar todos los campos obligatorios del formulario. (Éstos presentan un * a su lado).

### Editar/Eliminar Tareas de una petición

1. Presionar el enlace Edit o Elim correspondiente a la tarea según la operación que desee realizar.
2. Para la opción editar: modificar los datos que desee del formulario y presionar el botón aceptar.

   Para la opción eliminar: presionar Aceptar en la ventana de alerta.

*Condiciones:* Sólo se podrán editar o eliminar aquellas tareas cuyo estatus sea *Por Iniciar*. Al editar se deben llenar todos los campos obligatorios. (Éstos presentan un * a su lado).

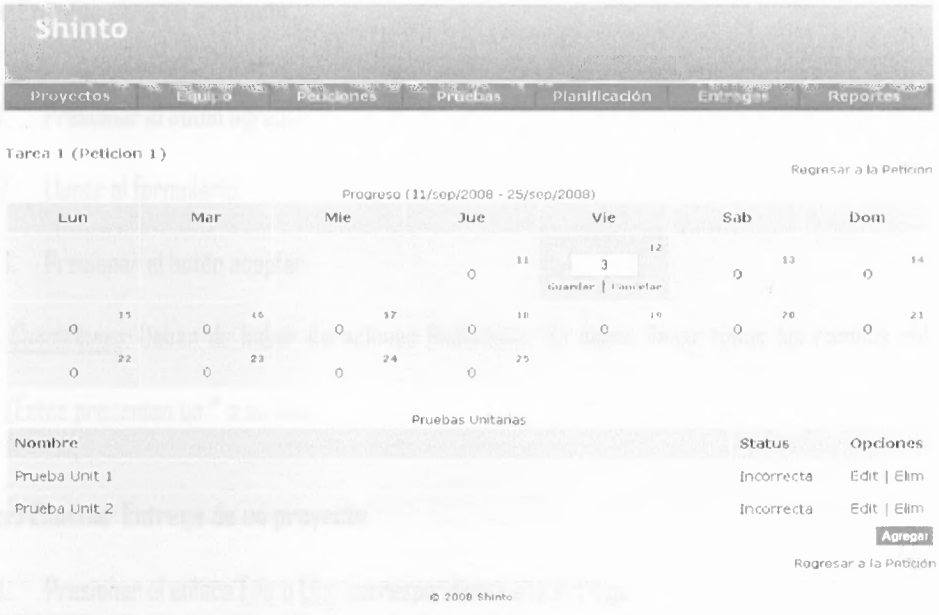
### Iniciar/Finalizar Tarea

1. Presionar el enlace (Iniciar)/ (Finalizar) correspondiente a la Tarea cuyo estatus desee modificar.
2. Presionar aceptar en la ventana de alerta.

*Condiciones:* Para poder modificar el estatus de una tarea, la petición a la que esté asociada debe ser iniciada previamente.

### Ingresar/Editar horas por Tarea

1. Presionar el enlace del nombre de la Tarea.




2. Hacer doble *click* sobre el día cuyas horas desee Ingresar/Editar.

3. Presionar el enlace **Guardar.**

*Condiciones:* Para poder ingresar/editar las horas de una tarea, la misma debe ser iniciada previamente.

## Entregas

### Crear Entrega para un proyecto

1. Presionar el botón agregar.
2. Llenar el formulario.
3. Presionar el botón aceptar.

*Condiciones:* Deben de haber iteraciones finalizadas. Se deben llenar todos los campos obligatorios. (Éstos presentan un * a su lado).

### Editar/Eliminar Entrega de un proyecto

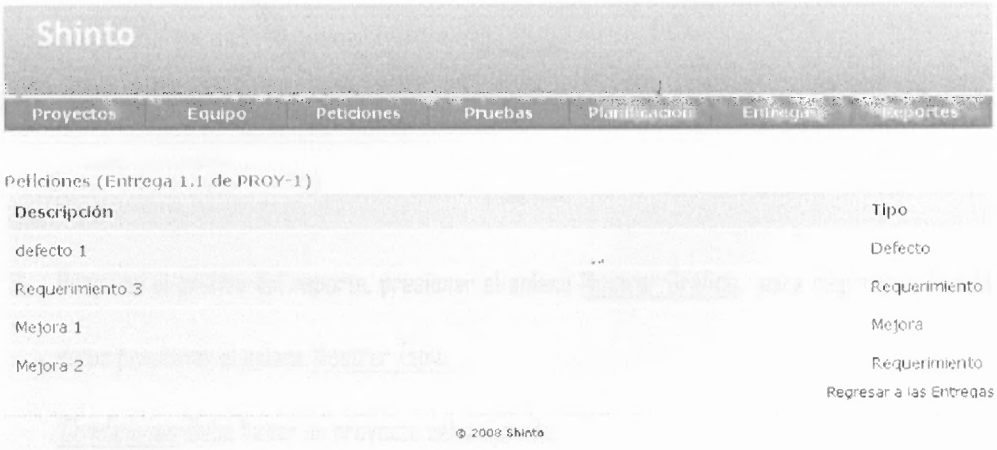
1. Presionar el enlace Edit o Elim correspondiente a la entrega.
2. Para la opción editar: modificar los datos del formulario y presionar el botón aceptar.

   Para la opción eliminar: presionar Aceptar en la ventana de alerta.

*Condiciones:* Al editar se deben llenar todos los campos obligatorios. (Éstos presentan un * a su lado).

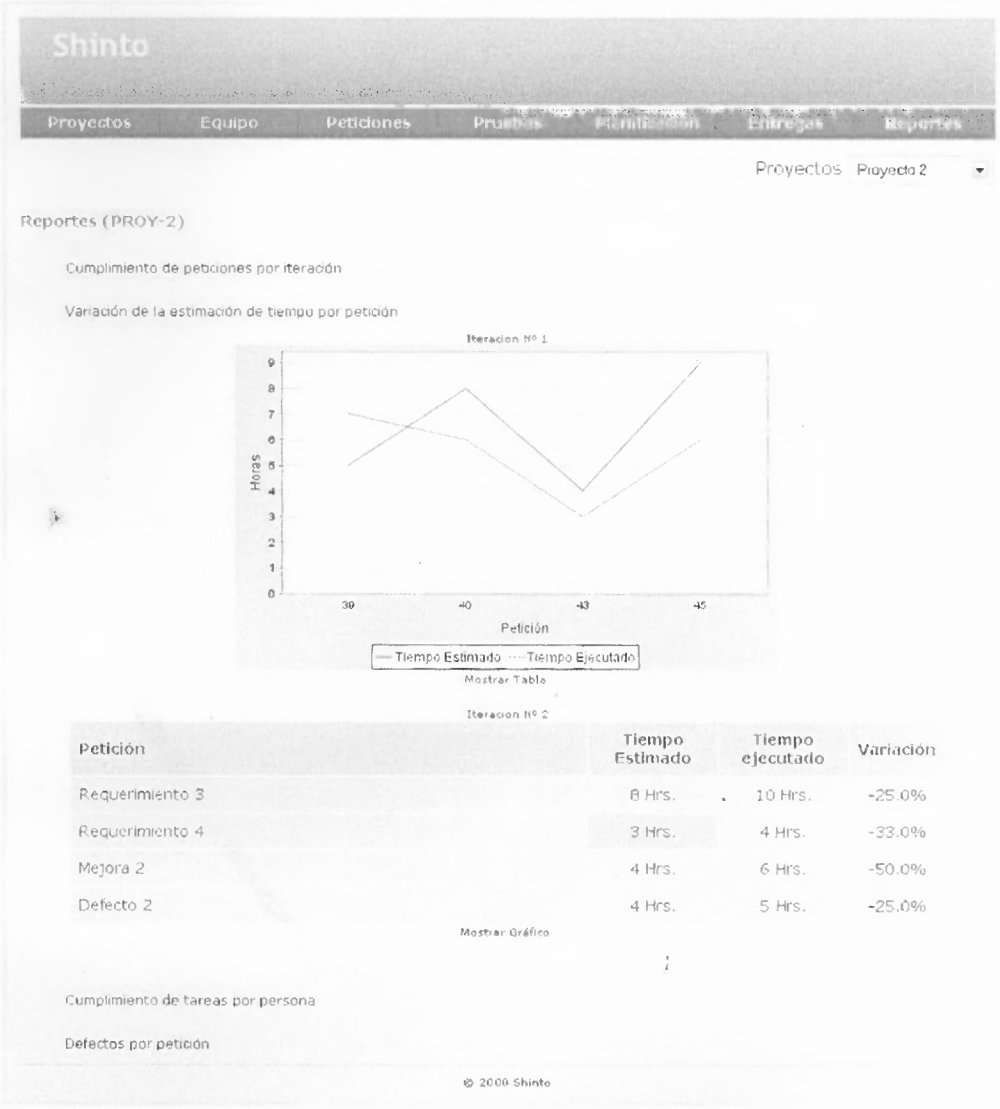
### Visualizar Peticiones asociadas al a Entrega

1. Presionar el enlace de la versión de la entrega cuyas peticiones desea visualizar.



# Reportes

1. Presionar el enlace del reporte que desea ver.




2. Para ver el gráfico del reporte, presionar el enlace Mostrar Gráfico, para regresar a la tabla de datos presionar el enlace Mostrar Tabla.

   *Condiciones:* Debe haber un proyecto seleccionado.