

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Trabajo dirigido

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Tutor

Ingeniero Alexander García Pérez

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

DEDICO ESTE LOGRO:

A Dios que me bendijo con el don de la constancia y la perseverancia, además por poner en mi camino la ingeniería de sistema, porque atreves de ella puedo explotar las habilidades y destrezas que él me brindo.

A mi madre Luz Jhomary Escobar a la que le herede el ingenio y el amor por el estudio, por tanto amor y apoyo incondicional que me ha dado, a ella le dedico este triunfo con todo el amor del mundo.

A mi abuela Carmen Díaz que está en el cielo, a ella le debo todo lo que soy.

A mi hermana Ximena Buitrago futura ingeniera de sistemas, diamante en bruto, la razón de luchar por un mañana.

A mi hermano Leonardo Restrepo un gran hombre que fue testigo de tantos esfuerzos y sacrificios. Mi Dios no me dio un padre pero me dio el mejor hermano del mundo.

A mi pareja Has Tirajo, por brindarme su amor, su apoyo incondicional en los momentos más difíciles de mi vida, por la confianza puesta en mí.

AGRADECIMIENTOS

Me place agradecer:

Al ingeniero Alexander García Pérez quien fue el gestor de la idea de proyecto, por su ayuda y colaboración en el desarrollo del proyecto y su apoyo incondicional durante la carrera.

Al cuerpo de docentes y a la universidad EAN, por compartir sus conocimientos y brindarnos una educación de alta calidad.

TABLA DE CONTENIDO

1	IDENTIFICACIÓN DEL PROBLEMA	9
1.1	OBJETIVO GENERAL	9
1.2	OBJETIVOS ESPECÍFICOS	9
2	JUSTIFICACIÓN	10
3	ALCANCE	11
4	MARCO DE REFERENCIA	12
4.1	MARCO TEÓRICO	12
4.2	MARCO CONCEPTUAL	13
4.2.1	<i>Metodología de desarrollo de software</i>	13
4.2.2	<i>Arquitectura de aplicaciones web ASP.Net a N Capas</i>	14
4.2.3	<i>Biblioteca de vínculos dinámicos (DLL, Dynamic-Link Library)</i>	15
5	METODOLOGÍA	16
5.1	ANÁLISIS	16
5.2	DISEÑO	17
5.3	CODIFICACIÓN	17
5.4	PRUEBAS	17
5.5	INSTALACIÓN	18
5.6	EXPLOTACIÓN	18
5.7	MANTENIMIENTO	19
6	CRONOGRAMA	19
7	PRESUPUESTO	21
8	DESARROLLO	22
8.1	ANÁLISIS	22
8.1.1	<i>Recolección de información mediante entrevista</i>	22
8.1.2	<i>Análisis de requisito</i>	24
8.1.2.1	Definir los requisitos del software	24
8.1.2.1.1	<i>Requerimientos de interfaz</i>	25
8.1.2.1.2	<i>Requerimientos no funcionales</i>	25
8.1.2.1.3	<i>Requerimientos funcionales</i>	26
8.1.3	<i>Viabilidad</i>	46
8.2	DISEÑO	48
8.2.1	<i>Casos de Uso</i>	48
8.2.1.1	Caso de uso administrar proyectos	49
8.2.1.2	Caso de uso asociar estudiante a proyecto	51
8.2.1.3	Caso de uso administrar tareas	54
8.2.1.4	Caso de uso asociar estudiante a tarea	57
8.2.1.5	Caso de uso administrar estado de las tareas	59
8.2.1.6	Caso de uso revisar avances de tareas del proyecto	62
8.2.1.7	Caso de uso consultar tareas de un proyecto	64
8.2.1.8	Caso de uso enviar avances de tareas de un proyecto	66
8.2.1.9	Caso de uso consultar estado de los avances de tareas de un proyecto	68
8.2.1.10	Caso de uso enviar alerta de estados de los avances de tareas de un proyecto	70

8.2.2	<i>Integración Casos de Uso ProjectTracker</i>	72
8.2.3	<i>Diagramas de Clase</i>	103
8.2.3.1	Capa de presentación	103
8.2.3.2	Capa EntityLayerProjectTracker	104
8.2.3.3	Capa BusinessLogicLayerProjectTracker	105
8.2.3.4	Capa DataAccessLayerProjectTracker	107
8.2.4	<i>Modelo Entidad Relación</i>	111
8.2.5	<i>Diccionario de datos de la base de datos Project Tracker</i>	118
8.3	ROLES DE USUARIO	118
8.4	CODIFICACIÓN	121
8.4.1	<i>Prototipo</i>	125
8.5	PRUEBAS.....	127
8.6	INSTALACIÓN.....	130
9	CONCLUSIONES	133
10	LISTA DE REFERENCIAS	135

TABLA DE ILUSTRACIONES

ILUSTRACIÓN 1. ARQUITECTURA DE APLICACIONES WEB ASP.NET A N CAPAS	15
ILUSTRACIÓN 2. DIAGRAMA DE GANTT.....	20
ILUSTRACIÓN 3. CRONOGRAMA - LÍNEA DEL TIEMPO	20
ILUSTRACIÓN 4. CASOS DE USO PT- 007 Y PT- 008	51
ILUSTRACIÓN 5. CASOS DE USO PT-010 Y PT-009	57
ILUSTRACIÓN 6. CASO DE USO PT-011.....	61
ILUSTRACIÓN 7. CASO DE USO PT-018.....	64
ILUSTRACIÓN 8. CASO DE USO PT-019.....	66
ILUSTRACIÓN 9. CASO DE USO PT-020.....	68
ILUSTRACIÓN 10. CASO DE USO PT-021.....	70
ILUSTRACIÓN 11. CASO DE USO PT-022	72
ILUSTRACIÓN 12. INTEGRACIÓN CASOS DE USO PROJECTTRACKER.....	102
ILUSTRACIÓN 13. CLASE DE LA CAPA ENTITYLAYERPROJECTTRACKER	105
ILUSTRACIÓN 14. CLASE DE LA CAPA BUSINESSLOGICLAYERPROJECTTRACKER	106
ILUSTRACIÓN 15. CLASES DATAACCESSLAYERPROJECTTRACKER.....	108
ILUSTRACIÓN 16. DIAGRAMA DE DEPENDENCIA DE ENSAMBLADOS	109
ILUSTRACIÓN 17. DIAGRAMA DE CLASES.....	110
ILUSTRACIÓN 18. MODELO ENTIDAD RELACIÓN	117
ILUSTRACIÓN 19. CONFIGURACIÓN DE ACCESO.....	121
ILUSTRACIÓN 20. PÁGINA PRINCIPAL	125
ILUSTRACIÓN 21. PÁGINA PARA LA CREACIÓN Y EDICIÓN DE USUARIO	126
ILUSTRACIÓN 22. PÁGINA QUE MUESTRA EN UN GRÁFICO EL PROGRESO DE CADA TAREA DE UN PROYECTO	127
ILUSTRACIÓN 28. PRUEBA DE LA APLICACIÓN.....	127
ILUSTRACIÓN 29. DATOS ENLAZADOS CORRECTAMENTE	128
ILUSTRACIÓN 30. RESULTADOS DE LA PRUEBA	128
ILUSTRACIÓN 38. VIDEO TUTORIAL PARA LA CREACIÓN DE TAREAS DE UN PROYECTO	130

ÍNDICE DE TABLAS

TABLA 1. PRESUPUESTO PARA EL DESARROLLO DE LA APLICACIÓN	21
TABLA 2. REQUERIMIENTOS NO FUNCIONALES	25
TABLA 3. PLANTILLA DE REQUERIMIENTOS FUNCIONALES	27
TABLA 4. REQUERIMIENTO AUTENTICACIÓN DE USUARIOS	27
TABLA 5. REQUERIMIENTO ADMINISTRACIÓN DE USUARIOS	28
TABLA 6. REQUERIMIENTO ASIGNACIÓN DE ROLES	30
TABLA 7. REQUERIMIENTO ADMINISTRACIÓN DE FACULTADES Y PROGRAMAS	30
TABLA 8. REQUERIMIENTO LOG DE ERRORES	31
TABLA 9. ADMINISTRACIÓN DE PROYECTOS	32
TABLA 10. REQUERIMIENTO ADMINISTRACIÓN DE TAREAS	33
TABLA 11. REQUERIMIENTO ADMINISTRACIÓN DE ESTADO DE LAS TAREAS	34
TABLA 12. REQUERIMIENTO CONSULTAR DE GRÁFICOS DE ESTADO DE TAREAS	35
TABLA 13. REQUERIMIENTO CONSULTA DE DATOS DEL PROYECTO	36
TABLA 14. REQUERIMIENTO CIERRE DEL PROYECTO	37
TABLA 15. REQUERIMIENTO CONSULTA DE REPORTE DEL ESTADO DEL PROYECTO	38
TABLA 16. REQUERIMIENTO CONSULTA DE REPORTE DE HISTORIAL DE PROYECTOS POR TUTOR	39
TABLA 17. CONSULTAR REPORTE DE HISTORIAL PROYECTOS POR PROGRAMA	40
TABLA 18. REQUERIMIENTO REVISIÓN DE AVANCES DE TAREAS DEL PROYECTO	41
TABLA 19. REQUERIMIENTO CONSULTA DE TAREAS DE UN PROYECTO	42
TABLA 20. REQUERIMIENTO ENVÍO DE AVANCES DE TAREAS DEL PROYECTO	42
TABLA 21. REQUERIMIENTO CONSULTA DE AVANCES DE TAREAS DE UN PROYECTO	43
TABLA 22. REQUERIMIENTO ENVÍO DE ALERTA DE ESTADOS DE LOS AVANCES DE TAREAS DE UN PROYECTO	44
TABLA 23. REQUERIMIENTO DIAGRAMA DE GANTT	45
TABLA 24. VIABILIDAD DEL PROYECTO	46
TABLA 25. CASO DE USO ADMINISTRAR PROYECTOS	49
TABLA 26. CASO DE USO ASOCIAR ESTUDIANTE A PROYECTO	52
TABLA 27. CASO DE USO ADMINISTRAR TAREAS	54
TABLA 28. CASO DE USO ASOCIAR ESTUDIANTE A TAREA	58
TABLA 29. CASO DE USO ADMINISTRAR ESTADO DE LAS TAREAS	59
TABLA 30. CASO DE USO REVISAR AVANCES DE TAREAS DEL PROYECTO.	62
TABLA 31. CASO DE USO CONSULTAR TAREAS DE UN PROYECTO	64
TABLA 32. CASO DE USO ENVIAR AVANCES DE TAREAS DE UN PROYECTO	66
TABLA 33. CASO DE USO CONSULTAR ESTADO DE LOS AVANCES DE TAREAS DE UN PROYECTO	69
TABLA 34. CASO DE USO ENVIAR ALERTA DE ESTADOS DE LOS AVANCES DE TAREAS DE UN PROYECTO	71
TABLA 35. ROLES DE USUARIO	118
TABLA 36. CASO DE PRUEBA	128
TABLA 37. REQUISITOS MÍNIMOS PARA EL FUNCIONAMIENTO DE LA APLICACIÓN	131

1 IDENTIFICACIÓN DEL PROBLEMA

Actualmente en la universidad EAN no existe alguna aplicación que lleve el control sobre los avances y las tutorías para los proyectos de grado. En algunos casos el control se lleva mediante hojas de Excel, pero en la mayoría de casos ni siquiera se lleva algún control, en consecuencia no se puede obtener una trazabilidad, ni reportes de los progresos y tiempos utilizados en la ejecución de los proyectos, además esto expone los datos a libre manipulación, se desajustan los cronogramas de trabajo y en algunos casos los estudiantes exceden el tiempo máximo de entrega del proyecto, por lo tanto no se pueden evaluar concretamente el estado del proyecto en un tiempo determinado. Por otro lado, se han presentado problemas entre los estudiantes y los tutores del proyecto; en dichas situaciones no se ha podido concluir el responsable debido a que no hay argumentos válidos que sustenten las fallas presentadas en los avances por parte de los estudiantes o del tutor.

OBJETIVOS

1.1 Objetivo general

Crear un prototipo Web para manejar el proceso de proyectos de grado de los estudiantes de la Universidad EAN.

1.2 Objetivos específicos

- Realizar la recolección de información para identificar las necesidades del cliente.
- Definir la lista de requerimientos solicitados por el cliente.

- Modelar los diagramas UML y la base de datos.
- Definir de los roles usuario de la aplicación.
- Desarrollar del prototipo de la aplicación web.

2 JUSTIFICACIÓN

Se creara una aplicación para el control de los avances de los proyectos de grado, esta se basará en primera instancia en el análisis del proceso de evaluación del desarrollo de los proyectos que existe actualmente en la universidad EAN, mediante dicho análisis se obtendrán las fuentes del levantamiento de información para el desarrollo de la aplicación, además es evidente que es esta aplicación es una necesidad latente para la facultad de ingeniería de la universidad EAN y a futuro seria de utilidad para las demás facultades.

Esta idea nació de una propuesta realizada por el comité de trabajos de grado de la facultad de ingeniería, la cual describía el problema que tiene la universidad para llevar el seguimiento de los proyecto de grado.

Con la implantación de esta aplicación se puede obtener una trazabilidad de proceso de desarrollo de los proyectos, también a través de la aplicación web se descentraliza el seguimiento de los proyecto, esto sería una ventaja para las partes implicadas en el proyecto, ya que se tendría acceso a la información del proyecto en tiempo real.

Además es un gran impacto tecnológico para las facultades no relacionadas con la de ingeniería, por otro lado, a través de este desarrollo demostramos las competencias adquiridas

durante el proceso de formación como ingeniero de sistemas. Es de resaltar que se obtendrá un mejoramiento en los procesos de la universidad mediante la implementación de TIC.

3 ALCANCE

Se desarrollará un prototipo funcional y acorde a los objetivos planteados, pero es de aclarar que no será implementado en los servidores de la universidad, por lo contrario se publicara la aplicación en un Hosting privado o en un equipo local de la universidad EAN, para que pueda ser utilizado por la facultada de ingeniería, en consecuencia no se generaran costos para la universidad.

Se entregaran video tutoriales donde se explicara el funcionamiento de la aplicación, por lo tanto no se harán manuales de usuario.

El código fuente tendrá los comentarios respectivos y estará regido por las reglas y métricas de código de Microsoft Visual Studio 2010, esto indica que no se hará manual de código fuente para el desarrollador. Por otro lado, luego de la instalación y entrega de la aplicación no se realizara mantenimientos o ajustes de código, para estos efectos se entregara el código fuente de la aplicación para posteriores cambios, correcciones o adiciones de nuevas funcionalidades.

4 MARCO DE REFERENCIA

4.1 Marco teórico

Al desarrollar este proyecto la universidad EAN tendrá una aplicación web con la cual podrá llevar un control y seguimiento en los avances de los proyectos de grado, esta será una gran herramienta para evaluar el desempeño de los estudiantes y tutores frente a desarrollo de dichos proyectos.

Para lograr los objetivos planteados, se debe diseñar un software de alta calidad, para esto es necesario implementar una metodología de desarrollo acorde al equipo de trabajo y a la envergadura del software a crear, también se debe tener en cuenta los cambios y problemas comunes a los que está atado un desarrollo de software, para garantizar dicha calidad se debe acudir a una metodología ágil y enfocada a el desarrollo personal de software o a equipos pequeños de desarrollo, ya que este desarrollo será realizado por un solo integrante.

En la actualidad existen muchas metodologías para el desarrollo de software tales como XP, Scrum, MSF, PSP y TSP entre otras, es de resaltar que casi todas las metodologías se centran o tienen ítems similares, por tal razón, se utilizara una metodología de desarrollo personal, la cual abstrae los principales pasos utilizados para el desarrollo de software.

Dichos pasos serán citados y explicados en el punto de metodología.

Estos son:

- Análisis
- Diseño
- Codificación
- Pruebas
- Instalación
- Explotación
- Mantenimiento

4.2 Marco conceptual

4.2.1 Metodología de desarrollo de software.

El propósito principal de la metodología de desarrollo de software es conducir y facilitar la creación del software a través de técnica y procedimientos estructurados, pero vale resaltar que cada proyecto de construcción de software requiere una metodología acorde a sus necesidades, objetivos y estructura, dicha metodología debe ofrecer que hacer, como, cuando, quien debe hacerlo y también determina las etapas del ciclo de vida del desarrollo de software.

A través de la evolución del software estas metodologías han tenido una gran evolución arrancando desde las prácticas artesanales, pasando por el desarrollo estructurado hasta el desarrollo orientado a objetos. (Prieto Tinoco & Cortes Leiva, 2006)

4.2.2 *Arquitectura de aplicaciones web ASP.Net a N Capas*

Para el desarrollo del proyecto se utilizara un patrón de arquitectura común para crear aplicaciones ASP.Net.

Se implementará una arquitectura conocida como 4-Tier o arquitectura a cuatro capas, la primera capa es la de presentación “Presentation Layer”, dicha capa esencialmente está compuesta por la interfaz del usuario o interfaz web donde se realiza la captura y visualización de datos, luego viene la capa de negocios o lógica “Business Logic Layer”, esta sirve como conector para el proceso, aplicación de algoritmos e intercambio de datos entre la capa de presentación y la capa que interactúa con la base de datos. La capa de acceso a datos “Data Access Layer”, se encarga de interactuar con la base de datos, entre dicha interacciones se puede resaltar comandos como SELECT, INSERT, UPDATE y DELETE, etc.

Por ultimo está la capa de entidades llamada “Entity Layer”, esta capa sirve para intercomunicar las tres capas anteriores. Cualquiera de las cuatro capas, incluye ella misma puede hacer referencia a sus entidades a través de objetos para transportar los datos entre ellas.

De forma explícita una clase de la capa “Entity Layer” es una entidad que representa una tabla de la base de datos y cada atributo de la entidad es un campo de la tabla a la que se hace referencia. (Microsoft Corporation, 2012b)

A continuación se ilustra la relación de las capas de la arquitectura del proyecto descrita anteriormente (véase Ilustración 1).

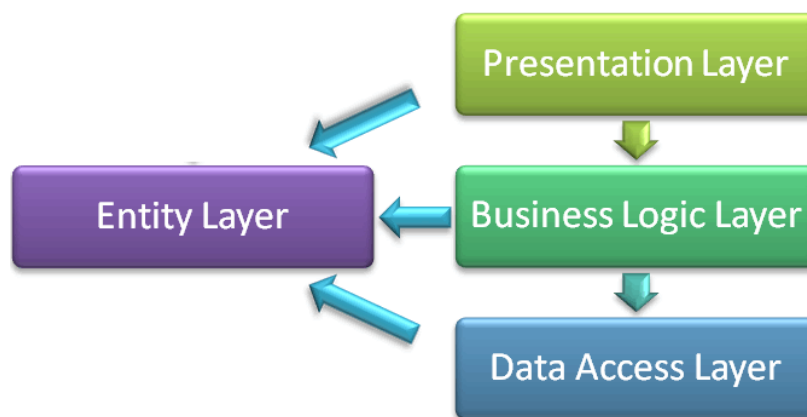


Ilustración 1. Arquitectura de aplicaciones web ASP.Net a N Capas

4.2.3 Biblioteca de vínculos dinámicos (DLL, Dynamic-Link Library)

La DLL comúnmente se conoce como librería de clases, en esta se almacena una serie de clases con cierta variedad de atributos, propiedades, métodos e interfaces entre otros, que pueden ser compartidos y reutilizados por otros proyectos o aplicaciones haciendo referencia a dicho archivo y consecuentemente a las clases que con tiene. (Microsoft Corporation, 2012a)

Un ejemplo claro son las referencias que se hacen entre las diferentes capas utilizadas dentro del proyecto Project Tracker, al generar cada solución asociada a una capa este produce un archivo de extensión .dll que podrá ser agregado y reutilizado, un ejemplo claro es la dll generada por la capa Entity Layer; de forma general esta es referenciada y luego se utilizan sus clases para tener acceso a las entidades para transporta la información a través de objetos entre las capas de Presentation Layer, Business Logic Layer y Data Access Layer.

5 METODOLOGÍA

Para este proyecto se aplicó una metodología que está basada en la ciclo de vida de la metodología clásica también conocida como cascada compuesta principalmente por 7 pasos, estos son análisis, diseño, codificación, pruebas, instalación, explotación y mantenimiento. Esta metodología es apropiada para el proyecto ya que los requerimientos son establecidos al inicio y no cambian durante el desarrollo del proyecto. (Prieto Tinoco & Cortes Leiva, 2006)

5.1 Análisis

En esta etapa, se recopila la información sobre los métodos, procesos o sistemas utilizados actualmente por el cliente para solventar las necesidades que quiere satisfacerse con el nuevo software, también se describen sus fallas, errores, deficiencias y posibles mejoras. La captura de esta información en primera instancia, debe ser mediante técnicas de recolección de información, por ejemplo, se puede realizar reuniones con el cliente o las personas que interactúan con dicho procesos donde se utilizan entrevistas o prototipos donde se captura y especifica los requerimientos o requisitos del software a desarrollar.

Por otro lado, también es muy importante definir los recursos necesarios para el desarrollo del proyecto tales como tiempo, herramientas, recursos humanos, presupuestos, entre otros.

Al final de esta etapa la especificación de requisitos de software deben ser revisados y aprobados por el cliente para continuar con la siguiente etapa del desarrollo. (Prieto Tinoco & Cortes Leiva, 2006)

5.2 Diseño

La finalidad de esta fase es lograr el diseño arquitectónico de la aplicación y la estructura de la base de datos. En base a los requisitos identificados anteriormente se procede a modelar el software a través de técnicas o herramientas tales como UML y el Modelo Entidad Relación.

Una buena aproximación a la definición de la etapa de diseño sería el proceso de traducción de los requisitos en una representación gráfica detallada que da un soporte y una guía a la etapa de codificación. (Prieto Tinoco & Cortes Leiva, 2006)

5.3 Codificación

En esta fase se debe realizar una traducción a un lenguaje que la maquina o computadora entienda, con el fin de esta realice una o varias tareas que produzcan una salida que cumplan con los requerimientos del cliente obtenidos en la etapa de análisis junto con los diagramas y datos obtenidos en la etapa de diseño. Esta codificación se debe realizar con estándares, documentación y métricas de alta calidad, a su vez debe acudir a la simplicidad y sencillez para optimizar los tiempos de respuesta de esta forma se garantiza el buen funcionamiento de la aplicación y el cumplimiento de los objetivos del proyecto. (Prieto Tinoco & Cortes Leiva, 2006)

En este proyecto se utiliza el lenguaje C# asp.net con Visual Studio 2010 en Framework 4 y la base de datos en MySQL 5.5.

5.4 Pruebas

Según (Prieto Tinoco & Cortes Leiva, 2006) “Una prueba del software debe ser lo suficientemente flexible para promover la creatividad y la adaptabilidad del software, y lo suficientemente rígida para permitir un buen seguimiento de la misma”.

Esencialmente en esta etapa se mide la calidad, también se verifica y valida el software, esto se hace mediante pruebas programadas, compuestas por las fases de planificación, diseño y ejecución, dichas pruebas se centran en comprobar que la aplicación cumpla con los requisitos solicitados.

Esta etapa puede ser ejecutada cada vez que se termine la codificación de un requerimiento o si se desea al final de toda la codificación, básicamente las pruebas consisten en ingresar una entrada que produzca uno o varios resultados que satisfagan el requerimiento asociado; finalmente se recopilan los datos obtenidos de las pruebas y se evalúan los resultados.

5.5 Instalación

Para implementar esta fase se debe analizar los resultados obtenidos de la etapa de pruebas, luego se debe corregir los errores, realizar mejoras o cambios sobre la aplicación acorde a dicho análisis.

En esta etapa el software se instala en el ambiente de producción, también se implementan de ser necesario el hardware y software adicional que se necesite para el funcionamiento de nueva aplicación. A lo largo de esta etapa se documenta los posibles errores que ocurran y las mejoras o cambios solicitados por el cliente para la aplicación, tanto en diseño, como en la lógica del negocio. Al finalizar esta etapa se debe crear un documento donde se especifique el estado del software y la aprobación del cliente frente a los requisitos solicitados. Esta etapa, no será implementada en el proyecto.

5.6 Explotación

El objetivo principal de esta etapa es que el usuario final o cliente se haga cargo por completo de la operación del software. En esta fase el usuario ha sido capacitado para utilizar y

operar la nueva aplicación, además cuenta con un soporte de cómo utilizar el software, para esto comúnmente se utilizan manuales de usuario o video tutoriales de la aplicación. (Prieto Tinoco & Cortes Leiva, 2006)

5.7 Mantenimiento

En muchos casos el software requiere ajuste o correcciones de errores, también en ocasiones las funciones que presta el software, no cumple con las expectativas de cliente y se debe acudir a mejoras o cambio en la aplicación, todo esto puede ser producido por la implementación de nuevo hardware, cambio en las interfaces externas a las que se hace referencia dentro del proyecto o por cambios, o nuevas necesidades del cliente. A esta etapa se le denomina “Mantenimiento” y muy ligada a la etapa de codificación, ya que si se tiene un código fuente de alta calidad se obtendrá una buena escalabilidad del software. (Prieto Tinoco & Cortes Leiva, 2006)

Para el desarrollo del proyecto esta fase no se tendrá en cuenta, pero es de aclarar que para efectos de algún mantenimiento se entregara el código fuente de la aplicación y el script de la base de datos.

6 CRONOGRAMA

Las siguientes imágenes contienen el cronograma de trabajo para el desarrollo de la aplicación, en estas se describen las tareas asociadas al plan de desarrollo propuesto en la metodología de trabajo (véase Ilustración 2 y 3).

Gantt project		
Nombre	Fecha de inicio	Fecha de fin
Analisis	16/03/12	20/04/12
Identificación de las necesidades del cliente	16/03/12	30/03/12
Estudio de viabilidad	30/03/12	3/04/12
Análisis económico	3/04/12	7/04/12
Análisis técnico	6/04/12	10/04/12
Modelado del sistema	9/04/12	20/04/12
Diseño	20/04/12	27/04/12
Diseño Modelo E/R	20/04/12	24/04/12
Estilos e Interfaz Gráfica	23/04/12	27/04/12
Codificación	27/04/12	29/10/12
Pruebas	29/10/12	1/11/12
Instalación	1/11/12	9/11/12
Puesta en marcha	1/11/12	8/11/12
Evaluación y ajuste	5/11/12	8/11/12
Informe final	8/11/12	9/11/12
Entrega del prototipo de la App	8/11/12	9/11/12

Ilustración 2. Diagrama de Gantt

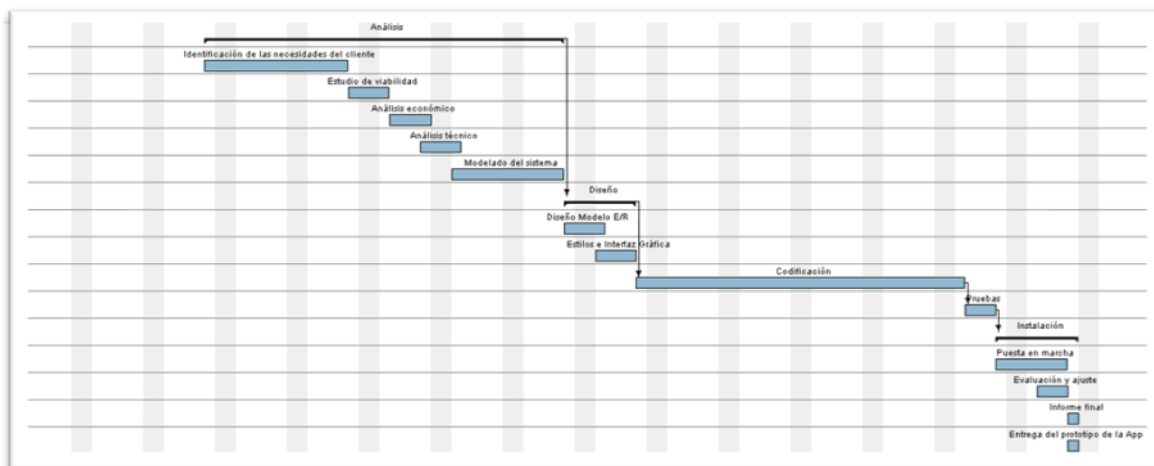


Ilustración 3. Cronograma - Línea del Tiempo

7 PRESUPUESTO

Tabla 1. Presupuesto para el desarrollo de la aplicación

Días	Horas de trabajo por semana	Total horas	Valor hora desarrollo	Valor total horas desarrollo	Diseño gráfico	Hosting privado	Controles web	Valor total del desarrollo
238 d	36 h	1224 h	35.000 COL\$	42'840.000 COL\$	1'000.000 COL\$	256.000 COL\$	995,433 COL\$	43'091.433 COL\$

Tal como se observa en la Tabla 1, el cálculo del presupuesto de esta aplicación se basa en el análisis de las horas de desarrollo consumidas en los 238 días planteados en el cronograma, para este caso tenemos que el costo promedio de una hora de desarrollo es 35.000 pesos colombianos, a razón de 16 horas trabajadas los fines de semana y 20 horas promedio entre semana, esto nos da un total de 1224 horas, para un valor aproximado del desarrollo de 42'840.000 pesos colombianos, es de aclarar que en el valor de hora de desarrollo se incluyen los gastos adjuntos que conllevan las reuniones de la recolección de información, investigación, pruebas, instalación y despliegue, entre otros.

Por otro lado, también se incluye en el presupuesto el valor de las licencias de los controles web que serán utilizados, esta es la Suite de Controles Obout licencia Enterprise, la cual tiene un valor de 599 Dólares, esto incluye soporte por 2 años y pueden ser utilizados sin límites en cualquier aplicación; pero vale aclarar que para uso estudiantil como lo es este proyecto la licencia es gratis y no expira.

En caso de ser publicada la aplicación en un Hosting privado, su valor oscila 256.000 pesos colombianos, con las siguientes características generales: 1 Gb de espacio, 50 Gb de tráfico mensual, ASP.NET, MySQL y 20 correos corporativos.

También se debe incluir el diseño gráfico de la página web, el cual está compuesto de dos propuestas de diseño y tiene un valor aproximado de 1'000.000 de pesos colombianos.

Para resumir, el presupuesto del desarrollo de la aplicación tiene un valor aproximado de 43'091.433 pesos colombianos.

8 DESARROLLO

Para el desarrollo del proyecto utilizamos una metodología personalizada donde se extrajo las principales fase del desarrollo de software, seguidamente se detallara el ciclo del desarrollo.

8.1 Análisis

8.1.1 Recolección de información mediante entrevista

La recolección de información se realizó mediante una reunión con el profesor Alexander García coordinador de trabajos de grado, en esta reunión él planteaba la necesidad de crear una aplicación web para el control de avances de los proyecto de grado, él conoce plenamente el proceso que se utiliza actualmente para llevar este control e indico que era ineficiente, básicamente existen 3 problemas, el primero es el poco seguimiento que se le realiza a los proyectos de grado, y es llevado en Excel, además es vulnerable a alteraciones de la información contenida, el segundo radica en que a los proyectos de grado de los estudiantes se les asigna un tutor y este debe cumplir con asesorías, ayudas o guías que se acuerden entre los integrantes a lo largo del proyecto, pero en muchas

ocasiones tanto por parte del tutor o de los integrantes no se llevan a cabo estas tareas y los efectos colaterales negativos se ven en el momento de entregar el proyecto de grado, pues normalmente estos proyectos no cumple con las expectativas de un proyecto de grado de calidad, en ocasiones se presentan altercados entre los estudiantes y los tutores debido a esta razón y la universidad no tiene ningún soporte o herramientas con que identificar la falla, el tercer problema es que no se puede tener una trazabilidad de los proyecto, no se sabe que tareas los constituyen, en que tiempos son ejecutadas, el tiempo de ejecución propuesto frente a tiempo real en que se logró el proyecto, no se puede identificar si se tuvo acompañamiento del tutor entre otros.

Además en ocasiones los estudiantes abandonan los proyectos o exceden los tiempos para la entrega, en si no existe una forma de controlar los avances de los proyectos de grado.

Como anteriormente se mencionó, el cliente solicita una aplicación web la en la cual se pueda llevar un control de los avances de los proyectos de grado, el cliente literalmente solicita unas funcionalidades que son el núcleo del proyecto en la siguiente secuencia, se debe capturar los datos asociados al proyecto, como título, descripción, autores, la fecha de creación, este proyecto debe tener un tutor, integrantes y tareas asociadas. La aplicación debe tener los roles de administrador con acceso a la configuración del proyecto, como creación de usuarios y roles, el rol de tutor es el que crea tareas, consulta avances de las tareas de los proyectos, consulta datos del proyecto, genera reportes de los avances, asocia estudiantes a un proyecto; si el tutor crea un proyecto, automáticamente el proyecto queda asignado a este tutor, el rol de administrador puede asignar proyecto a cualquier tutor, el rol de estudiantes puede enviar avances de tareas y revisar las

respuestas de los avances del proyecto, consultar y obtener reportes del estado del proyecto. La aplicación debe enviar alertas vía correo electrónico del estado de las tareas del proyecto, esto indica el tiempo restante o vencido de la tarea frente a la fecha final de ejecución y si ya fue realiza o no. Se deben generar los reportes de estados de los avances del proyecto, historial de proyectos por tutor y por programa académico. Se debe crear una gráfica donde se muestre el avance de las tareas por proyecto, también se debe crear un diagrama de Gantt donde se especifique los tiempos de ejecución de las tareas y sus recursos asociados.

En la reunión se acordó las herramientas de desarrollo para la aplicación, especialmente el cliente especifica que la base de datos sea en MySQL y se llega al acuerdo que el código sea realizado en C# asp.net y el diseño de los estilos de la aplicación fuese a selección del desarrollador.

A partir de la información obtenida en esta reunión se estructuraron los requerimientos para el desarrollo del software.

8.1.2 Análisis de requisito.

8.1.2.1 Definir los requisitos del software

Tipos de requerimientos.

- **RI: Requerimiento de interfaz.**
- **RNF: Requerimiento no funcional.**
- **RF: Requerimiento funciona.**

8.1.2.1.1 *Requerimientos de interfaz*

El sistema será una aplicación que no interactúa con sistemas externos y por tal razón no se implementan interfaces con otras aplicaciones.

8.1.2.1.2 *Requerimientos no funcionales*

Los requerimientos no funcionales son impuestos por el cliente y detallan comportamientos del sistema en los que la interacción del usuario final no afecta directamente el funcionamiento de este, un claro ejemplo es el motor de base de datos, cantidad mínima de usuario que debe soportar la aplicación, lenguaje de programación entre otras. La Tabla 2 describe los requerimientos no funcionales solicitados por cliente incluyendo un edificador, la descripción, fecha de captura y prioridad del requerimiento.

Tabla 2. **Requerimientos no funcionales**

Código del Requerimiento	Descripción	Fecha	Prioridad
RNF-001	La aplicación debe ser web para permitir su utilización de forma descentralizada.	16/03/2012	ALTA
RNF-002	La aplicación debe trabajar de forma independiente ya no va a interactuar con alguna de las aplicaciones existente de la Universidad EAN.	16/03/2012	ALTA
RNF-003	La aplicación debe tener una base de datos en MySQL.	16/03/2012	ALTA

RNF-004	No se debe borrar información de la base de datos por cuestión de seguridad y de acceso al historial de datos, se debe tener un campo que indique la visibilidad del dato.	16/03/2012	ALTA
RNF-005	Toda la aplicación debe tener una interfaz gráfica en lenguaje español.	16/03/2012	MEDIA
RNF-006	La información almacenada en la aplicación debe ser coherente y se debe ajustar a los anteproyectos de grado de la universidad EAN.	16/03/2012	ALTA
RNF-007	La aplicación deberá ser sometida a una serie de pruebas para garantizar que esté acorde a los requerimientos solicitados por el cliente.	16/03/2012	ALTA
RNF-008	El código fuente de la aplicación debe tener buenas prácticas de desarrollo de software para garantizar su buen despliegue y mantenimiento.	16/03/2012	ALTA

8.1.2.1.3 Requerimientos funcionales

Los requerimientos funcionales detallan las interacciones del sistema con los usuarios y en algunos casos con otros sistemas externos, en la Tabla 3 se muestra la plantilla que se utilizó para tabular dichos requerimientos, en esta se capturan minuciosamente los requerimientos del cliente, especificando datos básicos como nombre del proyecto, nombre y fecha en que se

capturo le requerimiento, entre otros, además se realiza un detallado de los datos requeridos, resultados esperados, restricciones y funcionalidades específicas del requerimiento.

En las Tablas 4 a la Tabla 23 se describen los requisitos funcionales.

Tabla 3. **Plantilla de requerimientos funcionales**

PROYECT TRACKER		Código
Especificación de requerimientos		Código del requerimiento
Nombre	Nombre del requerimiento	Fecha
		Fecha de captura del requerimiento
Entradas	Datos requeridos del requerimiento	
Salidas	Resultados del proceso de las entradas	
Restricciones	Restricciones de la salida	
Proceso		
Descripción detallada de las funcionalidades del requerimiento		

- El requerimiento RF-001, trata de la autenticación de los usuarios en la aplicación, en general el sistema debe ofrecer un formulario de autenticación y recuperación de contraseña, la información detalla del requerimiento se muestra en la Tabla 4.

Tabla 4. **Requerimiento autenticación de usuarios**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-001
Nombre	Autenticar usuarios de la aplicación	Fecha

		24/03/2012
Entradas	Nombre y Clave de usuario. Dirección de correo electrónico.	
Salidas	Usuario ingresa al sistema.	
Restricciones	Los campos son obligatorios. El usuario no podrá ingresar o recuperar la contraseña de ingreso a la aplicación si no está creado previamente.	
Proceso		
<p>El sistema debe mostrar un formulario donde se autentique o recupere la contraseña los usuario de la aplicación, en caso de autenticación el sistema debe validar la dupla usuario y clave, si es correcta debe permitir el acceso a la aplicación, en caso de que el usuario necesite recuperar la clave el sistema debe ofrecer la opción de recuperar contraseña por el nombre de usuario o dirección de correo electrónico, luego de que el usuario ingrese alguna de las opciones el sistema debe validar la existencia del usuario asociado a esto datos, si es válida la información el sistema debe enviar un correo electrónico al usuario con la información de su cuenta.</p>		

- El requerimiento RF-002, consisten principalmente en la gestión y administración de usuarios de la aplicación; el sistema debe ofrecer un formulario en el cual se pueda crear y editar los datos de los usuarios de la aplicación, la información detalla del requerimiento se muestra en la Tabla 5.

Tabla 5. **Requerimiento Administración de usuarios**

PROYECT TRACKER	Código
------------------------	---------------

Especificación de requerimientos		RF-002
Nombre	Administrar Usuarios	Fecha
		24/03/2012
Entradas	Nombre de usuario, número de identificación, nombre y apellidos, teléfono 1 y teléfono 2, correo electrónico, programa académico al que pertenece	
Salidas	Usuario creado o editado en el sistema. Envío de correo confirmando creación de usuario.	
Restricciones	Los campos nombre de usuario, número de identificación, nombre y apellidos, teléfono 1, correo electrónico y programa académico al que pertenece son obligatorios. El nombre y la dirección de correo son únicos en la aplicación. Este formulario solo está habilitado para el administrador.	
Proceso		
El administrador puede ingresar los datos de usuario para ser creado o editado, el sistema valida los datos, si el usuario selecciona la opción de guardar, el sistema crea o edita los datos del usuario en la base de datos y confirma la transacción, el sistema debe enviar un correo a la dirección de correo electrónico del usuario creado o editado con los datos de su cuenta.		

- El requerimiento RF-003, consiste en la asignación de roles a un usuario de la aplicación, existen tres roles que son estudiante, tutor y administrador, el software debe ofrecer un formulario donde se pueda asignar y desasignar los roles a un usuario por parte del administrador de la aplicación, la información detalla del requerimiento se muestra en la Tabla 6.

Tabla 6. **Requerimiento asignación de roles**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-003
Nombre	Asignar Roles de Usuarios.	Fecha
		24/03/2012
Entradas	Id del usuario, tipos de roles.	
Salidas	Usuario asociado a uno o varios roles.	
Restricciones	El usuario debe estar creado previamente	
Proceso		
El administrador puede asignar y desasignar los roles de usuario, se debe mostrar una lista de los roles existentes tales como administrador, tutor y estudiante, el sistema debe guardar esta asociación en la base de datos y confirmar la transacción.		

- El requerimiento RF-04, describe la gestión y administración de las facultades y sus programas, esta se debe realizar a través de un formulario donde pueda crear y actualizar las facultades y sus programas, la información detalla del requerimiento se muestra en la Tabla 7.

Tabla 7. **Requerimiento Administración de facultades y programas**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-004
Nombre	Administrar facultades y programas	Fecha
		24/03/2012
Entradas	Nombre de la facultad, nombres de los programas asociados a la facultad	
Salidas	Programas asociados a la facultad	
Restricciones	El nombre de la facultad es único.	

	El nombre del programa es único para una facultad.
Proceso	
<p>El administrador puede crear, editar y visualizar los datos de la facultad y los programas asociados a ella.</p> <p>El sistema guarda la información en la base de datos y confirma la transacción.</p>	

- El requerimiento RF-005, consiste en que el sistema ofrezca un formulario que muestre los detalles de los errores controlados que ocurran en la aplicación, la información detalla del requerimiento se muestra en la Tabla 8.

Tabla 8. **Requerimiento Log de errores**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-005
Nombre	Consultar log de errores	Fecha
		24/03/2012
Entradas	Errores del sistema	
Salidas	Datos asociados a los errores del sistema	
Restricciones	No aplica	
Proceso		
<p>El administrador puede visualizar los errores del sistema y su descripción, dicha descripción debe incluir la excepción, método, clase, proyecto y fecha de cuando se produjo el error.</p> <p>El sistema debe guardar en la base de datos los errores controlados de la aplicación.</p>		

- El requerimiento RF-006, trata de la gestión y administración de los datos básicos del proyecto como los son nombre, tutor e integrantes, el sistema debe ofrecer un formulario para la creación y edición de los proyectos, la información detalla del requerimiento se muestra en la Tabla 9.

Tabla 9. Administración de proyectos

PROYECT TRACKER		Código
Especificación de requerimientos		RF-006
Nombre	Administrar Proyectos	Fecha
		24/03/2012
Entradas	Nombre del proyecto. Tutor del proyecto. Integrantes del proyecto.	
Salidas	Creación del proyecto con un tutor y estudiantes asociados.	
Restricciones	El nombre del proyecto es único. Los tutores deben estar creados previamente Los estudiantes deben estar creados previamente Un estudiante solo puede pertenecer a un proyecto.	
Proceso		
El tutor puede crear, editar y visualizar los proyectos, se debe crear el nombre del proyecto, se asocia un tutor a uno o varios proyectos y se asignar uno o varios estudiantes a un proyecto. El sistema debe validar la información y las asociaciones del proyecto, si son válidas son guardadas en la base de datos y se confirma la transacción.		

- El requerimiento RF-007, trata de la gestión y administración de los datos básicos del proyecto como los son nombre, tutor e integrantes, el sistema debe ofrecer un formulario para la creación y edición de los proyectos, la información detalla del requerimiento se muestra en la Tabla 10.

Tabla 10. **Requerimiento Administración de tareas**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-007
Nombre	Administrar Tareas	Fecha
		24/03/2012
Entradas	Nombre de la tarea Descripción de la tarea Comentarios de la tarea Archivos adjuntos Proyecto al que pertenece Tutor asociado Estudiantes que integran de la tarea Fecha de inicio y fecha final de la tarea	
Salidas	Creación o edición de la tarea de un proyecto asociada a uno o varios estudiantes en un rango de fechas determinado. Envío de correo electrónico a los integrantes de la tarea con la descripción de esta.	
Restricciones	El nombre de la tarea debe ser único para el proyecto. Cada integrante debe ser único.	

	<p>La fecha de inicio debe ser menor o igual a la fecha final de la tarea.</p> <p>El administrador puede crear tareas para cualquier proyecto.</p> <p>El tutor solo puede crear tareas para los proyectos a los que está asignado.</p>
Proceso	
<p>El tutor puede crear o editar tareas para un proyecto, se deben capturar ciertos datos de la tarea como lo son su título, descripción, comentarios, archivos adjuntos, proyecto al que pertenece, visualizar el tutor de la tarea, integrantes y el rango de fecha de ejecución o elaboración de la tarea. Además el sistema debe validar que el título de la tarea sea único para el proyecto y que el estudiante este agregado solo una vez para esta tarea.</p> <p>Si la validación es correcta se guardan los datos de la tarea en la base de datos, se confirma la transacción y se envía un correo electrónico con la descripción de la tarea a cada uno de los integrantes de la tarea, si el sistema no puede enviar el correo electrónico se informa al actual usuario del fallo en el envío.</p>	

- El requerimiento RF-008, trata de la gestión y administración del estado de las tareas de un proyecto, el sistema debe ofrecer un formulario para la creación y edición de los estados y porcentajes de avance de las tareas, la información detalla del requerimiento se muestra en la Tabla 11.

Tabla 11. **Requerimiento Administración de Estado de las tareas**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-008
Nombre	Administrar Estado de las tareas	Fecha
		24/03/2012
Entradas	Identificador de la tarea.	

	<p>Proyecto al que pertenece la tarea.</p> <p>Estados de la tarea "creada, en progreso, sin solución, invalida, duplicada y finalizada".</p> <p>Rango de porcentaje de avance de la tarea entre el 0% y el 100%</p>
Salidas	Creación o edición del estado de la tarea
Restricciones	<p>Los valores por defecto asignados a la tarea son estado creado y porcentaje de avance 0%.</p> <p>El administrador tiene acceso a todos los proyectos.</p> <p>El tutor tiene acceso solo a los proyectos que le han sido asignados.</p>
Proceso	
<p>El administrador o el tutor pueden crear o editar el estado de la tarea, se selecciona un proyecto y a continuación se muestran las tareas asociadas a este, el usuario selecciona una tarea y luego asigna un estado a la tarea y un porcentaje de avance.</p> <p>El sistema guarda la información del estado y confirma la transacción.</p>	

- En el requerimiento RF-009, el sistema debe ofrecer un formulario con una gráfica de barras horizontales donde se represente los estados y porcentajes de avances de las tareas de un proyecto, la información detalla del requerimiento se muestra en la Tabla 12.

Tabla 12. **Requerimiento consultar de gráficos de estado de tareas**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-009
Nombre	Consultar gráficos de estado de tareas	Fecha

		24/03/2012
Entradas	Nombre del proyecto.	
Salidas	Grafico comparativo de los avances de las tareas con una breve descripción.	
Restricciones	<p>El administrador tiene acceso a todos los proyectos.</p> <p>El tutor tiene acceso solo a los proyectos que le han sido asignados.</p> <p>El estudiante tiene acceso solo al proyecto que le ha sido asignado.</p>	
Proceso		
<p>El usuario selecciona un proyecto y el sistema genera un gráfico de barras horizontales donde se evidencia el porcentaje de avance de la tarea, cada barra asociada a una tarea debe tener una breve descripción de la tarea con su estado y porcentaje de avance.</p>		

- En el requerimiento RF-010, el sistema debe ofrecer un formulario para la consulta de los datos básicos de un proyecto, la información detalla del requerimiento se muestra en la Tabla 13.

Tabla 13. **Requerimiento consulta de datos del proyecto**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-010
Nombre	Consultar datos del proyecto	Fecha
		24/03/2012
Entradas	Nombre del proyecto.	
Salidas	Visualización de los datos asociados a un proyecto	
Restricciones	El administrador tiene acceso a todos los proyectos.	

	El tutor tiene acceso solo a los proyectos que le han sido asignados. El estudiante tiene acceso solo al proyecto que le ha sido asignado.
Proceso	
<p>El usuario selecciona un proyecto y el sistema muestra los datos básicos del proyecto en la pantalla de inicio de cada rol.</p> <p>Se debe mostrar el nombre del proyecto, tutor e integrantes asociados, fecha de creación, fecha de inicio y fecha final del proyecto, estado del proyecto "abierto o cerrado".</p>	

- En el requerimiento RF-011, el sistema debe ofrecer un formulario para el cierre y reapertura de los proyectos, la información detalla del requerimiento se muestra en la Tabla 14.

Tabla 14. **Requerimiento cierre del proyecto**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-011
Nombre	Cerrar proyecto	Fecha
		24/03/2012
Entradas	Nombre del proyecto.	
Salidas	Proyecto cerrado o abierto	
Restricciones	<p>Solo se permite cerrar o abrir proyecto a los usuarios con rol de administrador o tutor.</p> <p>El administrador tiene acceso a todos los proyectos.</p> <p>El tutor tiene acceso solo a los proyectos que le han sido asignados.</p>	
Proceso		
El usuario selecciona un proyecto y el sistema muestra las opciones de cerrar o volver a abrir el proyecto, si el usuario		

selecciona cerrar proyecto, el sistema muestra la opción de cerrar las tareas asociadas a este proyecto con el estado de finalizadas al 100%. El usuario selecciona la opción de guarda, el sistema guarda la información y confirma la transacción.

- En el requerimiento RF-012, el sistema debe ofrecer un formulario para la consulta de reportes del estado de los proyectos, la información detalla del requerimiento se muestra en la Tabla 15.

Tabla 15. **Requerimiento consulta de reporte del estado del proyecto**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-012
Nombre	Consultar reporte estado del proyecto	Fecha
		24/03/2012
Entradas	Nombre del proyecto.	
Salidas	Reporte del estado del proyecto en formato PDF	
Restricciones	<p>El administrador tiene acceso a todos los proyectos.</p> <p>El tutor tiene acceso solo a los proyectos que le han sido asignados.</p> <p>El estudiante tiene acceso solo al proyecto asociado.</p>	
Proceso		
<p>El usuario selecciona un proyecto y ver reporte, el sistema generar un reporte en formato PDF que contiene los datos básicos de proyecto y su estado, sus tareas, avances de tareas, respuestas de avances de tareas y toda la traza del proyecto hasta la fecha del sistema.</p>		

- En el requerimiento RF-013, el sistema debe ofrecer un formulario para la consulta de reportes del historial de proyectos por tutor, la información detalla del requerimiento se muestra en la Tabla 16.

Tabla 16. **Requerimiento consulta de reporte de historial de proyectos por tutor**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-013
Nombre	Consultar reporte de historial proyectos por tutor.	Fecha
		24/03/2012
Entradas	Nombre del tutor. Rango de fechas entre inicial y final para filtrar los proyectos por fecha de creación.	
Salidas	Reporte del historial del proyecto asociados a un tutor en formato PDF	
Restricciones	El administrador tiene acceso a todos los tutores. El tutor tiene acceso solo a él.	
Proceso		
El usuario selecciona un tutor, luego ingresa la fecha inicial y la fecha final del filtro asociado a la fecha de creación de los proyecto, a continuación selecciona ver reporte, el sistema generar un reporte en formato PDF que contiene los datos básicos de proyecto y su estado, sus tareas, avances de tareas, respuestas de avances de tareas y toda la traza de los proyectos asociados al tutor seleccionado en el rango de fechas ingresado.		

- En el requerimiento RF-014, el sistema debe ofrecer un formulario para la consulta de reportes del historial de proyectos de proyecto por cada programa de una facultad, la información detalla del requerimiento se muestra en la Tabla 17.

Tabla 17. Consultar reporte de historial proyectos por programa

PROYECT TRACKER		Código
Especificación de requerimientos		RF-014
Nombre	Consultar reporte de historial proyectos por programa.	Fecha
		24/03/2012
Entradas	Nombre de la facultad. Nombre del programa Rango de fechas entre inicial y final para filtrar los proyectos por fecha de creación.	
Salidas	Reporte del historial del proyecto de un programa en formato PDF	
Restricciones	El administrador es el único que puede generar este reporte.	
Proceso		
<p>El usuario selecciona un facultad, luego selecciona de una lista el programa que está asociado a la facultad seleccionada, ingresa la fecha inicial y la fecha final del filtro asociado a la fecha de creación de los proyecto, a continuación selecciona ver reporte, el sistema generar un reporte en formato PDF que contiene los datos básicos de proyecto y su estado, sus tareas, avances de tareas, respuestas de avances de tareas y toda la traza de los proyectos asociados al programa seleccionado en el rango de fechas ingresado.</p>		

- El requerimiento RF-15, trata de la gestión y administración de los avances de las tareas de un proyecto, el sistema debe ofrecer un formulario para la creación y edición de respuestas a los avances de las tareas, la información detalla del requerimiento se muestra en la Tabla 18.

Tabla 18. **Requerimiento revisión de avances de tareas del proyecto.**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-015
Nombre	Revisar avances de tareas del proyecto.	Fecha
		24/03/2012
Entradas	Nombre de proyecto. Tareas del proyecto. Avances de las tareas. Respuesta de avance. Archivo adjunto.	
Salidas	Se revisa un avance y se le asigna una respuesta. Envío de respuestas como archivo adjunto.	
Restricciones	El administrador tiene acceso a todos los proyectos. El tutor tiene acceso solo a los proyectos que le han sido asignados.	
Proceso		
El usuario selecciona un proyecto, el sistema muestra las tareas asociadas al proyecto y sus avances, el usuario selecciona un avance lo revisa y puede enviar una respuesta mediante un campo de texto o adjuntar un archivo, el sistema guarda la información y confirma la transacción.		

- En el requerimiento RF-016, el sistema debe ofrecer un formulario para la consulta de las tareas asociadas a un proyecto, en este se debe mostrar la descripción detallada de los datos vinculados a la tarea, la información detalla del requerimiento se muestra en la Tabla 19.

Tabla 19. **Requerimiento consulta de tareas de un proyecto**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-016
Nombre	Consultar tareas del proyecto.	Fecha
		24/03/2012
Entradas	Nombre de proyecto.	
Salidas	Datos de las tareas asociadas a un proyecto	
Restricciones	Solo pueden ser consultadas por el usuario estudiante	
Proceso		
<p>El sistema consulta las tareas que han sido asignadas al proyecto del usuario.</p> <p>El usuario puede visualizar los datos básicos de la tarea y su estado.</p>		

- En el requerimiento RF-017, el sistema debe ofrecer un formulario que permita capturar y enviar la información del avance de la tarea de un proyecto, la información detalla del requerimiento se muestra en la Tabla 20.

Tabla 20. **Requerimiento envío de avances de tareas del proyecto**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-017
Nombre	Enviar avances de tareas del proyecto	Fecha
		24/03/2012

Entradas	Nombre de la tarea. Título del avance. Descripción del avance. Archivo adjunto del avance.
Salidas	Se envía el avance de la tarea al tutor del proyecto.
Restricciones	Solo pueden ser enviados por el usuario estudiante
Proceso	
<p>El sistema consulta las tareas que han sido asignadas al usuario.</p> <p>El usuario ingresa el título, descripción y opcionalmente un archivo adjunto del avance de la tarea, selecciona guardar y el sistema envía el avance de la tarea al tutor asignado al proyecto y confirma la transacción.</p>	

- En el requerimiento RF-018, el sistema debe ofrecer un formulario para la consulta de los avances de las tareas de un proyecto de un usuario o estudiante, la información detalla del requerimiento se muestra en la Tabla 21.

Tabla 21. **Requerimiento consulta de avances de tareas de un proyecto**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-018
Nombre	Consultar avances de tareas de un proyecto	Fecha
		24/03/2012
Entradas	Nombre del proyecto	
Salidas	Se visualiza las tareas y sus avances con sus respectivos estado y respuestas	
Restricciones	Solo pueden ser enviados por el usuario estudiante	

Proceso
<p>El sistema consulta las tareas que han sido asignadas al usuario.</p> <p>El usuario visualiza las tareas que le ha sido asignadas, cada una de ellas contiene un descripción detallada de sus avances, estado y posibles respuestas.</p>

- En el requerimiento RF-019, el sistema debe implementar un proceso que envíe alertas de los estados del avance de las tareas asociadas a un proyecto, dicha alerta debe ser enviadas vía correo electrónico a todo los integrantes de la tarea, la información detalla del requerimiento se muestra en la Tabla 22.

Tabla 22. **Requerimiento envío de alerta de estados de los avances de tareas de un proyecto**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-019
Nombre	Enviar alerta de estados de los avances de tareas de un proyecto	Fecha
		24/03/2012
Entradas	Nombre del proyecto	
Salidas	Se envía un correo con el estado del proyecto.	
Restricciones	Solo pueden ser enviados por el usuario estudiante o tutor	
Proceso		
<p>El sistema muestra la opción de enviar alertas de los estados de los avances del proyecto, si el usuario selecciona enviar alerta, el sistema consulta las tareas de un proyecto con sus estados, avance, posibles respuestas y envía un correo con esta información a cada integrante del proyecto.</p>		

- En el requerimiento RF-020, el sistema debe ofrecer un formulario donde se implemente un diagrama de Gantt que contenga los tiempos de ejecución y los recursos asociados a las tareas de un proyecto, la información detalla del requerimiento se muestra en la Tabla 23.

Tabla 23. **Requerimiento diagrama de Gantt**

PROYECT TRACKER		Código
Especificación de requerimientos		RF-020
Nombre	Consultar grafico o diagrama de Gantt	Fecha
		24/03/2012
Entradas	Nombre del proyecto	
Salidas	Grafico o diagrama de Gantt con las tareas del proyecto en una línea de tiempo.	
Restricciones	<p>El administrador tiene acceso a todos los proyectos.</p> <p>El tutor tiene acceso solo a los proyectos que le han sido asignados.</p> <p>El estudiante tiene acceso solo al proyecto que le ha sido asignado.</p>	
Proceso		
<p>El usuario selecciona un proyecto y el sistema genera un diagrama o gráfico de Gantt que contiene las tareas del proyecto seleccionado y describe la fecha inicial y final de la tarea en una línea de tiempo desde el inicio del proyecto hasta su fin.</p>		

8.1.3 Viabilidad

En la viabilidad del proyecto analizamos y determinamos los tiempos de ejecución, presupuesto, herramientas técnicas y recursos humanos que pueden determinar la posibilidad técnica de realizar el software, cada ítem o factor esta descrito en la Tabla 24.

A través del análisis de los factores que inciden en la viabilidad se concluye que el proyecto es factible, ya que el tiempo de ejecución del proyecto está entre los plazos de entrega solicitados por el cliente, además el desarrollo del software no genera ningún costo real de inversión tanto para el desarrollador como para el cliente, a nivel de las herramientas técnicas se cuenta con todas ellas y sus licencias son estudiantiles o gratis, finalmente se cuenta con una persona para el desarrollo del proyecto la cual dispone de los conocimientos y del tiempo necesario para la realización del proyecto.

Tabla 24. Viabilidad del Proyecto

Item	Viabilidad
<p style="text-align: center;">Tiempo</p>	<p>El tiempo de ejecución estimado del proyecto está alrededor de 8 meses entre los meses de marzo y noviembre de 2012, este tiempo está justificado en el cronograma de trabajo, donde se estipula las tareas a desarrollar y el tiempo en días, además el estimado de las horas de trabajo promedio por semana se especifica en la sección de presupuesto. Es de resaltar que el cliente solicita una entrega preliminar o una primera versión del software para enero del 2013.</p>
<p style="text-align: center;">Presupuesto</p>	

	<p>La aplicación en realidad no genera ningún gasto relevante tanto para el desarrollador de proyecto, como para la Universidad EAN. Vale aclarar que en la sección de presupuesto se estima el valor de la aplicación, este caso aplica si se tuviese que contratar a un desarrollador o a una empresa de software.</p>
<p>Herramientas</p>	<ul style="list-style-type: none"> • Para la codificación y diseño de interface graficas de la aplicación se utilizara Asp.net C# Visual Studio 2010. • La base de datos se realizara en MySQL Workbench 5.2. • Para el modelado de la aplicación Modeling Project Visual Studio 2010. • Diccionario de base de datos wb-datadict-plugin-0.8 for MySQL. • El Modelo Entidad Relación se realizara en Data Modeling ERR MySQL Workbench 5.2. • Diseño gráfico y estilos de interfaz gráfica Adobe Fireworks CS4. • Los videos de los tutoriales se realizaran en Camtasia Xpress Studio 8.
<p>Personal</p>	<p>Respecto a los recursos de personal se cuenta con solo una persona para realizar todo el ciclo de vida del desarrollo de la aplicación. Dicha persona realizara toda la fase de análisis, diseño, desarrollo, pruebas, implantación y mantenimiento.</p>

8.2 Diseño

Para el modelado de la aplicación se diseñara dos diagramas fundamentales para estructurar y entender el funcionamiento de la aplicación frente a los requerimientos solicitados por el cliente, los diagramas a realizar son Casos de Uso y Diagrama de Clases, la herramienta que se utilizará para modelar la aplicación es Modeling Project Visual Studio 2010.

8.2.1 Casos de Uso

Los casos de uso documentan el comportamiento del sistema desde el punto de vista del usuario. El primer paso para realizar los casos de uso es identificar los actores que están asociados a una serie de roles, entendiendo como actor entidades que interactúan con el sistemas realizando actividades específicas, estos actores pueden ser personas, dispositivos de hardware u otros sistemas tales como un Webservice.

A continuación se describen los casos de uso más relevantes del sistema, en estos se describe el requerimiento asociado, los actores implicados, las condiciones previas y posteriores, las excepciones, los flujos principales y alternos. Para acceder a la información detallada de cada caso de uso identificado en el sistema véase el Anexo 1.

Convenciones

- PT-(Numero identificador) = ID del caso de uso.
- RF-(Numero identificador) = ID del requerimiento.

8.2.1.1 Caso de uso administrar proyectos

Este caso de uso está asociado al requerimiento funcional RF-006, el cual describe la gestión y administración de los proyectos, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario para crear y editar los datos básicos del proyecto, como el nombre y descripción entre otros, también se muestra la opción de vincular un tutor al proyecto, el actor ingresa los datos básicos del proyecto y selecciona un tutor para ser vinculado al proyecto, la información detallada del caso de uso se encuentra en la Tabla 25 y la gráfica asociada es la Ilustración 7.

Tabla 25. Caso de uso administrar proyectos

Identificador:	PT-007	Requerimiento asociado:	RF-006
Nombre:	Administrar proyectos		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos usuarios con rol de tutor. • Deben estar creados algunos usuarios con rol de estudiante. 		
Poscondiciones:	<ul style="list-style-type: none"> • Proyecto creado o actualizado. • Tutor asociado a uno o varios proyectos. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear o actualizar los proyectos. 			

2. El actor selecciona la opción de nuevo.
3. El sistema muestra los campos que van a estar asociados al proyecto.
4. El sistema muestra la lista de tutores para asignar a un proyecto.
5. El actor selecciona un tutor de la lista.
6. El sistema debe validar los campos obligatorios.
7. El sistema debe validar que no exista un proyecto con el mismo nombre para la facultad asociada.
8. El sistema realizar el caso de uso (PT-08)
9. El actor selecciona guardar.
10. El sistema almacena la información.
11. El sistema confirma la transacción.
12. El actor sale del caso de uso.

Flujo alternos:

- A. El actor selecciona la opción editar proyecto.
 1. En el paso 1.
 2. El sistema muestra los datos que se pueden actualizar.
 3. El actor cambia los campos que requiera.
 4. El sistema debe validar que no exista un proyecto con el mismo nombre para una facultad.
 5. El actor selecciona guardar.
 6. El sistema almacena los cambios.
 7. El actor sale del caso de uso.
- B. El sistema debe validar la asignación del proyecto.
 1. En el paso 5.
 2. El sistema debe validar el rol del actor autenticado, si es administrador y tutor puede seleccionar cualquier tutor para asignarlo a un proyecto, pero si el rol es tutor el proyecto a crear o actualizar será asignado al actor autenticado.

Excepciones:

A. Validación de campo obligatorios proyecto

1. En el paso 4 el sistema no guardar o actualiza la información si no existen los campos requeridos.
2. El sistema muestra un mensaje de alerta especificando los campos obligatorios.
3. Se regresa al paso 3.

B. Validación de nombre de la proyecto.

1. En el paso 4 el sistema debe validar que no exista un proyecto con el mismo nombre.
2. El sistema muestra un mensaje de alerta especificando el campo que debe cambiar.
3. Se regresa al paso 3.

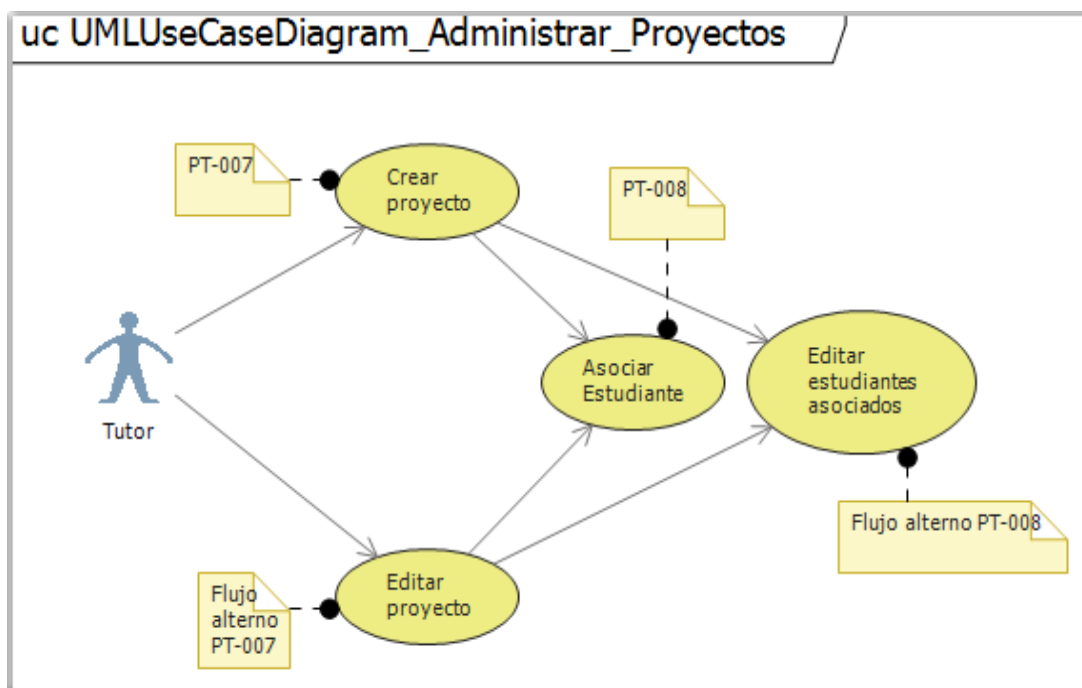


Ilustración 4. Casos de Uso PT- 007 y PT- 008

8.2.1.2 Caso de uso asociar estudiante a proyecto

Este caso de uso está asociado al requerimiento funcional RF-006, el cual describe la vinculación y la desvinculación de los integrantes al proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario con las listas de proyectos y estudiantes existentes, luego el actor selecciona un proyecto y los estudiantes para ser asociados a este proyecto, además el sistema ofrece la opción de editar la asociación de los integrantes del proyecto, la información detallada del caso de uso se encuentra en la Tabla 26 y la gráfica asociada es la Ilustración 10.

Tabla 26. Caso de uso asociar estudiante a proyecto

Identificador:	PT-008	Requerimiento asociado:	RF-006
Nombre:	Asociar estudiante a proyecto		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos usuarios con rol de estudiante. • Deben estar creados algunos proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Asociación de estudiantes a un proyecto 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para asociar los estudiantes a un proyecto. 2. El actor selecciona un proyecto. 3. El sistema muestra la lista de los estudiantes. 4. El actor selecciona un estudiante. 			

5. El sistema debe validar que el estudiante no este asociado a un proyecto.
6. El actor selecciona guardar.
7. El sistema almacena la información.
8. El sistema confirma la transacción.
9. El actor sale del caso de uso.

Flujo alternos:

- A. El actor selecciona la opción editar los estudiantes asociados al proyecto.
 1. En el paso 5.
 2. El sistema muestra los estudiantes ya asociados al proyecto para poder actualizar la asociación.
 3. El actor selecciona el estudiante a asociar al proyecto o lo inactiva.
 4. El sistema debe validar el estudiante no este asociado a otro proyecto o previamente ya este asociado al mismo proyecto.
 5. El actor selecciona guardar.
 6. El sistema almacena los cambios.
 7. El actor sale del caso de uso.

Excepciones:

- A. Validación de la asociación de estudiante al proyecto
 1. En el paso 5.
 2. El sistema debe validar la asociación.
 3. El sistema muestra un mensaje de alerta especificando que ya está asociado el estudiante a un proyecto.
 4. Se regresa al paso 4.

8.2.1.3 Caso de uso administrar tareas

Este caso de uso está asociado al requerimiento funcional RF-007, el cual describe la gestión y administración de las tareas de los proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario para crear y editar los datos asociados a la tareas, como el título, descripción, proyecto, integrantes y fecha de ejecución entre otros, la información detallada del caso de uso se encuentra en la Tabla 27 y la gráfica asociada es la Ilustración 8.

Tabla 27. Caso de uso administrar tareas

Identificador:	PT-009	Requerimiento asociado:	RF-007
Nombre:	Administrar tareas		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyecto • Debe estar asociados los tutores a un proyecto. • Deben estar asociados los estudiantes a un proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Crear tarea 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear y editar las tareas de un proyecto 			

2. El sistema muestra una lista de los proyectos asociadas al rol de usuario.
3. El actor selecciona un proyecto.
4. El sistema muestra el tutor asociado.
5. El sistema muestra la lista de los estudiantes asociados al proyecto seleccionado.
6. El sistema realizar el caso de uso (PT-010)
7. El sistema muestra una lista de fecha inicial y fecha final del proyecto.
8. El actor selecciona la fecha inicial y final de la tarea
9. El sistema muestra campos adicionales como el título, descripción, comentario y archivos adjuntos de la tarea
10. El actor agrega los datos del título y la descripción que son obligatorios, opcionalmente los comentarios y los archivos adjuntos.
11. El sistema valida los campos obligatorios.
12. El actor selecciona guardar.
13. El sistema almacena la información.
14. El sistema confirma la transacción.
15. El actor sale del caso de uso.

Flujo alternos:

- A. Visualización de proyectos
 1. En el paso 2.
 2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador y tutor muestra todos los proyectos.
- B. El actor selecciona la opción editar la tarea.
 1. En el paso 1.
 2. El sistema muestra los datos asociados a la tarea.
 3. Si se necesita editar los estudiantes ya asociados. El sistema realiza el caso de uso (PT-010).
 4. El actor cambia los campos adicionales que requiera.

<ol style="list-style-type: none">5. El actor selecciona guardar.6. El sistema almacena los cambios.7. El actor sale del caso de uso.
Excepciones:
<p>A. Validación de campo obligatorios</p> <ol style="list-style-type: none">1. En el paso 10.2. El sistema no guardar o actualiza la información si no existen los campos requeridos.3. El sistema muestra un mensaje de alerta especificando los campos obligatorios.4. Se regresa al paso 10. <p>B. Validación fechas</p> <ol style="list-style-type: none">1. En el paso 8.2. El sistema debe validar que la fecha final sea mayo que la fecha inicial.3. El sistema muestra un mensaje de alerta de la inconsistencia.4. Regresa al paso 8

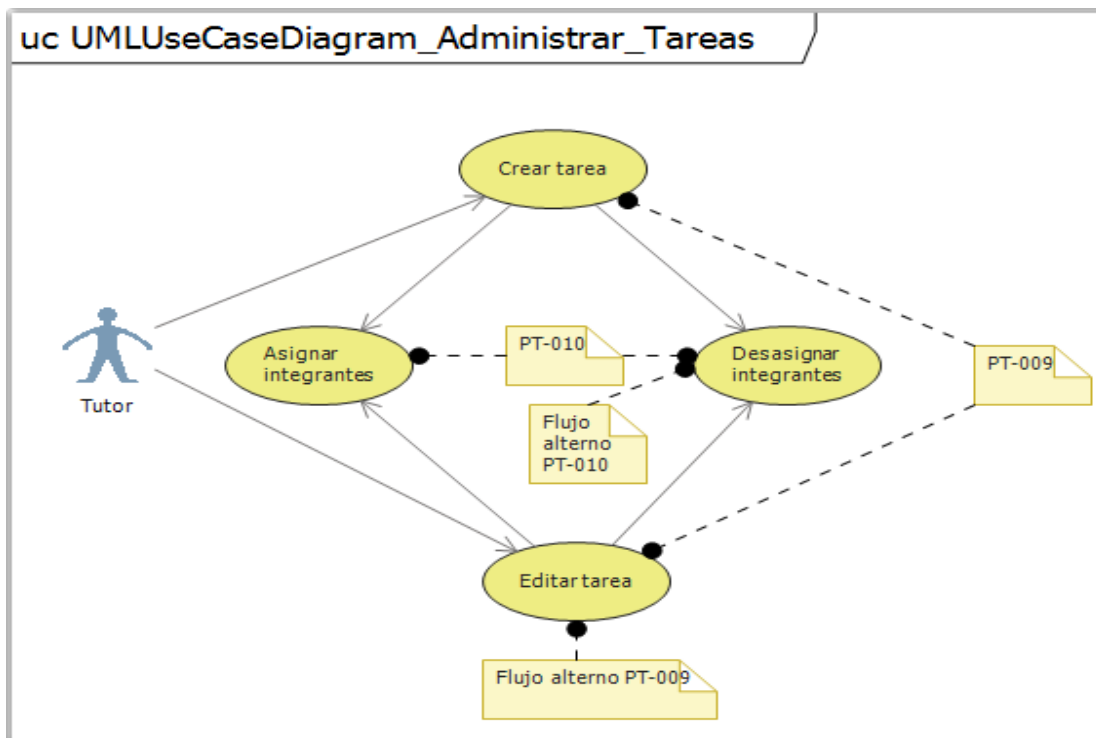


Ilustración 5. Casos de Uso PT-010 y PT-009

8.2.1.4 Caso de uso asociar estudiante a tarea

Este caso de uso está asociado al requerimiento funcional RF-007, el cual describe la asociación de los estudiantes su proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario con los proyectos, sus tareas y los estudiantes asociados al proyecto, luego el actor selecciona los estudiantes y los vincula o desvincula de la tarea seleccionada, la información detallada del caso de uso se encuentra en la Tabla 29 y la gráfica asociada es la Ilustración 5.

Tabla 28. Caso de uso asociar estudiante a tarea

Identificador:	PT-010	Requerimiento asociado:	RF-007
Nombre:	Asociar estudiante a tarea		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Se debe cumplir las precondiciones del caso de uso(PT-009) • Se realizar el evento 1, 2 ,3 y 5 del caso de uso(PT-009) 		
Poscondiciones:	<ul style="list-style-type: none"> • Asociación de estudiantes a una tarea 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El actor selecciona un estudiante. 2. El actor asigna o desasigna uno o varios estudiantes a la tarea. 3. El sistema debe validar que el estudiante no este asociado previamente a la tarea. 4. El actor selecciona guardar. 5. El sistema almacena la información. 6. El sistema confirma la transacción. 7. El actor sale del caso de uso. 			
Flujo alternos:			
<ol style="list-style-type: none"> A. El actor selecciona la opción de desasignar los estudiantes asociados a la tarea. <ol style="list-style-type: none"> 1. En el paso 3. 2. El sistema muestra los estudiantes ya asociados al proyecto para poder actualizar la asociación. 3. El actor desasigna el estudiante de la tarea. 4. El actor selecciona guardar. 			

<ol style="list-style-type: none"> 5. El sistema almacena los cambios. 6. El actor sale del caso de uso.
Excepciones:
<ol style="list-style-type: none"> A. Validación de la asociación de estudiante al proyecto <ol style="list-style-type: none"> 1. En el paso 5. 2. El sistema debe validar la asociación. 3. El sistema muestra un mensaje de alerta especificando que ya está asociado el estudiante a un proyecto 4. Se regresa al paso 4.

8.2.1.5 Caso de uso administrar estado de las tareas

Este caso de uso está asociado al requerimiento funcional RF-008, y describe la gestión y administración de los estados de las tareas de los proyectos, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario para asociar un estado y un porcentaje de avance a la tarea, la información detallada del caso de uso se encuentra en la Tabla 29 y la gráfica asociada es la Ilustración 7.

Tabla 29. Caso de uso administrar estado de las tareas

Identificador:	PT-011	Requerimiento asociado:	RF-008
-----------------------	--------	--------------------------------	--------

Nombre:	Administrar estado de las tareas
Actores:	Tutor
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creadas las tareas de un proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Estado de la tarea creado o actualizado
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear o editar el estado de la tarea. 2. El sistema muestra una lista de los proyectos. 3. El actor selecciona un proyecto. 4. El sistema una lista de la tareas asociadas al proyecto. 5. El actor selecciona una tarea. 6. El sistema una lista de los estados de la tarea "creada, en progreso, sin solución, invalida, duplicada y finalizada". 7. El actor selecciona un estado. 8. El sistema muestra una lista con los valores del porcentaje de avance entre 0% y 100%. 9. El actor selecciona un valor del porcentaje. 10. El actor selecciona guardar. 11. El sistema almacena la información. 12. El sistema confirma la transacción. 13. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 2. 	

<ol style="list-style-type: none"> 2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador y tutor muestra todos los proyectos. <p>B. El actor actualiza el estado de la tarea.</p> <ol style="list-style-type: none"> 1. En el paso 7. 2. El actor selecciona un estado de la tarea y un valor del porcentaje. 3. El actor selecciona guardar. 4. El sistema almacena los cambios. 5. El actor sale del caso de uso.
Excepciones:
A. No aplica

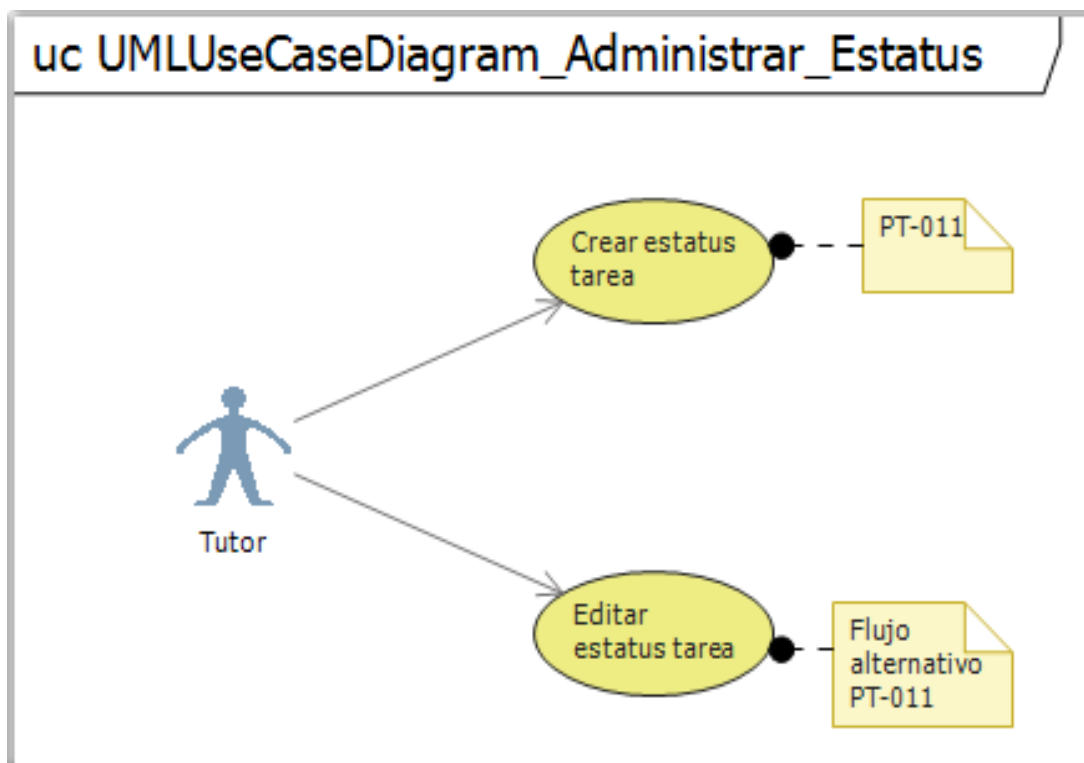


Ilustración 6. Caso de Uso PT-011

8.2.1.6 Caso de uso revisar avances de tareas del proyecto.

Este caso de uso está asociado al requerimiento funcional RF-013, para este caso el sistema muestra un formulario con las lista de avances de las tareas de un proyectos, luego el actor tiene la opción de revisar los datos asociados al avance de la tarea y puede agregar una respuesta a dicho avance, el actor implicado es el tutor, la información detallada del caso de uso se encuentra en la Tabla 30 y la gráfica asociada es la Ilustración 8.

Tabla 30. Caso de uso revisar avances de tareas del proyecto.

Identificador:	PT-018	Requerimiento asociado:	RF-015
Nombre:	Revisar avances de tareas del proyecto.		
Alias:	Revisar avances.		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyectos. • Deben estar creadas algunas tareas asociadas a un proyecto. • Deben estar creados algunos avances asociados a una tarea. 		
Poscondiciones:	<ul style="list-style-type: none"> • Revisión del avance de las tareas de un proyecto y su correspondiente retroalimentación. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para revisar y enviar respuestas a los avances de las tareas de un proyecto. 			

2. El sistema muestra una lista de los proyectos.
3. El actor selecciona un proyecto.
4. El sistema muestra una lista de tareas asociadas al proyecto.
5. El actor selecciona una tarea.
6. El sistema muestra los avances asociados a una tarea seleccionada por el actor.
7. El actor selecciona un avance de una tarea.
8. El actor revisa el avance.
9. El actor ingresa una respuesta del avance.
10. El actor selecciona guardar.
11. El sistema guarda la respuesta.
12. El sistema actualiza el estado del avance.
13. El actor sale del caso de uso.

Flujo alternos:

A. Visualización de proyectos

1. En el paso 2.
2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador muestra todos los proyectos.

Excepciones:

A. Validación de campo obligatorios de la repuesta

1. En el paso 10.
2. Si el actor no agrega una respuesta.
3. El sistema muestra un mensaje de alerta especificando los campos obligatorios.
4. Se regresa al paso 9.

B. Adjuntar archivo de respuesta.

1. En el paso 9.
2. El sistema muestra la opción de adjuntar archivos.

3. El actor carga uno o varios archivos de las respuestas de los avances de la tarea.
4. El sistema continua en al paso 10.

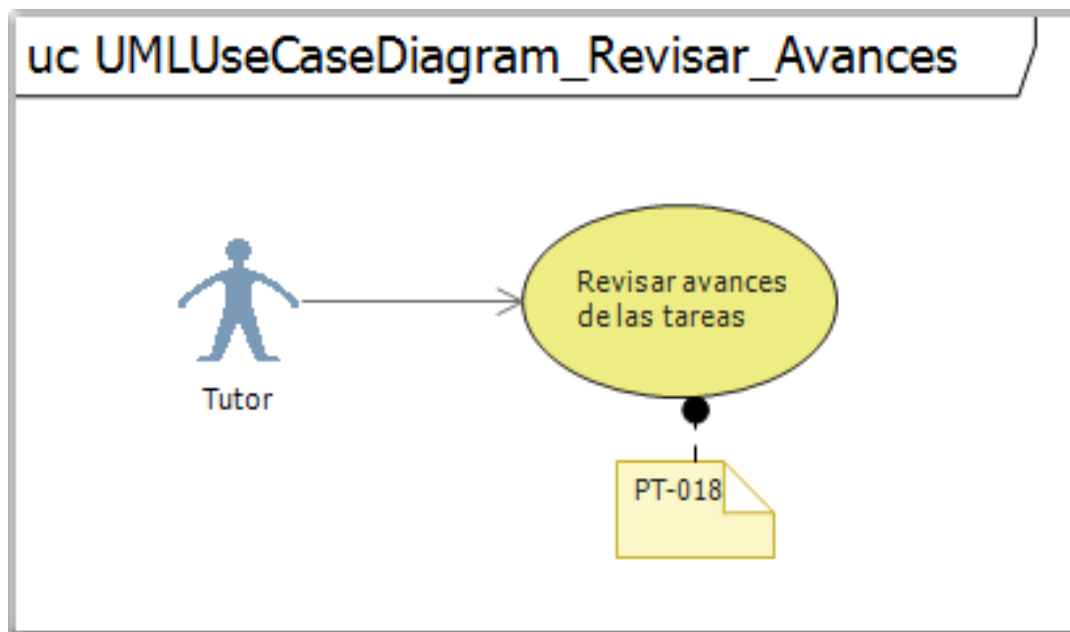


Ilustración 7. Caso de Uso PT-018

8.2.1.7 Caso de uso consultar tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-016, básicamente el sistemas muestra al actor un formulario con la lista de tareas de un proyecto, el actor selecciona una tarea y el sistema le muestra una descripción detalla de los datos asociados de la tarea, tales como título, descripción, integrantes, etc. El actor implicado es el tutor, la información detallada del caso de uso se encuentra en la Tabla 31 y la gráfica asociada es la Ilustración 15.

Tabla 31. Caso de uso consultar tareas de un proyecto

Identificador:	PT-019	Requerimiento asociado:	RF-016
-----------------------	--------	--------------------------------	--------

Nombre:	Consultar tareas de un proyecto
Alias:	Consultar tareas
Actores:	Estudiante
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los datos de las tareas de un proyecto del actor
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar los datos de las tareas de un proyecto asociado al actor. 2. El sistema muestra la opción de visualizar las tareas de todos los integrantes o solo las tareas del actor. 3. El actor selecciona la opción de visualizar las tareas de todos los integrantes. 4. El sistema muestra una lista de las tareas con sus datos, el último estado y la diferencia de días entre la fecha del sistema y la fecha final de la tarea. 5. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. Visualización de tareas del actor</p> <ol style="list-style-type: none"> 1. En el paso 3. 2. El actor selecciona la opción de ver solo sus tareas. 3. El sistema continúa en el paso 4. 	
Excepciones:	
A. No aplica	

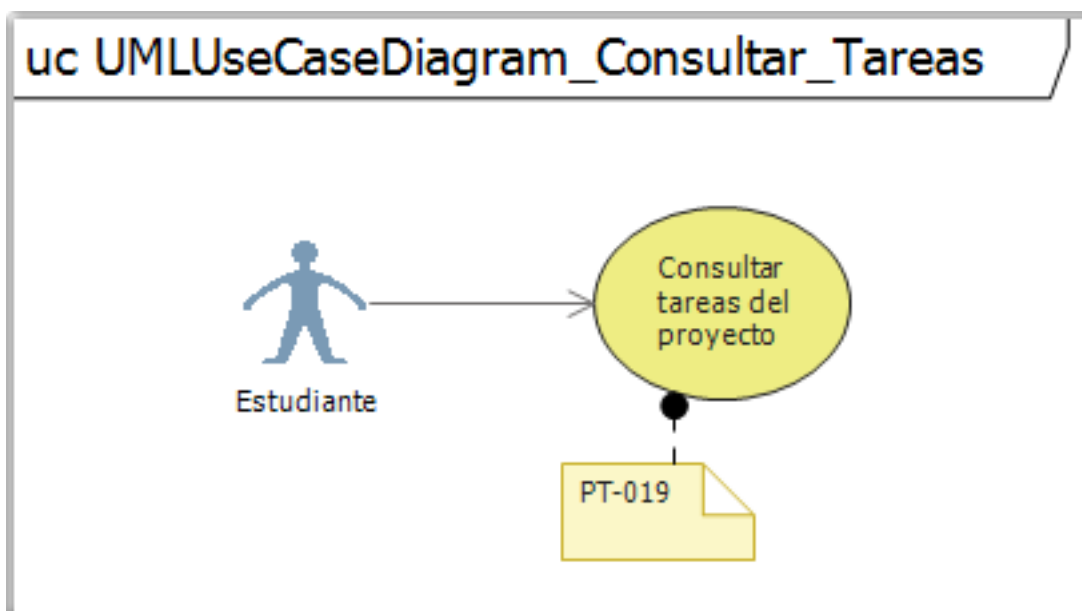


Ilustración 8. Caso de Uso PT-019

8.2.1.8 Caso de uso enviar avances de tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-017, el sistema ofrece un formulario con la lista de tareas de un proyecto asociado a un estudiante, luego el actor selecciona una tarea y adjunto los datos del avance de dicha tarea, el actor implicado es el estudiante, la información detallada del caso de uso se encuentra en la Tabla 32 y la gráfica asociada es la Ilustración 10.

Tabla 32. Caso de uso enviar avances de tareas de un proyecto

Identificador:	PT-020	Requerimiento asociado:	RF-017
-----------------------	--------	--------------------------------	--------

Nombre:	Enviar avances de tareas de un proyecto
Alias:	Enviar avances
Actores:	Estudiante
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Los avances de las tareas han sido enviados
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para enviar los avances de las tareas del actor. 2. El sistema muestra la lista de las tareas asignadas al actor. 3. El actor selecciona una tarea. 4. El actor ingresa los datos del avance. 5. El actor selecciona guardar. 6. El sistema almacena la información. 7. El sistema confirma la transacción. 8. El actor sale del caso de uso. 	
Flujo alternos:	
<ol style="list-style-type: none"> A. Validación de campo obligatorios <ol style="list-style-type: none"> 1. En el paso 5. 2. El sistema no guardar o actualiza la información si no existen los campos requeridos. 3. El sistema muestra un mensaje de alerta especificando los campos obligatorios. 4. Se regresa al paso 5. B. Adjuntar archivo del avance 	

<ol style="list-style-type: none"> 1. En el paso 4. 2. El sistema muestra la opción de enviar el avance del proyecto en un archivo adjunto. 3. El actor carga el archivo adjunto. 4. El sistema continúa en el paso 5.
Excepciones:
B. No aplica

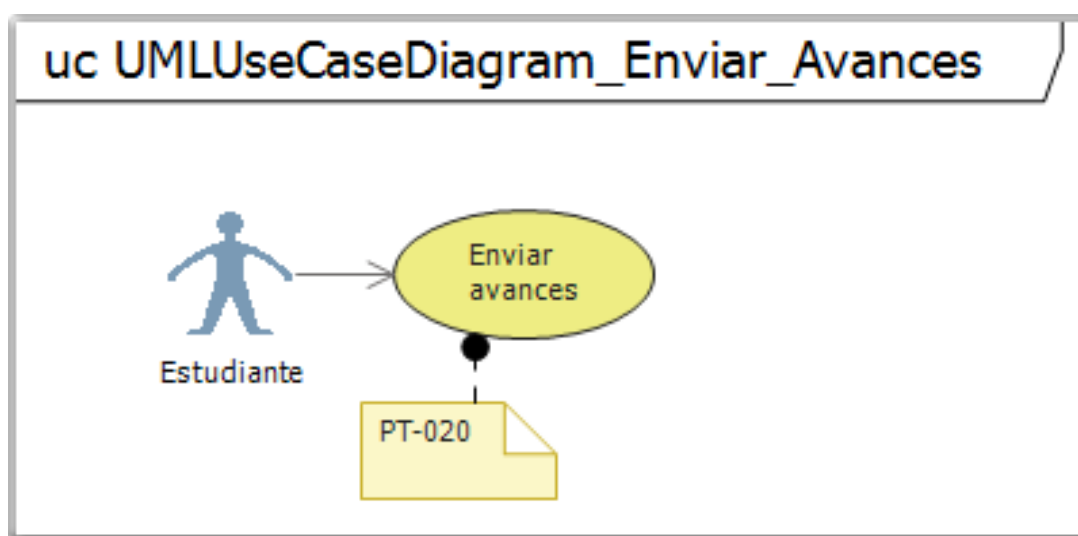


Ilustración 9. Caso de Uso PT-020

8.2.1.9 Caso de uso consultar estado de los avances de tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-018, básicamente el sistemas muestra al actor un formulario con la lista de tareas de un proyecto, el actor selecciona una tarea y el sistema le muestra una descripción detalla del estado de los avances de la tarea y sus posibles repuestas. El actor implicado es el estudiante, la información detallada del caso de uso se encuentra en la Tabla 33 y la gráfica asociada es la Ilustración 11.

Tabla 33. Caso de uso consultar estado de los avances de tareas de un proyecto

Identificador:	PT-021	Requerimiento asociado:	RF-018
Nombre:	Consultar estado de los avances de tareas de un proyecto		
Alias:	Consultar estado del avance		
Actores:	Estudiante		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto • Deben estar creados los avances del proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los avances de las tareas del proyecto 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para consultar los avances de las tareas del proyecto. 2. El sistema muestra la lista de las tareas asignadas del proyecto. 3. El actor selecciona una tarea. 4. El sistema muestra la lista de los avances de la tarea seleccionada por el actor. 5. El sistema muestra la descripción de los avances con sus estados y sus posibles repuestas. 6. El actor sale del caso de uso. 			
Flujo alternos:			
A. No aplica.			
Excepciones:			
A. No aplica			

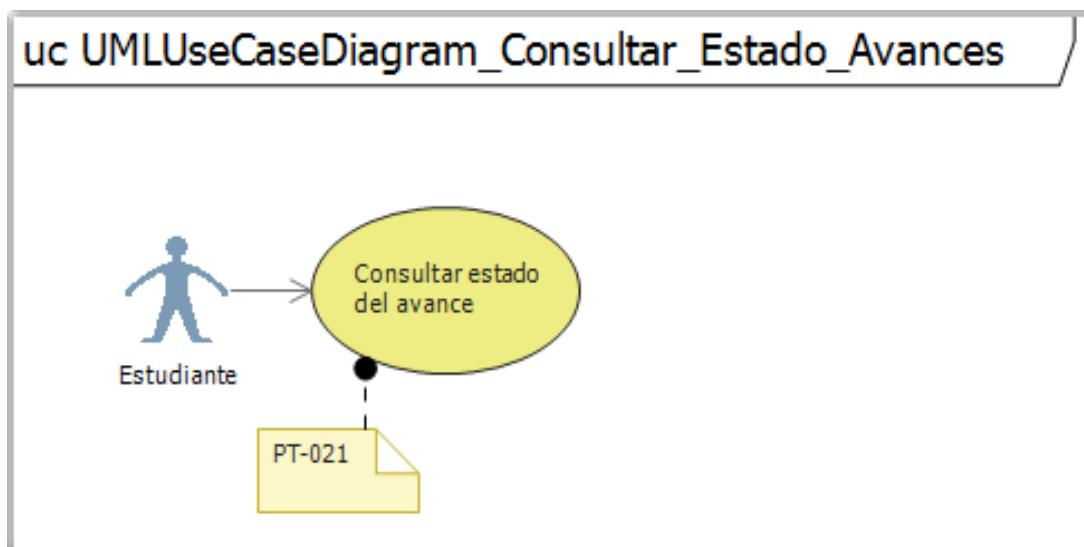


Ilustración 10. Caso de Uso PT-021

8.2.1.10 Caso de uso enviar alerta de estados de los avances de tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-019, básicamente el sistema muestra al actor un formulario para enviar las alertas de los estados de los avances de las tareas, si el actor selecciona enviar alertas, el sistema envía un correo electrónico a todos los integrantes y al tutor del proyecto con un reporte del estado de los avances de las tareas del proyecto. Los actores implicados son el tutor y el estudiante, la información detallada del caso de uso se encuentra en la Tabla 34 y la gráfica asociada es la Ilustración 12.

Tabla 34. Caso de uso enviar alerta de estados de los avances de tareas de un proyecto

Identificador:	PT-022	Requerimiento asociado:	RF-018
Nombre:	Enviar alerta de estados de los avances de tareas de un proyecto		
Alias:	Enviar alertas		
Actores:	Tutor y Estudiante		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto • Deben estar creados los avances del proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Envió de correo de alertas de los avances de las tareas del proyecto 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para enviar los estados de las tareas del proyecto. 2. El sistema consulta los estados de las tareas del proyecto del actor. 3. El sistema envía un correo con el reporte de los estados de las tareas del proyecto a los integrantes y al tutor del proyecto del actor. 4. El actor sale del caso de uso. 			
Flujo alternos:			
A. No aplica.			
Excepciones:			
A. No aplica			

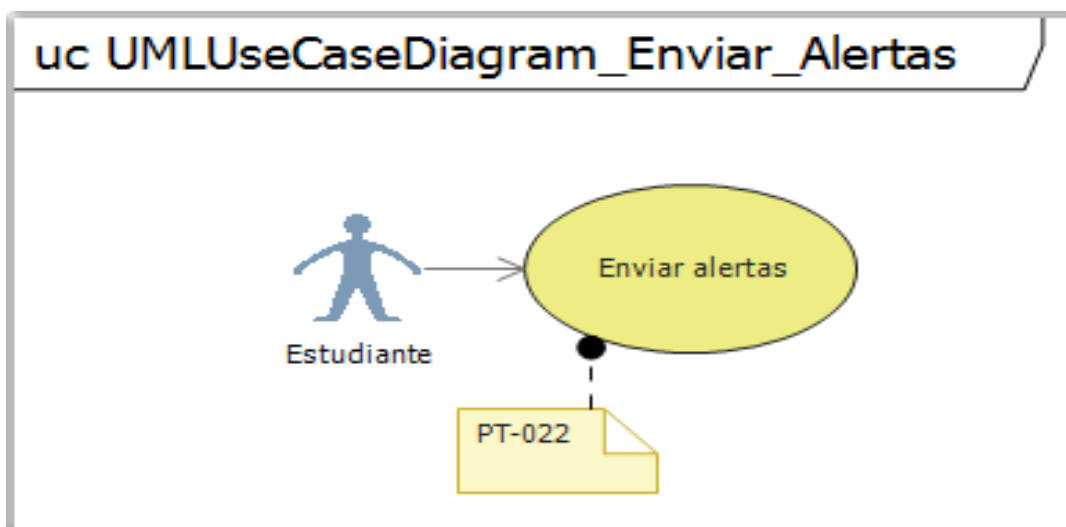


Ilustración 11.Caso de Uso PT-022

8.2.2 Integración Casos de Uso ProjectTracker

En los casos de uso cada actor tiene una serie de roles asociados para interactuar con el sistemas, en la ilustración 12 se muestra la integración de todos los casos de uso para obtener una perspectiva global del comportamiento del sistema.

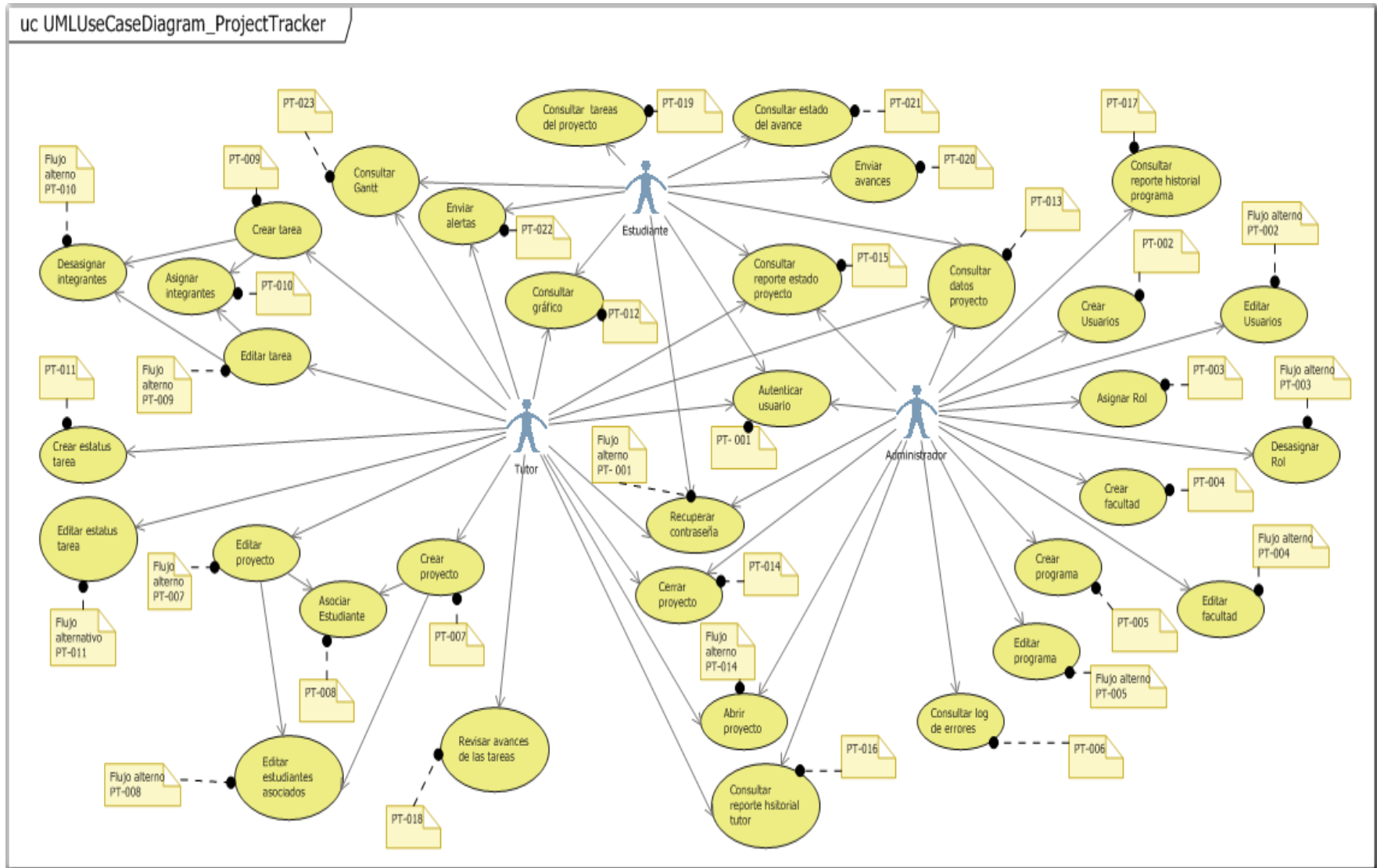


Ilustración 12. Integración Casos de Uso ProjectTracker

8.2.3 Diagramas de Clase

El diagrama de clase describe la estructura arquitectónica de la aplicación, para tener una idea clara del diagrama es necesario entender las funcionalidades de cada capa utilizada en la arquitectura.

8.2.3.1 *Capa de presentación*

En la Ilustración 16, la capa de presentación es llamada Genéricos y es la encargada principalmente de la interfaz con el usuario, en esta se almacenan la paginas web, controles de usuario, archivos java script y estilos, entre otros, además captura los datos para ser procesados y retorna al usuario una respuesta.

En la Ilustración 16, se anidan dos clases principales que son heredadas por la clases parciales de la paginas web, esta capa tiene una interacción directa con las capas de entidades “EntityLayerProjectTracker” y la capa de lógica “BusinessLogicLayerProjectTracker”, donde la capa entidades se encarga de transportar los datos en forma de objetos entre la capa de presentación y la capa de lógica.

8.2.3.2 Capa EntityLayerProjectTracker.

La capa llamada EntityLayerProjectTracker, es una librería de clases que se encarga de representar las tablas de la base de datos en entidades, de esta forma se agrupan los datos necesarios para una transacción sobre una tabla, esta agrupación se hace asignando los datos requeridos en la transacción a los atributos de un objeto de una entidad. Como se ha mencionado antes la función principal de esta capa es lograr una comunicación entre las capas de presentación, lógica y de acceso a datos, para que dichas capas puedan acceder a los datos enviados o recuperados de un método, función o transacción, la descripción detallada de las clases contenidas en esta capa se encuentran en la gráfica del Anexo 3.

La ilustración 13 muestra un ejemplo de las clases contenidas en la capa EntityLayerProjectTracker, la clase llamada PT_MENU representa la tabla PT_MENU de la base de datos, en esta clase todos los atributos están asociados a los campos de la tabla mencionada.

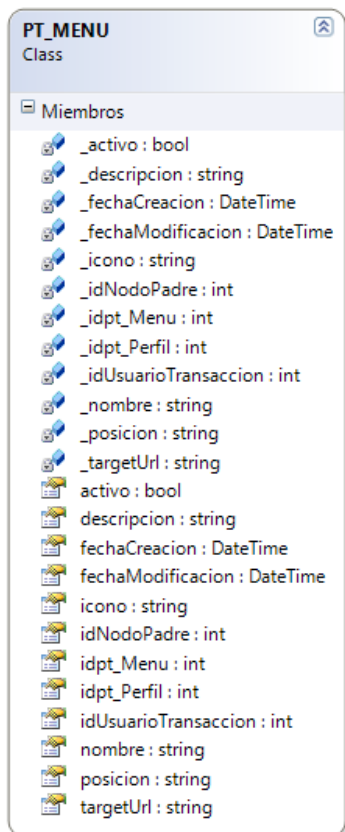


Ilustración 13. Clase de la capa EntityLayerProjectTracker

8.2.3.3 Capa BusinessLogicLayerProjectTracker

La capa de lógica "BusinessLogicLayerProjectTracker", es una librería de clases que se encarga de la lógica de la aplicación, se podría interpretar como la capa intermedia entre la interfaz de usuario y la base de datos, de esta forma sirve de puente de comunicación entre la capa de presentación y la capa de acceso a datos "DataAccessLayerProjectTracker", en esta capa se exponen métodos, funciones entre otras que realizan proceso lógicos de la aplicación, un ejemplo sería la construcción de los datos necesario para el menú de la aplicación, en primera instancia la capa de presentación envía a la capa de lógica un objeto con los datos del usuario

autenticado, luego la capa de lógica envía este objeto a la capa de acceso a datos para realizar una consulta del menú asociado al id del usuario autenticado y queda a la espera del retorno de los datos del menú, a continuación la capa de acceso a datos realiza la consulta y retorna un paquete de datos asociados al menú como direcciones URL, descripción del enlace, nombre del icono, orden de los ítems etc., posteriormente la capa de lógica analiza los datos aplicando la configuración de navegación asociada al usuario autenticado, al finalizar este proceso se retorna un paquete de datos a la capa de presentación para que esta arme la interfaz gráfica del menú. La descripción detallada de las clases contenidas en esta capa se encuentra en la gráfica del Anexo 4.

El ejemplo de la clase que encarga del menú en la capa BusinessLogicLayerProjectTracker, se muestra en la Ilustración 14. Esta tiene una variable global privada que hace referencia a la capa de acceso a datos y es instancia en cada método de la clase para gestionar transacciones asociadas al menú.

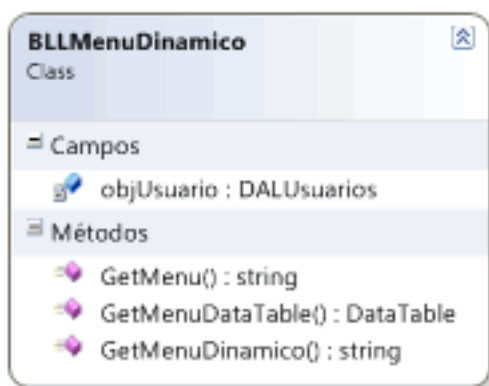


Ilustración 14. Clase de la capa BusinessLogicLayerProjectTracker

8.2.3.4 Capa DataAccessLayerProjectTracker

La capa de acceso a datos "DataAccessLayerProjectTracker", es una librería de clases que se encarga de las transacciones con la base de datos como lo son consultar, creación, actualización y eliminación registros, esta capa tiene una interacción directa con la capa de lógica, de esta forma la capa de lógica instancia a la capa de acceso a datos, esta capa en la mayoría de métodos retorna un valor o conjunto de datos para ser analizados posteriormente por la capa de lógica, por otra parte, la capa de acceso a datos hace referencia a librerías externas como lo es el driver de conexión a la base de datos de MySQL, esta librería expone las herramientas necesarias para comunicar la aplicación con la base de datos. La descripción detallada de las clases contenidas en esta capa se encuentra en la gráfica del Anexo 5.

La capa externos como se ha mencionado anteriormente contiene librería externas como MySQL o librerías del Framework que se utilizan en toda la aplicación. Un ejemplo de interacción entre las clases de la capa de acceso a datos se observa en la Ilustración 15, donde las clases DALFacultad_Programa y DALGenerica hacen referencia a la clase DALTransaccion.

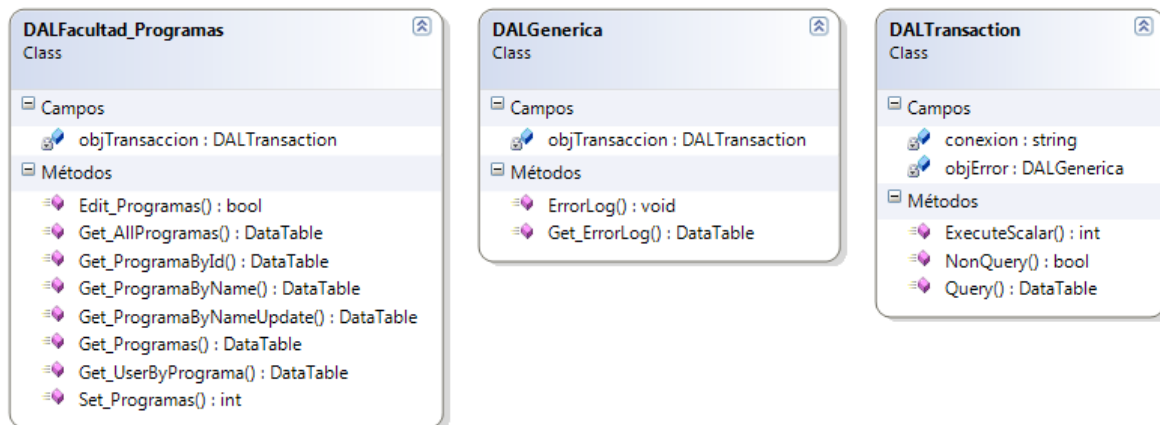


Ilustración 15. Clases DataAccessLayerProjectTracker

La Ilustración 16 muestra la interacción entre las capas descritas anteriormente.

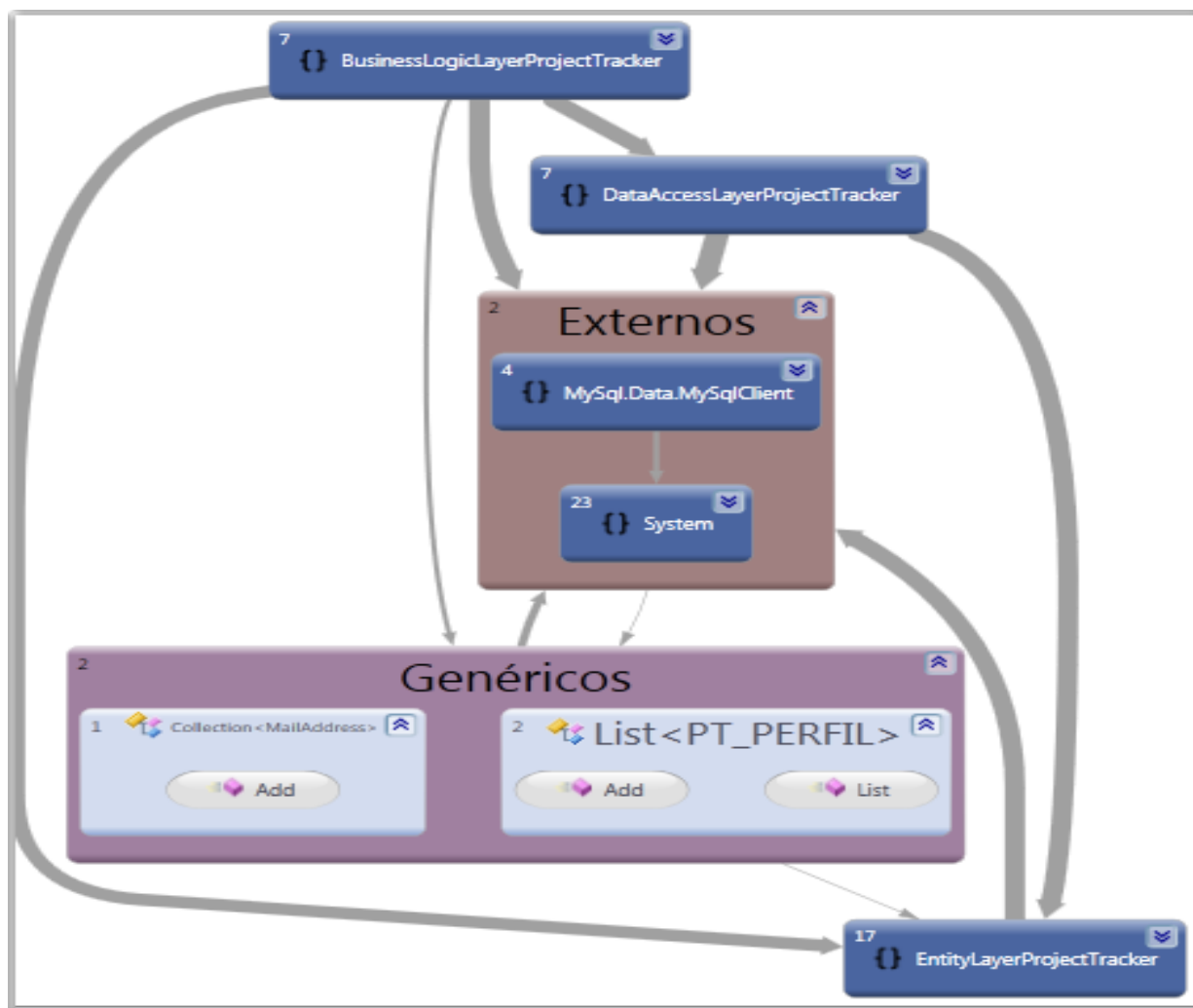


Ilustración 16. Diagrama de dependencia de ensamblados

8.2.4 Modelo Entidad Relación

En el modelo entidad relación se representa la estructura de la base de datos a través de las entidades, las cuales representan tablas que almacenan información de interés de la aplicación en la base de datos, a su vez las entidades poseen atributos que son proporcionales a los campos de cada tabla e indican un valor de interés para el sistema, las entidades del modelo entidad relación fueron abstraídas de los requerimiento del sistema; entre las entidades existen relaciones las cuales representan una asociación entre ellas, por ejemplo, la entidad pt_proyectos representa a los proyectos y pt_proyectos_tutores a los tutores, su relación básica describe que un proyecto tiene solo tutor asociado, pero un tutor puede tener muchos proyectos asociados. Las descripciones de los atributos de las entidades están en la sección del diccionario de datos. En la Ilustración 18 se muestra detalladamente el modelo entidad relación.

A continuación se describe las relaciones entre las entidades y el prefijo pt indica las iniciales del nombre de base de datos ProjectTracker:

- Entidad pt_alerta_tareas

Esta entidad tiene una relación con pt_proyectos, esta describe que un proyecto puede tener muchas alertas, esto indica que la aplicación puede enviar varias alertas del estado del proyecto durante el ciclo de vida del proyecto.

- Entidad pt_error_log

Esta entidad no tiene relaciones, en ella se almacenan los errores controlados de la aplicación.

- Entidad pt_estatus

Esta entidad tiene una relación de con pt_tarea_estatus, la cual describe que una tarea puede tener varios estados a lo largo del ciclo de vida del proyecto, los posibles estados de las tareas son creada, duplicada, en progreso, finalizada, invalida y sin solución.

- Entidad pt_estatus_avances

La entidad tiene una relación con pt_tarea_avances_estatus, la cual indica que un avance de una tarea puede tener varios estados como enviado, visto y revisado.

- Entidad pt_facultad

La entidad tiene una relación con pt_proyectos_integrantes, esta indica que una facultad tiene muchos programas y un programa pertenece a una sola facultad.

- Entidad pt_facultad_programas

La entidad tiene una relación con varias entidades, la primera es con pt_facultad que se explicó anteriormente en las relaciones de pt_facultad, la segunda es con pt_usuarios la cual indica que un usuario pertenece a un programa y un programa tiene muchos usuarios, la palabra usuario se refiere a estudiantes y profesores.

- Entidad pt_menu

La entidad tiene una relación con pt_perfil, la cual indica que un perfil de usuario tiene asociado varias paginad del menú y a su vez una página del menú está asociada a un solo perfil.

- Entidad pt_perfil

La entidad tiene una relación con `pt_usuario_perfil`, esta relación describe que un usuario puede tener varios perfiles no excluyentes y un perfil puede pertenecer a muchos usuarios, vale aclarar que la entidad `pt_usuario_perfil` sirve de tabla intermedia o puente con la entidad `pt_usuarios`, la razón de esta entidad intermedia es para poder representar bien la relación muchos a muchos entre el perfil y los usuarios, además de esta forma se simplifican las consultas que incluya dicha relación. Los perfiles de la aplicación son estudiante, tutor y administrador. La decisión de que los perfiles no sean excluyentes se tomó porque puede ocurrir que un administrador sea también tutor o un tutor sea estudiante, en si puede existir cualquier combinación entre los tipos de perfiles.

- Entidad `pt_proyectos`

Esta es una entidad que tiene varias relaciones, la primera es con la tabla `pt_proyectos_tutores`, esta relación describe que un proyecto tiene un tutor, pero un tutor puede tener varios proyectos, la segunda relación es con la entidad `pt_proyectos_integrantes` esta describe que un proyecto tiene muchos integrantes o estudiantes y que un estudiante solo pertenece a un proyecto, por ultimo tiene una relación `pt_alerta_tareas` la cual anteriormente fue descrita en las relaciones de la entidad `pt_alerta_tareas`.

- Entidad `pt_proyectos_integrantes`

Esta entidad tiene una relación con `pt_proyectos` descrita anteriormente en las relaciones de la entidad `pt_proyectos`.

- Entidad `pt_proyectos_tutores`

Esta entidad tiene una relación con pt_proyectos descrita anteriormente en las relaciones de la entidad pt_proyectos.

- Entidad pt_tarea

Esta entidad posee varias relaciones, la primera es con pt_tarea_integrantes y consiste en que una tarea tiene varios integrantes o estudiantes y un integrante tiene muchas tareas, la segunda relación es con pt_tarea_avances esta indica que una tarea puede tener muchos avances y un avance pertenece a una tarea, la tercera relación es con pt_tarea_estatus esta relación consiste en que una tarea puede tener varios estados, los estados están descritos en la entidad pt_estatus, la última relación es con la entidad pt_tarea_archivos esta consiste en que una tarea puede tener varios archivos adjuntos y un archivo adjunto pertenece a una sola tarea.

- Entidad pt_tarea_archivos

La entidad tiene una relación con pt_tarea, dicha relación fue descrita en las relaciones de la entidad pt_tarea.

- Entidad pt_tarea_avances

La entidad tiene varias relaciones, la primera relación es con la entidad pt_tarea, dicha relación fue descrita en las relaciones entidad pt_tarea, la segunda es con la entidad pt_tarea_avances_estatus, esta relación esta descrita en las relaciones de la entidad pt_estatus_avances, la tercera relación es con pt_tarea_avances_respuestas indica que un avance de una tarea puede tener varias repuestas y una respuesta

pertenece a un solo avance, la última relación es con la entidad pt_tarea_avances_archivos, esta relación indica que un avance de una tarea puede tener varios archivos adjuntos y un archivo adjunto pertenece a un solo avance.

- Entidad pt_tarea_avances_archivos

La entidad tiene una relación con pt_tarea_avances, dicha relación fue descrita en las relaciones entidad pt_tarea_avances.

- Entidad pt_tarea_avances_estatus

La entidad tiene una relación con pt_tarea_avances, dicha relación fue descrita en las relaciones entidad pt_tarea_avances.

- Entidad pt_tarea_avances_respuestas

La entidad tiene una relación con pt_tarea_avances, dicha relación fue descrita en las relaciones entidad pt_tarea_avances.

- Entidad pt_tarea_avances_respuestas_archivos

Esta entidad tiene una relación con pt_tarea_avances_respuestas, esta relación indica que una respuesta de un avance puede tener varios archivos adjuntos como respuesta y a su vez un archivo solo pertenece a una respuesta.

- Entidad pt_tarea_estatus

La entidad tiene una relación con pt_tarea, dicha relación fue descrita en las relaciones entidad pt_tarea.

- Entidad pt_tarea_integrantes

La entidad tiene una relación con pt_tarea, dicha relación fue descrita en las relaciones entidad pt_tarea.

- Entidad pt_usuario_perfil

La entidad tiene una relación con pt_perfil, dicha relación fue descrita en las relaciones entidad pt_perfil.

- Entidad pt_usuarios

Esta entidad tiene varias relaciones, la primera es con pt_usuario_perfil esta relación sirve de puente o tabla intermedia con la entidad pt_perfil y ayuda a representar de manera óptima la relación de muchos a muchos, lo cual indica que un usuario puede tener varios perfiles y un perfil puede pertenecer a varios usuarios, la segunda relación es con la entidad pt_proyectos_tutores es indica que un usuario con el perfil de tutor puede tener varios proyectos y un proyecto pertenece a solo tutor, la tercera relación es con la entidad pt_proyectos_integrantes la cual indica que un usuario con el perfil de estudiante pertenece a un proyecto y un proyecto puede tener muchos integrantes, por ultimo esta la relación con la entidad pt_facultad_programas, la cual indica que un usuario pertenece a un programa de una facultad y un programa puede tener muchos usuarios.

Para una mayor calidad en la imagen en la Ilustración 18 del modelo entidad relación véase el Anexo 6.

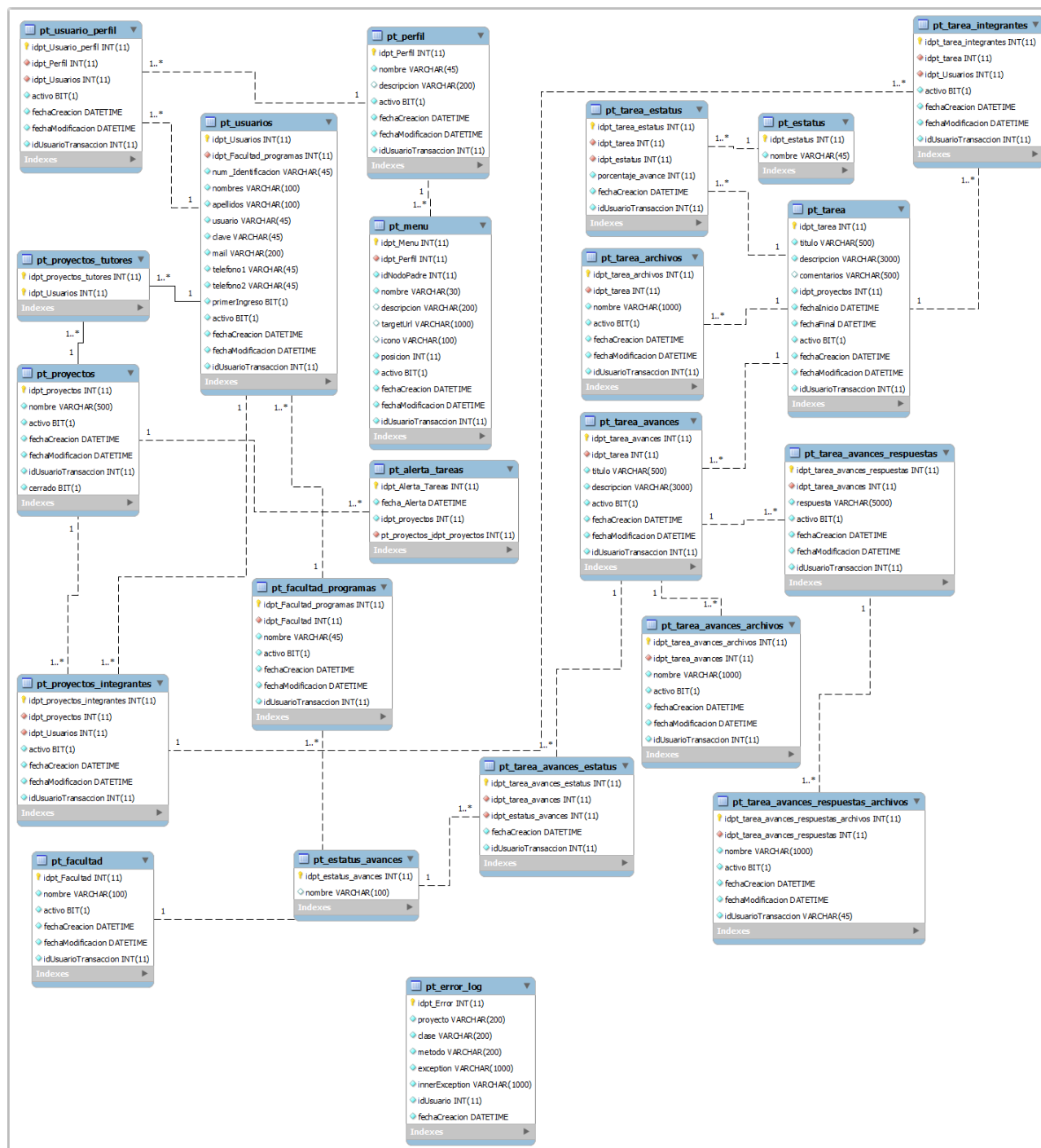


Ilustración 18. Modelo Entidad Relación

8.2.5 Diccionario de datos de la base de datos Project Tracker

En el diccionario de datos se describen todas las tablas de la base de datos, dicha descripción contiene una breve explicación de los datos que debe almacenar la tabla, también contiene los nombres de los campos con sus tipos de datos, restricciones, valores por defecto, además indica si los campos son llaves primarias o foráneas, si el campo recibe datos nulos, si son únicos o si el campo se genera de forma auto numérica, en resumen el diccionario de datos contiene la descripción de la estructura de cada tabla de la base de datos.

El diccionario de datos de la base de datos Project Tracker se encuentra en el anexo 7.

8.3 Roles de Usuario

La aplicación deberá tener tres tipos de roles de usuario “Administrador, tutor y estudiante”, cada uno de ellos asociados a cierto perfil que le permitirá ejecutar transacciones y visualizar formularios de la aplicación asociados a dicho rol.

- Un usuario puede tener cualquier combinación de roles entre estos tres tipos.
- Si el usuario tiene rol de tutor y administrador tiene acceso a administrar todos los proyectos.

Tabla 35. Roles de usuario

Rol	Permisos
Estudiante	<ul style="list-style-type: none"> • Consultar los datos del proyecto asignado. • Consultar tareas del proyecto asignado. • Consultar diagrama de Gantt del proyecto asignado.

	<ul style="list-style-type: none"> • Consultar gráficos de avances del proyecto asignado. • Enviar avances del proyecto asignado. • Consultar avances, estados y respuestas de los avances
Tutor	<ul style="list-style-type: none"> • Administrar proyectos asignados • Administrar tareas de los proyectos asignados • Administrar estado de las tareas de los proyectos asignados • Consultar gráficos de avances de los proyectos asignados. • Revisar y enviar respuestas de los avances de los proyectos asignados • Consultar reportes de estados de proyecto asignado, historial de los proyectos asignados. • Envío de alertas de los proyectos asignados. • Consultar diagrama de Gantt de los proyectos asignados.
Administrador	<ul style="list-style-type: none"> • Administración Usuarios • Asignación de roles • Administración de Facultades y programas • Consultar log de errores. • Consultar proyectos. • Cerrar o abrir proyectos • Consultar reportes de estados de proyecto, historial de proyectos por tutor y por programa. • Administrar todos los proyectos • Asociar tutores a proyectos • Administrar todas las tareas. • Administrar todos los estados de la tareas • Consultar diagrama de Gantt de todos los proyecto. • Consultar todos los gráficos de avances

	<ul style="list-style-type: none">• Revisar y enviar respuestas de los avances de todos los proyectos.
--	--

En la ilustración 19, representa la configuración de permisos de acceso por rol en el archivo Webconfig de la App. En este segmento de configuración XML, se asigna el acceso a carpetas que contienen las páginas web, estas carpetas están ubicadas en la raíz del proyecto y sus nombres son Admin, Tutor y Estudiante, cada carpeta contiene una serie de páginas asociadas a cada rol.

```

<location path="Admin">
  <system.web>
    <authorization>
      <allow roles="ADMINISTRADOR"/>
      <deny users="*" />
    </authorization>
  </system.web>
</location>
<location path="Tutor">
  <system.web>
    <authorization>
      <allow roles="TUTOR"/>
      <deny users="*" />
    </authorization>
  </system.web>
</location>
<location path="Estudiante">
  <system.web>
    <authorization>
      <allow roles="ESTUDIANTE"/>
      <deny users="*" />
    </authorization>
  </system.web>
</location>

```

Ilustración 19. Configuración de acceso

8.4 Codificación

En esta etapa creamos todo el código asociado al modelo de 4 capas, dividimos la solución en 4 proyectos de desarrollo.

El primer proyecto contiene la capa de presentación la cual se encarga de interactuar con el usuario final, esta ofrece una interfaz gráfica para capturar los datos requeridos para realizar un proceso del software y a su vez muestra los resultados de dicho proceso al usuario final. Esta

capa contiene las pagina web, las master page, los estilos de las paginas, archivos de java script y JQuery entre otros, además en esta capa se gestiona las sesiones de la aplicación, básicamente estas sesiones almacenan datos importantes para las transacciones que ejecuta el usuario autenticado, por otra parte esta capa también se encarga de la gestión de seguridad y configuración de la aplicación a través de los archivos Webconfig y Global.asax, en el archivo Webconfig se configuran las cadenas de conexión, acceso a archivos y carpetas, versiones de Framework, configuración del correo electrónico que utiliza la aplicación, versiones de las DLL referenciadas y muchos más elementos de configuración, en el archivo Global.asax esencialmente se manejan eventos asociados a las variables de sesión y de aplicación como los son inicio y final de sesión, inicio y finalización de la aplicación y errores de aplicación entre otros. También a través de esta capa se hace referencia a librerías o DLL, estas pueden ser internas o externas, en el proyecto esta capa hace referencia a librerías internas asociadas a las diferentes capas de la aplicación como las librerías de lógica del negocio o de entidades, también hace referencia a librerías externas como el driver de conexión de MySQL, librerías de controles web como la Suite de Obout, Ajax Control Tool Kit y librerías de ItexSharp para generar reportes.

La capa de negocio contiene la lógica del negocio y provee un puente de comunicación entre la capa de acceso a datos y la capa de presentación, este proyecto es una librerías de clases se llama BusinessLogicLayerProjectTracker y se encarga de la lógica previa que se le realizan a los datos antes de una transacción con la base de datos o previo a ser enviado a la capa de presentación como una respuesta de la capa de acceso a datos. Esencialmente esta capa es instanciada por la capa de presentación, si se ejecute algún método de la capa de lógica esta a su vez analiza los datos enviados y aplica la lógica requerida, luego envía estos dato ya preparados

a la capa de acceso a datos para ser procesados en una transacción con la base de datos, a continuación la capa de acceso a datos retorna un valor o un grupo de datos a la capa de lógica, enseguida la capa de lógica de ser necesario aplica los algoritmos o lógica necesaria y envía esta información de vuelta a la capa de presentación para que los resultados sean mostradas al usuario.

La capa de persistencia o de acceso a datos contiene toda la lógica de comunicación transaccional con la base de datos, como se menciona anteriormente esta capa se conecta a la base de datos y ejecuta las transacciones indicada estas pueden ser select, insert, update o delete, en general se utilizó procedimientos almacenados, ya que provee una buena velocidad de respuesta frente a otros métodos utilizados en .Net, la estructura de las clases es realmente sencilla, en general los métodos contenidos en las clases retorna un conjunto de datos si el procedimiento almacenado es un select, si el procedimiento almacenado es un insert devuelve el id del registro creado y si el procedimiento almacenado es un update o delete retorna un valor booleano que indica si se realizó la transacción o no, finalmente en todas las clases se hace referencia a una clase llama DALTransaction.cs la cual se encarga de ejecutar las transacciones y retornar los resultados.

Por ultimo tenemos la capa de entidades que provee un puente de intercomunicación entre las capas anteriores, esta permite que viajen los datos mediante objetos de entidades que representan tablas de la base de datos, por ejemplo la tabla pt_usuarios es representada por la entidad PT_USUARIOS y sus atributos son cada uno de los campos de la tabla mencionada, para aclarar la intercomunicación entre las capas está el siguiente ejemplo, se desea consultar los datos del usuario por su id, la secuencia inicial seria la capa de presentación captura el id, luego crea un objeto de tipo PT_USUARIOS y a su atributo idUsuario le asigna el valor capturado, luego crea

un objeto de la capa de lógica que gestione los usuarios y que contenga un método para la consulta de usuarios, este método espera como parámetro un objeto de tipo PT_USUARIOS y retorna un conjunto de resultados, luego de capturar el objeto si es necesario aplica el algoritmo indicado y construye un objeto de la capa de acceso a datos y llama el método que consulta el usuario, a continuación la capa de acceso a datos en el método invocado también espera como parámetro un objeto de tipo PT_USUARIOS, luego este método utiliza e implementa los métodos y propiedades expuestas por el driver de MySQL para hacer la transacción, para este caso se recupera el atributo idUsuario del objeto PT_USUARIOS y se ejecuta el procedimiento almacenado que consulta al usuario por id, luego se retorna el conjunto de datos a la capa de lógica y esta a su vez a la capa de presentación para ser mostrados en la interfaz gráfica al usuario final.

Unos de los desarrollos de codificación más complejos de la aplicación fue el desarrollo de un gráfico del diagrama de Gantt. Debido a que los controles para asp.net que ofrece el mercado para realizar diagramas de Gantt son muy costosos y los controles gratis que se encuentran son de mala calidad, por tal razón se decidió crear un control de usuario personalizado que cumpliera las función básicas de un diagrama de Gantt, este control permite asignar tareas con sus recursos de personal, datos, archivos adjuntos y tiempos de ejecución, dicho grafico ilustra un cronograma y una línea de tiempo que permite las tareas secuenciales o paralelas. El desarrollo de este control es uno de los grandes logros de la aplicación.

8.4.1 Prototipo

El código de esta aplicación fue hecho en C# asp.net con Framework 4.0 en Visual Studio 2010, se utilizaron para la interfaz gráfica los controles de la Suite de Obout, JQuery y Ajax y el diseño de estilo fue hecho en CS5 Fireworks, el modelado de la aplicación fue en Modeling Project con Visual Studio 2010 y la base de datos fue construida en MySQL 5.5 con la interfaz gráfica de Workbench 5.2.

Desde la Ilustración 20 a la Ilustración 22 se muestra algunas imagen que hacen referencia a alguna página web del prototipo creado.

- La Ilustración 20 muestra la pantalla de inicio de la aplicación la cual contiene el logo del nombre del proyecto y a su vez sirve como botón de inicio para mostrar el formulario de autenticación.



Ilustración 20. Página principal

- En la Ilustración 21 se muestra la página de creación y edición de usuarios, en esta se capturan los datos como nombre de usuario, clave, dirección de correo, programa al que pertenece entre otros, esta página esa asociada al perfil de administrador.

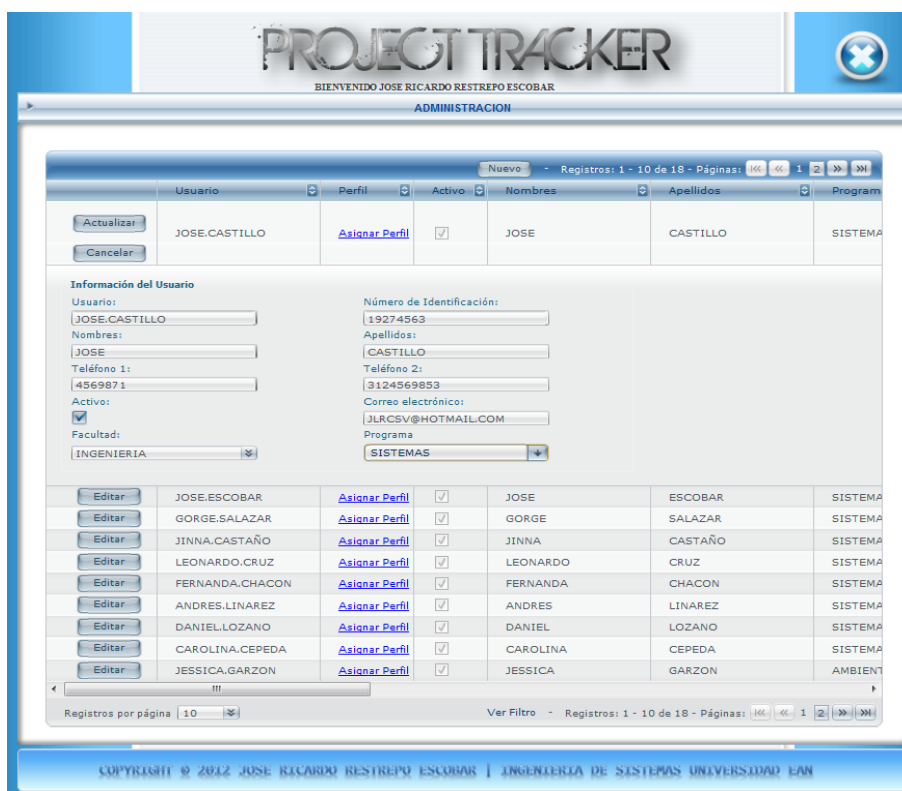


Ilustración 21. Página para la creación y edición de usuario

- En la Ilustracion 22 se muestra la página que contiene el gráfico del progreso de cada tarea de un proyecto, el flujo de la pagina consiste en que el usuario selecciona un proyecto a los que tiene acceso asociados a su rol y el sistema crea el grafico.

Ilustración 22. Página que muestra en un gráfico el progreso de cada tarea de un proyecto

8.5 Pruebas

La primera técnica que se utilizó para realizar las pruebas fue utilizar los test unitarios mediante un proyecto de Testing en Visual Studio 2010, en estos se examinaban el comportamiento de los métodos de la aplicación de cada clase en todas las capas.

- Ejemplo de test sobre un método de la aplicación.

```
[TestMethod]
public void TestMethod_GetProyectos()
{
    BLLProyectos objProyecto = new BLLProyectos();
    PT_USUARIOS entityUsuario = new PT_USUARIOS();
    entityUsuario.idpt_Usuarios = 0;
    entityUsuario.activo = true;
    DataTable dt = objProyecto.Get_Proyectos(entityUsuario);
    bool hasData = true;
    if (dt.Rows.Count > 0)
    {
        Assert.AreEqual(hasData, true);
    }
}
```

Ilustración 23. Prueba de la aplicación

- Resultado del test.

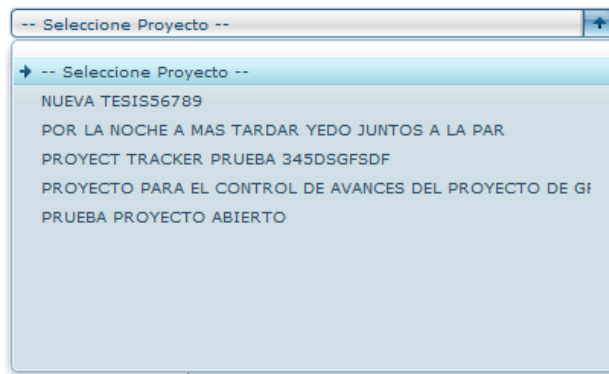


Ilustración 24. Datos enlazados correctamente

Ejecución de pruebas completado Resultados: 1/1 correctas; Elementos comprobados: 0			
Resultado	Nombre de la prueba	Proyecto	Mensaje de error
Pasada	TestMethod_GetProyectos	TestProjectTracker	

Ilustración 25. Resultados de la prueba

- La segunda técnica utilizada fue realizar test unitario en la interfaz de usuario, este consistía en definir una prueba de una rutina de la aplicación en una plantilla la cual debería ser ejecutada por un usuario final. A continuación se representa un ejemplo de una plantilla para el test de consulta de reportes del historial de proyectos por tutor.

Tabla 36. Caso de prueba

Id Caso de prueba:	CP-014	Sistema:	Project Tracker	
Diseñado por:	José Ricardo Restrepo Escobar	Subsistema:	Consulta de reportes del historial de proyecto por tutor	
Ejecutado por:	José Ricardo Restrepo Escobar	Fecha de diseño	Fecha de Ejecución	
		04/11/2012	04/11/2012	
Descripción corta:	Prueba para el módulo de reportes de los proyectos de un tutor			
Precondiciones	El usuario debe estar autenticado. El usuario debe tener el rol de tutor o administrador El rol de usuario debe tener acceso a la página de reportes			
Procedimiento				
Paso	Acción	Respuesta esperada del sistema	Aprobado	Fallo

1	Seleccionar en el menú principal la opción de reportes	El sistema muestra el menú de navegación y redirige a la página solicitada	X	
2	Seleccionar un tutor de la lista	El sistema muestra una lista desplegable con los nombre y apellidos de los tutores	X	
3	Seleccionar la fecha inicial y final del reporte	El sistema muestra dos listas fechas, una corresponde a la fecha inicial y la otra a la fecha final	X	
4	Dar clic en el botón ver reporte	El sistema muestra un botón para generar el reporte	X	
5	Se ejecuta la post - condición 001		X	
Post-condiciones				
<ol style="list-style-type: none"> 1. El sistema consulta los reportes de los proyectos asociados a un tutor, filtrados por un rango de fechas asociadas a la fecha de creación de los proyectos. 2. El sistema genera un reporte en formato PDF. 				
Flujos alternativos				
Flujo		Respuesta esperada del sistema		
1. El usuario selecciona un tutor al que no se le asignaron proyectos y presiona ver reporte.		El sistema muestra un mensaje de alerta indicando que el tutor no tiene proyectos asociados		
2. El usuario no selecciona un tutor		El sistema indica que es obligatorio		

y presiona ver reporte.	seleccionar un tutor.
-------------------------	-----------------------

A través de los video tutoriales se puede apreciar el correcto funcionamiento del software y a su vez sirve de guía al usuario para el manejo de la aplicación (véase Ilustración 113).



Ilustración 26. Video tutorial para la creación de tareas de un proyecto

8.6 Instalación

En la Tabla 7 se describe los requisitos mínimos para el funcionamiento de la aplicación, tales como los sistemas operativos compatibles, los requisitos de hardware, motor de base de datos, versión de Framework y navegadores web compatibles entre otros.

Tabla 37. **Requisitos mínimos para el funcionamiento de la aplicación**

<p>Sistema operativo compatibles</p>	<ul style="list-style-type: none"> • Windows Server 2003 SP2 • Windows Vista SP1 o posterior • Windows Server 2008 (no admitido en el rol Server Core) • Windows 7 • Windows Server 2008 R2 (no admitido en el rol Server Core) • Windows 7 SP1 • Windows Server 2008 R2 SP1
<p>Requisitos de hardware</p>	<ul style="list-style-type: none"> • Mínimo recomendado: Pentium 1 GHz o superior con 512 MB de RAM o más • Espacio en disco mínimo para la publicación de la aplicación web: 50MB
<p>Motor de base de datos</p>	<ul style="list-style-type: none"> • MySQL Server 5.5
<p>Navegadores Compatibles</p>	<ul style="list-style-type: none"> • Firefox Mozilla 16.0.2 • Windows Internet Explorer • Google Chrome 23.0.127164
<p>Versión del IIS</p>	<ul style="list-style-type: none"> • Microsoft Internet Information

	Services (IIS) 6.0 y superiores
Versión del Framework	<ul style="list-style-type: none">• Microsoft .NET Framework 4
Otros	<ul style="list-style-type: none">• Adobe Flash Player 11.5.502.110• Adobe Reader XI (11.0)

El procedimiento para la instalación de la aplicación en un ambiente de pruebas o producción, se encuentra en el anexo 8.

9 CONCLUSIONES

Analizando las herramientas utilizadas en el desarrollo de Microsoft Visual Studio 2010, se concluye que es una buena decisión a la hora de desarrollar software, esta ofrece varias opciones de herramientas gratuitas para plasmar la creatividad, el ingenio, la capacidad de resolver problemas y hasta el espíritu emprendedor de crear aplicaciones que son fuentes de negocio, además brinda un desarrollo ágil y tiempos más cortos de desarrollo, con gran calidad en las métricas de código, escalabilidad y modelado. Sin dejar atrás a MySQL Workbench, que además es open source, es interesante resaltar la propuesta que ofrece un ambiente gráfico al usuario más amigable con ciertos aires a la interfaz de usuario de SQL Server, dicha interfaz permite un desarrollo de la base de datos más rápido frente a otras interfaces de usuario para MySQL que ofrece el mercado.

Una de las principales conclusiones del proyecto es el beneficio y el gran impacto que genera la aplicación en la universidad EAN. Es evidente que en la universidad EAN no existe alguna aplicación que lleve el control sobre los avances y las tutorías para los proyectos de grado. Mediante el uso de la aplicación Project Tracker se puede obtener los tiempos de ejecución, estado del proyecto y sus tareas, en si toda la traza del proyecto, por tal se deduce que el proyecto realizado es de vital importancia para la universidad EAN. También es de alto impacto para las otras facultades diferentes a Ingeniería ya que tienen una herramienta que los acerca aún más a la utilización de las TIC's

La necesidad y la ambición de resolver problemas es el combustible para todo desarrollador de software, pero vale aclarar que su talón de Aquiles es el perfeccionismo improductivo, tales casos se vivieron en todo el ciclo de desarrollo del proyecto, a veces se excedían los tiempos de

desarrollo por tratar de brindar una aplicación amigable, con interfaces gráficas agradables, fáciles de manejar y buen rendimiento, pero esa la clave y esencia de un buen desarrollo.

Esta aplicación es una gran herramienta para prevenir el plagio, desarrollo de proyectos por terceros o directamente la compra del proyecto, pues al tener la traza del proyecto se regula la interacción de los estudiantes con el este, también obliga a que los tutores hagan un acompañamiento real a los estudiantes en el desarrollo del proyecto, previniendo a si proyectos de grado de baja calidad o con demasiadas correcciones.

Para finalizar concluyo que el desarrollo de este proyecto fue una experiencia fructífera para mi crecimiento personal y profesional.

10 LISTA DE REFERENCIAS

- Ceballos, J. (2006). *Enciclopedia de Microsoft*. Madrid, España: Alfaomega.
- Deitel, H. M. (2007). *Cómo programar C#*. México: Pearson Educación.
- Fowler, M. (1999). *UML gota a gota*. México: Pearson Educación.
- Marín Sierra, A. (2007). Desarrollo de aplicaciones web con ASP.NET 2.0. En A. M. Sierra, *Desarrollo de aplicaciones web con ASP.NET 2.0* (págs. 2-8, 107-114). México D.F.: Alfaomega Grupo Editor, S.A.
- Microsoft Corporation. (2012). *3-tier Architecture with ASP.NET 2.0*. Recuperado el 1 de abril de 2012, de [www.msdn.microsoft.com: http://msdn.microsoft.com/en-us/library/aa581769.aspx](http://msdn.microsoft.com/en-us/library/aa581769.aspx)
- Microsoft Corporation. (2012a). *Definición y explicación de un archivo .DLL*. Recuperado el 5 de Diciembre de 2012, de [www.msdn.microsoft.com: http://support.microsoft.com/kb/87934/es](http://support.microsoft.com/kb/87934/es)
- Microsoft Corporation. (2012b). *www.msdn.microsoft.com*. Recuperado el 1 de abril de 2012, de <http://msdn.microsoft.com/en-us/library/aa581769.aspx>
- Oracle Corporation. (2011). *MySQL 5.0 Reference Manual*. Recuperado el 27 de Mayo de 2012, de [http://dev.mysql.com/: http://dev.mysql.com/doc/refman/5.0/es/index.html](http://dev.mysql.com/doc/refman/5.0/es/index.html)
- Prieto Tinoco, J. I., & Cortes Leiva, E. (2006). *Sistemas y Aplicaciones Informáticas*. Sevilla: Mad, S.I.
- Rumbaugh, J. (1999). *El Lenguaje Unificado del Modelado*. Madrid, España: Addison Wesley.
- Stevens, P. (2002). *Utilización de UML en ingeniería de Software con Objetos y Componentes*. Madrid: Pearson Education.

LICENCIA DE USO – AUTORIZACIÓN DE LOS AUTORES

Actuando en nombre propio identificado (s) de la siguiente forma:

Nombre Completo Jose Ricardo Restrepo Escobar

Tipo de documento de identidad: C.C. T.I. C.E. Número: 00051283

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

El (Los) suscrito(s) en calidad de autor (es) del trabajo de tesis, monografía o trabajo de grado, documento de investigación, denominado:

Aplicación web para el control y seguimiento de avances del proyecto de grado "Project Tracker"

Dejo (dejamos) constancia que la obra contiene información confidencial, secreta o similar: SI NO
(Si marqué (marcamos) SI, en un documento adjunto explicaremos tal condición, para que la Universidad EAN mantenga restricción de acceso sobre la obra).

Por medio del presente escrito autorizo (autorizamos) a la Universidad EAN, a los usuarios de la Biblioteca de la Universidad EAN y a los usuarios de bases de datos y sitios webs con los cuales la Institución tenga convenio, a ejercer las siguientes atribuciones sobre la obra anteriormente mencionada:

- A. Conservación de los ejemplares en la Biblioteca de la Universidad EAN.
- B. Comunicación pública de la obra por cualquier medio, incluyendo Internet
- C. Reproducción bajo cualquier formato que se conozca actualmente o que se conozca en el futuro
- D. Que los ejemplares sean consultados en medio electrónico
- E. Inclusión en bases de datos o redes o sitios web con los cuales la Universidad EAN tenga convenio con las mismas facultades y limitaciones que se expresan en este documento
- F. Distribución y consulta de la obra a las entidades con las cuales la Universidad EAN tenga convenio

Con el debido respeto de los derechos patrimoniales y morales de la obra, la presente licencia se otorga a título gratuito, de conformidad con la normatividad vigente en la materia y teniendo en cuenta que la Universidad EAN busca difundir y promover la formación académica, la enseñanza y el espíritu investigativo y emprendedor.

Manifiesto (manifestamos) que la obra objeto de la presente autorización es original, el (los) suscritos es (son) el (los) autor (es) exclusivo (s), fue producto de mi (nuestro) ingenio y esfuerzo personal y la realizó (zamos) sin violar o usurpar derechos de autor de terceros, por lo tanto la obra es de exclusiva autoría y tengo (tenemos) la titularidad sobre la misma. En vista de lo expuesto, asumo (asumimos) la total responsabilidad sobre la elaboración, presentación y contenidos de la obra, eximiendo de cualquier responsabilidad a la Universidad EAN por estos aspectos.

En constancia suscribimos el presente documento en la ciudad de Bogotá D.C.,

NOMBRE COMPLETO: <u>Jose Ivando Restrepo F</u>	NOMBRE COMPLETO: _____
FIRMA: <u>[Firma]</u>	FIRMA: _____
DOCUMENTO DE IDENTIDAD: <u>80051083</u>	DOCUMENTO DE IDENTIDAD: _____
FACULTAD: <u>Ingeniería</u>	FACULTAD: _____
PROGRAMA ACADÉMICO: <u> sistemas</u>	PROGRAMA ACADÉMICO: _____

NOMBRE COMPLETO: _____	NOMBRE COMPLETO: _____
FIRMA: _____	FIRMA: _____
DOCUMENTO DE IDENTIDAD: _____	DOCUMENTO DE IDENTIDAD: _____
FACULTAD: _____	FACULTAD: _____
PROGRAMA ACADÉMICO: _____	PROGRAMA ACADÉMICO: _____

Fecha de firma: 12 diciembre 2012

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Trabajo dirigido

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Tutor

Ingeniero Alexander García Pérez

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

TABLA DE ILUSTRACIONES

ILUSTRACIÓN 1. CASO DE USO PT-001	8
ILUSTRACIÓN 2. CASO DE USO PT-002	11
ILUSTRACIÓN 3. CASO DE USO PT-003	13
ILUSTRACIÓN 4. CASO DE USO PT-004	15
ILUSTRACIÓN 5. CASO DE USO PT-005	18
ILUSTRACIÓN 6. CASO DE USO PT-006	19
ILUSTRACIÓN 7. CASOS DE USO PT- 007 Y PT- 008	22
ILUSTRACIÓN 8. CASOS DE USO PT-009 Y PT-010.....	27
ILUSTRACIÓN 9. CASO DE USO PT-011	31
ILUSTRACIÓN 10. CASO DE USO PT - 012.....	33
ILUSTRACIÓN 11. CASO DE USO PT-013	35
ILUSTRACIÓN 12. CASO DE USO PT-014	37
ILUSTRACIÓN 13. CASOS DE USO PT-15, PT-16 Y PT-17	42
ILUSTRACIÓN 14. CASO DE USO PT-018	45
ILUSTRACIÓN 15. CASO DE USO PT-019	47
ILUSTRACIÓN 16. CASO DE USO PT-020	49
ILUSTRACIÓN 17. CASO DE USO PT-021	51
ILUSTRACIÓN 18.CASO DE USO PT-022	53
ILUSTRACIÓN 19. CASO DE USO PT-023	55

ÍNDICE DE TABLAS

TABLA 1. AUTENTICAR USUARIOS DE LA APLICACIÓN.....	6
TABLA 2. CASO DE USO ADMINISTRAR USUARIOS	9
TABLA 3. ASIGNAR ROLES DE USUARIOS.....	12
TABLA 4. CASO DE USO ADMINISTRAR FACULTAD	13
TABLA 5. CASO DE USO ADMINISTRAR PROGRAMAS	16
TABLA 6. CASO DE USO CONSULTAR LOG DE ERRORES.....	18
TABLA 7. ADMINISTRAR PROYECTOS	20
TABLA 8. CASO DE USO ASOCIAR ESTUDIANTE A PROYECTO	23
TABLA 9. CASO DE USO ADMINISTRAR TAREAS	25
TABLA 10. CASO DE USO ASOCIAR ESTUDIANTES A TAREA	28
TABLA 11. CASO DE USO ADMINISTRAR ESTADO DE LAS TAREAS.....	29
TABLA 12. CONSULTAR GRÁFICOS DEL ESTADO DE LAS TAREAS.....	32
TABLA 13. CASO DE USO CONSULTAR DATOS DEL PROYECTO	33
TABLA 14. CASO DE USO CERRAR PROYECTO.....	35
TABLA 15. CASO DE USO CONSULTAR REPORTE ESTADO DEL PROYECTO.	38
TABLA 16. CASO DE USO CONSULTAR REPORTE HISTORIAL DE PROYECTOS POR TUTOR	39
TABLA 17. CASO DE USO CONSULTAR REPORTE HISTORIAL PROYECTOS POR PROGRAMA.....	41
TABLA 18. CASO DE USO REVISAR AVANCES DE TAREAS DEL PROYECTO.	43
TABLA 19. CASO DE USO CONSULTAR TAREAS DE UN PROYECTO	45
TABLA 20. CASO DE USO ENVIAR AVANCES DE TAREAS DE UN PROYECTO	47
TABLA 21. CASO DE USO CONSULTAR ESTADO DE LOS AVANCES DE TAREAS DE UN PROYECTO	50
TABLA 22. CASO DE USO ENVIAR ALERTA DE ESTADOS DE LOS AVANCES DE TAREAS DE UN PROYECTO	51
TABLA 23. CASO DE USO CONSULTAR DIAGRAMA DE GANTT.....	53

ANEXO 1

1. Casos de Uso

En el siguiente documento se describen los casos de uso identificados para el software Project Tracker.

Los casos de uso documentan el comportamiento del sistema desde el punto de vista del usuario. El primer paso para realizar los casos de uso es identificar los actores que están asociados a una serie de roles, entendiendo como actor entidades que interactúan con el sistemas realizando actividades específicas, estos actores pueden ser personas, dispositivos de hardware u otros sistemas tales como un Webservice.

A continuación se describen los casos de uso más relevantes del sistema, en estos se describe el requerimiento asociado, los actores implicados, las condiciones previas y posteriores, las excepciones, los flujos principales y alternos.

Convenciones

- PT-(Numero identificador) = ID del caso de uso.
- RF-(Numero identificador) = ID del requerimiento.

1.1. Caso de uso Autenticar usuarios de la aplicación

Este caso de uso está asociado al requerimiento funcional RF-001 y describe la autenticación de usuarios en la aplicación, los actores implicados son administrador, tutor y estudiante, la descripción general del casos de uso indica que el sistema muestra al usuario un formulario de autenticación con los campos de usuario y clave, luego el actor ingresas los datos requeridos y le solicita al sistema la autenticación, en este instante el sistema valida que el usuario exista y que esta sea su clave, si la validación es exitosa el sistema envía al actor a la página de inicio, de ser errada la validación se muestra un mensaje de advertencia indicando el fallo de la autenticación, también el sistema debe ofrecer la recuperación de la contraseña ya sea por nombre de usuario o por su el correo electrónico, la información detallada del caso de uso se encuentra en la Tabla 1 y la gráfica asociada es la Ilustración 1.

Tabla 1. Autenticar usuarios de la aplicación

Identificador:	PT-001	Requerimiento asociado:	RF-001
Nombre:	Autenticar usuarios de la aplicación		
Alias:	Autenticar usuario		
Actores:	Administrador, Tutor y Estudiante.		
Precondiciones:	<ul style="list-style-type: none"> El actor debe estar creado en el sistema. 		
Poscondiciones:	<ul style="list-style-type: none"> El actor ha sido autenticado con éxito. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> El sistema muestra el formulario de autenticación al actor. El sistema muestra la opción de autenticarse o de recuperar su contraseña. El actor ingresa su usuario y clave. 			

4. El actor selección ingresar.
5. El sistema verifica la existencia de la dupla usuario y clave.
6. Si la autenticación es exitosa el sistema dirige al actor a una página de inicio asociado al rol de dicho actor.
7. El sistema debe validar que no exista un usuario con el mismo nombre y correo electrónico.
8. El actor selecciona guardar.
9. El sistema almacena la información.
10. El sistema confirma la transacción.
11. El actor sale del caso de uso.

Flujo alternos:

- A. El actor selecciona la opción recuperar la contraseña de usuario.
1. En el paso 2.
 2. El sistema muestra un formulario para recuperar la contraseña.
 3. El sistema muestra las opciones de recuperar contraseña por nombre de usuario o dirección de correo electrónico.
 4. El sistema debe validar que el usuario o dirección de correo electrónico exista.
 5. El sistema envía un correo a la dirección asociada con el usuario o directamente a la dirección ingresada por el actor.
 6. El sistema confirma el envío del correo.

Excepciones:

- A. Validación del usuario y su contraseña
1. En el paso 4 si usuario con su contraseña no son correctos el sistema muestra un mensaje de *"Autenticación Errada, inténtelo de nuevo"*.
 2. Se regresa la paso 3.
- B. Validar usuario o correo electrónico.
1. En el paso 2 en la opción de recuperar contraseña el sistema debe validar que el usuario o correo eléctrico, si no existen el sistema muestra un mensaje informando especificando que

no existen dichos datos.

2. Se sistema se mantiene en la opción de recuperar contraseñan.

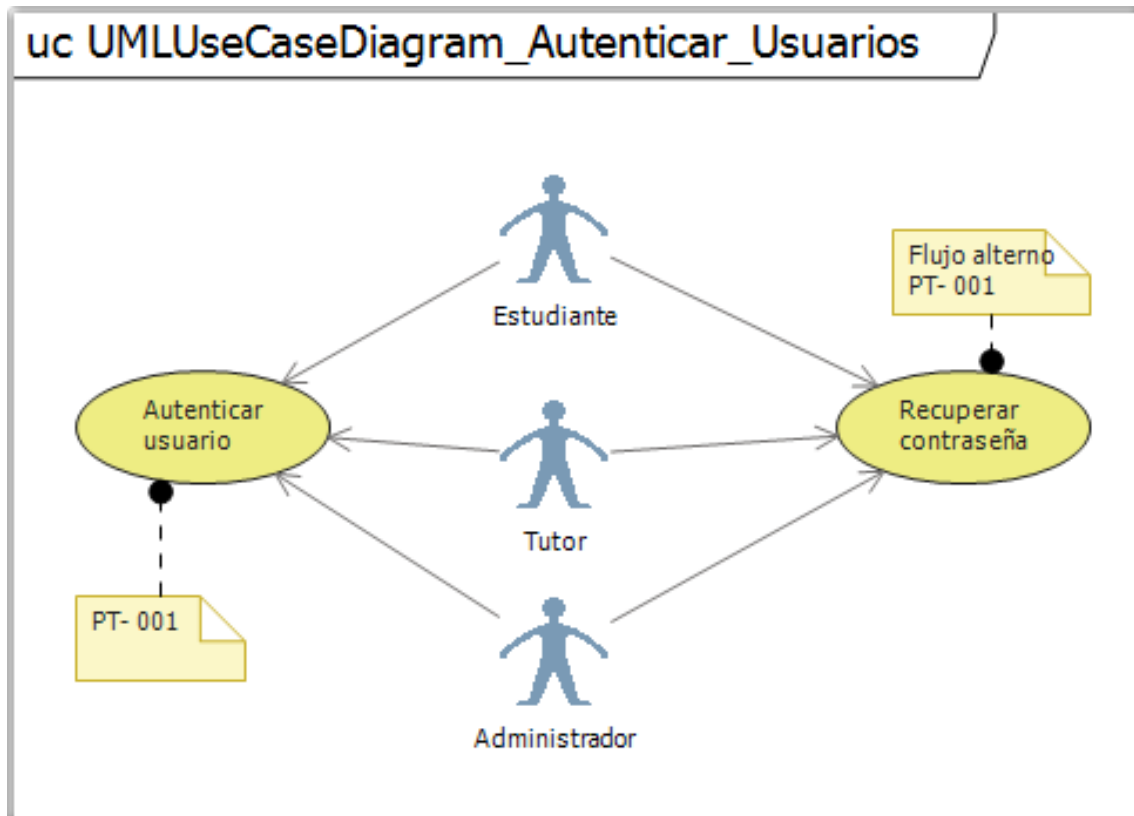


Ilustración 1. Caso de Uso PT-001

1.2. Caso de uso Administrar Usuarios

Este caso de uso está asociado al requerimiento funcional RF-002 y describe la creación y edición de usuario del sistema, el actor implicado es el administrador, la descripción general del casos de uso indica que el sistema muestra al usuario un formulario para ingresar los datos básico del usuario, como nombre, apellidos, usuario entre otros, dichos campos serán ingresados por el actor, además el sistema debe ofrecer la opción de creación y edición de usuarios, así mismo debe validar que el nombre de

usuario y dirección de correo sean únicos, la información detallada del caso de uso se encuentra en la Tabla 2 y la gráfica asociada es la Ilustración 2.

Tabla 2. Caso de uso Administrar Usuarios

Identificador:	PT-002	Requerimiento asociado:	RF-002
Nombre:	Administrar Usuarios.		
Actores:	Administrador		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • El actor debe estar creado en la aplicación con el rol de administrador. 		
Poscondiciones:	<ul style="list-style-type: none"> • El usuario ha sido creado en el sistema. • El usuario ha sido actualizado en el sistema. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear o actualizar usuarios. 2. El actor selecciona la opción de nuevo. 3. El sistema muestra los campos que van a estar asociados al usuario. 4. El sistema debe validar que no exista un usuario con el mismo nombre y correo electrónico. 5. El actor selecciona guardar. 6. El sistema almacena la información. 7. El sistema envía un correo electrónico al usuario creado con los datos de su cuenta 8. El sistema confirma la transacción. 9. El actor sale del caso de uso. 			
Flujo alternos:			

- A. El actor selecciona la opción editar usuario.
1. En el paso 1.
 2. El sistema muestra los datos que se pueden actualizar.
 3. El actor cambia los campos que requiera.
 4. El sistema debe validar que no exista un usuario con el mismo nombre o correo electrónico.
 5. El actor selecciona guardar.
 6. El sistema almacena los cambios.
 7. El actor sale del caso de uso.

Excepciones:

- A. Validación de campo obligatorios
1. En el paso 5 el sistema no guarda o actualiza la información si no existen los campos requeridos.
 2. El sistema muestra un mensaje de alerta especificando los campos obligatorios.
 3. Se regresa al paso 3.
- B. Validación de nombre de Usuario o correo electrónico.
1. En el paso 5
 2. El sistema debe validar que no exista un usuario con el mismo nombre de usuario o dirección de correo electrónico.
 3. El sistema muestra un mensaje de alerta especificando los campos que debe cambiar.
 4. Se regresa al paso 3.
- C. Envío de correo electrónico con los datos de la cuenta de usuario.
1. En el paso 7.
 2. El sistema envía un correo electrónico al usuario creado o editado con los datos de su cuenta.
 3. Si el sistema no puede enviar el correo electrónico, se muestra un mensaje de alerta especificando que no se puede enviar el correo electrónico.

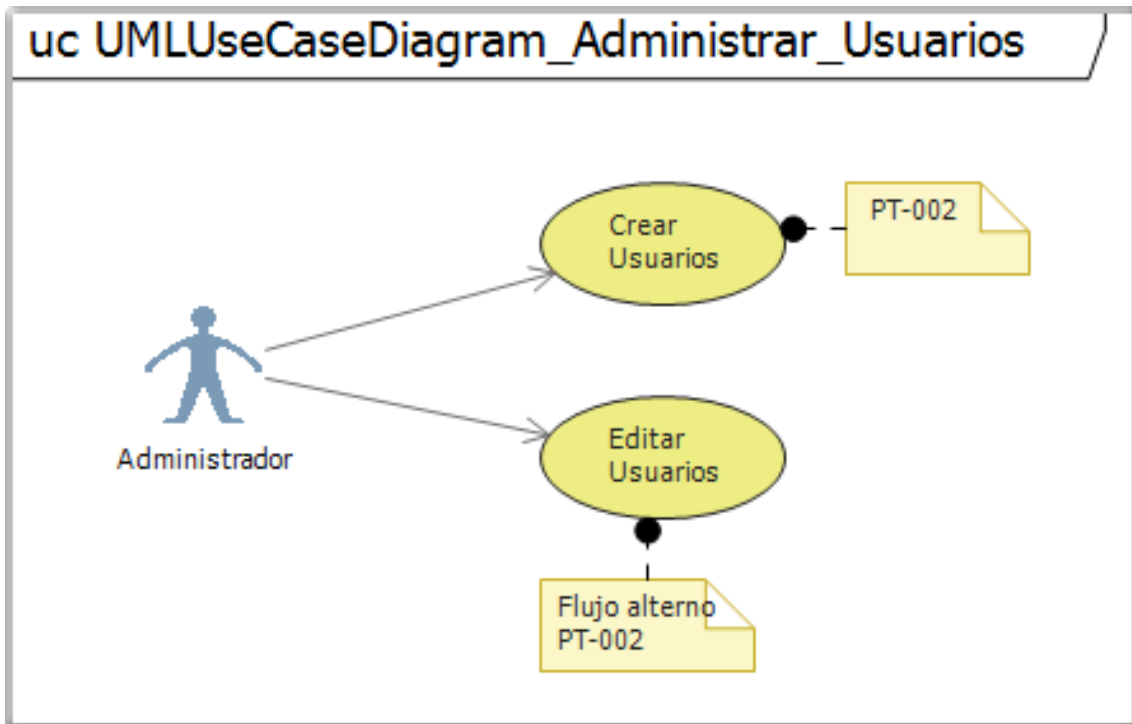


Ilustración 2. Caso de Uso PT-002

1.3. Caso de uso asignar roles de usuarios.

Este caso de uso está asociado al requerimiento funcional RF-003 y describe la vinculación y desvinculación de roles a un usuario, el actor implicado es el administrador, la descripción general del casos de uso indica que el sistema muestra al usuario un formulario con los usuarios creados y sus roles asociados, así mismo el sistema da la opción de asociar o desasociar roles a un usuario determinado, la información detallada del caso de uso se encuentra en la Tabla 3 y la gráfica asociada es la Ilustración 3.

Tabla 3. Asignar Roles de Usuarios.

Identificador:	PT-003	Requerimiento asociado:	RF-003
Nombre:	Asignar roles de usuarios.		
Actores:	Administrador		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • El usuario debe estar creado para asignarle un rol. • Tipos de roles: Administrador, Tutor, Estudiante. 		
Poscondiciones:	<ul style="list-style-type: none"> • Al usuario ha sido asignado o desasignado a un rol. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para asignar o desasignar roles a un usuario. 2. El sistema muestra los usuarios existentes. 3. El actor selecciona usuario. 4. El sistema muestra las opciones de roles. 5. El actor asigna o desasigna los roles al usuario. 6. El actor selecciona guardar. 7. El sistema confirma la transacción. 8. El actor sale del caso de uso. 			
Flujo alternos:			
A. No aplica.			
Excepciones:			
A. No aplica.			

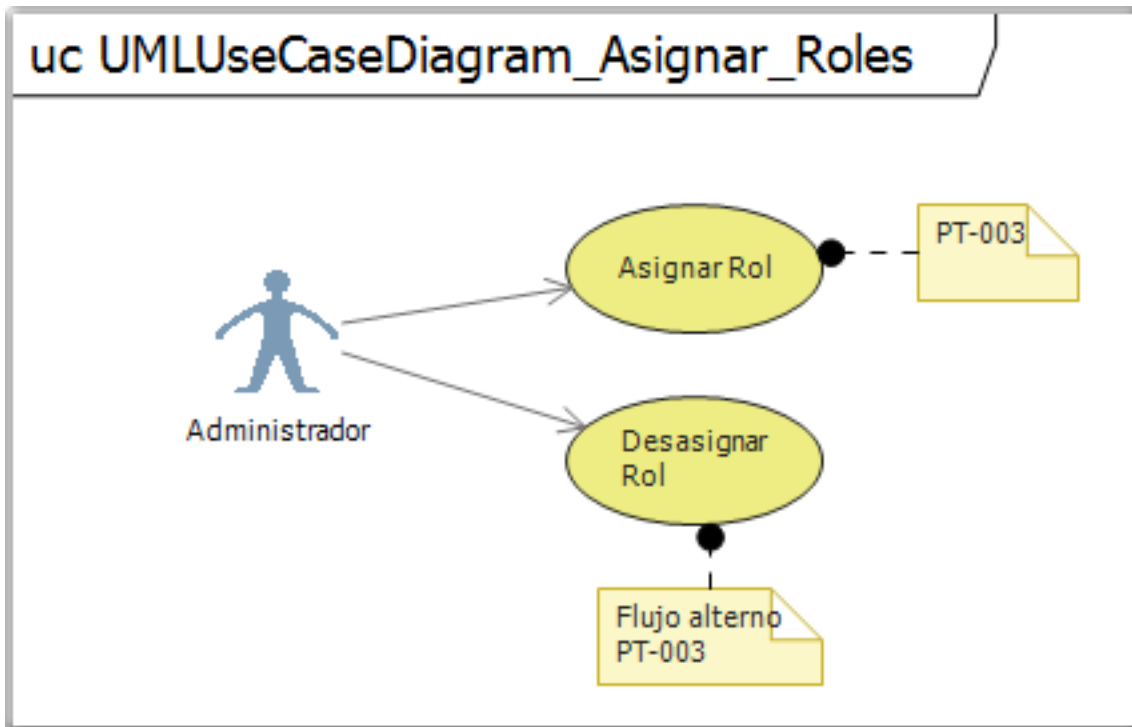


Ilustración 3. Caso de uso PT-003

1.4. Caso de uso Administrar facultad

Este caso de uso está asociado al requerimiento funcional RF-004 y describe la gestión y administración de facultades, el actor implicado es el administrador, la descripción general indica que el sistema ofrece un formulario el cual permite crear y editar las facultades, esencialmente el actor ingresa el nombre de la facultad y el sistema valida que este sea único, la información detallada del caso de uso se encuentra en la Tabla 4 y la gráfica asociada es la Ilustración 4.

Tabla 4. Caso de uso administrar facultad

Identificador:	PT-004	Requerimiento asociado:	RF-004
-----------------------	--------	--------------------------------	--------

Nombre:	Administrar facultad
Actores:	Administrador
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema.
Poscondiciones:	<ul style="list-style-type: none"> • Se ha creado una facultad. • Se han creado y asociado los programas a una facultad.
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear o actualizar la facultad. 2. El actor selecciona la opción de nuevo. 3. El sistema muestra los campos que van a estar asociados la facultad. 4. El sistema debe validar los campos obligatorios. 5. El sistema debe validar que no exista una facultad con el mismo nombre. 6. El actor selecciona guardar. 7. El sistema almacena la información. 8. El sistema confirma la transacción. 9. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. El actor selecciona la opción editar facultad.</p> <ol style="list-style-type: none"> 1. En el paso 1. 2. El sistema muestra los datos que se pueden actualizar. 3. El actor cambia los campos que requiera. 4. El sistema debe validar que no exista una facultad con el mismo nombre. 5. El actor selecciona guardar. 6. El sistema almacena los cambios. 7. El actor sale del caso de uso. 	
Excepciones:	

A. Validación de campo obligatorios facultad

1. En el paso 4 si el actor administra las facultades el sistema no guardar o actualiza la información si no existen los campos requeridos.
2. El sistema muestra un mensaje de alerta especificando los campos obligatorios.
3. Se regresa al paso 3.

B. Validación de nombre de la facultad.

1. En el paso 4 si el actor administra las facultades el sistema debe validar que no exista una facultad con el mismo nombre.
2. El sistema muestra un mensaje de alerta especificando el campo que debe cambiar.
3. Se regresa al paso 3.

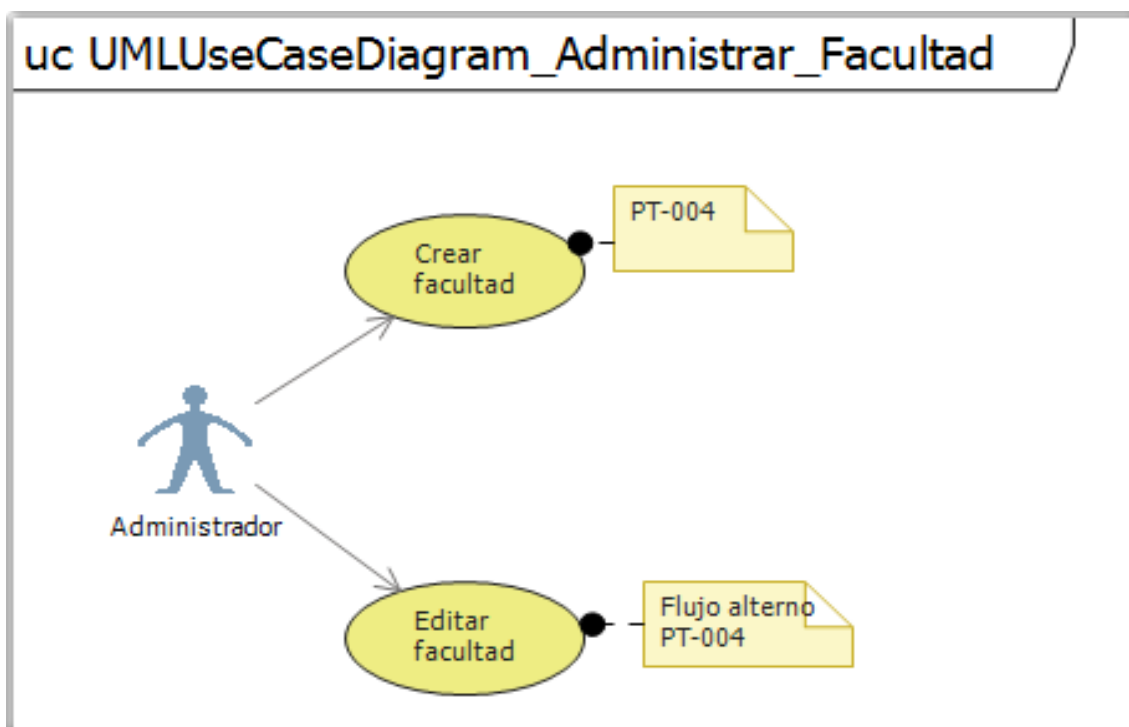


Ilustración 4. Caso de Uso PT-004

1.5. Caso de uso administrar programas

Este caso de uso está asociado al requerimiento funcional RF-004 y describe la gestión y administración de los programas de una facultad, el actor implicado es el administrador, la descripción general indica que el sistema ofrece un formulario el cual permite crear y editar los programas de una facultad, esencialmente el sistema muestra las facultades existente y el actor selecciona una de ellas, luego el actor ingresa el nombre del programa y el sistema valida que este sea único para esa facultad, la información detallada del caso de uso se encuentra en la Tabla 5 y la gráfica asociada es la Ilustración 5.

Tabla 5. Caso de uso Administrar programas

Identificador:	PT-005	Requerimiento asociado:	RF-004
Nombre:	Administrar programas		
Actores:	Administrador		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. 		
Poscondiciones:	<ul style="list-style-type: none"> • Se ha creado o actualiza un programa asociado a una facultad. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear o actualizar los programas de una facultad. 2. El actor selecciona la opción de nuevo. 3. El sistema muestra los campos que van a estar asociados al programa. 4. El sistema debe validar los campos obligatorios. 			

5. El sistema debe validar que no exista un programa con el mismo nombre para la facultad asociada.
6. El actor selecciona guardar.
7. El sistema almacena la información.
8. El sistema confirma la transacción.
9. El actor sale del caso de uso.

Flujo alternos:

- A. El actor selecciona la opción editar programa.
 1. En el paso 1.
 2. El sistema muestra los datos que se pueden actualizar.
 3. El actor cambia los campos que requiera.
 4. El sistema debe validar que no exista un programa con el mismo nombre para una facultad.
 5. El actor selecciona guardar.
 6. El sistema almacena los cambios.
 7. El actor sale del caso de uso.

Excepciones:

- A. Validación de campo obligatorios programa
 1. En el paso 4 el sistema no guarda o actualiza la información si no existen los campos requeridos.
 2. El sistema muestra un mensaje de alerta especificando los campos obligatorios.
 3. Se regresa al paso 3.

- B. Validación de nombre de la programa.
 1. En el paso 4 el sistema debe validar que no exista un programa con el mismo nombre.
 2. El sistema muestra un mensaje de alerta especificando el campo que debe cambiar.
 3. Se regresa al paso 3.

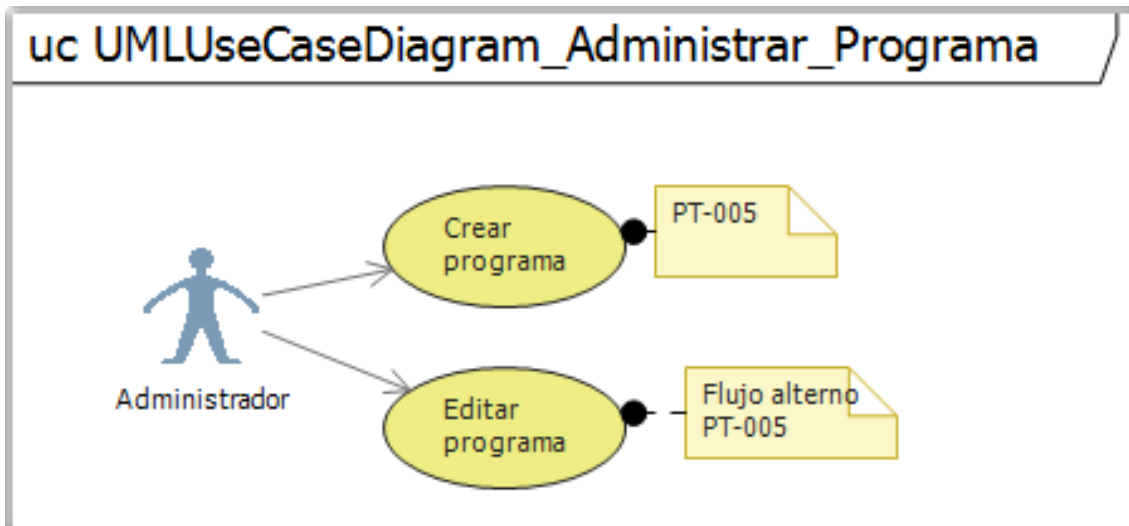


Ilustración 5. Caso de Uso PT-005

1.6. Caso de uso consultar log de errores

Este caso de uso está asociado al requerimiento funcional RF-005 y describe la consulta de errores controlados que ocurren en el sistema, el actor implicado es el administrador, básicamente el sistema muestra al actor la descripción detallada de los errores, la información detallada del caso de uso se encuentra en la Tabla 6 y la gráfica asociada es la Ilustración 6.

Tabla 6. Caso de uso consultar log de errores

Identificador:	PT-006	Requerimiento asociado:	RF-005
Nombre:	Consultar log de errores		
Actores:	Administrador		
Precondiciones:	<ul style="list-style-type: none"> El actor se ha autenticado en el sistema. 		
Poscondiciones:	<ul style="list-style-type: none"> Consulta de los errores del sistema. 		

Flujo normal de los eventos:
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar los errores que ha presentado el sistema filtrado por la fecha en que ocurrió el error y ordenado en forma descendente por dicha fecha. 2. El actor sale del caso de uso.
Flujo alternos:
B. No aplica.
Excepciones:
B. No aplica.

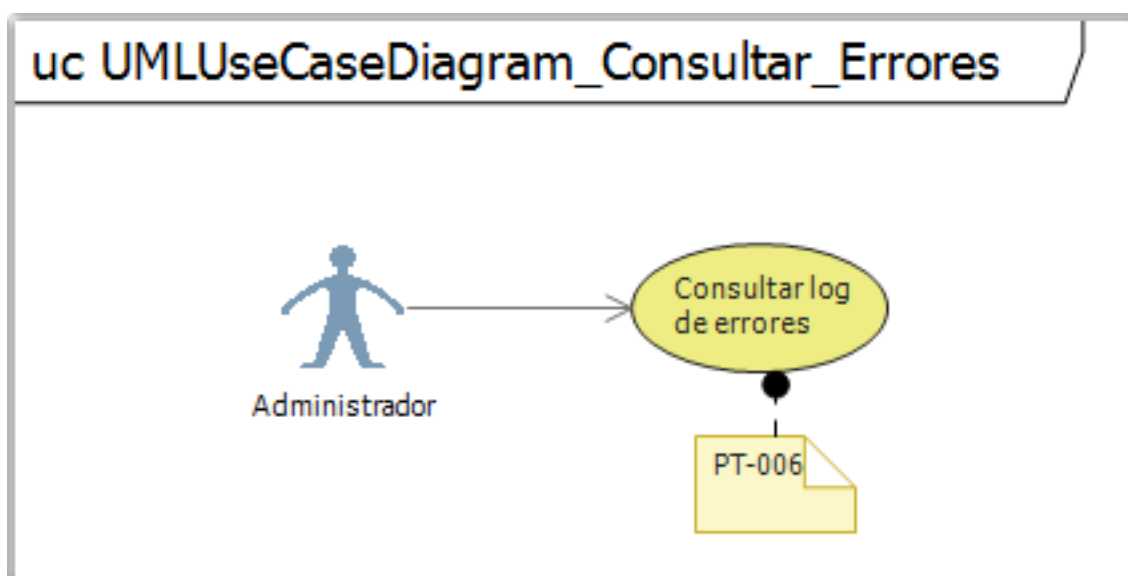


Ilustración 6. Caso de Uso PT-006

1.7. Caso de uso administrar proyectos

Este caso de uso está asociado al requerimiento funcional RF-006 y describe la gestión y administración de los proyectos, el actor implicado es el tutor, básicamente el

sistema muestra al actor un formulario para crear y editar los datos básicos del proyecto, como el nombre y descripción entre otros, también se muestra la opción de vincular un tutor al proyecto, básicamente el actor ingresa los datos básicos del proyecto y selecciona un tutor para ser vinculado al proyecto, la información detallada del caso de uso se encuentra en la Tabla 7 y la gráfica asociada es la Ilustración 7.

Tabla 7. Administrar proyectos

Identificador:	PT-007	Requerimiento asociado:	RF-006
Nombre:	Administrar Proyectos		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos usuarios con rol de tutor. • Deben estar creados algunos usuarios con rol de estudiante. 		
Poscondiciones:	<ul style="list-style-type: none"> • Proyecto creado o actualizado. • Tutor asociado a uno o varios proyectos. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear o actualizar los proyectos. 2. El actor selecciona la opción de nuevo. 3. El sistema muestra los campos que van a estar asociados al proyecto. 4. El sistema muestra la lista de tutores para asignar a un proyecto. 5. El actor selecciona un tutor de la lista. 6. El sistema debe validar los campos obligatorios. 7. El sistema debe validar que no exista un proyecto con el mismo nombre para la facultad asociada. 8. El sistema realizar el caso de uso (PT-08) 9. El actor selecciona guardar. 10. El sistema almacena la información. 			

<p>11. El sistema confirma la transacción.</p> <p>12. El actor sale del caso de uso.</p>
<p>Flujo alternos:</p>
<p>A. El actor selecciona la opción editar proyecto.</p> <ol style="list-style-type: none"> 1. En el paso 1. 2. El sistema muestra los datos que se pueden actualizar. 3. El actor cambia los campos que requiera. 4. El sistema debe validar que no exista un proyecto con el mismo nombre para una facultad. 5. El actor selecciona guardar. 6. El sistema almacena los cambios. 7. El actor sale del caso de uso. <p>B. El sistema debe validar la asignación del proyecto.</p> <ol style="list-style-type: none"> 1. En el paso 5. 2. El sistema debe validar el rol del actor autenticado, si es administrador y tutor puede seleccionar cualquier tutor para asignarlo a un proyecto, pero si el rol es tutor el proyecto a crear o actualizar será asignado al actor autenticado.
<p>Excepciones:</p>
<p>A. Validación de campo obligatorios proyecto</p> <ol style="list-style-type: none"> 1. En el paso 4 el sistema no guardar o actualiza la información si no existen los campos requeridos. 2. El sistema muestra un mensaje de alerta especificando los campos obligatorios. 3. Se regresa al paso 3. <p>B. Validación de nombre de la proyecto.</p> <ol style="list-style-type: none"> 1. En el paso 4 el sistema debe validar que no exista un proyecto con el mismo nombre. 2. El sistema muestra un mensaje de alerta especificando el campo que debe cambiar. 3. Se regresa al paso 3.

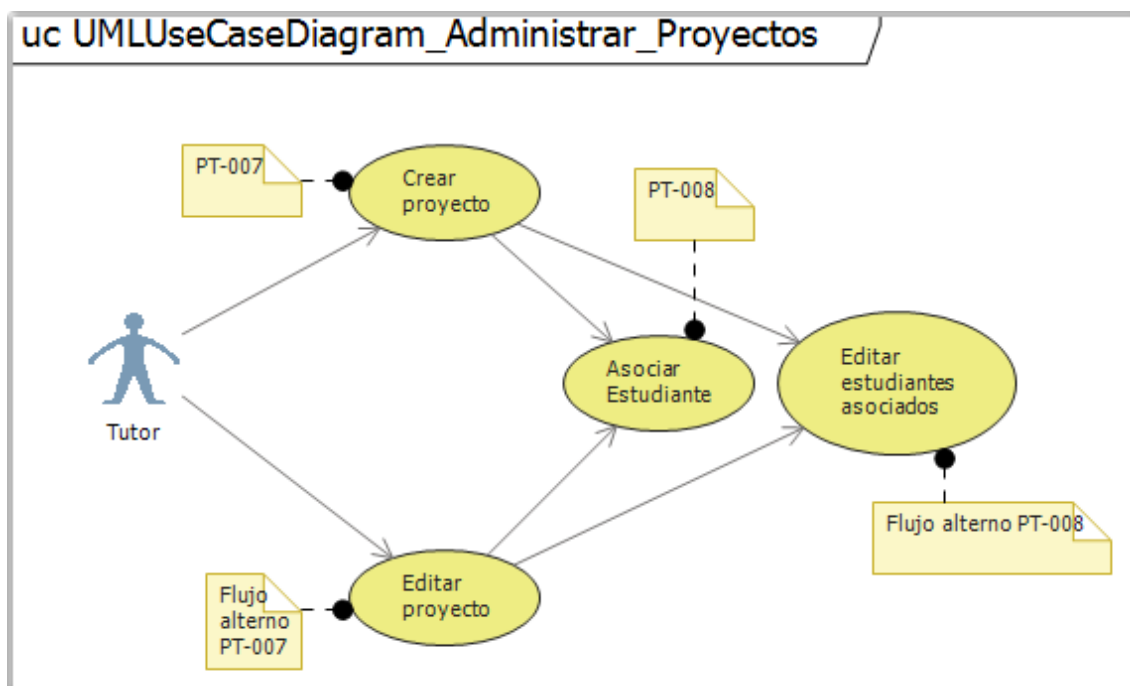


Ilustración 7. Casos de Uso PT- 007 y PT- 008

1.8. Caso de uso asociar estudiante a proyecto

Este caso de uso está asociado al requerimiento funcional RF-006 y describe la vinculación y la desvinculación de los integrantes al proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario con las listas de proyectos y estudiantes existentes, luego el actor selecciona un proyecto y los estudiantes para ser asociados a este proyecto, además el sistema ofrece la opción de editar la asociación de los integrantes del proyecto, la información detallada del caso de uso se encuentra en la Tabla 8 y la gráfica asociada es la Ilustración 7.

Tabla 8. Caso de uso asociar estudiante a proyecto

Identificador:	PT-008	Requerimiento asociado:	RF-006
Nombre:	Asociar estudiante a proyecto		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos usuarios con rol de estudiante. • Deben estar creados algunos proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Asociación de estudiantes a un proyecto 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para asociar los estudiantes a un proyecto. 2. El actor selecciona un proyecto. 3. El sistema muestra la lista de los estudiantes. 4. El actor selecciona un estudiante. 5. El sistema debe validar que el estudiante no este asociado a un proyecto. 6. El actor selecciona guardar. 7. El sistema almacena la información. 8. El sistema confirma la transacción. 9. El actor sale del caso de uso. 			
Flujo alternos:			
<ol style="list-style-type: none"> A. El actor selecciona la opción editar los estudiantes asociados al proyecto. <ol style="list-style-type: none"> 1. En el paso 5. 			

2. El sistema muestra los estudiantes ya asociados al proyecto para poder actualizar la asociación.
3. El actor selecciona el estudiante a asociar al proyecto o lo inactiva.
4. El sistema debe validar el estudiante no este asociado a otro proyecto o previamente ya este asociado al mismo proyecto.
5. El actor selecciona guardar.
6. El sistema almacena los cambios.
7. El actor sale del caso de uso.

Excepciones:

A. Validación de la asociación de estudiante al proyecto

1. En el paso 5.
2. El sistema debe validar la asociación.
3. El sistema muestra un mensaje de alerta especificando que ya está asociado el estudiante a un proyecto
4. Se regresa al paso 4.

1.9. Caso de uso administrar tareas

Este caso de uso está asociado al requerimiento funcional RF-007 y describe la gestión y administración de las tareas de los proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario para crear y editar los datos asociados a la tareas, como el título, descripción, proyecto, integrantes y fecha de ejecución entre otros, la información detallada del caso de uso se encuentra en la Tabla 9 y la gráfica asociada es la Ilustración 8.

Tabla 9. Caso de uso administrar tareas

Identificador:	PT-009	Requerimiento asociado:	RF-007
Nombre:	Administrar tareas		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyecto • Debe estar asociados los tutores a un proyecto. • Deben estar asociados los estudiantes a un proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Crear tarea 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear y editar las tareas de un proyecto 2. El sistema muestra una lista de los proyectos asociadas al rol de usuario. 3. El actor selecciona un proyecto. 4. El sistema muestra el tutor asociado. 5. El sistema muestra la lista de los estudiantes asociados al proyecto seleccionado. 6. El sistema realizar el caso de uso (PT-010) 7. El sistema muestra una lista de fecha inicial y fecha final del proyecto. 8. El actor selecciona la fecha inicial y final de la tarea 9. El sistema muestra campos adicionales como el título, descripción, comentario y archivos adjuntos de la tarea 10. El actor agrega los datos del título y la descripción que son obligatorios, opcionalmente los comentarios y los archivos adjuntos. 11. El sistema valida los campos obligatorios. 12. El actor selecciona guardar. 13. El sistema almacena la información. 14. El sistema confirma la transacción. 15. El actor sale del caso de uso. 			

Flujo alternos:
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none">1. En el paso 2.2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador y tutor muestra todos los proyectos. <p>B. El actor selecciona la opción editar la tarea.</p> <ol style="list-style-type: none">1. En el paso 1.2. El sistema muestra los datos asociados a la tarea.3. Si se necesita editar los estudiantes ya asociados. El sistema realiza el caso de uso (PT-010).4. El actor cambia los campos adicionales que requiera.5. El actor selecciona guardar.6. El sistema almacena los cambios.7. El actor sale del caso de uso.
Excepciones:
<p>A. Validación de campo obligatorios</p> <ol style="list-style-type: none">1. En el paso 10.2. El sistema no guardar o actualiza la información si no existen los campos requeridos.3. El sistema muestra un mensaje de alerta especificando los campos obligatorios.4. Se regresa al paso 10. <p>B. Validación fechas</p> <ol style="list-style-type: none">1. En el paso 8.2. El sistema debe validar que la fecha final sea mayor que la fecha inicial.3. El sistema muestra un mensaje de alerta de la inconsistencia.4. Regresa al paso 8

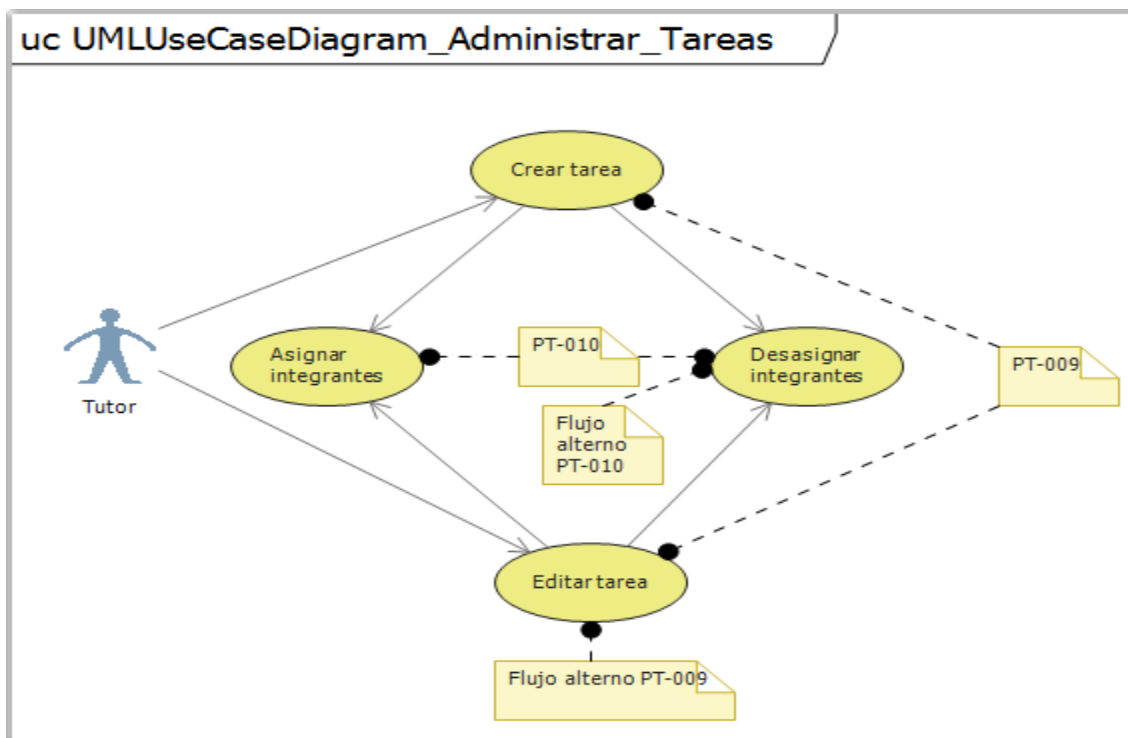


Ilustración 8. Casos de Uso PT-009 y PT-010

1.10. Caso de uso asociar estudiantes a tarea

Este caso de uso está asociado al requerimiento funcional RF-007 y describe la asociación de los estudiantes su proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario con los proyecto, sus tareas y los estudiantes asociados al proyecto, luego el actor selecciona los estudiantes y los vincula o desvincula de la tarea seleccionada, la información detallada del caso de uso se encuentra en la Tabla 10 y la gráfica asociada es la Ilustración 8.

Tabla 10. Caso de uso asociar estudiantes a tarea

Identificador:	PT-010	Requerimiento asociado:	RF-007
Nombre:	Asociar estudiante a tarea		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Se debe cumplir las precondiciones del caso de uso(PT-009) • Se realizar el evento 1, 2 ,3 y 5 del caso de uso(PT-009) 		
Poscondiciones:	<ul style="list-style-type: none"> • Asociación de estudiantes a una tarea 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El actor selecciona un estudiante. 2. El actor asigna o desasigna uno o varios estudiantes a la tarea. 3. El sistema debe validar que el estudiante no este asociado previamente a la tarea. 4. El actor selecciona guardar. 5. El sistema almacena la información. 6. El sistema confirma la transacción. 7. El actor sale del caso de uso. 			
Flujo alternos:			
<p>A. El actor selecciona la opción de desasignar los estudiantes asociados a la tarea.</p> <ol style="list-style-type: none"> 1. En el paso 3. 2. El sistema muestra los estudiantes ya asociados al proyecto para poder actualizar la asociación. 3. El actor desasigna el estudiante de la tarea. 4. El actor selecciona guardar. 5. El sistema almacena los cambios. 6. El actor sale del caso de uso. 			

Excepciones:
<p>A. Validación de la asociación de estudiante al proyecto</p> <ol style="list-style-type: none"> 1. En el paso 5. 2. El sistema debe validar la asociación. 3. El sistema muestra un mensaje de alerta especificando que ya está asociado el estudiante a un proyecto 4. Se regresa al paso 4.

1.11. Caso de uso administrar estado de la tarea

Este caso de uso está asociado al requerimiento funcional RF-008 y describe la gestión y administración de los estados de las tareas de los proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario para asociar un estado y un porcentaje de avance a la tarea, la información detallada del caso de uso se encuentra en la Tabla 11 y la gráfica asociada es la Ilustración 9.

Tabla 11. Caso de uso administrar estado de las tareas

Identificador:	PT-011	Requerimiento asociado:	RF-008
Nombre:	Administrar estado de las tareas		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creadas las tareas de un proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Estado de la tarea creado o actualizado 		

Flujo normal de los eventos:

1. El sistema muestra el formulario para crear o editar el estado de la tarea.
2. El sistema muestra una lista de los proyectos.
3. El actor selecciona un proyecto.
4. El sistema una lista de la tareas asociadas al proyecto.
5. El actor selecciona una tarea.
6. El sistema una lista de los estados de la tarea "creada, en progreso, sin solución, invalida, duplicada y finalizada".
7. El actor selecciona un estado.
8. El sistema muestra una lista con los valores del porcentaje de avance entre 0% y 100%.
9. El actor selecciona un valor del porcentaje.
10. El actor selecciona guardar.
11. El sistema almacena la información.
12. El sistema confirma la transacción.
13. El actor sale del caso de uso.

Flujo alternos:

A. Visualización de proyectos

1. En el paso 2.
2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador y tutor muestra todos los proyectos.

B. El actor actualiza el estado de la tarea.

1. En el paso 7.
2. El actor selecciona un estado de la tarea y un valor del porcentaje.
3. El actor selecciona guardar.
4. El sistema almacena los cambios.
5. El actor sale del caso de uso.

Excepciones:

A. No aplica

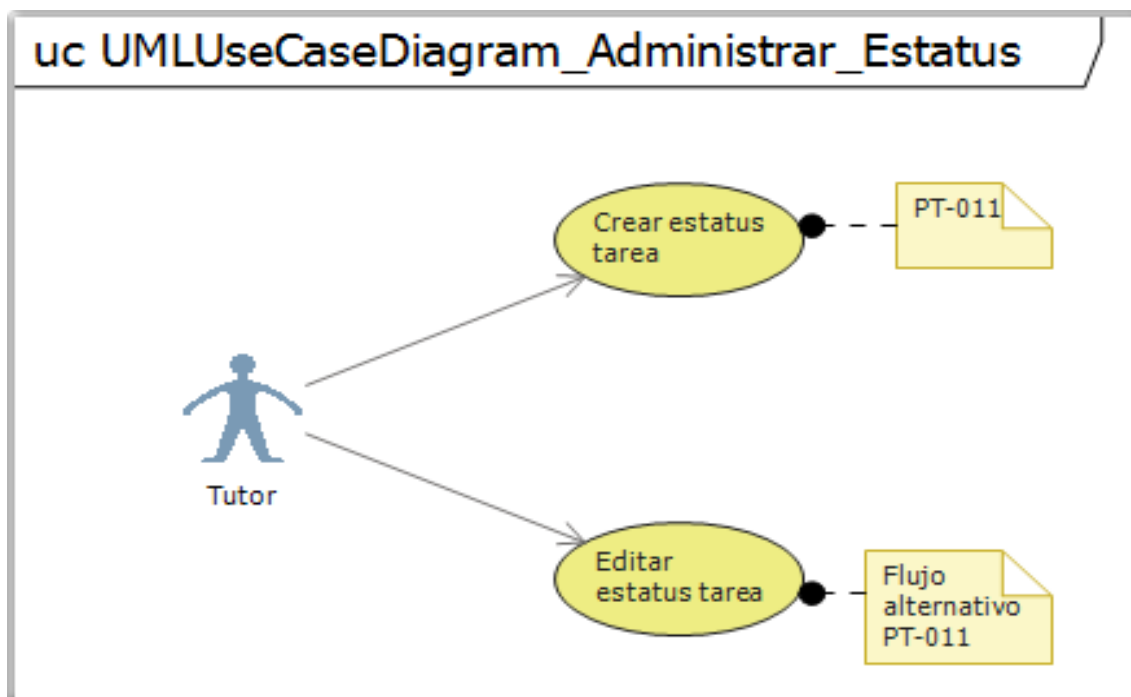


Ilustración 9. Caso de Uso PT-011

1.12. Caso de uso consultar gráficos del estado de las tareas

Este caso de uso está asociado al requerimiento funcional RF-009 y consiste en consultar el gráfico de barras horizontales que representa el porcentaje de avance de la tarea, el sistema muestra la lista de proyectos, luego el actor selecciona un proyecto y el sistema construye el gráfico, la información detallada del caso de uso se encuentra en la Tabla 12 y la gráfica asociada es la Ilustración 10.

Tabla 12. Consultar gráficos del estado de las tareas

Identificador:	PT-012	Requerimiento asociado:	RF-009
Nombre:	Consultar gráficos del estado de las tareas		
Actores:	Tutor y estudiante		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creadas las tareas de un proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los estado de las tareas de un proyecto en un gráfico de barras horizontales. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar los estado de las tareas. 2. El sistema muestra una lista de los proyectos. 3. El actor selecciona un proyecto. 4. El sistema muestra el gráfico asociado al estado de las tareas. 5. El actor selecciona una barra del gráfico. 6. El sistema muestra la descripción de estado de la tarea 7. El actor sale del caso de uso. 			
Flujo alternos:			
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 2. 2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador y tutor muestra todos los proyectos, si el rol es estudiante muestra el proyecto al que fue asociado. 			
Excepciones:			
B. No aplica			

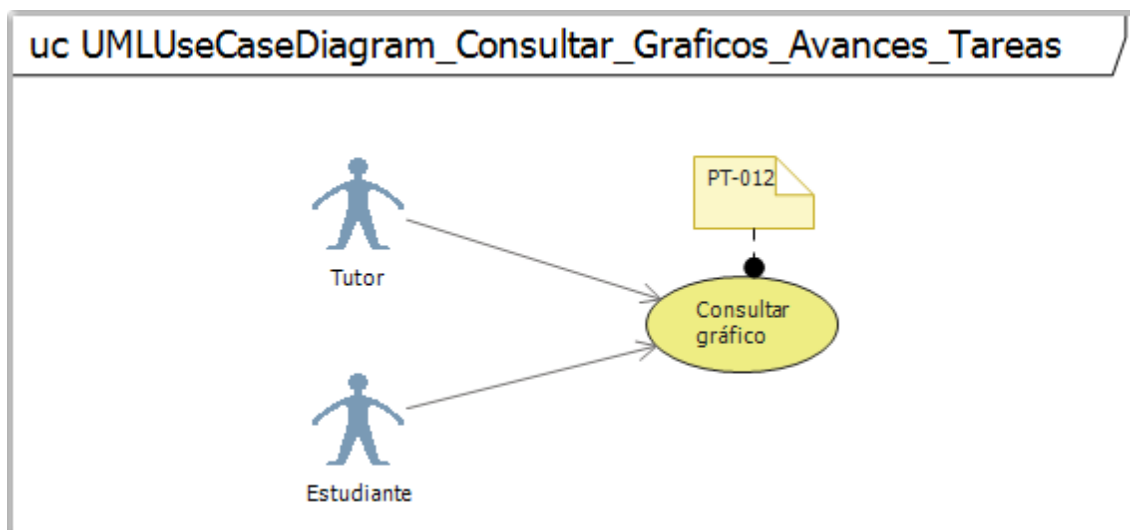


Ilustración 10. Caso de Uso PT - 012

1.13. Caso de uso consultar datos del proyecto

Este caso de uso está asociado al requerimiento funcional RF-010, para este caso el sistema muestra un formulario con los datos básicos del proyecto, actores implicados administrador, tutor y estudiante, la información detallada del caso de uso se encuentra en la Tabla 13 y la gráfica asociada es la Ilustración 11.

Tabla 13. Caso de uso consultar datos del proyecto

Identificador:	PT-013	Requerimiento asociado:	RF-0010
Nombre:	consultar datos del proyecto		
Actores:	Administrador, Tutor y estudiante		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto • Deben estar creadas las tareas de un proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los datos básicos del proyecto 		

Flujo normal de los eventos:
<ol style="list-style-type: none">1. El sistema muestra el formulario para visualizar los datos básicos del proyecto.2. El sistema muestra una lista de los proyectos.3. El actor selecciona un proyecto.4. El sistema muestra los datos básicos del proyecto.5. El actor sale del caso de uso.
Flujo alternos:
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none">1. En el paso 2.2. El sistema despliega la lista de proyectos asociados al usuario, si el rol es de tutor le muestra los proyectos que se le han asignado, si el rol del usuario es de administrador y tutor muestra todos los proyectos, si el actor tiene rol de estudiante se muestra solo el proyecto al que está asociado.
Excepciones:
<p>A. No aplica</p>

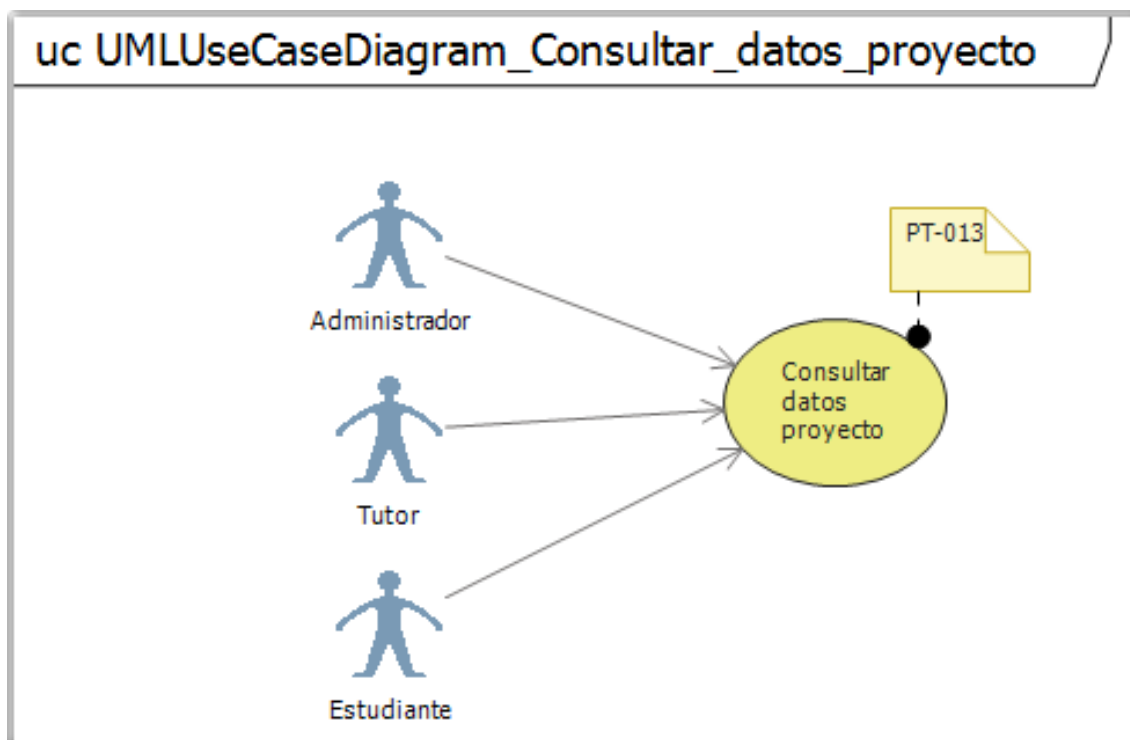


Ilustración 11. Caso de Uso PT-013

1.14. Caso de uso cerrar proyecto

Este caso de uso está asociado al requerimiento funcional RF-011, para este caso el sistema muestra un formulario con las lista de proyectos y ofrece la opción de cerrar o reabrir un proyecto y sus tareas asociadas, actores implicados administrador y tutor, la información detallada del caso de uso se encuentra en la Tabla 14 y la gráfica asociada es la Ilustración 12.

Tabla 14. Caso de uso cerrar proyecto

Identificador:	PT-014	Requerimiento asociado:	RF-011
Nombre:	Cerrar proyecto		

Actores:	Administrador y tutor
Precondiciones:	<ul style="list-style-type: none"> • Se debe cumplir las precondiciones del caso de uso(PT-002) • Debe estar creado el proyecto • Deben estar creadas las tareas de un proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los datos básicos del proyecto
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar los datos básicos del proyecto. 2. El sistema muestra una lista de los proyectos. 3. El actor selecciona un proyecto. 4. El sistema muestra la opción de cerrar el proyecto. 5. El actor selecciona cerrar o abrir el proyecto. 6. El actor selecciona guardar. 7. El sistema almacena la información. 8. El sistema confirma la transacción. 9. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 2. 2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador y tutor muestra todos los proyectos. <p>B. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 3. 2. El sistema muestra la opción de finalizar las tareas del proyecto al 100%. 3. El actor selecciona esta opción. 	

Excepciones:
A. No aplica

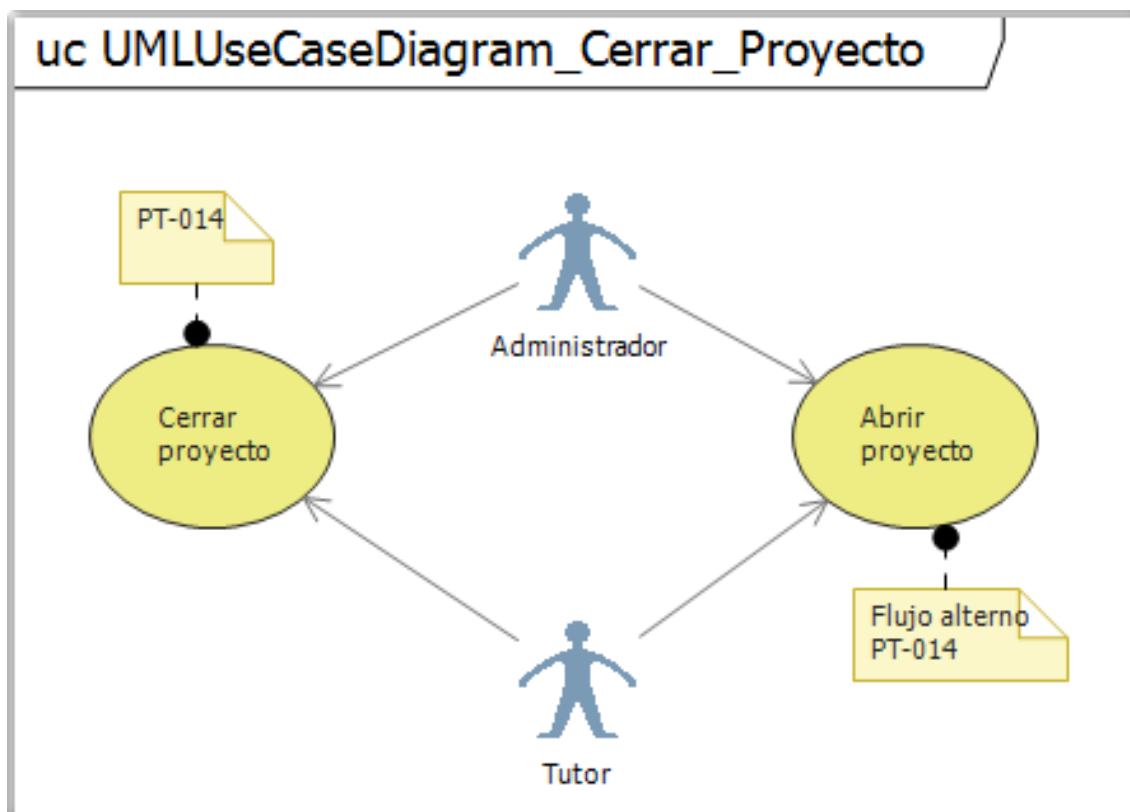


Ilustración 12. Caso de Uso PT-014

1.15. Caso de uso consultar reporte de estado del proyecto

Este caso de uso está asociado al requerimiento funcional RF-012, para este caso el sistema muestra un formulario con las lista de proyectos y ofrece la opción de generar un reporte de los estados del proyecto, actores implicados administrador, tutor y estudiantes, la información detallada del caso de uso se encuentra en la Tabla 15 y la gráfica asociada es la Ilustración 13.

Tabla 15. Caso de uso consultar reporte estado del proyecto.

Identificador:	PT-015	Requerimiento asociado:	RF-012
Nombre:	Consultar reporte estado del proyecto.		
Alias:	Consultar reporte estado proyecto.		
Actores:	Administrador, tutor y estudiante.		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyectos. • Deben estar creadas algunas tareas asociadas a un proyecto. 		
Poscondiciones:	<ul style="list-style-type: none"> • Reporte del estado de un proyecto en formato PDF 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para consultar los reportes asociados a un proyecto. 2. El sistema muestra una lista de los proyectos asociadas al rol de usuario. 3. El actor selecciona un proyecto. 4. El sistema muestra la opción de ver reporte. 5. El actor selecciona ver reporte. 6. El sistema genera el reporte en formato PDF. 7. El sistema confirma la transacción. 8. El actor sale del caso de uso. 			
Flujo alternos:			
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 1. 2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador muestra todos los proyectos, si el actor tiene rol de estudiante se muestra solo el proyecto al que está asociado. 			

Excepciones:
A. No aplica.

1.16. Caso de uso consultar reporte historial de proyectos por tutor

Este caso de uso está asociado al requerimiento funcional RF-013, para este caso el sistema muestra un formulario con las lista de tutores y ofrece la opción de generar un reporte del historial de los proyectos asociados al tutor seleccionado, actores implicados administrador y tutor, la información detallada del caso de uso se encuentra en la Tabla 16 y la gráfica asociada es la Ilustración 13.

Tabla 16. Caso de uso consultar reporte historial de proyectos por tutor

Identificador:	PT-016	Requerimiento asociado:	RF-013
Nombre:	Consultar reporte historial de proyectos por tutor.		
Alias:	Consultar reporte historial tutor		
Actores:	Administrador y tutor.		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyectos. • Deben estar creadas algunas tareas asociadas a un proyecto. 		
Poscondiciones:	<ul style="list-style-type: none"> • Reporte del historial de proyectos de un tutor en formato PDF. 		
Flujo normal de los eventos:			

<ol style="list-style-type: none"> 1. El sistema muestra el formulario para consultar el reporte del historial proyectos asociados a un tutor. 2. El sistema muestra una lista de los tutores. 3. El actor selecciona un tutor. 4. El sistema muestra una lista de fecha inicial y fecha final para realizar el filtro de la fecha de creación de los proyectos asignados al tutor seleccionado. 5. El actor selecciona la fecha inicial y final. 6. El sistema muestra la opción de ver reporte. 7. El actor selecciona ver reporte. 8. El sistema genera el reporte en formato PDF. 9. El sistema confirma la transacción. 10. El actor sale del caso de uso.
Flujo alternos:
<ol style="list-style-type: none"> 1. Visualización de tutores <ol style="list-style-type: none"> 1.1. En el paso 1. 1.2. El sistema despliega la lista de tutores. 1.3. Si el rol del actor es tutor, la lista solo mostrara el mismo actor, si el rol del usuario es de administrador muestra todos los tutores.
Excepciones:
<ol style="list-style-type: none"> A. No aplica.

1.17. Caso de uso consultar reporte historial proyectos por programa

Este caso de uso está asociado al requerimiento funcional RF-013, para este caso el sistema muestra un formulario con las lista de los programas por facultad y ofrece la opción de generar un reporte del historial de los proyectos asociados a la facultad

seleccionada, actor implicado administrador, la información detallada del caso de uso se encuentra en la Tabla 17 y la gráfica asociada es la Ilustración 13.

Tabla 17. Caso de uso consultar reporte historial proyectos por programa

Identificador:	PT-017	Requerimiento asociado:	RF-013
Nombre:	Consultar reporte Historial proyectos por Programa.		
Alias:	Consultar reporte Historial programa		
Actores:	Administrador.		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyectos. • Deben estar creadas algunas tareas asociadas a un proyecto. 		
Poscondiciones:	<ul style="list-style-type: none"> • Reporte del historial de proyectos asociados a una facultad en formato PDF. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para consultar el reporte del historial proyectos asociados a un programa. 2. El sistema muestra una lista de las facultades. 3. El actor selecciona una facultad. 4. El sistema muestra una lista de los programas asociados a la facultad seleccionada por el actor. 5. El actor selecciona un programa. 6. El sistema muestra una lista de fecha inicial y fecha final para realizar el filtro de la fecha de creación de los proyectos asociados al programa. 7. El actor selecciona la fecha inicial y final. 8. El sistema muestra la opción de ver reporte. 9. El actor selecciona ver reporte. 10. El sistema genera el reporte en formato PDF. 			

<p>11. El sistema confirma la transacción.</p> <p>12. El actor sale del caso de uso.</p>
Flujo alternos:
A. No aplica.
Excepciones:
A. No aplica.

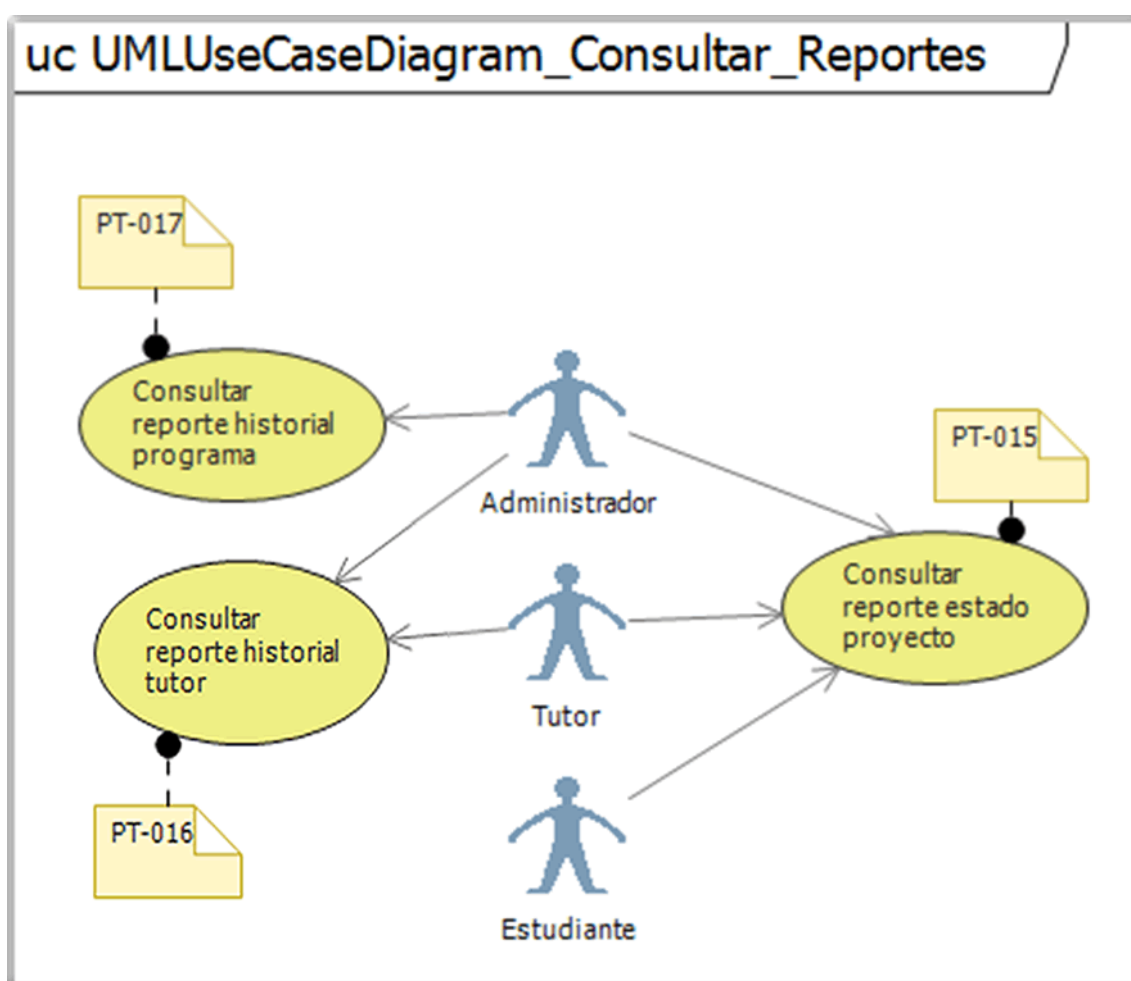


Ilustración 13. Casos de Uso PT-15, PT-16 y PT-17

1.18. Caso de uso revisar avances de tareas del proyecto.

Este caso de uso está asociado al requerimiento funcional RF-014, para este caso el sistema muestra un formulario con las lista de avances de las tareas de un proyecto, luego el actor tiene la opción de revisar los datos asociados al avance de la tarea y puede agregar una respuesta a dicho avance, el actor implicado es el tutor, la información detallada del caso de uso se encuentra en la Tabla 18 y la gráfica asociada es la Ilustración 14.

Tabla 18. Caso de uso revisar avances de tareas del proyecto.

Identificador:	PT-018	Requerimiento asociado:	RF-014
Nombre:	Revisar avances de tareas del proyecto.		
Alias:	Revisar avances.		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyectos. • Deben estar creadas algunas tareas asociadas a un proyecto. • Deben estar creados algunos avances asociados a una tarea. 		
Poscondiciones:	<ul style="list-style-type: none"> • Revisión del avance de las tareas de un proyecto y su correspondiente retroalimentación. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para revisar y enviar respuestas a los avances de las tareas de un proyecto. 			

2. El sistema muestra una lista de los proyectos.
3. El actor selecciona un proyecto.
4. El sistema muestra una lista de tareas asociadas al proyecto.
5. El actor selecciona una tarea.
6. El sistema muestra los avances asociados a una tarea seleccionada por el actor.
7. El actor selecciona un avance de una tarea.
8. El actor revisa el avance.
9. El actor ingresa una respuesta del avance.
10. El actor selecciona guardar.
11. El sistema guarda la respuesta.
12. El sistema actualiza el estado del avance.
13. El actor sale del caso de uso.

Flujo alternos:

A. Visualización de proyectos

1. En el paso 2.
2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador muestra todos los proyectos.

Excepciones:

A. Validación de campo obligatorios de la respuesta

1. En el paso 10.
2. Si el actor no agrega una respuesta.
3. El sistema muestra un mensaje de alerta especificando los campos obligatorios.
4. Se regresa al paso 9.

B. Adjuntar archivo de respuesta.

1. En el paso 9.
2. El sistema muestra la opción de adjuntar archivos.
3. El actor carga uno o varios archivos de las respuestas de los avances de la tarea.
4. El sistema continúa en el paso 10.

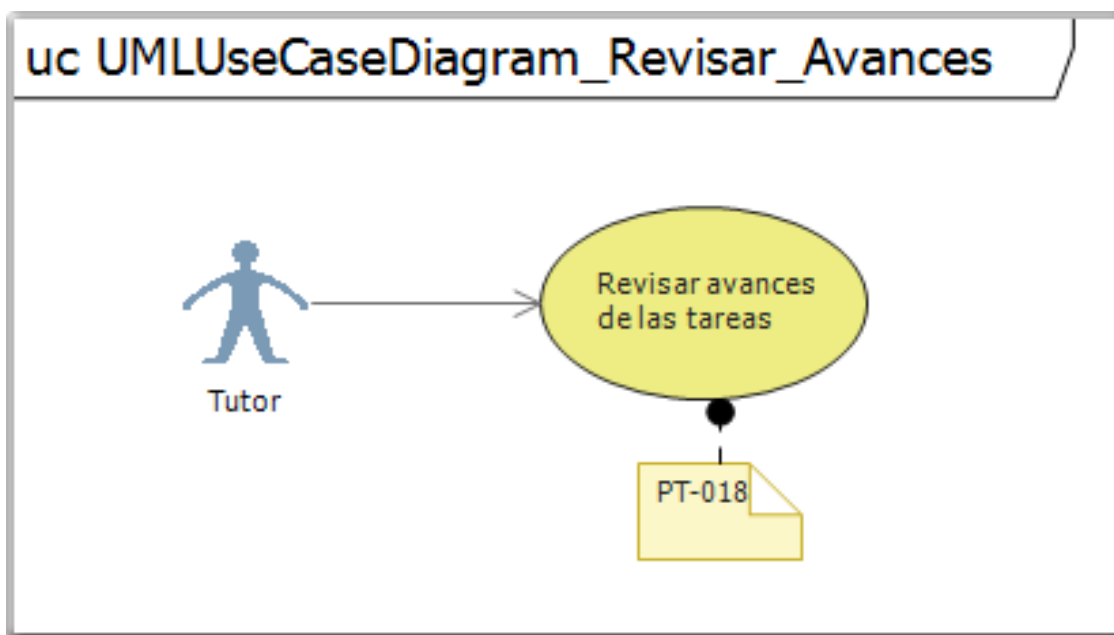


Ilustración 14. Caso de Uso PT-018

1.19. Caso de uso consultar tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-016, básicamente el sistema muestra al actor un formulario con la lista de tareas de un proyecto, el actor selecciona una tarea y el sistema le muestra una descripción detallada de los datos asociados de la tarea, tales como título, descripción, integrantes, etc. El actor implicado es el tutor, la información detallada del caso de uso se encuentra en la Tabla 19 y la gráfica asociada es la Ilustración 15.

Tabla 19. Caso de uso consultar tareas de un proyecto

Identificador:	PT-019	Requerimiento asociado:	RF-016
-----------------------	--------	--------------------------------	--------

Nombre:	Consultar tareas de un proyecto
Alias:	Consultar tareas
Actores:	Estudiante
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los datos de las tareas de un proyecto del actor
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar los datos de las tareas de un proyecto asociado al actor. 2. El sistema muestra la opción de visualizar las tareas de todos los integrantes o solo las tareas del actor. 3. El actor selecciona la opción de visualizar las tareas de todos los integrantes. 4. El sistema muestra una lista de las tareas con sus datos, el último estado y la diferencia de días entre la fecha del sistema y la fecha final de la tarea. 5. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. Visualización de tareas del actor</p> <ol style="list-style-type: none"> 1. En el paso 3. 2. El actor selecciona la opción de ver solo sus tareas. 3. El sistema continúa en el paso 4. 	
Excepciones:	
A. No aplica	

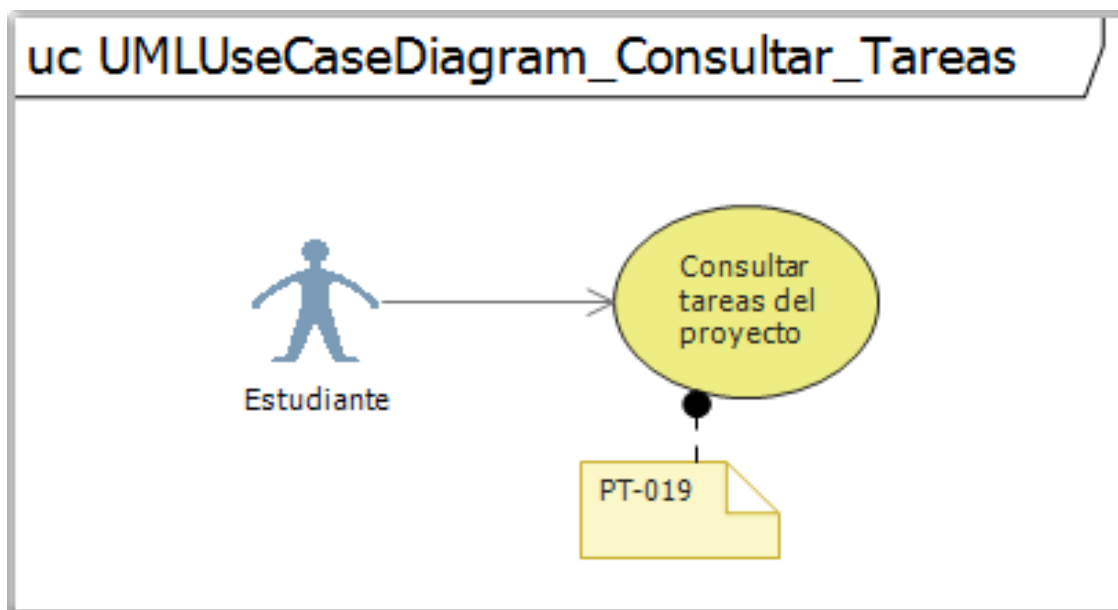


Ilustración 15. Caso de Uso PT-019

1.20. Caso de uso enviar avances de tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-017, el sistema ofrece un formulario con la lista de tareas de un proyecto asociado a un estudiante, luego el actor selecciona una tarea y adjunto los datos del avance de dicha tarea, el actor implicado es el estudiante, la información detallada del caso de uso se encuentra en la Tabla 19 y la gráfica asociada es la Ilustración 16.

Tabla 20. Caso de uso enviar avances de tareas de un proyecto

Identificador:	PT-020	Requerimiento asociado:	RF-017
Nombre:	Enviar avances de tareas de un proyecto		
Alias:	Enviar avances		

Actores:	Estudiante
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Los avances de las tareas han sido enviados
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para enviar los avances de las tareas del actor. 2. El sistema muestra la lista de las tareas asignadas al actor. 3. El actor selecciona una tarea. 4. El actor ingresa los datos del avance. 5. El actor selecciona guardar. 6. El sistema almacena la información. 7. El sistema confirma la transacción. 8. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. Validación de campo obligatorios</p> <ol style="list-style-type: none"> 1. En el paso 5. 2. El sistema no guardar o actualiza la información si no existen los campos requeridos. 3. El sistema muestra un mensaje de alerta especificando los campos obligatorios. 4. Se regresa al paso 5. <p>B. Adjuntar archivo del avance</p> <ol style="list-style-type: none"> 1. En el paso 4. 2. El sistema muestra la opción de enviar el avance del proyecto en un archivo adjunto. 3. El actor carga el archivo adjunto. 4. El sistema continúa en el paso 5. 	

Excepciones:
B. No aplica

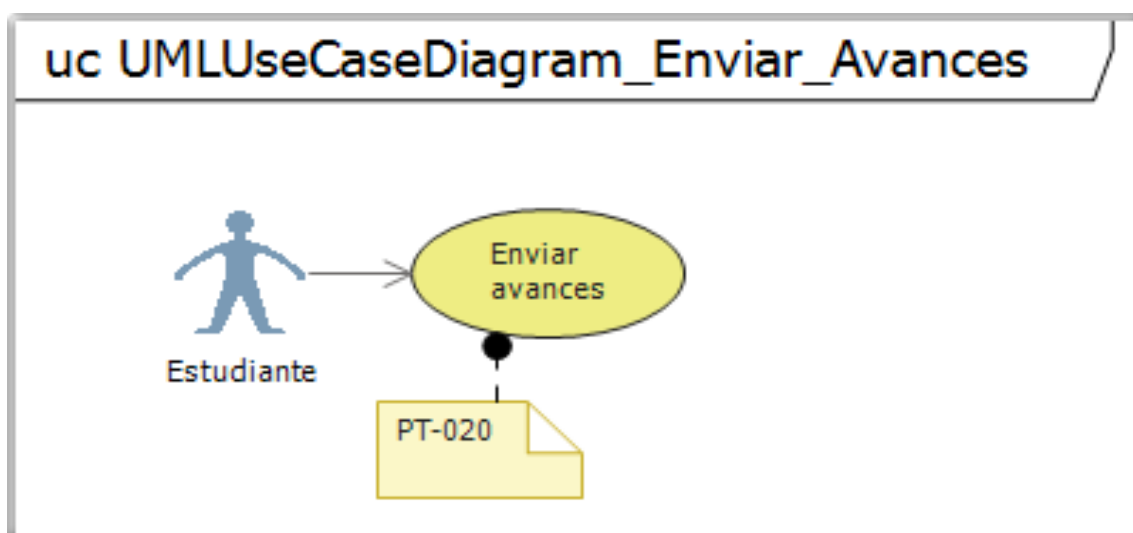


Ilustración 16. Caso de Uso PT-020

1.21. Caso de uso consultar estado de los avances de tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-018, básicamente el sistema muestra al actor un formulario con la lista de tareas de un proyecto, el actor selecciona una tarea y el sistema le muestra una descripción detallada del estado de los avances de la tarea y sus posibles repuestas. El actor implicado es el estudiante, la información detallada del caso de uso se encuentra en la Tabla 21 y la gráfica asociada es la Ilustración 15.

Tabla 21. Caso de uso consultar estado de los avances de tareas de un proyecto

Identificador:	PT-021	Requerimiento asociado:	RF-018
Nombre:	Consultar estado de los avances de tareas de un proyecto		
Alias:	Consultar estado del avance		
Actores:	Estudiante		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto • Deben estar creados los avances del proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los avances de las tareas del proyecto 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para consultar los avances de las tareas del proyecto. 2. El sistema muestra la lista de las tareas asignadas del proyecto. 3. El actor selecciona una tarea. 4. El sistema muestra la lista de los avances de la tarea seleccionada por el actor. 5. El sistema muestra la descripción de los avances con sus estados y sus posibles repuestas. 6. El actor sale del caso de uso. 			
Flujo alternos:			
A. No aplica.			
Excepciones:			
A. No aplica			

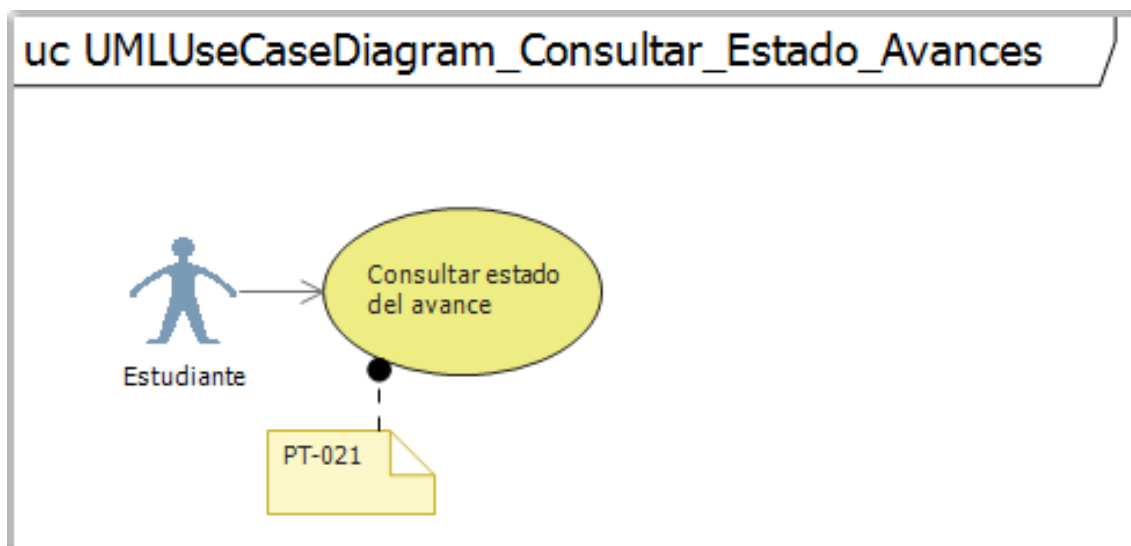


Ilustración 17. Caso de Uso PT-021

1.22. Caso de uso enviar alerta de estados de los avances de tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-019, básicamente el sistema muestra al actor un formulario para enviar las alertas de los estados de los avances de las tareas, si el actor selecciona enviar alertas, el sistema envía un correo electrónico a todos los integrantes y al tutor del proyecto con un reporte del estado de los avances de las tareas del proyecto. Los actores implicados son el tutor y el estudiante, la información detallada del caso de uso se encuentra en la Tabla 22 y la gráfica asociada es la Ilustración 18.

Tabla 22. Caso de uso enviar alerta de estados de los avances de tareas de un proyecto

Identificador:	PT-022	Requerimiento asociado:	RF-019
-----------------------	--------	--------------------------------	--------

Nombre:	Enviar alerta de estados de los avances de tareas de un proyecto
Alias:	Enviar alertas
Actores:	Tutor y Estudiante
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto • Deben estar creados los avances del proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Envío de correo de alertas de los avances de las tareas del proyecto
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para enviar los estados de las tareas del proyecto. 2. El sistema consulta los estados de las tareas del proyecto del actor. 3. El sistema envía un correo electrónico con el reporte de los estados de las tareas del proyecto a los integrantes y al tutor del proyecto del actor. 4. El actor sale del caso de uso. 	
Flujo alternos:	
A. No aplica.	
Excepciones:	
A. No aplica	

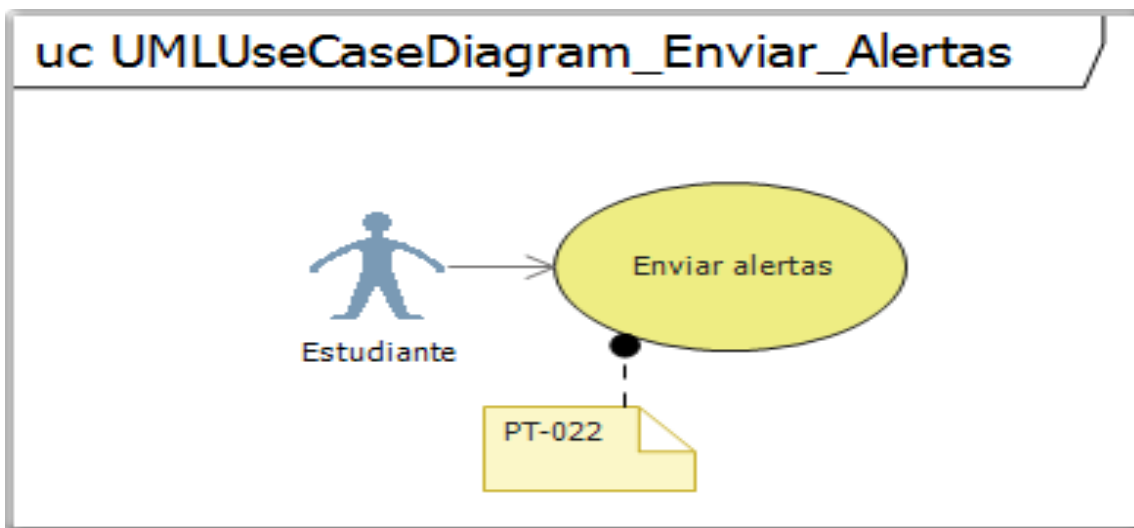


Ilustración 18.Caso de Uso PT-022

1.23. Caso de uso consultar diagrama de Gantt

Este caso de uso está asociado al requerimiento funcional RF-020 y consiste en consultar el gráfico o diagrama de Gantt de las tareas de un proyecto, básicamente el sistema grafica una línea de tiempo con las fechas de ejecución de las tareas, además el sistema ofrece la opción de visualizar las descripción detallada de cada tarea, la información detallada del caso de uso se encuentra en la Tabla 23 y la gráfica asociada es la Ilustración 19.

Tabla 23. Caso de uso consultar diagrama de Gantt

Identificador:	PT-023	Requerimiento asociado:	RF-020
Nombre:	Consultar diagrama de Gantt		

Alias:	Consultar Gantt
Actores:	Tutor y Estudiante
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar algunos proyectos • Deben estar creadas las tareas del proyecto • Deben estar creados los estados del proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Grafico del diagrama de Gantt
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar el diagrama de Gantt de un proyecto. 2. El sistema muestra las lista de proyectos. 3. El actor selecciona un proyecto. 4. El sistema genera un diagrama de Gantt del proyecto. 5. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 2. 2. El sistema despliega la lista de proyectos asociados al usuario, si el rol es de tutor le muestra los proyectos que se le han asignado, si el rol del usuario es de administrador y tutor muestra todos los proyectos, si el actor tiene rol de estudiante se muestra solo el proyecto al que está asociado. 	
Excepciones:	
A. No aplica	

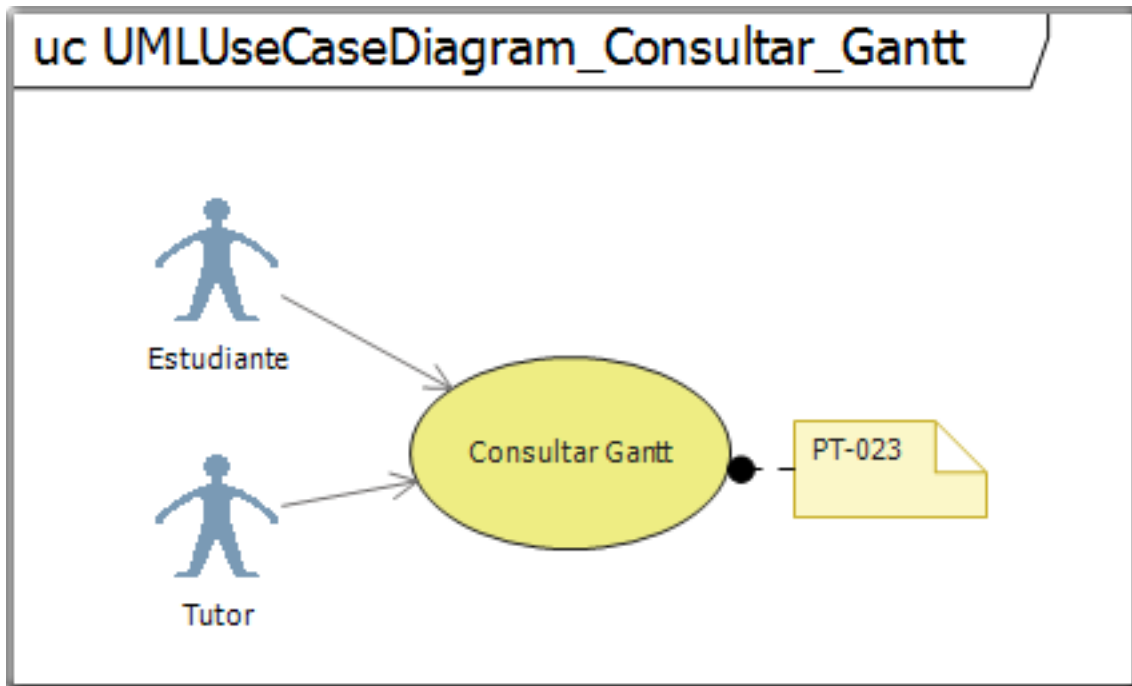


Ilustración 19. Caso de Uso PT-023

ANEXO 2

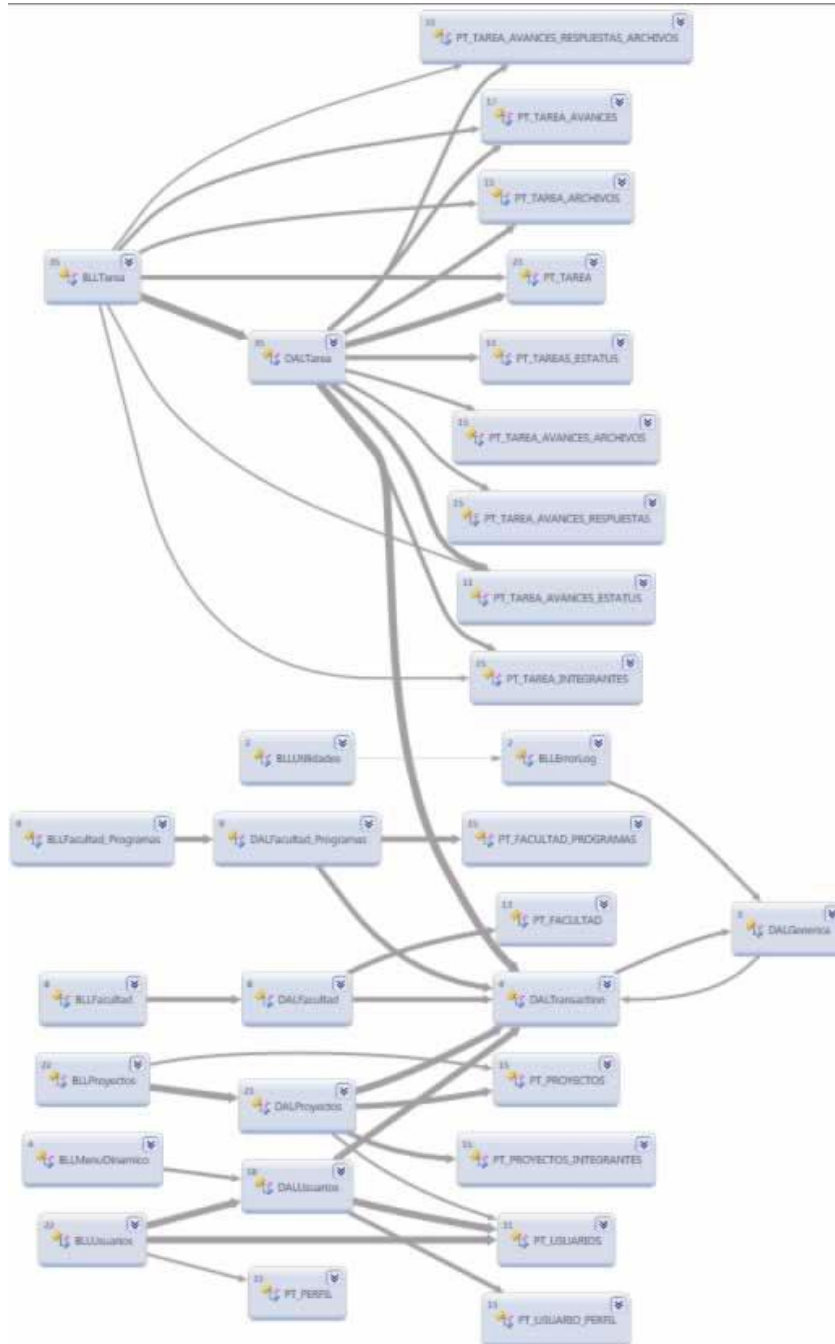


Ilustración 20. Diagrama de Clases

ANEXO 3

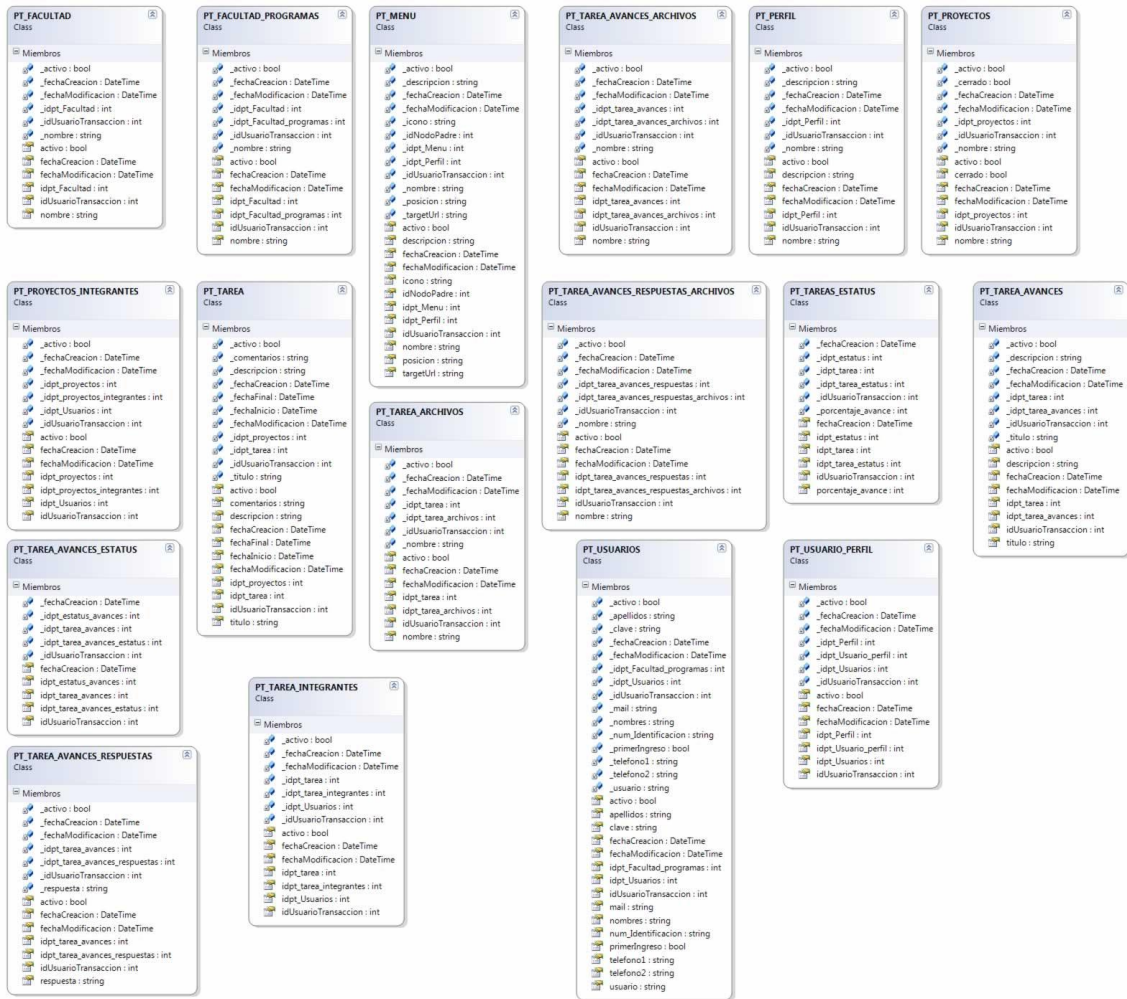


Ilustración 21. Capa EntityLayerProjectTracker

ANEXO 5

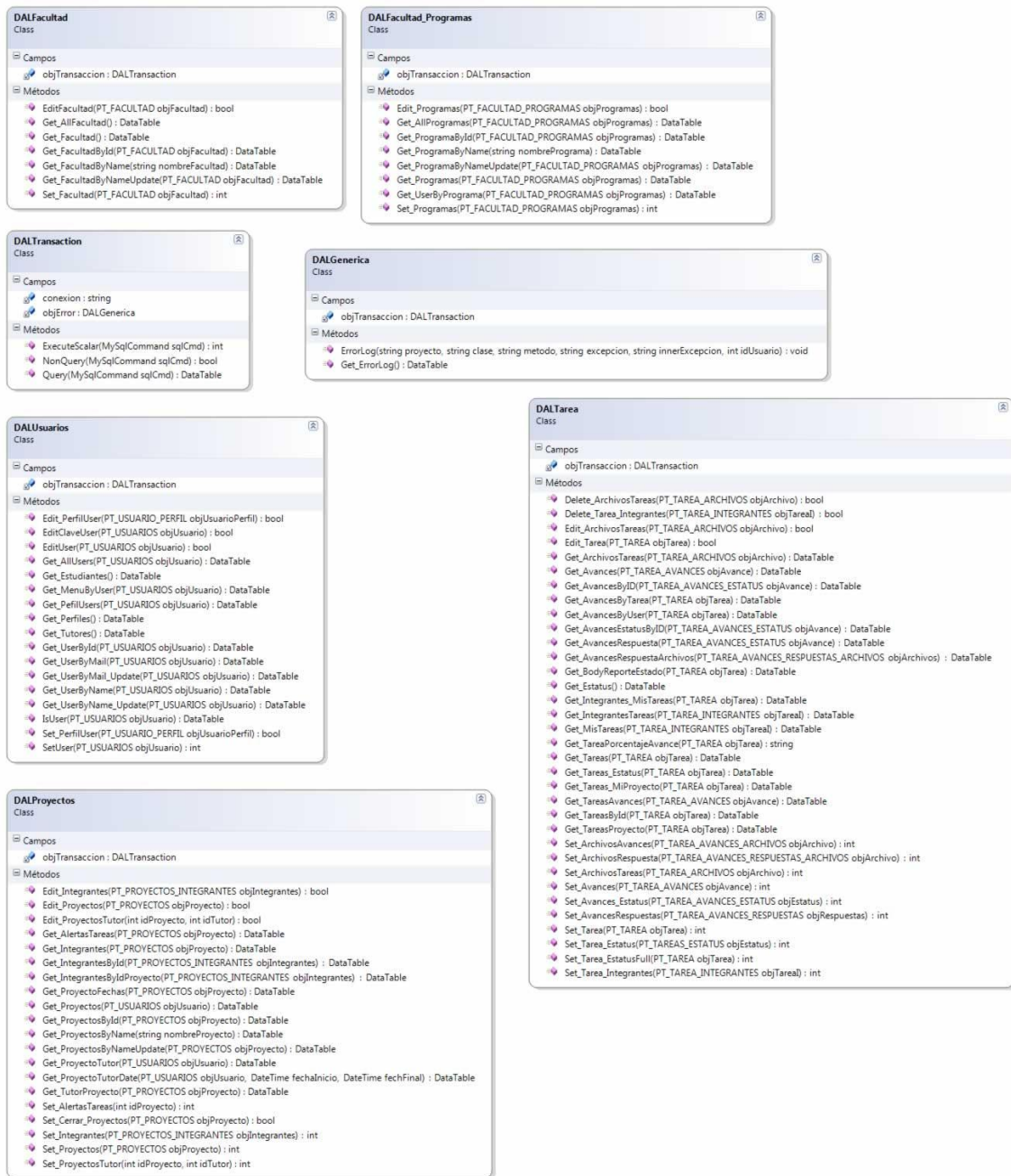


Ilustración 23. Capa DataAccessLayerProjectTracker

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Trabajo dirigido

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Tutor

Ingeniero Alexander García Pérez

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

TABLA DE ILUSTACIONES

Ilustración 1. Diccionario de datos tabla pt_alerta_tareas	4
Ilustración 2. Diccionario de datos tabla pt_error_log	5
Ilustración 3. Diccionario de datos tabla pt_estatus	5
Ilustración 4. Diccionario de datos tabla pt_estatus_avances.....	5
Ilustración 5. Diccionario de datos tabla pt_facultad	6
Ilustración 6. Diccionario de datos tabla pt_facultad_programas	6
Ilustración 7. Diccionario de datos tabla pt_menu	7
Ilustración 8. Diccionario de datos tabla pt_perfil	7
Ilustración 9. Diccionario de datos tabla pt_proyectos.....	8
Ilustración 10. Diccionario de datos tabla pt_proyectos_integrantes	8
Ilustración 11. Diccionario de datos tabla pt_proyectos_tutores.....	9
Ilustración 12. Diccionario de datos tabla pt_tarea	9
Ilustración 13. Diccionario de datos tabla pt_tarea_archivos.....	10
Ilustración 14. Diccionario de datos tabla pt_tarea_avances.....	10
Ilustración 15. Diccionario de datos tabla pt_tarea_avances_archivos	11
Ilustración 16. Diccionario de datos tabla pt_tarea_avances_estatus.....	11
Ilustración 17. Diccionario de datos tabla pt_tarea_avances_respuestas	12
Ilustración 18. Diccionario de datos tabla pt_tarea_avances_respuestas_archivos.....	12
Ilustración 19. Diccionario de datos tabla pt_tarea_estatus	13
Ilustración 20. Diccionario de datos tabla pt_tarea_integrantes.....	13
Ilustración 21. Diccionario de datos tabla pt_usuario_perfil.....	13
Ilustración 22. Diccionario de datos tabla pt_usuarios.....	14

ANEXO 7

1. DICCIONARIO DE DATOS

En el diccionario de datos se describen todas las tablas de la base de datos, dicha descripción contiene una breve explicación de los datos que debe almacenar la tabla, también contiene los nombre de los campos con sus tipos de datos, restricciones, valores por defecto, además indica si los campos son llaves primarias o foráneas, si el campo recibe datos nulos, si son únicos o si el campo se genera de forma auto numérica, en resumen el diccionario de dato contiene la descripción de la estructura de cada tabla de la base de datos.

1.1. Tabla pt alerta tareas

Esta tabla almacena el log de las alerta de los estados de las tareas de un proyecto que han sido enviadas por el software (Véase Ilustración 1).

pt_alerta_tareas										
Almacena el log de alertas de tareas de un proyecto										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Alerta_Tareas	INT(11)	✓	✓					✓		Id del alerta
fecha_Alerta	DATETIME		✓							Fecha en la que se envió el alerta
idpt_proyectos	INT(11)		✓							Id del proyecto que se le envió el alerta

Ilustración 1. Diccionario de datos tabla pt_alerta_tareas

1.2. Tabla pt error log

Esta tabla almacena los errores controlados de la aplicación web. (Véase Ilustración 2).

pt_error_log											
Almacenamos las excepciones de la aplicación											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_Error	INT(11)	✓	✓					✓		Id del error	
proyecto	VARCHAR(200)		✓							Nombre la capa de aplicación donde ocurrió el error	
clase	VARCHAR(200)		✓							Nombre de la clase donde ocurrió el error	
metodo	VARCHAR(200)		✓							Nombre del metodo donde ocurrió el error	
exception	VARCHAR(1000)		✓							Descripción de la excepción	
innerException	VARCHAR(1000)		✓							Descripción de la excepción interna	
idUsuario	INT(11)		✓							Id del usuario asociado el error	
fechaCreacion	DATETIME		✓							Fecha en la que se produjo el error	

Ilustración 2. Diccionario de datos tabla pt_error_log

1.3. Tabla pt estatus

Esta tabla almacena los tipos de estados que puede tener una tarea, estas pueden ser, creada, duplicada, en progreso, finalizada, invalida y sin solución. (Véase Ilustración 3).

pt_estatus											
Almacena los tipos de estatus de la tarea											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_estatus	INT(11)	✓	✓					✓		Id del estatus de la tarea	
nombre	VARCHAR(45)		✓							Nombre del estatus de la tarea	

Ilustración 3. Diccionario de datos tabla pt_estatus

1.4. Tabla pt estatus avances

Esta tabla almacena y guardan los tipos de estados que tiene un avance, los tipos son: enviado, visto y revisado. (Véase Ilustración 4).

pt_estatus_avances											
Almacena los tipos de estatus de Iso avances											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_estatus_avances	INT(11)	✓	✓					✓		Id del estatus del avance	
nombre	VARCHAR(100)		✓							Nombre del estatus de la tarea	

Ilustración 4. Diccionario de datos tabla pt_estatus_avances

1.5. Tabla pt facultad

Esta tabla almacena los datos de las facultades de la universidad EAN. (Véase Ilustración 5).

pt_facultad										
Almacena las facultades de la universidad EAN										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Facultad	INT(11)	✓	✓					✓		Id de la facultad
nombre	VARCHAR(100)		✓							Nombre de la facultad
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción

Ilustración 5. Diccionario de datos tabla pt_facultad

1.6. Tabla pt facultad

Esta tabla almacena los programas de las facultades de la universidad EAN. (Véase Ilustración 1).

pt_facultad_programas										
Almacena los programas de las facultades de la universidad EAN										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Facultad_programas	INT(11)	✓	✓					✓		Id del programa
idpt_Facultad	INT(11)		✓							Id de la facultad
nombre	VARCHAR(45)		✓							Nombre del programa
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción

Ilustración 6. Diccionario de datos tabla pt_facultad_programas

1.7. Tabla pt menú

Esta tabla almacena los datos asociados al menú que se mostrara en la interfaz gráfica del usuario estas son: las direcciones URL, el nombre de los enlaces, iconos, entre otros. (Véase Ilustración 7).

pt_menu										
Tabla para almacenar los datos para construir el menú dinámico										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Menu	INT(11)	✓	✓					✓		ID del menú
idpt_Perfil	INT(11)		✓							id del perfil del asociado
idNodoPadre	INT(11)		✓						'0'	Nodo padre, si es un nodo raíz este valor es cero
nombre	VARCHAR(30)		✓							Nombre o texto del nodo
descripcion	VARCHAR(200)									Breve descripción del nodo del menú
targetUrl	VARCHAR(1000)								NULL	Url a la que se debe redirigir cuando se dé click sobre este nodo del menú
icono	VARCHAR(100)								NULL	nombre de la imagen asociada a este nodo del menu, solo los nodos raiz tienen imagen
posicion	INT(11)		✓							Indica la posición u orden en que se va a ubicar el nodo en el menú
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizó la transacción

Ilustración 7. Diccionario de datos tabla pt_menu

1.8. Tabla pt perfil

Esta tabla almacena los tipos de perfiles que se usaran en la aplicación, estos son: estudiante, tutor y administrador. (Véase Ilustración 8).

pt_perfil										
Datos del perfil de usuario										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Perfil	INT(11)	✓	✓					✓		Id del perfil
nombre	VARCHAR(45)		✓							Nombre del perfil
descripcion	VARCHAR(200)									Descripción breve del perfil
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizó la transacción

Ilustración 8. Diccionario de datos tabla pt_perfil

1.9. Tabla pt proyectos

Esta tabla almacena los datos básicos de un proyecto como son: el título, la descripción, entre otros. (Véase Ilustración 9).

pt_proyectos											
Almacena los datos de los proyectos											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_proyectos	INT(11)	✓	✓					✓		Id del proyecto	
nombre	VARCHAR(500)		✓							Nombre del proyecto	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	
cerrado	BIT(1)		✓							Indica si el proyecto esta abierto 0 o cerrado 1.	

Ilustración 9. Diccionario de datos tabla pt_proyectos

1.10. Tabla pt proyectos integrantes

Esta tabla almacena los estudiantes que integran un proyecto. (Véase Ilustración 10).

pt_proyectos_integrantes											
Almacena los integrantes de un proyecto											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_proyectos_integrantes	INT(11)	✓	✓					✓		Id del registro	
idpt_proyectos	INT(11)		✓							Id del proyecto	
idpt_Usuarios	INT(11)		✓							Id del estudiante o usuario	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 10. Diccionario de datos tabla pt_proyectos_integrantes

1.11. Tabla pt proyectos tutores

Esta tabla almacena los proyectos que pertenecen a un tutor. (Véase Ilustración 11).

pt_proyectos_tutores										
Almacena la relación de los tutores y proyectos										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_proyectos_tutores	INT(11)	✓	✓							Id del proyecto
idpt_Usuarios	INT(11)	✓	✓							Id del tutor o usuario

Ilustración 11. Diccionario de datos tabla pt_proyectos_tutores

1.12. Tabla pt tarea

Esta tabla almacena todos los datos asociados a una tarea de un proyecto, los datos pueden ser nombre de la tarea, fecha de inicio y finalización de la misma. (Véase Ilustración 12).

pt_tarea										
Almacena las tareas de un proyecto										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_tarea	INT(11)	✓	✓					✓		Id de la tarea
titulo	VARCHAR(500)		✓							Título de la tarea
descripcion	VARCHAR(3000)		✓							Descripción de la tarea
comentarios	VARCHAR(500)									Comentarios de la tarea
idpt_proyectos	INT(11)		✓							Id del proyecto
fechaInicio	DATETIME		✓							Fecha de inicio de la tarea
fechaFinal	DATETIME		✓							Fecha final de la tarea
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizó la transacción

Ilustración 12. Diccionario de datos tabla pt_tarea

1.13. Tabla pt tarea archivos

Esta tabla almacena los posibles archivos adjuntos asociados a una tarea de un proyecto. (Véase Ilustración 13).

pt_tarea_archivos											
Almacena los datos de los archivos adjuntos de las tareas											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_archivos	INT(11)	✓	✓					✓		Id del registro	
idpt_tarea	INT(11)		✓							Id de la tarea	
nombre	VARCHAR(1000)		✓							Nombre del archivo	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 13. Diccionario de datos tabla pt_tarea_archivos

1.14. Tabla pt tarea avances

Esta tabla almacena los avances de las tareas de un proyecto. (Véase Ilustración 14).

pt_tarea_avances											
Almacena los avances de la tarea											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_avances	INT(11)	✓	✓					✓		Id del avance	
idpt_tarea	INT(11)		✓							Id de la tarea	
titulo	VARCHAR(500)		✓							Título del avance	
descripcion	VARCHAR(3000)		✓							Descripción del avance	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 14. Diccionario de datos tabla pt_tarea_avances

1.15. Tabla pt tarea avances archivos

Esta tabla almacena los datos adjuntos de la tarea de un proyecto. (Véase Ilustración 15).

pt_tarea_avances_archivos											
Almacena los datos de los archivos adjuntos de los avances de la tareas											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_avances_archivos	INT(11)	✓	✓					✓		Id del registro	
idpt_tarea_avances	INT(11)		✓							Id del avance	
nombre	VARCHAR(1000)		✓							Nombre del archivo	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 15. Diccionario de datos tabla pt_tarea_avances_archivos

1.16. Tabla pt tarea avances estatus

Esta tabla almacena los estados que ha tenido el avance durante el ciclo de la tarea.

(Véase Ilustración 16).

pt_tarea_avances_estatus											
Almacena el estatus de los avances de las tareas											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_avances_estatus	INT(11)	✓	✓					✓		Id del registro	
idpt_tarea_avances	INT(11)		✓							Id de la tarea	
idpt_estatus_avances	INT(11)		✓							Id del estatus del avance	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 16. Diccionario de datos tabla pt_tarea_avances_estatus

1.17. Tabla pt avances respuestas

Esta tabla guarda las respuestas de los avances de una tarea. (Véase Ilustración 17).

pt_tarea_avances_respuestas											
Almacena las respuestas de los avances de las tareas											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_avances_respuestas	INT(11)	✓	✓					✓		Id de la respuesta del avance	
idpt_tarea_avances	INT(11)		✓							Id del avance	
respuesta	VARCHAR(5000)		✓							Descripción de la respuesta	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 17. Diccionario de datos tabla pt_tarea_avances_respuestas

1.18. Tabla pt tarea avances respuestas archivos

Esta tabla almacena los datos adjuntos de los avances de las tareas de un proyecto.

(Véase Ilustración 18).

pt_tarea_avances_respuestas_archivos											
Almacena los datos de los archivos adjuntos de las respuestas de los avances											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_avances_respuestas_archivos	INT(11)	✓	✓					✓		Id del registro	
idpt_tarea_avances_respuestas	INT(11)		✓							Id de la respuesta	
nombre	VARCHAR(1000)		✓							Nombre del archivo	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	VARCHAR(45)		✓							Id del usuario que realizo la transacción	

Ilustración 18. Diccionario de datos tabla pt_tarea_avances_respuestas_archivos

1.19. Tabla pt tarea estatus

Esta tabla almacena todos los estados que puede tener una tarea durante el ciclo de vida del proyecto. (Véase Ilustración 19).

pt_tarea_estatus											
Almacena el estatus de la tarea y el porcentaje de avance.											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_estatus	INT(11)	✓	✓					✓		Id del registro	
idpt_tarea	INT(11)		✓							Id de la tarea	
idpt_estatus	INT(11)		✓							Id del estatus	
porcentaje_avance	INT(11)		✓						'0'	Porcentaje de avance de la tarea	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 19. Diccionario de datos tabla pt_tarea_estatus

1.20. Tabla pt tarea integrantes

Esta tabla almacena los integrantes o estudiantes asociados a las tareas del proyecto.

(Véase Ilustración 20).

pt_tarea_integrantes										
Almacena los integrantes de una tarea										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_tarea_integrantes	INT(11)	✓	✓					✓		Id del registro
idpt_tarea	INT(11)		✓							Id de la tarea
idpt_Usuarios	INT(11)		✓							Id del estudiante o usuario
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción

Ilustración 20. Diccionario de datos tabla pt_tarea_integrantes

1.21. Tabla pt usuario perfil

Esta tabla almacena los perfiles que puede tener el usuario en la aplicación. (Véase Ilustración 21).

pt_usuario_perfil										
Tabla para asociar el usuario a un perfil										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Usuario_perfil	INT(11)	✓	✓					✓		Id del registro auto numérico
idpt_Perfil	INT(11)		✓							id del perfil
idpt_Usuarios	INT(11)		✓							id del usuario al que se le asigna el perfil
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción

Ilustración 21. Diccionario de datos tabla pt_usuario_perfil

1.22. Tabla pt usuarios

Esta tabla almacena los datos de los usuarios de la aplicación como son nombres, apellidos, programa al que pertenece, nombre de usuario, claves, entre otros. (Véase Ilustración 22).

pt_usuarios										
Almacena los datos de los usuarios de la aplicación										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Usuarios	INT(11)	✓	✓					✓		id del Usuario
idpt_Facultad_programas	INT(11)		✓							Id del programa de una facultad a la que pertenece un estudiante
num_Identificacion	VARCHAR(45)		✓							Número de identificación
nombres	VARCHAR(100)		✓							Nombres del usuario
apellidos	VARCHAR(100)		✓							Apellidos del usuario
usuario	VARCHAR(45)		✓							Nombre de usuario de la aplicación
clave	VARCHAR(45)		✓							Clave de usuario
mail	VARCHAR(200)		✓							dirección de correo electrónico del usuario
telefono1	VARCHAR(45)		✓							Teléfono 1 del usuario
telefono2	VARCHAR(45)		✓							Teléfono 2 del usuario
primerIngreso	BIT(1)		✓						b'1'	Indica si el usuario ingresa a la aplicación por primera vez
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción

Ilustración 22. Diccionario de datos tabla pt_usuarios

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Trabajo dirigido

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Tutor

Ingeniero Alexander García Pérez

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

ANEXO 8

CONFIGURACIÓN PARA LA INSTALACIÓN DE LA APLICACIÓN

- **Procedimiento**

La aplicación debe ser publicada en un sistema operativo Windows, la máquina que va funcionar como Host debe tener instalado el Framework 4.0, el IIS 6.0 y MySQL como motor de base de datos, luego se debe restaurar un backup de la base de datos y debe estar creado por defecto un administrador de la aplicación.

Dentro de la aplicación web se debe modificar el archivo Webconfig y se debe ingresar los datos asociados a la cadena de conexión.

Para el envío de correos se debe crear una cuenta de correo, en el mismo archivo de Webconfig se debe agregar la dirección de correo, clave y servicio SMTP

Por parte de usuario final este debe tener instalado adobe Flash, Reader y los navegadores anteriormente citados.

A continuación se describe como crear el usuario administrador en directamente sobre la base de datos.

- Abrir la base de datos ProjectTracker y la tabla pt_usuarios
- Agregar un usuario con los siguientes datos.

idpt_Usuarios= Es auto numérico.

idpt_Facultad_programas = Id del programa de una facultad a la que pertenece un estudiante, para el caso de administrador se ingresa 17 el cual es un código genérico.

num_Identificacion= Número de identificación.

Nombres = Nombres del usuario.

Apellidos = Apellidos del usuario.

Usuario = Nombre de usuario de la aplicación.

Clave = Clave de usuario.

Mail = dirección de correo electrónico del usuario.

telefono1 = Teléfono 1 del usuario.

telefono2 = Teléfono 2 del usuario.

PrimerIngreso = 1.

Activo = 1.

FechaCreacion = Ingresar la fecha del día.

FechaModificacion = Ingresar la fecha del día.

IdUsuarioTransaccion = Ingresar 0.

- Después de agregar el usuario se debe agregar el perfil de administrador.

Para esto se debe abrir la tabla pt_usuario_perfil.

- Se debe agregar los siguientes datos:

idpt_Usuario_perfil = auto numérico.

pt_perfil = se debe ingresar el valor 1.

idpt_Usuarios = id del usuario que se creó en la tabla pt_usuarios.

activo = 1.

FechaCreacion = Ingresar la fecha del día.

FechaModificacion = Ingresar la fecha del día.

IdUsuarioTransaccion = Ingresar 0.

Luego del anterior procedimiento se podrá ingresar a la aplicación con rol de administrador.

LICENCIA DE USO – AUTORIZACIÓN DE LOS AUTORES

Actuando en nombre propio identificado (s) de la siguiente forma:

Nombre Completo Jose Ricardo Restrepo Escobar

Tipo de documento de identidad: C.C. T.I. C.E. Número: 00051283

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

El (Los) suscrito(s) en calidad de autor (es) del trabajo de tesis, monografía o trabajo de grado, documento de investigación, denominado:

Aplicación web para el control y seguimiento de avances del proyecto de grado "Project Tracker"

Dejo (dejamos) constancia que la obra contiene información confidencial, secreta o similar: SI NO
(Si marqué (marcamos) SI, en un documento adjunto explicaremos tal condición, para que la Universidad EAN mantenga restricción de acceso sobre la obra).

Por medio del presente escrito autorizo (autorizamos) a la Universidad EAN, a los usuarios de la Biblioteca de la Universidad EAN y a los usuarios de bases de datos y sitios webs con los cuales la Institución tenga convenio, a ejercer las siguientes atribuciones sobre la obra anteriormente mencionada:

- A. Conservación de los ejemplares en la Biblioteca de la Universidad EAN.
- B. Comunicación pública de la obra por cualquier medio, incluyendo Internet
- C. Reproducción bajo cualquier formato que se conozca actualmente o que se conozca en el futuro
- D. Que los ejemplares sean consultados en medio electrónico
- E. Inclusión en bases de datos o redes o sitios web con los cuales la Universidad EAN tenga convenio con las mismas facultades y limitaciones que se expresan en este documento
- F. Distribución y consulta de la obra a las entidades con las cuales la Universidad EAN tenga convenio

Con el debido respeto de los derechos patrimoniales y morales de la obra, la presente licencia se otorga a título gratuito, de conformidad con la normatividad vigente en la materia y teniendo en cuenta que la Universidad EAN busca difundir y promover la formación académica, la enseñanza y el espíritu investigativo y emprendedor.

Manifiesto (manifestamos) que la obra objeto de la presente autorización es original, el (los) suscritos es (son) el (los) autor (es) exclusivo (s), fue producto de mi (nuestro) ingenio y esfuerzo personal y la realizo (zamos) sin violar o usurpar derechos de autor de terceros, por lo tanto la obra es de exclusiva autoría y tengo (tenemos) la titularidad sobre la misma. En vista de lo expuesto, asumo (asumimos) la total responsabilidad sobre la elaboración, presentación y contenidos de la obra, eximiendo de cualquier responsabilidad a la Universidad EAN por estos aspectos.

En constancia suscribimos el presente documento en la ciudad de Bogotá D.C.,

NOMBRE COMPLETO: <u>José Iván Restrepo F</u>	NOMBRE COMPLETO: _____
FIRMA: <u>[Firma]</u>	FIRMA: _____
DOCUMENTO DE IDENTIDAD: <u>80051203</u>	DOCUMENTO DE IDENTIDAD: _____
FACULTAD: <u>Ingeniería</u>	FACULTAD: _____
PROGRAMA ACADÉMICO: <u>Sistemas</u>	PROGRAMA ACADÉMICO: _____

NOMBRE COMPLETO: _____	NOMBRE COMPLETO: _____
FIRMA: _____	FIRMA: _____
DOCUMENTO DE IDENTIDAD: _____	DOCUMENTO DE IDENTIDAD: _____
FACULTAD: _____	FACULTAD: _____
PROGRAMA ACADÉMICO: _____	PROGRAMA ACADÉMICO: _____

Fecha de firma: 12 diciembre 2012

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Trabajo dirigido

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Tutor

Ingeniero Alexander García Pérez

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

TABLA DE ILUSTRACIONES

ILUSTRACIÓN 1. CASO DE USO PT-001	8
ILUSTRACIÓN 2. CASO DE USO PT-002	11
ILUSTRACIÓN 3. CASO DE USO PT-003	13
ILUSTRACIÓN 4. CASO DE USO PT-004	15
ILUSTRACIÓN 5. CASO DE USO PT-005	18
ILUSTRACIÓN 6. CASO DE USO PT-006	19
ILUSTRACIÓN 7. CASOS DE USO PT- 007 Y PT- 008	22
ILUSTRACIÓN 8. CASOS DE USO PT-009 Y PT-010.....	27
ILUSTRACIÓN 9. CASO DE USO PT-011	31
ILUSTRACIÓN 10. CASO DE USO PT - 012.....	33
ILUSTRACIÓN 11. CASO DE USO PT-013	35
ILUSTRACIÓN 12. CASO DE USO PT-014	37
ILUSTRACIÓN 13. CASOS DE USO PT-15, PT-16 Y PT-17	42
ILUSTRACIÓN 14. CASO DE USO PT-018	45
ILUSTRACIÓN 15. CASO DE USO PT-019	47
ILUSTRACIÓN 16. CASO DE USO PT-020	49
ILUSTRACIÓN 17. CASO DE USO PT-021	51
ILUSTRACIÓN 18. CASO DE USO PT-022	53
ILUSTRACIÓN 19. CASO DE USO PT-023	55

ÍNDICE DE TABLAS

TABLA 1. AUTENTICAR USUARIOS DE LA APLICACIÓN.....	6
TABLA 2. CASO DE USO ADMINISTRAR USUARIOS	9
TABLA 3. ASIGNAR ROLES DE USUARIOS.....	12
TABLA 4. CASO DE USO ADMINISTRAR FACULTAD	13
TABLA 5. CASO DE USO ADMINISTRAR PROGRAMAS	16
TABLA 6. CASO DE USO CONSULTAR LOG DE ERRORES.....	18
TABLA 7. ADMINISTRAR PROYECTOS	20
TABLA 8. CASO DE USO ASOCIAR ESTUDIANTE A PROYECTO	23
TABLA 9. CASO DE USO ADMINISTRAR TAREAS	25
TABLA 10. CASO DE USO ASOCIAR ESTUDIANTES A TAREA	28
TABLA 11. CASO DE USO ADMINISTRAR ESTADO DE LAS TAREAS.....	29
TABLA 12. CONSULTAR GRÁFICOS DEL ESTADO DE LAS TAREAS.....	32
TABLA 13. CASO DE USO CONSULTAR DATOS DEL PROYECTO	33
TABLA 14. CASO DE USO CERRAR PROYECTO.....	35
TABLA 15. CASO DE USO CONSULTAR REPORTE ESTADO DEL PROYECTO.	38
TABLA 16. CASO DE USO CONSULTAR REPORTE HISTORIAL DE PROYECTOS POR TUTOR	39
TABLA 17. CASO DE USO CONSULTAR REPORTE HISTORIAL PROYECTOS POR PROGRAMA.....	41
TABLA 18. CASO DE USO REVISAR AVANCES DE TAREAS DEL PROYECTO.	43
TABLA 19. CASO DE USO CONSULTAR TAREAS DE UN PROYECTO	45
TABLA 20. CASO DE USO ENVIAR AVANCES DE TAREAS DE UN PROYECTO	47
TABLA 21. CASO DE USO CONSULTAR ESTADO DE LOS AVANCES DE TAREAS DE UN PROYECTO.....	50
TABLA 22. CASO DE USO ENVIAR ALERTA DE ESTADOS DE LOS AVANCES DE TAREAS DE UN PROYECTO	51
TABLA 23. CASO DE USO CONSULTAR DIAGRAMA DE GANTT.....	53

ANEXO 1

1. Casos de Uso

En el siguiente documento se describen los casos de uso identificados para el software Project Tracker.

Los casos de uso documentan el comportamiento del sistema desde el punto de vista del usuario. El primer paso para realizar los casos de uso es identificar los actores que están asociados a una serie de roles, entendiendo como actor entidades que interactúan con el sistemas realizando actividades específicas, estos actores pueden ser personas, dispositivos de hardware u otros sistemas tales como un Webservice.

A continuación se describen los casos de uso más relevantes del sistema, en estos se describe el requerimiento asociado, los actores implicados, las condiciones previas y posteriores, las excepciones, los flujos principales y alternos.

Convenciones

- PT-(Numero identificador) = ID del caso de uso.
- RF-(Numero identificador) = ID del requerimiento.

1.1. Caso de uso Autenticar usuarios de la aplicación

Este caso de uso está asociado al requerimiento funcional RF-001 y describe la autenticación de usuarios en la aplicación, los actores implicados son administrador, tutor y estudiante, la descripción general del casos de uso indica que el sistema muestra al usuario un formulario de autenticación con los campos de usuario y clave, luego el actor ingresas los datos requeridos y le solicita al sistema la autenticación, en este instante el sistema valida que el usuario exista y que esta sea su clave, si la validación es exitosa el sistema envía al actor a la página de inicio, de ser errada la validación se muestra un mensaje de advertencia indicando el fallo de la autenticación, también el sistema debe ofrecer la recuperación de la contraseña ya sea por nombre de usuario o por su el correo electrónico, la información detallada del caso de uso se encuentra en la Tabla 1 y la gráfica asociada es la Ilustración 1.

Tabla 1. Autenticar usuarios de la aplicación

Identificador:	PT-001	Requerimiento asociado:	RF-001
Nombre:	Autenticar usuarios de la aplicación		
Alias:	Autenticar usuario		
Actores:	Administrador, Tutor y Estudiante.		
Precondiciones:	<ul style="list-style-type: none"> El actor debe estar creado en el sistema. 		
Poscondiciones:	<ul style="list-style-type: none"> El actor ha sido autenticado con éxito. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> El sistema muestra el formulario de autenticación al actor. El sistema muestra la opción de autenticarse o de recuperar su contraseña. El actor ingresa su usuario y clave. 			

4. El actor selección ingresar.
5. El sistema verifica la existencia de la dupla usuario y clave.
6. Si la autenticación es exitosa el sistema dirige al actor a una página de inicio asociado al rol de dicho actor.
7. El sistema debe validar que no exista un usuario con el mismo nombre y correo electrónico.
8. El actor selecciona guardar.
9. El sistema almacena la información.
10. El sistema confirma la transacción.
11. El actor sale del caso de uso.

Flujo alternos:

- A. El actor selecciona la opción recuperar la contraseña de usuario.
1. En el paso 2.
 2. El sistema muestra un formulario para recuperar la contraseña.
 3. El sistema muestra las opciones de recuperar contraseña por nombre de usuario o dirección de correo electrónico.
 4. El sistema debe validar que el usuario o dirección de correo electrónico exista.
 5. El sistema envía un correo a la dirección asociada con el usuario o directamente a la dirección ingresada por el actor.
 6. El sistema confirma el envío del correo.

Excepciones:

- A. Validación del usuario y su contraseña
1. En el paso 4 si usuario con su contraseña no son correctos el sistema muestra un mensaje de *"Autenticación Errada, inténtelo de nuevo"*.
 2. Se regresa la paso 3.
- B. Validar usuario o correo electrónico.
1. En el paso 2 en la opción de recuperar contraseña el sistema debe validar que el usuario o correo eléctrico, si no existen el sistema muestra un mensaje informando especificando que

no existen dichos datos.

2. Se sistema se mantiene en la opción de recuperar contraseñan.

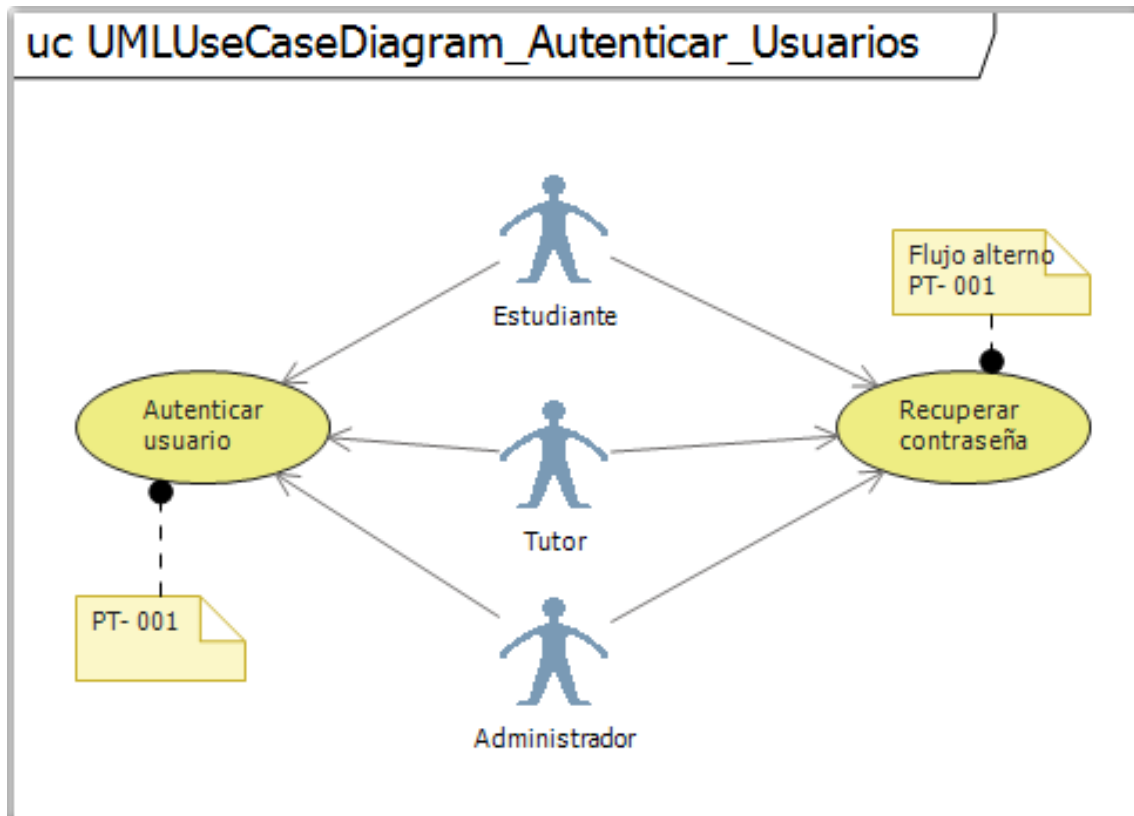


Ilustración 1. Caso de Uso PT-001

1.2. Caso de uso Administrar Usuarios

Este caso de uso está asociado al requerimiento funcional RF-002 y describe la creación y edición de usuario del sistema, el actor implicado es el administrador, la descripción general del casos de uso indica que el sistema muestra al usuario un formulario para ingresar los datos básico del usuario, como nombre, apellidos, usuario entre otros, dichos campos serán ingresados por el actor, además el sistema debe ofrecer la opción de creación y edición de usuarios, así mismo debe validar que el nombre de

usuario y dirección de correo sean únicos, la información detallada del caso de uso se encuentra en la Tabla 2 y la gráfica asociada es la Ilustración 2.

Tabla 2. Caso de uso Administrar Usuarios

Identificador:	PT-002	Requerimiento asociado:	RF-002
Nombre:	Administrar Usuarios.		
Actores:	Administrador		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • El actor debe estar creado en la aplicación con el rol de administrador. 		
Poscondiciones:	<ul style="list-style-type: none"> • El usuario ha sido creado en el sistema. • El usuario ha sido actualizado en el sistema. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear o actualizar usuarios. 2. El actor selecciona la opción de nuevo. 3. El sistema muestra los campos que van a estar asociados al usuario. 4. El sistema debe validar que no exista un usuario con el mismo nombre y correo electrónico. 5. El actor selecciona guardar. 6. El sistema almacena la información. 7. El sistema envía un correo electrónico al usuario creado con los datos de su cuenta 8. El sistema confirma la transacción. 9. El actor sale del caso de uso. 			
Flujo alternos:			

- A. El actor selecciona la opción editar usuario.
1. En el paso 1.
 2. El sistema muestra los datos que se pueden actualizar.
 3. El actor cambia los campos que requiera.
 4. El sistema debe validar que no exista un usuario con el mismo nombre o correo electrónico.
 5. El actor selecciona guardar.
 6. El sistema almacena los cambios.
 7. El actor sale del caso de uso.

Excepciones:

- A. Validación de campo obligatorios
1. En el paso 5 el sistema no guarda o actualiza la información si no existen los campos requeridos.
 2. El sistema muestra un mensaje de alerta especificando los campos obligatorios.
 3. Se regresa al paso 3.
- B. Validación de nombre de Usuario o correo electrónico.
1. En el paso 5
 2. El sistema debe validar que no exista un usuario con el mismo nombre de usuario o dirección de correo electrónico.
 3. El sistema muestra un mensaje de alerta especificando los campos que debe cambiar.
 4. Se regresa al paso 3.
- C. Envío de correo electrónico con los datos de la cuenta de usuario.
1. En el paso 7.
 2. El sistema envía un correo electrónico al usuario creado o editado con los datos de su cuenta.
 3. Si el sistema no puede enviar el correo electrónico, se muestra un mensaje de alerta especificando que no se puede enviar el correo electrónico.

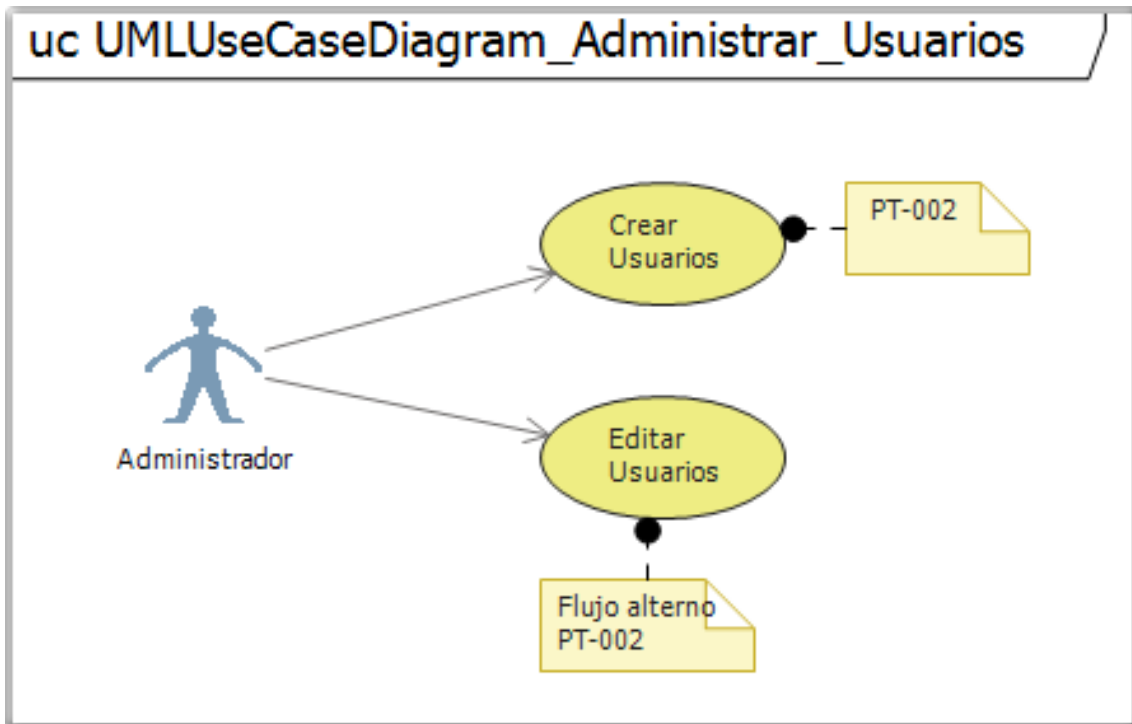


Ilustración 2. Caso de Uso PT-002

1.3. Caso de uso asignar roles de usuarios.

Este caso de uso está asociado al requerimiento funcional RF-003 y describe la vinculación y desvinculación de roles a un usuario, el actor implicado es el administrador, la descripción general del casos de uso indica que el sistema muestra al usuario un formulario con los usuarios creados y sus roles asociados, así mismo el sistema da la opción de asociar o desasociar roles a un usuario determinado, la información detallada del caso de uso se encuentra en la Tabla 3 y la gráfica asociada es la Ilustración 3.

Tabla 3. Asignar Roles de Usuarios.

Identificador:	PT-003	Requerimiento asociado:	RF-003
Nombre:	Asignar roles de usuarios.		
Actores:	Administrador		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • El usuario debe estar creado para asignarle un rol. • Tipos de roles: Administrador, Tutor, Estudiante. 		
Poscondiciones:	<ul style="list-style-type: none"> • Al usuario ha sido asignado o desasignado a un rol. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para asignar o desasignar roles a un usuario. 2. El sistema muestra los usuarios existentes. 3. El actor selecciona usuario. 4. El sistema muestra las opciones de roles. 5. El actor asigna o desasigna los roles al usuario. 6. El actor selecciona guardar. 7. El sistema confirma la transacción. 8. El actor sale del caso de uso. 			
Flujo alternos:			
A. No aplica.			
Excepciones:			
A. No aplica.			

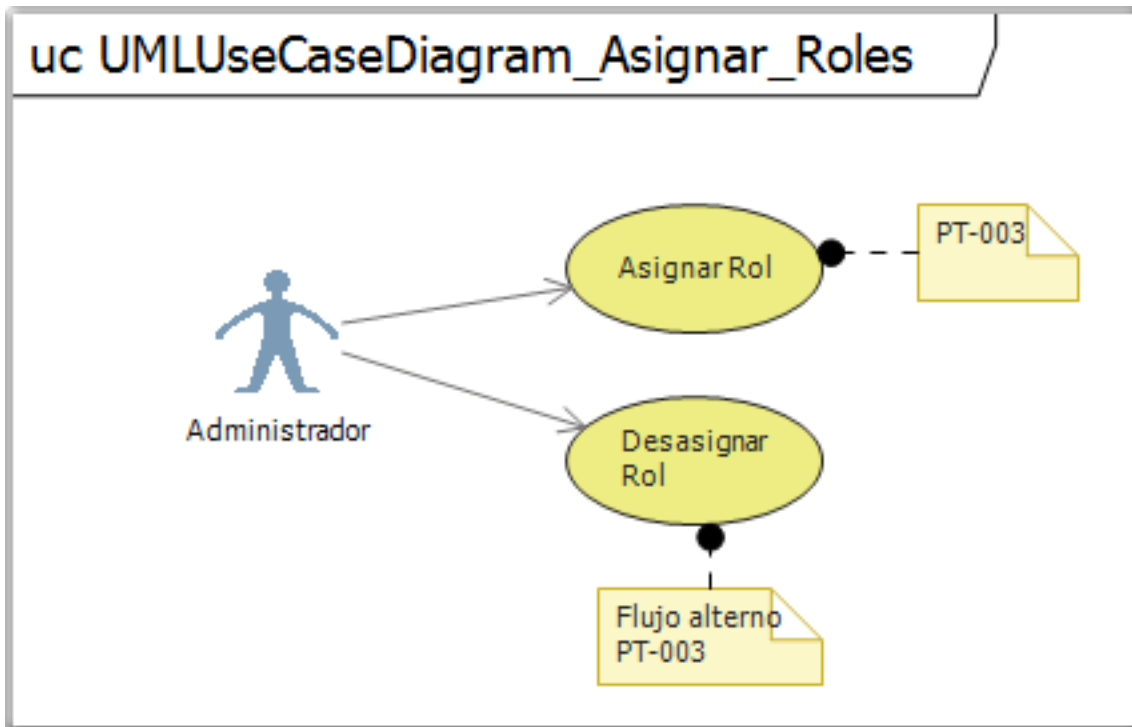


Ilustración 3. Caso de uso PT-003

1.4. Caso de uso Administrar facultad

Este caso de uso está asociado al requerimiento funcional RF-004 y describe la gestión y administración de facultades, el actor implicado es el administrador, la descripción general indica que el sistema ofrece un formulario el cual permite crear y editar las facultades, esencialmente el actor ingresa el nombre de la facultad y el sistema valida que este sea único, la información detallada del caso de uso se encuentra en la Tabla 4 y la gráfica asociada es la Ilustración 4.

Tabla 4. Caso de uso administrar facultad

Identificador:	PT-004	Requerimiento asociado:	RF-004
-----------------------	--------	--------------------------------	--------

Nombre:	Administrar facultad
Actores:	Administrador
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema.
Poscondiciones:	<ul style="list-style-type: none"> • Se ha creado una facultad. • Se han creado y asociado los programas a una facultad.
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear o actualizar la facultad. 2. El actor selecciona la opción de nuevo. 3. El sistema muestra los campos que van a estar asociados la facultad. 4. El sistema debe validar los campos obligatorios. 5. El sistema debe validar que no exista una facultad con el mismo nombre. 6. El actor selecciona guardar. 7. El sistema almacena la información. 8. El sistema confirma la transacción. 9. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. El actor selecciona la opción editar facultad.</p> <ol style="list-style-type: none"> 1. En el paso 1. 2. El sistema muestra los datos que se pueden actualizar. 3. El actor cambia los campos que requiera. 4. El sistema debe validar que no exista una facultad con el mismo nombre. 5. El actor selecciona guardar. 6. El sistema almacena los cambios. 7. El actor sale del caso de uso. 	
Excepciones:	

A. Validación de campo obligatorios facultad

1. En el paso 4 si el actor administra las facultades el sistema no guardar o actualiza la información si no existen los campos requeridos.
2. El sistema muestra un mensaje de alerta especificando los campos obligatorios.
3. Se regresa al paso 3.

B. Validación de nombre de la facultad.

1. En el paso 4 si el actor administra las facultades el sistema debe validar que no exista una facultad con el mismo nombre.
2. El sistema muestra un mensaje de alerta especificando el campo que debe cambiar.
3. Se regresa al paso 3.

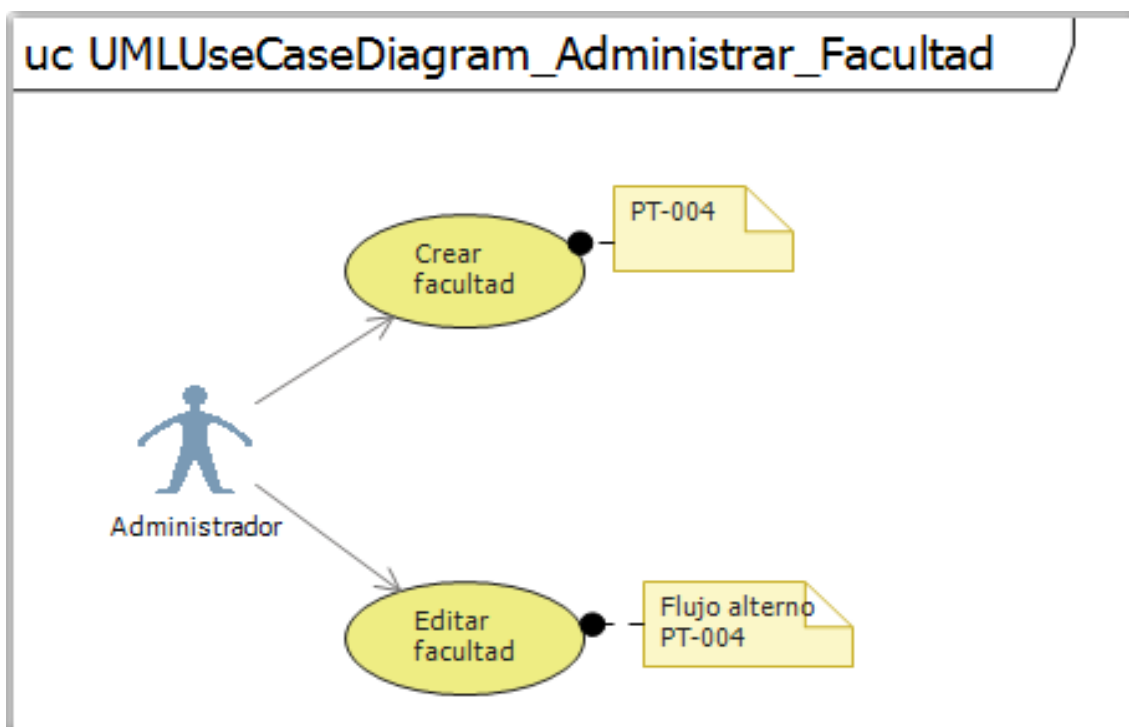


Ilustración 4. Caso de Uso PT-004

1.5. Caso de uso administrar programas

Este caso de uso está asociado al requerimiento funcional RF-004 y describe la gestión y administración de los programas de una facultad, el actor implicado es el administrador, la descripción general indica que el sistema ofrece un formulario el cual permite crear y editar los programas de una facultad, esencialmente el sistema muestra las facultades existente y el actor selecciona una de ellas, luego el actor ingresa el nombre del programa y el sistema valida que este sea único para esa facultad, la información detallada del caso de uso se encuentra en la Tabla 5 y la gráfica asociada es la Ilustración 5.

Tabla 5. Caso de uso Administrar programas

Identificador:	PT-005	Requerimiento asociado:	RF-004
Nombre:	Administrar programas		
Actores:	Administrador		
Precondiciones:	<ul style="list-style-type: none"> El actor se ha autenticado en el sistema. 		
Poscondiciones:	<ul style="list-style-type: none"> Se ha creado o actualiza un programa asociado a una facultad. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> El sistema muestra el formulario para crear o actualizar los programas de una facultad. El actor selecciona la opción de nuevo. El sistema muestra los campos que van a estar asociados al programa. El sistema debe validar los campos obligatorios. 			

5. El sistema debe validar que no exista un programa con el mismo nombre para la facultad asociada.
6. El actor selecciona guardar.
7. El sistema almacena la información.
8. El sistema confirma la transacción.
9. El actor sale del caso de uso.

Flujo alternos:

- A. El actor selecciona la opción editar programa.
 1. En el paso 1.
 2. El sistema muestra los datos que se pueden actualizar.
 3. El actor cambia los campos que requiera.
 4. El sistema debe validar que no exista un programa con el mismo nombre para una facultad.
 5. El actor selecciona guardar.
 6. El sistema almacena los cambios.
 7. El actor sale del caso de uso.

Excepciones:

- A. Validación de campo obligatorios programa
 1. En el paso 4 el sistema no guarda o actualiza la información si no existen los campos requeridos.
 2. El sistema muestra un mensaje de alerta especificando los campos obligatorios.
 3. Se regresa al paso 3.

- B. Validación de nombre de la programa.
 1. En el paso 4 el sistema debe validar que no exista un programa con el mismo nombre.
 2. El sistema muestra un mensaje de alerta especificando el campo que debe cambiar.
 3. Se regresa al paso 3.

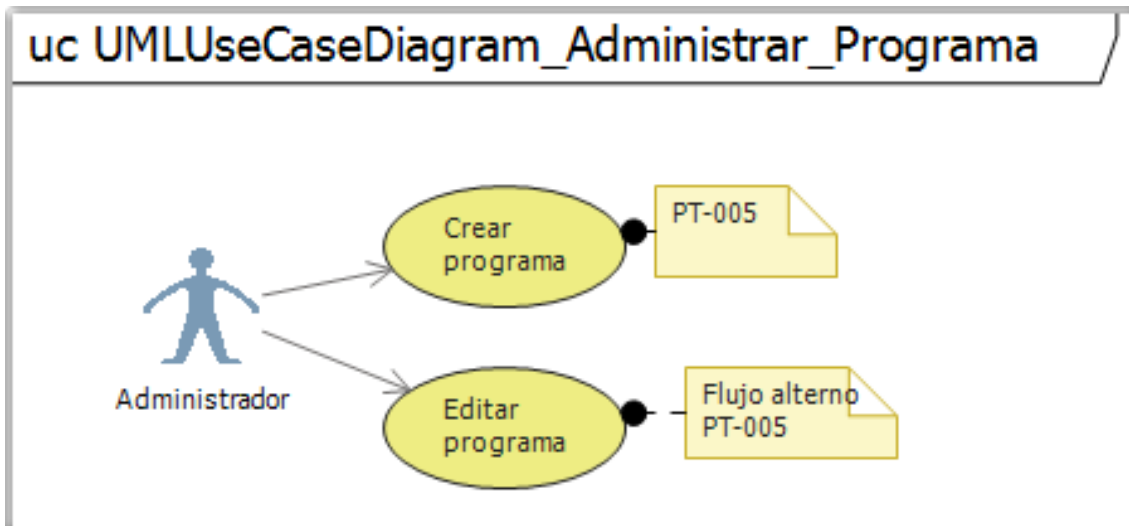


Ilustración 5. Caso de Uso PT-005

1.6. Caso de uso consultar log de errores

Este caso de uso está asociado al requerimiento funcional RF-005 y describe la consulta de errores controlados que ocurren en el sistema, el actor implicado es el administrador, básicamente el sistema muestra al actor la descripción detallada de los errores, la información detallada del caso de uso se encuentra en la Tabla 6 y la gráfica asociada es la Ilustración 6.

Tabla 6. Caso de uso consultar log de errores

Identificador:	PT-006	Requerimiento asociado:	RF-005
Nombre:	Consultar log de errores		
Actores:	Administrador		
Precondiciones:	<ul style="list-style-type: none"> El actor se ha autenticado en el sistema. 		
Poscondiciones:	<ul style="list-style-type: none"> Consulta de los errores del sistema. 		

Flujo normal de los eventos:
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar los errores que ha presentado el sistema filtrado por la fecha en que ocurrió el error y ordenado en forma descendente por dicha fecha. 2. El actor sale del caso de uso.
Flujo alternos:
B. No aplica.
Excepciones:
B. No aplica.

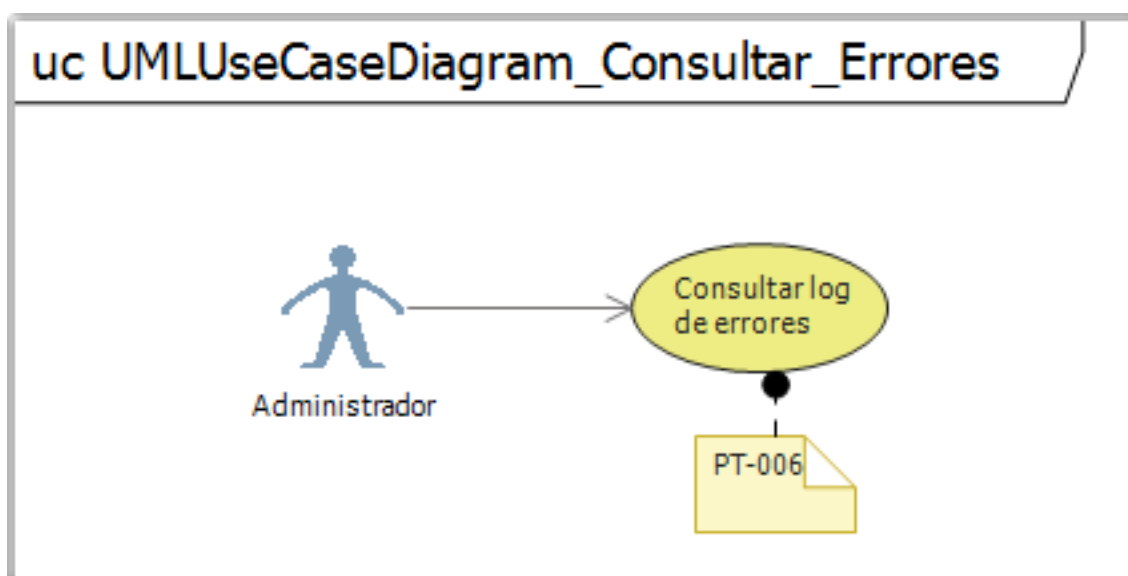


Ilustración 6. Caso de Uso PT-006

1.7. Caso de uso administrar proyectos

Este caso de uso está asociado al requerimiento funcional RF-006 y describe la gestión y administración de los proyectos, el actor implicado es el tutor, básicamente el

sistema muestra al actor un formulario para crear y editar los datos básicos del proyecto, como el nombre y descripción entre otros, también se muestra la opción de vincular un tutor al proyecto, básicamente el actor ingresa los datos básicos del proyecto y selecciona un tutor para ser vinculado al proyecto, la información detallada del caso de uso se encuentra en la Tabla 7 y la gráfica asociada es la Ilustración 7.

Tabla 7. Administrar proyectos

Identificador:	PT-007	Requerimiento asociado:	RF-006
Nombre:	Administrar Proyectos		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos usuarios con rol de tutor. • Deben estar creados algunos usuarios con rol de estudiante. 		
Poscondiciones:	<ul style="list-style-type: none"> • Proyecto creado o actualizado. • Tutor asociado a uno o varios proyectos. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear o actualizar los proyectos. 2. El actor selecciona la opción de nuevo. 3. El sistema muestra los campos que van a estar asociados al proyecto. 4. El sistema muestra la lista de tutores para asignar a un proyecto. 5. El actor selecciona un tutor de la lista. 6. El sistema debe validar los campos obligatorios. 7. El sistema debe validar que no exista un proyecto con el mismo nombre para la facultad asociada. 8. El sistema realizar el caso de uso (PT-08) 9. El actor selecciona guardar. 10. El sistema almacena la información. 			

<ol style="list-style-type: none">11. El sistema confirma la transacción.12. El actor sale del caso de uso.
Flujo alternos:
<ol style="list-style-type: none">A. El actor selecciona la opción editar proyecto.<ol style="list-style-type: none">1. En el paso 1.2. El sistema muestra los datos que se pueden actualizar.3. El actor cambia los campos que requiera.4. El sistema debe validar que no exista un proyecto con el mismo nombre para una facultad.5. El actor selecciona guardar.6. El sistema almacena los cambios.7. El actor sale del caso de uso.B. El sistema debe validar la asignación del proyecto.<ol style="list-style-type: none">1. En el paso 5.2. El sistema debe validar el rol del actor autenticado, si es administrador y tutor puede seleccionar cualquier tutor para asignarlo a un proyecto, pero si el rol es tutor el proyecto a crear o actualizar será asignado al actor autenticado.
Excepciones:
<ol style="list-style-type: none">A. Validación de campo obligatorios proyecto<ol style="list-style-type: none">1. En el paso 4 el sistema no guardar o actualiza la información si no existen los campos requeridos.2. El sistema muestra un mensaje de alerta especificando los campos obligatorios.3. Se regresa al paso 3.B. Validación de nombre de la proyecto.<ol style="list-style-type: none">1. En el paso 4 el sistema debe validar que no exista un proyecto con el mismo nombre.2. El sistema muestra un mensaje de alerta especificando el campo que debe cambiar.3. Se regresa al paso 3.

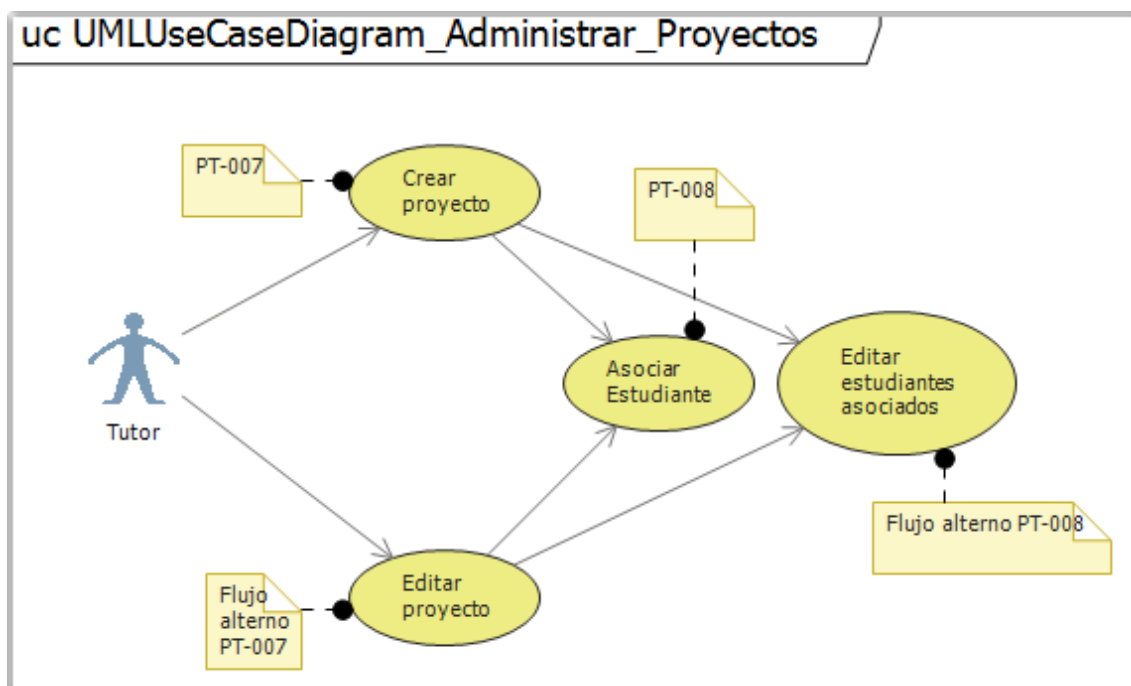


Ilustración 7. Casos de Uso PT- 007 y PT- 008

1.8. Caso de uso asociar estudiante a proyecto

Este caso de uso está asociado al requerimiento funcional RF-006 y describe la vinculación y la desvinculación de los integrantes al proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario con las listas de proyectos y estudiantes existentes, luego el actor selecciona un proyecto y los estudiantes para ser asociados a este proyecto, además el sistema ofrece la opción de editar la asociación de los integrantes del proyecto, la información detallada del caso de uso se encuentra en la Tabla 8 y la gráfica asociada es la Ilustración 7.

Tabla 8. Caso de uso asociar estudiante a proyecto

Identificador:	PT-008	Requerimiento asociado:	RF-006
Nombre:	Asociar estudiante a proyecto		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos usuarios con rol de estudiante. • Deben estar creados algunos proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Asociación de estudiantes a un proyecto 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para asociar los estudiantes a un proyecto. 2. El actor selecciona un proyecto. 3. El sistema muestra la lista de los estudiantes. 4. El actor selecciona un estudiante. 5. El sistema debe validar que el estudiante no este asociado a un proyecto. 6. El actor selecciona guardar. 7. El sistema almacena la información. 8. El sistema confirma la transacción. 9. El actor sale del caso de uso. 			
Flujo alternos:			
<ol style="list-style-type: none"> A. El actor selecciona la opción editar los estudiantes asociados al proyecto. <ol style="list-style-type: none"> 1. En el paso 5. 			

2. El sistema muestra los estudiantes ya asociados al proyecto para poder actualizar la asociación.
3. El actor selecciona el estudiante a asociar al proyecto o lo inactiva.
4. El sistema debe validar el estudiante no este asociado a otro proyecto o previamente ya este asociado al mismo proyecto.
5. El actor selecciona guardar.
6. El sistema almacena los cambios.
7. El actor sale del caso de uso.

Excepciones:

A. Validación de la asociación de estudiante al proyecto

1. En el paso 5.
2. El sistema debe validar la asociación.
3. El sistema muestra un mensaje de alerta especificando que ya está asociado el estudiante a un proyecto
4. Se regresa al paso 4.

1.9. Caso de uso administrar tareas

Este caso de uso está asociado al requerimiento funcional RF-007 y describe la gestión y administración de las tareas de los proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario para crear y editar los datos asociados a la tareas, como el título, descripción, proyecto, integrantes y fecha de ejecución entre otros, la información detallada del caso de uso se encuentra en la Tabla 9 y la gráfica asociada es la Ilustración 8.

Tabla 9. Caso de uso administrar tareas

Identificador:	PT-009	Requerimiento asociado:	RF-007
Nombre:	Administrar tareas		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyecto • Debe estar asociados los tutores a un proyecto. • Deben estar asociados los estudiantes a un proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Crear tarea 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para crear y editar las tareas de un proyecto 2. El sistema muestra una lista de los proyectos asociadas al rol de usuario. 3. El actor selecciona un proyecto. 4. El sistema muestra el tutor asociado. 5. El sistema muestra la lista de los estudiantes asociados al proyecto seleccionado. 6. El sistema realizar el caso de uso (PT-010) 7. El sistema muestra una lista de fecha inicial y fecha final del proyecto. 8. El actor selecciona la fecha inicial y final de la tarea 9. El sistema muestra campos adicionales como el título, descripción, comentario y archivos adjuntos de la tarea 10. El actor agrega los datos del título y la descripción que son obligatorios, opcionalmente los comentarios y los archivos adjuntos. 11. El sistema valida los campos obligatorios. 12. El actor selecciona guardar. 13. El sistema almacena la información. 14. El sistema confirma la transacción. 15. El actor sale del caso de uso. 			

Flujo alternos:
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none">1. En el paso 2.2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador y tutor muestra todos los proyectos. <p>B. El actor selecciona la opción editar la tarea.</p> <ol style="list-style-type: none">1. En el paso 1.2. El sistema muestra los datos asociados a la tarea.3. Si se necesita editar los estudiantes ya asociados. El sistema realiza el caso de uso (PT-010).4. El actor cambia los campos adicionales que requiera.5. El actor selecciona guardar.6. El sistema almacena los cambios.7. El actor sale del caso de uso.
Excepciones:
<p>A. Validación de campo obligatorios</p> <ol style="list-style-type: none">1. En el paso 10.2. El sistema no guardar o actualiza la información si no existen los campos requeridos.3. El sistema muestra un mensaje de alerta especificando los campos obligatorios.4. Se regresa al paso 10. <p>B. Validación fechas</p> <ol style="list-style-type: none">1. En el paso 8.2. El sistema debe validar que la fecha final sea mayor que la fecha inicial.3. El sistema muestra un mensaje de alerta de la inconsistencia.4. Regresa al paso 8

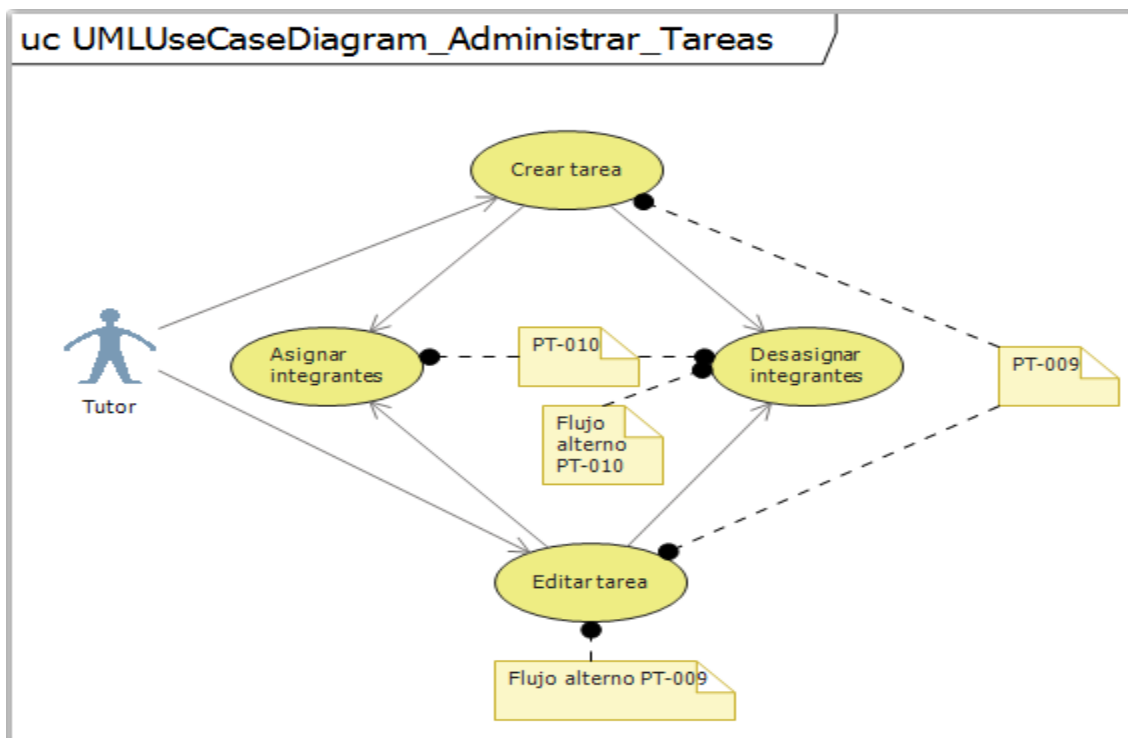


Ilustración 8. Casos de Uso PT-009 y PT-010

1.10. Caso de uso asociar estudiantes a tarea

Este caso de uso está asociado al requerimiento funcional RF-007 y describe la asociación de los estudiantes su proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario con los proyecto, sus tareas y los estudiantes asociados al proyecto, luego el actor selecciona los estudiantes y los vincula o desvincula de la tarea seleccionada, la información detallada del caso de uso se encuentra en la Tabla 10 y la gráfica asociada es la Ilustración 8.

Tabla 10. Caso de uso asociar estudiantes a tarea

Identificador:	PT-010	Requerimiento asociado:	RF-007
Nombre:	Asociar estudiante a tarea		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Se debe cumplir las precondiciones del caso de uso(PT-009) • Se realizar el evento 1, 2 ,3 y 5 del caso de uso(PT-009) 		
Poscondiciones:	<ul style="list-style-type: none"> • Asociación de estudiantes a una tarea 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El actor selecciona un estudiante. 2. El actor asigna o desasigna uno o varios estudiantes a la tarea. 3. El sistema debe validar que el estudiante no este asociado previamente a la tarea. 4. El actor selecciona guardar. 5. El sistema almacena la información. 6. El sistema confirma la transacción. 7. El actor sale del caso de uso. 			
Flujo alternos:			
<p>A. El actor selecciona la opción de desasignar los estudiantes asociados a la tarea.</p> <ol style="list-style-type: none"> 1. En el paso 3. 2. El sistema muestra los estudiantes ya asociados al proyecto para poder actualizar la asociación. 3. El actor desasigna el estudiante de la tarea. 4. El actor selecciona guardar. 5. El sistema almacena los cambios. 6. El actor sale del caso de uso. 			

Excepciones:
<p>A. Validación de la asociación de estudiante al proyecto</p> <ol style="list-style-type: none"> 1. En el paso 5. 2. El sistema debe validar la asociación. 3. El sistema muestra un mensaje de alerta especificando que ya está asociado el estudiante a un proyecto 4. Se regresa al paso 4.

1.11. Caso de uso administrar estado de la tarea

Este caso de uso está asociado al requerimiento funcional RF-008 y describe la gestión y administración de los estados de las tareas de los proyecto, el actor implicado es el tutor, básicamente el sistema muestra al actor un formulario para asociar un estado y un porcentaje de avance a la tarea, la información detallada del caso de uso se encuentra en la Tabla 11 y la gráfica asociada es la Ilustración 9.

Tabla 11. Caso de uso administrar estado de las tareas

Identificador:	PT-011	Requerimiento asociado:	RF-008
Nombre:	Administrar estado de las tareas		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creadas las tareas de un proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Estado de la tarea creado o actualizado 		

Flujo normal de los eventos:

1. El sistema muestra el formulario para crear o editar el estado de la tarea.
2. El sistema muestra una lista de los proyectos.
3. El actor selecciona un proyecto.
4. El sistema una lista de la tareas asociadas al proyecto.
5. El actor selecciona una tarea.
6. El sistema una lista de los estados de la tarea "creada, en progreso, sin solución, invalida, duplicada y finalizada".
7. El actor selecciona un estado.
8. El sistema muestra una lista con los valores del porcentaje de avance entre 0% y 100%.
9. El actor selecciona un valor del porcentaje.
10. El actor selecciona guardar.
11. El sistema almacena la información.
12. El sistema confirma la transacción.
13. El actor sale del caso de uso.

Flujo alternos:

A. Visualización de proyectos

1. En el paso 2.
2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador y tutor muestra todos los proyectos.

B. El actor actualiza el estado de la tarea.

1. En el paso 7.
2. El actor selecciona un estado de la tarea y un valor del porcentaje.
3. El actor selecciona guardar.
4. El sistema almacena los cambios.
5. El actor sale del caso de uso.

Excepciones:

A. No aplica

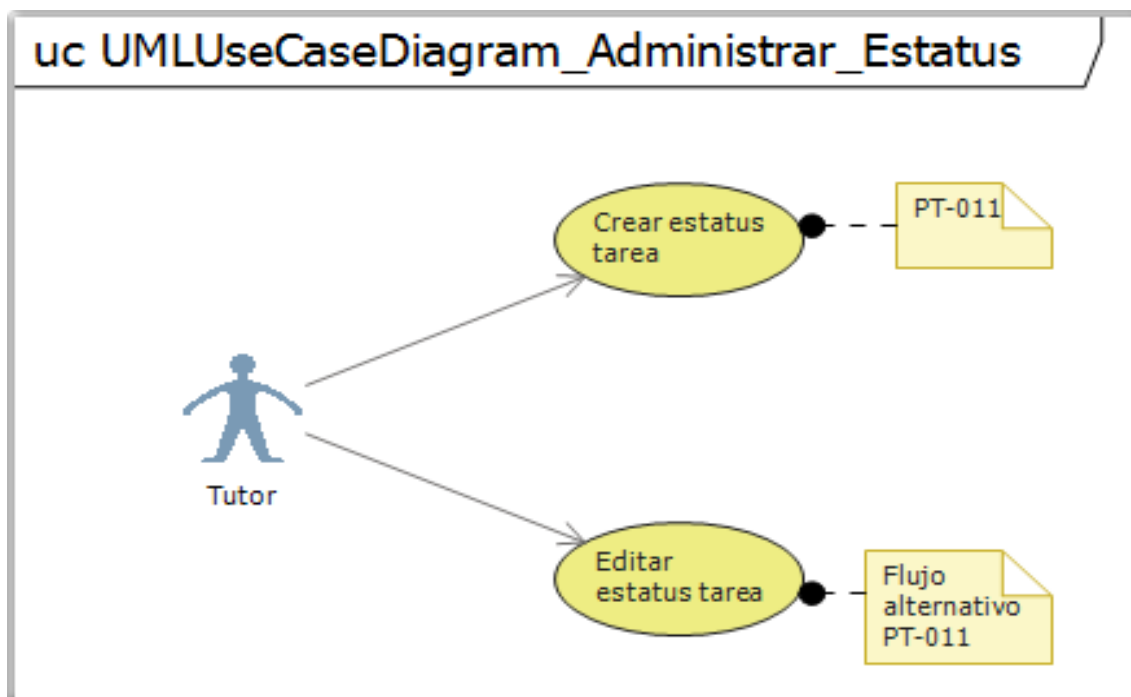


Ilustración 9. Caso de Uso PT-011

1.12. Caso de uso consultar gráficos del estado de las tareas

Este caso de uso está asociado al requerimiento funcional RF-009 y consiste en consultar el gráfico de barras horizontales que representa el porcentaje de avance de la tarea, el sistema muestra la lista de proyectos, luego el actor selecciona un proyecto y el sistema construye el gráfico, la información detallada del caso de uso se encuentra en la Tabla 12 y la gráfica asociada es la Ilustración 10.

Tabla 12. Consultar gráficos del estado de las tareas

Identificador:	PT-012	Requerimiento asociado:	RF-009
Nombre:	Consultar gráficos del estado de las tareas		
Actores:	Tutor y estudiante		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creadas las tareas de un proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los estado de las tareas de un proyecto en un gráfico de barras horizontales. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar los estado de las tareas. 2. El sistema muestra una lista de los proyectos. 3. El actor selecciona un proyecto. 4. El sistema muestra el gráfico asociado al estado de las tareas. 5. El actor selecciona una barra del gráfico. 6. El sistema muestra la descripción de estado de la tarea 7. El actor sale del caso de uso. 			
Flujo alternos:			
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 2. 2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador y tutor muestra todos los proyectos, si el rol es estudiante muestra el proyecto al que fue asociado. 			
Excepciones:			
B. No aplica			

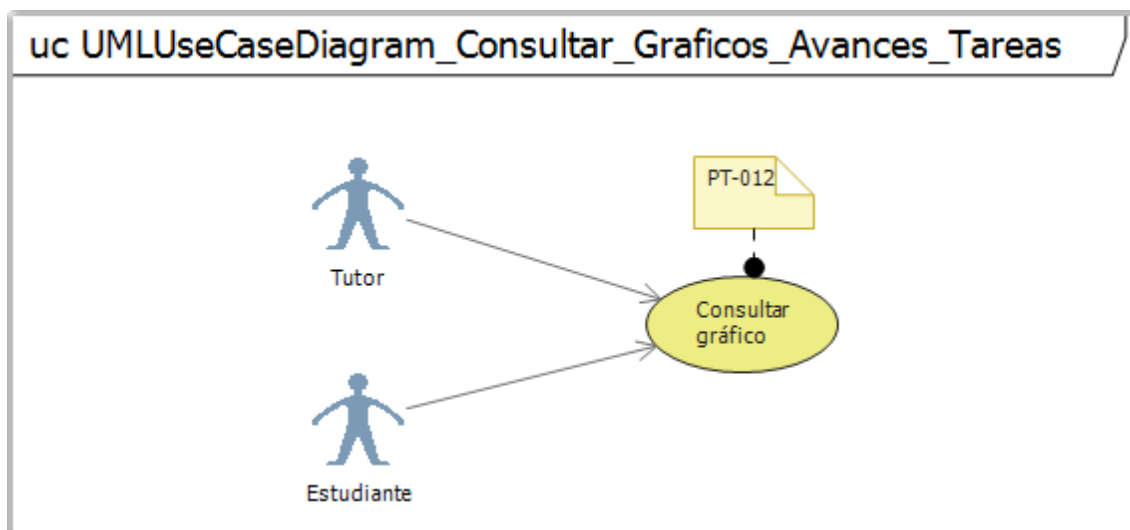


Ilustración 10. Caso de Uso PT - 012

1.13. Caso de uso consultar datos del proyecto

Este caso de uso está asociado al requerimiento funcional RF-010, para este caso el sistema muestra un formulario con los datos básicos del proyecto, actores implicados administrador, tutor y estudiante, la información detallada del caso de uso se encuentra en la Tabla 13 y la gráfica asociada es la Ilustración 11.

Tabla 13. Caso de uso consultar datos del proyecto

Identificador:	PT-013	Requerimiento asociado:	RF-0010
Nombre:	consultar datos del proyecto		
Actores:	Administrador, Tutor y estudiante		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto • Deben estar creadas las tareas de un proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los datos básicos del proyecto 		

Flujo normal de los eventos:
<ol style="list-style-type: none">1. El sistema muestra el formulario para visualizar los datos básicos del proyecto.2. El sistema muestra una lista de los proyectos.3. El actor selecciona un proyecto.4. El sistema muestra los datos básicos del proyecto.5. El actor sale del caso de uso.
Flujo alternos:
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none">1. En el paso 2.2. El sistema despliega la lista de proyectos asociados al usuario, si el rol es de tutor le muestra los proyectos que se le han asignado, si el rol del usuario es de administrador y tutor muestra todos los proyectos, si el actor tiene rol de estudiante se muestra solo el proyecto al que está asociado.
Excepciones:
<p>A. No aplica</p>

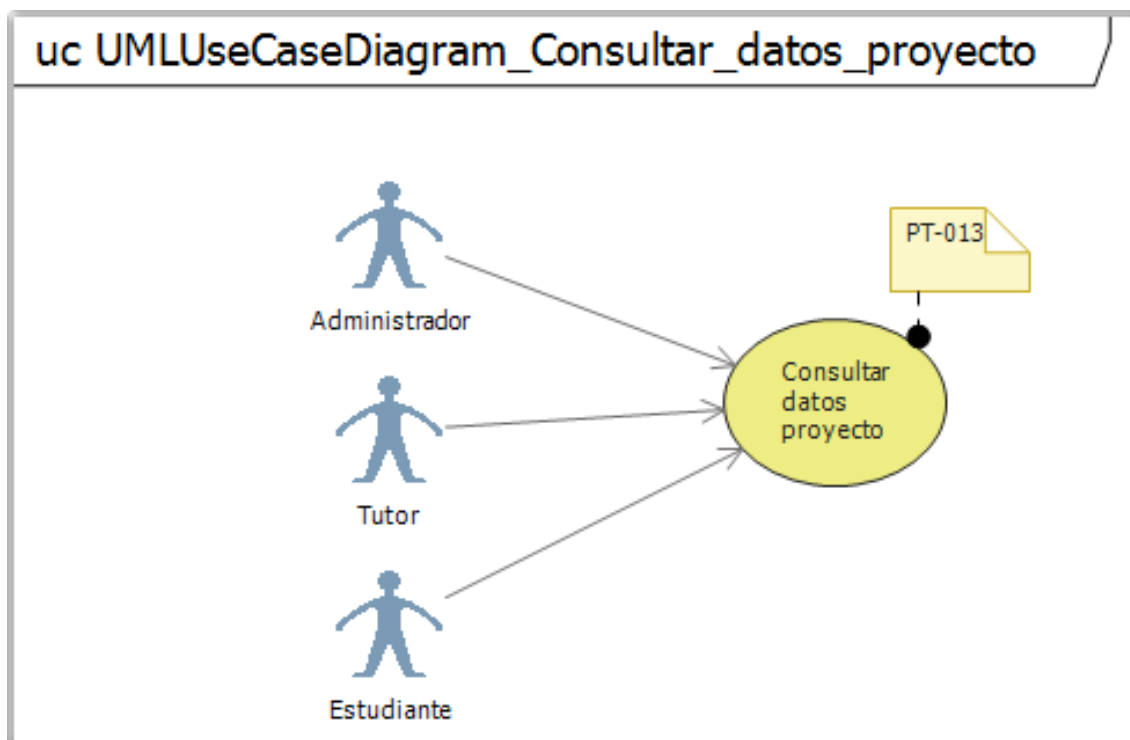


Ilustración 11. Caso de Uso PT-013

1.14. Caso de uso cerrar proyecto

Este caso de uso está asociado al requerimiento funcional RF-011, para este caso el sistema muestra un formulario con las lista de proyectos y ofrece la opción de cerrar o reabrir un proyecto y sus tareas asociadas, actores implicados administrador y tutor, la información detallada del caso de uso se encuentra en la Tabla 14 y la gráfica asociada es la Ilustración 12.

Tabla 14. Caso de uso cerrar proyecto

Identificador:	PT-014	Requerimiento asociado:	RF-011
Nombre:	Cerrar proyecto		

Actores:	Administrador y tutor
Precondiciones:	<ul style="list-style-type: none"> • Se debe cumplir las precondiciones del caso de uso(PT-002) • Debe estar creado el proyecto • Deben estar creadas las tareas de un proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los datos básicos del proyecto
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar los datos básicos del proyecto. 2. El sistema muestra una lista de los proyectos. 3. El actor selecciona un proyecto. 4. El sistema muestra la opción de cerrar el proyecto. 5. El actor selecciona cerrar o abrir el proyecto. 6. El actor selecciona guardar. 7. El sistema almacena la información. 8. El sistema confirma la transacción. 9. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 2. 2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador y tutor muestra todos los proyectos. <p>B. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 3. 2. El sistema muestra la opción de finalizar las tareas del proyecto al 100%. 3. El actor selecciona esta opción. 	

Excepciones:
A. No aplica

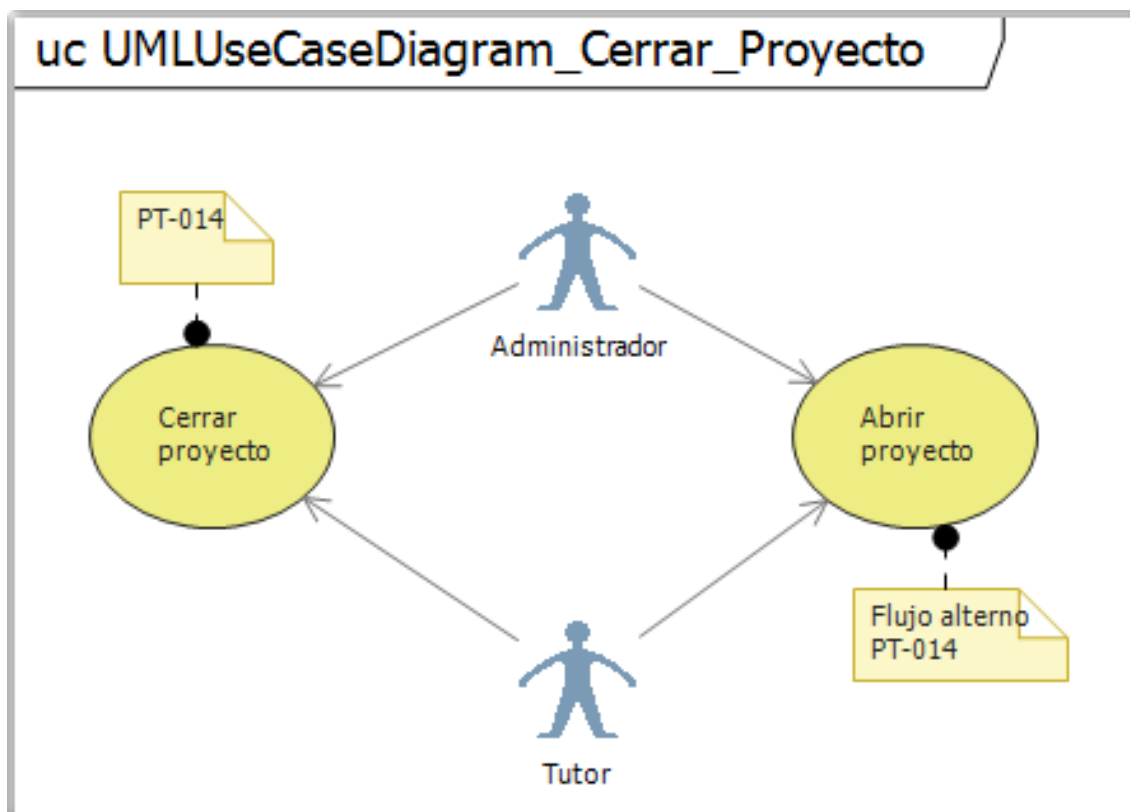


Ilustración 12. Caso de Uso PT-014

1.15. Caso de uso consultar reporte de estado del proyecto

Este caso de uso está asociado al requerimiento funcional RF-012, para este caso el sistema muestra un formulario con las lista de proyectos y ofrece la opción de generar un reporte de los estados del proyecto, actores implicados administrador, tutor y estudiantes, la información detallada del caso de uso se encuentra en la Tabla 15 y la gráfica asociada es la Ilustración 13.

Tabla 15. Caso de uso consultar reporte estado del proyecto.

Identificador:	PT-015	Requerimiento asociado:	RF-012
Nombre:	Consultar reporte estado del proyecto.		
Alias:	Consultar reporte estado proyecto.		
Actores:	Administrador, tutor y estudiante.		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyectos. • Deben estar creadas algunas tareas asociadas a un proyecto. 		
Poscondiciones:	<ul style="list-style-type: none"> • Reporte del estado de un proyecto en formato PDF 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para consultar los reportes asociados a un proyecto. 2. El sistema muestra una lista de los proyectos asociadas al rol de usuario. 3. El actor selecciona un proyecto. 4. El sistema muestra la opción de ver reporte. 5. El actor selecciona ver reporte. 6. El sistema genera el reporte en formato PDF. 7. El sistema confirma la transacción. 8. El actor sale del caso de uso. 			
Flujo alternos:			
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 1. 2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador muestra todos los proyectos, si el actor tiene rol de estudiante se muestra solo el proyecto al que está asociado. 			

Excepciones:
A. No aplica.

1.16. Caso de uso consultar reporte historial de proyectos por tutor

Este caso de uso está asociado al requerimiento funcional RF-013, para este caso el sistema muestra un formulario con las lista de tutores y ofrece la opción de generar un reporte del historial de los proyectos asociados al tutor seleccionado, actores implicados administrador y tutor, la información detallada del caso de uso se encuentra en la Tabla 16 y la gráfica asociada es la Ilustración 13.

Tabla 16. Caso de uso consultar reporte historial de proyectos por tutor

Identificador:	PT-016	Requerimiento asociado:	RF-013
Nombre:	Consultar reporte historial de proyectos por tutor.		
Alias:	Consultar reporte historial tutor		
Actores:	Administrador y tutor.		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyectos. • Deben estar creadas algunas tareas asociadas a un proyecto. 		
Poscondiciones:	<ul style="list-style-type: none"> • Reporte del historial de proyectos de un tutor en formato PDF. 		
Flujo normal de los eventos:			

<ol style="list-style-type: none"> 1. El sistema muestra el formulario para consultar el reporte del historial proyectos asociados a un tutor. 2. El sistema muestra una lista de los tutores. 3. El actor selecciona un tutor. 4. El sistema muestra una lista de fecha inicial y fecha final para realizar el filtro de la fecha de creación de los proyectos asignados al tutor seleccionado. 5. El actor selecciona la fecha inicial y final. 6. El sistema muestra la opción de ver reporte. 7. El actor selecciona ver reporte. 8. El sistema genera el reporte en formato PDF. 9. El sistema confirma la transacción. 10. El actor sale del caso de uso.
Flujo alternos:
<ol style="list-style-type: none"> 1. Visualización de tutores <ol style="list-style-type: none"> 1.1. En el paso 1. 1.2. El sistema despliega la lista de tutores. 1.3. Si el rol del actor es tutor, la lista solo mostrara el mismo actor, si el rol del usuario es de administrador muestra todos los tutores.
Excepciones:
<ol style="list-style-type: none"> A. No aplica.

1.17. Caso de uso consultar reporte historial proyectos por programa

Este caso de uso está asociado al requerimiento funcional RF-013, para este caso el sistema muestra un formulario con las lista de los programas por facultad y ofrece la opción de generar un reporte del historial de los proyectos asociados a la facultad

seleccionada, actor implicado administrador, la información detallada del caso de uso se encuentra en la Tabla 17 y la gráfica asociada es la Ilustración 13.

Tabla 17. Caso de uso consultar reporte historial proyectos por programa

Identificador:	PT-017	Requerimiento asociado:	RF-013
Nombre:	Consultar reporte Historial proyectos por Programa.		
Alias:	Consultar reporte Historial programa		
Actores:	Administrador.		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyectos. • Deben estar creadas algunas tareas asociadas a un proyecto. 		
Poscondiciones:	<ul style="list-style-type: none"> • Reporte del historial de proyectos asociados a una facultad en formato PDF. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para consultar el reporte del historial proyectos asociados a un programa. 2. El sistema muestra una lista de las facultades. 3. El actor selecciona una facultad. 4. El sistema muestra una lista de los programas asociados a la facultad seleccionada por el actor. 5. El actor selecciona un programa. 6. El sistema muestra una lista de fecha inicial y fecha final para realizar el filtro de la fecha de creación de los proyectos asociados al programa. 7. El actor selecciona la fecha inicial y final. 8. El sistema muestra la opción de ver reporte. 9. El actor selecciona ver reporte. 10. El sistema genera el reporte en formato PDF. 			

<p>11. El sistema confirma la transacción.</p> <p>12. El actor sale del caso de uso.</p>
Flujo alternos:
A. No aplica.
Excepciones:
A. No aplica.

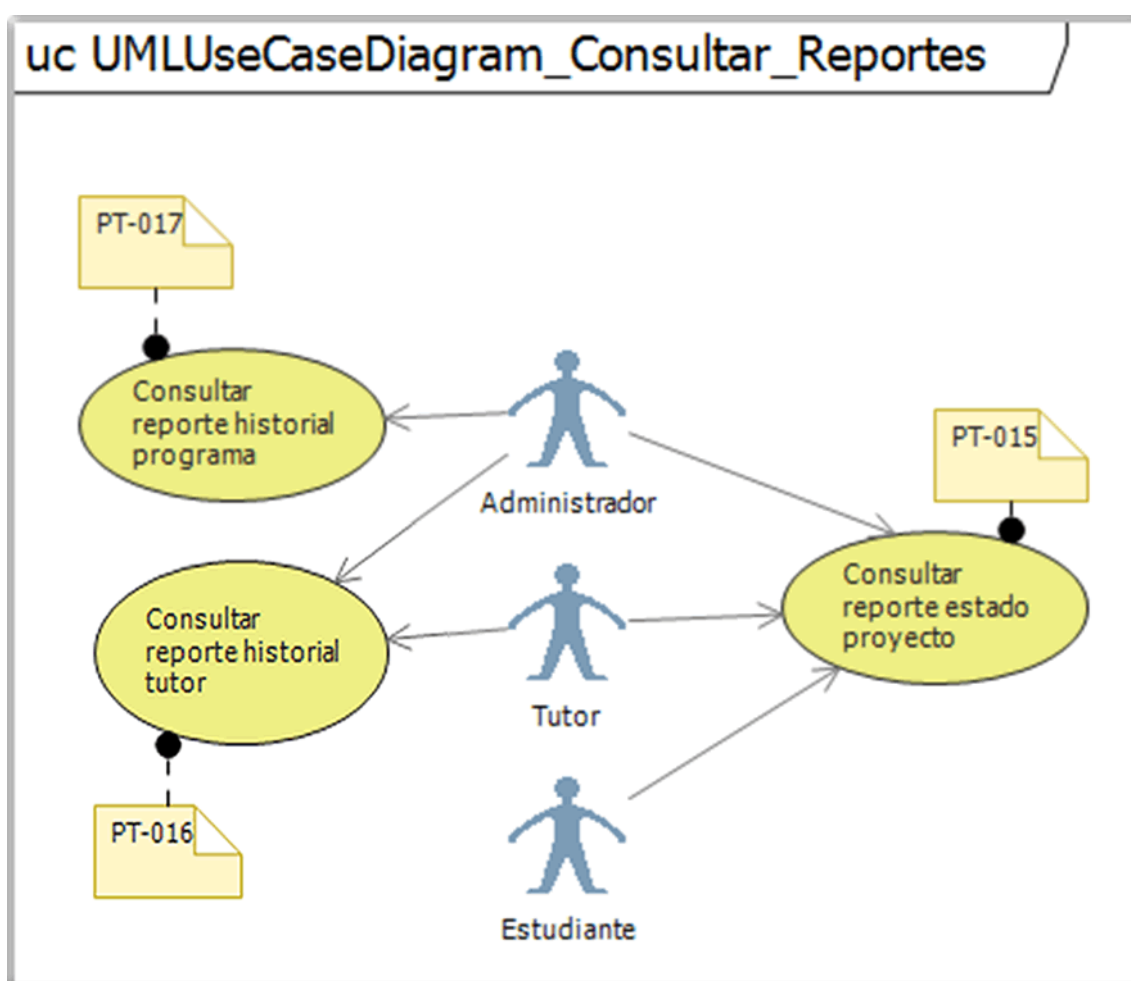


Ilustración 13. Casos de Uso PT-15, PT-16 y PT-17

1.18. Caso de uso revisar avances de tareas del proyecto.

Este caso de uso está asociado al requerimiento funcional RF-014, para este caso el sistema muestra un formulario con las lista de avances de las tareas de un proyecto, luego el actor tiene la opción de revisar los datos asociados al avance de la tarea y puede agregar una respuesta a dicho avance, el actor implicado es el tutor, la información detallada del caso de uso se encuentra en la Tabla 18 y la gráfica asociada es la Ilustración 14.

Tabla 18. Caso de uso revisar avances de tareas del proyecto.

Identificador:	PT-018	Requerimiento asociado:	RF-014
Nombre:	Revisar avances de tareas del proyecto.		
Alias:	Revisar avances.		
Actores:	Tutor		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Deben estar creados algunos proyectos. • Deben estar creadas algunas tareas asociadas a un proyecto. • Deben estar creados algunos avances asociados a una tarea. 		
Poscondiciones:	<ul style="list-style-type: none"> • Revisión del avance de las tareas de un proyecto y su correspondiente retroalimentación. 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para revisar y enviar respuestas a los avances de las tareas de un proyecto. 			

2. El sistema muestra una lista de los proyectos.
3. El actor selecciona un proyecto.
4. El sistema muestra una lista de tareas asociadas al proyecto.
5. El actor selecciona una tarea.
6. El sistema muestra los avances asociados a una tarea seleccionada por el actor.
7. El actor selecciona un avance de una tarea.
8. El actor revisa el avance.
9. El actor ingresa una respuesta del avance.
10. El actor selecciona guardar.
11. El sistema guarda la respuesta.
12. El sistema actualiza el estado del avance.
13. El actor sale del caso de uso.

Flujo alternos:

A. Visualización de proyectos

1. En el paso 2.
2. El sistema despliega la lista de proyectos asociados al usuario con rol de tutor, si el rol del usuario es de administrador muestra todos los proyectos.

Excepciones:

A. Validación de campo obligatorios de la respuesta

1. En el paso 10.
2. Si el actor no agrega una respuesta.
3. El sistema muestra un mensaje de alerta especificando los campos obligatorios.
4. Se regresa al paso 9.

B. Adjuntar archivo de respuesta.

1. En el paso 9.
2. El sistema muestra la opción de adjuntar archivos.
3. El actor carga uno o varios archivos de las respuestas de los avances de la tarea.
4. El sistema continúa en el paso 10.

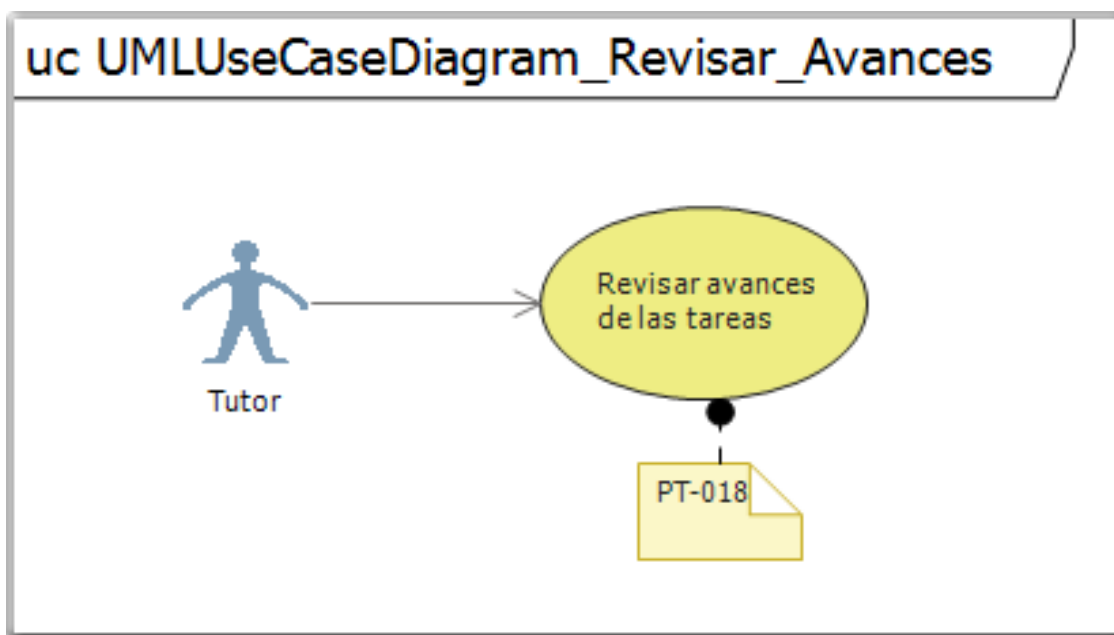


Ilustración 14. Caso de Uso PT-018

1.19. Caso de uso consultar tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-016, básicamente el sistema muestra al actor un formulario con la lista de tareas de un proyecto, el actor selecciona una tarea y el sistema le muestra una descripción detallada de los datos asociados de la tarea, tales como título, descripción, integrantes, etc. El actor implicado es el tutor, la información detallada del caso de uso se encuentra en la Tabla 19 y la gráfica asociada es la Ilustración 15.

Tabla 19. Caso de uso consultar tareas de un proyecto

Identificador:	PT-019	Requerimiento asociado:	RF-016
-----------------------	--------	--------------------------------	--------

Nombre:	Consultar tareas de un proyecto
Alias:	Consultar tareas
Actores:	Estudiante
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los datos de las tareas de un proyecto del actor
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar los datos de las tareas de un proyecto asociado al actor. 2. El sistema muestra la opción de visualizar las tareas de todos los integrantes o solo las tareas del actor. 3. El actor selecciona la opción de visualizar las tareas de todos los integrantes. 4. El sistema muestra una lista de las tareas con sus datos, el último estado y la diferencia de días entre la fecha del sistema y la fecha final de la tarea. 5. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. Visualización de tareas del actor</p> <ol style="list-style-type: none"> 1. En el paso 3. 2. El actor selecciona la opción de ver solo sus tareas. 3. El sistema continúa en el paso 4. 	
Excepciones:	
A. No aplica	

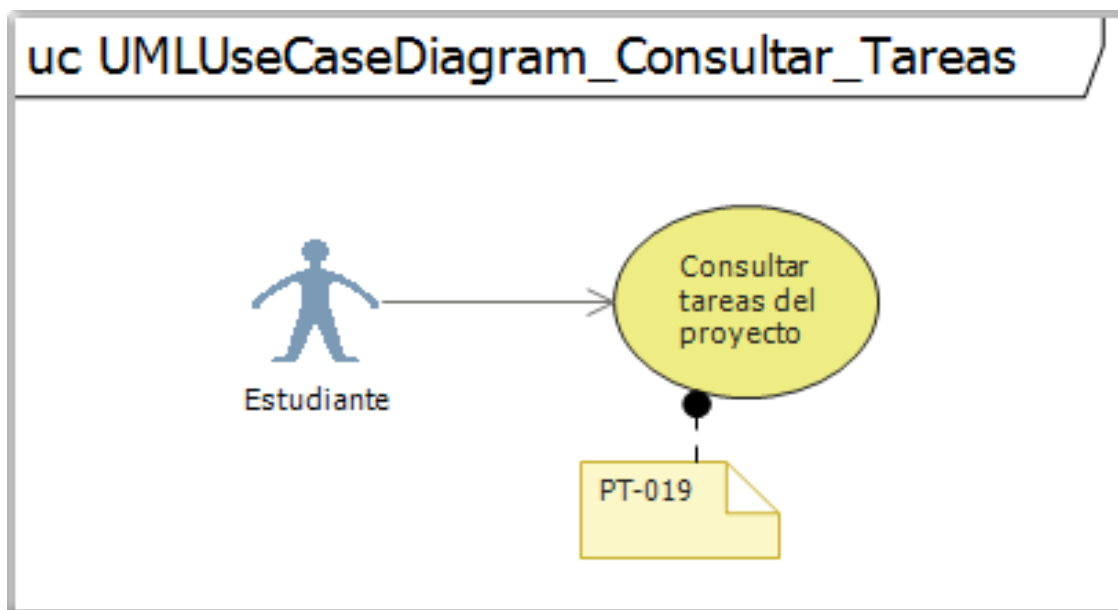


Ilustración 15. Caso de Uso PT-019

1.20. Caso de uso enviar avances de tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-017, el sistema ofrece un formulario con la lista de tareas de un proyecto asociado a un estudiante, luego el actor selecciona una tarea y adjunto los datos del avance de dicha tarea, el actor implicado es el estudiante, la información detallada del caso de uso se encuentra en la Tabla 19 y la gráfica asociada es la Ilustración 16.

Tabla 20. Caso de uso enviar avances de tareas de un proyecto

Identificador:	PT-020	Requerimiento asociado:	RF-017
Nombre:	Enviar avances de tareas de un proyecto		
Alias:	Enviar avances		

Actores:	Estudiante
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Los avances de las tareas han sido enviados
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para enviar los avances de las tareas del actor. 2. El sistema muestra la lista de las tareas asignadas al actor. 3. El actor selecciona una tarea. 4. El actor ingresa los datos del avance. 5. El actor selecciona guardar. 6. El sistema almacena la información. 7. El sistema confirma la transacción. 8. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. Validación de campo obligatorios</p> <ol style="list-style-type: none"> 1. En el paso 5. 2. El sistema no guardar o actualiza la información si no existen los campos requeridos. 3. El sistema muestra un mensaje de alerta especificando los campos obligatorios. 4. Se regresa al paso 5. <p>B. Adjuntar archivo del avance</p> <ol style="list-style-type: none"> 1. En el paso 4. 2. El sistema muestra la opción de enviar el avance del proyecto en un archivo adjunto. 3. El actor carga el archivo adjunto. 4. El sistema continúa en el paso 5. 	

Excepciones:
B. No aplica

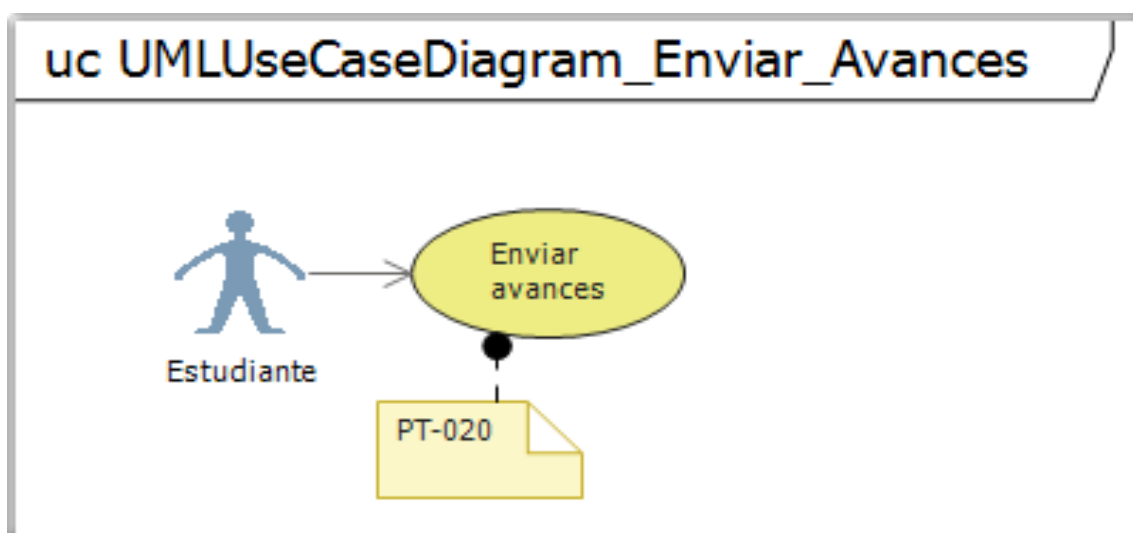


Ilustración 16. Caso de Uso PT-020

1.21. Caso de uso consultar estado de los avances de tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-018, básicamente el sistema muestra al actor un formulario con la lista de tareas de un proyecto, el actor selecciona una tarea y el sistema le muestra una descripción detallada del estado de los avances de la tarea y sus posibles repuestas. El actor implicado es el estudiante, la información detallada del caso de uso se encuentra en la Tabla 21 y la gráfica asociada es la Ilustración 15.

Tabla 21. Caso de uso consultar estado de los avances de tareas de un proyecto

Identificador:	PT-021	Requerimiento asociado:	RF-018
Nombre:	Consultar estado de los avances de tareas de un proyecto		
Alias:	Consultar estado del avance		
Actores:	Estudiante		
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto • Deben estar creados los avances del proyecto 		
Poscondiciones:	<ul style="list-style-type: none"> • Visualización de los avances de las tareas del proyecto 		
Flujo normal de los eventos:			
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para consultar los avances de las tareas del proyecto. 2. El sistema muestra la lista de las tareas asignadas del proyecto. 3. El actor selecciona una tarea. 4. El sistema muestra la lista de los avances de la tarea seleccionada por el actor. 5. El sistema muestra la descripción de los avances con sus estados y sus posibles repuestas. 6. El actor sale del caso de uso. 			
Flujo alternos:			
A. No aplica.			
Excepciones:			
A. No aplica			

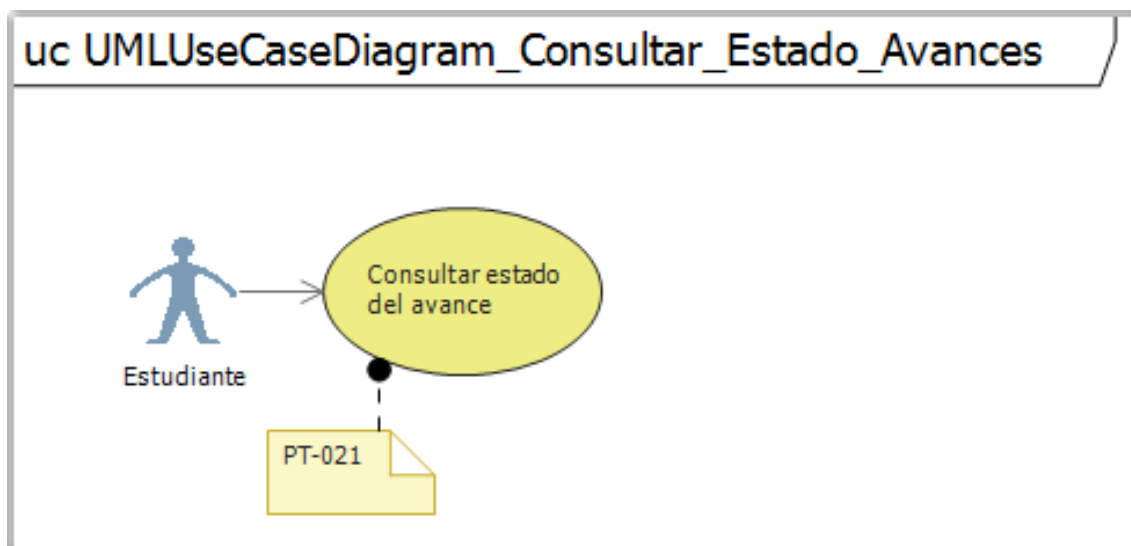


Ilustración 17. Caso de Uso PT-021

1.22. Caso de uso enviar alerta de estados de los avances de tareas de un proyecto

Este caso de uso está asociado al requerimiento funcional RF-019, básicamente el sistema muestra al actor un formulario para enviar las alertas de los estados de los avances de las tareas, si el actor selecciona enviar alertas, el sistema envía un correo electrónico a todos los integrantes y al tutor del proyecto con un reporte del estado de los avances de las tareas del proyecto. Los actores implicados son el tutor y el estudiante, la información detallada del caso de uso se encuentra en la Tabla 22 y la gráfica asociada es la Ilustración 18.

Tabla 22. Caso de uso enviar alerta de estados de los avances de tareas de un proyecto

Identificador:	PT-022	Requerimiento asociado:	RF-019
-----------------------	--------	--------------------------------	--------

Nombre:	Enviar alerta de estados de los avances de tareas de un proyecto
Alias:	Enviar alertas
Actores:	Tutor y Estudiante
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar creado el proyecto del actor • Deben estar creadas las tareas del proyecto • Deben estar creados los avances del proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Envío de correo de alertas de los avances de las tareas del proyecto
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para enviar los estados de las tareas del proyecto. 2. El sistema consulta los estados de las tareas del proyecto del actor. 3. El sistema envía un correo electrónico con el reporte de los estados de las tareas del proyecto a los integrantes y al tutor del proyecto del actor. 4. El actor sale del caso de uso. 	
Flujo alternos:	
A. No aplica.	
Excepciones:	
A. No aplica	

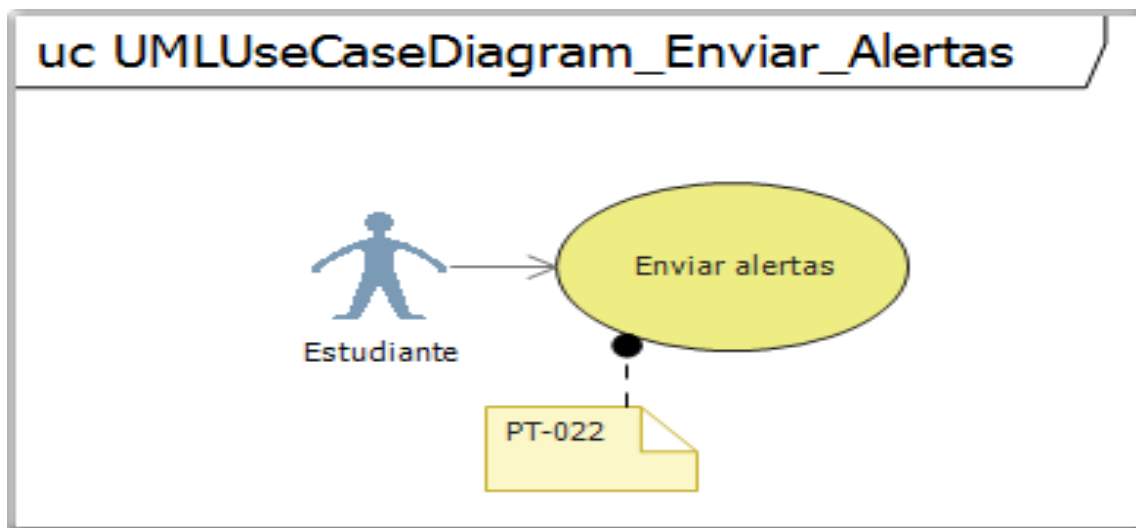


Ilustración 18.Caso de Uso PT-022

1.23. Caso de uso consultar diagrama de Gantt

Este caso de uso está asociado al requerimiento funcional RF-020 y consiste en consultar el gráfico o diagrama de Gantt de las tareas de un proyecto, básicamente el sistema grafica una línea de tiempo con las fechas de ejecución de las tareas, además el sistema ofrece la opción de visualizar las descripción detallada de cada tarea, la información detallada del caso de uso se encuentra en la Tabla 23 y la gráfica asociada es la Ilustración 19.

Tabla 23. Caso de uso consultar diagrama de Gantt

Identificador:	PT-023	Requerimiento asociado:	RF-020
Nombre:	Consultar diagrama de Gantt		

Alias:	Consultar Gantt
Actores:	Tutor y Estudiante
Precondiciones:	<ul style="list-style-type: none"> • El actor se ha autenticado en el sistema. • Debe estar algunos proyectos • Deben estar creadas las tareas del proyecto • Deben estar creados los estados del proyecto
Poscondiciones:	<ul style="list-style-type: none"> • Grafico del diagrama de Gantt
Flujo normal de los eventos:	
<ol style="list-style-type: none"> 1. El sistema muestra el formulario para visualizar el diagrama de Gantt de un proyecto. 2. El sistema muestra las lista de proyectos. 3. El actor selecciona un proyecto. 4. El sistema genera un diagrama de Gantt del proyecto. 5. El actor sale del caso de uso. 	
Flujo alternos:	
<p>A. Visualización de proyectos</p> <ol style="list-style-type: none"> 1. En el paso 2. 2. El sistema despliega la lista de proyectos asociados al usuario, si el rol es de tutor le muestra los proyectos que se le han asignado, si el rol del usuario es de administrador y tutor muestra todos los proyectos, si el actor tiene rol de estudiante se muestra solo el proyecto al que está asociado. 	
Excepciones:	
A. No aplica	

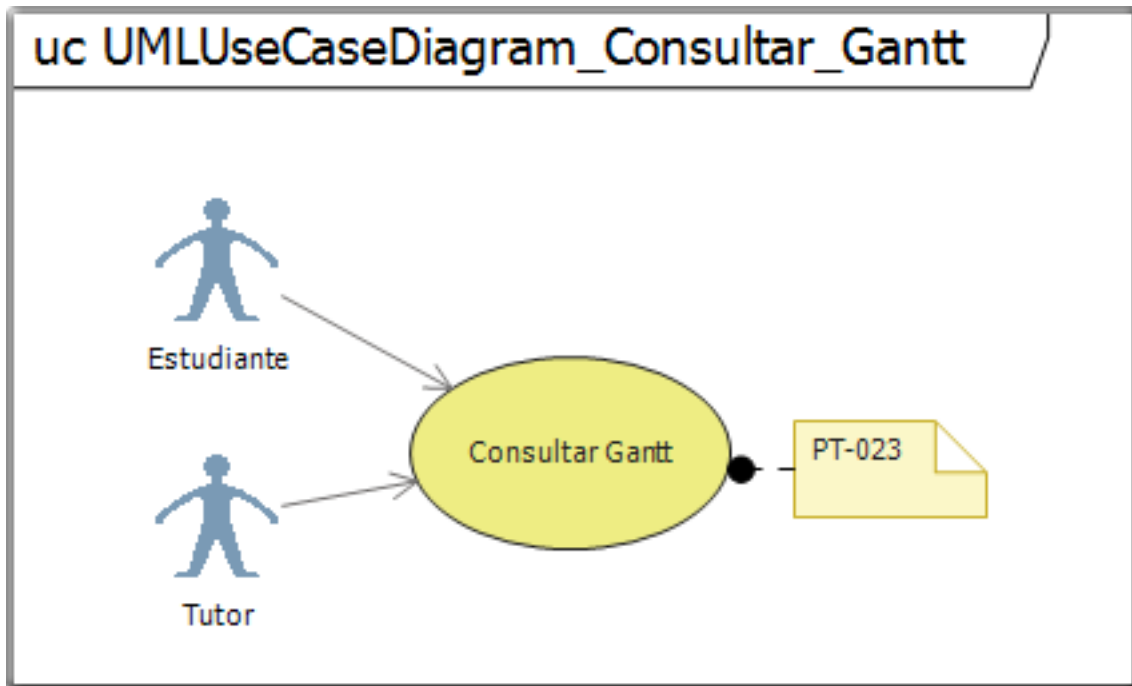


Ilustración 19. Caso de Uso PT-023

ANEXO 3

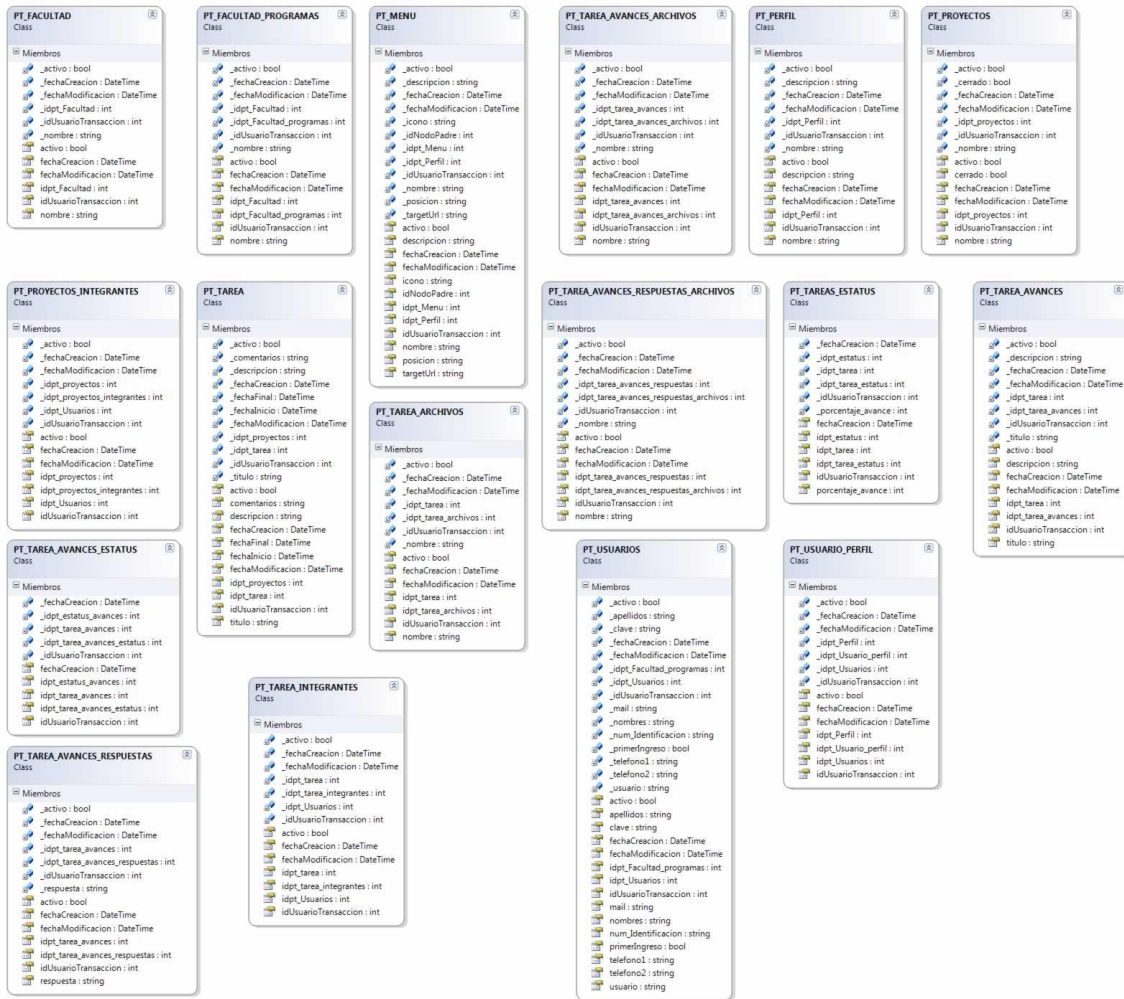


Ilustración 21. Capa EntityLayerProjectTracker

ANEXO 5

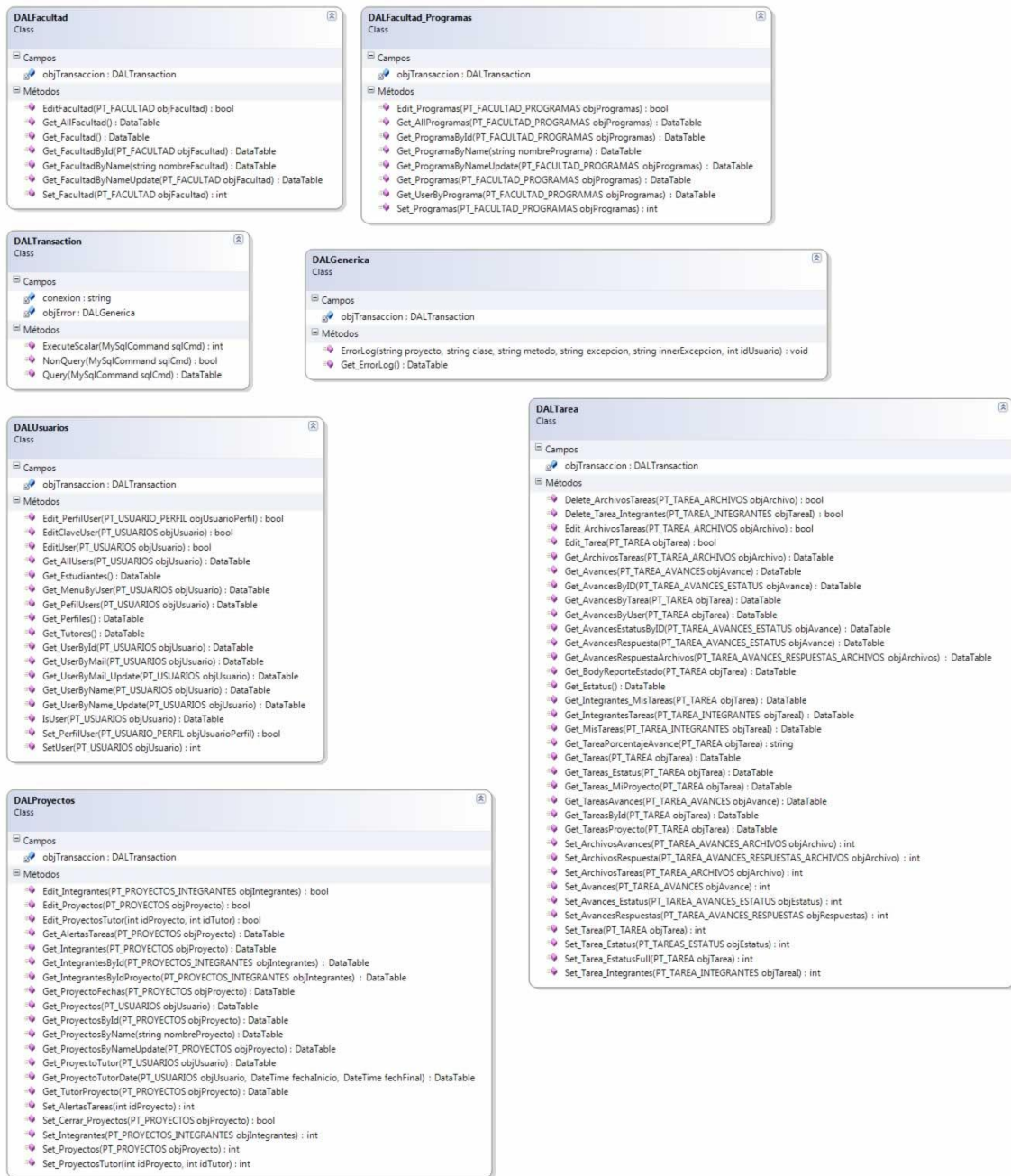


Ilustración 23. Capa DataAccessLayerProjectTracker

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Trabajo dirigido

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Tutor

Ingeniero Alexander García Pérez

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

TABLA DE ILUSTACIONES

Ilustración 1. Diccionario de datos tabla pt_alerta_tareas	4
Ilustración 2. Diccionario de datos tabla pt_error_log	5
Ilustración 3. Diccionario de datos tabla pt_estatus	5
Ilustración 4. Diccionario de datos tabla pt_estatus_avances.....	5
Ilustración 5. Diccionario de datos tabla pt_facultad	6
Ilustración 6. Diccionario de datos tabla pt_facultad_programas	6
Ilustración 7. Diccionario de datos tabla pt_menu	7
Ilustración 8. Diccionario de datos tabla pt_perfil	7
Ilustración 9. Diccionario de datos tabla pt_proyectos.....	8
Ilustración 10. Diccionario de datos tabla pt_proyectos_integrantes	8
Ilustración 11. Diccionario de datos tabla pt_proyectos_tutores.....	9
Ilustración 12. Diccionario de datos tabla pt_tarea	9
Ilustración 13. Diccionario de datos tabla pt_tarea_archivos.....	10
Ilustración 14. Diccionario de datos tabla pt_tarea_avances.....	10
Ilustración 15. Diccionario de datos tabla pt_tarea_avances_archivos	11
Ilustración 16. Diccionario de datos tabla pt_tarea_avances_estatus.....	11
Ilustración 17. Diccionario de datos tabla pt_tarea_avances_respuestas	12
Ilustración 18. Diccionario de datos tabla pt_tarea_avances_respuestas_archivos.....	12
Ilustración 19. Diccionario de datos tabla pt_tarea_estatus	13
Ilustración 20. Diccionario de datos tabla pt_tarea_integrantes.....	13
Ilustración 21. Diccionario de datos tabla pt_usuario_perfil.....	13
Ilustración 22. Diccionario de datos tabla pt_usuarios.....	14

ANEXO 7

1. DICCIONARIO DE DATOS

En el diccionario de datos se describen todas las tablas de la base de datos, dicha descripción contiene una breve explicación de los datos que debe almacenar la tabla, también contiene los nombre de los campos con sus tipos de datos, restricciones, valores por defecto, además indica si los campos son llaves primarias o foráneas, si el campo recibe datos nulos, si son únicos o si el campo se genera de forma auto numérica, en resumen el diccionario de dato contiene la descripción de la estructura de cada tabla de la base de datos.

1.1. Tabla pt alerta tareas

Esta tabla almacena el log de las alerta de los estados de las tareas de un proyecto que han sido enviadas por el software (Véase Ilustración 1).

pt_alerta_tareas										
Almacena el log de alertas de tareas de un proyecto										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Alerta_Tareas	INT(11)	✓	✓					✓		Id del alerta
fecha_Alerta	DATETIME		✓							Fecha en la que se envió el alerta
idpt_proyectos	INT(11)		✓							Id del proyecto que se le envió el alerta

Ilustración 1. Diccionario de datos tabla pt_alerta_tareas

1.2. Tabla pt error log

Esta tabla almacena los errores controlados de la aplicación web. (Véase Ilustración 2).

pt_error_log											
Almacenamos las excepciones de la aplicación											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_Error	INT(11)	✓	✓					✓		Id del error	
proyecto	VARCHAR(200)		✓							Nombre la capa de aplicación donde ocurrió el error	
clase	VARCHAR(200)		✓							Nombre de la clase donde ocurrió el error	
metodo	VARCHAR(200)		✓							Nombre del metodo donde ocurrió el error	
exception	VARCHAR(1000)		✓							Descripción de la excepción	
innerException	VARCHAR(1000)		✓							Descripción de la excepción interna	
idUsuario	INT(11)		✓							Id del usuario asociado el error	
fechaCreacion	DATETIME		✓							Fecha en la que se produjo el error	

Ilustración 2. Diccionario de datos tabla pt_error_log

1.3. Tabla pt estatus

Esta tabla almacena los tipos de estados que puede tener una tarea, estas pueden ser, creada, duplicada, en progreso, finalizada, invalida y sin solución. (Véase Ilustración 3).

pt_estatus											
Almacena los tipos de estatus de la tarea											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_estatus	INT(11)	✓	✓					✓		Id del estatus de la tarea	
nombre	VARCHAR(45)		✓							Nombre del estatus de la tarea	

Ilustración 3. Diccionario de datos tabla pt_estatus

1.4. Tabla pt estatus avances

Esta tabla almacena y guardan los tipos de estados que tiene un avance, los tipos son: enviado, visto y revisado. (Véase Ilustración 4).

pt_estatus_avances											
Almacena los tipos de estatus de Iso avances											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_estatus_avances	INT(11)	✓	✓					✓		Id del estatus del avance	
nombre	VARCHAR(100)		✓							Nombre del estatus de la tarea	

Ilustración 4. Diccionario de datos tabla pt_estatus_avances

1.5. Tabla pt facultad

Esta tabla almacena los datos de las facultades de la universidad EAN. (Véase Ilustración 5).

pt_facultad										
Almacena las facultades de la universidad EAN										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Facultad	INT(11)	✓	✓					✓		Id de la facultad
nombre	VARCHAR(100)		✓							Nombre de la facultad
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción

Ilustración 5. Diccionario de datos tabla pt_facultad

1.6. Tabla pt facultad

Esta tabla almacena los programas de las facultades de la universidad EAN. (Véase Ilustración 1).

pt_facultad_programas										
Almacena los programas de las facultades de la universidad EAN										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Facultad_programas	INT(11)	✓	✓					✓		Id del programa
idpt_Facultad	INT(11)		✓							Id de la facultad
nombre	VARCHAR(45)		✓							Nombre del programa
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción

Ilustración 6. Diccionario de datos tabla pt_facultad_programas

1.7. Tabla pt menú

Esta tabla almacena los datos asociados al menú que se mostrara en la interfaz gráfica del usuario estas son: las direcciones URL, el nombre de los enlaces, iconos, entre otros. (Véase Ilustración 7).

pt_menu											
Tabla para almacenar los datos para construir el menú dinámico											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_Menu	INT(11)	✓	✓					✓		ID del menú	
idpt_Perfil	INT(11)		✓							id del perfil del asociado	
idNodoPadre	INT(11)		✓						'0'	Nodo padre, si es un nodo raíz este valor es cero	
nombre	VARCHAR(30)		✓							Nombre o texto del nodo	
descripcion	VARCHAR(200)									Breve descripción del nodo del menú	
targetUrl	VARCHAR(1000)								NULL	Url a la que se debe redirigir cuando se dé click sobre este nodo del menú	
icono	VARCHAR(100)								NULL	nombre de la imagen asociada a este nodo del menu, solo los nodos raiz tienen imagen	
posicion	INT(11)		✓							Indica la posición u orden en que se va a ubicar el nodo en el menú	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizó la transacción	

Ilustración 7. Diccionario de datos tabla pt_menu

1.8. Tabla pt perfil

Esta tabla almacena los tipos de perfiles que se usaran en la aplicación, estos son: estudiante, tutor y administrador. (Véase Ilustración 8).

pt_perfil											
Datos del perfil de usuario											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_Perfil	INT(11)	✓	✓					✓		Id del perfil	
nombre	VARCHAR(45)		✓							Nombre del perfil	
descripcion	VARCHAR(200)									Descripción breve del perfil	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizó la transacción	

Ilustración 8. Diccionario de datos tabla pt_perfil

1.9. Tabla pt proyectos

Esta tabla almacena los datos básicos de un proyecto como son: el título, la descripción, entre otros. (Véase Ilustración 9).

pt_proyectos											
Almacena los datos de los proyectos											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_proyectos	INT(11)	✓	✓					✓		Id del proyecto	
nombre	VARCHAR(500)		✓							Nombre del proyecto	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	
cerrado	BIT(1)		✓							Indica si el proyecto esta abierto 0 o cerrado 1.	

Ilustración 9. Diccionario de datos tabla pt_proyectos

1.10. Tabla pt proyectos integrantes

Esta tabla almacena los estudiantes que integran un proyecto. (Véase Ilustración 10).

pt_proyectos_integrantes											
Almacena los integrantes de un proyecto											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_proyectos_integrantes	INT(11)	✓	✓					✓		Id del registro	
idpt_proyectos	INT(11)		✓							Id del proyecto	
idpt_Usuarios	INT(11)		✓							Id del estudiante o usuario	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 10. Diccionario de datos tabla pt_proyectos_integrantes

1.11. Tabla pt proyectos tutores

Esta tabla almacena los proyectos que pertenecen a un tutor. (Véase Ilustración 11).

pt_proyectos_tutores										
Almacena la relación de los tutores y proyectos										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_proyectos_tutores	INT(11)	✓	✓							Id del proyecto
idpt_Usuarios	INT(11)	✓	✓							Id del tutor o usuario

Ilustración 11. Diccionario de datos tabla pt_proyectos_tutores

1.12. Tabla pt tarea

Esta tabla almacena todos los datos asociados a una tarea de un proyecto, los datos pueden ser nombre de la tarea, fecha de inicio y finalización de la misma. (Véase Ilustración 12).

pt_tarea										
Almacena las tareas de un proyecto										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_tarea	INT(11)	✓	✓					✓		Id de la tarea
titulo	VARCHAR(500)		✓							Título de la tarea
descripcion	VARCHAR(3000)		✓							Descripción de la tarea
comentarios	VARCHAR(500)									Comentarios de la tarea
idpt_proyectos	INT(11)		✓							Id del proyecto
fechaInicio	DATETIME		✓							Fecha de inicio de la tarea
fechaFinal	DATETIME		✓							Fecha final de la tarea
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizó la transacción

Ilustración 12. Diccionario de datos tabla pt_tarea

1.13. Tabla pt tarea archivos

Esta tabla almacena los posibles archivos adjuntos asociados a una tarea de un proyecto. (Véase Ilustración 13).

pt_tarea_archivos											
Almacena los datos de los archivos adjuntos de las tareas											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_archivos	INT(11)	✓	✓					✓		Id del registro	
idpt_tarea	INT(11)		✓							Id de la tarea	
nombre	VARCHAR(1000)		✓							Nombre del archivo	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 13. Diccionario de datos tabla pt_tarea_archivos

1.14. Tabla pt tarea avances

Esta tabla almacena los avances de las tareas de un proyecto. (Véase Ilustración 14).

pt_tarea_avances											
Almacena los avances de la tarea											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_avances	INT(11)	✓	✓					✓		Id del avance	
idpt_tarea	INT(11)		✓							Id de la tarea	
titulo	VARCHAR(500)		✓							Título del avance	
descripcion	VARCHAR(3000)		✓							Descripción del avance	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 14. Diccionario de datos tabla pt_tarea_avances

1.15. Tabla pt tarea avances archivos

Esta tabla almacena los datos adjuntos de la tarea de un proyecto. (Véase Ilustración 15).

pt_tarea_avances_archivos											
Almacena los datos de los archivos adjuntos de los avances de la tareas											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_avances_archivos	INT(11)	✓	✓					✓		Id del registro	
idpt_tarea_avances	INT(11)		✓							Id del avance	
nombre	VARCHAR(1000)		✓							Nombre del archivo	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 15. Diccionario de datos tabla pt_tarea_avances_archivos

1.16. Tabla pt tarea avances estatus

Esta tabla almacena los estados que ha tenido el avance durante el ciclo de la tarea.

(Véase Ilustración 16).

pt_tarea_avances_estatus											
Almacena el estatus de los avances de las tareas											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_avances_estatus	INT(11)	✓	✓					✓		Id del registro	
idpt_tarea_avances	INT(11)		✓							Id de la tarea	
idpt_estatus_avances	INT(11)		✓							Id del estatus del avance	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 16. Diccionario de datos tabla pt_tarea_avances_estatus

1.17. Tabla pt avances respuestas

Esta tabla guarda las respuestas de los avances de una tarea. (Véase Ilustración 17).

pt_tarea_avances_respuestas											
Almacena las respuestas de los avances de las tareas											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_avances_respuestas	INT(11)	✓	✓					✓		Id de la respuesta del avance	
idpt_tarea_avances	INT(11)		✓							Id del avance	
respuesta	VARCHAR(5000)		✓							Descripción de la respuesta	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 17. Diccionario de datos tabla pt_tarea_avances_respuestas

1.18. Tabla pt tarea avances respuestas archivos

Esta tabla almacena los datos adjuntos de los avances de las tareas de un proyecto.

(Véase Ilustración 18).

pt_tarea_avances_respuestas_archivos											
Almacena los datos de los archivos adjuntos de las respuestas de los avances											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_avances_respuestas_archivos	INT(11)	✓	✓					✓		Id del registro	
idpt_tarea_avances_respuestas	INT(11)		✓							Id de la respuesta	
nombre	VARCHAR(1000)		✓							Nombre del archivo	
activo	BIT(1)		✓							Indica la visibilidad del registro	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro	
idUsuarioTransaccion	VARCHAR(45)		✓							Id del usuario que realizo la transacción	

Ilustración 18. Diccionario de datos tabla pt_tarea_avances_respuestas_archivos

1.19. Tabla pt tarea estatus

Esta tabla almacena todos los estados que puede tener una tarea durante el ciclo de vida del proyecto. (Véase Ilustración 19).

pt_tarea_estatus											
Almacena el estatus de la tarea y el porcentaje de avance.											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idpt_tarea_estatus	INT(11)	✓	✓					✓		Id del registro	
idpt_tarea	INT(11)		✓							Id de la tarea	
idpt_estatus	INT(11)		✓							Id del estatus	
porcentaje_avance	INT(11)		✓						'0'	Porcentaje de avance de la tarea	
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro	
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción	

Ilustración 19. Diccionario de datos tabla pt_tarea_estatus

1.20. Tabla pt tarea integrantes

Esta tabla almacena los integrantes o estudiantes asociados a las tareas del proyecto.

(Véase Ilustración 20).

pt_tarea_integrantes										
Almacena los integrantes de una tarea										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_tarea_integrantes	INT(11)	✓	✓					✓		Id del registro
idpt_tarea	INT(11)		✓							Id de la tarea
idpt_Usuarios	INT(11)		✓							Id del estudiante o usuario
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción

Ilustración 20. Diccionario de datos tabla pt_tarea_integrantes

1.21. Tabla pt usuario perfil

Esta tabla almacena los perfiles que puede tener el usuario en la aplicación. (Véase Ilustración 21).

pt_usuario_perfil										
Tabla para asociar el usuario a un perfil										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Usuario_perfil	INT(11)	✓	✓					✓		Id del registro auto numérico
idpt_Perfil	INT(11)		✓							id del perfil
idpt_Usuarios	INT(11)		✓							id del usuario al que se le asigna el perfil
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción

Ilustración 21. Diccionario de datos tabla pt_usuario_perfil

1.22. Tabla pt usuarios

Esta tabla almacena los datos de los usuarios de la aplicación como son nombres, apellidos, programa al que pertenece, nombre de usuario, claves, entre otros. (Véase Ilustración 22).

pt_usuarios										
Almacena los datos de los usuarios de la aplicación										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idpt_Usuarios	INT(11)	✓	✓					✓		id del Usuario
idpt_Facultad_programas	INT(11)		✓							Id del programa de una facultad a la que pertenece un estudiante
num_Identificacion	VARCHAR(45)		✓							Número de identificación
nombres	VARCHAR(100)		✓							Nombres del usuario
apellidos	VARCHAR(100)		✓							Apellidos del usuario
usuario	VARCHAR(45)		✓							Nombre de usuario de la aplicación
clave	VARCHAR(45)		✓							Clave de usuario
mail	VARCHAR(200)		✓							dirección de correo electrónico del usuario
telefono1	VARCHAR(45)		✓							Teléfono 1 del usuario
telefono2	VARCHAR(45)		✓							Teléfono 2 del usuario
primerIngreso	BIT(1)		✓						b'1'	Indica si el usuario ingresa a la aplicación por primera vez
activo	BIT(1)		✓							Indica la visibilidad del registro
fechaCreacion	DATETIME		✓							Fecha en que se creó el registro
fechaModificacion	DATETIME		✓							Fecha en que se modificó el registro
idUsuarioTransaccion	INT(11)		✓							Id del usuario que realizo la transacción

Ilustración 22. Diccionario de datos tabla pt_usuarios

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Trabajo dirigido

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

UNIVERSIDAD EAN

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS



Proyecto de grado

Aplicación web para el control y seguimiento de avances del proyecto de grado

“Project Tracker”

Tutor

Ingeniero Alexander García Pérez

Por

José Ricardo Restrepo Escobar

Bogotá, Colombia

2012

ANEXO 8

CONFIGURACIÓN PARA LA INSTALACIÓN DE LA APLICACIÓN

- **Procedimiento**

La aplicación debe ser publicada en un sistema operativo Windows, la máquina que va funcionar como Host debe tener instalado el Framework 4.0, el IIS 6.0 y MySQL como motor de base de datos, luego se debe restaurar un backup de la base de datos y debe estar creado por defecto un administrador de la aplicación.

Dentro de la aplicación web se debe modificar el archivo Webconfig y se debe ingresar los datos asociados a la cadena de conexión.

Para el envío de correos se debe crear una cuenta de correo, en el mismo archivo de Webconfig se debe agregar la dirección de correo, clave y servicio SMTP

Por parte de usuario final este debe tener instalado adobe Flash, Reader y los navegadores anteriormente citados.

A continuación se describe como crear el usuario administrador en directamente sobre la base de datos.

- Abrir la base de datos ProjectTracker y la tabla pt_usuarios
- Agregar un usuario con los siguientes datos.

idpt_Usuarios= Es auto numérico.

idpt_Facultad_programas = Id del programa de una facultad a la que pertenece un estudiante, para el caso de administrador se ingresa 17 el cual es un código genérico.

num_Identificacion= Número de identificación.

Nombres = Nombres del usuario.

Apellidos = Apellidos del usuario.

Usuario = Nombre de usuario de la aplicación.

Clave = Clave de usuario.

Mail = dirección de correo electrónico del usuario.

telefono1 = Teléfono 1 del usuario.

telefono2 = Teléfono 2 del usuario.

PrimerIngreso = 1.

Activo = 1.

FechaCreacion = Ingresar la fecha del día.

FechaModificacion = Ingresar la fecha del día.

IdUsuarioTransaccion = Ingresar 0.

- Después de agregar el usuario se debe agregar el perfil de administrador.

Para esto se debe abrir la tabla pt_usuario_perfil.

- Se debe agregar los siguientes datos:

idpt_Usuario_perfil = auto numérico.

pt_perfil = se debe ingresar el valor 1.

idpt_Usuarios = id del usuario que se creó en la tabla pt_usuarios.

activo = 1.

FechaCreacion = Ingresar la fecha del día.

FechaModificacion = Ingresar la fecha del día.

IdUsuarioTransaccion = Ingresar 0.

Luego del anterior procedimiento se podrá ingresar a la aplicación con rol de administrador.

LICENCIA DE USO – AUTORIZACIÓN DE LOS AUTORES

Actuando en nombre propio identificado (s) de la siguiente forma:

Nombre Completo Jose Ricardo Restrepo Escobar

Tipo de documento de identidad: C.C. T.I. C.E. Número: 00051283

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

El (Los) suscrito(s) en calidad de autor (es) del trabajo de tesis, monografía o trabajo de grado, documento de investigación, denominado:

Aplicación web para el control y seguimiento de avances del proyecto de grado "Project Tracker"

Dejo (dejamos) constancia que la obra contiene información confidencial, secreta o similar: SI NO
(Si marqué (marcamos) SI, en un documento adjunto explicaremos tal condición, para que la Universidad EAN mantenga restricción de acceso sobre la obra).

Por medio del presente escrito autorizo (autorizamos) a la Universidad EAN, a los usuarios de la Biblioteca de la Universidad EAN y a los usuarios de bases de datos y sitios webs con los cuales la Institución tenga convenio, a ejercer las siguientes atribuciones sobre la obra anteriormente mencionada:

- A. Conservación de los ejemplares en la Biblioteca de la Universidad EAN.
- B. Comunicación pública de la obra por cualquier medio, incluyendo Internet
- C. Reproducción bajo cualquier formato que se conozca actualmente o que se conozca en el futuro
- D. Que los ejemplares sean consultados en medio electrónico
- E. Inclusión en bases de datos o redes o sitios web con los cuales la Universidad EAN tenga convenio con las mismas facultades y limitaciones que se expresan en este documento
- F. Distribución y consulta de la obra a las entidades con las cuales la Universidad EAN tenga convenio

Con el debido respeto de los derechos patrimoniales y morales de la obra, la presente licencia se otorga a título gratuito, de conformidad con la normatividad vigente en la materia y teniendo en cuenta que la Universidad EAN busca difundir y promover la formación académica, la enseñanza y el espíritu investigativo y emprendedor.

Manifiesto (manifestamos) que la obra objeto de la presente autorización es original, el (los) suscritos es (son) el (los) autor (es) exclusivo (s), fue producto de mi (nuestro) ingenio y esfuerzo personal y la realizo (zamos) sin violar o usurpar derechos de autor de terceros, por lo tanto la obra es de exclusiva autoría y tengo (tenemos) la titularidad sobre la misma. En vista de lo expuesto, asumo (asumimos) la total responsabilidad sobre la elaboración, presentación y contenidos de la obra, eximiendo de cualquier responsabilidad a la Universidad EAN por estos aspectos.

En constancia suscribimos el presente documento en la ciudad de Bogotá D.C.,

NOMBRE COMPLETO: <u>José Iván Restrepo F</u>	NOMBRE COMPLETO: _____
FIRMA: <u>[Firma]</u>	FIRMA: _____
DOCUMENTO DE IDENTIDAD: <u>80051203</u>	DOCUMENTO DE IDENTIDAD: _____
FACULTAD: <u>Ingeniería</u>	FACULTAD: _____
PROGRAMA ACADÉMICO: <u>Sistemas</u>	PROGRAMA ACADÉMICO: _____

NOMBRE COMPLETO: _____	NOMBRE COMPLETO: _____
FIRMA: _____	FIRMA: _____
DOCUMENTO DE IDENTIDAD: _____	DOCUMENTO DE IDENTIDAD: _____
FACULTAD: _____	FACULTAD: _____
PROGRAMA ACADÉMICO: _____	PROGRAMA ACADÉMICO: _____

Fecha de firma: 12 diciembre 2012