

Universidad de Cantabria – Facultad de Ciencias
Ingeniería en Informática
Ingeniería del Software I

Ejemplo Completo de Análisis y Diseño

Este documento es un extracto formado por algunas partes del Proyecto Fin de Carrera desarrollado por el alumno Francisco Novillo Pérez en el año 2007 en la Escuela Superior de Informática de Ciudad Real.

Es un ejemplo de la aplicación del Proceso Unificado de Desarrollo (PUD) a un proyecto para desarrollar un “sistema de gestión y realización vía web de experimentos y encuestas”, orientado a su aplicación en investigaciones experimentales en ingeniería del software y otros campos afines.

Las partes incluidas contienen:

- Los objetivos del proyecto.
- La aplicación del PUD, con las diversas fases e iteraciones realizadas, y los resultados de requisitos, análisis y diseño para los principales casos de uso abordados.
- Un anexo con el resto de casos de uso.

2. - OBJETIVOS

Este proyecto pretende elaborar un entorno para la creación de experimentos y encuestas que además debe proporcionar el soporte necesario a su realización por parte de los sujetos y al análisis final de resultados. Con este propósito, se implementará una herramienta con los siguientes objetivos principales:

- Proporcionar el soporte necesario que permita la definición de experimentos o encuestas a partir de la especificación de las actividades a realizar y las preguntas y tipo de respuestas (si/no, abiertas, tipo test...) necesarias.
- Generar automáticamente un sitio web en el que el usuario pueda realizar el experimento o la encuesta a partir de la especificación de ciertos parámetros tales como el nombre, las instrucciones, los modelos o las tareas.
- Almacenar los resultados en distintos formatos y proporcionar un posterior análisis de los mismos.

Se tratará de una herramienta con interfaz Web para facilitar el acceso e interoperabilidad y debe ser flexible, es decir, debe permitir la definición de cualquier tipo de experimento o encuesta. Debe permitir a su vez la visualización de los resultados obtenidos.

El entorno de desarrollo será la plataforma .NET de Microsoft, se utilizará el paquete de desarrollo de Visual Studio .NET 2005, en particular Visual Basic.NET y ASP.NET 2.0.

Para almacenar las definiciones de los experimentos y las respuestas de los sujetos que los realizan se utilizará el Sistema Gestor de Bases de Datos Microsoft SQL Server, en su versión 2005. Para mostrar los resultados de los experimentos se hará uso de la herramienta Crystal Reports integrada en el paquete Visual Studio .NET 2005.

En general, todos estos objetivos coinciden con los requisitos funcionales y de entorno que se especifican en el apartado 5.2.2, dentro de la fase de requisitos.

5.- DESARROLLO DE LA HERRAMIENTA *EMPIRICAL-WEBGEN*

En este apartado se aborda todo el proceso de desarrollo de la herramienta objetivo del presente proyecto. Ya que para ello se ha seguido el *PUD*, la información quedará estructurada según el esquema típico que sigue este proceso y que se ha comentado en el apartado 4.1.

5.1. Plan de Iteraciones de Empirical-WebGen

En este apartado se presentan de forma resumida las iteraciones realizadas para el desarrollo de la herramienta *Empirical-WebGen* en el contexto de las fases del proceso unificado que se han descrito anteriormente.

❖ INICIO

➤ Iteración 1

En esta primera iteración, el flujo de trabajo que más se desarrollo fue el de requisitos. Los pasos que se realizaron en esta iteración fueron los siguientes:

- Se realizó una lista a priori de los **requisitos** que debía reunir el sistema en relación a su funcionalidad.
- Se fijaron las características relativas al diseño de un experimento.
- Se identificaron los casos de uso “Definir Encuesta o Experimento” y “Realizar Encuesta o Experimento”. Estos son los casos de uso más importantes del proyecto; el primero por dar soporte a la creación de nuevas encuestas y experimentos, y el segundo por ser el encargado de la generación, control y almacenamiento del Sitio Web que representa una encuesta o experimento.
- Se estudió la viabilidad del sistema en relación a la tecnología y los medios disponibles.
- Se realizó el **análisis** del sistema mediante el modelo de casos de uso.

- Respecto a los flujos de **implementación** y **pruebas**, se construyó un prototipo de un experimento en local para gestionar los riesgos más importantes como el de capturar las respuestas y la interacción con el usuario.

➤ Iteración 2

- En esta iteración se matizaron algunos aspectos de los **requisitos** ya establecidos, como la agrupación de las tareas en los llamados “grupos de tareas” que a su vez componen los modelos.
- Se especificaron nuevos **requisitos** como la modificación de las encuestas y experimentos.
- Se realizaron los diagramas de clases e interacción de los casos de uso identificados en el flujo de **análisis**.
- En cuanto al **diseño**, se definieron las clases necesarias para la gestión de la información y se estudiaron las proporcionadas por ASP .NET 2.0 como la SqlDataSource que permite la extracción de datos de la base de datos.
- Se fue refinando la **implementación** de los casos de uso definidos en la primera iteración y se comenzó la implementación de la modificación de encuestas y experimentos, al tiempo que se realizaron las **pruebas** oportunas.

❖ ELABORACIÓN

➤ Iteración 3

Los pasos que se realizaron en esta iteración fueron los siguientes:

- Se recopilaron algunos **requisitos** que estaban pendientes, como la visualización de los resultados de los sujetos que realizan las encuestas o experimentos.
- Se realizó el **análisis** y **diseño** de los nuevos requisitos y se mejoraron los diagramas realizados en la iteración anterior.
- Se comenzó la **implementación** y **pruebas** de los nuevos requisitos y se continuó con la de los requisitos previos, incluyendo mejoras como nuevos

tipos de tareas, la captura del tiempo de realización de los grupos de tareas o la aleatorización de modelos, grupos de tareas y tareas.

➤ Iteración 4

- Se refinó el **requisito** de la visualización de resultados, distinguiendo entre informes de tiempos e informes de respuestas.
- Se identificaron los nuevos **requisitos** de registro y autenticación, distinguiendo entre los roles de usuario y administrador.
- Se realizó el **análisis** y **diseño** de los nuevos requisitos.
- Se fue refinando el **diseño** de los casos de uso identificados en iteraciones anteriores.
- Se prosiguió con la **implementación** y las **pruebas** de todas las funcionalidades del sistema.

❖ CONSTRUCCIÓN

➤ Iteración 5

- Se realizaron ligeros cambios en los **requisitos** ya establecidos, como la posibilidad de exportación de los informes en otros formatos, o el filtro de los usuarios por la dirección IP en la visualización de informes.
- Se realizaron los correspondientes cambios en los flujos de **análisis** y **diseño**.
- Se fue mejorando la **implementación**, teniendo siempre presente la arquitectura de las tres capas.
- Se realizaron las primeras **pruebas** unitarias y globales de la herramienta y se identificaron los problemas surgidos.

➤ Iteración 6

En esta iteración se realizaron principalmente algunas modificaciones en la **implementación** y las **pruebas** necesarias para que la herramienta funcionara correctamente.

❖ TRANSICIÓN

➤ Iteración 7

Por último se realizaron **pruebas** de la herramienta tanto a nivel local como a nivel remoto, instalando la aplicación en un servidor. Además de las pruebas de funcionalidad, se llevaron a cabo pruebas para medir el estrés de la aplicación cuando varios usuarios utilizan el sistema al mismo tiempo. (Véase apartado 5.3.3.4).

5.2. Inicio

El objetivo global de la fase de inicio es poner en marcha el proyecto. Para ello habrá que delimitar el ámbito del sistema propuesto para determinar la funcionalidad que tiene que cubrir el proyecto y conocer los límites dentro de los cuales buscar los riesgos en cuanto a la viabilidad del mismo. Así pues, el flujo de trabajo en el que se concentra la mayor parte del esfuerzo que se lleva a cabo en esta fase es el de requisitos, que se describirá a continuación.

5.2.1. Modelo del Dominio

En primer lugar, es necesario tener un conocimiento claro de la estructura de un experimento o encuesta genérica. Esto queda reflejado en el modelo del dominio (Figura 5.1).

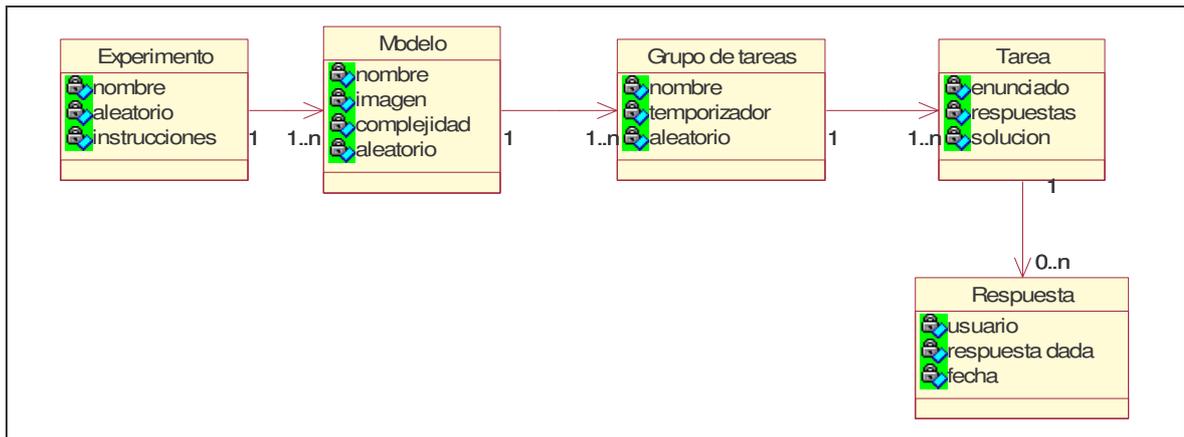


Figura 5.1. Modelo del Dominio

- Un **experimento** genérico tendrá un nombre y unas instrucciones, y constará de uno o más modelos.
- Los **modelos** están formados por uno o más grupos de tareas y se mostrarán en una página Web con su imagen (opcional) y su nombre. El orden en el que aparecen los modelos puede ser el mismo en el que se diseñaron o puede ser aleatorio si así se ha especificado en el experimento. Además, un modelo podrá incluir una pregunta de valoración de la complejidad si se desea.
- Los **grupos de tareas** constan de una o más tareas y al igual que los modelos pueden aparecer de forma aleatoria si así se especifica en el diseño del modelo. También tendrán un nombre y la opción de tomar el tiempo que tarda un usuario en contestar al grupo de tareas.
- Las **tareas** tendrán un enunciado y, dependiendo del tipo (Si/No, Verdadero/Falso, Elección Múltiple, Respuesta Abierta o Ejercicio) pueden tener las posibles respuestas así como la solución.
- Por último, las tareas estarán asociadas a las **respuestas** que den los usuarios en un momento concreto.

5.2.2. Requisitos

Este es el flujo de trabajo en el que se hace mayor hincapié en la fase de inicio del proyecto. Se averiguarán qué actores interactúan con el sistema y qué servicios les ofrece éste. De esta manera se podrá tener una visión global del alcance del sistema.

5.2.2.1- Requisitos Candidatos

De manera preliminar, debe elaborarse una lista inicial que enumere las características que debe tener el sistema. Dicha lista evolucionará dinámicamente a lo largo de la vida del proyecto según se incorporen nuevas ideas.

A continuación se enumeran una serie de ideas concebidas *a priori* sobre las características que debe reunir el sistema:

- ❖ Se debe dar soporte al diseño de cualquier tipo de experimento y encuesta. Ha de ser por tanto un sistema flexible.
- ❖ Se debe generar un sitio web para cada experimento o encuesta que permita su ejecución por parte de cualquier persona previamente registrada.
- ❖ Los resultados de las personas que realizan las encuestas o los experimentos se deben almacenar en la base de datos.
- ❖ Soporte para la visualización de los resultados obtenidos.
- ❖ Los elementos que se definan han de poder modificarse.
- ❖ Se deben soportar distintos roles de seguridad para limitar el acceso a algunas funcionalidades.

También merecen especial atención los siguientes requisitos no funcionales:

- ❖ Debe ser una herramienta con interfaz Web para facilitar su acceso e interoperabilidad.
- ❖ Debe desarrollarse usando el entorno Microsoft .NET, en concreto los lenguajes ASP .NET 2.0 y Visual Basic .NET.

5.2.2.2- Modelo de Casos de Uso

La Figura 5.2 muestra gráficamente el modelo de casos de uso del sistema.

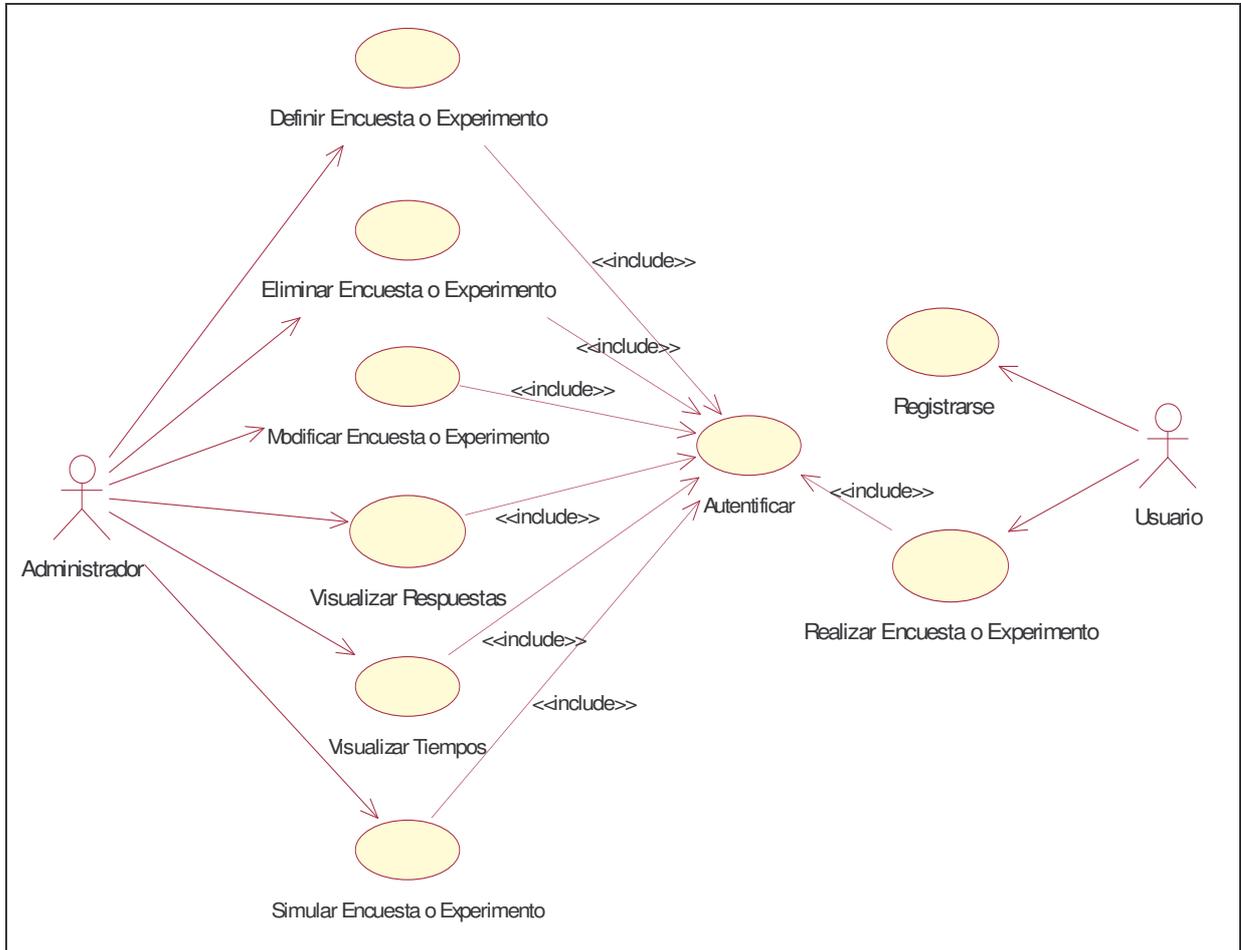


Figura 5.2. Modelo de casos de uso

Todos los casos de uso incluyen al de *Autenticar* como primer paso, excepto el de *Registrarse* que es la primera acción que debe realizar un *Usuario* para poder usar el sistema.

Para ilustrar el resto de fases del proceso unificado de desarrollo seguido se han escogido los tres casos de uso principales, a saber, *Definir Encuesta o Experimento*, *Realizar Encuesta o Experimento* y *Visualizar Tiempos*. El resto de casos de uso se encuentran detallados en el anexo A. A continuación se explican detalladamente estos elementos y los dos actores que intervienen en el modelo.

5.2.2.2.1- Actores

En este apartado se detallan las entidades que intervienen en el proceso.

- ❖ **Administrador.** Puede definir un nuevo experimento o nueva encuesta, modificarlos, eliminarlos, simularlos y visualizar los resultados.
- ❖ **Usuario.** Representa a las personas que una vez registradas pueden realizar encuestas o experimentos previamente definidos.

5.2.2.2.2- Casos de Uso

A continuación se describen con mayor detalle las características y comportamiento asociado con los tres casos de uso que capturan los requisitos funcionales fundamentales de la aplicación.

❖ **Definir Encuesta o Experimento**

El administrador utiliza este caso de uso para crear una nueva encuesta o experimento a través de la definición de unos parámetros.

Precondición.

El administrador debe indicar al sistema que quiere definir una nueva encuesta o experimento.

Flujo de sucesos.

Camino básico.

1. El administrador crea una nueva encuesta o experimento asignando un nombre, estableciendo las instrucciones y definiendo la aleatoriedad de los modelos que lo componen.
2. Crea un nuevo modelo estableciendo sus atributos.
3. Crea un nuevo grupo de tareas definiendo sus atributos.
4. Crea una nueva tarea.
5. El administrador puede crear nuevos modelos, grupos o tareas.

6. La instancia de caso de uso termina.

Caminos alternativos.

En los pasos 2, 3 y 4 puede volver a cualquiera de los pasos predecesores para modificar los atributos de la encuesta o experimento, del modelo, o del grupo de tareas.

Poscondición.

El nuevo experimento o encuesta queda creado.

❖ Visualizar Tiempos

El administrador indica al sistema que quiere mostrar los tiempos almacenados de una encuesta o experimento ya definida. El administrador podrá elegir unos parámetros opcionales.

Precondición.

La encuesta o experimento de la que se desea visualizar las respuestas debe haberse creado con anterioridad.

Flujo de sucesos.

Camino básico.

1. El administrador selecciona el informe de tiempos.
2. El administrador puede escoger entre mostrar el detalle de todos los usuarios que han realizado la encuesta o experimento o no mostrarlo.
3. Puede realizar un filtrado de los usuarios que han realizado la encuesta o experimento por fechas.
4. El sistema muestra el informe.
5. La instancia de caso de uso termina.

Caminos alternativos.

En el paso 4, el administrador puede exportar los resultados en varios formatos.

Poscondición.

Se muestra el informe de tiempos del experimento o encuesta seleccionada por el administrador.

❖ **Realizar Encuesta o Experimento**

Este caso de uso es similar al de “Simular Encuesta o Experimento” solo que en esta ocasión es el usuario el que realiza una de las encuestas o experimentos disponibles y todos los datos quedan almacenados en la base de datos.

Precondición.

La encuesta o el experimento debe estar definido previamente. El usuario, que debe estar registrado, tiene que indicar al sistema que quiere realizar dicha encuesta o experimento.

Flujo de sucesos.

Camino básico.

1. El usuario elige la encuesta o el experimento a realizar.
2. El sistema muestra una página de instrucciones que el usuario debe leer y aceptar para poder continuar.
3. El usuario realiza la encuesta o experimento contestando a las tareas que se le presentan.
4. El sistema muestra una página de agradecimiento.
5. La instancia de caso de uso termina.

Caminos alternativos.

En el paso 3, el usuario puede abortar la realización de la encuesta o experimento volviendo al estado inicial.

Poscondición.

La encuesta o experimento queda realizado y todos los datos guardados.

5.3. Elaboración y Construcción

Los principales objetivos de estas fases son:

- Recopilar la mayor parte de los requisitos que estén pendientes.
- Elaborar una arquitectura estable para guiar el desarrollo del sistema y su evolución en un futuro.
- Continuar vigilando los riesgos del proyecto.
- Al final de estas etapas deberá quedar una versión operativa inicial (versión beta).

Debido a la escasez, en cuanto a número, pero no en complejidad, de los casos de uso no queda justificado tratar de forma separada ambas fases. Por ello, y aprovechando la adaptabilidad del *PUD* se realizarán a la vez.

Dado que los requerimientos del sistema no sufren cambios sustanciales de los obtenidos en la fase de inicio, los flujos de trabajo a los que se prestará especial atención son el resto, es decir, análisis, diseño, implementación y pruebas.

5.3.1. Análisis

En esta fase se analizarán los casos de uso de mayor complejidad con el fin de alcanzar una arquitectura estable.

5.3.1.1. Modelo de Análisis

El presente *modelo de análisis* es una descripción de la arquitectura en la que se recogen los artefactos relevantes para establecer los cimientos sobre los cuales se apoyarán las siguientes iteraciones. Dichos artefactos serán realizaciones de casos de uso-análisis, clases de análisis o paquetes de análisis.

5.3.1.1.1. Realización de Casos de Uso-Análisis

❖ Definir Encuesta o Experimento

Diagrama de clases.

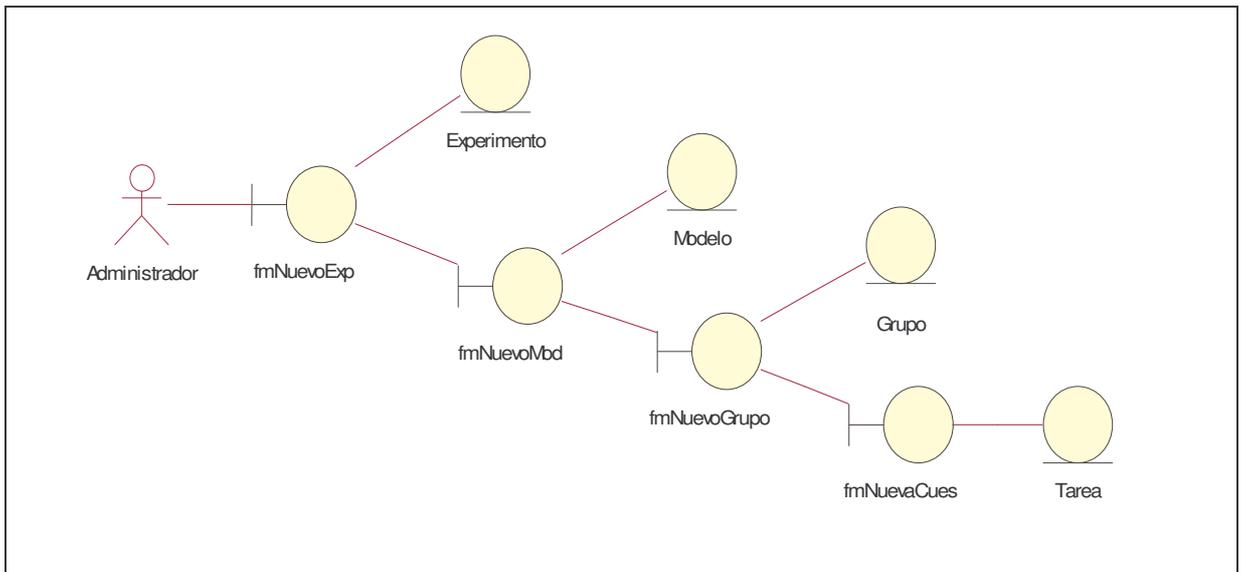


Figura 5.3. Clases del caso de uso-análisis Definir Encuesta o Experimento

Diagrama de colaboración.

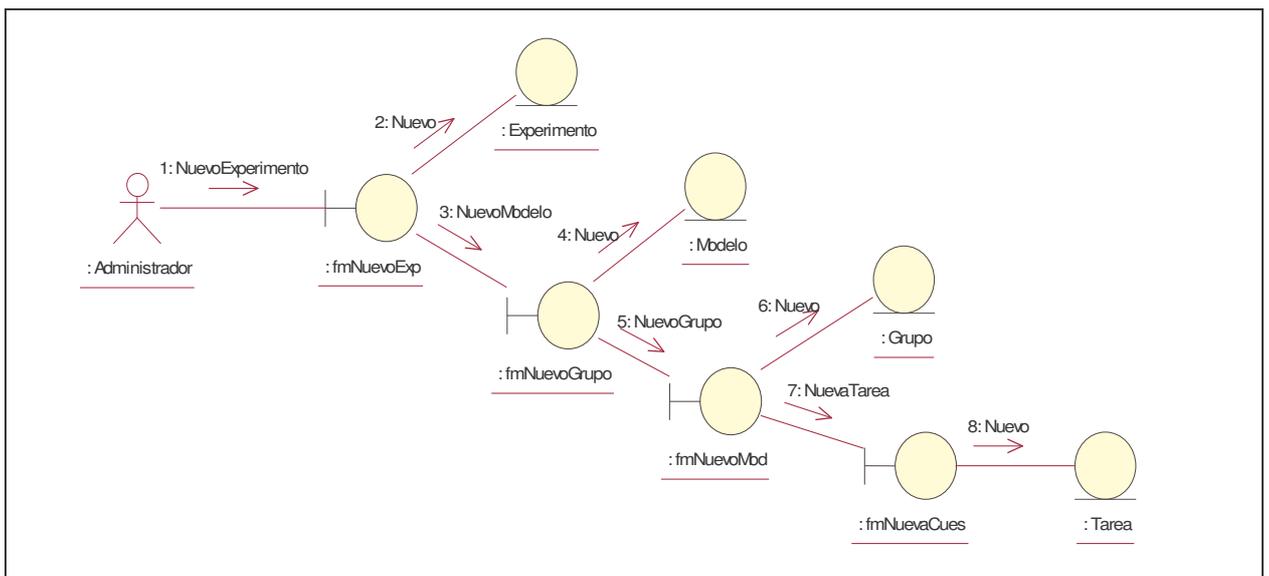


Figura 5.4. Interacción de las clases para el caso de uso Definir Encuesta o Experimento

Flujo de sucesos.

El *Administrador* decide crear una nueva encuesta o experimento e instancia el formulario *fmNuevoExp* (1). Configura los parámetros y se crea el nuevo experimento (2) cuando el *Administrador* crea el primer modelo (3). Se definen ahora los parámetros del modelo y se almacena (4) cuando se decide crear el primer grupo de tareas (5). Del mismo modo se establecen los parámetros del nuevo grupo y se crea (6) justo antes de instanciar el formulario *fmNuevaCues* (7). Por último se crea una nueva tarea (8).

❖ **Visualizar Tiempos**

Diagrama de clases.

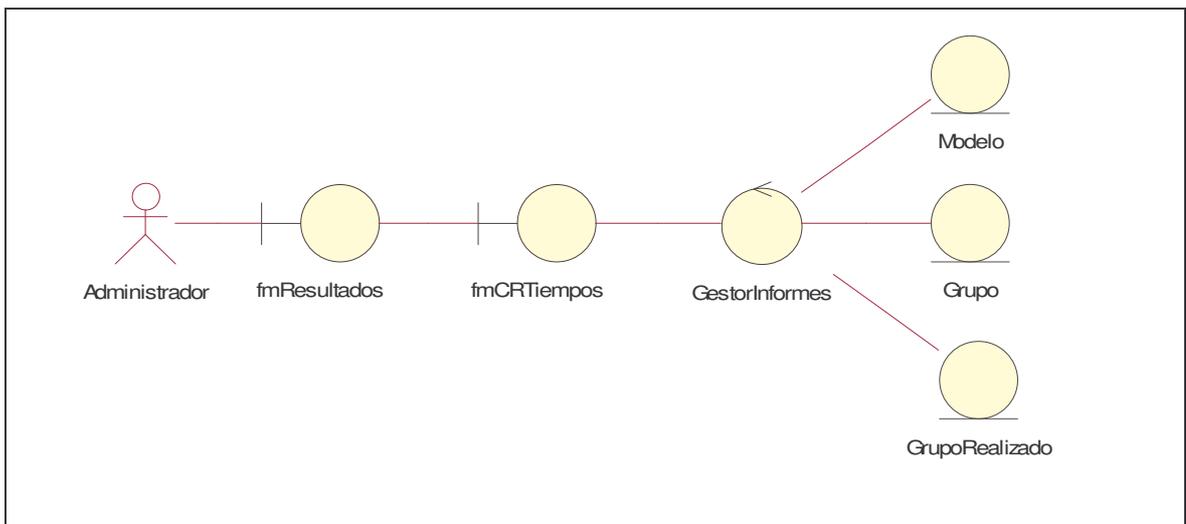


Figura 5.5. Clases del caso de uso Visualizar Tiempos

Diagrama de Colaboración.

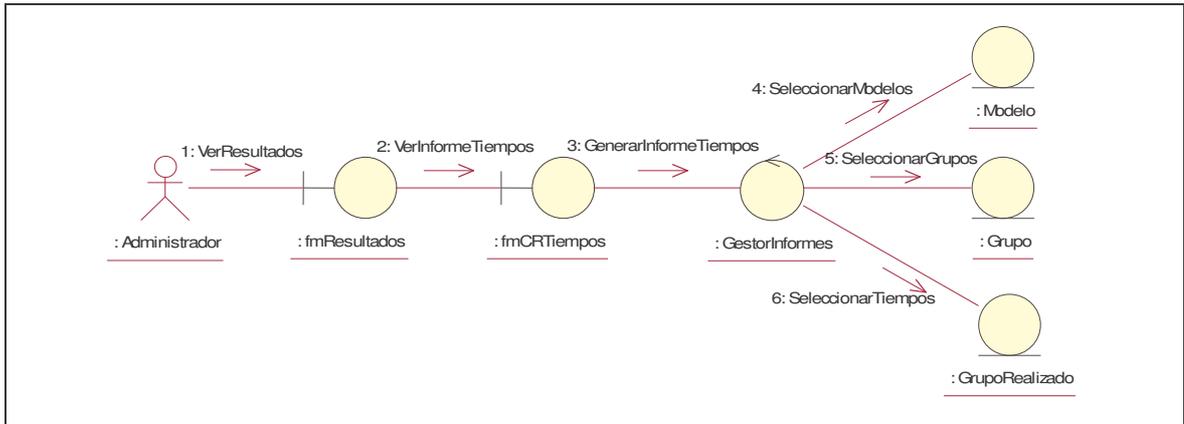


Figura 5.6. Interacción de las clases para el caso de uso Visualizar Tiempos

Flujo de sucesos.

El *Administrador* selecciona la encuesta o experimento del que desea ver los resultados (1). A continuación selecciona el tipo de informe “Tiempos” y configura algún parámetro si desea (2) para pedir después a la clase *GestorInformes* que genere el informe de tiempos (3). Esta clase seleccionará los modelos, grupos de tareas y tiempos empleados en los grupos del experimento de las respectivas tablas en la base de datos (4, 5, 6).

❖ Realizar Encuesta o Experimento

Diagrama de clases.

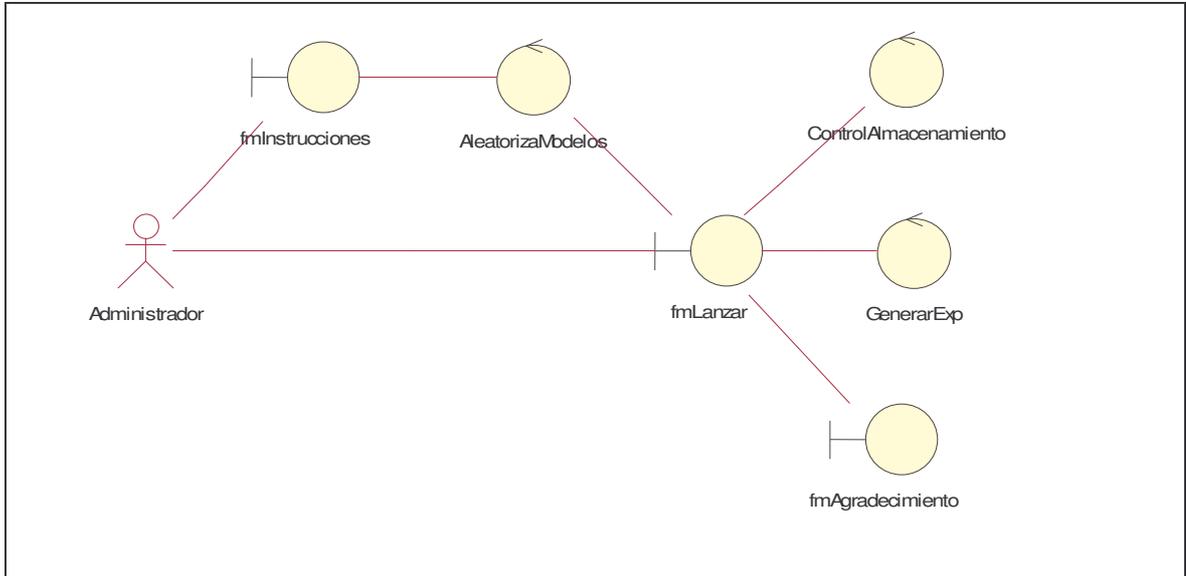


Figura 5.7. Clases del caso de uso-análisis Realizar Encuesta o Experimento

Diagrama de colaboración.

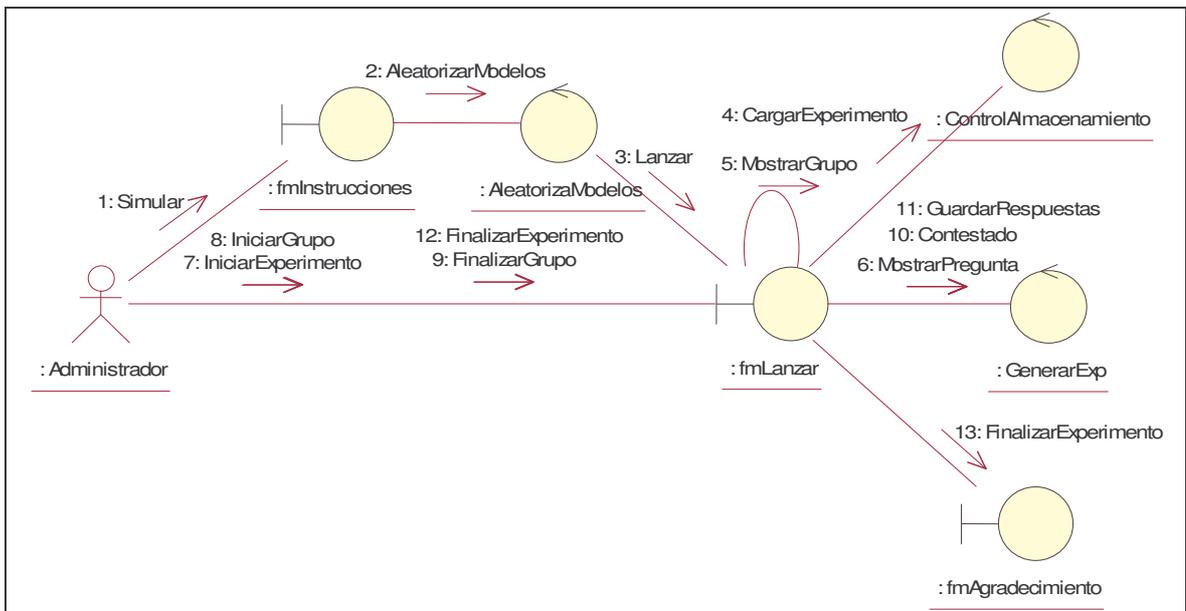


Figura 5.8. Interacción de las clases para el caso de uso Realizar Encuesta o Experimento

Flujo de sucesos.

El *Usuario* elige una encuesta o experimento para realizar y en primer lugar se instancia el formulario *fmInstrucciones* (1). Se aleatorizan los modelos si estuviese así diseñado el experimento (2) y se lanza dicho experimento (3). Se carga el experimento (4) y se muestran los grupos (5) con sus respectivas preguntas (6) aunque estarán ocultos hasta que el *Usuario* inicie el experimento (7) y pulse el botón de inicio de grupo (8). El *Usuario* realizará las tareas y cuando finalice pulsará el botón de fin de grupo (9). El sistema comprueba que se han contestado correctamente las tareas (10) en cuyo caso guardará las respuestas (11) y continuará mostrando el resto de grupos o modelos si los hubiera. Cuando el *Usuario* ha completado todas las tareas finalizará el experimento (12) y se mostrará una página de agradecimiento (13).

5.3.1.1.2. Clases de Análisis

En este apartado se describe el análisis que se ha realizado sobre las clases identificadas en el modelo de análisis. Los objetivos principales que se pretenden son:

- Identificar y mantener las responsabilidades de una clase del análisis, basadas en su papel en las realizaciones de los casos de uso.
- Identificar y mantener los atributos y relaciones de las clases del análisis.

En las siguientes tablas se muestra la información relativa a las clases identificadas en el modelo de análisis. En dichas tablas también aparecen las relaciones entre las distintas clases.

Clase	Usuario
Atributos	Login: cadena Password: cadena Nombre: cadena Apellido1: cadena Apellido2: cadena Tipo: cadena Estudios: cadena Curso: cadena Años_exp: entero Empresa: cadena
Responsabilidades	Gestionar toda la información relativa a los Usuarios registrados en el sistema.
Asociaciones	

Tabla 5.1. Clase Usuario

Clase	GenerarExp
Atributos	
Responsabilidades	Generar la página web que contiene una encuesta o experimento y contener la información necesaria para almacenar las respuestas dadas por los usuarios en dicha encuesta o experimento.
Asociaciones	

Tabla 5.2. Clase GenerarExp

Clase	AleatorizaModelos
Atributos	
Responsabilidades	Aleatorizar los modelos de un experimento en caso de que su diseño así lo exija.
Asociaciones	

Tabla 5.3. Clase AleatorizaModelos

Clase	Experimento
Atributos	Nombre: cadena Aleatorio: lógico Instrucciones: cadena
Responsabilidades	Gestionar toda la información relativa a las Encuestas y Experimentos.
Asociaciones	Posee una relación de agregación con la clase Modelo.

Tabla 5.4. Clase Experimento

Clase	Modelo
Atributos	Id_experimento: entero Nombre: cadena Archivo_url: cadena Orden: entero Complejidad: lógico Aleatorio: lógico
Responsabilidades	Gestionar la información relativa a un modelo.
Asociaciones	Tiene una relación de agregación con la clase Grupo.

Tabla 5.5. Clase Modelo

Clase	Grupo
Atributos	Id_modelo: entero Nombre: cadena Temporizador: lógico Aleatorio: lógico Orden: entero
Responsabilidades	Gestionar la información necesaria para el almacenamiento de los grupos de tareas.
Asociaciones	Posee una relación de agregación con la clase Tarea.

Tabla 5.6. Clase Grupo

Clase	Tarea
Atributos	Id_grupo: entero Enunciado: cadena Tipo: cadena Num_respuestas: entero Resp1: cadena Resp2: cadena Resp3: cadena Resp4: cadena Resp5: cadena Resp6: cadena Solucion: cadena Orden: entero
Responsabilidades	Gestionar la información relativa a una tarea.
Asociaciones	Tiene una relación de asociación con la clase Respuesta

Tabla 5.7. Clase Tarea

Clase	Respuesta
Atributos	Id_pregunta: entero Respuesta: entero Login: cadena Fecha: fecha y hora
Responsabilidades	Gestionar la información relativa a una respuesta.
Asociaciones	

Tabla 5.8. Clase Respuesta

5.3.2. Diseño

En esta fase, se realiza el diseño arquitectónico del sistema, seleccionando los casos de uso-análisis más representativos para conformar una línea base de la arquitectura.

5.3.2.1. Modelo de Diseño

5.3.2.1.1. Realizaciones de Caso de Uso-Diseño

❖ Definir Encuesta o Experimento

Diagrama de Clases

La principal novedad que incorpora este diagrama consiste en la inclusión de un gestor de almacenamiento llamado *Persistencia*, que se utilizará para desacoplar la capa de dominio de la capa de almacenamiento, como se verá en el apartado 5.3.2.1.2.

La misión de este gestor es la de servir de interfaz de comunicación de datos entre las distintas clases y la base de datos y evitar que sean éstas las que accedan directamente a los datos almacenados.

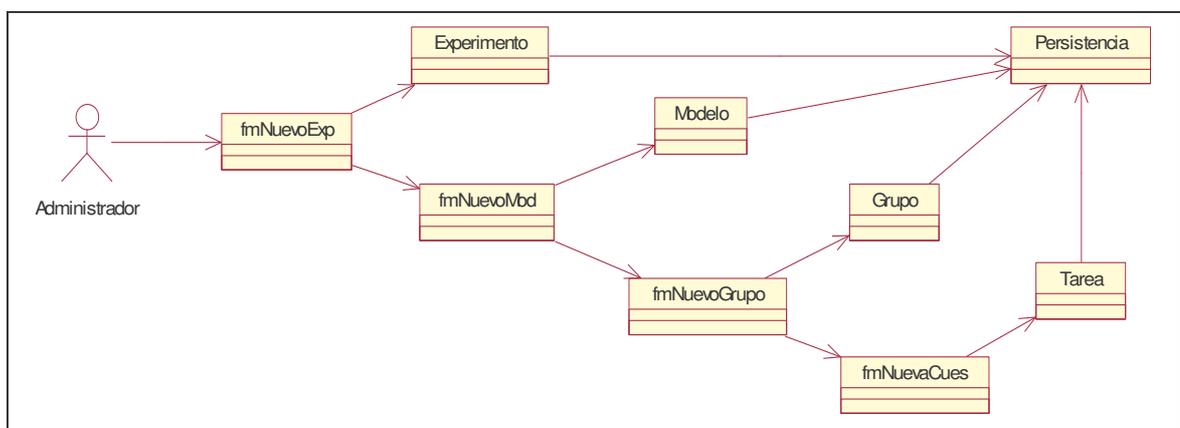


Figura 5.9. Clases del caso de uso-diseño Definir Encuesta o Experimento

Diagrama de Secuencia

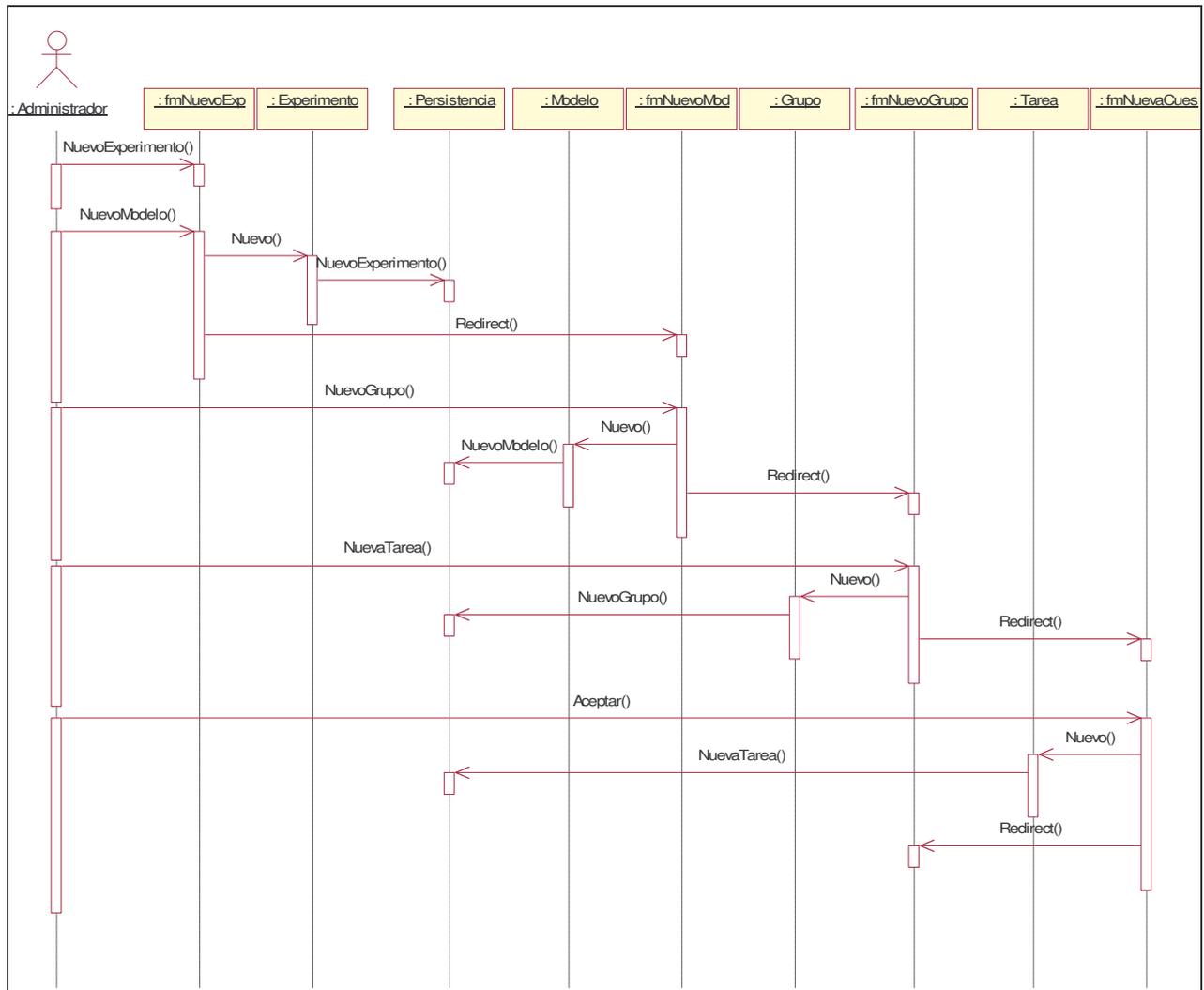


Figura 5.10. Diagrama de Secuencia del caso de uso Definir Encuesta o Experimento

❖ **Visualizar Tiempos**

Diagrama de Clases

Para el diseño de este caso de uso se han utilizado las clases *CrystalReportViewer*, *CrystalReportSource* y *Report*, pertenecientes al paquete *CrystalDecisions* de Visual Studio. Estas clases acceden a la base de datos a través de su propia interfaz sin necesidad del gestor *Persistencia*.

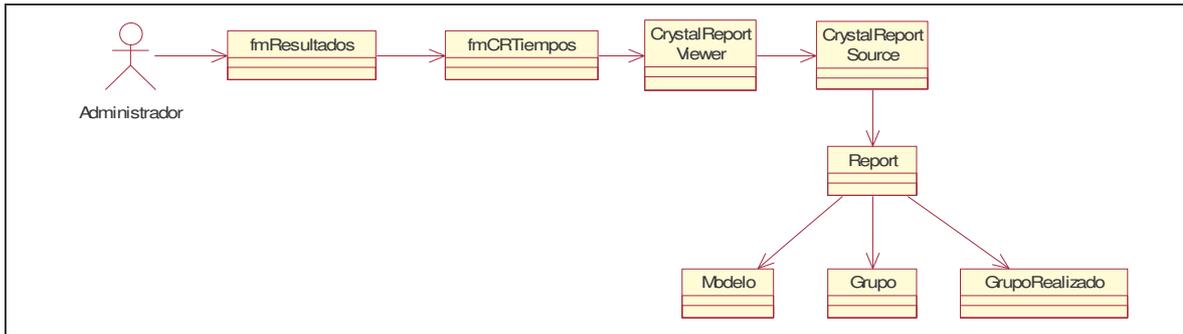


Figura 5.11. Clases del caso de uso-diseño Visualizar Tiempos

Diagrama de Secuencia

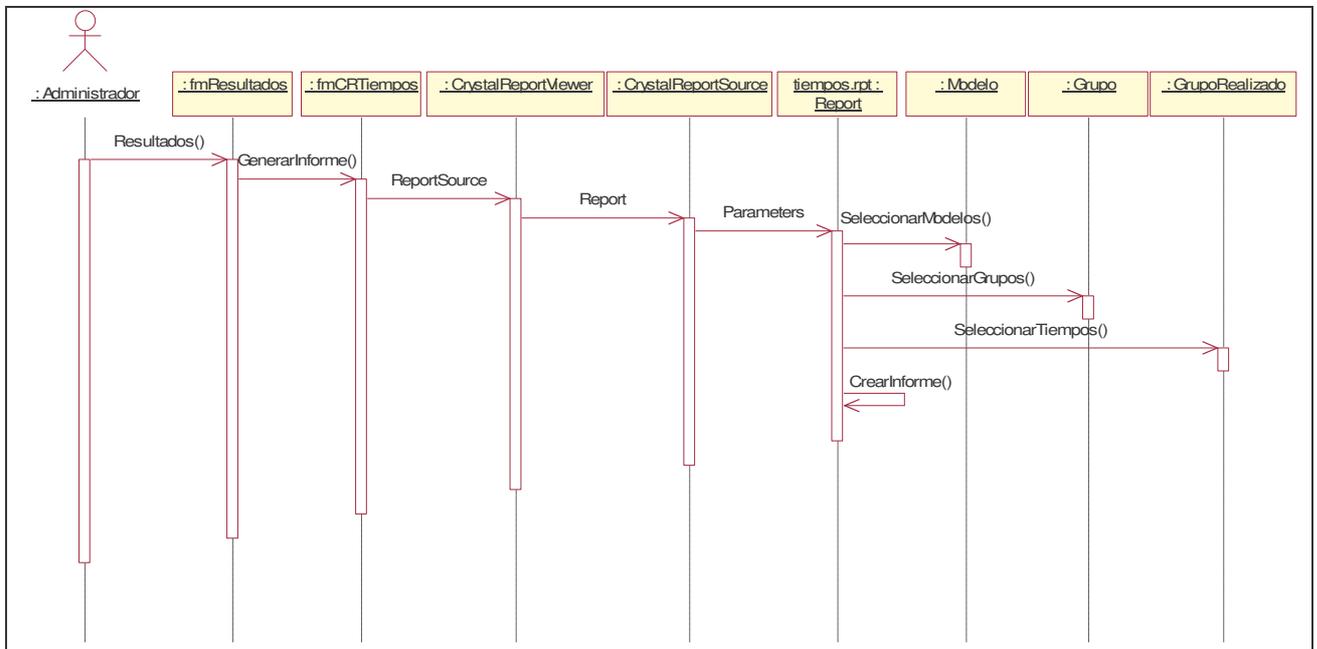


Figura 5.12. Diagrama de Secuencia del caso de uso Visualizar Tiempos

❖ Realizar Encuesta o Experimento

Diagrama de Clases

En este apartado se ha optado por omitir el gestor Persistencia para simplificar los diagramas de clases y de secuencia.

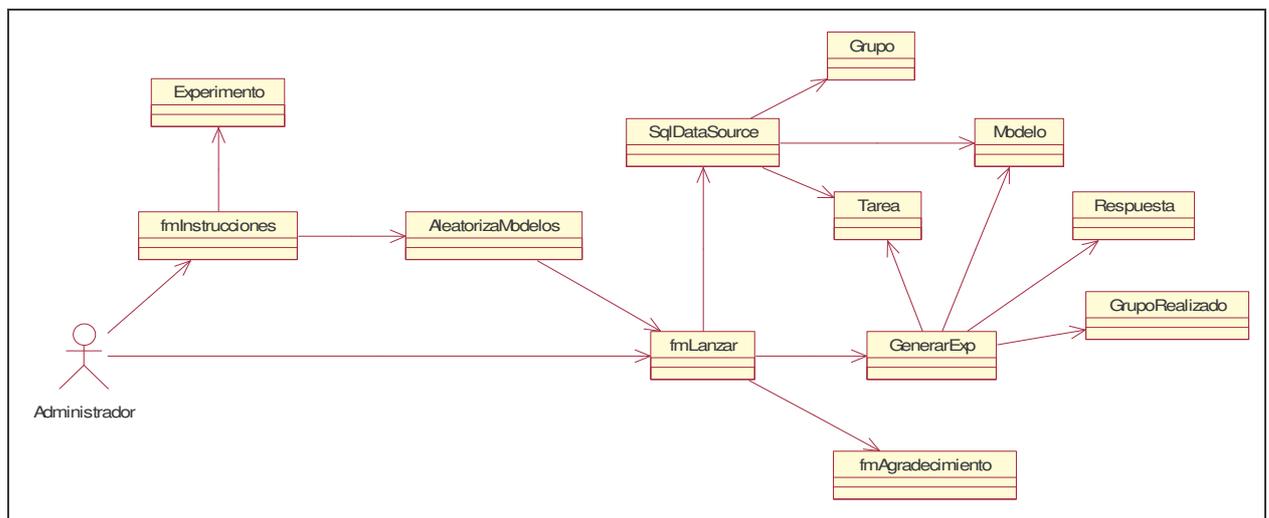


Figura 5.13. Clases del caso de uso-diseño Realizar Encuesta o Experimento

Diagrama de Secuencia

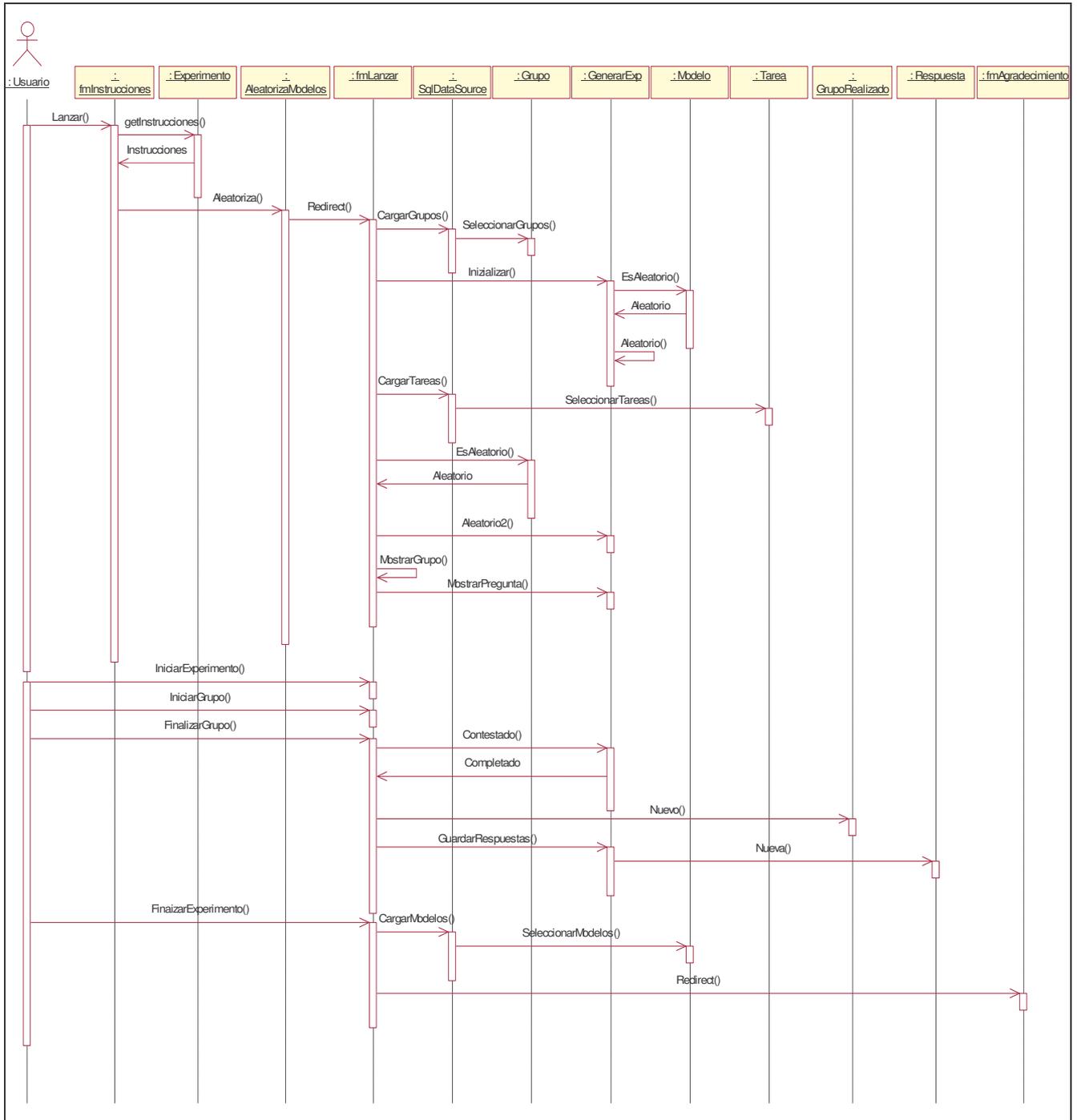


Figura 5.14. Diagrama de Secuencia del caso de uso Realizar Encuesta o Experimento

5.3.2.1.2. Arquitectura en Tres Capas

A la hora de diseñar la arquitectura de la herramienta, se ha considerado como la más adecuada la utilización de la arquitectura clásica de tres capas (Larman, 1998), debido a que al usar dicha arquitectura es posible aislar la lógica de la aplicación y convertirla en una capa intermedia bien definida. Estas capas verticales son:

❖ **Capa de presentación**, que proporciona la interfaz de la aplicación con el usuario final u otras aplicaciones y está formada por ventanas gráficas o clases que actúan como interfaz, para que el usuario u otra aplicación introduzca y reciba la información necesaria. No realiza apenas tareas de procesamiento, ya que se limitan a pasar y recibir información de las clases de la capa de dominio. En el caso de la herramienta *Empirical-WebGen*, se compone de una serie de formularios a través de los cuales los diferentes usuarios accederán a las diversas funcionalidades de la herramienta.

❖ **Capa de dominio**, o lógica empresarial, constituida por los elementos que efectúan las operaciones necesarias para el correcto funcionamiento de la aplicación. Se comunica tanto con la capa de presentación, como con la de almacenamiento para recibir información y enviar resultados. En el caso de *Empirical-WebGen*, todas las clases de procesamiento de datos y cálculo de resultados se sitúan en esta capa.

❖ **Capa de almacenamiento**, o de servicios de datos. En esta capa se encuentran las clases de la aplicación cuyo objetivo es la gestión del almacenamiento persistente. *Empirical-WebGen* utiliza la interfaz Persistencia para que sólo una clase acceda directamente al origen físico de los datos, favoreciendo así que el nivel de acoplamiento de la aplicación disminuya.

La arquitectura de tres niveles aísla cada uno de los principales elementos de funcionamiento, de forma que la presentación es independiente de las reglas de proceso y de la lógica empresarial que, a su vez, es independiente de los datos. Este modelo requiere muchas más tareas previas de análisis y diseño, pero reduce considerablemente los costos de mantenimiento y aumenta la flexibilidad funcional a largo plazo. La Figura 5.15 muestra la arquitectura de tres capas del diseño final de la herramienta:

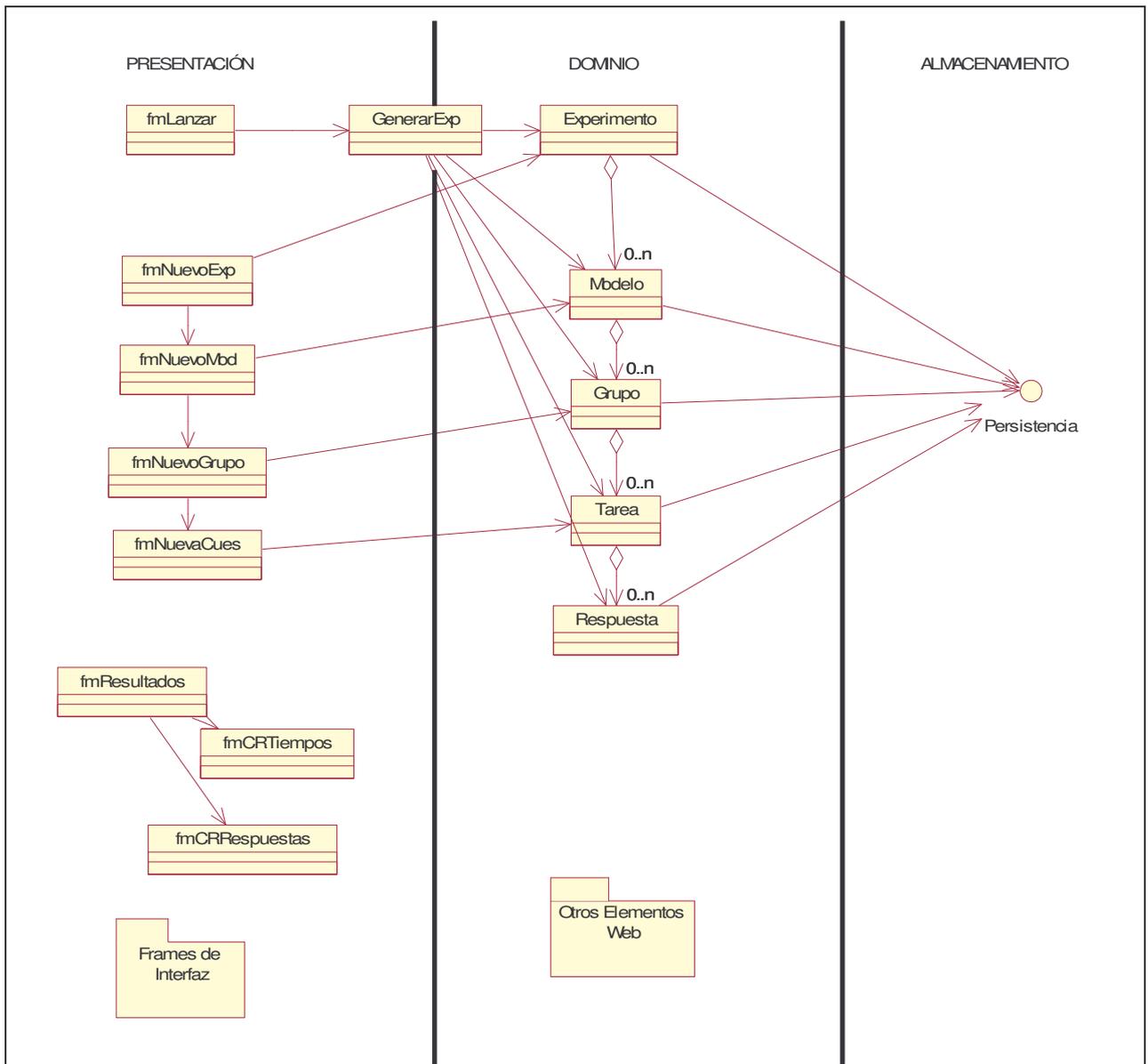


Figura 5.15. Arquitectura en Tres Capas de la Aplicación

La clase *GenerarExp* se sitúa entre la capa de presentación y la de dominio ya que además de realizar las gestiones oportunas para la carga de la encuesta o experimento, utiliza elementos de presentación para generar la parte del sitio Web que muestra dicha encuesta o experimento.

El paquete *Frames de Interfaz* al que se hace referencia en la capa de presentación incluye un conjunto de formularios o frames cuyo objetivo es el de servir de interfaz de comunicación directa y visual entre la aplicación y el usuario. Al no tener una especial

relevancia ni complejidad, se ha optado por no referenciarlos a todos de manera explícita y sí hacerlo con los otros formularios que se utilizan en las acciones más importantes de la aplicación.

El paquete Otros Elementos Web de la capa de Dominio hace referencia a las clases (Usuario, Idioma, GrupoRealizado, RespSN, RespVF, RespElecUna, RespElecVarias, RespAbierta, RespEjercicio, RespComplejidad y AleatorizaModelos) que intervienen en la carga y almacenamiento de una encuesta o experimento, pero al no tener una especial relevancia se ha optado por no referenciarlas para no incrementar la complejidad de ese diagrama. (Figura 5.15)

5.3.2.1.3. Diseño de la Base de Datos

La herramienta *Empirical-WebGen* debe almacenar de manera permanente toda la información relacionada con las encuestas y experimentos. Por motivos de compatibilidad de versiones e integración dentro del entorno Visual Studio .NET, se ha elegido implementar la base de datos de la herramienta utilizando el Sistema Gestor de Bases de Datos SQL Server en su versión 2005.

Por un lado es necesario almacenar toda la información relativa a las encuestas y los experimentos que se definan. Para ello, las tablas que se utilizarán son: **Experimentos**, **Modelos**, **GruposTareas** y **Preguntas**.

Por otro lado se necesita almacenar los datos de los usuarios y sus resultados al realizar una encuesta o experimento. Las tablas usadas para este fin son: **Usuarios**, **GruposRealizados**, **Rsino**, **Rvf**, **Relecuna**, **Relecvarias**, **Rabierta**, **Rmodif** y **Rcomplejidad**.

Todas estas clases surgen de la necesidad de persistencia de las clases a la que representan. Así, y mediante la aplicación del patrón de diseño *Un Camino de Herencia Una Tabla* para las clases **RespSN**, **RespVF**, **RespElecUna**, **RespElecVarias**, **RespAbierta** y **RespEjercicio** y *Una Clase Una Tabla* para el resto (Larman, 1998) pasan a formar parte de la base de datos.

En la Figura 5.16 se muestra el esquema de la base de datos de *Empirical-WebGen* obtenido a partir de los diagramas de clases vistos anteriormente.

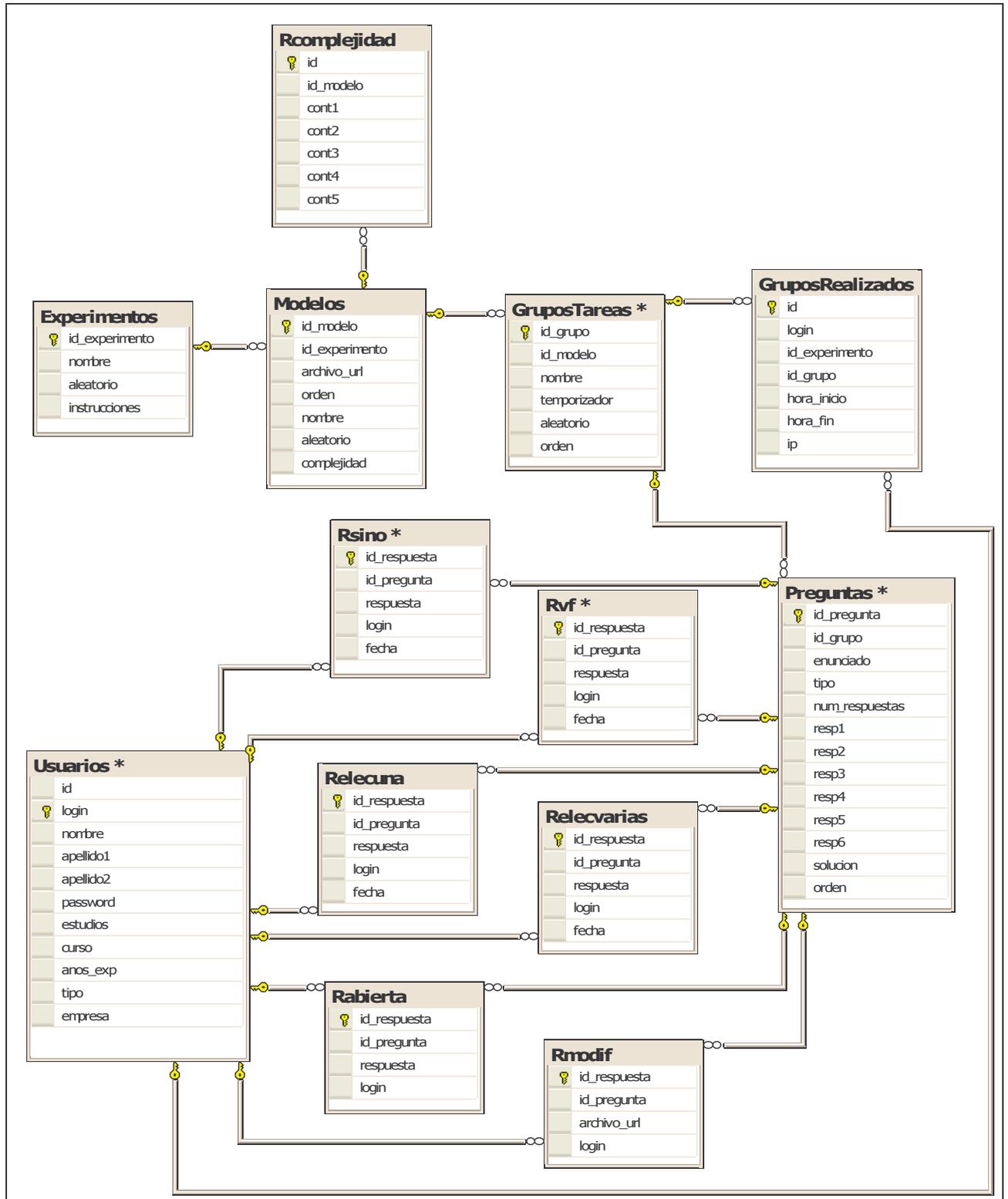


Figura 5.16. Base de datos *Empirical-WebGen*

Como se puede observar en la Figura 5.16, el diseño de la base de datos facilita a *Empirical-WebGen* el almacenamiento y gestión de la información asociada a las encuestas y experimentos, y a los usuarios y sus respuestas dadas.

5.3.3. Implementación

En esta fase se implementan físicamente los servicios, clases y subsistemas que conforman la línea base y se integran, con el fin de establecer los cimientos sobre los cuales incorporar los casos de uso menos prioritarios.

A partir del modelo de diseño construido en la fase anterior, se ha desarrollado el código de la aplicación utilizando como lenguaje Visual Basic.NET y ASP .NET 2.0. En el anexo A se incluye el código fuente de las clases más representativas e importantes de la herramienta *Empirical-WebGen*.

Teniendo en cuenta lo visto en el apartado 4.2 sobre el uso de patrones de interacción para el diseño de interfaces Web usables, se muestra a continuación algún ejemplo de aplicación de dichos patrones en la implementación de la herramienta.

❖ Patrón de interacción para el formato de las fechas.

Nombre: Formatos de Datos de fechas.

Autor: Martijn Van Wellie.

Problema: El usuario desea introducir datos de fechas y no desea preocuparse por la sintaxis del dato.

Principio: Guiar al usuario y prevenir errores.

Contexto: Todos los sistemas que requieran que el usuario introduzca fechas

Fuerzas: Los datos de fechas tienen múltiples sintaxis.

Solución: Permitir que el usuario elija la fecha de un calendario como en el mundo real.

ANEXO A. DIAGRAMAS DE LA REALIZACIÓN DE CASOS DE USO DEL SISTEMA

En este anexo se recogen las fases de inicio, elaboración y construcción de los casos de uso menos significativos del modelo de casos de uso de *Empirical-WebGen*.

A.1 - Autenticar

A.1.1- Inicio

Es el primer caso de uso que deben realizar tanto los administradores como los usuarios registrados para poder acceder al resto de funcionalidades del sistema.

Precondición.

El administrador o usuario debe estar registrado.

Flujo de sucesos.

Camino básico.

1. El administrador o usuario introduce su login y su password.
2. El sistema valida los datos y autentifica al administrador o usuario.
3. La instancia de caso de uso termina.

Caminos alternativos.

En el paso 2, el sistema detecta que el administrador o el usuario han introducido datos incorrectos. El sistema avisa mediante algún mensaje y se vuelve al paso 1.

Poscondición.

El administrador o el usuario queda autenticado.

A.1.2- Elaboración y Construcción

A.1.2.1- Realizaciones de Casos de Uso-Análisis

Diagrama de clases.



Figura A.1. Clases del caso de uso Autenticar

Diagrama de Colaboración.

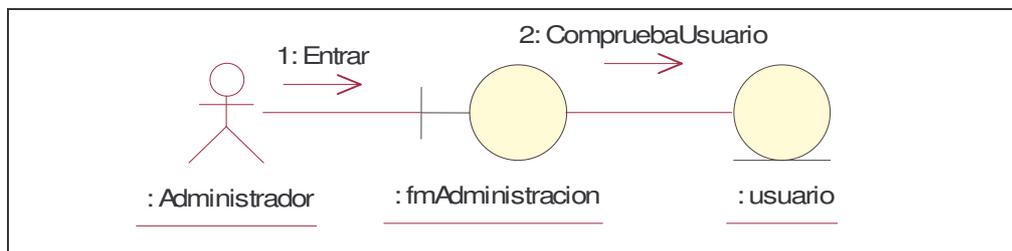


Figura A.2. Interacción de las clases para el caso de uso Autenticar

Flujo de sucesos.

El *Administrador* introduce su password y su login (1) y el sistema comprueba que existe para verificarlo (2).

A.1.2.2- Realizaciones de Casos de Uso-Diseño

Diagrama de Clases

Como se puede observar en la Figura A.3, la principal novedad del diagrama es la incorporación de la clase Persistencia, única interfaz encargada de interactuar con la base de datos aislando así la capa de almacenamiento.

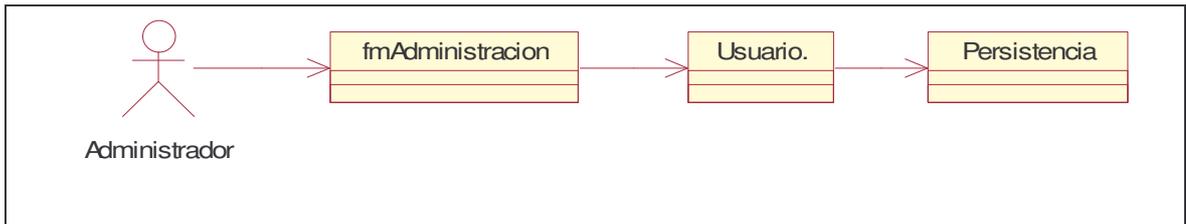


Figura A.3. Clases del caso de uso-diseño Autenticar

Diagrama de Secuencia

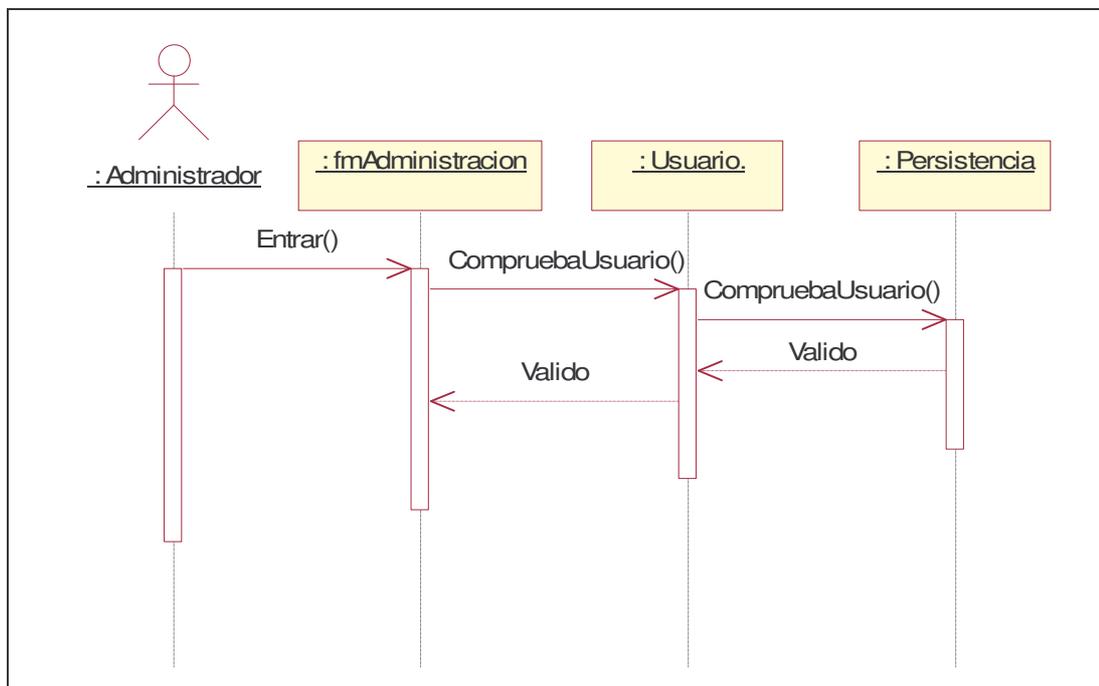


Figura A.4. Diagrama de Secuencia del caso de uso Autenticar

A.2 - Registrarse

A.2.1- Inicio

El usuario se registra en el sistema para poder realizar una encuesta o experimento.

Precondición.

El usuario debe indicar al sistema que quiere registrarse.

Flujo de sucesos.

Camino básico.

1. El usuario introduce sus datos personales.
2. La instancia de caso de uso termina.

Caminos alternativos.

En el paso 1, el usuario introduce datos incorrectos. El sistema avisa mediante algún mensaje y se vuelve al paso 1.

Poscondición.

El nuevo usuario queda registrado.

A.2.2- Elaboración y Construcción

A.2.2.1- Realizaciones de Casos de Uso-Análisis

Diagrama de clases.

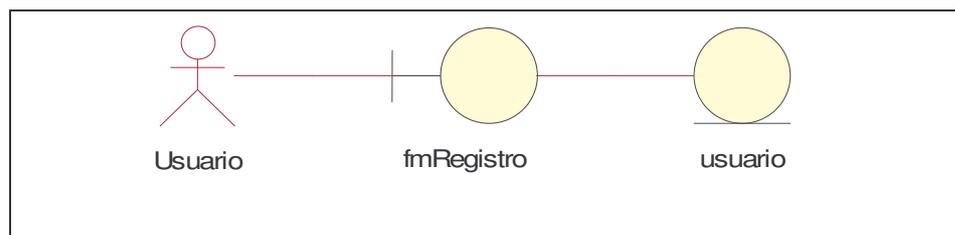


Figura A.5. Clases del caso de uso Registrarse

Diagrama de Colaboración.

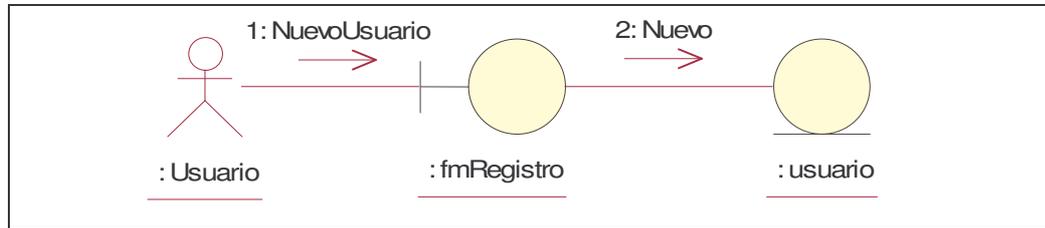


Figura A.6. Interacción de las clases para el caso de uso Registrarse

Flujo de sucesos.

El *Usuario* introduce sus datos personales (1) y se crea el nuevo usuario en la base de datos (2).

A.2.2.2- Realizaciones de Casos de Uso-Diseño

Diagrama de Clases

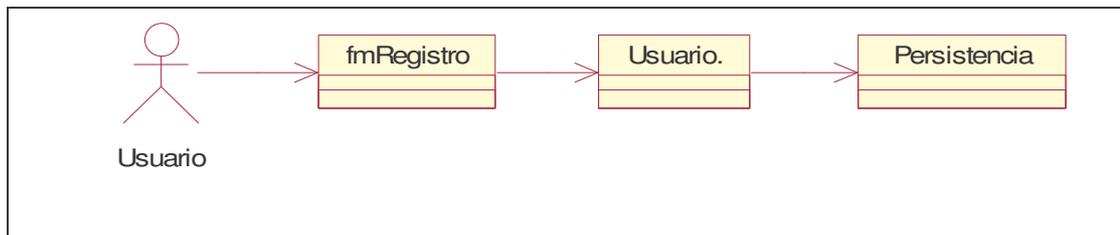


Figura A.7. Clases del caso de uso-diseño Registrarse

Diagrama de Secuencia

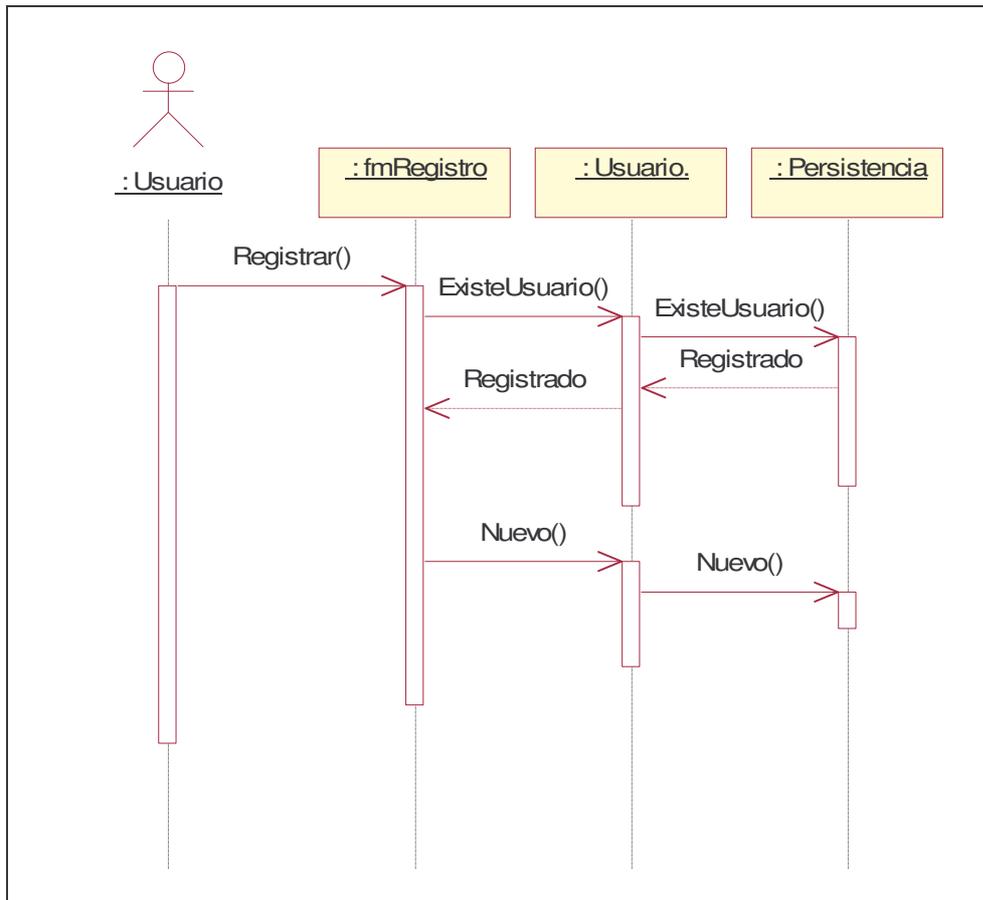


Figura A.8. Diagrama de Secuencia del caso de uso Registrarse

A.3 - Modificar Encuesta o Experimento

A.3.1- Inicio

El administrador utiliza este caso de uso para cambiar las características de una encuesta o experimento ya existente.

Precondición.

La encuesta o el experimento que se pretende modificar debe haberse creado con anterioridad.

Flujo de sucesos.

Camino básico.

1. El administrador puede cambiar el nombre de la encuesta o experimento, las instrucciones, o la aleatoriedad de sus modelos.
2. Puede agregar nuevos modelos, modificarlos o eliminar los ya existentes.
3. Puede crear nuevos grupos de tareas, editarlos para su posterior modificación o eliminar los ya existentes.
4. El administrador puede crear nuevas tareas, editarlas o eliminar las que ya existan.
5. La instancia de caso de uso termina.

Poscondición.

El nuevo experimento o encuesta queda modificado.

A.3.2- Elaboración y Construcción

A.3.2.1- Realizaciones de Casos de Uso-Análisis

Diagrama de clases.

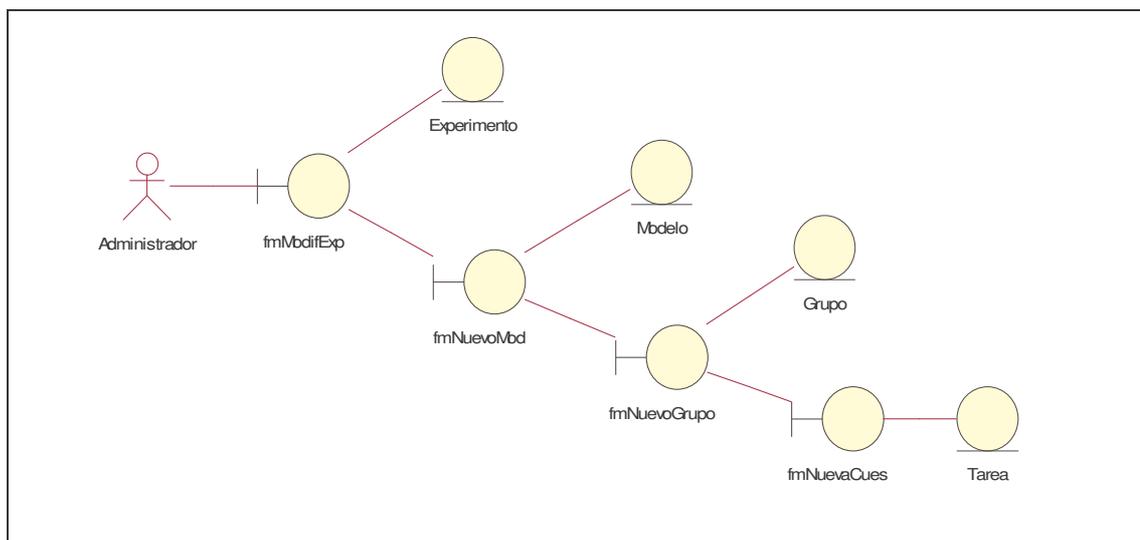


Figura A.9. Clases del caso de uso-análisis Modificar Encuesta o Experimento

Diagrama de colaboración.

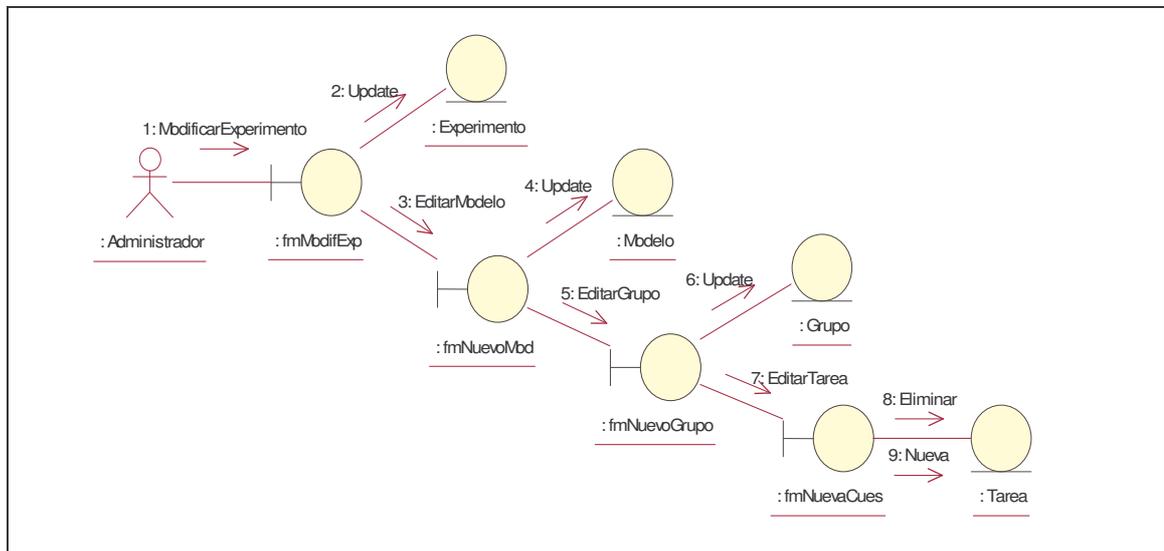


Figura A.10. Interacción de las clases para el caso de uso Modificar Encuesta o Experimento

Flujo de sucesos.

El *Administrador* decide modificar una encuesta o experimento e instancia el formulario *fmModifExp* (1). Puede editar y cambiar los parámetros oportunos de la encuesta o experimento y guardar los cambios (2) o puede editar uno de los modelos (3). Del mismo modo, puede cambiar cualquiera de los parámetros del modelo y guardar (4) o editar uno de los grupos de tareas (5). También aquí podrá realizar cambios en los atributos del grupo (6) o seleccionar una de las tareas para editarla (7). En este punto, podrá realizar cambios en la tarea, con lo que se eliminaría la antigua (8) y se crearía una nueva (9).

A.3.2.2- Realizaciones de Casos de Uso-Diseño

Diagrama de Clases

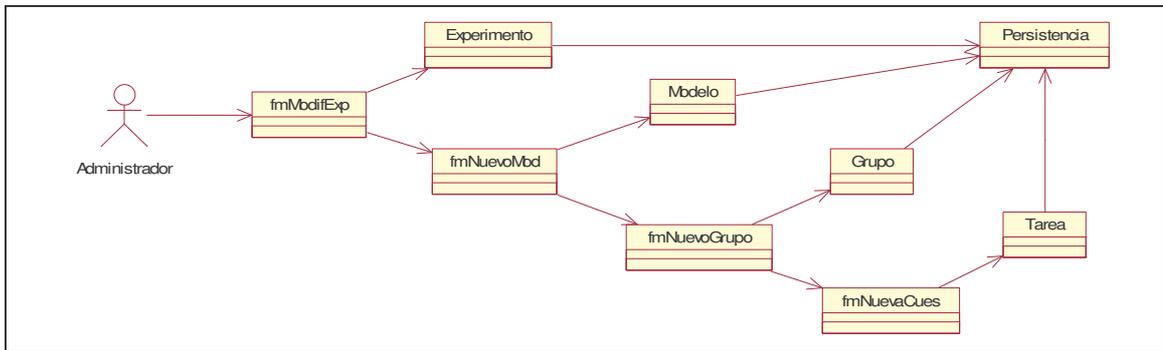


Figura A.11. Clases del caso de uso-diseño Modificar Encuesta o Experimento

Diagrama de Secuencia

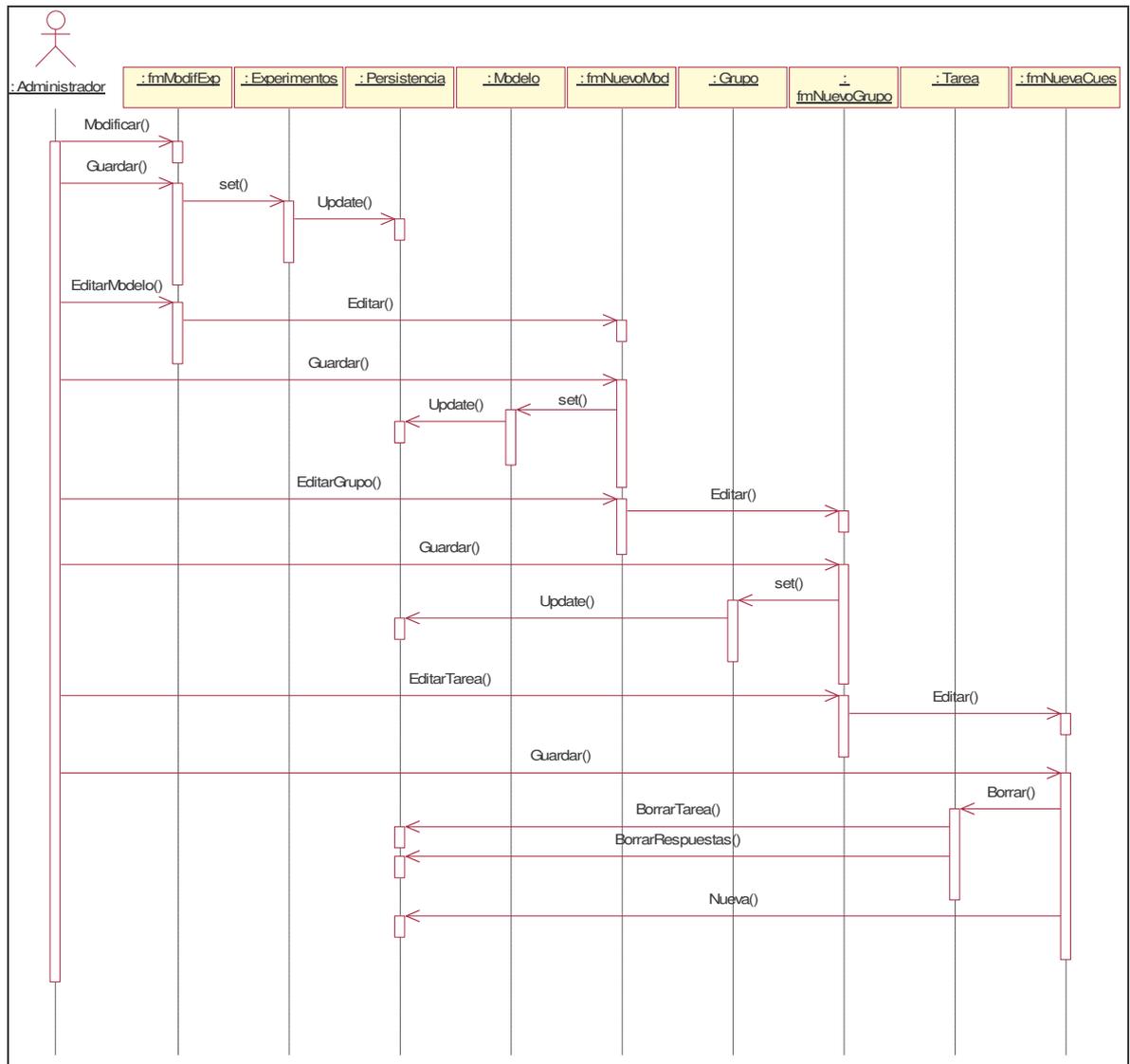


Figura A.12. Diagrama de Secuencia del caso de uso Modificar Encuesta o Experimento

A.4 - Eliminar Encuesta o Experimento

A.4.1- Inicio

Este caso de uso es utilizado por el administrador para eliminar una encuesta o experimento.

Precondición.

La encuesta o el experimento que se quiere eliminar debe haberse creado con anterioridad.

Flujo de sucesos.

Camino básico.

1. El administrador indica al sistema la encuesta o experimento que desea eliminar.
2. La instancia de caso de uso termina.

Poscondición.

La encuesta o experimento queda eliminado.

A.4.2- Elaboración y Construcción

A.4.2.1- Realizaciones de Casos de Uso-Análisis

Diagrama de clases.

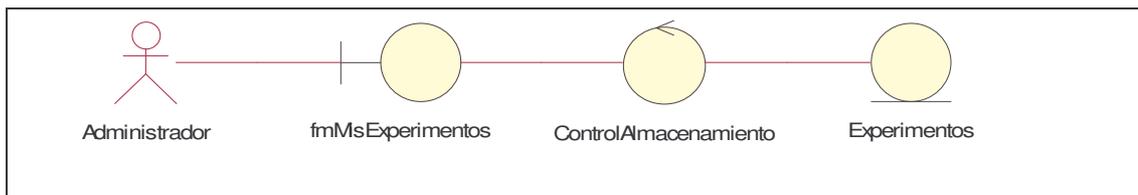


Figura A.13. Clases del caso de uso-análisis Eliminar Encuesta o Experimento

Diagrama de colaboración.

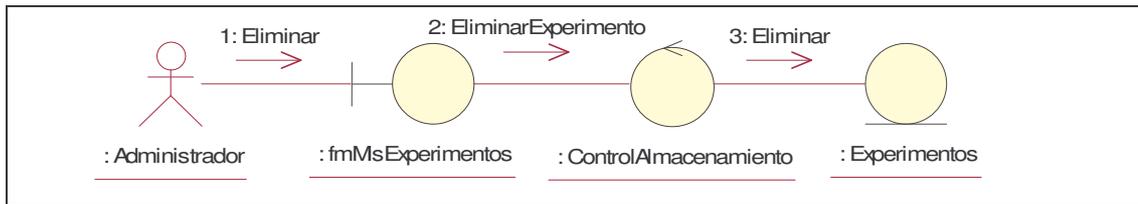


Figura A.14. Interacción de las clases para el caso de uso Eliminar Encuesta o Experimento

Flujo de sucesos.

El *Administrador* decide eliminar una de las encuestas o experimentos que aparecen listados en el formulario *fmMisExperimentos* (1). Se invoca a la clase *ControlAlmacenamiento* (2) que a su vez llama al experimento o encuesta que debe ser eliminado (3).

A.4.2.2- Realizaciones de Casos de Uso-Diseño

Diagrama de Clases

En esta ocasión, el objeto GridView de ASP .NET junto con el objeto SqlDataSource son los encargados de listar por una parte el conjunto de encuestas y experimentos, y de eliminar cualquiera de ellos a través de un comando de borrado asociado a un botón.

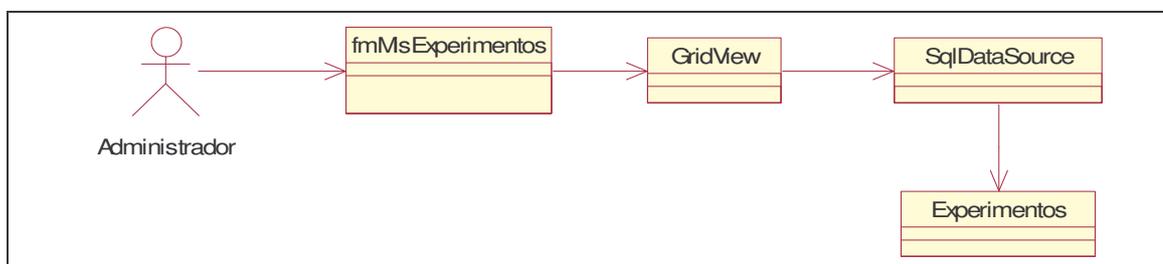


Figura A.15. Clases del caso de uso-diseño Eliminar Encuesta o Experimento

Diagrama de Secuencia

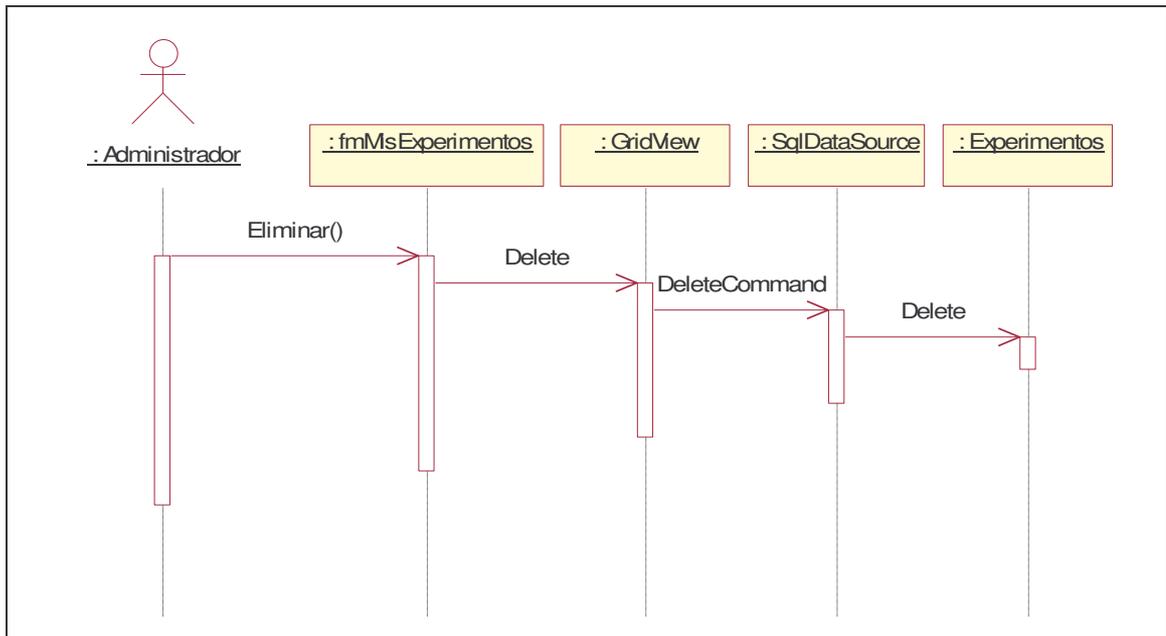


Figura A.16. Diagrama de Secuencia del caso de uso Eliminar Encuesta o Experimento

A.5 - Simular Encuesta o Experimento

A.5.1- Inicio

El administrador simula una de las encuestas o experimentos disponibles.

Precondición.

La encuesta o el experimento debe estar definido previamente.

Flujo de sucesos.

Camino básico.

1. El administrador elige la encuesta o el experimento a simular.
2. El sistema muestra una página de instrucciones que el administrador debe aceptar para poder continuar.
3. El administrador simula la encuesta o experimento contestando a las tareas que se le presentan tal y como la haría un usuario.
4. El sistema muestra una página de agradecimiento.

5. La instancia de caso de uso termina.

Caminos alternativos.

En el paso 3, el administrador puede abortar la realización de la encuesta o experimento volviendo al estado inicial.

A.5.2- Elaboración y Construcción

A.5.2.1- Realizaciones de Casos de Uso-Análisis

Diagrama de clases.

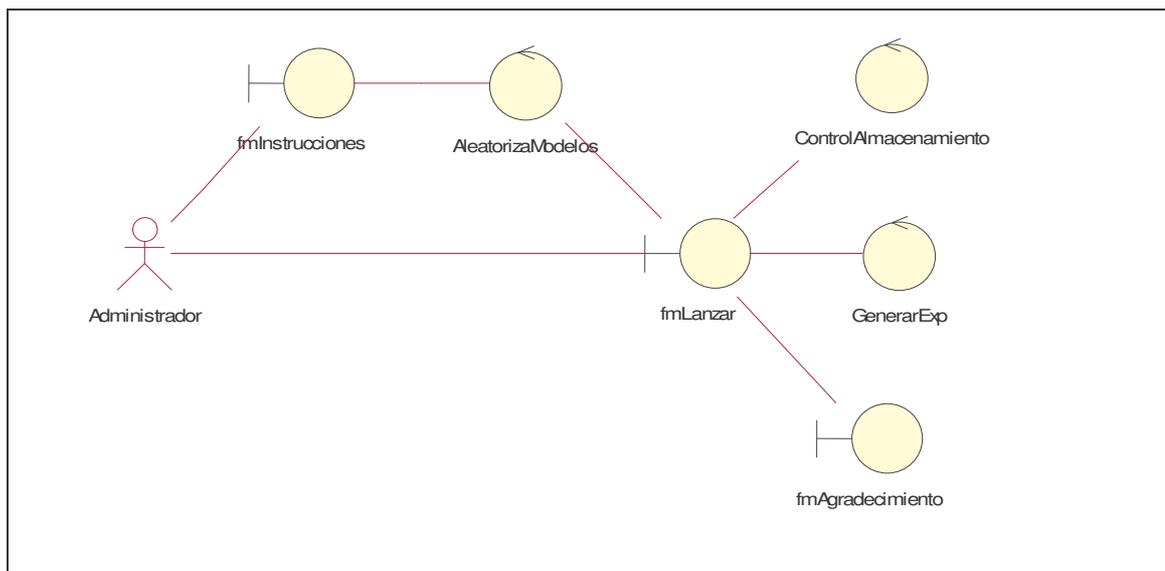


Figura A.17. Clases del caso de uso-análisis Simular Encuesta o Experimento

Diagrama de colaboración.

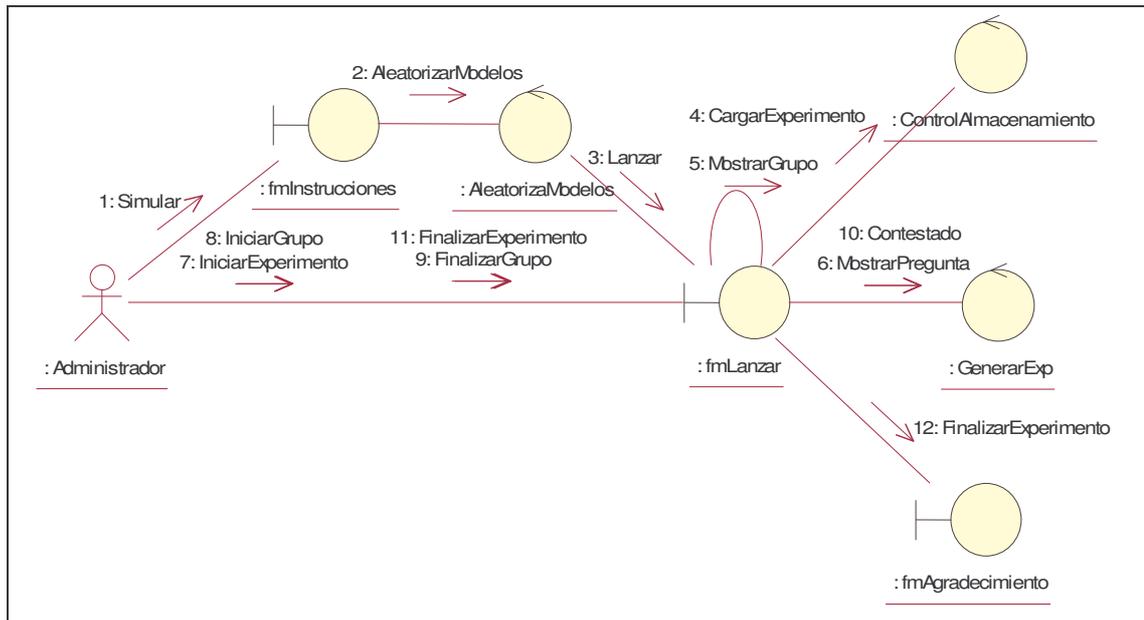


Figura A.18. Interacción de las clases para el caso de uso Simular Encuesta o Experimento

Flujo de sucesos.

El *Administrador* decide simular una encuesta o experimento y en primer lugar se instancia el formulario *fmInstrucciones* (1). Se aleatorizan los modelos si estuviese así diseñado el experimento (2) y se lanza dicho experimento (3). Se carga el experimento (4) y se muestran los grupos (5) con sus respectivas preguntas (6) aunque estarán ocultos hasta que el *Administrador* inicie el experimento (7) y pulse el botón de inicio de grupo (8). El *Administrador* realizará las tareas y cuando finalice pulsará el botón de fin de grupo (9). El sistema comprueba que se han contestado correctamente las tareas (10) y continuará mostrando el resto de grupos o modelos si los hubiera. Cuando el *Administrador* ha completado todas las tareas finalizará el experimento (11) y se mostrará una página de agradecimiento (12).

A.5.2.2- Realizaciones de Casos de Uso-Diseño

Diagrama de Clases

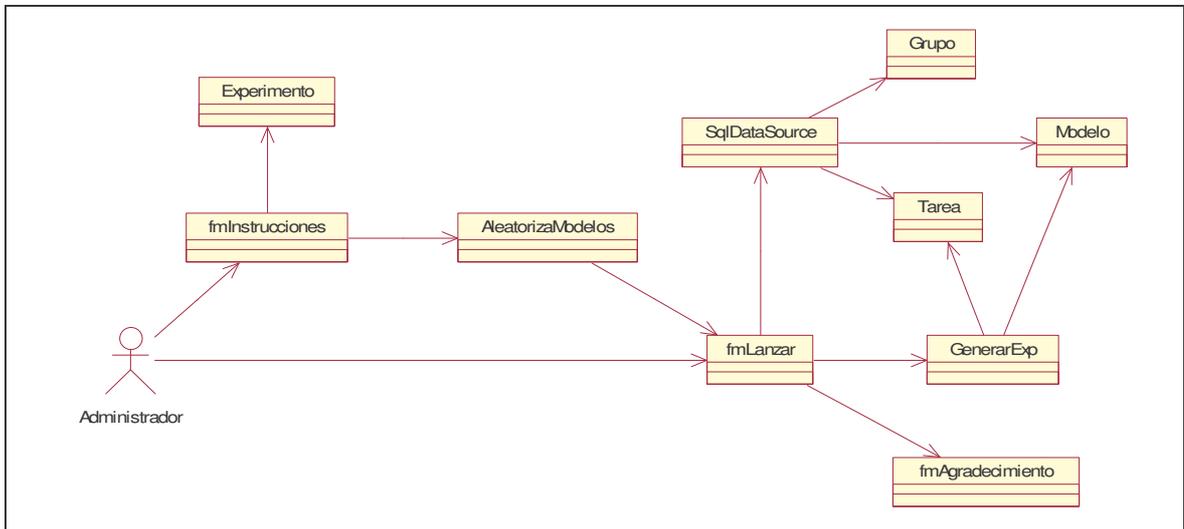


Figura A.19. Clases del caso de uso-diseño Simular Encuesta o Experimento

Se ha evitado poner la clase Persistencia, que estaría relacionada con las clases Experimento, Grupo, Modelo y Tarea para simplificar los diagramas de caso de uso y secuencia.

Diagrama de Secuencia

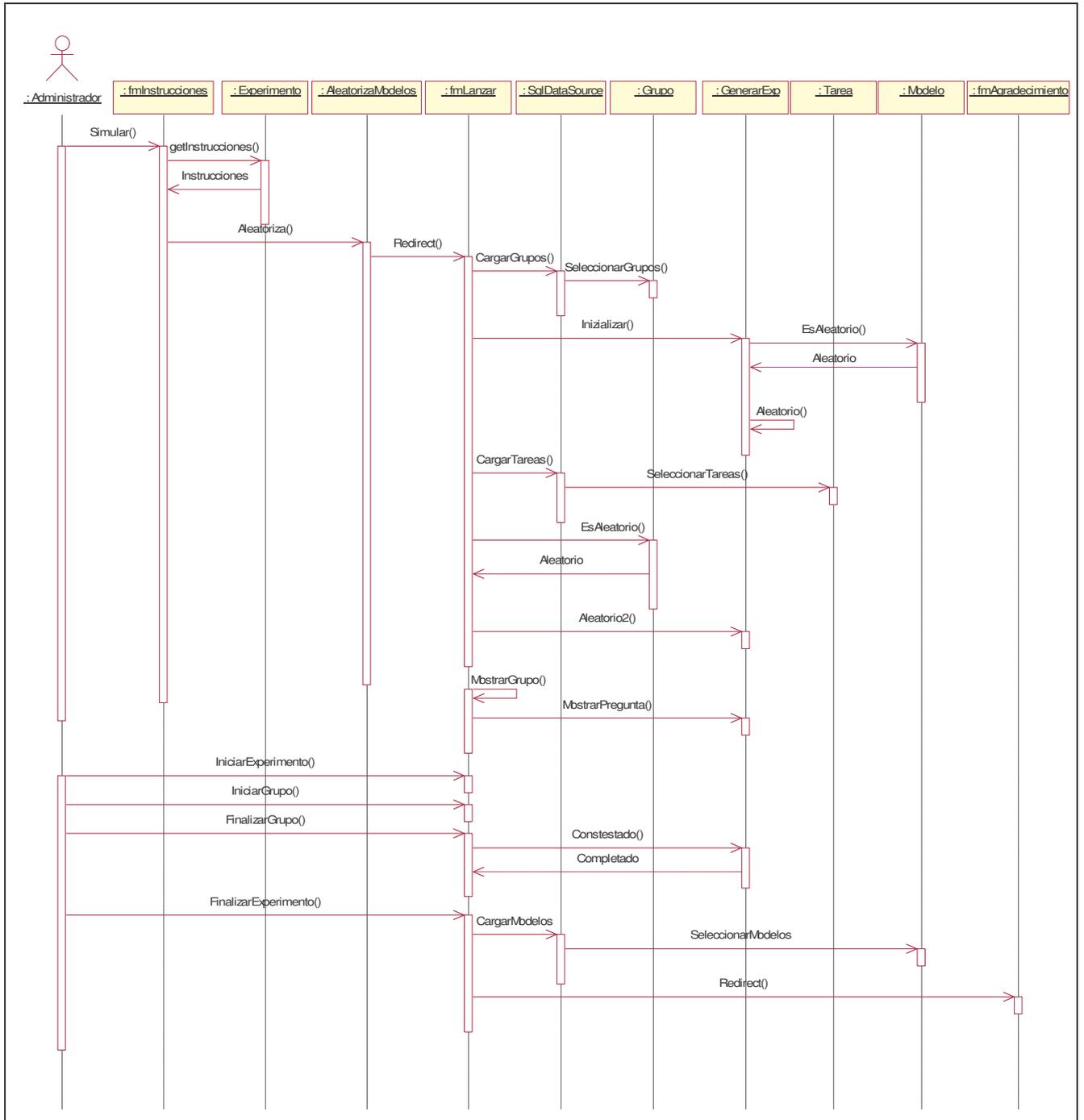


Figura A.20. Diagrama de Secuencia del caso de uso Simular Encuesta o Experimento

A.6 - Visualizar Respuestas

A.6.1- Inicio

Este caso de uso se inicia cuando el administrador indica que quiere mostrar las respuestas almacenados de una encuesta o experimento ya definida. El administrador podrá elegir unos parámetros opcionales.

Precondición.

La encuesta o experimento de la que se desea visualizar las respuestas debe haberse creado con anterioridad.

Flujo de sucesos.

Camino básico.

1. El administrador selecciona el informe de respuestas.
2. El administrador puede escoger entre mostrar el detalle de todos los usuarios que han realizado la encuesta o experimento o no mostrarlo.
3. Puede realizar un filtrado de los usuarios que han realizado la encuesta o experimento por fechas.
4. El sistema muestra el informe.
5. La instancia de caso de uso termina.

Caminos alternativos.

En el paso 4, el administrador puede exportar los resultados en varios formatos.

Poscondición.

Se muestra el informe de respuestas del experimento o encuesta seleccionada por el administrador.

A.6.2- Elaboración y Construcción

A.6.2.1- Realizaciones de Casos de Uso-Análisis

Diagrama de clases.

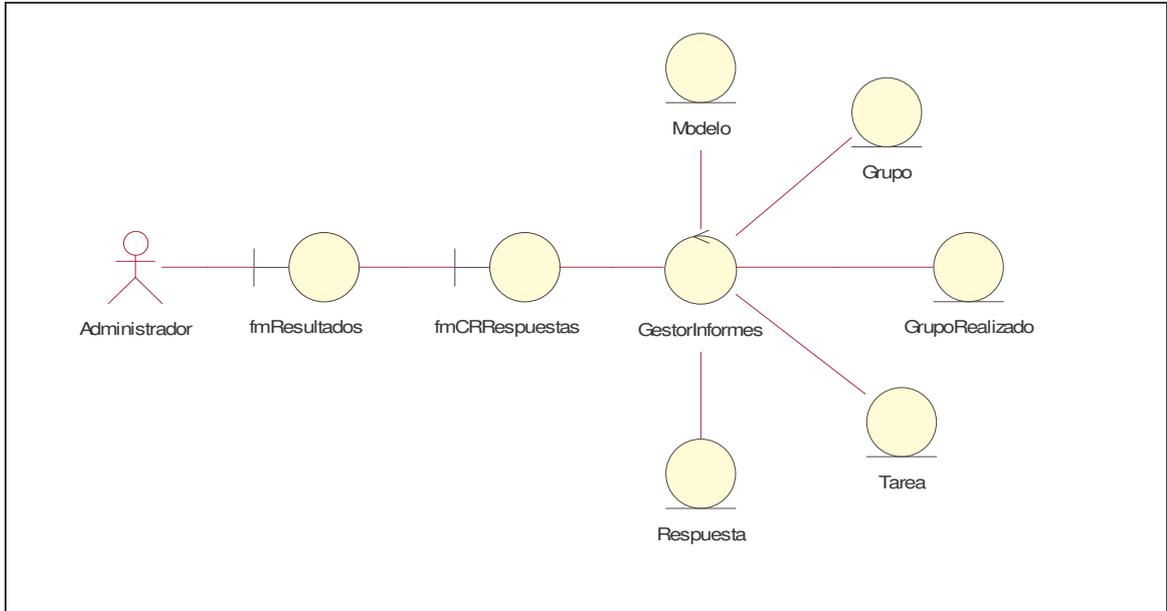


Figura A.21. Clases del caso de uso Visualizar Respuestas

Diagrama de Colaboración.

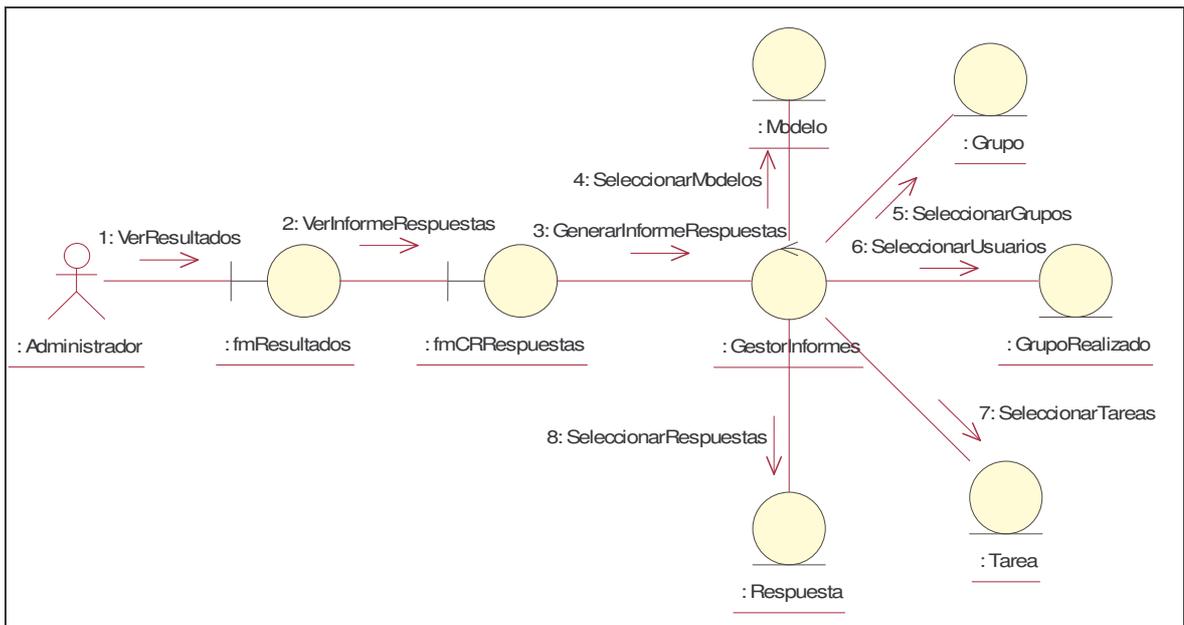


Figura A.22. Interacción de las clases para el caso de uso Visualizar Respuestas

Flujo de sucesos.

El *Administrador* selecciona la encuesta o experimento del que desea ver los resultados (1). A continuación selecciona el tipo de informe “Respuestas” y configura algún parámetro si desea (2) para pedir después a la clase *GestorInformes* que genere el informe de respuestas (3). Esta clase seleccionará los modelos, grupos de tareas, usuarios, tareas y respuestas de las respectivas tablas en la base de datos (del 4 al 8).

A.6.2.2- Realizaciones de Casos de Uso-Diseño

Diagrama de Clases

Para el diseño de este caso de uso se han utilizado las clases *CrystalReportViewer*, *CrystalReportSource* y *Report*, pertenecientes al paquete *CrystalDecisions* de Visual Studio. Estas clases acceden a la base de datos a través de su propia interfaz sin necesidad del gestor *Persistencia*.

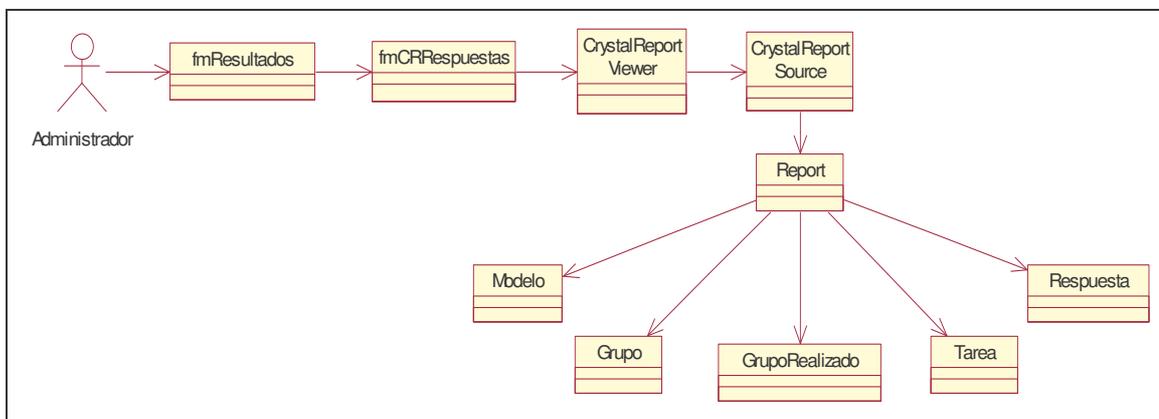


Figura A.23. Clases del caso de uso-diseño Visualizar Respuestas

Diagrama de Secuencia

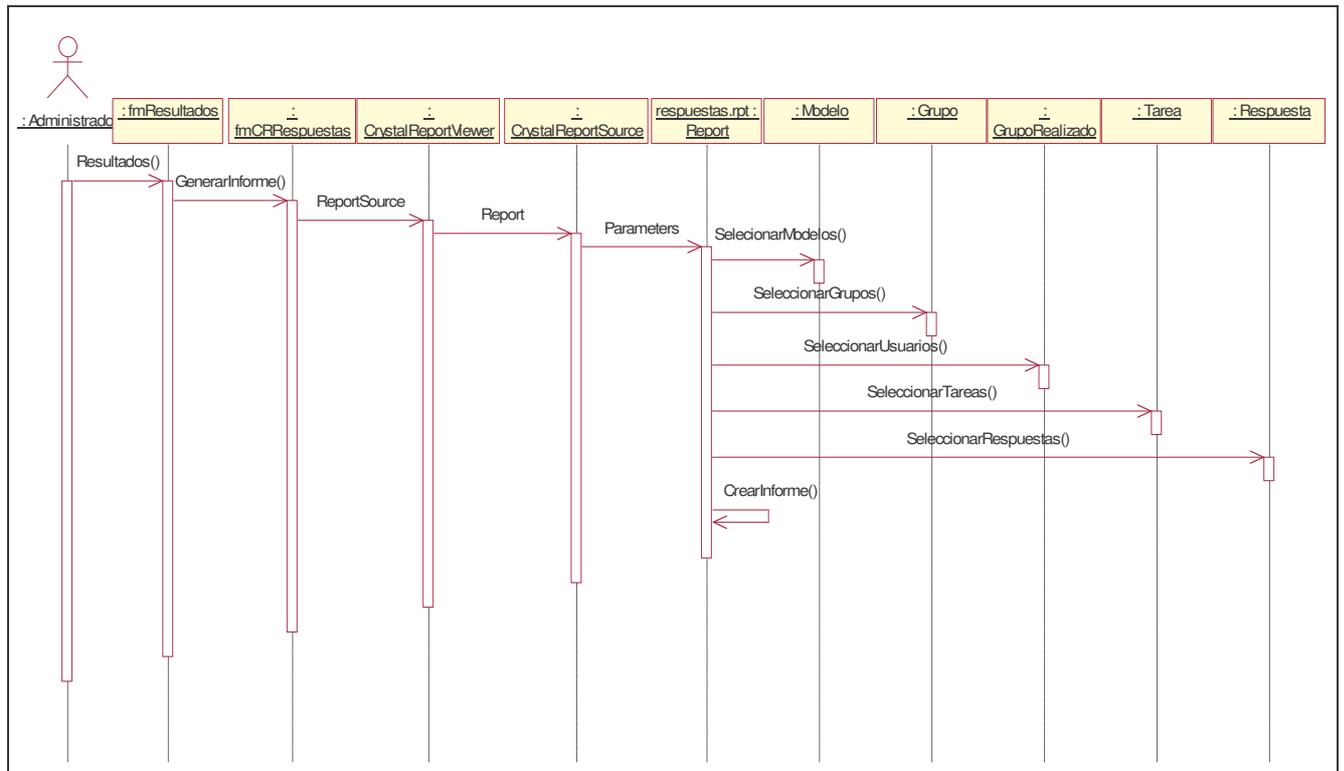


Figura A.24. Diagrama de Secuencia del caso de uso Visualizar Respuestas