



CLASS CONTROL

DOCUMENTO DE DISEÑO, VERSIÓN 1.3

Desarrollo de una aplicación para la plataforma social educativa *edmodo*.

TFG1314-40



30 DE JUNIO DE 2014

ALUMNO: CRISTIAN TEJEDOR GARCÍA

TUTORA: YANIA CRESPO GONZÁLEZ-CARVAJAL



Escuela Técnica Superior
de **Ingeniería Informática**

<i>Class Control</i>	Versión: 1.3
Documento de Diseño	30/06/2014

Historial de Revisiones

Fecha	Versión	Descripción	Autor
22/02/2014	0.8	Versión preliminar del Documento Diseño sin los diagramas de secuencia del sistema y el diagrama de clases de diseño.	Cristian T.G.
28/02/2014	0.9	Añadidos diagramas de secuencia y diagrama de clases de diseño. Mejorados bocetos gráficos de la interfaz de Usuario.	Cristian T.G.
05/03/2014	1.0	Versión inicial del documento. Revisión general de apartados	Cristian T.G.
26/03/2014	1.1	Descripción de casos de uso de diseño mejorada. Diagramas de secuencia y diagrama de Clases de diseño.	Cristian T.G.
01/04/2014	1.2	Revisión de diagramas de secuencia.	Cristian T.G.
30/06/2014	1.3	Patrones de diseño aplicados.	Cristian T.G.

TABLA 1: HISTORIAL DE REVISIONES

<i>Class Control</i>	Versión: 1.3
Documento de Diseño	30/06/2014

Contenido

Índice De Tablas	3
Índice De Ilustraciones	4
1. Introducción	7
1.1. Propósito	7
1.2. Alcance	7
1.3. Definiciones, acrónimos y abreviaturas	7
1.4. Referencias	7
2. Arquitectura propuesta	8
1.1. Visión global	8
1.2. Diseño de la Arquitectura	13
1.2.1. Descomposición en subsistemas	13
1.2.2. Topología del sistema	13
1.2.3. Gestión de la persistencia	14
1.2.4. Aspectos globales y de seguridad	14
1.2.5. Aspectos de rendimiento y tamaño	14
1.3. Diseño de los subsistemas	15
1.3.1. Vista de casos de uso. Realización de casos de uso	15
1.3.2. Patrones de diseño utilizados	42
1.3.3. Vista lógica	44
3. Modelo de datos	46
3.1. Diagrama relacional de la base de datos	46
4. Diseño de interfaces de usuario	47
4.1. Bocetos de las interfaces de usuario	47
4.1.1. Profesor	47
4.1.2. Alumno	57
4.1.3. Padre	60
4.1.4. Páginas de error	64
4.2. Mapa web del sistema final	65

Índice De Tablas

Tabla 1: Historial de revisiones	2
Tabla 2: Descripción del CU01	16
Tabla 3: Descripción del CU02	17
Tabla 4: Descripción del CU03	17

<i>Class Control</i>	Versión: 1.3
Documento de Diseño	30/06/2014

Tabla 5: Descripción del CU04	18
Tabla 6: Descripción del CU05	18
Tabla 7: Descripción del CU06	19
Tabla 8: Descripción del CU09	21
Tabla 9: Descripción del CU10	22
Tabla 10: Descripción del CU11	23
Tabla 11: Descripción del CU12	23
Tabla 12: Descripción del CU13	24
Tabla 13: Descripción del CU14	24
Tabla 14: Descripción del CU15	25
Tabla 15: Descripción del CU16	25
Tabla 16: Descripción del CU17	26
Tabla 17: Descripción del CU18	26
Tabla 18: Descripción del CU19	27
Tabla 19: Descripción del CU21	29
Tabla 20: Descripción del CU22	30
Tabla 21: Descripción del CU23	30
Tabla 22: Descripción del CU24	31

Índice De Ilustraciones

Ilustración 1: Arquitectura J2EE	8
Ilustración 2: Capas de la Arquitectura JEE	9
Ilustración 3: Esquema Lógico de la Arquitectura J2EE	10
Ilustración 4: Capas de la Arquitectura de una Aplicación J2EE	10
Ilustración 5: Patrón MVC detallado	11
Ilustración 6: Interacción en el patrón MVC en una aplicación web	12
Ilustración 7: Descomposición En Subsistemas	13
Ilustración 8: Diagrama De Despliegue Del Sistema	13
Ilustración 9: Diagrama de Casos de Uso de Diseño	15
Ilustración 10: CU07 desde Edmodo. Paso 1	19
Ilustración 11: CU07 desde Edmodo. Paso 2	20
Ilustración 12: CU08 desde Edmodo. Paso 1	20
Ilustración 13: CU08 desde Edmodo. Paso 2	21
Ilustración 14: CU20 desde Edmodo. Paso 1	27
Ilustración 15: CU20 desde Edmodo. Paso 2	28
Ilustración 16: CU20 desde Edmodo. Paso 3	28
Ilustración 17: CU20 desde Edmodo. Paso 4	29
Ilustración 18: Diagrama de secuencia De Ejemplo DAO	32
Ilustración 19: Diagrama de secuencia del CU01	33
Ilustración 20: Diagrama de secuencia del CU02	34
Ilustración 21: Diagrama de clases de Diseño	35
Ilustración 22: Clases <i>Controller</i>	36
Ilustración 23: Acceso a la base de datos	36
Ilustración 24: Modelado de la persistencia	37
Ilustración 25: DAOs	38
Ilustración 26: Clases Entity - Parte 1	39

<i>Class Control</i>	Versión: 1.3
Documento de Diseño	30/06/2014

Ilustración 27: Clases Entity - Parte 2.....	40
Ilustración 28: Clases útiles de acceso a la API de Edmodo	41
Ilustración 29: Clases para la internacionalización	41
Ilustración 30: Diagrama de secuencia del patrón Front Controller tomado de [6].....	42
Ilustración 31: Diagrama explicativo del patrón Façade tomado de [8]	43
Ilustración 32: Diagrama de clases del patrón Abstract Factory tomado de [9].....	43
Ilustración 33: Diagrama de secuencia del patrón DAO tomado de [10].....	44
Ilustración 34: Diagrama de Paquetes del sistema.....	45
Ilustración 35: Diagrama Relacional de la Base De Datos	46
Ilustración 36: Boceto De Profesor: Página Principal	47
Ilustración 37: Boceto De Profesor: Crear Asignatura o Competencia	47
Ilustración 38: Boceto De Profesor: Crear Asignatura o Competencia 2	48
Ilustración 39: Boceto De Profesor: Seleccionar Asignatura o Competencia	48
Ilustración 40: Boceto De Profesor: Asignatura o Competencia	48
Ilustración 41: Boceto De Profesor: Asistencia.....	49
Ilustración 42: Boceto De Profesor: Indicar Ausencias.....	49
Ilustración 43: Boceto De Profesor: Indicar Ausencias 2.....	49
Ilustración 44: Boceto De Profesor: Marcar Días No Lectivos.....	50
Ilustración 45: Boceto De Profesor: Visualizar Informe Asistencia	50
Ilustración 46: Boceto De Profesor: Conducta Y Actitud	50
Ilustración 47: Boceto De Profesor: Indicar Aspecto 1.....	51
Ilustración 48: Boceto De Profesor: Indicar Aspecto 2.....	51
Ilustración 49: Boceto De Profesor: Añadir Aspectos De Conducta y Actitud	51
Ilustración 50: Boceto De Profesor: Visualizar Informe Conducta y Actitud.....	52
Ilustración 51: Boceto De Profesor: Pruebas y Trabajos	52
Ilustración 52: Boceto De Profesor: Seleccionar Prueba o Trabajo	52
Ilustración 53: Boceto De Profesor: Calificar Prueba o Trabajo	53
Ilustración 54: Boceto De Profesor: Crear Prueba o Trabajo	53
Ilustración 55: Boceto De Profesor: Visualizar Informe Evaluación	53
Ilustración 56: Boceto De Profesor: Indicar Pesos en la Evaluación Total	54
Ilustración 57: Boceto De Profesor: Editar Asignatura o Competencia	54
Ilustración 58: Boceto De Profesor: Modificar Asignatura o Competencia 1	54
Ilustración 59: Boceto De Profesor: Modificar Asignatura o Competencia 2	55
Ilustración 60: Boceto De Profesor: Eliminar Asignatura o Competencia.....	55
Ilustración 61: Boceto De Profesor: Obtener Informes 1.....	55
Ilustración 62: Boceto De Profesor: Obtener Informes 2.....	56
Ilustración 63: Boceto De Alumno: Página Principal	57
Ilustración 64: Boceto De Alumno: Seleccionar Asignatura o Competencia	57
Ilustración 65: Boceto De Alumno: Asignatura o Competencia	57
Ilustración 66: Boceto De Alumno: Visualizar Evaluación	58
Ilustración 67: Boceto De Alumno: Visualizar Asistencia	58
Ilustración 68: Boceto De Alumno: Visualizar conducta o actitud	58
Ilustración 69: Boceto De Alumno: Visualizar Calificación Prueba.....	59
Ilustración 70: Boceto De Alumno: Obtener Informes 1	59
Ilustración 71: Boceto De Alumno: Obtener Informes 2	59
Ilustración 72: Boceto De Padre: Página Principal.....	60
Ilustración 73: Boceto De Padre: Página Principal Del Hijo	60

<i>Class Control</i>	Versión: 1.3
Documento de Diseño	30/06/2014

Ilustración 74: Boceto De Padre: Seleccionar Asignatura o Competencia	60
Ilustración 75: Boceto De Padre: Asignatura o Competencia	61
Ilustración 76: Boceto De Padre: Visualizar Asistencia.....	61
Ilustración 77: Boceto De Padre: Visualizar Conducta o actitud	61
Ilustración 78: Boceto De Padre: Visualizar Evaluación.....	62
Ilustración 79: Boceto De Padre: Visualizar Calificación Prueba	62
Ilustración 80: Boceto De Padre: Obtener Informes 1	62
Ilustración 81: Boceto De Padre: Obtener Informes 2	63
Ilustración 82: Boceto De Página de Error 403.....	64
Ilustración 83: Boceto De Página de Error 404.....	64
Ilustración 84: Boceto De Página de Error 500.....	64
Ilustración 85: Mapa web del sistema - Parte 1.....	65
Ilustración 86: Mapa web del sistema - Parte 2.....	66

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

1. Introducción

1.1. Propósito

El propósito de este documento es mostrar la Arquitectura final del sistema, junto a los subsistemas de diseño y los bocetos interfaz gráfica de la aplicación.

1.2. Alcance

El presente documento profundiza en las estructuras y comportamiento descritos en *el documento de Requisitos* para acercarlos a la implementación real del sistema. Mientras que el Modelo de Análisis se centra en los requisitos funcionales, el Modelo de Diseño abarcará también los requisitos no funcionales, y tendrá en cuenta el lenguaje y las tecnologías que serán usadas en la implementación.

La versión de este documento se corresponde con la última versión del proyecto.

1.3. Definiciones, acrónimos y abreviaturas

Las definiciones, acrónimos y abreviaturas utilizadas a lo largo del proyecto se incluyen en el *documento de Glosario de Términos*.

1.4. Referencias

- [1] <<Balsamiq Mockups>>. [En línea] Disponible en: <http://balsamiq.com/products/mockups/> [Accedido: 02-marzo-2014]
- [2] <<ClearDB Heroku add-on>>. [En línea] Disponible en: <https://addons.heroku.com/cleardb> [Accedido: 02-marzo-2014]
- [3] <<Dynos and the Dyno Manager>>. [En línea] Disponible en: <https://devcenter.heroku.com/articles/dynos> [Accedido: 12-marzo-2014]
- [4] <<The Java EE 6 Tutorial>>. [En línea] Disponible en: <http://docs.oracle.com/javaee/6/tutorial/doc/> [Accedido: 14-marzo-2014]
- [5] <<Core J2EE Patterns - Data Access Object>>. [En línea] Disponible en: <http://www.oracle.com/technetwork/java/dataaccessobject-138824.html> [Accedido: 20-marzo-2014]
- [6] <<Core J2EE Patterns - Front Controller>>. [En línea] Disponible en: <http://www.oracle.com/technetwork/java/frontcontroller-135648.html> [Accedido: 14-abril-2014]
- [7] <<GRASP (object-oriented design)>>. [En línea] Disponible en: [http://en.wikipedia.org/wiki/GRASP_\(object-oriented_design\)](http://en.wikipedia.org/wiki/GRASP_(object-oriented_design)) [Accedido: 14-abril-2014]
- [8] <<Facade in Java>>. [En línea] Disponible en: http://sourcemaking.com/design_patterns/facade/java/1 [Accedido: 14-abril-2014]
- [9] <<Patrón Abstract Factory en Java>>. [En línea] Disponible en: <http://lineadecodigo.com/patrones/patron-abstract-factory/> [Accedido: 14-abril-2014]
- [10] <<Core J2EE Patterns - Data Access Object>>. [En línea] Disponible en: <http://www.oracle.com/technetwork/java/dataaccessobject-138824.html> [Accedido: 14-abril-2014]

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

2. Arquitectura propuesta

1.1. Visión global

Se parte de la arquitectura Java Empresarial JEE o J2EE.

La edición empresarial es llamada Java *Enterprise Edition* (Java EE) a partir de la versión 5. En versiones anteriores a ésta se le llama Java 2 *Enterprise Edition*. Está destinada a cubrir las necesidades de aplicaciones empresariales (en servidores). Esta edición define las características que se requieren para poder realizar aplicaciones que se ejecuten de forma portable a través de servidores de aplicaciones JEE certificados.

Java EE es una especificación que engloba dentro de sí misma a un conjunto de especificaciones (APIs) relacionadas. Java EE se fundamenta en Java *Standard Edition*, y aprovecha todas las características básicas presentes en la edición estándar de la plataforma.

La plataforma Java EE está definida por una especificación. Similar a otras especificaciones del *Java Community Process*, Java EE es también considerada informalmente como un estándar debido a que los suministradores deben cumplir ciertos requisitos de conformidad para declarar que sus productos son conformes a Java EE; no obstante sin un estándar de ISO o ECMA.

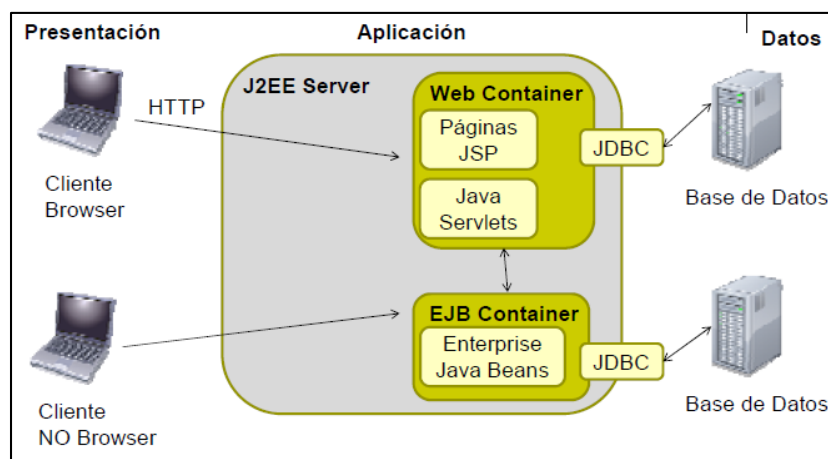


ILUSTRACIÓN 1: ARQUITECTURA J2EE

Para ponerla en funcionamiento, se necesita un servidor J2EE, que se apoya ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones que permite los siguientes servicios:

- HTTP y HTTPS: Protocolos estándares utilizados para comunicaciones web y para comunicaciones seguras sobre *Secure Socket Layer* (SSL), respectivamente. La API para clientes está definida por el paquete `java.net.*` y la API para servidor está definida por las clases de servlets y JSP.
- JNDI: *Java Naming and Directory Interface*.
- *Java DataBase connectivity* (JDBC): Una API estándar para acceder a los recursos de una base de datos relacional de una forma independiente del proveedor. Esta API consta de dos partes, una interfaz para ser utilizada por las componentes y una interfaz de proveedores para definir drivers específicos. Oficialmente JDBC no es un acrónimo, aunque comúnmente se utiliza el nombre de Java Database Connectivity.
- *Java Persistence API*.
- *Java Messaging Service* (JMS)
- *Remote Method Invocations* (RMI)

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

- JavaMail: Una API que permite crear aplicaciones Java para mensajería y envío de correo electrónico en forma independiente de la plataforma y del protocolo a utilizar.
- Java Transaction API (JTA).

La arquitectura JEE implica un modelo de aplicaciones distribuidas en diversas capas o niveles (*tiers*), esquemáticamente se muestra en la *Ilustración 28*:

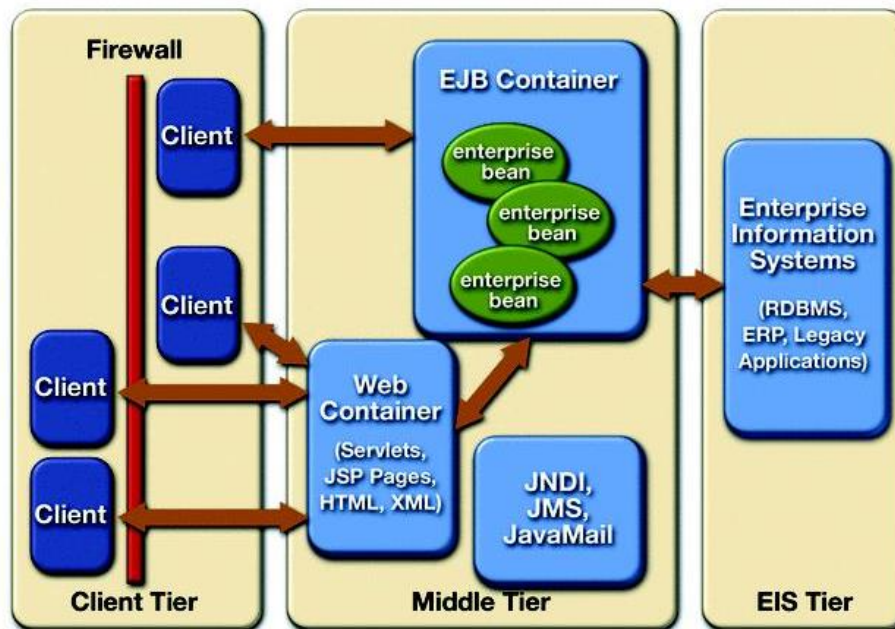


ILUSTRACIÓN 2: CAPAS DE LA ARQUITECTURA JEE

- La **capa cliente** (*client tier*) admite diversos tipos de clientes (HTML, Applets y aplicaciones Java). Esta capa corresponde a lo que se encuentra en el computador del cliente. Es la interfaz gráfica del sistema y se encarga de interactuar con el usuario.
- La **capa intermedia** (*middle tier*) contiene dos subcapas/contenedores:
 - **Contenedor web:** Se encuentra en el servidor web y contiene la lógica de presentación que se utiliza para generar una respuesta al cliente. Recibe los datos del usuario desde la capa cliente y basado en éstos genera una respuesta apropiada a la solicitud. J2EE utiliza en esta capa las componentes Java Servlets y Java Server Pages para crear los datos que se enviarán al cliente.
 - **Contenedor EJB:** Entorno donde se ejecutan los EJB. Hospeda y maneja un EJBs de la misma forma que un servidor Web Java hospeda un *servlet* o un navegador hospeda un *applet*. Un EJB no puede funcionar fuera de un contenedor EJB. El contenedor EJB maneja cualquier aspecto del *bean* en tiempo de ejecución, incluyendo acceso remoto al *bean*, seguridad, persistencia, transacciones, concurrencia y acceso y *pooling* de recursos. El contenedor EJB también suele proporcionar servicios relacionados con la escalabilidad de la aplicación, como son la definición de *clusters* de contenedores, el balanceo de carga o la tolerancia a fallos.
- La **tercera capa** (*EIS tier*) dentro de esta visión sintética es la de aplicaciones 'backend' como ERP, EIS, bases de datos, etc. Esta capa es responsable del sistema de información de la empresa o Enterprise Information System (EIS) que incluye bases de datos, sistema de procesamiento de datos, sistemas legados y sistemas de planificación de recursos. Es el punto donde las aplicaciones J2EE se integran con otros sistemas no J2EE o con sistemas legados.

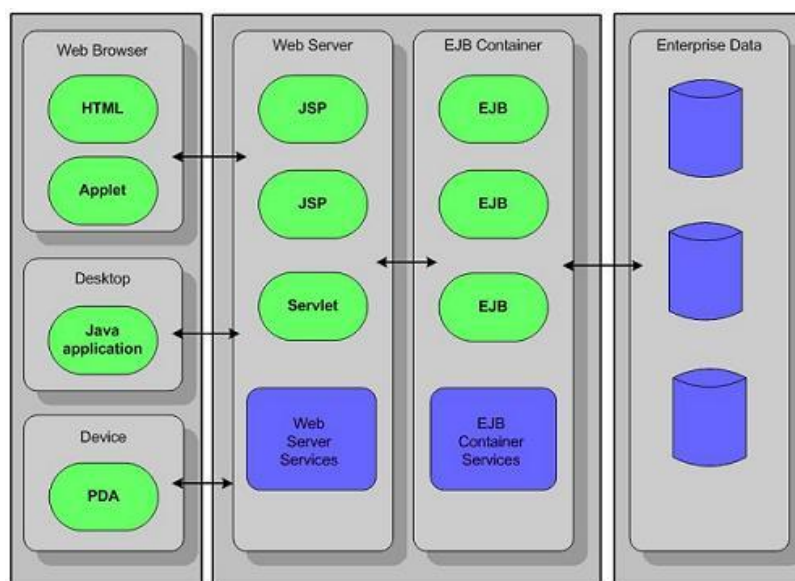


ILUSTRACIÓN 3: ESQUEMA LÓGICO DE LA ARQUITECTURA J2EE

Un concepto clave de la arquitectura es el de *contenedor* un entorno de ejecución estandarizado que ofrece unos servicios por medio de componentes. Los componentes externos al contenedor tienen una forma estándar de acceder a los servicios de dicho contenedor, con independencia del fabricante. Incluyen descriptores de despliegue (*deployment descriptors*), que son archivos XML que sirven para configurar el entorno de ejecución: rutas de acceso a aplicaciones, control de transacciones, parámetros de inicialización, etc. Requieren un servidor *Enterprise Edition*. Como se ha mencionado antes existen dos contenedores en la arquitectura J2EE: el contenedor Web y el contenedor Enterprise JavaBeans.

Las aplicaciones J2EE son divididas en cuatro capas: la capa cliente, la capa web, la capa negocio y la capa datos. En ellas se encuentran elementos como: *servlets*, páginas *JSP* y recursos estáticos como páginas *HTML*, imágenes, etc.

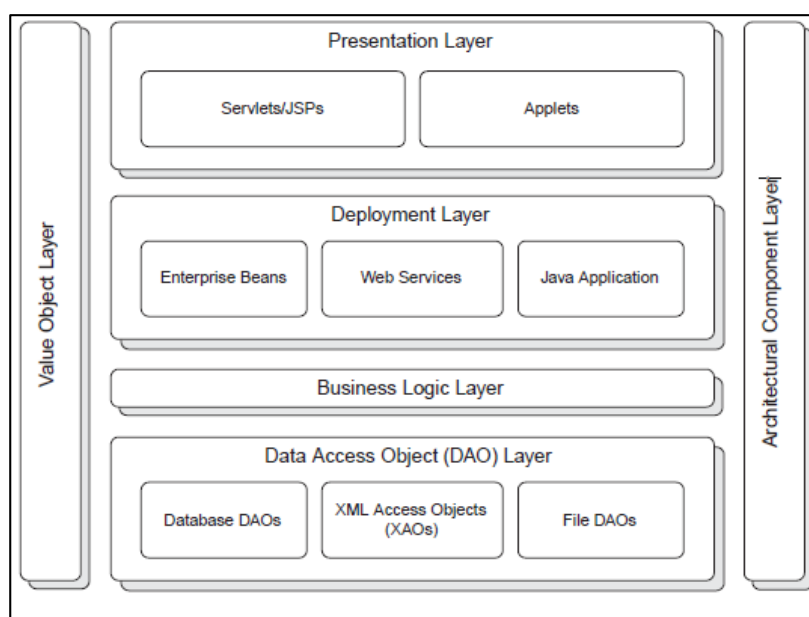


ILUSTRACIÓN 4: CAPAS DE LA ARQUITECTURA DE UNA APLICACIÓN J2EE

<i>Class Control</i>	Versión: 1.3
Documento de Diseño	30/06/2014

En cuanto a la implementación, una aplicación web JEE se conllustración en el fichero web.xml y se empaqueta en un fichero .war, que se ejecuta en el servidor de aplicaciones.

El *ServletContext* es un objeto que se crea cuando la aplicación web se inicia y que se destruye cuando la aplicación se detiene. Puede contener:

1. Parámetros de inicialización (especificados en web.xml).
2. Permite almacenar atributos (objetos) que pueden ser accedidos por todos los *servlets* y páginas JSP de la aplicación.
3. Este objeto permite a un *servlet* comunicarse con su contenedor.
4. Existe un único contexto por cada aplicación web.

Esto convierte al contexto en un mecanismo simple para comunicar información entre todos los *servlets* que corran en una misma aplicación web.

Ciclo de vida de los Servlets

Los *servlets* son gestionados por el contenedor, que decide cuándo crear y destruir un *servlet*. Al crear un *servlet*:

1. Antes de que éste responda a ninguna petición el contenedor va a ejecutar su método `init()`.
 - Este método permite al *servlet* realizar tareas de inicialización.
2. El contenedor sólo va a tener una única instancia del *servlet* para responder a todas las peticiones que le lleguen:
 - Múltiples *thread* pueden estar ejecutando el cuerpo del *servlet* para responder a múltiples peticiones de múltiples usuarios.

Cuando el contenedor destruya el *servlet*:

1. Antes de destruirlo invocará al método `destroy()`
2. Este método permite al *servlet* realizar cualquier labor de limpieza de recursos que haya podido adquirir.

MVC

Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Fue descrito por primera vez en 1979 por Trygve Reenskaug, trabajador de Smalltalk, en unos laboratorios de investigación de Xerox.

El patrón MVC se ve frecuentemente en aplicaciones web, y será seguido en este caso para desarrollar la aplicación *Class Control*. La vista suele estar compuesta de páginas HTML (si son estáticas) o JSP (datos dinámicos), el modelo es el sistema de gestión de Base de Datos junto al Object Relational Mapping (ORM) que mapea en objetos los datos relacionales almacenados en el sistema gestor de Base de Datos, y la lógica de negocio es el controlador, encargado de recibir los eventos de entrada desde la vista. A continuación se detallan más los elementos.

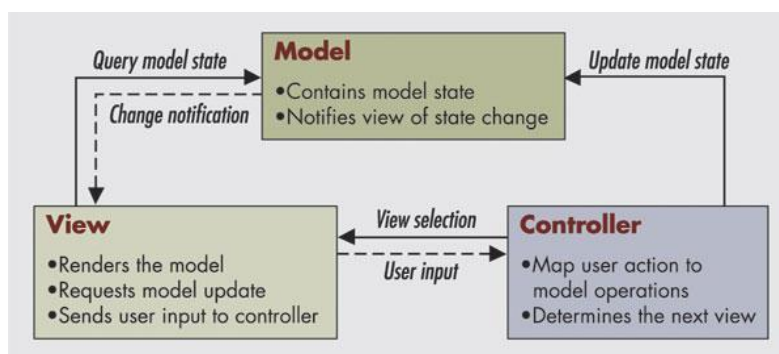


ILUSTRACIÓN 5: PATRÓN MVC DETALLADO

<i>Class Control</i>	Versión: 1.3
Documento de Diseño	30/06/2014

- **Vista:** presenta el modelo en un formato adecuado para interactuar (interfaz de usuario). Responsable de la lógica de presentación y captura de datos de nuestro sistema al exterior y viceversa. Este elemento se implementa con páginas HTML y páginas JSP.

- **Controlador:** responsable de la lógica operacional de negocio. Traslada las peticiones de la Vista al Modelo, y según la respuesta, la redirecciona o no (según sea MVC activo o pasivo) a la Vista. Carga objetos y opera con ellos.

Responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y, probablemente, a la vista. Además facilita el mantenimiento y la modificación de las aplicaciones.

El controlador se implementa con *servlets* que realizan los siguientes pasos:

1. Controlar el flujo de la aplicación.
2. Leer los parámetros del objeto *request*.
3. Actualizar los datos del modelo.
4. Enviar el modelo a una página JSP para presentar la información (vista).

- **Modelo:** contiene la lógica de negocio real, el dominio de la aplicación (VO: *Value Object*) con sus clases *get* y *set*, y los objetos de acceso a datos (DAO) que implementen las operaciones CRUD (*Create, Read, Update, Delete*). Esto a grandes rasgos, ya que dependiendo de la aplicación, además de los patrones VO y DAO, se puede incorporar un *Facade*. Esta capa interactúa o bien directamente (por ejemplo, mediante JDBC) o a través de la capa de persistencia, con servidores de bases de datos, LDAP...

El modelo se implementa mediante clases Java (JavaBeans o Enterprise Java Beans).

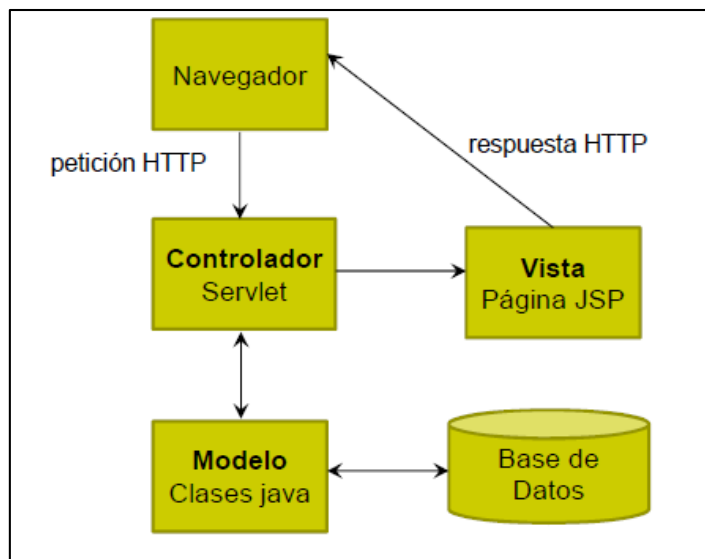


ILUSTRACIÓN 6: INTERACCIÓN EN EL PATRÓN MVC EN UNA APLICACIÓN WEB

1.2. Diseño de la Arquitectura

1.2.1. Descomposición en subsistemas

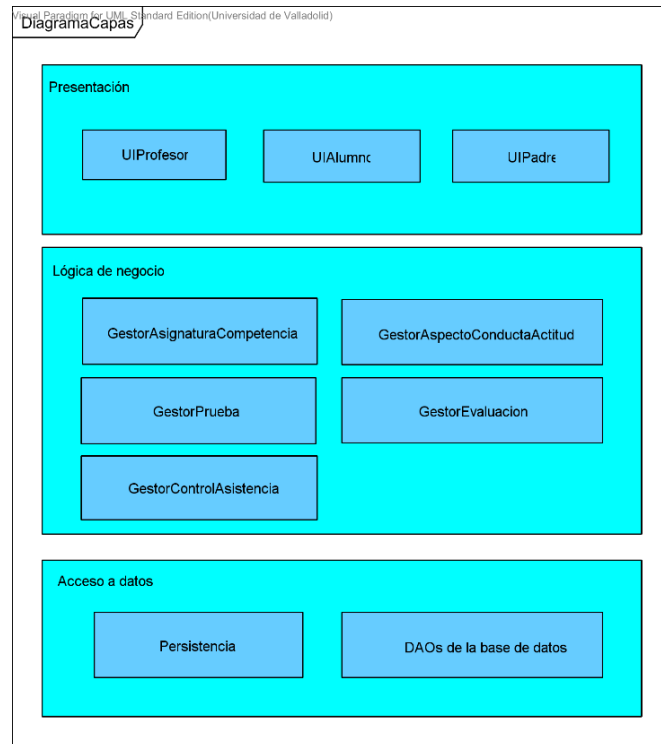


ILUSTRACIÓN 7: DESCOMPOSICIÓN EN SUBSISTEMAS

1.2.2. Topología del sistema

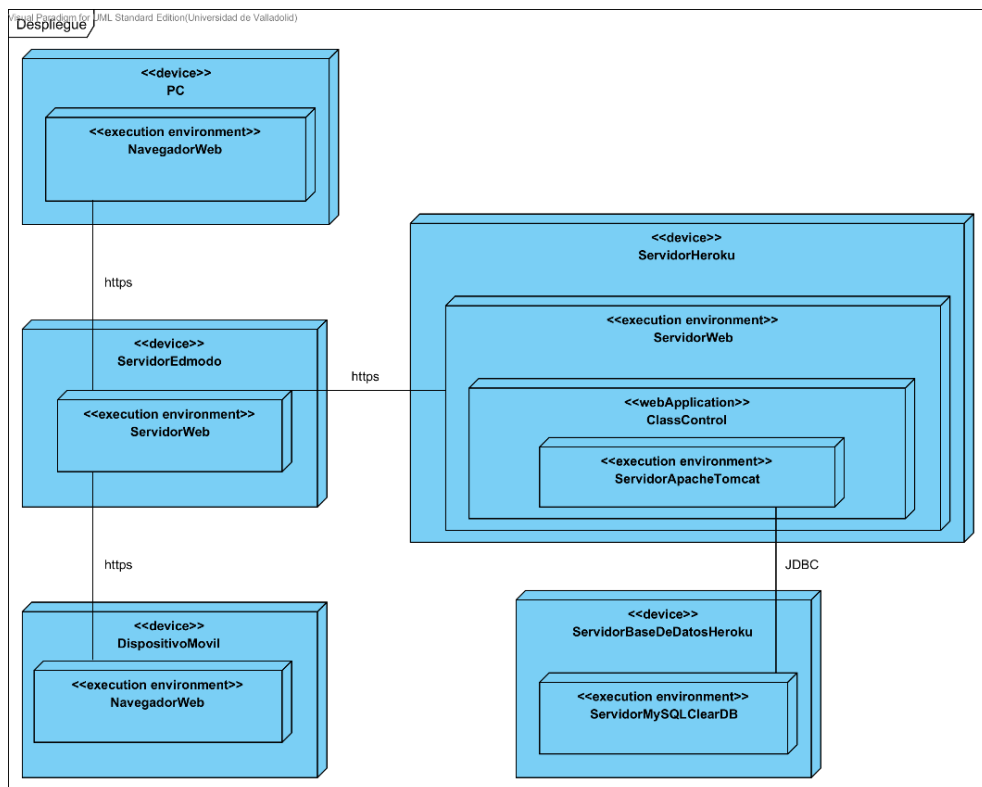


ILUSTRACIÓN 8: DIAGRAMA DE DESPLIEGUE DEL SISTEMA

<i>Class Control</i>	Versión: 1.3
Documento de Diseño	30/06/2014

1.2.3. Gestión de la persistencia

Para la parte de persistencia de datos, se cuenta con un servidor de base de datos MySQL en la nube, proporcionado por la extensión *ClearDB*^[2] de la plataforma *Heroku* (en su versión gratuita), con las tablas necesarias para el almacenamiento de los datos persistentes necesarios descritos en la **Ilustración 35: Diagrama Relacional de la Base De Datos**.

Todos los datos almacenados serán complementarios a los que almacene la plataforma de *edmodo* y en ningún caso serán redundantes para evitar incoherencias.

Se implementarán copias de seguridad nocturnas gracias al *plugin* descrito anteriormente.

1.2.4. Aspectos globales y de seguridad

El acceso a la aplicación web será mediante protocolo *https*, por lo tanto será necesario el uso de certificados por parte del servidor para el establecimiento seguro de la conexión con los clientes. De esta forma se consigue que los datos viajen de manera segura y encriptada por la red.

Nadie que no pertenezca a la plataforma *edmodo*, haya iniciado sesión en ella y accedido desde la misma a *Class Control* podrá visualizar ningún dato del sistema.

Para ello el sistema comprueba en cada petición los datos del usuario si son de *edmodo*, en caso de no serlo se muestra al usuario un mensaje explicando la negación de acceso.

1.2.5. Aspectos de rendimiento y tamaño

En cuanto al rendimiento del sistema, *Heroku* proporciona un procesador gratuito^[3] que es capaz de soportar hasta 256 procesos e hilos a la vez. Además de contar con 512 MB de memoria RAM.

Además estará el servidor activo todo el tiempo que sea necesario sin coste alguno. Para ver más detalles del servidor de *Heroku* ver la referencia^[3].

En cuanto a la capacidad de almacenamiento en la base de datos, debido a la restricción del *plugin ClearDB* será como máximo 5MB, y soportará 10 conexiones concurrentes.

1.3. Diseño de los subsistemas

1.3.1. Vista de casos de uso. Realización de casos de uso

1.3.1.1. Diagrama de Casos de Uso

Respecto al diagrama de Casos de Uso del *documento de Análisis de Requisitos* se indica que no se pueden realizar los siguientes Casos de Uso debido a que la plataforma *edmodo* no lo permite a través de su API, pero sí desde ella misma:

EliminarPrueba, *ModificarPrueba* y *EliminarAspectoConductaActitud*.

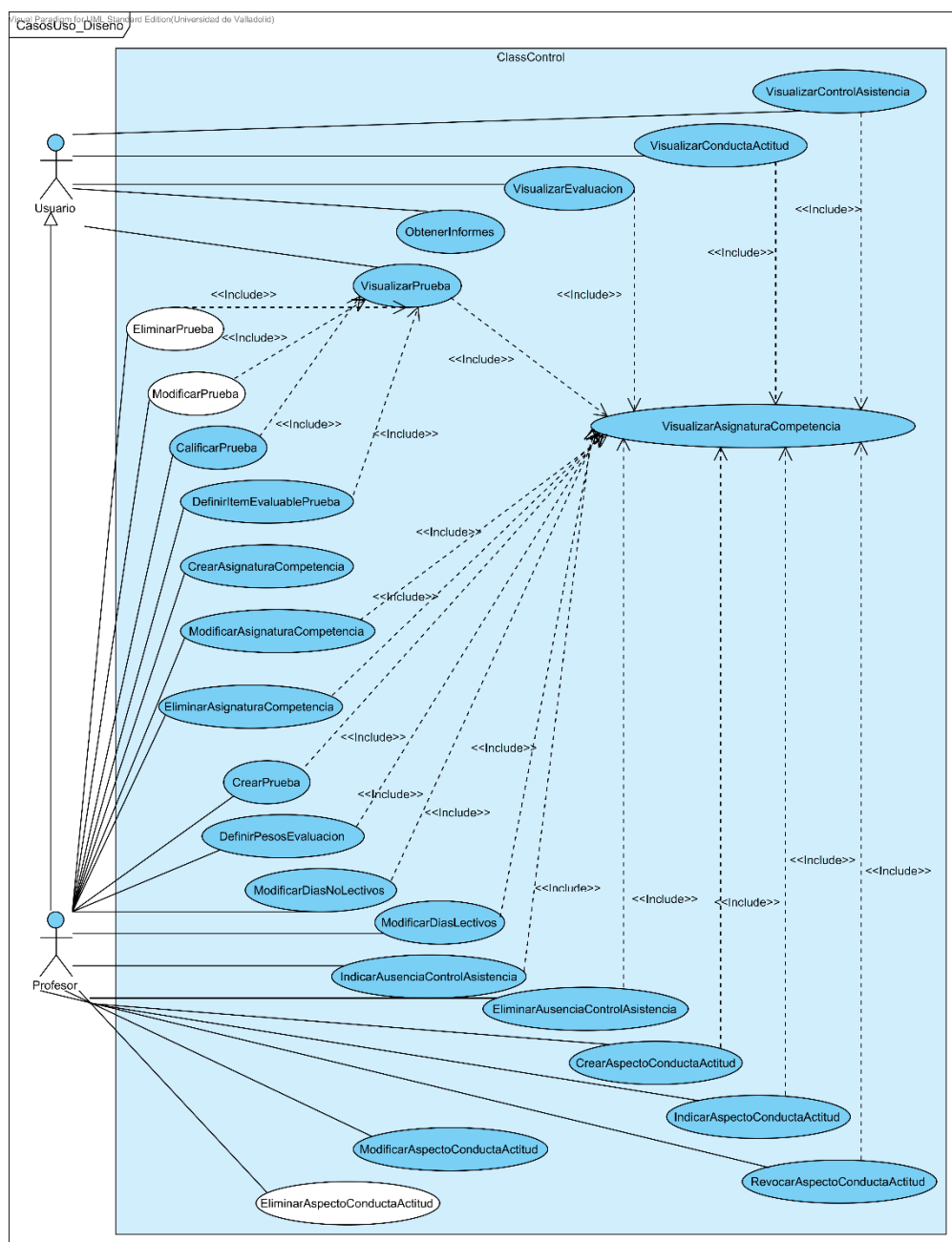


ILUSTRACIÓN 9: DIAGRAMA DE CASOS DE USO DE DISEÑO

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

1.3.1.2. Especificación de Casos de Uso de diseño

Se especifica el proceso de los Casos de Uso de cara a la tecnología utilizada y a la interfaz para el Usuario.

CU01	CrearAsignaturaCompetencia	
Versión	1.0.	
Dependencias	RF01, RF02, RF03 y RF04.	
Descripción	Un Profesor desea crear una asignatura o competencia nueva.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	<ol style="list-style-type: none"> 1. El caso de uso se inicia cuando el Profesor pulsa el botón “Añadir nueva asignatura o competencia”. 2. El sistema muestra un formulario con los datos a rellenar de la nueva asignatura o competencia: grupo de <i>edmodo</i> en un <i>combobox</i>, título en un área de texto, fechas de inicio y de fin en selectores de fechas, períodos de evaluación en un área de texto numérica junto a sus fechas de inicio y fin en selectores de fechas, descripción opcional en un área de texto y si está asociada o no a una asignatura o competencia primaria existente en un <i>combobox</i> y su peso en un área de texto numérica. 3. El Profesor rellena los datos. 4. El sistema comprueba que no estén sin rellenar: el grupo de <i>edmodo</i>, el título, las fechas de inicio y fin y al menos un período de evaluación. El sistema comprueba que el grupo de <i>edmodo</i> exista, el título no esté repetido, las fechas de inicio sean inferior a las fechas de fin, que no haya más de cien periodos y si existe la asignatura o competencia asociada en caso de indicarlo. El sistema registra la nueva asignatura o competencia en la base de datos y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 5. El caso de uso finaliza. 	
Postcondición	La asignatura o competencia queda registrada en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón “Atrás” o sale de la página web actual en cualquier instante.
		1. Se avanza al paso 5 del flujo básico y el Caso de Uso queda sin efecto.
Excepciones	EX01	Datos incorrectos
		<ol style="list-style-type: none"> 1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 2 del flujo básico.
Frecuencia	Baja.	

TABLA 2: DESCRIPCIÓN DEL CU01

CU02	VisualizarAsignaturaCompetencia	
Versión	1.0.	
Dependencias	RF05.	
Descripción	Un Usuario desea visualizar las acciones disponibles de una asignatura o competencia.	
Actor primario	Usuario.	
Precondiciones	<ol style="list-style-type: none"> 1. El Usuario está autenticado en la plataforma <i>edmodo</i>. 2. Si el Usuario es un padre, es necesario haber seleccionado un hijo. 	
Flujo básico	<ol style="list-style-type: none"> 1. El caso de uso se inicia cuando el Usuario pulsa el botón “Visualizar asignatura o competencia”. 2. El sistema muestra todas las asignaturas y competencias asociadas al Usuario en forma de botones. 3. El Usuario pulsa sobre el botón de una asignatura o competencia. 	

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

	4. El sistema muestra las acciones disponibles de la asignatura o competencia en forma de botones. 5. El caso de uso finaliza.	
Postcondición	Ninguna.	
Flujo alternativo	FA01	El Usuario selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 5 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	Ninguna.	
Frecuencia	Alta.	

TABLA 3: DESCRIPCIÓN DEL CU02

CU03	ModificarAsignaturaCompetencia	
Versión	1.0.	
Dependencias	RF03, RF04 y RF06.	
Descripción	Un Profesor desea modificar los datos de una asignatura o competencia.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Profesor pulsa el botón "Modificar asignatura o competencia". 3. El sistema muestra los datos posibles de ser modificados: fechas de inicio y de fin en selectores de fechas, períodos de evaluación en un área de texto numérica junto a sus fechas de inicio y fin en selectores de fechas, descripción opcional en un área de texto y si está asociada o no a una asignatura o competencia primaria existente en un <i>combobox</i> . 4. El Profesor rellena los nuevos cambios. 5. El sistema comprueba que no estén sin indicar: las fechas de inicio y fin y al menos un período de evaluación. El sistema comprueba que las fechas de inicio sean inferior a las fechas de fin, que no haya más de cien periodos, y si existe la asignatura o competencia asociada en caso de indicarla. El sistema registra los cambios en la base de datos y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 6. El caso de uso finaliza.	
	La asignatura o competencia ha sido actualizada en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 6 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	EX01	Datos incorrectos
	1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 3 del flujo básico.	
Frecuencia	Baja.	

TABLA 4: DESCRIPCIÓN DEL CU03

CU04	EliminarAsignaturaCompetencia	
Versión	1.0.	
Dependencias	RF07.	
Descripción	Un Profesor desea eliminar una asignatura o competencia existente.	
Actor primario	Profesor.	
Precondiciones	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Profesor pulsa el botón "Eliminar asignatura o competencia".	

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

	<p>3. El sistema muestra una ventana de texto emergente con una advertencia de seguridad para confirmar la acción de eliminar.</p> <p>4. Si el Profesor confirma la acción</p> <p>4.1. El sistema comprueba que la asignatura o competencia existe. El sistema elimina la asignatura o competencia de la base de datos y todos los datos relacionados* con ella de la base de datos (ver EX01).</p> <p>* Nota: las pruebas y los aspectos de conducta y actitud de la asignatura o competencia deberán ser eliminados desde la plataforma <i>edmodo</i>.</p> <p>5. Sino</p> <p>5.1. Se avanza al paso 6 del flujo básico.</p> <p>6. El caso de uso finaliza.</p>	
Postcondición	La asignatura o competencia ha sido eliminada del sistema	
Flujo alternativo	FA01	El Profesor selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
		1. Se avanza al paso 6 del flujo básico y el Caso de Uso queda sin efecto.
Excepciones	EX01	Datos incorrectos
	<p>1. Se muestra una ventana de texto de advertencia al Profesor.</p> <p>2. Se avanza al paso 6 del flujo básico.</p>	
Frecuencia	Baja.	

TABLA 5: DESCRIPCIÓN DEL CU04

CU05	CrearPrueba	
Versión	1.0.	
Dependencias	RF08 y RF09.	
Descripción	Un Profesor desea crear una nueva prueba de una asignatura o competencia.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	<p>1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia.</p> <p>2. El Profesor pulsa el botón "Crear prueba".</p> <p>3. El sistema muestra los datos de creación de una nueva prueba: título en un área de texto, fecha en un selector de fecha, calificación máxima en un área de texto numérica, tipo de la misma en un <i>combobox</i> y descripción en un área de texto (habilitado si su tipo es <i>entrega de trabajo offline</i> u <i>online</i>).</p> <p>4. El Profesor rellena los datos oportunos.</p> <p>5. El sistema comprueba que se han indicado el título, la fecha y que sea correcta y esté en el periodo lectivo, la calificación máxima, el tipo de prueba y la descripción (si es una entrega de trabajo). El sistema registra la nueva prueba en la base de datos y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01).</p> <p>6. El caso de uso finaliza.</p>	
	La nueva prueba ha sido registrada en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
		1. Se avanza al paso 6 del flujo básico y el Caso de Uso queda sin efecto.
Excepciones	EX01	Datos incorrectos
	<p>1. Se muestra una ventana de texto de advertencia al Profesor.</p> <p>2. Se retorna al paso 3 del flujo básico.</p>	
Frecuencia	Alta.	

TABLA 6: DESCRIPCIÓN DEL CU05

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

CU06	VisualizarPrueba	
Versión	1.0.	
Dependencias	RF10 y RF11.	
Descripción	Un Usuario desea visualizar una nueva prueba de una asignatura o competencia.	
Actor primario	Usuario.	
Precondiciones	1. El Usuario está autenticado en la plataforma <i>edmodo</i> . 2. Si el Usuario es un padre, es necesario haber seleccionado un hijo.	
Flujo básico	1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Usuario pulsa el botón “Visualizar prueba”. 3. El sistema muestra todas las pruebas relacionadas de la asignatura o competencia en forma de botones. 4. El Usuario pulsa sobre el botón de la prueba. 5. El sistema muestra los datos de la prueba, y el comentario opcional si es visible para el padre o alumno, o ambos. 6. El caso de uso finaliza.	
Postcondición	La asignatura o competencia queda registrada en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón “Atrás” o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 6 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	Ninguna.	
Frecuencia	Alta.	

TABLA 7: DESCRIPCIÓN DEL CU06

CU07	ModificarPrueba
-------------	------------------------

En cuanto a este caso de uso, debido a que en la API de *edmodo* no existe manera de realizarlo, se detallan a continuación los pasos a seguir desde la propia plataforma para llevarlo a cabo:

1. Una vez dentro de la plataforma *edmodo* (como Profesor) se busca la prueba en el tablón general y a la derecha de la misma se pulsa sobre el icono de una rueda dentada.
2. Se elige “Editar mensaje” en el desplegable.



ILUSTRACIÓN 10: CU07 DESDE EDMODO. PASO 1

3. Se realizan los cambios oportunos.
4. Se aceptan los cambios pulsando sobre el botón “Guardar”.

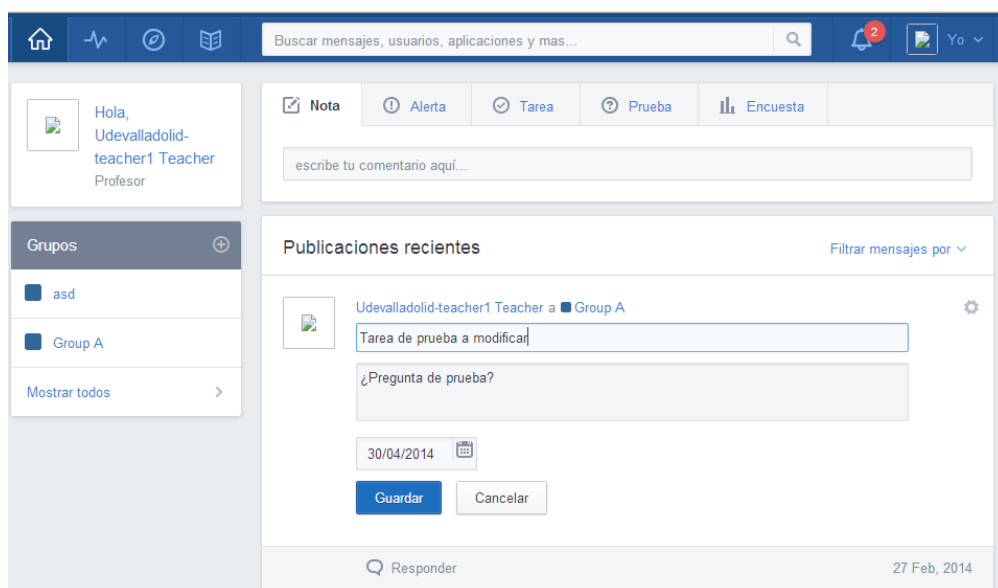


ILUSTRACIÓN 11: CU07 DESDE EDMODO. PASO 2

- La prueba ha sido modificada correctamente.

CU08	EliminarPrueba
------	----------------

En cuanto a este caso de uso, debido a que en la API de *edmodo* no existe manera de realizarlo, se detallan a continuación los pasos a seguir desde la propia plataforma para llevarlo a cabo:

- Una vez dentro de la plataforma *edmodo* (como Profesor) se busca la prueba en el tablón general y a la derecha de la misma se pulsa sobre el icono de una rueda dentada.
- Se elige “Eliminar mensaje” en el desplegable.

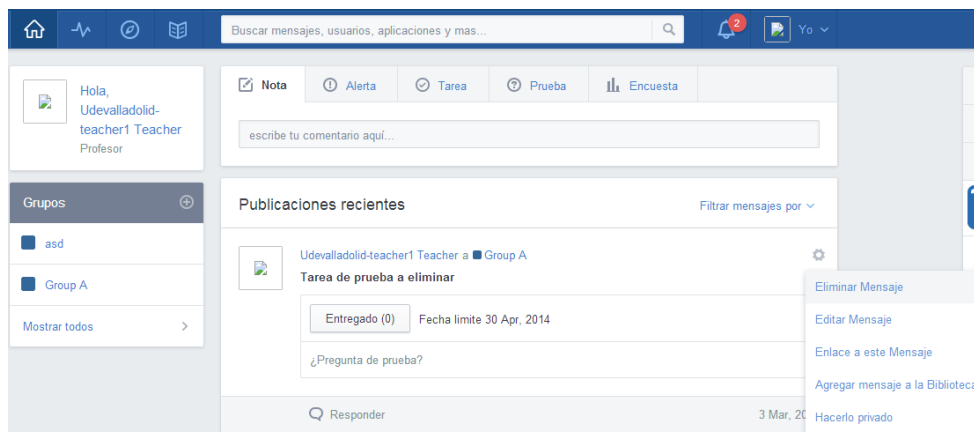


ILUSTRACIÓN 12: CU08 DESDE EDMODO. PASO 1

- Se elige “OK” en la ventana emergente.

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

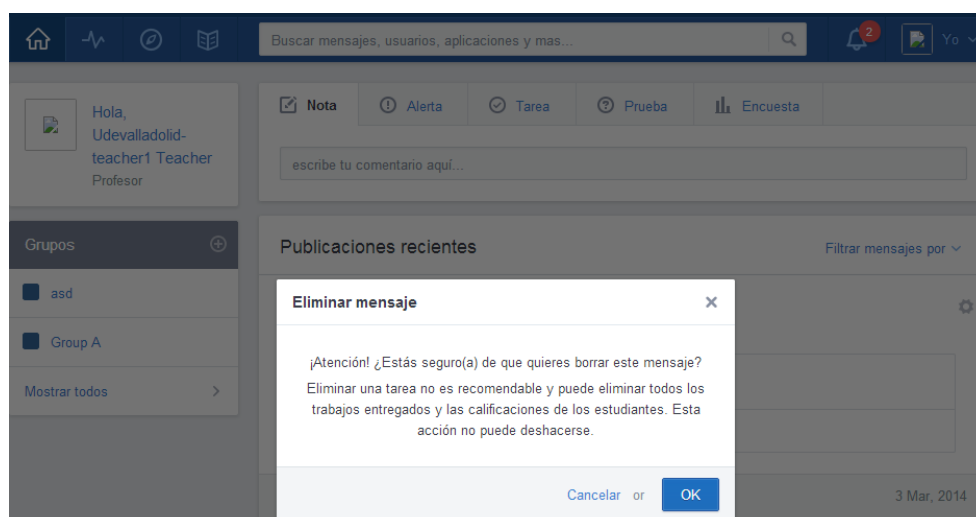


ILUSTRACIÓN 13: CU08 DESDE EDMODO. PASO 2

7. La prueba ha sido eliminada correctamente.

CU09	CalificarPrueba	
Versión	1.0.	
Dependencias	RF14, RF15 y RF16.	
Descripción	Un Profesor desea calificar una prueba evaluable.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	<ol style="list-style-type: none"> 1. Se ejecuta el Caso de Uso VisualizarPrueba. 2. El Profesor pulsa el botón "Calificar prueba". 3. El sistema muestra los datos a indicar en una tabla para cada alumno: valor obtenido en los ítems evaluables de la prueba en formato numérico, comentario opcional, a especificar si es visible para padre o alumno, o ambos, el valor obtenido en los ítems evaluables de la prueba y las bonificaciones o penalizaciones que computan en la media de la calificación en formato numérico. 4. El profesor rellena los datos. 5. El sistema comprueba que cada alumno exista, que el valor obtenido sea menor que el máximo posible en la prueba y que la media de todos los componentes que forman la calificación final no supere la máxima calificación. El sistema guarda la calificación y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 6. El caso de uso finaliza. 	
Postcondición	La calificación de la prueba ha sido guardada con éxito en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
		1. Se avanza al paso 6 del flujo básico y el Caso de Uso queda sin efecto.
Excepciones	EX01	Datos incorrectos
		<ol style="list-style-type: none"> 1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 3 del flujo básico.
Frecuencia	Media.	

TABLA 8: DESCRIPCIÓN DEL CU09

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

CU10	DefinirItemEvaluablePrueba	
Versión	1.0.	
Dependencias	RF17.	
Descripción	Un Profesor desea definir un ítem evaluable para una prueba.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	<ol style="list-style-type: none"> 1. Se ejecuta el Caso de Uso VisualizarPrueba. 2. El Usuario pulsa el botón “Definir ítem evaluable”. 3. El sistema muestra los datos a indicar: nombre del ítem en un área de texto y su peso en la prueba en un área de texto numérica. 4. El Profesor rellena los datos correspondientes. 5. El sistema comprueba que el nombre no esté vacío, y no esté repetido, que el tanto por ciento que indica su peso esté indicado entre 0 y 100 y que su peso junto a los demás elementos que constituyen la calificación de la prueba no supere 100%. El sistema registra el ítem y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 6. El caso de uso finaliza. 	
Postcondición	El ítem evaluable ha sido registrado al alumno en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón “Atrás” o sale de la página web actual en cualquier instante.
		1. Se avanza al paso 6 del flujo básico y el Caso de Uso queda sin efecto.
Excepciones	EX01	Datos incorrectos
		<ol style="list-style-type: none"> 1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 3 del flujo básico.
Frecuencia	Media.	

TABLA 9: DESCRIPCIÓN DEL CU10

CU11	DefinirPesosEvaluacion	
Versión	1.0.	
Dependencias	RF16, RF18 y RF19.	
Descripción	Un Profesor desea definir los pesos de una asignatura o competencia.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	<ol style="list-style-type: none"> 1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Profesor pulsa el botón “Definir pesos en la evaluación”. 3. El sistema muestra los periodos de evaluación con un área de texto para indicar su peso, las pruebas actuales de la asignatura o competencia en cada evaluación con un área de texto para indicar su peso en cada uno y además muestra un área de texto para el valor individual y máximo de aspectos positivos y negativos. 4. El Profesor modifica los pesos que desee. 5. El sistema comprueba que los tanto por ciento de los pesos estén indicados entre 0 y 100, que la suma de los periodos y el máximo de aspectos positivos no supere el 100%, que el peso total de las pruebas de cada periodo no supere el 100% y que cada aspecto negativo y positivo no sea superior al máximo respectivamente. El sistema registra los pesos de la evaluación y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 6. El caso de uso finaliza. 	
Postcondición	Los pesos de la evaluación quedan registrados en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón “Atrás” o sale de la página web actual en cualquier instante.

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

	1. Se avanza al paso 6 del flujo básico y el Caso de Uso queda sin efecto.
Excepciones	EX01 Datos incorrectos
	1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 3 del flujo básico.
Frecuencia	Baja.

TABLA 10: DESCRIPCIÓN DEL CU11

CU12	ModificarDiasNoLectivos	
Versión	1.0.	
Dependencias	RF21 y RF23.	
Descripción	Un Profesor desea modificar los días no lectivos de una asignatura o competencia existente.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Profesor pulsa el botón "Días no lectivos". 3. El sistema muestra las fechas no lectivas actuales en un selector de fechas. 4. El Profesor selecciona las fechas no lectivas en el selector de fechas. 5. El sistema comprueba que la(s) fecha(s) indicada(s) esté(n) dentro del período definido en la asignatura o competencia. El sistema guarda la(s) fecha(s) no lectiva(s) en la base de datos y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 6. El caso de uso finaliza.	
Postcondición	La(s) fecha(s) no lectiva(s) queda(n) registrada(s) en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
		1. Se avanza al paso 6 del flujo básico y el Caso de Uso queda sin efecto.
Excepciones	EX01	Datos incorrectos
		1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 3 del flujo básico.
Frecuencia	Baja.	

TABLA 11: DESCRIPCIÓN DEL CU12

CU13	ModificarDiasLectivos	
Versión	1.0.	
Dependencias	RF21 y RF24.	
Descripción	Un Profesor desea modificar los días lectivos de una asignatura o competencia existente.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Profesor pulsa el botón "Días lectivos". 3. El sistema muestra las fechas lectivas y lectivas actuales en un selector de fechas. 4. El Profesor selecciona las fechas lectivas en el selector de fechas. 5. El sistema comprueba que la(s) fecha(s) indicada(s) esté(n) dentro del período definido en la asignatura o competencia. El sistema guarda en la base de datos la(s) fecha(s) lectiva(s) y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 6. El caso de uso finaliza.	
Postcondición	La(s) fecha(s) lectiva(s) queda(n) registrada(s) en el sistema.	

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

Flujo alternativo	FA01	El Profesor selecciona el botón “Atrás” o sale de la página web actual en cualquier instante.
		1. Se avanza al paso 6 del flujo básico y el Caso de Uso queda sin efecto.
Excepciones	EX01	Datos incorrectos
		1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 3 del flujo básico.
Frecuencia		Baja.

TABLA 12: DESCRIPCIÓN DEL CU13

CU14	IndicarAusenciaControlAsistencia	
Versión	1.0.	
Dependencias	RF21, RF22 y RF25.	
Descripción	Un Profesor desea indicar una ausencia de un alumno a clase.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico		1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Profesor pulsa el botón “Indicar ausencias”. 3. El sistema muestra una lista de los alumnos de la asignatura en un <i>comobox</i> . 4. El Profesor selecciona el alumno al que desea aplicar la(s) ausencia(s). 5. El sistema muestra un calendario académico y las ausencias de dicho alumno (si las tuviera). 6. El Profesor selecciona la(s) nueva(s) fecha(s) de ausencia. 7. El sistema pregunta si quiere indicar más ausencias. 8. El sistema comprueba que la(s) fecha(s) indicada(s) esté(n) dentro del período lectivo de la asignatura o competencia. El sistema registra la ausencia al alumno seleccionado y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 9. El caso de uso finaliza.
Postcondición	La(s) ausencia(s) del alumno queda(n) registrada(s) en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón “Atrás” o sale de la página web actual en cualquier instante.
		1. Se avanza al paso 9 del flujo básico y el Caso de Uso queda sin efecto.
Excepciones	EX01	Datos incorrectos
		1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 5 del flujo básico.
Frecuencia		Media.

TABLA 13: DESCRIPCIÓN DEL CU14

CU15	EliminarAusenciaControlAsistencia	
Versión	1.0.	
Dependencias	RF21 y RF25.	
Descripción	Un Profesor desea eliminar una ausencia de un alumno.	
Actor primario	Profesor.	
Precondiciones	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico		1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Profesor pulsa el botón “Eliminar ausencias”. 3. El sistema muestra una lista de los alumnos de la asignatura en un <i>combobox</i> . 4. El Profesor selecciona el alumno que desea el alumno al que desea revocar la(s) ausencia(s). 5. El sistema muestra las ausencias de dicho alumno.

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

	6. El Profesor indica la(s) ausencia(s) que desea eliminar. 7. El sistema elimina la(s) ausencia(s) indicada(s) y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito. 8. El caso de uso finaliza.	
Postcondición	La(s) ausencia(s) del alumno queda(n) eliminada(s) del sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 8 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	Ninguna	
Frecuencia	Baja.	

TABLA 14: DESCRIPCIÓN DEL CU15

CU16	CrearAspectoConductaActitud	
Versión	1.0.	
Dependencias	RF26 y RF30.	
Descripción	Un Profesor desea crear un aspecto de conducta o actitud.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	1. El Profesor pulsa el botón "Crear aspecto de conducta y actitud". 2. El sistema muestra los datos a indicar: título en un área de texto, descripción en un área de texto, una dirección web de la imagen que le identifica en un área de texto y dos botones de radio para elegir el tipo al que pertenece. 3. El Profesor rellena los datos del formulario. 4. El sistema comprueba que hayan sido indicados el título, la descripción, la dirección y el tipo. También comprueba que el título no esté repetido y que se cumplen las restricciones RES07 y RES08 del <i>documento de especificación de requisitos</i> . El sistema registra el aspecto de conducta o actitud y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 5. El caso de uso finaliza.	
	El aspecto de conducta o actitud queda registrado en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 5 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	EX01	Datos incorrectos
	1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 2 del flujo básico.	
Frecuencia	Baja.	

TABLA 15: DESCRIPCIÓN DEL CU16

CU17	IndicarAspectoConductaActitud	
Versión	1.0.	
Dependencias	RF26, RF27 y RF28.	
Descripción	Un Profesor desea indicar un aspecto de conducta o actitud a un alumno.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Profesor pulsa el botón "Indicar aspectos de conducta y actitud". 3. El sistema muestra una lista con los alumnos de la asignatura en un <i>combobox</i> .	

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

	4. El Profesor selecciona un alumno. 5. El sistema muestra dos listas de aspectos positivos y negativos. 6. El Profesor indica aquellos aspectos que desee. 7. El sistema comprueba que los aspectos indicados existan. El sistema registra los aspectos de conducta o actitud en el alumno seleccionado y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 8. El caso de uso finaliza.	
Postcondición	El (los) aspecto(s) queda(n) asociado(s) al alumno en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 8 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	EX01	Datos incorrectos
	1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 3 del flujo básico.	
Frecuencia	Media.	

TABLA 16: DESCRIPCIÓN DEL CU17

CU18	RevocarAspectoConductaActitud	
Versión	1.0.	
Dependencias	RF26, RF29 y RF33.	
Descripción	Un Profesor desea revocar aspectos de conducta o actitud a alumnos.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	
Flujo básico	1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Profesor pulsa el botón "Revocar aspectos de conducta y actitud". 3. El sistema muestra una lista con los alumnos de la asignatura en un <i>combobox</i> . 4. El Profesor selecciona un alumno. 5. El sistema muestra los aspectos de conducta y actitud que actualmente posee ese alumno. 6. El Profesor indica aquellos aspectos a revocar. 7. El sistema comprueba que los aspectos indicados existan. El sistema revoca los aspectos de conducta o actitud del alumno y muestra un mensaje de texto al Profesor indicando que la acción ha sido llevada a cabo con éxito (ver EX01). 8. El caso de uso finaliza.	
	El (los) aspecto(s) de conducta o actitud del alumno queda(n) revocado(s).	
Flujo alternativo	FA01	El Profesor selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 8 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	EX01	Datos incorrectos
	1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 3 del flujo básico.	
Frecuencia	Baja.	

TABLA 17: DESCRIPCIÓN DEL CU18

CU19	ModificarAspectoConductaActitud	
Versión	1.0.	
Dependencias	RF26, RF31 y RF33.	
Descripción	Un Profesor desea modificar un aspecto de conducta o actitud existente.	
Actor primario	Profesor.	
Precondición	El Profesor está autenticado en la plataforma <i>edmodo</i> .	

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

Flujo básico	1. El Profesor pulsa el botón “Modificar aspecto de conducta y actitud”. 2. El sistema muestra las dos listas de aspectos de conducta y actitud de la asignatura o competencia. 3. El profesor indica el aspecto a modificar. 4. El sistema muestra los datos de dicho aspecto: el título, la descripción y la dirección de la imagen que le identifica. 5. El Profesor indica los cambios oportunos. 6. El sistema comprueba que hayan sido indicados el título, la descripción y la dirección. También comprueba que el título no esté repetido y que se cumplen las ¡Error! No se encuentra el origen de la referencia. y ¡Error! No se encuentra el origen de la referencia. El sistema registra los cambios del aspecto de conducta o actitud y comunica al Profesor que la acción ha sido llevada a cabo con éxito (ver EX01). 7. El caso de uso finaliza.	
Postcondición	El aspecto de conducta o actitud queda modificado en el sistema.	
Flujo alternativo	FA01	El Profesor selecciona el botón “Atrás” o sale de la página web actual en cualquier instante.
		1. Se avanza al paso 8 del flujo básico y el Caso de Uso queda sin efecto.
Excepciones	EX01	Datos incorrectos
		1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 3 del flujo básico.
Frecuencia	Baja.	

TABLA 18: DESCRIPCIÓN DEL CU19

CU20	EliminarAspectoConductaActitud
------	--------------------------------

En cuanto a este caso de uso, debido a que en la API de *edmodo* no existe manera de realizarlo, se detallan a continuación los pasos a seguir desde la propia plataforma para llevarlo a cabo:

1. Una vez dentro de la plataforma *edmodo* (en el *dashboard* del desarrollador) se pulsa sobre el botón “+Add/Edit badge” (por lo que se aprecia, de momento la traducción de *edmodo* no es completa al castellano).

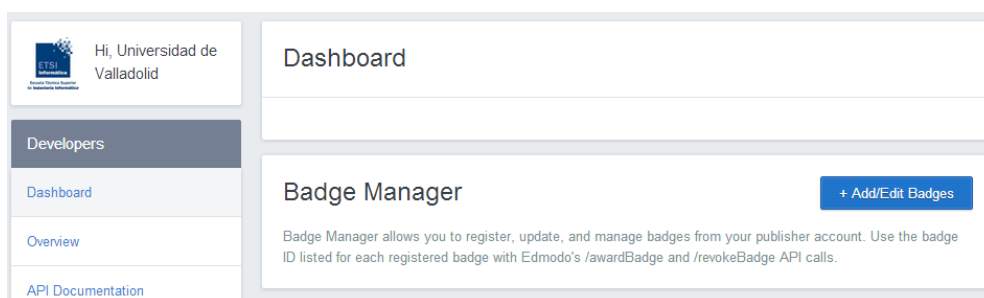


ILUSTRACIÓN 14: CU20 DESDE EDMODO. PASO 1

2. Se elige el aspecto de conducta o actitud creado en el apartado “Badges” (por lo que se aprecia, de momento la traducción de *edmodo* no es completa al castellano).

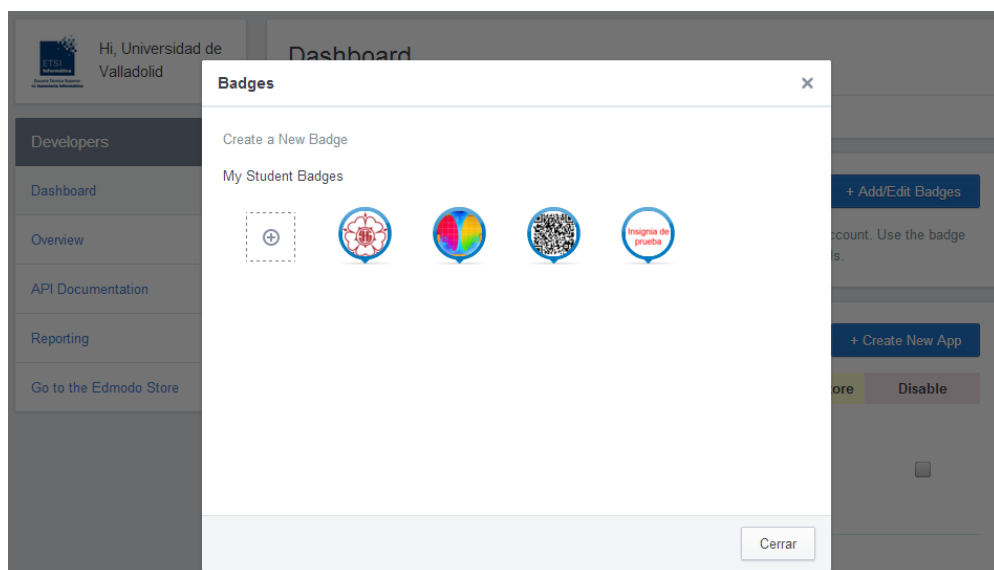


ILUSTRACIÓN 15: CU20 DESDE EDMODO. PASO 2

- Una vez seleccionado, se debe elegir la opción “Eliminar insignia”.

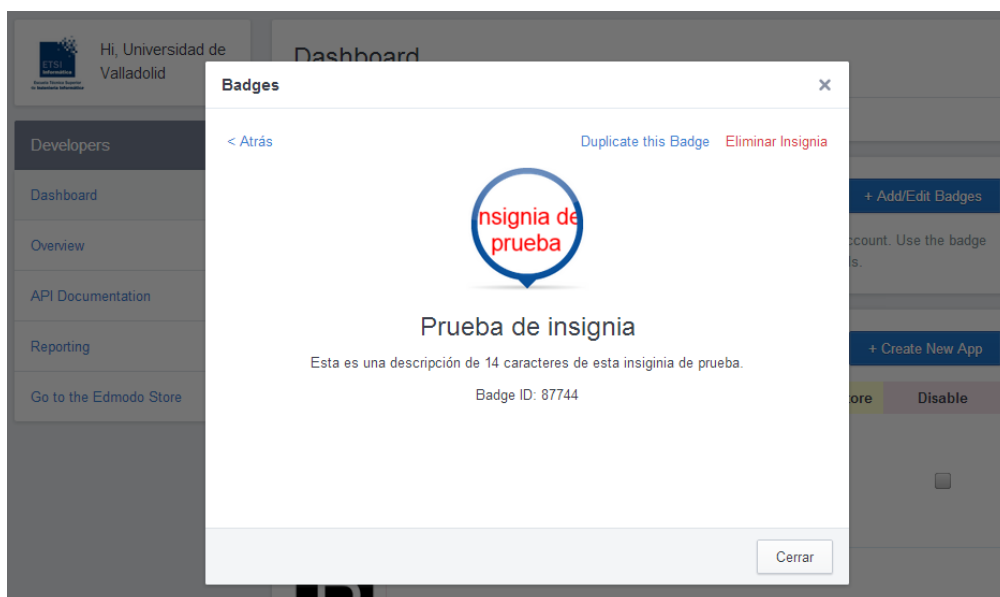


ILUSTRACIÓN 16: CU20 DESDE EDMODO. PASO 3

- Se elige la opción “Aceptar” en la ventana emergente.

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

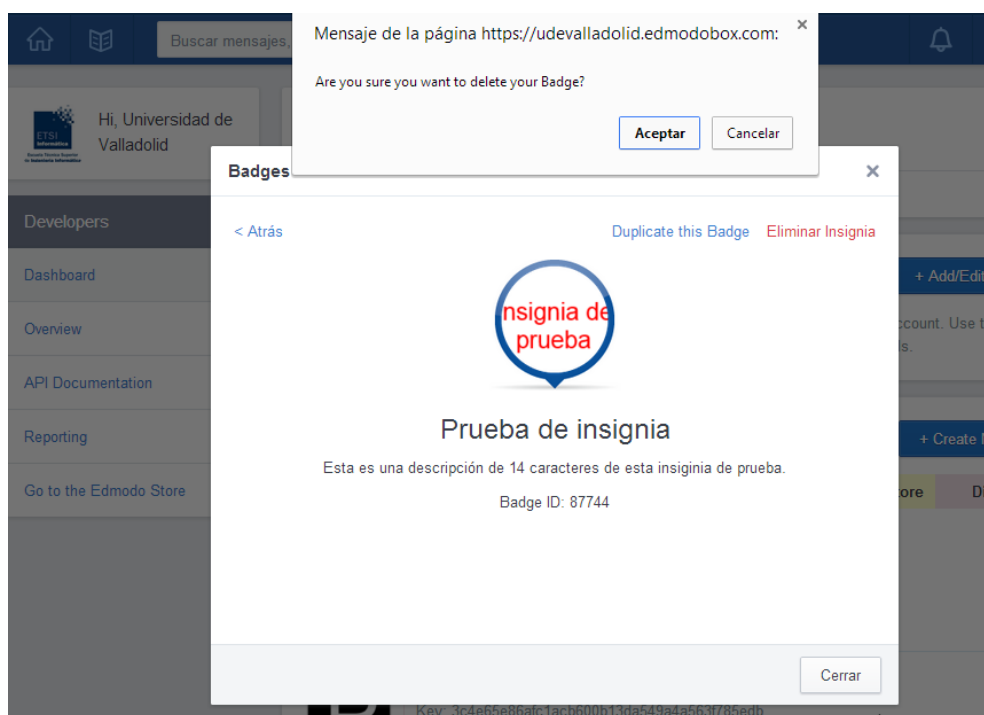


ILUSTRACIÓN 17: CU20 DESDE EDMODO. PASO 4

5. El aspecto de conducta o actitud ha sido eliminado correctamente.

CU21	VisualizarControlAsistencia	
Versión	1.0.	
Dependencias	RF25.	
Descripción	Un Usuario desea visualizar el control de asistencia asociado a un alumno.	
Actor primario	Usuario.	
Precondiciones	1. El Usuario está autenticado en la plataforma <i>edmodo</i> . 2. Si el Usuario es un padre, es necesario haber seleccionado un hijo. 3. Si el Usuario es un Profesor, es necesario haber seleccionado un alumno.	
Flujo básico	1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Usuario pulsa el botón “Visualizar control de asistencia” de la asignatura o competencia seleccionada. 3. El sistema muestra los días de ausencia relacionados con el alumno seleccionado en un visualizador de fechas. 4. El caso de uso finaliza.	
Postcondición	Ninguna.	
Flujo alternativo	FA01	El Profesor selecciona el botón “Atrás” o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 4 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	Ninguna.	
Frecuencia	Baja.	

TABLA 19: DESCRIPCIÓN DEL CU21

CU22	VisualizarConductaActitud
Versión	1.0.
Dependencias	RF33.

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

Descripción	Un Usuario desea visualizar el control de conducta y actitud asociado a un alumno.	
Actor primario	Usuario.	
Precondiciones	1. El Usuario está autenticado en la plataforma <i>edmodo</i> . 2. Si el Usuario es un padre, es necesario haber seleccionado un hijo. 3. Si el Usuario es un Profesor, es necesario haber seleccionado un alumno.	
Flujo básico	1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Usuario pulsa el botón “Visualizar control de conducta y actitud”. 3. El sistema muestra los aspectos de conducta y actitud asociados al alumno seleccionado en forma de botones. 4. El caso de uso finaliza.	
Postcondición	Ninguna.	
Flujo alternativo	FA01	El Profesor selecciona el botón “Atrás” o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 4 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	Ninguna.	
Frecuencia	Media.	

TABLA 20: DESCRIPCIÓN DEL CU22

CU23	VisualizarEvaluacion	
Versión	1.0.	
Dependencias	RF20.	
Descripción	Un Usuario desea visualizar la evaluación de un alumno de una asignatura o competencia.	
Actor primario	Usuario.	
Precondiciones	1. El Usuario está autenticado en la plataforma <i>edmodo</i> . 2. Si el Usuario es un padre, es necesario haber seleccionado un hijo. 3. Si el Usuario es un Profesor, es necesario haber seleccionado un alumno.	
Flujo básico	1. Se ejecuta el Caso de Uso VisualizarAsignaturaCompetencia. 2. El Usuario pulsa el botón “Visualizar evaluación” de la asignatura o competencia seleccionada. 3. El sistema muestra los datos de los elementos de la evaluación de la asignatura o competencia asociados al alumno seleccionado. 4. El caso de uso finaliza.	
Postcondición	Ninguna.	
Flujo alternativo	FA01	El Usuario selecciona el botón “Atrás” o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 4 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	Ninguna.	
Frecuencia	Media.	

TABLA 21: DESCRIPCIÓN DEL CU23

CU24	ObtenerInformes	
Versión	1.0.	
Dependencias	RF20, RF25, RF33, RF34 y RF35.	
Descripción	Un Usuario desea obtener informes de control de asistencia, conducta, actitud y evaluación de un alumno en una asignatura o competencia.	
Actor primario	Usuario.	
Precondiciones	1. El Usuario está autenticado en la plataforma <i>edmodo</i> . 2. Si el Usuario es un padre, es necesario haber seleccionado un hijo.	

<i>Class Control</i>	Versión: 1.3
Documento de Diseño	30/06/2014

	3. Si el Usuario es un Profesor, es necesario haber seleccionado un alumno.	
Flujo básico	1. El caso de uso comienza cuando el Usuario pulsa el botón "Obtener informes". 2. El sistema muestra las opciones disponibles para generarlos: si se quiere o no informes de control de asistencia, conducta y actitud y de evaluación mediante <i>checkboxes</i> , y además muestra un selector de fechas de inicio y fin con las fechas por defecto 3. El Usuario indica los apartados del informe que desea. 4. El sistema comprueba que la fecha de inicio es menor a la fecha de fin (en caso de indicarse esta última) y que ambas son correctas respecto a las fechas de inicio y fin de la asignatura o competencia. El sistema genera un informe con los datos indicados. 5. El caso de uso finaliza.	
Postcondición	Ninguna.	
Flujo alternativo	FA01	El Usuario selecciona el botón "Atrás" o sale de la página web actual en cualquier instante.
	1. Se avanza al paso 5 del flujo básico y el Caso de Uso queda sin efecto.	
Excepciones	EX01	Datos incorrectos
	1. Se muestra una ventana de texto de advertencia al Profesor. 2. Se retorna al paso 2 del flujo básico.	
Frecuencia	Baja.	

TABLA 22: DESCRIPCIÓN DEL CU24

1.3.1.3. Diagramas de interacción

A continuación se muestra cómo interactúa el sistema con la base de datos, y que será llevado a cabo por todos aquellos casos de uso que interactúen con la misma. Quedará reflejado en su respectivo diagrama de secuencia mediante el calificador UML “ref”.

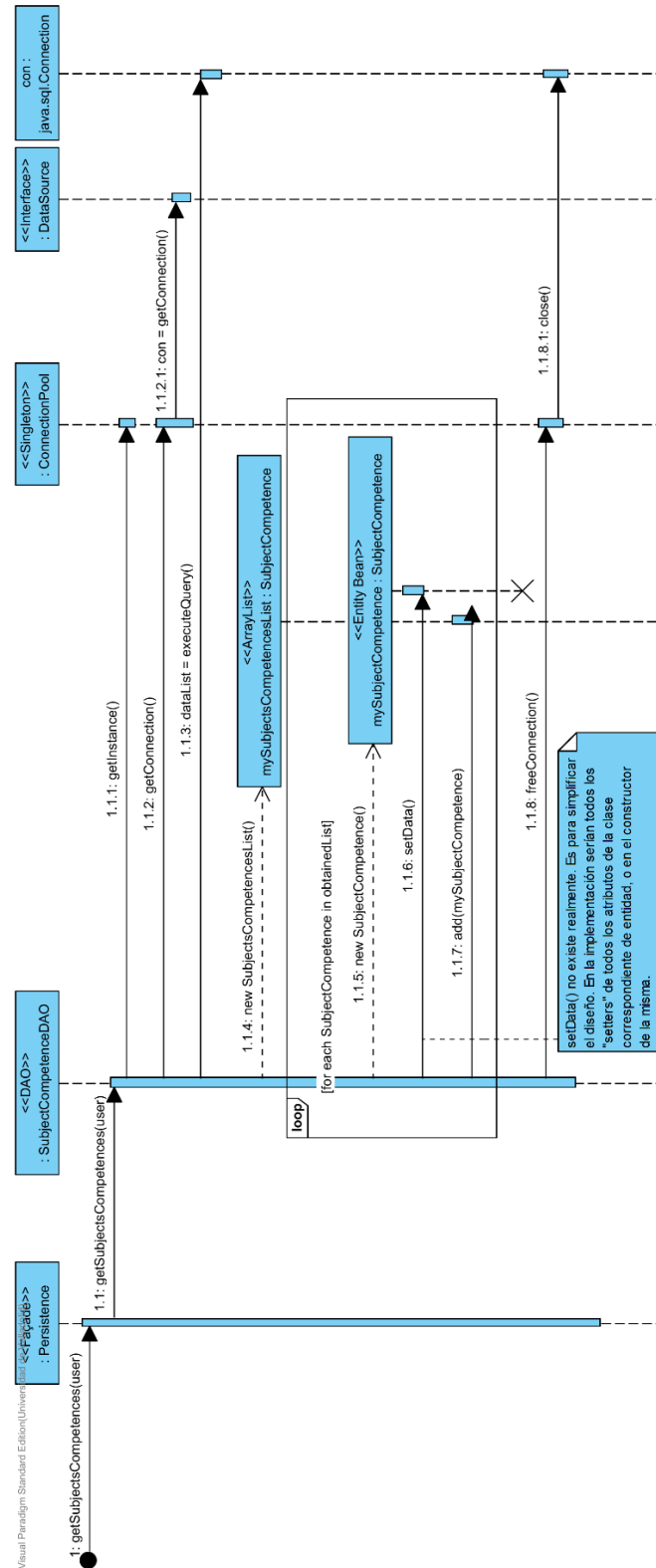


ILUSTRACIÓN 18: DIAGRAMA DE SECUENCIA DE EJEMPLO DAO

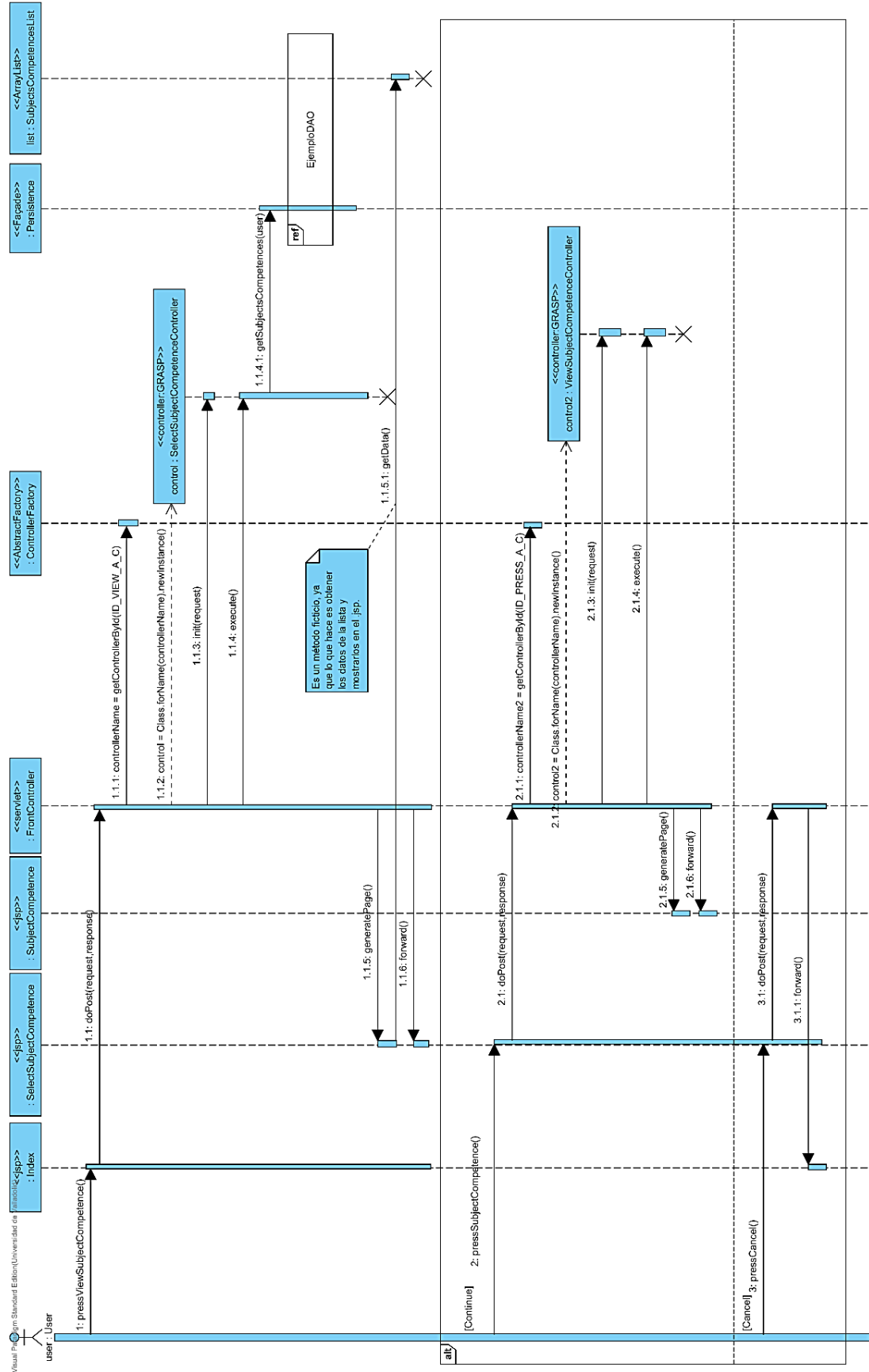


ILUSTRACIÓN 20: DIAGRAMA DE SECUENCIA DEL CU02

1.3.1.4. Diagrama de clases de diseño

Se muestra a continuación el diagrama de clases de Diseño con las interacciones entre las mismas. Después se irán detallando algunas de ellas, ya que, por motivos de espacio, no se puede mostrar todos los atributos, métodos y demás detalles de todos los elementos de la aplicación en un único diagrama.

Nótese que debido a la complejidad en número de elementos de la aplicación *Class Control*, en la **ILUSTRACIÓN 21** sólo se muestran aquellos elementos relacionados con la selección, visualización y creación de asignaturas o competencias, y en las ilustraciones consecutivas se mostrarán más elementos, agrupados por funcionalidad común, con sus características.

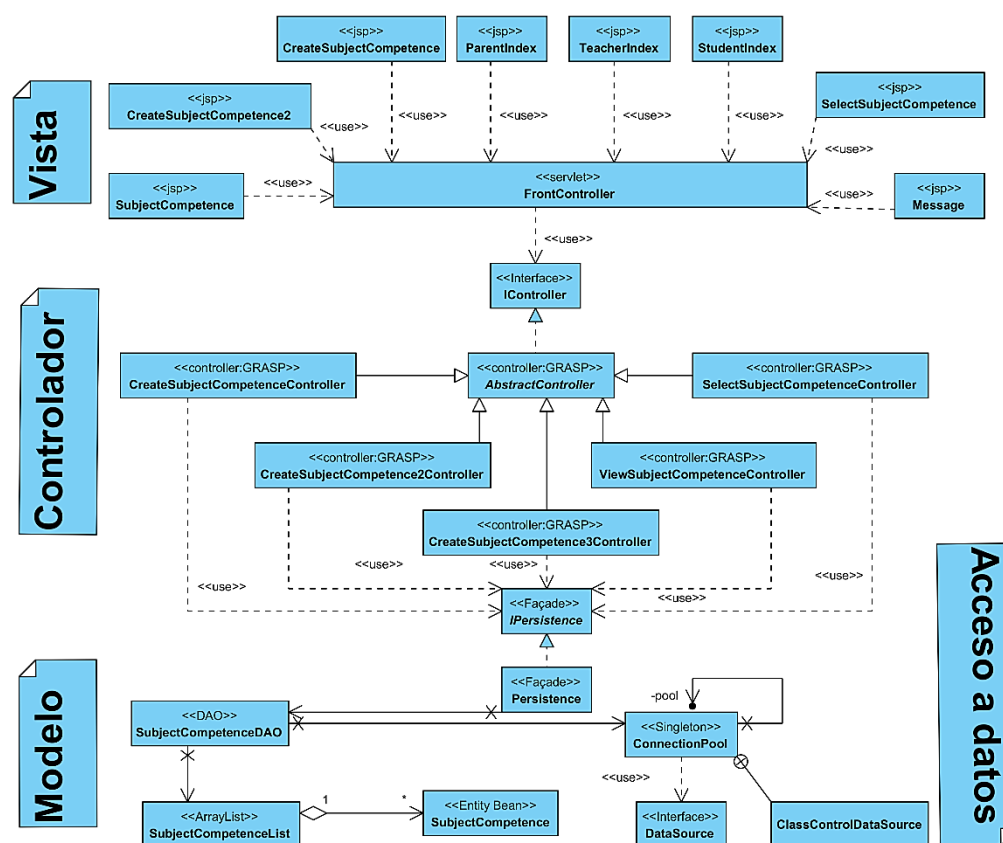


ILUSTRACIÓN 21: DIAGRAMA DE CLASES DE DISEÑO

Se muestran a continuación en la **ILUSTRACIÓN 22** los controladores encargados de la selección, visualización y creación de asignaturas y competencias. Por motivos obvios de espacio, no se muestran los demás controladores de la aplicación, que además, siguen el mismo patrón.

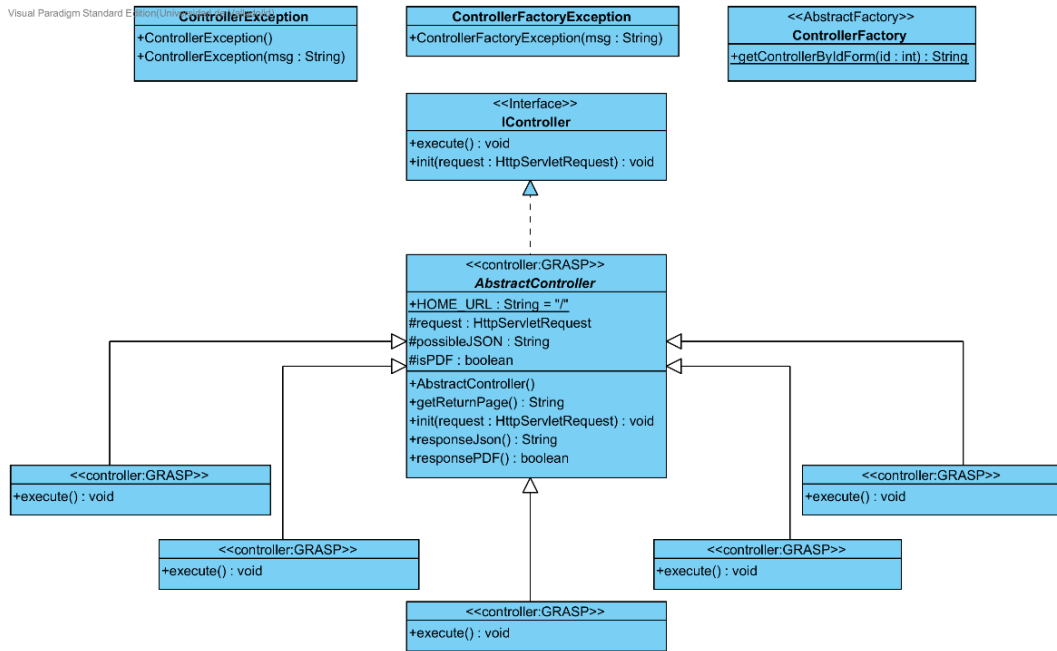


ILUSTRACIÓN 22: CLASES CONTROLLER

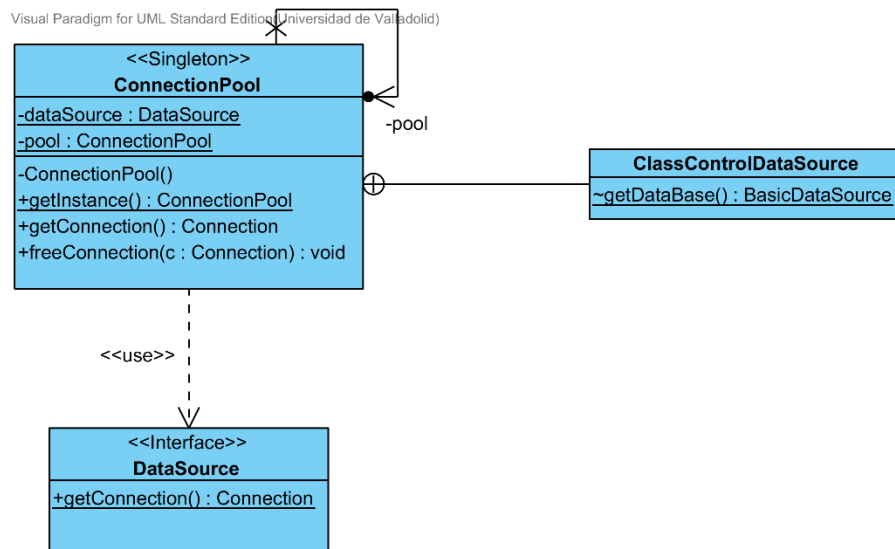


ILUSTRACIÓN 23: ACCESO A LA BASE DE DATOS

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014



ILUSTRACIÓN 24: MODELADO DE LA PERSISTENCIA

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

Visual Paradigm for UML Standard Edition(University of Valladolid)

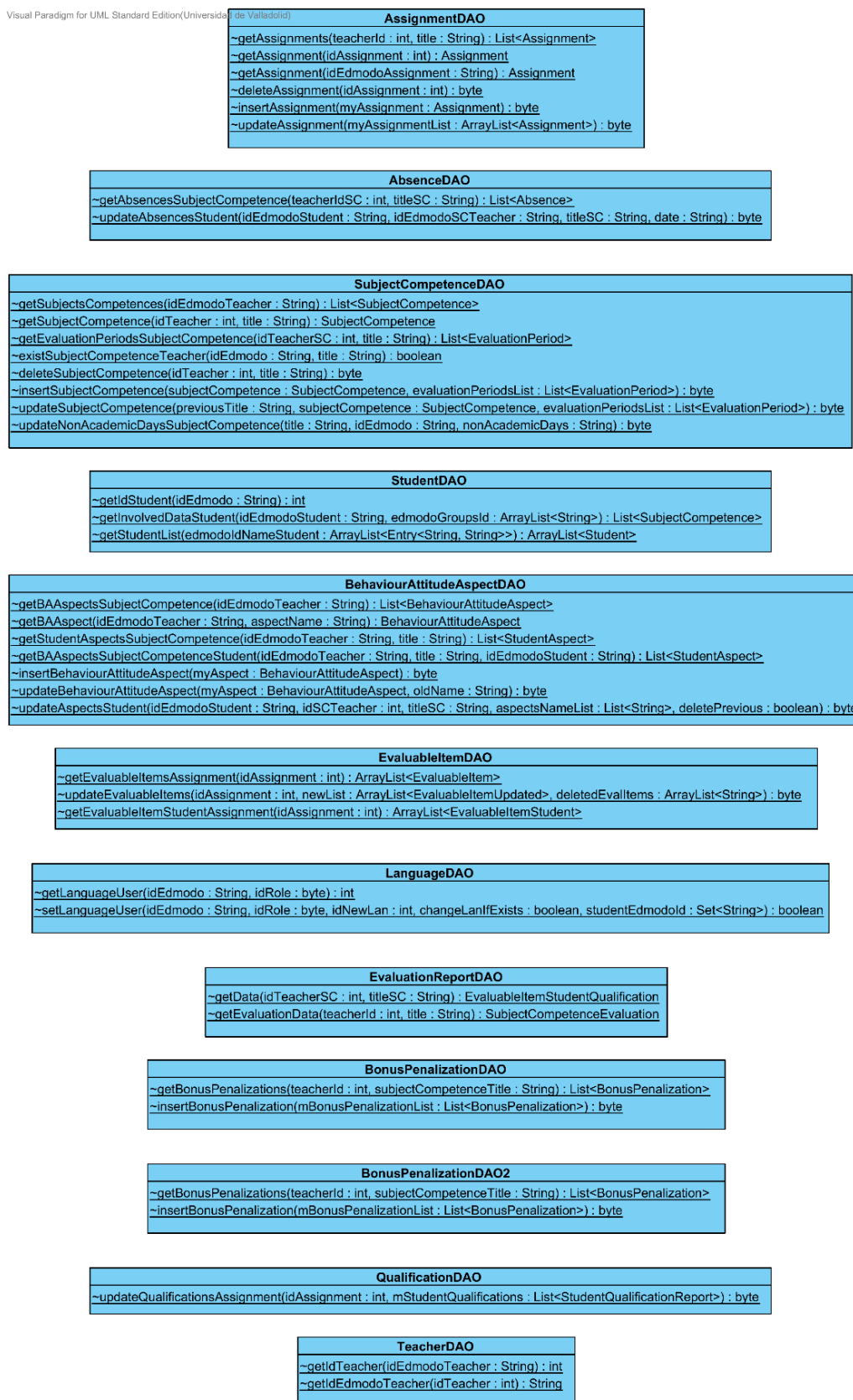


ILUSTRACIÓN 25: DAOs

Visual Paradigm for UML, Standard Edition (Universidad de Valladolid)

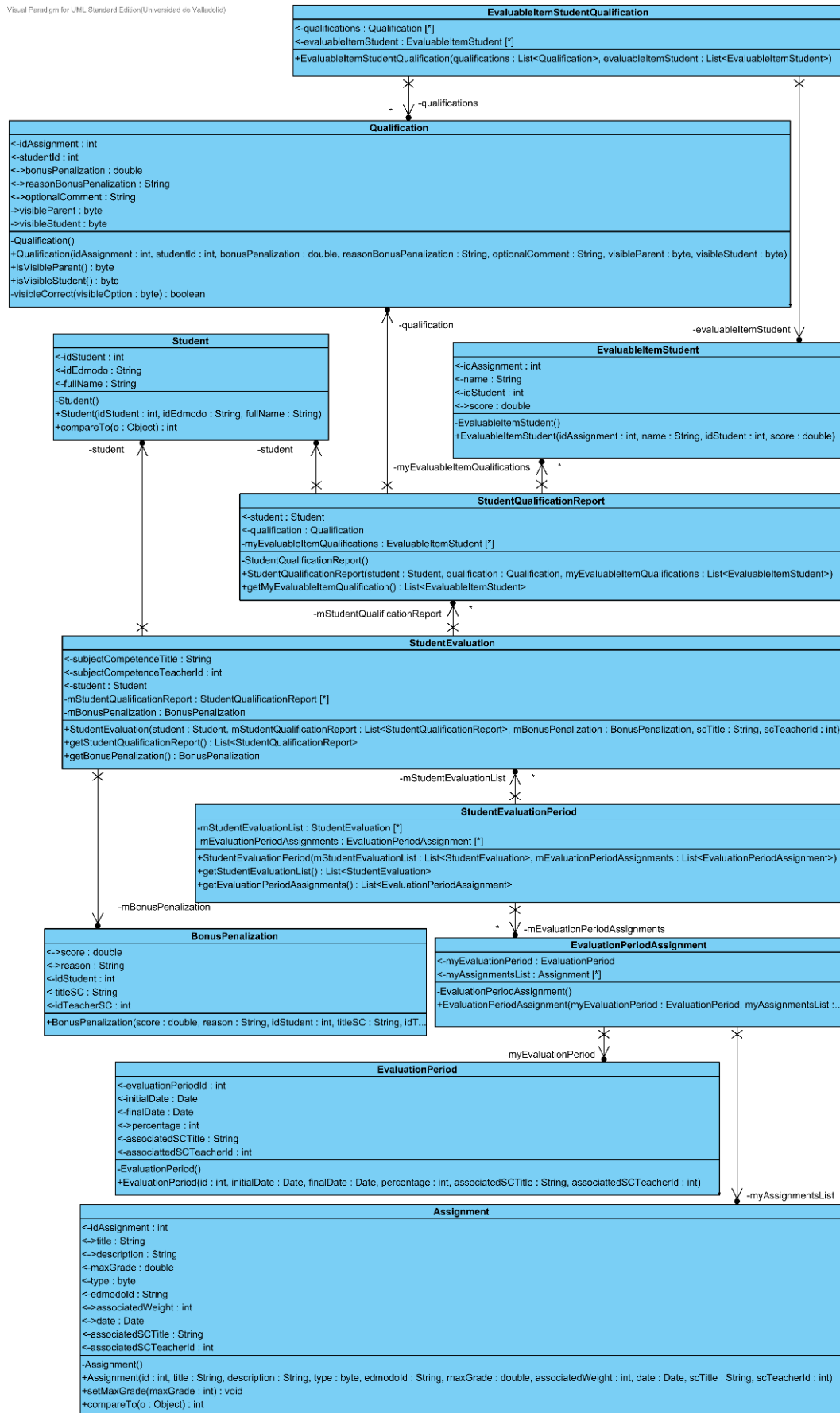


ILUSTRACIÓN 26: CLASES ENTITY – PARTE 1

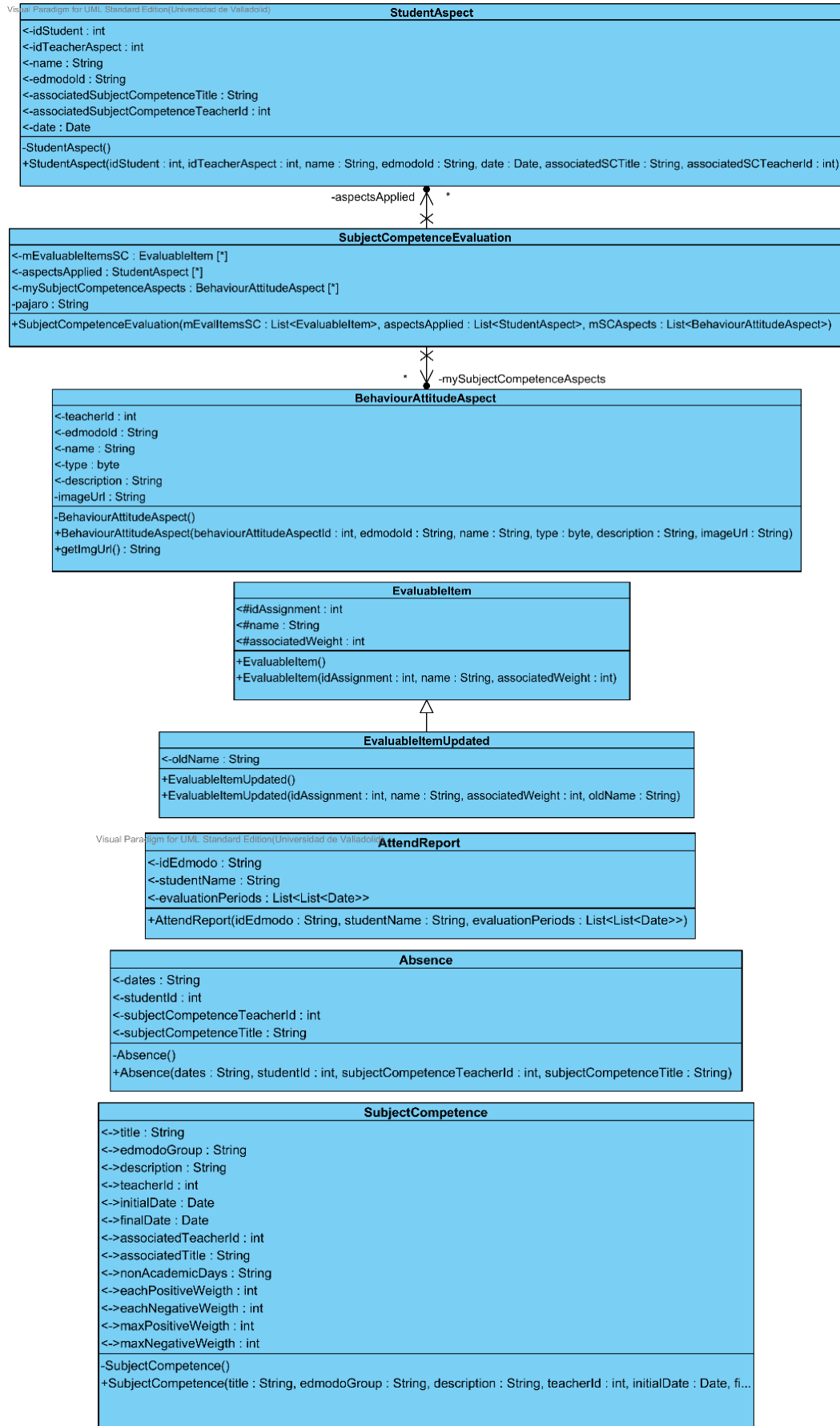


ILUSTRACIÓN 27: CLASES ENTITY – PARTE 2

Visual Paradigm for UML Standard Edition(Universidad de Valladolid)

```

EdmodoAPI
-+edmodoCall(url : String, method : String) : StringBuilder
+launchRequests(launchKey : String) : String
+groupsForUser(accessToken : String, idUser : String) : String
+members(accessToken : String, idGroups : String []) : String
+children(accessToken : String, edmodoParentId : String) : String
+registerBadge(title : String, description : String, imageUrl : String) : String
+updateBadge(idEdmodoBadge : String, title : String, description : String, imageUrl : String) : String
+badgesAwarded(accessToken : String, edmodold : String) : String
+awardBadge(accessToken : String, idEdmodoBadge : String, idEdmodoUser : String) : String
+revokeBadge(accessToken : String, idEdmodoBadge : String, idEdmodoUser : String) : String
+assignmentsByAppForUser(accesToken : String, idEdmodoUser : String) : String
+assignmentStatus(accesToken : String, assignmentId : String, userEdmodold : String) : String
+gradesSetByAppForGroup(accesToken : String, idEdmodoGroup : String) : String
+createAssignment(accessToken : String, idEdmodoUser : String, title : String, description : String, dueDate : Date, totalGrade : int, edmodoGroup : String) : String
+newGrade(accessToken : String, title : String, totalGrade : int, edmodoGroup : String) : String
+turnInAssignment(accesToken : String, assignmentId : String, studentEdmodold : String, content : String) : String
+setGrade(accessToken : String, studentEdmodold : String, score : String, total : double, gradeAssignmentEdmodold : String, type : byte, content : String) : String
+users(accessToken : String, users : String []) : String
+getClassControlUserRole(edmodoUserType : String) : byte

```

EdmodoUtil

```

+getGroupNamesAndEdmodId(idEdmodoUser : String, request : HttpServletRequest, edmodoData : boolean) : ArrayList<Entry<String, String>>
+getGroupMembers(idEdmodoGroup : String, request : HttpServletRequest, edmodoData : boolean) : ArrayList<Entry<String, String>>
+getGroupStudents(idEdmodoGroup : String, request : HttpServletRequest, edmodoData : boolean) : ArrayList<Student>
+getStudent(idEdmodoStudent : String, request : HttpServletRequest) : Student

```

ILUSTRACIÓN 28: CLASES ÚTILES DE ACCESO A LA API DE EDMODO

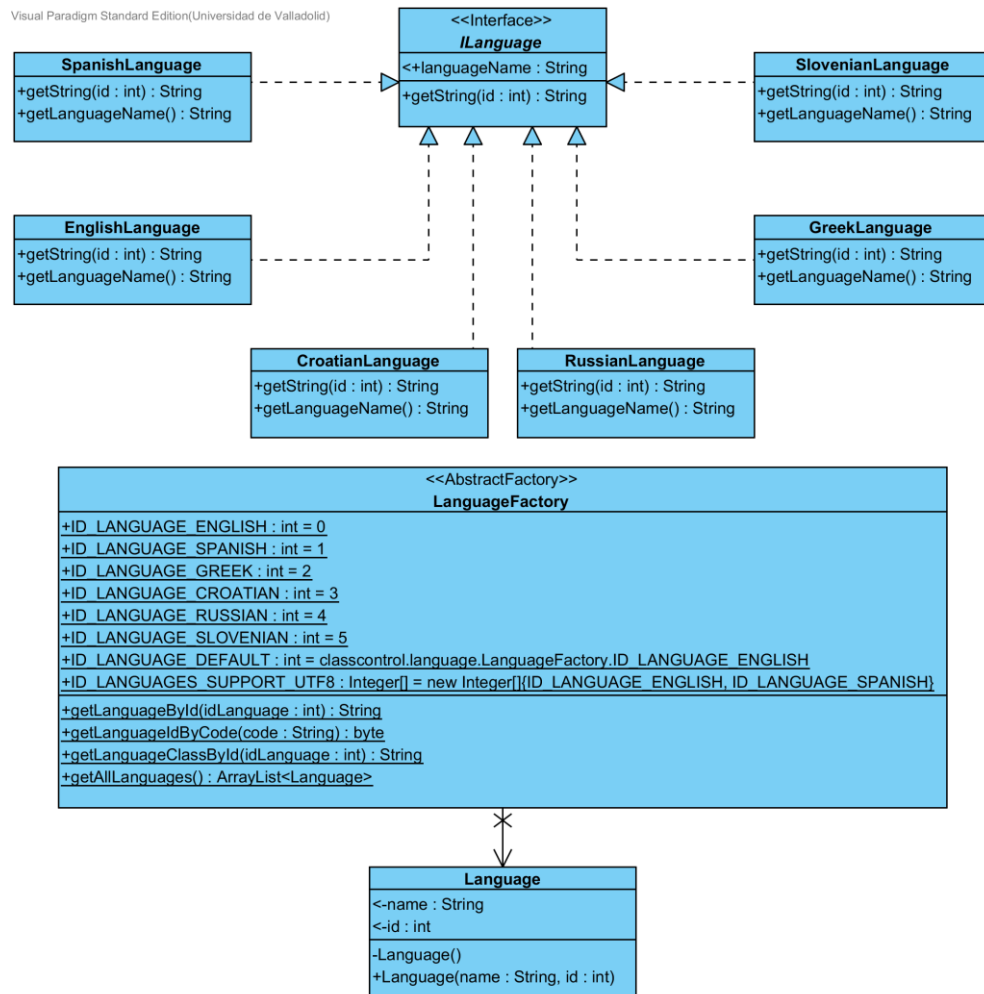


ILUSTRACIÓN 29: CLASES PARA LA INTERNACIONALIZACIÓN

<i>Class Control</i>	Versión: 1.3
Documento de Diseño	30/06/2014

1.3.2. Patrones de diseño utilizados

En cuanto a los patrones de diseño implementados, son los siguientes:

Patrón Front Controller

El patrón de diseño Front Controller^[6] sirve para centralizar todas las peticiones de los Usuarios a las vistas de la aplicación (archivos .jsp) en un único punto, el que se encargará de delegar las responsabilidades correspondientes.

La ILUSTRACIÓN 30 muestra la secuencia de peticiones en el patrón Front Controller.

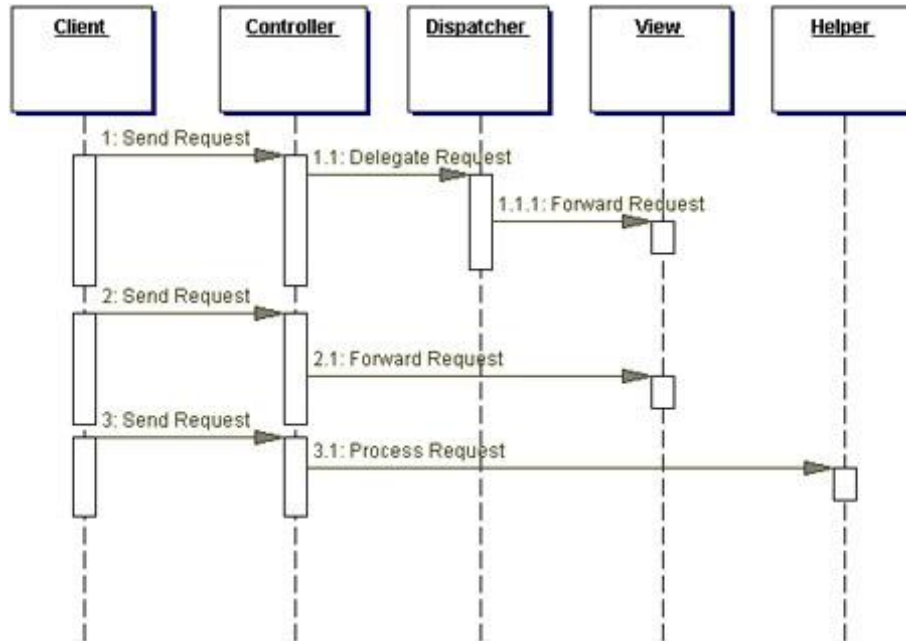


ILUSTRACIÓN 30: DIAGRAMA DE SECUENCIA DEL PATRÓN FRONT CONTROLLER TOMADO DE [6]

En la aplicación *Class Control*, las vistas llaman a un único *servlet* que hace de Front Controller, y éste se encarga de procesar todas las peticiones, delegando la petición a los controladores oportunos (controladores GRASP^[7]).

Patrón Façade

El patrón de diseño Façade^[8] sirve para tener un único punto de comunicación entre dos capas, y así evitar dependencias uno a uno entre clientes y servidores, siendo la fachada como un intermediario.

En la aplicación *Class Control*, se puede apreciar este patrón en las interfaces *IPersistence.java* (ver ILUSTRACIÓN 24: MODELADO DE LA PERSISTENCIA), *IController.java* (ver ILUSTRACIÓN 22: CLASES CONTROLLER) e *ILanguage.java* (ver ILUSTRACIÓN 29: CLASES PARA LA INTERNACIONALIZACIÓN).

La ILUSTRACIÓN 31 muestra un esquema de aplicación del patrón Façade en una aplicación web J2EE.

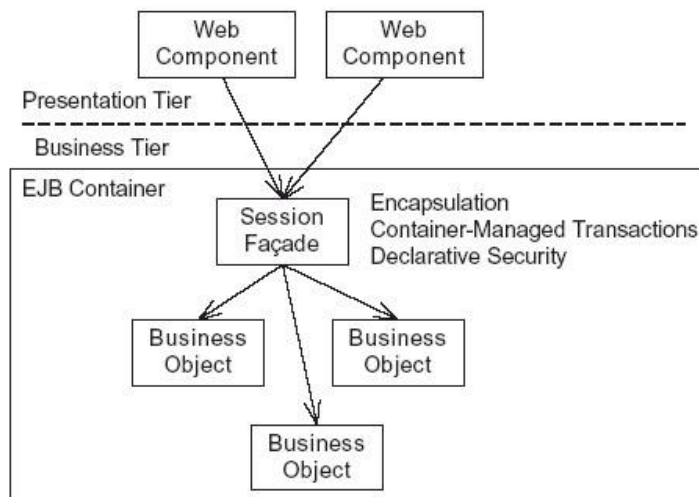


ILUSTRACIÓN 31: DIAGRAMA EXPLICATIVO DEL PATRÓN FAÇADE TOMADO DE [8]

Patrón Abstract Factory

El patrón de diseño Abstract Factory^[9] sirve para evitar acoplamientos, y se encarga de proporcionar a un cliente su proveedor correspondiente sin que el cliente sepa quién es, sólo necesita un identificador único para encargarse de proporcionárselo.

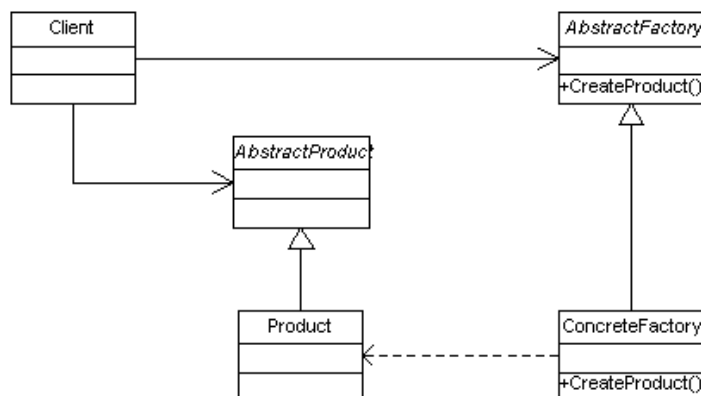


ILUSTRACIÓN 32: DIAGRAMA DE CLASES DEL PATRÓN ABSTRACT FACTORY TOMADO DE [9]

En la aplicación *Class Control*, se puede apreciar este patrón en *ControllerFactory.java* (ver ILUSTRACIÓN 22: CLASES CONTROLLER) y en *LanguageFactory.java* (ver ILUSTRACIÓN 29: CLASES PARA LA INTERNACIONALIZACIÓN).

Patrones DAO y Singleton

El patrón de diseño DAO^[10] sirve para acceder a la base de datos mediante un Connection Pool (*pool* de conexiones), un único punto de acceso, que abre una conexión de la base de datos y realiza peticiones a la misma, para que esté todo más controlado y sea más eficiente que ir abriendo conexiones cada vez que se quiera hacer alguna consulta o actualización.

En Java el patrón DAO se implementa con el patrón Singleton^[Error! No se encuentra el origen de la referencia], que permite tener una sola instancia en ejecución del ya mencionado *pool* de conexiones.

Gracias a este patrón se separa la capa de controladores con la de acceso a datos.

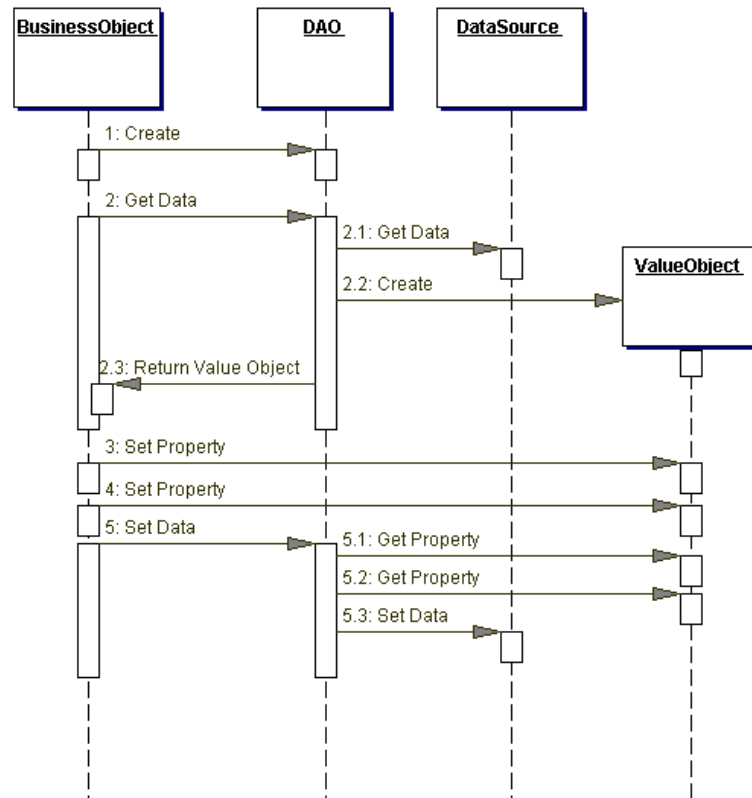


ILUSTRACIÓN 33: DIAGRAMA DE SECUENCIA DEL PATRÓN DAO TOMADO DE [10]

En la aplicación *Class Control*, se puede visualizar el uso de este patrón en la **ILUSTRACIÓN 25: DAOS**.

1.3.3. Vista lógica

1.3.2.1. Paquetes de la aplicación *Class Control*

El siguiente diagrama muestra la arquitectura lógica de la aplicación. Nótese que los paquetes WebUI y Utilities son ficticios, ya que su funcionalidad es indicar que los paquetes reales que poseen tienen unas determinadas dependencias.

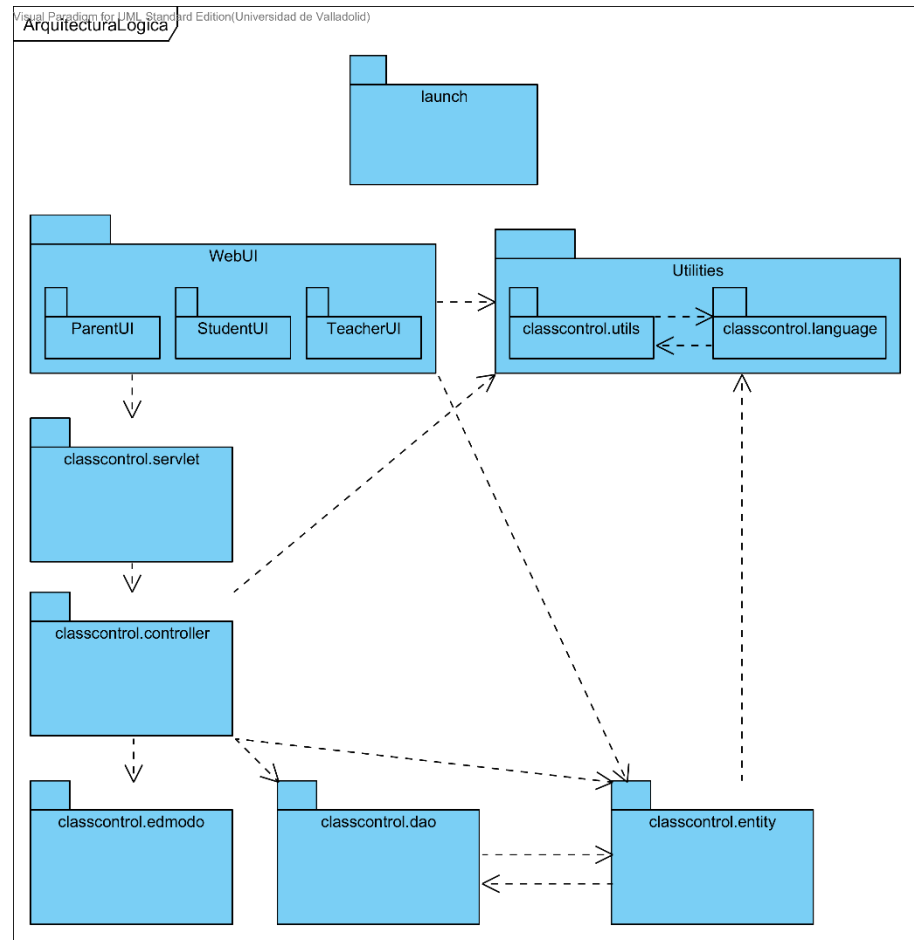


ILUSTRACIÓN 34: DIAGRAMA DE PAQUETES DEL SISTEMA

3. Modelo de datos

3.1. Diagrama relacional de la base de datos

El siguiente diagrama corresponde al diseño de la base de datos de la aplicación *Class Control*.

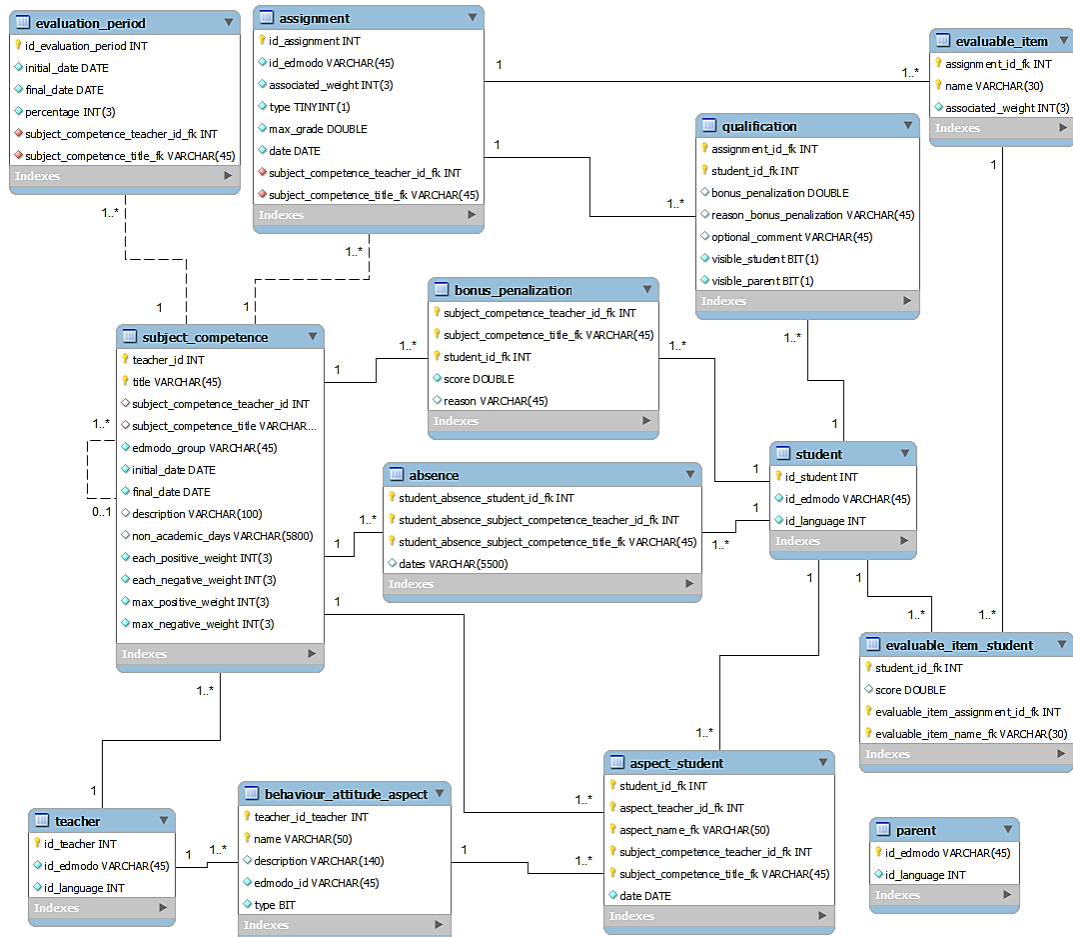


ILUSTRACIÓN 35: DIAGRAMA RELACIONAL DE LA BASE DE DATOS

Class Control	Versión: 1.3
Documento de Diseño	30/06/2014

4. Diseño de interfaces de usuario

4.1. Bocetos de las interfaces de usuario

Los siguientes bocetos están realizados con la herramienta *Balsamic Mockups 2.2.19* [1].

Representan la interfaz web de la aplicación. Cabe destacar que los elementos de cada una de las pantallas (o *.jsp*) están ordenados de mayor a menor importancia para el Usuario. Se ha buscado una gran adaptabilidad a cualquier plataforma y sencillez en el bocetaje o *sketching*.

4.1.1 Profesor

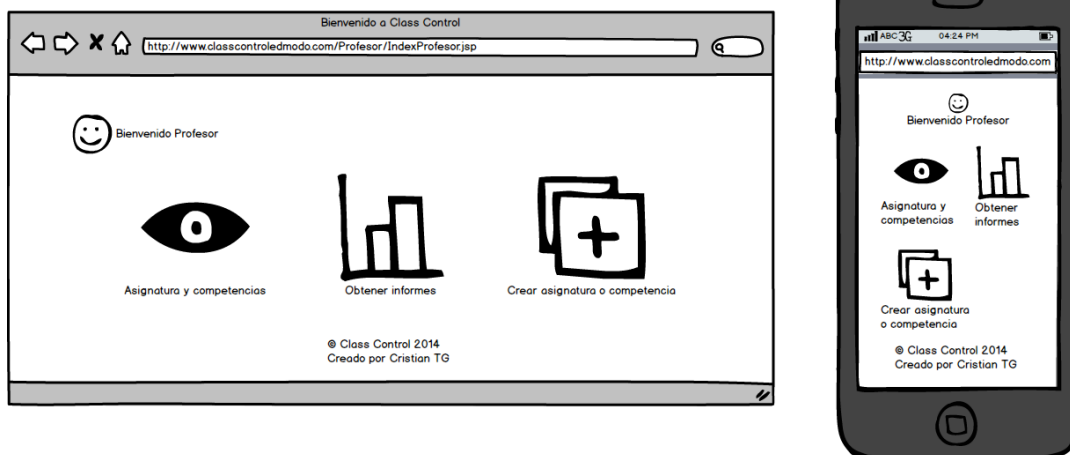


ILUSTRACIÓN 36: BOCETO DE PROFESOR: PÁGINA PRINCIPAL

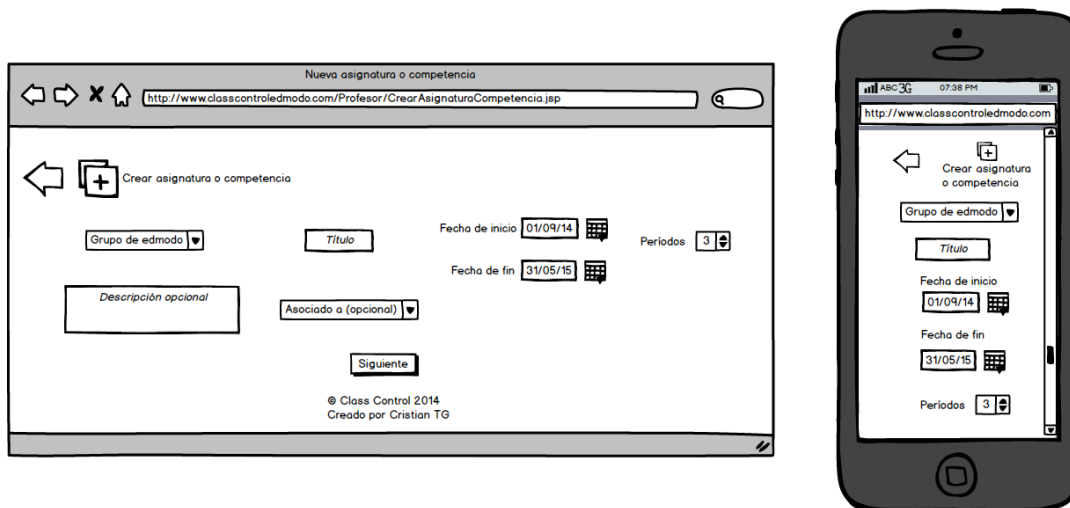


ILUSTRACIÓN 37: BOCETO DE PROFESOR: CREAR ASIGNATURA O COMPETENCIA

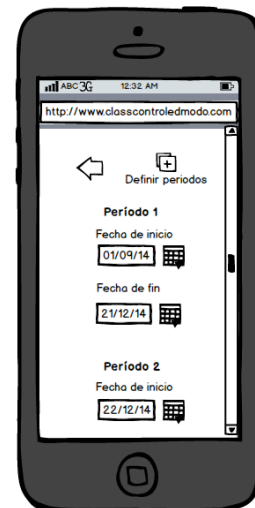
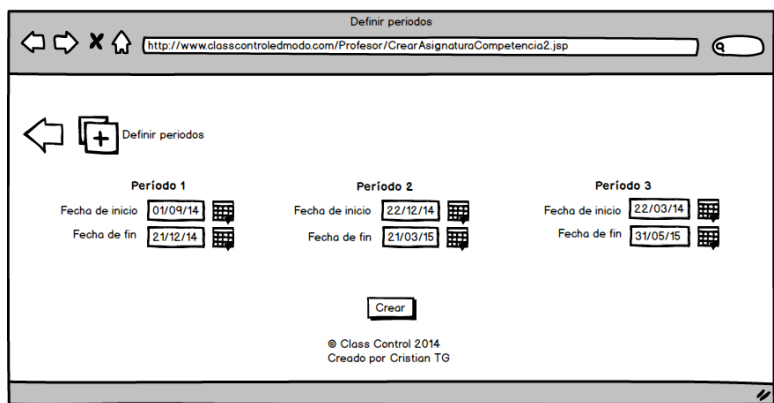


ILUSTRACIÓN 38: BOCETO DE PROFESOR: CREAR ASIGNATURA O COMPETENCIA 2

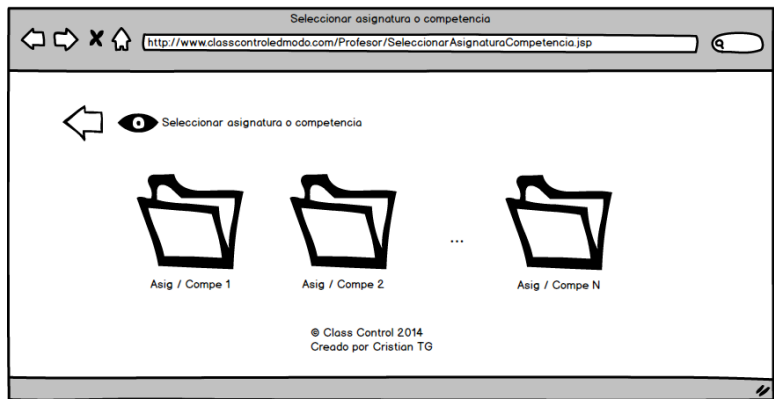


ILUSTRACIÓN 39: BOCETO DE PROFESOR: SELECCIONAR ASIGNATURA O COMPETENCIA

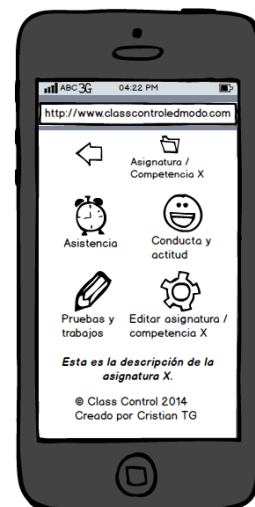
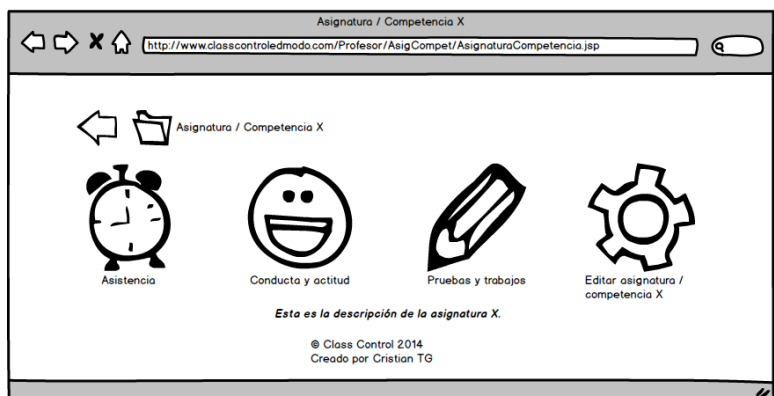


ILUSTRACIÓN 40: BOCETO DE PROFESOR: ASIGNATURA O COMPETENCIA

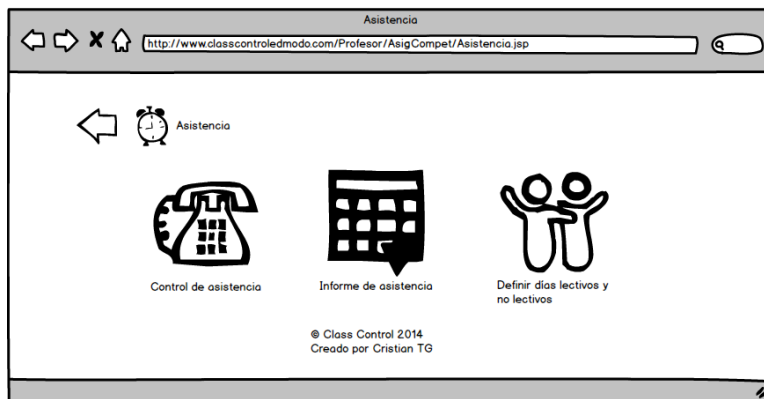


ILUSTRACIÓN 41: BOCETO DE PROFESOR: ASISTENCIA

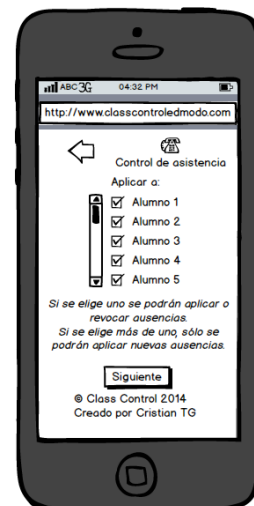
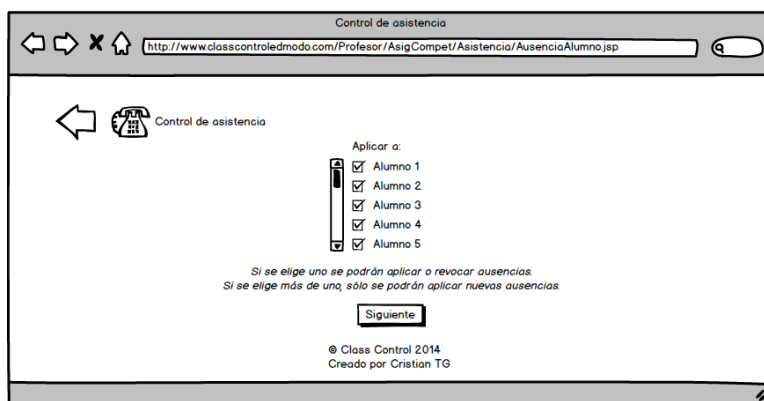


ILUSTRACIÓN 42: BOCETO DE PROFESOR: INDICAR AUSENCIAS

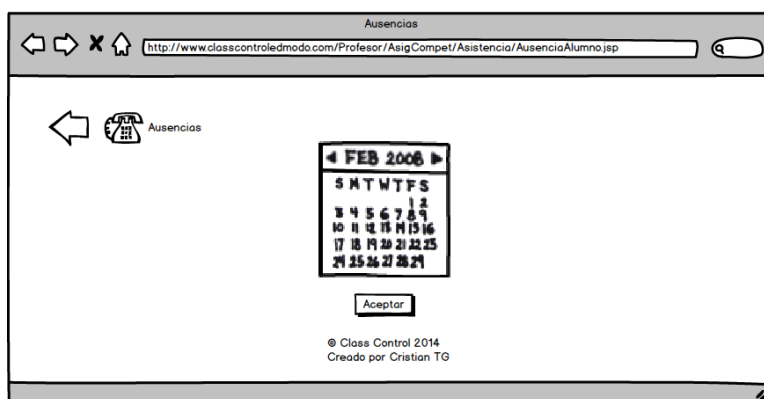


ILUSTRACIÓN 43: BOCETO DE PROFESOR: INDICAR AUSENCIAS 2

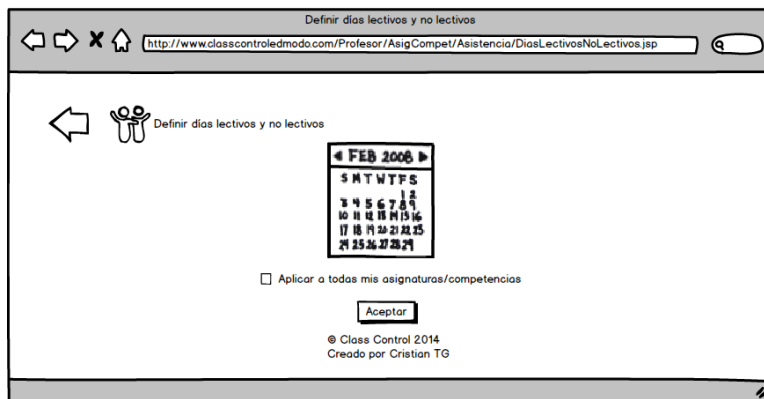


ILUSTRACIÓN 44: BOCETO DE PROFESOR: MARCAR DÍAS NO LECTIVOS

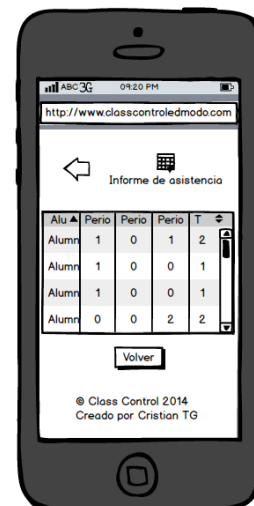
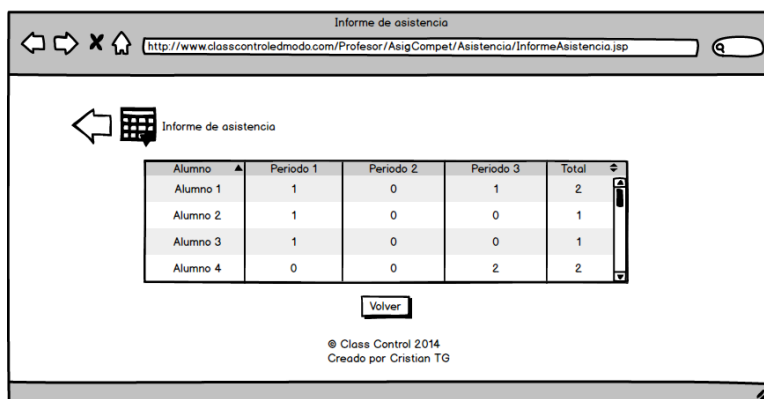


ILUSTRACIÓN 45: BOCETO DE PROFESOR: VISUALIZAR INFORME ASISTENCIA

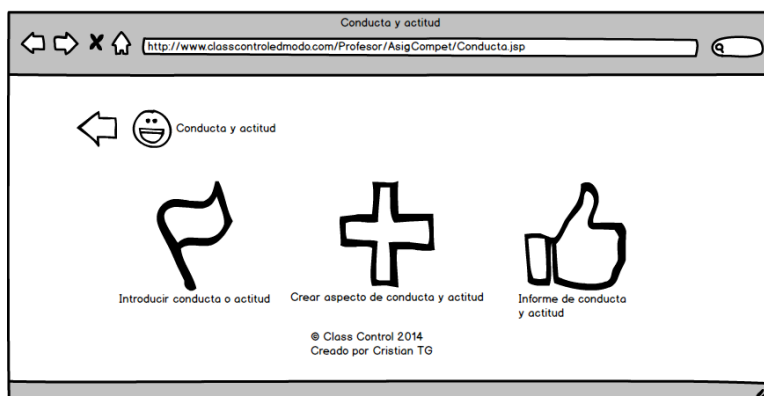


ILUSTRACIÓN 46: BOCETO DE PROFESOR: CONDUCTA Y ACTITUD

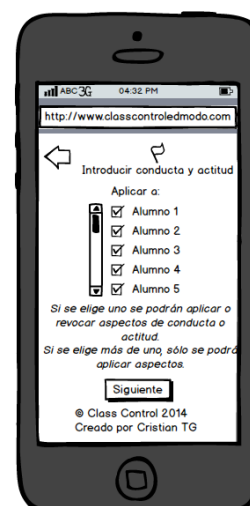
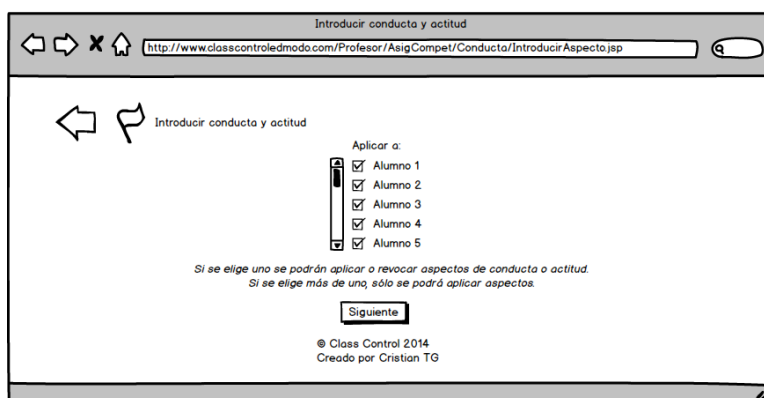


ILUSTRACIÓN 47: BOCETO DE PROFESOR: INDICAR ASPECTO 1

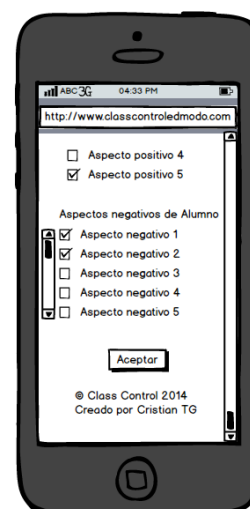
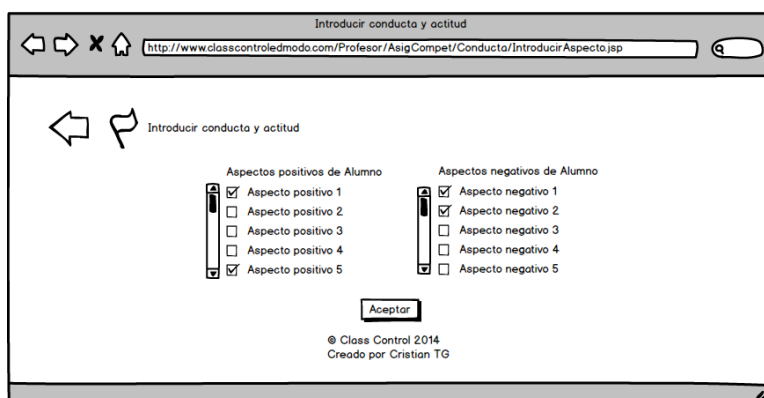


ILUSTRACIÓN 48: BOCETO DE PROFESOR: INDICAR ASPECTO 2

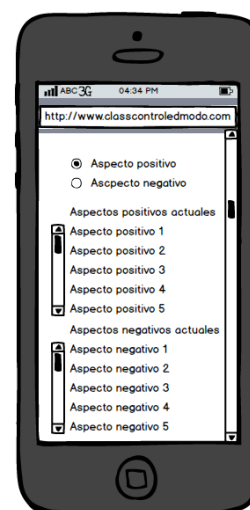
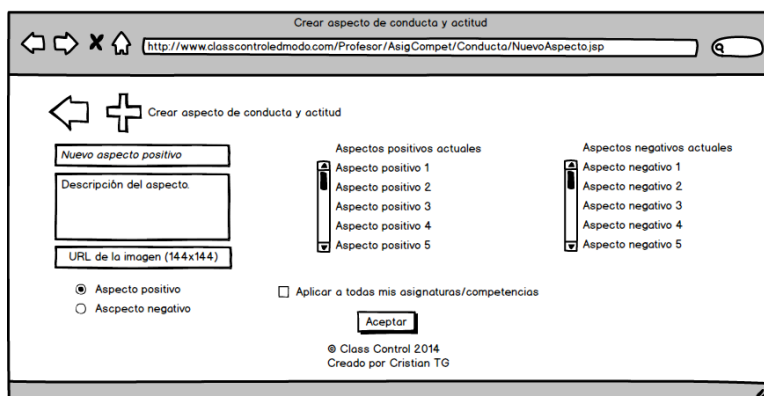


ILUSTRACIÓN 49: BOCETO DE PROFESOR: AÑADIR ASPECTOS DE CONDUCTA Y ACTITUD

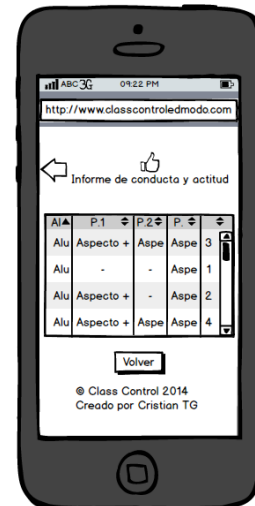
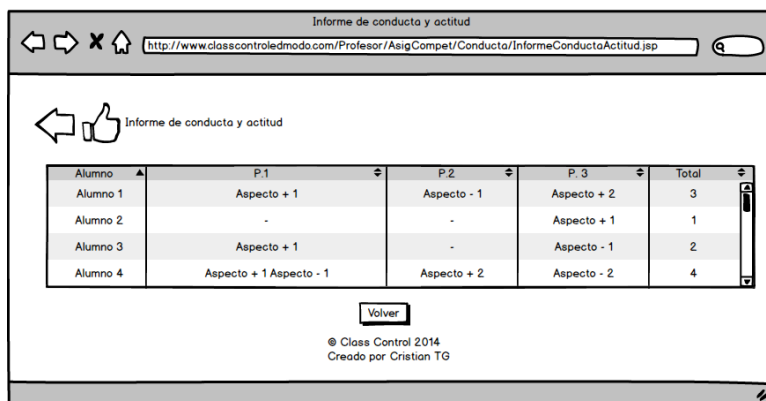


ILUSTRACIÓN 50: BOCETO DE PROFESOR: VISUALIZAR INFORME CONDUCTA Y ACTITUD

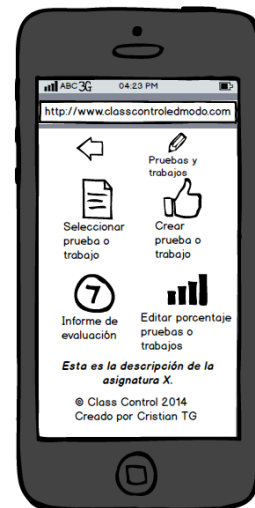
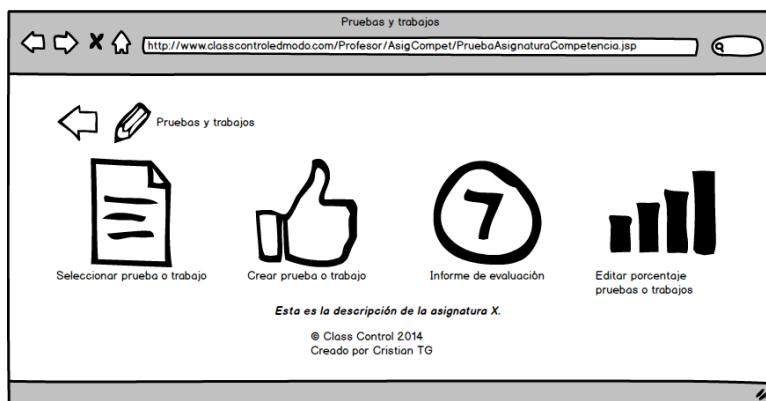


ILUSTRACIÓN 51: BOCETO DE PROFESOR: PRUEBAS Y TRABAJOS

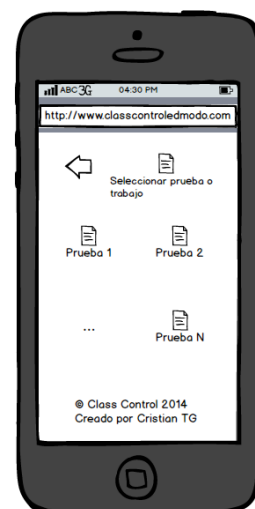
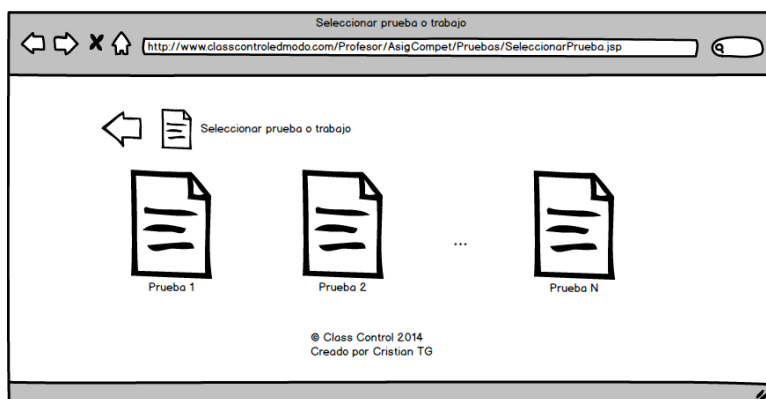


ILUSTRACIÓN 52: BOCETO DE PROFESOR: SELECCIONAR PRUEBA O TRABAJO

Calificar prueba

http://www.classcontroledmodo.com/Profesor/Pruebas/Prueba/CalificarPrueba.jsp

Calificar prueba

Calificación máxima 10

Alumno	Obtenido	Comentario	V	V	I. I	I. II	I. III	Bonif	Motivo	Final
Alumno 1	8	Mejora cada día	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	5	5	10	+0.4	Trabajador	8.4
Alumno 2	7	-	<input type="checkbox"/>	<input type="checkbox"/>	2	2	2	-0.2	No hace	6.3
Alumno 3	6	Es muy	<input checked="" type="checkbox"/>	<input type="checkbox"/>	10	4	5	-0.2	Habla mucho	5.78

Aceptar

© Class Control 2014
Creado por Cristian TG

Calificar prueba

Calificación máxima 10

Al	O	C	V	V	I. I	I. II	I. III	B	M	F
A 8	M	Mej	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	5	5	10	+0.4	Tr	8.4
A 7	-	-	<input type="checkbox"/>	<input type="checkbox"/>	2	2	2	-0.2	No	6.3
A 6	Es	Es	<input checked="" type="checkbox"/>	<input type="checkbox"/>	10	4	5	-0.2	Ha	5.78

Aceptar

© Class Control 2014
Creado por Cristian TG

ILUSTRACIÓN 53: BOCETO DE PROFESOR: CALIFICAR PRUEBA O TRABAJO

Crear prueba o trabajo

http://www.classcontroledmodo.com/Profesor/AsigCompet/Pruebas/CrearPrueba.jsp

Crear prueba o trabajo

Título

Fecha 12/11/14

Calificación máxima

Tipo

Descripción

Crear

© Class Control 2014
Creado por Cristian TG

Crear prueba o trabajo

Título

Fecha 01/09/14

Calificación máxima

Tipo

ILUSTRACIÓN 54: BOCETO DE PROFESOR: CREAR PRUEBA O TRABAJO

Informe de evaluación

http://www.classcontroledmodo.com/Profesor/AsigCompet/Pruebas/InformeEvaluacion.jsp

Informe de evaluación

Final

Alumno	Media	Bonif Penaliz	Motivo	Final
Alumno 1	8	+0.5	Trabajador	8.5
Alumno 2	7	+0.9	Muy aplicado	7.9
Alumno 3	6	+0.2	Se esfuerza	6.2

Periodo 1: 20/09/14 - 20/12/14

Alumno	Media
Alumno 1	8
Alumno 2	7

Informe de evaluación

Final

Al	B	M	F
Alu 8	+0.5	Trabajad	8
Alu 7	+0.9	Muy	7
Alu 6	+0.2	Se	6

Periodo 1: 20/09/14 - 20/12/14

Alumno	Obtenido
Alumno 1	8
Alumno 2	7

ILUSTRACIÓN 55: BOCETO DE PROFESOR: VISUALIZAR INFORME EVALUACIÓN

Editar porcentaje pruebas o trabajos

http://www.classcontroledmodo.com/Profesor/AsigCompet/Pruebas/PorcentajePruebas.jsp

Definir porcentajes evaluación

Periodo 1	Periodo 2	Periodo 3	Aspectos positivos	Aspectos negativos	Competencia secundaria
Prueba 1 90 %	Prueba 2 90 %	Prueba 3 90 %	Cada uno 1 % Máximo 5 %	Cada uno 1 % Máximo 5 %	5 %

Aceptar

http://www.classcontroledmodo.com

Editar porcentaje pruebas o trabajos

Prueba 1	Prueba 2	Prueba 3
20 %	30 %	35 %

ILUSTRACIÓN 56: BOCETO DE PROFESOR: INDICAR PESOS EN LA EVALUACIÓN TOTAL

Editar asignatura / competencia X

http://www.classcontroledmodo.com/Profesor/AsigCompet/Editar.jsp

Modificar asignatura / competencia X Eliminar asignatura / competencia X

© Class Control 2014
Creado por Cristian TG

http://www.classcontroledmodo.com

Editar asignatura / competencia X

Modificar asignatura / competencia X Eliminar asignatura / competencia X

© Class Control 2014
Creado por Cristian TG

ILUSTRACIÓN 57: BOCETO DE PROFESOR: EDITAR ASIGNATURA O COMPETENCIA

Modificar asignatura / competencia X

http://www.classcontroledmodo.com/Profesor/AsigCompet/Editar/ModificarAsignaturaCompetencia.jsp

Título

Fecha de inicio 01/09/14 Fecha de fin 31/05/15 Periodos 3

Descripción opcional

Asociado a (opcional)

Siguiente

© Class Control 2014
Creado por Cristian TG

http://www.classcontroledmodo.com

Modificar asignatura / competencia X

Título

Fecha de inicio 01/09/14 Fecha de fin 31/05/15 Periodos 3

ILUSTRACIÓN 58: BOCETO DE PROFESOR: MODIFICAR ASIGNATURA O COMPETENCIA 1

Modificar asignatura o competencia

← → × ↵ <http://www.classcontroledmodo.com/Profesor/AsigCompet/Editar/ModificarAsignaturaCompetencia2.jsp>

← | Modificar asignatura o competencia

Período 1	Período 2	Período 3
Fecha de inicio: 01/09/14	Fecha de inicio: 22/12/14	Fecha de inicio: 22/03/14
Fecha de fin: 21/12/14	Fecha de fin: 21/03/15	Fecha de fin: 31/05/15

Modificar

© Class Control 2014
Creado por Cristian TG

← | Modificar asignatura o competencia

Período 1

Fecha de inicio: 01/09/14

Fecha de fin: 21/12/14

Período 2

Fecha de inicio: 22/12/14

ILUSTRACIÓN 59: BOCETO DE PROFESOR: MODIFICAR ASIGNATURA O COMPETENCIA 2

Eliminar asignatura / competencia X

← ⚙ <http://www.classcontroledmodo.com/Profesor/AsigCompet/Editar/EliminarAsignaturaCompetencia.jsp>

← ⚙ Eliminar asignatura / competencia X

¡Atención!

¿Está seguro que desea eliminar esta asignatura/competencia? No se podrá deshacer la operación.

OK No

© Class Control 2014
Creado por Cristian TG

← ⚙ Eliminar asignatura / competencia X

¡Atención!

¿Está seguro que desea eliminar esta asignatura/competencia? No se podrá deshacer la operación.

OK No

© Class Control 2014
Creado por Cristian TG

ILUSTRACIÓN 60: BOCETO DE PROFESOR: ELIMINAR ASIGNATURA O COMPETENCIA

Obtener informes

← → × ↵ <http://www.classcontroledmodo.com/Profesor/obtenerInformes.jsp>

← | Seleccionar opciones

Asignaturas y competencias	Controles	Fecha inicio obligatoria
<input checked="" type="checkbox"/> Asignatura 1	<input checked="" type="checkbox"/> Evaluación	12/11/14
<input checked="" type="checkbox"/> Competencia 1	<input checked="" type="checkbox"/> Asistencia	Fecha fin opcional
<input checked="" type="checkbox"/> Asignatura 2	<input checked="" type="checkbox"/> Conducta y actitud	12/03/15
<input type="checkbox"/> Competencia 2		

Siguiente

© Class Control 2014
Creado por Cristian TG

Fecha inicio obligatoria: 12/11/14

Fecha fin opcional: 12/03/15

Siguiente

© Class Control 2014
Creado por Cristian TG

ILUSTRACIÓN 61: BOCETO DE PROFESOR: OBTENER INFORMES 1

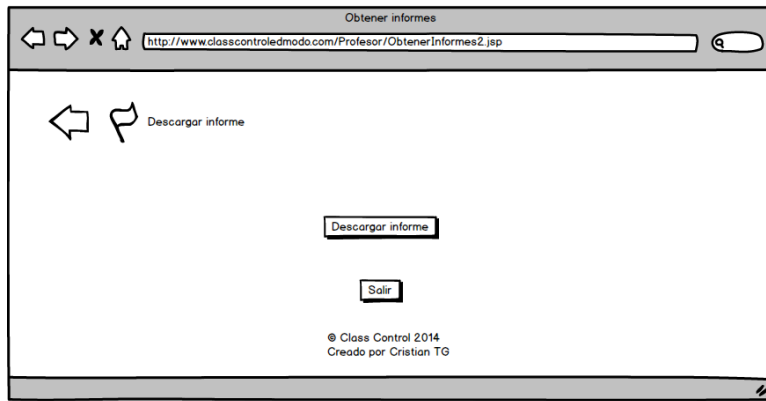


ILUSTRACIÓN 62: BOCETO DE PROFESOR: OBTENER INFORMES 2

4.1.2 Alumno

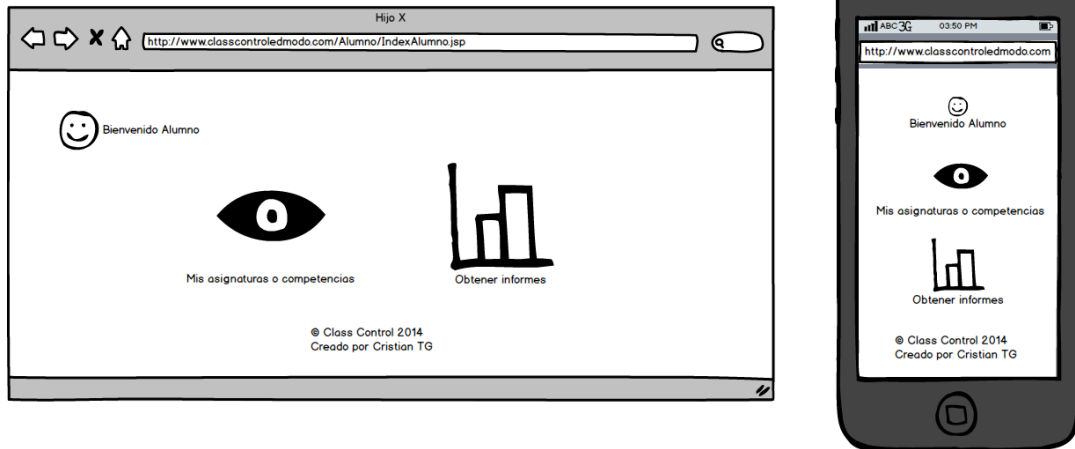


ILUSTRACIÓN 63: BOCETO DE ALUMNO: PÁGINA PRINCIPAL

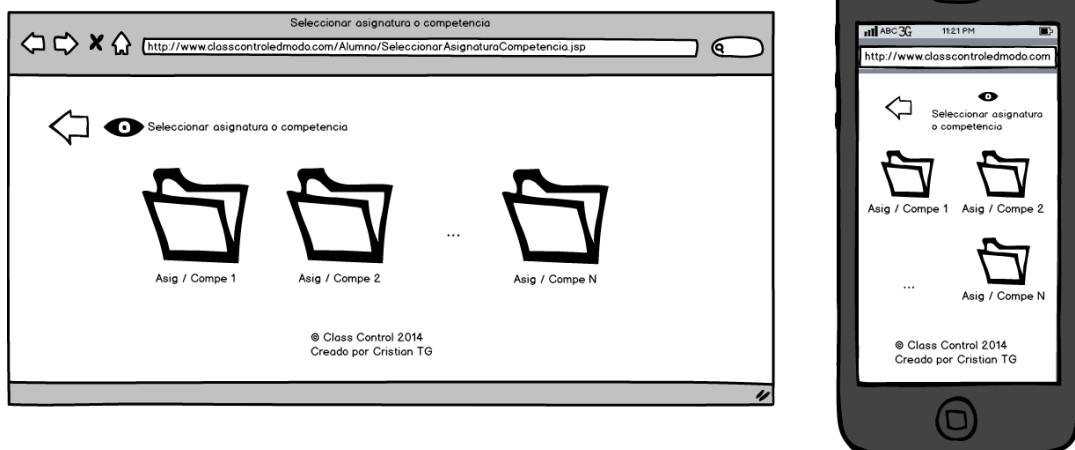


ILUSTRACIÓN 64: BOCETO DE ALUMNO: SELECCIONAR ASIGNATURA O COMPETENCIA

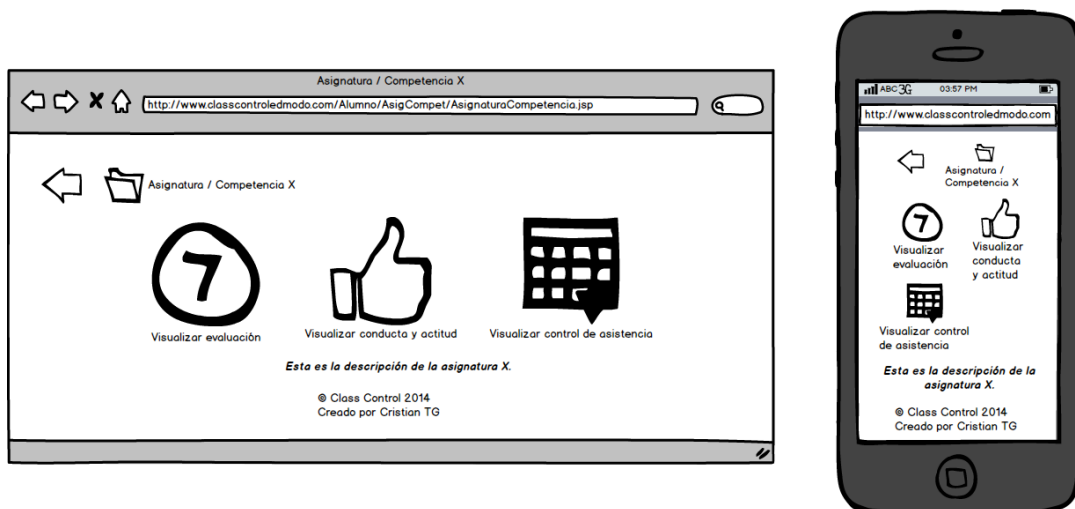


ILUSTRACIÓN 65: BOCETO DE ALUMNO: ASIGNATURA O COMPETENCIA

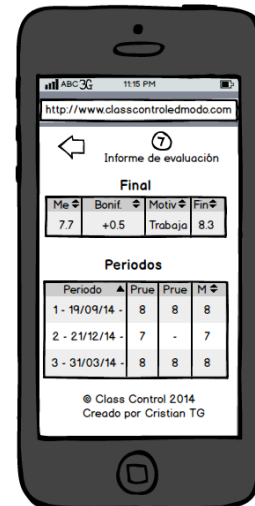
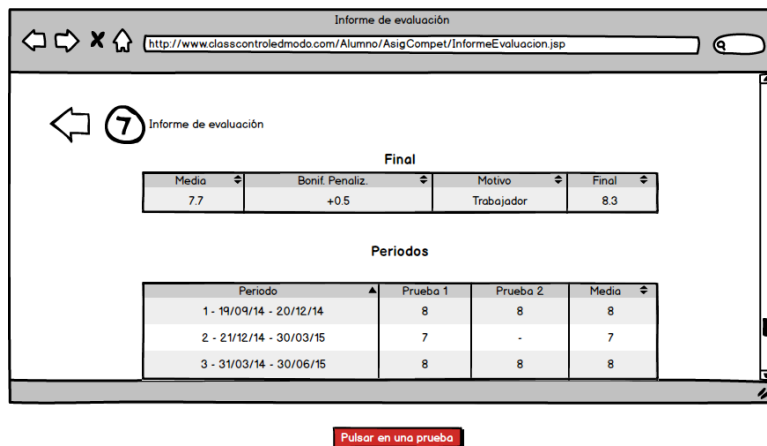


ILUSTRACIÓN 66: BOCETO DE ALUMNO: VISUALIZAR EVALUACIÓN

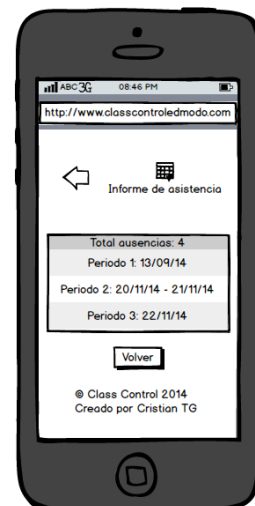
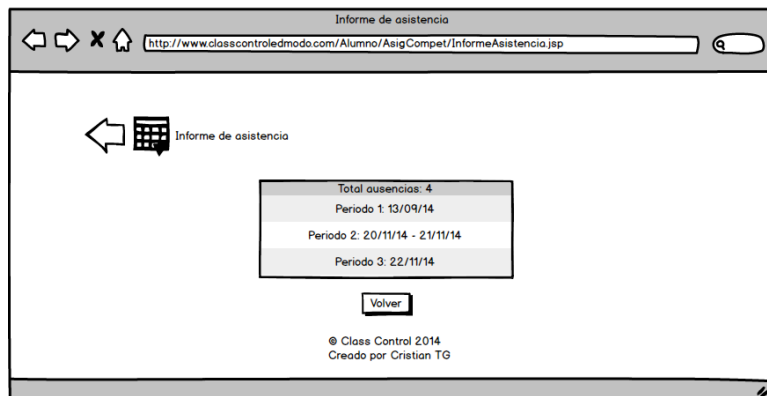


ILUSTRACIÓN 67: BOCETO DE ALUMNO: VISUALIZAR ASISTENCIA

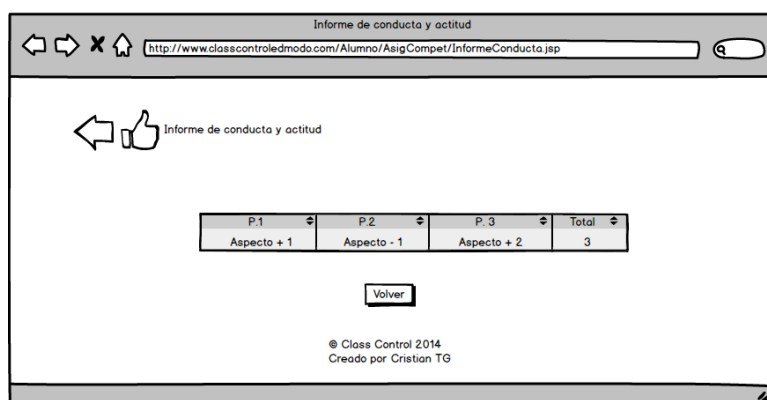


ILUSTRACIÓN 68: BOCETO DE ALUMNO: VISUALIZAR CONDUCTA O ACTITUD

Calificación prueba

← → X ↵ <http://www.classcontrolmodo.com/Padre/Alumno/calificacionPrueba.jsp>

← ✓ Visualizar prueba

Calificación máxima 10
% máximo de los items evaluables 10

Alumno	Obtenido 90%	Comentario	I 1 2%	II 2% ^v	III 6%	Bonif Penal.	Motivo	Fina
Alumno 1	8	Mejora cada día	5	5	10	+0.4	Trabajador	8.4

Aceptar

© Class Control 2014
Creado por Cristian TG

← ✓ Visualizar prueba

Calificación máxima 10
% máximo de los items evaluables 10

O	C	I	B	F
A	8	Mej	5	1 +0

Aceptar

ILUSTRACIÓN 69: BOCETO DE ALUMNO: VISUALIZAR CALIFICACIÓN PRUEBA

Obtener informes

← → X ↵ <http://www.classcontrolmodo.com/Alumno/ObtenerInformes.jsp>

← ↵ Seleccionar opciones

Asignaturas y competencias

- ☒ Asignatura 1
- ☒ Competencia 1
- ☒ Asignatura 2
- ☐ Competencia 2

Controles

- ☒ Evaluación
- ☒ Asistencia
- ☒ Conducta y actitud

Fecha inicio obligatoria 12/11/14

Fecha fin opcional 12/03/15

Siguiente

© Class Control 2014
Creado por Cristian TG

Fecha inicio obligatoria 12/11/14

Fecha fin opcional 12/03/15

Siguiente

© Class Control 2014
Creado por Cristian TG

ILUSTRACIÓN 70: BOCETO DE ALUMNO: OBTENER INFORMES 1

Obtener informes

← → X ↵ <http://www.classcontrolmodo.com/Alumno/ObtenerInformes2.jsp>

← ↵ Descargar informe

Descargar informe

Salir

© Class Control 2014
Creado por Cristian TG

← ↵ Descargar informe

Descargar informe

Salir

© Class Control 2014
Creado por Cristian TG

ILUSTRACIÓN 71: BOCETO DE ALUMNO: OBTENER INFORMES 2

4.1.3 Padre

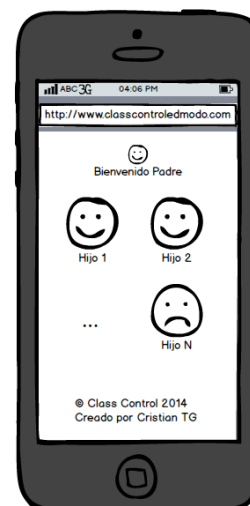
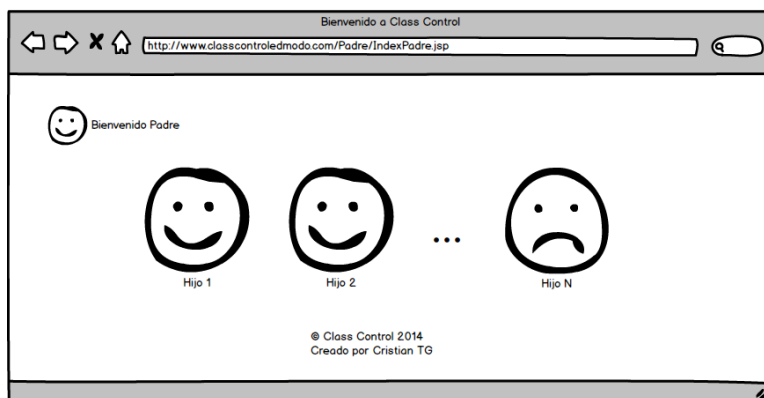


ILUSTRACIÓN 72: BOCETO DE PADRE: PÁGINA PRINCIPAL

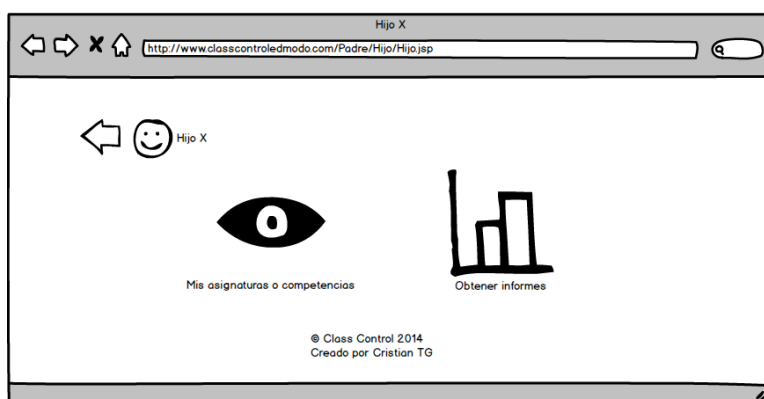


ILUSTRACIÓN 73: BOCETO DE PADRE: PÁGINA PRINCIPAL DEL HIJO

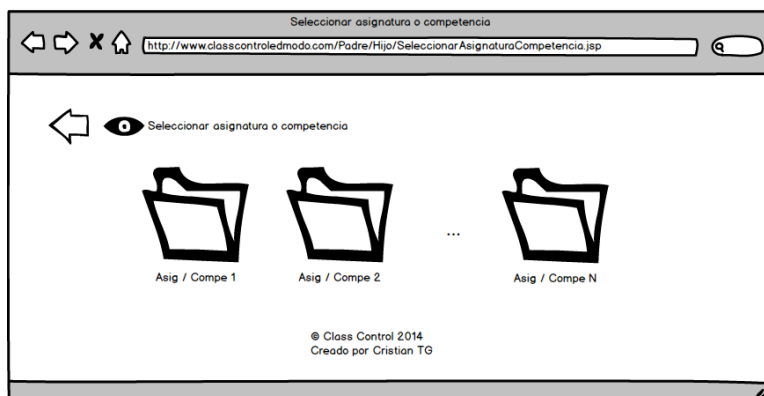


ILUSTRACIÓN 74: BOCETO DE PADRE: SELECCIONAR ASIGNATURA O COMPETENCIA

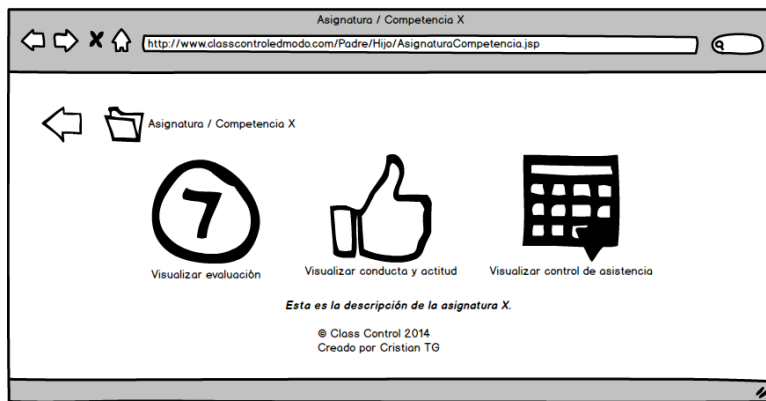


ILUSTRACIÓN 75: BOCETO DE PADRE: ASIGNATURA O COMPETENCIA

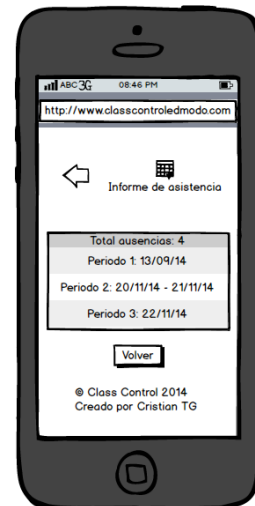
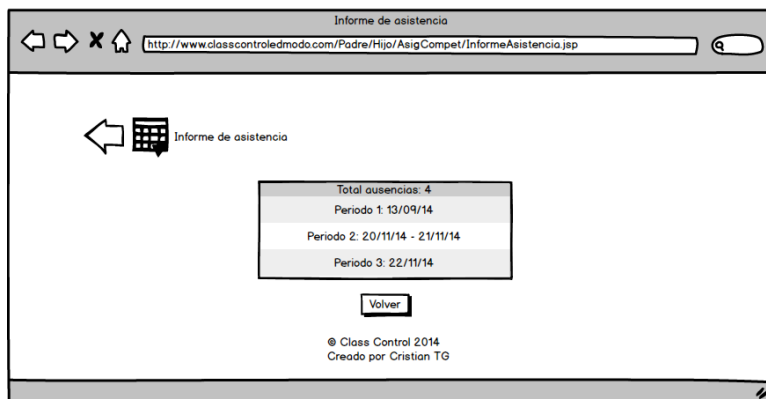


ILUSTRACIÓN 76: BOCETO DE PADRE: VISUALIZAR ASISTENCIA

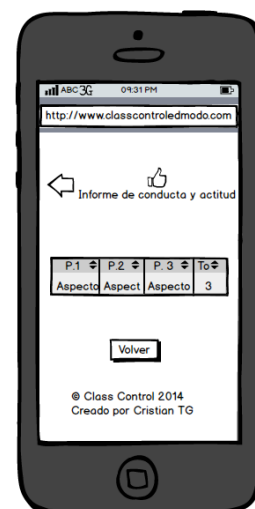
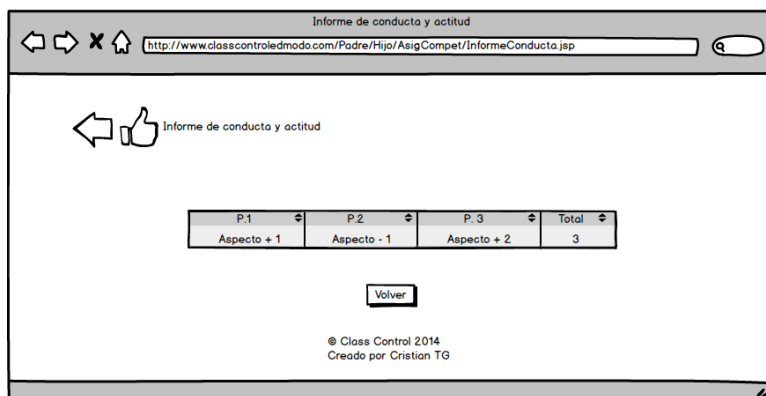


ILUSTRACIÓN 77: BOCETO DE PADRE: VISUALIZAR CONDUCTA O ACTITUD

Informe de evaluación

http://www.classcontrolmoda.com/Padre/Hijo/AsigCompet/InformeEvaluacion.jsp

Informe de evaluación

Final			
Media	Bonif. Penaliz.	Motivo	Final
7.7	+0.5	Trabajador	8.3

Periodos			
Periodo	Prueba 1	Prueba 2	Media
1 - 19/09/14 - 20/12/14	8	8	8
2 - 21/12/14 - 30/03/15	7	-	7
3 - 31/03/14 - 30/06/15	8	8	8

Pulsar en una prueba

Informe de evaluación

http://www.classcontrolmoda.com

Informe de evaluación

Final			
Me	Bonif	Motiv	F
7.7	+0.5	Trabaja	8

Periodos			
Periodo	Prue	Prue	M
1 - 19/09/14 -	8	8	8
2 - 21/12/14 -	7	7	7
3 - 31/03/14 -	8	8	8

© Class Control 2014
Creado por Cristian TG

ILUSTRACIÓN 78: BOCETO DE PADRE: VISUALIZAR EVALUACIÓN

Calificación prueba

http://www.classcontrolmoda.com/Padre/Hijo/Prueba/CalificacionPrueba.jsp

Visualizar prueba

Calificación máxima 10

% máximo de los ítems evaluables 10

Obtenido 90%	Comentario	I. I 2%	I. II 2% v	I. III 6%	Bonif. Penal.	Motivo	Final
8	Mejora cada día	5	5	10	+0.4	Trabajador	8.4

Aceptar

© Class Control 2014
Creado por Cristian TG

Visualizar prueba

http://www.classcontrolmoda.com

Visualizar prueba

Calificación máxima 10

% máximo de los ítems evaluables 10

O	C	I	B	V	P	M	F
8	Mej	5	5	10	+0.4	Tr	8.4

Aceptar

© Class Control 2014
Creado por Cristian TG

ILUSTRACIÓN 79: BOCETO DE PADRE: VISUALIZAR CALIFICACIÓN PRUEBA

Obtener informes

http://www.classcontrolmoda.com/Padre/ObtenerInformes.jsp

Seleccionar opciones

Asignaturas y competencias

- ☒ Asignatura 1
- ☒ Competencia 1
- ☒ Asignatura 2
- ☐ Competencia 2

Controles

- ☒ Evaluación
- ☒ Asistencia
- ☒ Conducta y actitud

Fecha inicio obligatoria 12/11/14

Fecha fin opcional 12/03/15

Siguiente

© Class Control 2014
Creado por Cristian TG

Fecha inicio obligatoria 12/11/14

Fecha fin opcional 12/03/15

Siguiente

© Class Control 2014
Creado por Cristian TG

ILUSTRACIÓN 80: BOCETO DE PADRE: OBTENER INFORMES 1

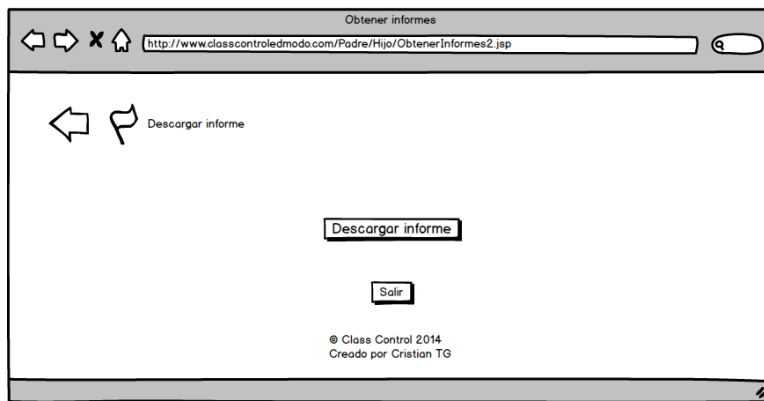


ILUSTRACIÓN 81: BOCETO DE PADRE: OBTENER INFORMES 2

4.1.4 Páginas de error

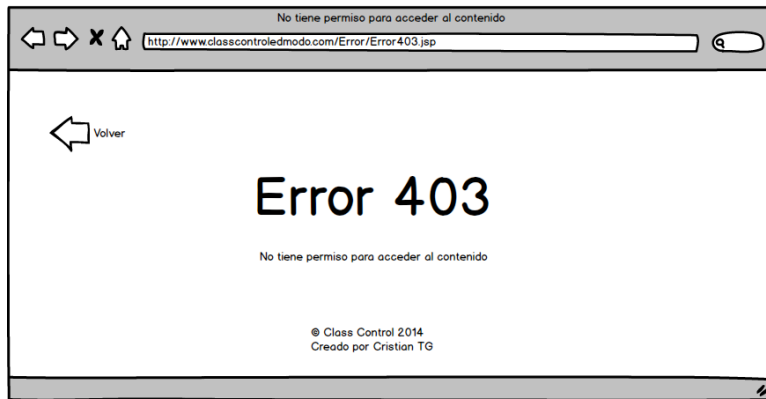


ILUSTRACIÓN 82: BOCETO DE PÁGINA DE ERROR 403



ILUSTRACIÓN 83: BOCETO DE PÁGINA DE ERROR 404

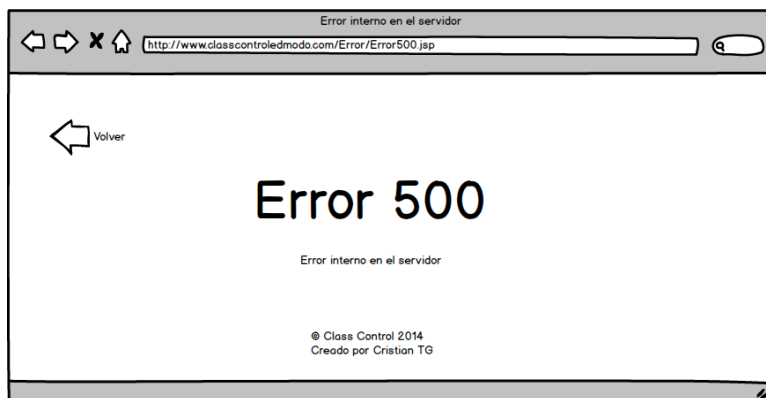


ILUSTRACIÓN 84: BOCETO DE PÁGINA DE ERROR 500

4.2. Mapa web del sistema final

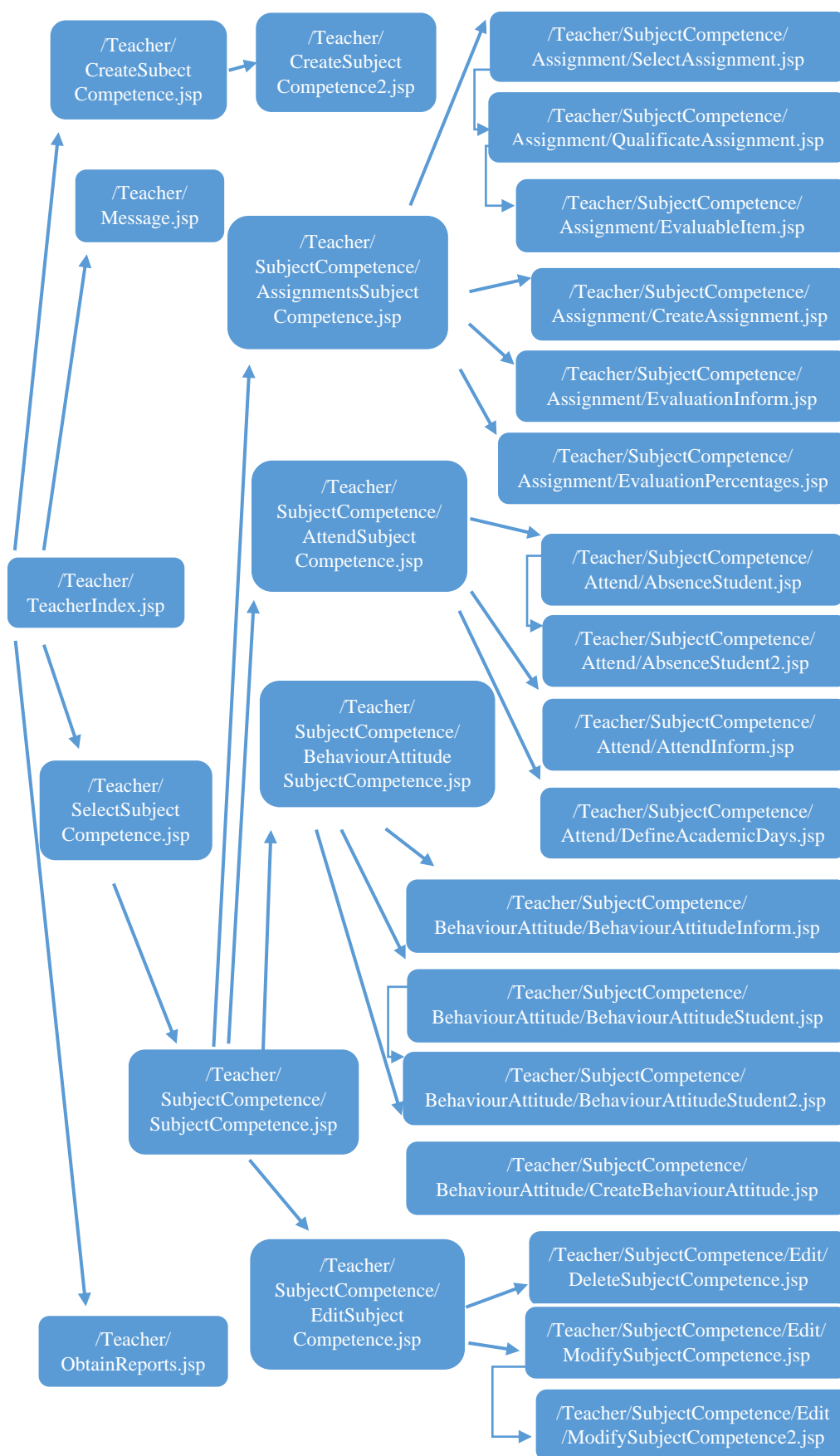


ILUSTRACIÓN 85: MAPA WEB DEL SISTEMA – PARTE 1



ILUSTRACIÓN 86: MAPA WEB DEL SISTEMA – PARTE 2