

UNIVERSIDAD REY JUAN CARLOS

INGENIERÍA INFORMÁTICA

Curso Académico 2004/2005

Proyecto de Fin de Carrera

**EDITORES COLABORATIVOS
CONFIGURABLES**

Autor: Néstor Carrasco Montero

Tutora: Estefanía Martín Barroso

AGRADECIMIENTOS

Dedico este proyecto a mis padres y a mi hermana por su cariño, comprensión, ayuda y apoyo en todo y desde siempre.

Quisiera agradecer a mi tutora Estefanía Martín Barroso su disponibilidad y dedicación así como a todas las personas que han contribuido en mi desarrollo profesional.

También quiero hacer mención de mis amigos que me han animado.

RESUMEN

En el presente proyecto se ha llevado a cabo el desarrollo de una aplicación capaz de generar dinámicamente espacios de trabajo colaborativos que llevan integrado un editor gráfico o un editor textual para que los estudiantes puedan realizar una serie de actividades en equipo. Esta aplicación puede ser utilizada por los profesores que quieran proponer diferentes actividades colaborativas a sus estudiantes y que necesiten para la realización de la misma un editor compartido ya sea gráfico o textual.

Para facilitar la inserción y la gestión de las actividades colaborativas, los profesores dispondrán de una herramienta de autor a través de la cual insertarán nuevos materiales y gestionarán los existentes. También podrán realizar la administración de los alumnos a través de la misma.

Cuando un alumno se conecta para realizar una actividad, se genera dinámicamente una página que contiene los datos de la misma. En el caso de actividades que tengan que resolverse a través de un editor gráfico, se mostrará el enunciado, los iconos asociados y el área de trabajo compartida. Si la actividad lleva integrado un editor textual, se proporcionará una serie de iconos para añadir y eliminar texto al documento compartido del área de trabajo.

El objetivo principal de la aplicación es que los estudiantes puedan resolver en grupos las actividades propuestas utilizando para ello editores colaborativos. Además se facilita al profesor la definición de las actividades colaborativas y la formación de sus grupos de trabajo a través de la herramienta de autor.

ÍNDICE

1. INTRODUCCIÓN.....	1
2. ESTADO DEL ARTE	3
2.1. TEORÍAS DEL APRENDIZAJE.....	4
2.2. COLABORACIÓN	5
2.3. CSCW y CSCL.....	9
2.4. ALGUNOS SISTEMAS CSCL ACTUALES.....	12
3. DESCRIPCIÓN INFORMÁTICA	14
3.1. DESCRIPCIÓN DEL PROBLEMA	15
3.2. METODOLOGÍA EMPLEADA.....	16
3.3. ESPECIFICACIÓN DE REQUISITOS	17
3.4. DISEÑO	22
3.4.1 EDITOR GRÁFICO Y HERRAMIENTA DE AUTOR.....	22
3.4.2 EDITOR TEXTUAL	28
3.4.3 PAQUETES COMUNES	30
3.5. IMPLEMENTACIÓN	31
3.5.1 EDITOR GRÁFICO Y HERRAMIENTA DE AUTOR.....	32
3.5.2 EDITOR TEXTUAL	46
3.5.3 PAQUETES COMUNES	49
3.6. FUNCIONAMIENTO Y PRUEBAS	53
3.6.1 HERRAMIENTA DE AUTOR.....	53
3.6.2 EDITOR GRÁFICO	63
3.6.3 EDITOR TEXTUAL	70
4. CONCLUSIONES.....	73
5. BIBLIOGRAFÍA	75
ANEXO 1: MANUAL DE INSTALACIÓN	79
ANEXO 2: MANUAL DE USUARIO	82

ÍNDICE DE FIGURAS

Figura 1: Taxonomía espacio – tiempo	11
Figura 2: Paquete Alumno del editor gráfico	24
Figura 3: Paquete Profesor de la herramienta de autor.....	25
Figura 4: Paquete ServidorAlumnoProfesor del editor gráfico y herramienta de autor .	26
Figura 5: Paquete Mensajes del editor gráfico y la herramienta de autor	26
Figura 6: Paquete BDCurso del editor gráfico y la herramienta de autor	27
Figura 7: Paquete InformacionAlumnos de los editores y la herramienta de autor	28
Figura 8: Paquete Alumno del editor textual.....	28
Figura 9: Paquete ServidorAlumnoProfesor del editor textual	29
Figura 10: Paquete Mensajes del editor textual.....	30
Figura 11: Paquete packLayoutAbsoluto	30
Figura 12: Paquete VentanaEstandar.....	31
Figura 13: Paquete Concurrencia	31
Figura 14: Clase VentanaAlumno	33
Figura 15: Clase AreaTrabajoJScrollPane.....	34
Figura 16: Clase ElementoGrafico	34
Figura 17: Clase VentanaAlumnoEventHandler	35
Figura 18: Clase AlumnoMessageManager	35
Figura 19: Clase Alumno.....	35
Figura 20: Clase ServerReader	35
Figura 21: Clase BDCurso.....	36
Figura 22: Clase Ejercicio	36
Figura 23: Clase Tema.....	37
Figura 24: Clase RegistroAlumno	37
Figura 25: Clase UserInfo	37
Figura 26: Clase UserInfoList	37
Figura 27: Clase MessageList	38
Figura 28: Clase MessageManager.....	38
Figura 29: Clase Message.....	38
Figura 30: Clase GeneralMessage	39
Figura 31: Clase FinalizacionEjercicioMessage.....	39
Figura 32: Clase InicioSesionMessage.....	39
Figura 33: Clase RegistroAlumnoMessage	39
Figura 34: Clase NuevoElementoGraficoMessage.....	39
Figura 35: Clase ElementoGraficoMessage	39
Figura 36: Clase BDCursoMessage.....	40
Figura 37: Clase CursoMessage	40
Figura 38: Clase TerminadoEjercicioMessage.....	40
Figura 39: Clase Profesor	42
Figura 40: Clase VentanaProfesor	42
Figura 41: Clase ControlTableModelEstandar_Temas	42
Figura 42: Clase ControlTableModelEstandar_Alumnos	43
Figura 43: Clase ControlTableModelEstandar_Ejercicios	43
Figura 44: Clase PanellconosProfesor.....	44
Figura 45: Clase ElementoLista	44

Figura 46: Clase ListClientReader	45
Figura 47: Clase GestorEnvioEjercicioTerminado.....	45
Figura 48: Clase Server	46
Figura 49: Clase ServerMessageManager	46
Figura 50: Clase VentanaAlumno del editor textual	47
Figura 51: Clase Alumno del editor textual.....	47
Figura 52: Clase FilaMessage.....	48
Figura 53: Clase RegistroAlumnoMessage	48
Figura 54: Clase Server	48
Figura 55: Clase LayoutAbsoluto.....	49
Figura 56: Clase LayoutAbsolutoConstraints	49
Figura 57: Interfaz LayoutManagerAbsoluto	49
Figura 58: Clase VentanaEstandar	50
Figura 59: Clase TablaEstandar.....	51
Figura 60: Clase ContenidoTabla.....	51
Figura 61: Clase TableModelEstandar	51
Figura 62: Interfaz ControlTableModel	51
Figura 63: Clase PanelIconos	52
Figura 64: Clase IconoGrafico	52
Figura 65: Clase DinamicProducerConsumerMonitor	52
Figura 66: Clase ProtectedObject.....	52
Figura 67: Clase ReadersWriters.....	53
Figura 68: Clase ProtectedObjectList.....	53
Figura 69: Menú principal herramienta de autor.....	54
Figura 70: Creación de un nuevo curso	55
Figura 71: Gestión del curso actual	56
Figura 72: Gestión de temas	57
Figura 73: Gestión de actividades gráficas.....	58
Figura 74: Inserción de iconos	59
Figura 75: Creación de alumnos y grupos de trabajo	60
Figura 76: Resultados y finalización de actividades gráficas.....	62
Figura 77: Inicio de sesión de un alumno en el editor gráfico	64
Figura 78: Inserción de un elemento en el área de trabajo	65
Figura 79: Desplazamiento de un elemento en el área de trabajo	66
Figura 80: Modificación del tamaño de un elemento del el área de trabajo.....	67
Figura 81: Eliminación de un elemento del área de trabajo	68
Figura 82: Aviso de envío de finalización y del tiempo de finalización.....	68
Figura 83: Aviso del tiempo de finalización de una actividad gráfica	69
Figura 84: Cambio de actividades gráficas y de grupos de trabajo.	69
Figura 85: Finalizar sesión	70
Figura 86: Inicio de sesión en el editor textual.....	71
Figura 87: Menú principal	82
Figura 88: Creación de un nuevo curso	83
Figura 89: Inserción de temas.....	84
Figura 90: Inserción de actividades gráficas	85
Figura 91: Resultados y finalización de actividades gráficas.....	86
Figura 92: Inserción de iconos	87
Figura 93: Creación de alumnos y grupos de trabajo	88



Figura 94: Inicio de sesión de un alumno en el editor gráfico	90
Figura 95: Inicio de sesión en el editor textual.....	91

SECCIÓN 1

INTRODUCCIÓN

El desarrollo de las nuevas tecnologías y de las comunicaciones ha contribuido al gran auge que tiene Internet actualmente. Internet es un medio que permite el acceso a la información, la comunicación y el entretenimiento con unas posibilidades sin precedentes. Al igual que el resto de la sociedad, el mundo educativo no podía quedarse al margen de este fenómeno.

A través de Internet, los usuarios encuentran una gran cantidad de información (enciclopedias, diccionarios, bancos de información especializados, etc.) que se encuentra disponible en cualquier momento y desde cualquier lugar. Esto hace posible que los usuarios complementen su formación de una manera cómoda y flexible. Por ello, se han desarrollado un gran número de sistemas de educación a distancia que ofrecen a los estudiantes un estudio más atractivo y cómodo de diferentes materias.

Muchas universidades y profesores utilizan Internet como medio para difundir el material impartido en sus clases, ofrecer información complementaria, realizar ejercicios tipo test para evaluar los conocimientos de sus alumnos sobre un determinado tema, etc.

Por otro lado, últimamente hay una tendencia al cambio en la enseñanza presencial de tal manera que los alumnos no sean meros receptores de información, sino que sean partícipes de su propio proceso de aprendizaje. Cada vez son más los profesores que proponen a sus estudiantes actividades colaborativas para fomentar la participación de los estudiantes en clase e inculcarles una responsabilidad en su estudio y aprendizaje. Los estudiantes colaboran dentro de un grupo de trabajo para la realización de las actividades propuestas por el profesor.

Gracias a esta tendencia en las clases tradicionales, a los avances tecnológicos que se han producido recientemente y al desarrollo de nuevas herramientas de trabajo en grupo (correo electrónico, chats, foros, etc.) han aparecido dos nuevas áreas de estudio e investigación '*Computer-Supported Cooperative Work*' ó **CSCW** y '*Computer Support for Collaborative Learning*' ó **CSCL**. Ambas soportan la colaboración entre los miembros de un grupo para la realización de una determinada actividad a través de Internet, facilitándoles las herramientas necesarias para el desarrollo de la misma. La primera se enfoca hacia el mundo empresarial, mientras que CSCL se centra en el ámbito de la enseñanza a distancia.

A continuación se presenta la estructura de este documento:

- ✦ En la siguiente sección se explicará cual es el estado del arte de los sistemas colaborativos, centrándose en los entornos colaborativos y educativos.
- ✦ En la tercera sección se describirá el problema a resolver de este Proyecto Fin de Carrera, se explicará la metodología que se ha utilizado para resolverlo, los requisitos de la aplicación, el diseño de la misma, la implementación y las pruebas realizadas.
- ✦ Por último se expondrán las conclusiones extraídas de la realización de este proyecto.

SECCIÓN 2

ESTADO DEL ARTE

A continuación se presenta un resumen del estado del arte centrado en los entornos colaborativos educativos. En primer lugar, se presentan varias teorías sobre el tipo de enseñanza y aprendizaje en las clases presenciales. A continuación se exponen las características de los entornos colaborativos, junto con sus beneficios y dificultades. Las dificultades de estos entornos son debidas principalmente a la gran variedad de configuraciones posibles de los mismos. Estas configuraciones motivaron el trabajo de los pedagogos, el cual se reflejó en varios métodos de intervención pedagógica como ‘*Student Team Learning*’, ‘*Group Investigation*’ y ‘*Learning Together*’. Todos estos métodos ofrecen diferentes criterios para la formación de grupos de trabajo que tengan alto rendimiento. Por último, se presentan las características de CSCW y CSCL así como varios sistemas CSCL actuales.

2.1. TEORÍAS DEL APRENDIZAJE

En general, existen dos enfoques dentro del proceso de enseñanza-aprendizaje en el contexto educativo:

- ✦ **Conductismo:** El alumno es un aprendiz pasivo, un simple receptor de la información que está a merced de los estímulos externos. En este enfoque, el profesor es el portador de la información y actúa como mero transmisor de los conocimientos.
- ✦ **Cognitivismo:** El alumno forma parte activa de su proceso de aprendizaje y el profesor es el encargado de facilitar este proceso al estudiante.

Actualmente, la tendencia en las clases tradicionales es la implantación del cognitivismo en el proceso de formación del alumno para conseguir un proceso de enseñanza-aprendizaje activo. Dentro del cognitivismo, cada vez cobra mayor importancia el **constructivismo**, cuyas características han sido descritas en varios artículos [1] [5].

Ausbel expone que se debe tender a conseguir un aprendizaje significativo debido a la percepción real del conocimiento en lugar de un aprendizaje meramente memorístico [1]. Los profesores deberían presentar a sus alumnos un guión junto con los objetivos de una actividad y guiarles en el proceso de aprendizaje aplicando los

conocimientos previos de tal manera que el alumno sea partícipe activo.

En [5] se muestra que el aprendizaje debería contextualizarse sobre problemas reales. Se debe buscar que el estudiante sea responsable de su proceso de aprendizaje y ofrecerle ayuda y apoyo temporalmente, de forma que se consiga una transferencia gradual de la responsabilidad del aprendizaje del profesor al alumno.

2.2. COLABORACIÓN

Las actividades colaborativas y los entornos colaborativos están basados en el mundo real, donde cada persona es miembro de varios grupos. En nuestra actividad diaria, nos encontramos continuamente interactuando dentro de grupos: en nuestro trabajo, con nuestra familia y amigos, etc. La manera de percibir o de tratar con el resto de los miembros de los grupos va formando nuestra identidad personal. Dentro de los grupos aprendemos a compartir, pensar e interactuar con el resto de los miembros.

El aprendizaje colaborativo fomenta que los estudiantes sean partícipes activos de su propio proceso de aprendizaje y no sean meros receptores de información [33] [41]. Los estudiantes colaboran dentro de grupos de trabajo y son responsables tanto de su trabajo individual como del trabajo global del grupo.

Desde los años 70, el aprendizaje colaborativo '*Cara a cara*' se ha aplicado en las clases tradicionales, aunque la mayoría de los estudios teóricos sobre esta área datan de los años 80 [32]. Durante estos años, surgieron diferentes métodos y estudios que intentan aplicar las tecnologías del aprendizaje colaborativo a estudiantes de diferentes edades y niveles. De estas experiencias se extrajo que el proceso de aprendizaje no es sólo la identificación del conocimiento adquirido, sino que también incluye la identificación de la información que no se conoce o las inconsistencias detectadas en el proceso de aprendizaje como los conceptos que deben ser aclarados o discutidos desde diferentes puntos de vista [42] [41]. En este sentido, el aprendizaje colaborativo es una actividad social que involucra a una comunidad de alumnos en la cual el conocimiento es compartido una vez que se ha adquirido (*construcción del conocimiento*) [6] [29] [8] [26].

Los entornos colaborativos tienen grandes beneficios tales como promover la **cooperación**, interacción y familiaridad entre estudiantes y profesores [http_ref01]. Además, facilitan el desarrollo de **habilidades de razonamiento** [23] [29] tales como expresar ideas, argumentar, realizar críticas constructivas e interactuar con otros estudiantes para construir una solución común [2]. Hay algunas experiencias que demuestran que la **motivación** de los estudiantes, su **participación** y **autoestima** se incrementan cuando obtienen buenos resultados en el desarrollo de actividades colaborativas [http_ref01]. Otro aspecto importante es que estos entornos ayudan a los estudiantes que tienen problemas de expresar sus opiniones en público. En las clases tradicionales cuando el profesor pregunta a un alumno, el centro de atención se encuentra sobre él. En los entornos colaborativos, la atención está distribuida entre todos los componentes del grupo y se crea un entorno seguro en el que el estudiante puede expresar sus ideas sin miedo a fallos o críticas ayudándoles a desarrollar **habilidades de comunicación**. Por último, los profesores pueden evaluar no sólo la solución final de la actividad propuesta sino todo el proceso de aprendizaje.

Una dificultad de estos entornos es que existe una gran variedad de posibles configuraciones que dependen de muchas variables como el tamaño del grupo de trabajo, el criterio de agrupación de los estudiantes, la naturaleza de la actividad colaborativa, la distribución del trabajo entre los miembros del grupo, etc. Estas posibles configuraciones dieron lugar a varios **métodos de intervención pedagógica** tales como ‘*Student Team Learning*’ [32] [33] [34], ‘*Group Investigation*’ [30] y ‘*Learning Together*’ [20] [21] [22] [24] [19].

Slavin desarrolló una serie de métodos pedagógicos para el aprendizaje cooperativo (‘**Student Team Learning**’). Uno de ellos es ‘*Cooperative Integrated Reading and Composition*’, en el cual los estudiantes son distribuidos en grupos de 8 a 15 miembros. El profesor explica un tema relacionado con unos textos que los estudiantes deben leer. Después de leerlos, los estudiantes trabajan en una serie de ejercicios colaborativos con el objetivo de conseguir una mejor comprensión del texto. Posteriormente, se evalúa la comprensión de los textos leídos a través de un test.

Otro de los métodos que desarrolló es ‘*Team Assisted Individualization*’, que combina el aprendizaje en grupo con el aprendizaje individual, aplicándolo al campo de las matemáticas. Los grupos están formados por 4 ó 5 personas. Cada estudiante

trabaja sobre un tema concreto. Después deberán intercambiar soluciones y revisar los resultados comparándolos con los de sus compañeros.

Observando los resultados obtenidos en estos métodos, Slavin concluyó que en los entornos colaborativos es necesario evitar que uno o dos miembros hagan todo el trabajo mientras que los otros adoptan una actitud pasiva. Por tanto, es necesario proporcionar a los miembros de los grupos algunos procedimientos que garanticen que todos ellos aprenden [33].

El método de intervención pedagógica '**Group Investigation**' fue desarrollado por Sharan para intentar evitar la competencia dentro de los grupos de trabajo. Los estudiantes son responsables de la creación de grupos de trabajo (2-6 personas) y de la selección de las actividades a realizar. Cada grupo elige una parte de la unidad que están estudiando y dividen el trabajo a realizar en varias subtareas. Cada subtarea será realizada de forma individual o por parejas. Una vez que todas las parejas o individuos han obtenido su solución parcial, se juntan para elaborar una solución común. A continuación el profesor evalúa estas soluciones. Este método fomenta la discusión entre los miembros del grupo a la hora de adoptar una solución común y desarrolla en los estudiantes la capacidad de síntesis.

Por último, '**Learning Together**' fue desarrollado por Johnson y Johnson y ofrece varias consideraciones a tener en cuenta para la formación de grupos de trabajo. El tamaño de los grupos depende del '**T.E.A.M.**' (límite de tiempo, experiencia de los estudiantes en el trabajo en grupo, edad de los alumnos y materiales disponibles). Aunque normalmente los grupos de aprendizaje colaborativo suelen estar formados por 2, 3 y 4 componentes, la regla básica es "cuanto más pequeño mejor". En este método se dan unas pautas que el profesor debe tener en cuenta a la hora de formar los grupos de trabajo:

- ✦ Añadir un nuevo miembro al grupo puede incrementar los recursos del mismo.
- ✦ Los grupos pequeños suelen tardar menos tiempo en organizarse y empezar a trabajar.
- ✦ El tamaño de los grupos estará influenciado por los materiales disponibles y por la naturaleza de la actividad.

- ✦ Los grupos pequeños incrementan el esfuerzo que tiene que hacer cada estudiante para resolver una determinada actividad colaborativa. Es más difícil que un alumno se mantenga pasivo en un grupo de trabajo pequeño.
- ✦ Los grupos pequeños ayudan a identificar y resolver dificultades que los estudiantes pueden tener al trabajar juntos (conflictos entre los miembros, liderazgo y control, etc.).

Dependiendo del perfil de los miembros de los grupos de trabajo y de la duración de los mismos, se pueden hacer dos tipos de clasificaciones. La primera tiene en cuenta el perfil de los miembros de los grupos: *homogéneos vs heterogéneos*. Normalmente, los grupos de trabajo homogéneos (estudiantes con las mismas habilidades, experiencias e intereses) pueden alcanzar objetivos específicos antes que los grupos heterogéneos. Sin embargo, los grupos heterogéneos obtienen más ventajas que los homogéneos en los entornos colaborativos. Si el proceso de formación de grupos se deja en manos de los estudiantes, éstos tienden a formar grupos homogéneos mientras que si lo realiza el profesor puede formar grupos aleatorios (heterogéneos) u homogéneos [20].

La segunda clasificación que se puede hacer sobre los grupos colaborativos está basada en la duración de los mismos. Según esta clasificación, podemos tener tres tipos de grupos de trabajo [38]:

- ✦ *Informales*: Se constituyen para discutir cuestiones o resolver problemas durante una sesión de clase tradicional. Existen durante un breve periodo de tiempo.
- ✦ *Formales*: Están encaminados para resolver una tarea que puede abarcar desde una sesión de clase a varias semanas.
- ✦ *De base*: Son grupos a largo plazo y controlan el avance y la progresión de cada uno de sus componentes en ámbitos que van más allá de lo meramente académico.

Al igual que Slavin [33], en [38] se enfatiza que es importante que los estudiantes sean conscientes de que el éxito del grupo depende del éxito de cada uno de ellos en los entornos colaborativos. El trabajo de la actividad tiene que estar dividido de tal manera que ningún miembro del grupo pueda permanecer pasivo. De esta

manera, se consigue que el éxito del grupo dependa del éxito de cada uno de los componentes (*interdependencia positiva*) y que cada uno de sus miembros sea responsable tanto de su trabajo individual como del trabajo colectivo del grupo de trabajo. Además es necesario proveer la interacción entre los componentes del grupo de trabajo facilitándoles las herramientas adecuadas [http_ref01].

2.3. CSCW y CSCL

El término **CSCW** (*Computer-Supported Cooperative Work*) lo introducen Grief y Cashman como “una vía para describir cómo la tecnología de los computadores puede ayudar a los usuarios a trabajar juntos en grupos” [10]. En [12] y [http_ref01] se describe CSCW como un sistema basado en una red de comunicaciones entre computadoras que soporta el trabajo en grupo para realizar tareas comunes y que proporciona una interfaz compartida.

Las primeras propuestas sobre el área del aprendizaje colaborativo utilizando ordenadores datan de los años 80 [31]. Más tarde, gracias al desarrollo de las nuevas comunicaciones y tecnologías derivadas del área del groupware, nacieron nuevas aplicaciones dentro de un área de estudio llamado ‘*Computer Support for Collaborative Learning*’ (**CSCL**) [25]. CSCL surge de la aplicación de CSCW al área del aprendizaje colaborativo. Las diferencias entre CSCW y CSCL es que CSCW suele centrarse más en las técnicas de comunicación mientras que CSCL se centra en lo que se está comunicando. CSCW se usa principalmente en entornos empresariales para facilitar la comunicación de los miembros de un grupo de trabajo, mientras que CSCL se aplica a entornos educativos para ayudar a los estudiantes a aprender juntos de una manera eficaz.

El objetivo de CSCL es mejorar el proceso de aprendizaje de los alumnos proponiéndoles actividades comunes y poniendo a su disposición herramientas que soporten la comunicación e interacción entre ellos, la resolución cooperativa de problemas o la construcción del conocimiento. En el campo de la enseñanza a través de Internet, las herramientas colaborativas han sido utilizadas para reducir la sensación de aislamiento que tienen los estudiantes en los cursos on-line y para promover el desarrollo de ciertas habilidades personales y sociales tales como razonar, argumentar o aprender a trabajar en grupo.

Una de las ventajas de CSCL es que permite que los estudiantes puedan trabajar desde su domicilio con otros compañeros sin necesidad de trasladarse a un lugar común u organizar una reunión de trabajo. Esto se traduce en un mejor aprovechamiento de los recursos y del tiempo ya que el estudiante puede elegir cualquier momento para conectarse con los compañeros y trabajar en grupo. Además esto lo puede realizar desde cualquier lugar con conexión a Internet.

De los métodos de intervención pedagógica aplicados al aprendizaje colaborativo presencial y explicados en la subsección anterior (*'Student Team Learning'*, *'Group Investigation'* y *'Learning Together'*) es posible extraer propuestas iniciales y adaptarlas al área de CSCL.

Desde un punto de vista teórico, CSCL está basado en varias teorías [26] tales como la *'Sociocultural Theory'* [40] [27] [28] [13], *'Constructivism Theory'* [8] [http_ref04] y *'Situated Cognition'* [7]. Todas estas teorías asumen que los individuos son agentes activos que buscan y construyen su conocimiento dentro de un contexto [http_ref01], además de asumir que el conocimiento está en constante evolución.

La *'Teoría sociocultural'* (*Sociocultural Theory*) enfatiza que la inteligencia humana se origina en nuestra sociedad o cultura. El aprendizaje consiste en la participación activa del estudiante vs. la presentación pasiva de la información por un experto. Los estudiantes construyen su conocimiento y aprenden a entender y apreciar las diferentes perspectivas a través de un diálogo con sus iguales.

La *'Teoría del Constructivismo'* (*Constructivism Theory*) postula que el conocimiento no es algo fijo si no que está en continua evolución. Cada persona va construyendo su conocimiento en base a su propia experiencia. En el aprendizaje colaborativo, las personas involucradas ven los problemas desde puntos de vista diferentes y son capaces de negociar y generar una solución común [http_ref01].

Otros estudios de esta área argumentan que el aprendizaje significativo sólo tendrá lugar si está dentro de un contexto físico y social dentro del cual puede ser usado (*'Situated Cognition'*) [7].

Por otro lado, dependiendo de la localización física de los estudiantes y del momento en el cual se produzcan las interacciones entre los miembros de los grupos de trabajo, se puede obtener una clasificación basada en estas dos dimensiones (ver figura número 1) [11].

	MISMO MOMENTO	DIFERENTE MOMENTO
MISMO LUGAR	Interacción cara a cara	Interacción asíncrona
DIFERENTE LUGAR	Interacción distribuida síncrona	Interacción distribuida asíncrona

Figura 1: Taxonomía espacio – tiempo

Según esta taxonomía, puede haber cuatro tipos de interacción:

- ✦ ‘*Cara a cara*’: Los estudiantes se encuentran en el mismo lugar e interactuando en el mismo instante de tiempo. Por ejemplo, los grupos de trabajo en las clases tradicionales o el pilotaje de un avión dentro de la cabina del mismo.
- ✦ ‘*Interacción asíncrona*’: Se produce en el mismo lugar pero en diferentes momentos. Por ejemplo, estudiantes que estén trabajando en una práctica con el ordenador en una clase presencial y vuelven al mismo aula al siguiente día para continuar con su trabajo. Otro ejemplo es el desarrollo de un proyecto común por parte de diferentes usuarios en la misma computadora.
- ✦ ‘*Interacción distribuida síncrona*’: Se produce cuando se interactúa desde diferentes lugares pero en el mismo instante de tiempo. Por ejemplo una teleconferencia o un chat.
- ✦ ‘*Interacción distribuida asíncrona*’: Tiene lugar desde diferentes lugares y en distintos instantes de tiempo. Como ejemplos tenemos los foros o el envío de mensajes por correo electrónico.

Si los alumnos se encuentran distribuidos en diferentes lugares, es necesario proporcionarles las herramientas necesarias que permitan la cooperación, comunicación en intercambio de ideas de los miembros del grupo, facilitar una interfaz compartida y controlar la concurrencia de las acciones sobre el espacio de trabajo

compartido. Algunas de las herramientas colaborativas más utilizadas son los foros, chats, ircs, correo electrónico [3] [4], herramientas para intercambiar archivos y documentación [http_ref05], sistemas de votación [http_ref02], calendarios [18] [http_ref02], videoconferencias [16] ó editores gráficos y textuales [http_ref03] [38]. Hay varios sistemas que combinan el uso de varias de estas herramientas como PHProject, una aplicación modular para la coordinación y colaboración de grupos de actividades a través de Internet [http_ref02].

2.4. ALGUNOS SISTEMAS CSCL ACTUALES

A continuación, se presenta un resumen de las características de algunos de los sistemas CSCL actuales como 'SNS' [http_ref06], 'DEGREE' [http_ref07], 'KÜKÄKÜKÄ' [http_ref08] y 'Lecture 2000' [http_ref09].

Shadow NetworkSpace (SNS)

Es un entorno CSCL que soporta una comunidad de estudiantes. Los miembros (profesores, estudiantes, etc.) tienen herramientas para representar, organizar, compartir recursos y colaborar. Incluye seguridad de acceso, roles de los usuarios y los tipos de los grupos, sistema de ficheros, tareas de administración, chat, foros, notas, creación de documentos y notificación de deberes.

Distance Education Environment for GRoup ExperiencEs (DEGREE)

DEGREE fue desarrollado por Verdejo y Barros. Es un sistema para realizar y evaluar experiencias del aprendizaje colaborativo en la educación a distancia. Este sistema colaborativo soporta la realización de una variedad de tareas de aprendizaje en pequeños grupos de estudiantes. Una experiencia puede constar de una o más actividades y cada actividad por una o más tareas. Estas actividades pueden ser independientes entre ellas o tener restricciones entre ellas. La colaboración entre los estudiantes es asíncrona. Cada estudiante recibe una notificación acerca de la evolución del trabajo común. Todas las acciones e interacciones entre estudiantes se guardan para analizar el resultado de la experiencia y el proceso que permite obtener el resultado.

Kükäkükä

Los autores construyeron este sistema pensando que los estudiantes podían referenciar, manipular y discutir acerca del uso de los artefactos externos (bocetos, pinturas y objetos) en el aprendizaje a través de Internet. Los entornos de discusión online deberían permitir la inclusión de los artefactos visuales (gráficas de datos, vídeos del trabajo del alumno, soluciones de problemas, etc.). Estos artefactos deberían existir fuera de los mensajes individuales y estar visibles durante la conversación. Los participantes deberían ser capaces de cambiar los artefactos en la discusión de forma natural. Para ello, crearon una interfaz que permite tener varios hilos de discusión activos. Los artefactos disponibles en cada una de las discusiones incluyen pequeñas redacciones de estudiantes, soluciones a pequeños diseños de trabajo, documentación de sus proyectos, etc.

Lecture 2000

Soporta el acceso al conocimiento, información y habilidades para construir conocimientos de forma individual o en un equipo. Esto proporciona a los alumnos un espacio de información y soporta la colaboración entre estudiantes y profesores con herramientas síncronas y asíncronas.

El espacio de información está estructurado de acuerdo con el concepto del hipertexto. Dependiendo de los intereses del usuario, se mostrará el nivel apropiado de detalle de las unidades. Este espacio de información está sujeto a continuas evoluciones que dan lugar a actualizaciones y a la integración de nuevo material. Tanto los profesores como los estudiantes pueden hacer modificaciones.

SECCIÓN 3

DESCRIPCIÓN INFORMÁTICA

En esta sección se presenta la descripción del problema que se ha resuelto en este Proyecto Fin de Carrera, después se detalla la especificación de requisitos, el diseño y cómo se ha realizado la implementación. Finalmente se describe el funcionamiento de la aplicación junto con las pruebas realizadas.

3.1. DESCRIPCIÓN DEL PROBLEMA

El objetivo de este Proyecto Fin de Carrera es crear una aplicación que genere dinámicamente espacios de trabajo colaborativos. Esta aplicación podrá ser utilizada por profesores para proponer a sus estudiantes distintas actividades que tendrán que resolver en grupos de trabajo. Para el desarrollo de las actividades, se podrá utilizar un editor gráfico o un editor textual compartido. Dentro de cada grupo, los estudiantes podrán tener asignados dos tipos de roles diferentes: estudiante 'jefe' del grupo o alumno normal. Si un alumno es 'jefe' de una determinada actividad, podrá finalizarla.

Para facilitar la gestión de las distintas actividades que precisen de un editor gráfico y sus grupos de trabajo, deberá existir una herramienta de autor a través de la cual los profesores puedan:

- ✦ Crear, modificar y eliminar temas.
- ✦ Definir, modificar y eliminar actividades gráficas para cada uno de los temas del curso.
- ✦ Añadir y eliminar alumnos al curso.
- ✦ Formar, modificar y eliminar grupos de trabajo.
- ✦ Definir roles en los grupos de trabajo: jefe del grupo de trabajo o estudiante normal.
- ✦ Finalizar una determinada actividad gráfica.
- ✦ Consultar y generar un documento pdf con el resultado del trabajo actual de los estudiantes.

Además se ha de facilitar a los profesores el proceso de creación de actividades gráficas colaborativas permitiéndoles la reutilización de material almacenado en la aplicación de cursos anteriores.

Los estudiantes podrán interactuar entre ellos y realizar las actividades gráficas propuestas de manera conjunta siempre y cuando estén dados de alta en la aplicación.

En el caso de actividades que precisen de un editor gráfico, el estudiante deberá poder seleccionar una actividad gráfica concreta de una lista que contendrá cada una de las actividades gráficas pendientes de realizar. Una vez seleccionada la actividad gráfica que desea realizar, se le deberá presentar al estudiante el enunciado de la misma, el conjunto de iconos asociados y el área de trabajo compartida en la que los estudiantes del mismo grupo de trabajo la desarrollarán. Cada actividad gráfica podrá ser propuesta a varios grupos de estudiantes. Si una actividad es finalizada por uno de los estudiantes ‘jefes’ del grupo, la aplicación deberá avisar a los miembros del mismo grupo de este hecho.

En el caso de actividades con un editor textual, el estudiante cargará la actividad textual a realizar. La aplicación permitirá al estudiante añadir filas de texto a un documento compartido y guardarlo en formato .pdf.

3.2. METODOLOGÍA EMPLEADA

Se ha seguido un proceso de desarrollo del software en el que se realizan un conjunto de fases que transforman una serie de requisitos en una aplicación software. Las fases han sido: especificación de requisitos, diseño, implementación y pruebas. En la realización de dicho proceso, ha sido bastante práctico dividir el trabajo en pequeñas partes o miniproyectos. Cada parte ha sido una ampliación de la anterior y por tanto se ha seguido un enfoque iterativo e incremental.

En la especificación de requisitos se han detallado todas las funcionalidades que la aplicación ha de cumplir. Para poder diseñar esta aplicación, se ha utilizado el ‘Lenguaje Unificado de Modelado’ (UML) [15] y la herramienta ‘Rational Rose’ describiendo así los módulos, las clases y las relaciones que posee.

La implementación de este proyecto se ha realizado en el lenguaje de programación Java para permitir el uso de la aplicación en todo tipo de plataformas.

Respecto a la base de datos de la aplicación, al ser sencilla y no requerir seguridad, exportación de información, redundancia, etc; se ha utilizado la serialización de objetos de Java para su almacenamiento. La serialización permite almacenar los objetos en ficheros que pueden ser recuperados posteriormente al inicio de la aplicación.

Finalmente, se han realizado pruebas para comprobar que los requisitos propuestos inicialmente se cumplen.

3.3. ESPECIFICACIÓN DE REQUISITOS

La primera etapa que hay que realizar para crear un producto software es la extracción de los requisitos del mismo. A continuación se detallan los requisitos que debe cumplir la aplicación desarrollada en este proyecto. Éstos se encuentran divididos en dos categorías: funcionales (RF) y no funcionales (RNF).

Requisitos funcionales (RF):

Primero se van a definir los requisitos generales y a continuación los que conciernen al uso de la aplicación por parte de los profesores (herramienta de autor) y por parte de los alumnos (editores).

✦ **Generales:**

- ✦ **RF.01 Servidor:** Para gestionar todas las actividades que necesiten tanto un editor gráfico como un editor textual se necesitará que se esté ejecutando el servidor correspondiente.

✦ **Profesor. Herramienta de autor:**

- ✦ **RF.02 Creación de temas:** El profesor podrá insertar nuevos temas en un curso. Cada tema tendrá un nombre y una serie de actividades gráficas asociadas.

- ✦ **RF.03 Modificación de temas:** El profesor podrá modificar los temas que contiene el curso añadiéndole ejercicios o eliminando los creados.
- ✦ **RF.04 Eliminación de temas:** Se podrán eliminar temas de un curso.
- ✦ **RF.05 Creación de actividades gráficas:** Se podrán crear actividades gráficas asociadas a cada uno de los temas. Cada actividad gráfica estará descrita por un nombre, un enunciado y una serie de iconos. La herramienta deberá permitir al profesor la reutilización del material almacenado en la aplicación o el suministro de nuevos iconos.
- ✦ **RF.06 Modificación de actividades gráficas:** El profesor podrá modificar tanto el enunciado de una actividad gráfica como los iconos asociados.
- ✦ **RF.07 Eliminación de actividades gráficas:** Se permitirá la eliminación de actividades gráficas de cada uno de los temas.
- ✦ **RF.08 Finalización de actividades gráficas:** El profesor podrá finalizar cada una de las actividades gráficas del curso indicando el tiempo restante. La aplicación informará a todos los alumnos que tengan pendiente esa actividad gráfica del tiempo que les queda para terminarla. Pasado ese tiempo, se impedirá a los alumnos que sigan trabajando en la actividad.
- ✦ **RF.09 Creación de alumnos:** El profesor podrá añadir alumnos a un curso suministrando un nombre de usuario y una contraseña.
- ✦ **RF.10 Eliminación de alumnos y finalización de sesión:** Se permitirá la eliminación de alumnos. Si el alumno eliminado estuviera conectado en ese momento, la aplicación finalizará su sesión de trabajo.
- ✦ **RF.11 Creación grupos de trabajo:** El profesor podrá añadir grupos de trabajo a un curso. Cada uno de ellos estará compuesto por una serie de alumnos y una actividad gráfica que tienen que realizar. Dentro de cada grupo los alumnos tendrán asignado el rol de jefe o el rol de estudiante normal.

✦ **RF.12 Modificación de grupos de trabajo:** Se permitirá la modificación de los grupos de trabajo y de los roles de los alumnos en cada uno de ellos.

✦ **RF.13 Visualización y almacenamiento de los resultados de un grupo de trabajo:** El profesor del curso podrá ver el estado actual del resultado en el que se encuentra cada actividad gráfica de un determinado grupo de trabajo en un momento concreto. Además se le permitirá guardarlo en un documento pdf.

✦ **Alumno:**

✦ **RF.14 Inicio de sesión:** En el caso del editor gráfico sólo podrán iniciar sesión los alumnos que tengan un nombre de usuario y contraseña válido. Tras la identificación, se mostrará al estudiante una pantalla con la lista de las actividades gráficas pendientes de realizar. En el caso de actividades que precisen un editor textual deberán iniciar sesión y se mostrará el documento realizado por el grupo de trabajo de la actividad textual en el caso que lo hubiera.

✦ **Alumno. Editor gráfico:**

✦ **RF.15 Selección de actividad gráfica o grupo de trabajo:** El alumno podrá seleccionar una actividad gráfica del listado de actividades gráficas pendientes. Tras ello, la aplicación visualizará su enunciado y le permitirá, si no está finalizada, trabajar en un área de trabajo en grupo. En ella se mostrará el trabajo previo en el caso que lo hubiera.

✦ **RF.16 Acciones sobre el área de trabajo de una actividad gráfica:** El alumno podrá insertar un nuevo icono, modificar su tamaño, moverlo o eliminarlo. La acción quedará reflejada en el área de trabajo del profesor y en el de todos los alumnos del mismo grupo.

- ✦ **RF.17 Finalización de actividad gráfica:** Los alumnos que tengan asignado el rol de jefe, podrán finalizar una actividad gráfica pulsando un botón de finalización de actividad gráfica.
- ✦ **RF.18 Aviso de cambio:** Si el profesor cambia los grupos de trabajo o las actividades gráficas pendientes por realizar de un alumno, aparecerá una ventana informando de este hecho y se mostrarán los nuevos grupos de trabajo, así como las actividades gráficas pendientes de realizar.
- ✦ **RF.19 Aviso de finalización de sesión:** Si el profesor finaliza la sesión de trabajo de un alumno, éste recibirá un mensaje por pantalla y se cerrará su sesión de trabajo.
- ✦ **RF.20 Aviso de finalización de actividades gráficas:** Si el profesor o un alumno finaliza una actividad gráfica, todos los compañeros del grupo de trabajo de la actividad gráfica recibirán un mensaje por pantalla indicando el tiempo que queda para finalizarla. Cuando el tiempo expire, los alumnos no podrán seguir trabajando en dicha actividad.

- ✦ **Alumno. Editor textual**
 - ✦ **RF.21 Insertar filas:** El alumno podrá insertar una fila en la parte superior o inferior de otra del documento compartido. Tras ello, la aplicación reflejará los cambios en las áreas de trabajo de todos los miembros del mismo grupo.
 - ✦ **RF.22 Eliminar filas:** El alumno podrá eliminar una fila de un documento. Tras ello, la aplicación actualizará el documento de todos los miembros del grupo de trabajo.
 - ✦ **RF.23 Iniciar documento:** El alumno podrá borrar todo el documento en cualquier momento. Tras ello, la aplicación lo visualizará en el documento de todos sus miembros del grupo de trabajo.

- ✦ **RF.24 Guardar documento:** Los alumnos podrá guardar el documento en formato pdf.

Requisitos no funcionales (RNF):

- ✦ **RNF.01 Persistencia:** Cualquier modificación realizada por profesores y alumnos quedará guardada en los servidores.
- ✦ **RNF.02 Portabilidad:** La aplicación debe ser portable a diversos sistemas operativos.
- ✦ **RNF.03 Interfaz gráfico:** Debe ser intuitivo y de fácil uso tanto para los alumnos como para los profesores.
- ✦ **RNF.04 Interfaz de la pantalla generada de la herramienta de autor:** La pantalla generada para la herramienta de autor del profesor debe contener cuatro botones, tres de ellos dedicados a las operaciones relacionadas con los temas, actividades gráficas y alumnos. El último botón contendrá la opción de salir de la herramienta.
- ✦ **RNF.05 Interfaz de la pantalla generada del editor gráfico:** La pantalla generada para las actividades que contengan un editor gráfico estará dividida en cinco zonas. La parte superior deberá contener en el lado izquierdo un listado con las actividades pendientes de realizar y en el lado derecho se cargará el enunciado de las mismas. En la parte lateral izquierda central se encontrarán ubicados los iconos de la actividad actual. En la parte inferior se mostrará una zona de mensajes y por último en la parte central derecha, se localizará el área compartida del editor gráfico.
- ✦ **RNF.06 Interfaz de la pantalla generada del editor textual:** La pantalla generada para las actividades que contengan un editor textual estará dividida en tres zonas. La parte superior contendrá cinco botones: añadir fila arriba, añadir fila abajo, eliminar fila, iniciar documento y guardar documento. En la parte central de la pantalla estará el área donde se irá construyendo el documento compartido y

que ocupará aproximadamente el 75% de la pantalla. Por último, en la parte inferior se mostrará un área con los mensajes de la aplicación.

3.4. DISEÑO

En el diseño de la aplicación, se ha seguido el paradigma cliente – servidor. En este paradigma, el cliente y el servidor se comunican por medio de mensajes. En el modo operativo habitual diseñado, el cliente manda un mensaje al servidor y éste le responde con otro o lo reenvía a una serie de clientes.

En el caso de la herramienta de autor y del editor gráfico se han diseñado como clientes diferentes que interactúan con un único servidor. En el caso del editor textual, se ha diseñado como un único cliente que interactúa con un único servidor. Por este motivo, se ha dividido la aplicación en dos partes diferenciadas. La primera corresponde al editor gráfico y a la herramienta de autor y la segunda al editor textual.

Lo primero que se ha diseñado en cada una de las partes han sido los paquetes. Cada uno representa un dominio y aporta funcionalidades que pueden utilizar otros paquetes. Cada paquete se ha diseñado mediante un diagrama de clases que representa su distribución del trabajo. A continuación se procede a explicar el diseño y funcionamiento de cada uno de ellos.

3.4.1 EDITOR GRÁFICO Y HERRAMIENTA DE AUTOR

Se ha diseñado un paquete para el cliente editor gráfico llamado '*Alumno*' (ver figura 2), un paquete para el cliente herramienta de autor llamado '*Profesor*' (ver figura 3), un paquete para el servidor llamado '*ServidorAlumnoProfesor*' (ver figura 4) y una serie de paquetes adicionales para que los utilicen los paquetes anteriores.

Funcionamiento del paquete '*Profesor*' e interacción con el servidor '*ServidorAlumnoProfesor*':

Cuando un profesor carga la herramienta de autor (representada en la clase '*VentanaProfesor*' de la figura 3), ésta le permite la creación o gestión de un curso. Si el profesor decide gestionar un curso, la herramienta se conecta al servidor para pedirle mediante un mensaje '*BDCursoMessage*' (ver figura 5) el curso vigente.

Cuando un profesor crea un curso nuevo o realiza un cambio en un curso actual (cambiar el enunciado o los iconos de las actividades gráficas, cambiar los integrantes de un grupo de trabajo, etc.), la herramienta envía un mensaje '*BDCursoMessage*' (ver figura 5) al servidor. Éste guarda los cambios realizados y envía un mensaje '*RegistroAlumno*' (ver figura 6) a todos los editores de los alumnos a los que les afecte el cambio en la base de datos. Tras ello, estos editores actualizarán las actividades gráficas a mostrar.

Cuando la herramienta recibe un cambio en el resultado de una actividad gráfica, lo guarda para permitir visualizar el nuevo resultado al profesor o guardarlo en un documento con formato pdf.

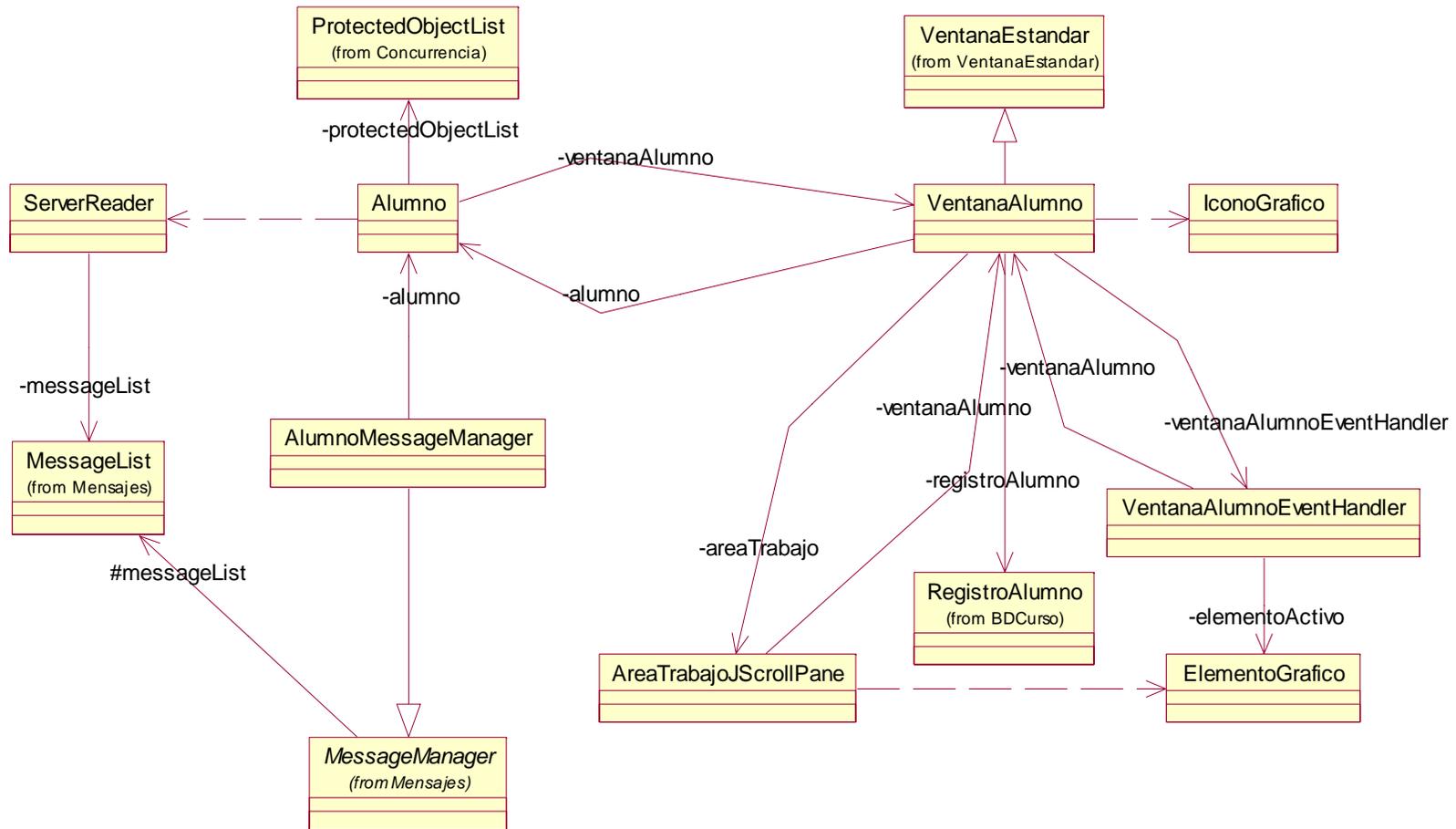


Figura 2: Paquete Alumno del editor gráfico

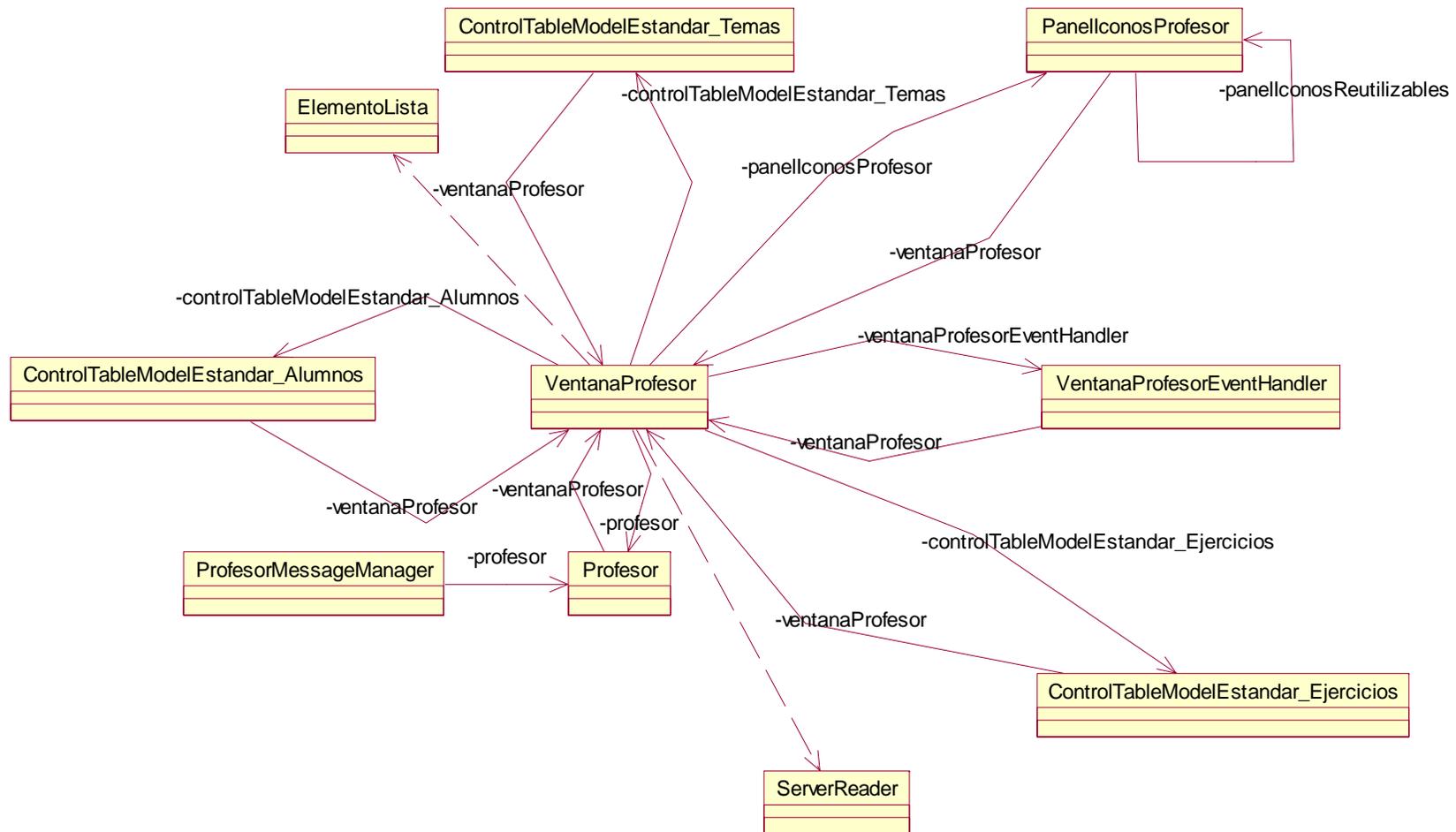


Figura 3: Paquete Profesor de la herramienta de autor

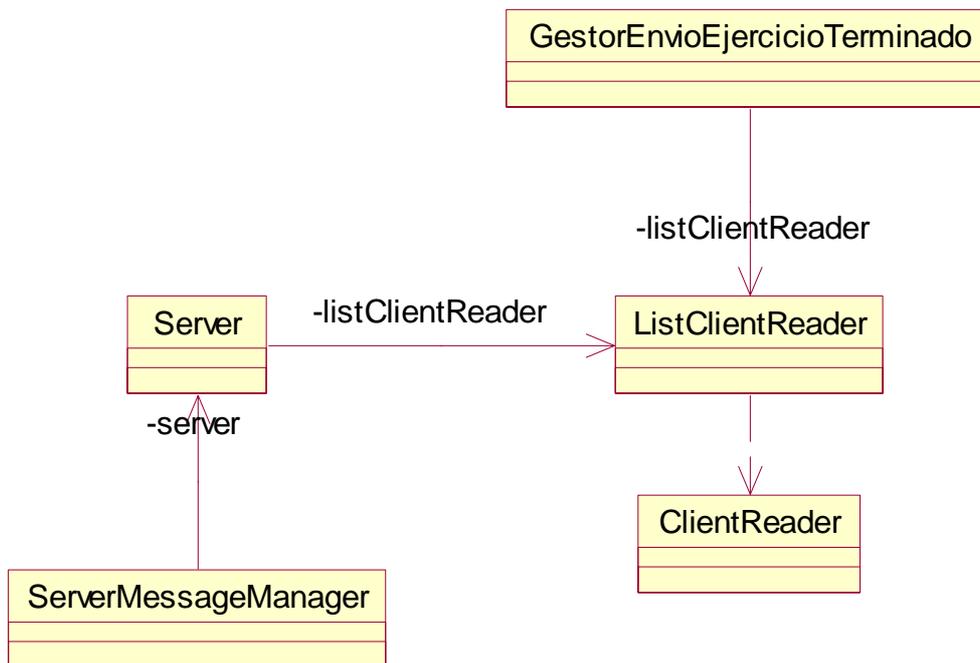


Figura 4: Paquete ServidorAlumnoProfesor del editor gráfico y herramienta de autor

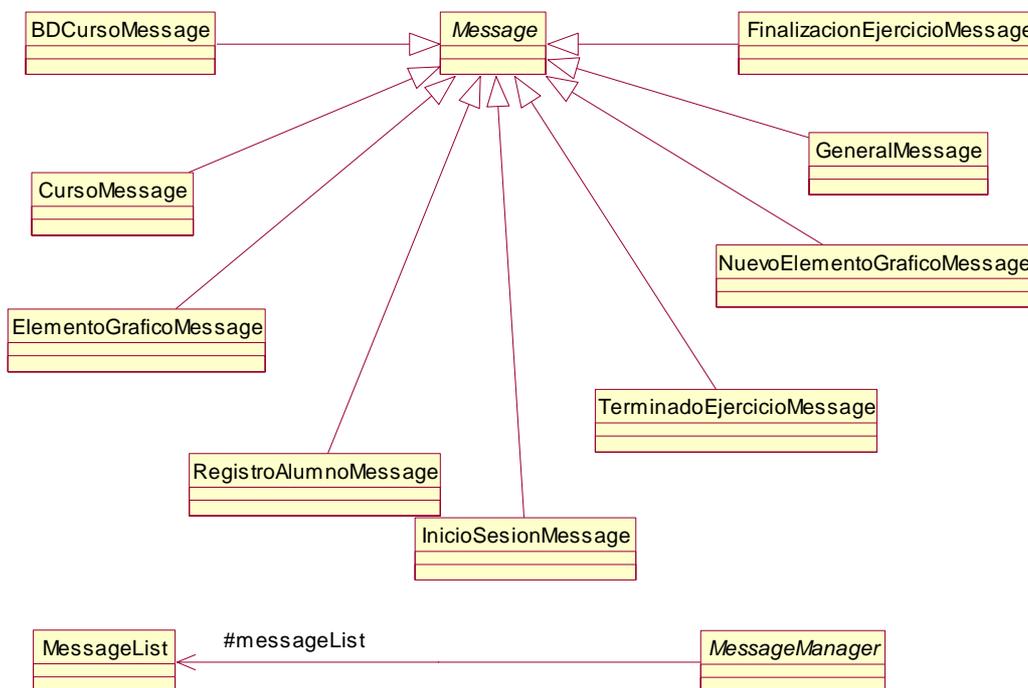


Figura 5: Paquete Mensajes del editor gráfico y la herramienta de autor

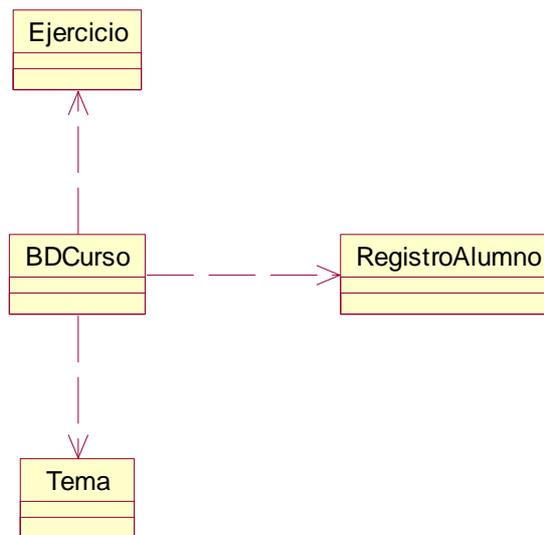


Figura 6: Paquete BDCurso del editor gráfico y la herramienta de autor

Funcionamiento del paquete *'Alumno'* del editor gráfico e interacción con el servidor *'ServidorAlumnoProfesor'*:

Cuando un alumno carga el editor gráfico (representado en la clase *VentanaAlumno* en la figura 2), éste le pide al alumno la información de inicio de sesión, es decir el nombre y la contraseña de usuario y la guarda en *'UserInfo'* (ver figura 7). Tras ello, el editor se conecta al servidor para mandársela mediante un mensaje *'InicioSesionMessage'* (ver figura 5). Cuando el servidor lo recibe, si el alumno se encuentra en la base de datos del curso (*'BDCurso'* en figura 6), envía al editor un mensaje *'RegistroAlumnoMessage'* (ver figura 5) con toda la información de los grupos de trabajo a los que pertenece el alumno (incluido el estado del resultado de las actividades gráficas). En el caso de que el alumno no se encuentre en la base de datos del curso, el servidor enviaría un mensaje al editor en el que se lo indicaría y el editor no permitiría alumno trabajar en el curso.

Una vez realizado el registro correctamente, el alumno accede mediante el editor gráfico a todas las actividades gráficas que ha recibido al registrarse y que tiene pendientes de realizar (*'Ejercicio'* en figura 6). Puede resolverlas colaborativamente con sus compañeros fijándose en el enunciado que le muestra el editor para cada una de ellas y usando los iconos gráficos asociados a cada actividad gráfica.

Cuando un alumno realiza un cambio en el área de trabajo de una actividad gráfica, el editor envía un mensaje *'NuevoElementoGraficoMessage'* ó *'ElementoGraficoMessage'* (ver figura 5) al servidor y este lo reenvía a todos sus compañeros de trabajo de esa actividad y al profesor. El servidor también guarda el cambio.

Una vez que un alumno con el rol de jefe finaliza una actividad gráfica, el editor envía un mensaje *'TerminadoEjercicioMessage'* (ver figura 5) al servidor y éste lo reenvía a todos sus compañeros de trabajo para esa actividad gráfica y al profesor.

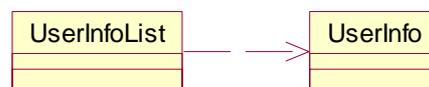


Figura 7: Paquete InformacionAlumnos de los editores y la herramienta de autor

3.4.2 EDITOR TEXTUAL

Se ha diseñado un paquete para el cliente Editor textual llamado *'Alumno'* (ver figura 8), un paquete para el servidor llamado *'ServidorAlumnoProfesor'* (ver figura 9) y una serie de paquetes adicionales para que los utilicen los paquetes anteriores.

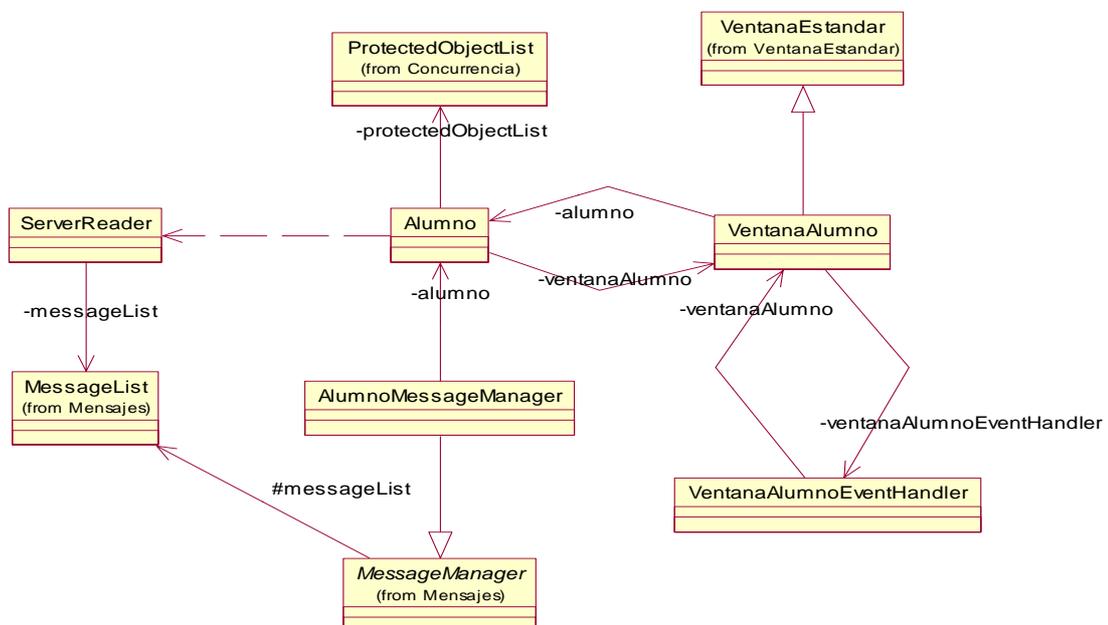


Figura 8: Paquete Alumno del editor textual

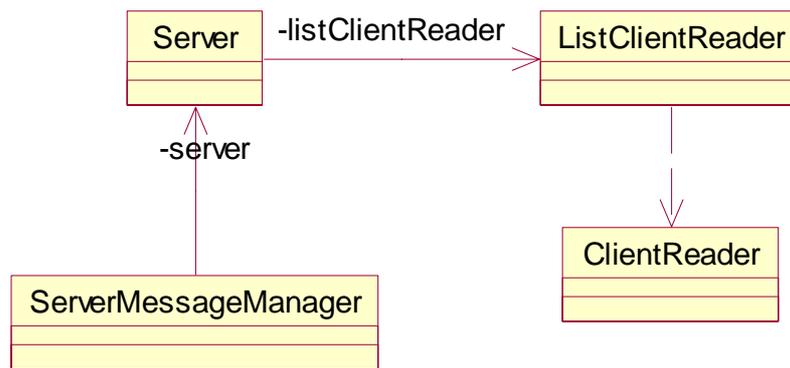


Figura 9: Paquete ServidorAlumnoProfesor del editor textual

Funcionamiento del paquete '*Alumno*' del editor textual e interacción con el servidor '*ServidorAlumnoProfesor*':

Cuando un alumno carga el editor textual (representado en la clase '*VentanaAlumno*' de la figura 8), éste le pide al alumno la información de inicio de sesión y la guarda en '*UserInfo*' (ver figura 7). A continuación, el editor se conecta al servidor para mandársela mediante un mensaje '*InicioSesionMensaje*' (ver figura 10). Cuando el servidor lo recibe envía al editor un mensaje '*RegistroAlumnoMensaje*' (ver figura 10) con toda la información del documento textual en el que trabajar (incluido su estado en ese instante).

Tras realizar el registro correctamente, el alumno accede mediante el editor textual al documento compartido. Puede trabajar en él colaborativamente con sus compañeros usando los botones de añadir fila arriba o abajo, eliminar fila e inicializar documento. También puede generar un pdf del documento actual.

Si un alumno realiza un cambio en el documento, el editor envía un mensaje '*FilaMensaje*' o '*GeneralMensaje*' (ver figura 10) al servidor. Éste lo reenvía a todos sus compañeros de trabajo y guarda el cambio realizado.

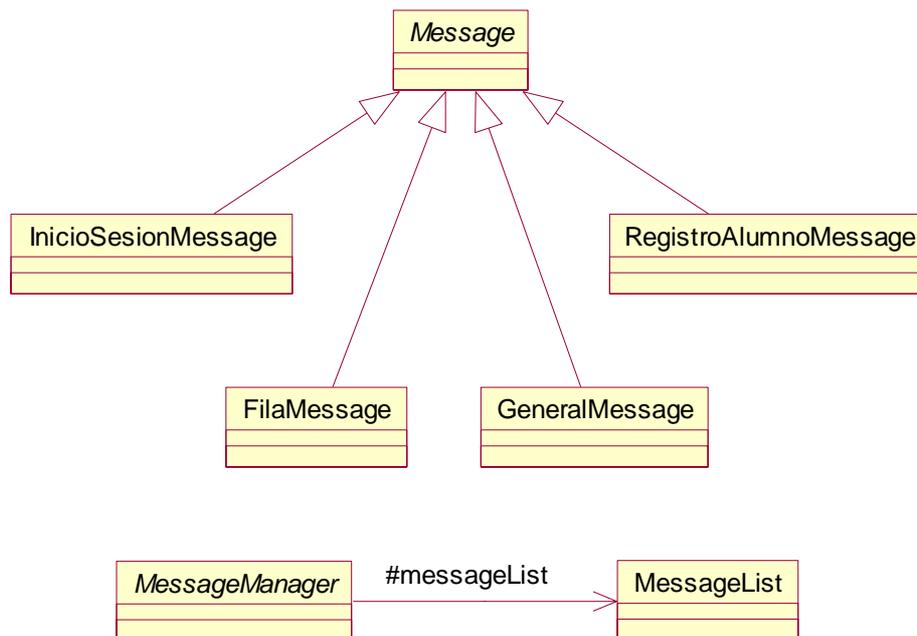


Figura 10: Paquete Mensajes del editor textual

3.4.3 PAQUETES COMUNES

Se han diseñado una serie de paquetes que utilizan tanto los editores como la herramienta de autor.

El paquete '*packLayoutAbsoluto*' (figura 11) aporta un Layout que permite insertar componentes gráficos en una posición determinada con coordenadas (x, y).

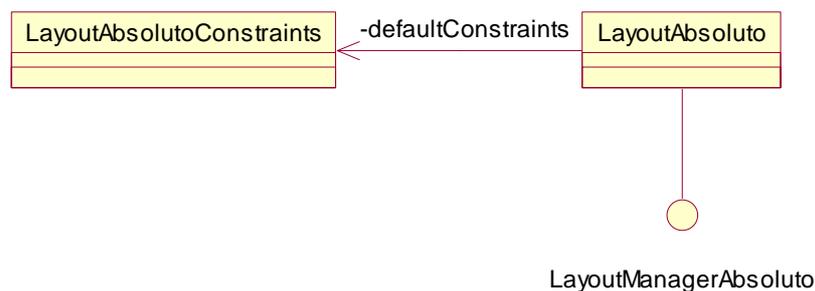


Figura 11: Paquete packLayoutAbsoluto

El paquete '*VentanaEstandar*' (figura 12) aporta el funcionamiento básico de una ventana.

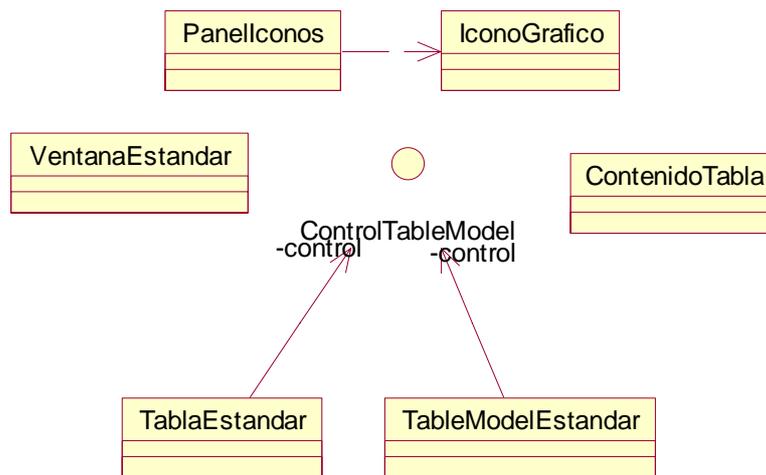


Figura 12: Paquete VentanaEstandar

El paquete '*Concurrencia*' (figura 13) aporta el mecanismo necesario para en trabajo en común de tareas concurrentes.

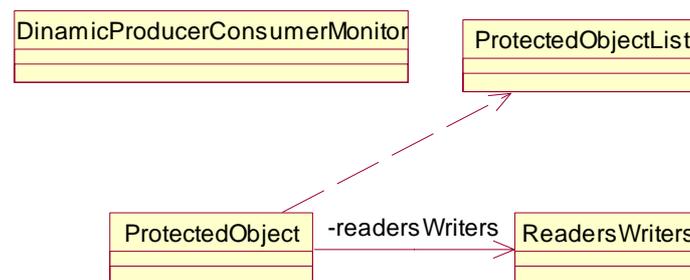


Figura 13: Paquete Concurrencia

3.5. IMPLEMENTACIÓN

En la implementación se ha seguido el paradigma de la programación orientada a objetos. Para ello se ha utilizado el lenguaje de programación Java ya que además de permitir utilizar funcionalidades de multiproceso, comunicaciones a través de Internet, interfaces, etc., hace que la aplicación pueda funcionar en cualquier plataforma. También se ha utilizado JavaPDF que sirve para dotar de capacidades de manejo de archivos PDF a programas escritos en lenguaje Java. Por ello, tanto en el cliente como en el servidor se tiene que tener una máquina con soporte para la ejecución de aplicaciones Java (JRE) así como acceso a la red de Internet o red local para la conexión de los clientes con el servidor.

Como entorno de desarrollo se ha utilizado jGRASP que sirve para facilitar la realización de programas en Java.

A continuación se procede a explicar la implementación de las clases de cada paquete explicados en la sección 3.4.

3.5.1 EDITOR GRÁFICO Y HERRAMIENTA DE AUTOR

Paquete Alumno

La clase '*VentanaAlumno*' (ver figura 14) implementa el interfaz gráfico del editor gráfico. Por tanto, cuando el alumno lo carga, esta clase se encarga de mostrar su ventana.

La clase '*AreaTrabajoJScrollPane*' (ver figura 15) se encarga de implementar el área de trabajo compartida del editor gráfico y la clase '*ElementoGrafico*' (ver figura 16) implementa los elementos que el alumno inserta en dicha área.

La clase '*VentanaAlumnoEventHandler*' (ver figura 17) recoge los eventos que se producen en el editor gráfico. Por tanto, cuando un alumno realiza una operación con el editor (pinchar, cambiar el tamaño de un elemento, etc.), esta clase se encarga de comunicar a la clase '*VentanaAlumno*' el evento que se ha producido.

VentanaAlumno (from Alumno)	
	nombreAlumno : String
	contraseñaAlumno : String
	posxColumna1 : int = 10
	posxColumna2 : int = 5 * 1 + VentanaEstandar.widthIcono * 1 + 10
	posxColumna3 : int = 5 * 2 + VentanaEstandar.widthIcono * 2 + 10
	botonesEjercicio : Vector = new Vector ()
	MOVE_BUTTON : int = 4
	RESICE_BUTTON : int = 5
	DELETE_BUTTON : int = 6
	botonSeleccionado : int
	tiempoFinalizacionEjercicio : int = 5
	botonSeleccionadoString : String
	contadorRegistro : int = 0
	getTiempoFinalizacionEjercicio()
	inhabilitaEjercicio()
	cerrarAplicacion()
	enviarFinalizacionEjercicioMessage()
	setBotonSeleccionado()
	procesaElementoGraficoMessage()
	procesaNuevoElementoGraficoMessage()
	getBotonSeleccionado()
	setBotonSeleccionadoString()
	getBotonSeleccionadoString()
	VentanaAlumno()
	getNombreAlumno()
	removeIconos()
	añadirElementoGrafico()
	eliminarElementoGrafico()
	enviarCambioElementoGrafico()
	enviarNuevoElementoGrafico()
	mostrarEjercicioSeleccionado()
	setRegistroAlumno()
	getRegistroAlumno()
	construirVentanaAlumno()
	setAreaTrabajoJScrollPane()
	getlistaEjerciciosAlumno()
	getenunciado()
	getareaTrabajo()
	getfinEjBoton()
	getpanel()
	getMove()
	getResize()
	getDetele()

Figura 14: Clase VentanaAlumno

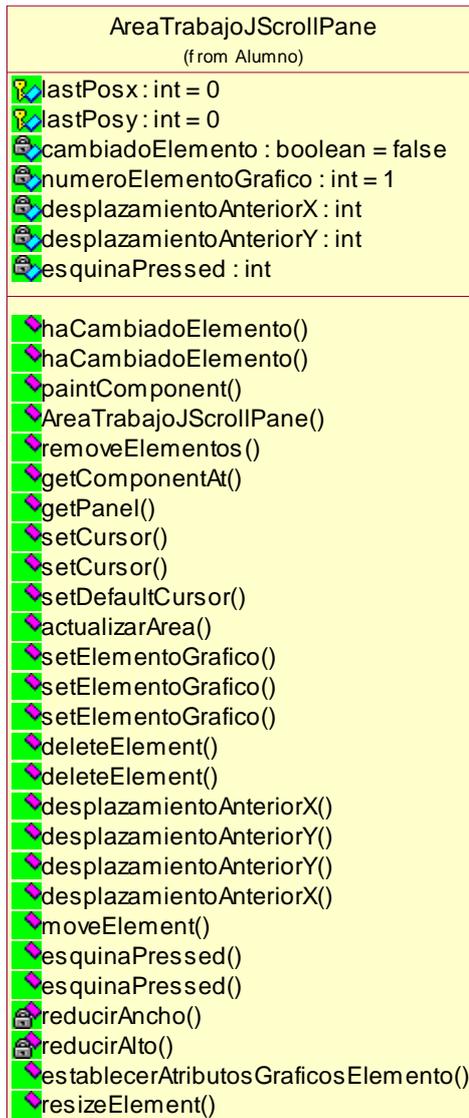


Figura 15: Clase AreaTrabajoJScrollPane

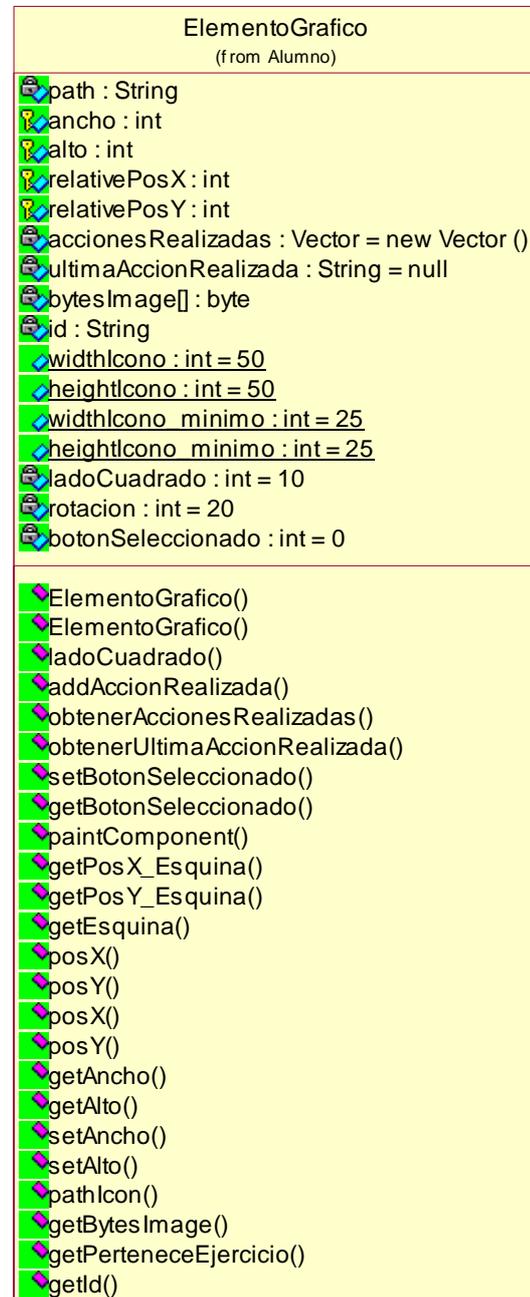


Figura 16: Clase ElementoGrafico

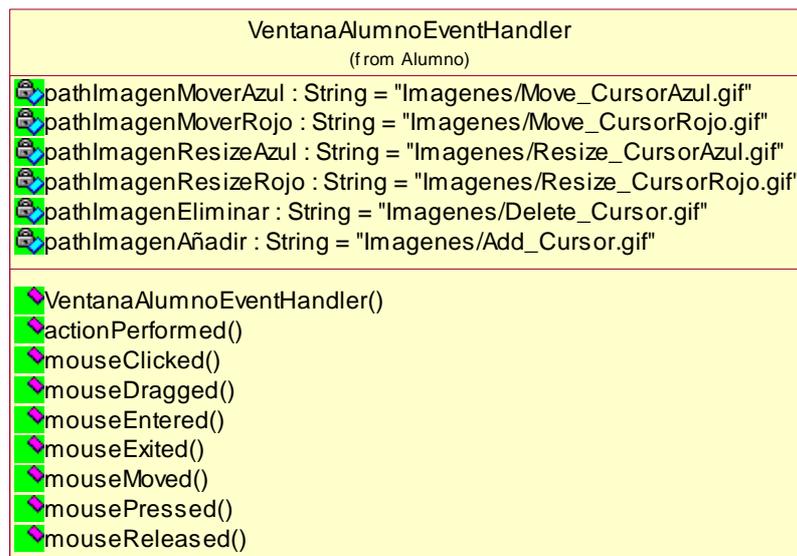


Figura 17: Clase VentanaAlumnoEventHandler

Las clases ‘AlumnoMessageManager’ (ver figura 18), ‘Alumno’ (ver figura 19) y ‘ServerReader’ (ver figura 20) se encargan de implementar la comunicación entre el editor gráfico y el servidor.

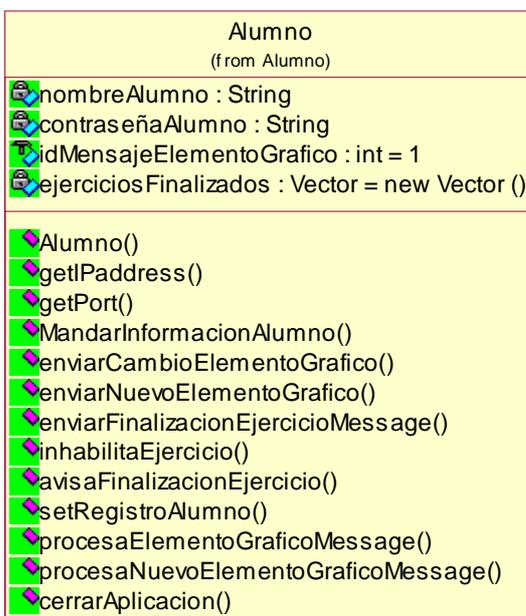


Figura 19: Clase Alumno

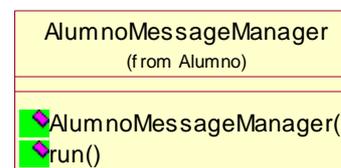


Figura 18: Clase AlumnoMessageManager

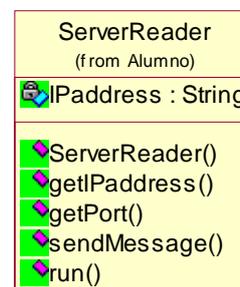


Figura 20: Clase ServerReader

Paquete BDCurso

La clase 'BDCurso' (ver figura 21) implementa la base de datos que utiliza el editor gráfico, la herramienta de autor y el servidor. Esta clase hace uso de las clases 'Ejercicio' (ver figura 22), 'Tema' (ver figura 23) y 'RegistroAlumno' (ver figura 24).

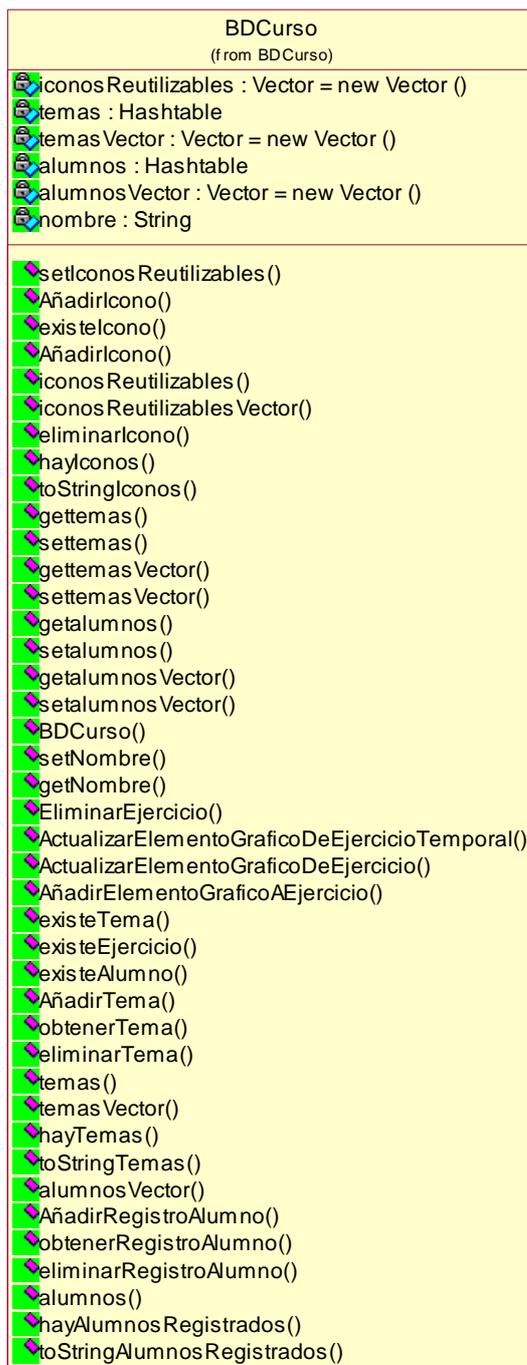


Figura 21: Clase BDCurso

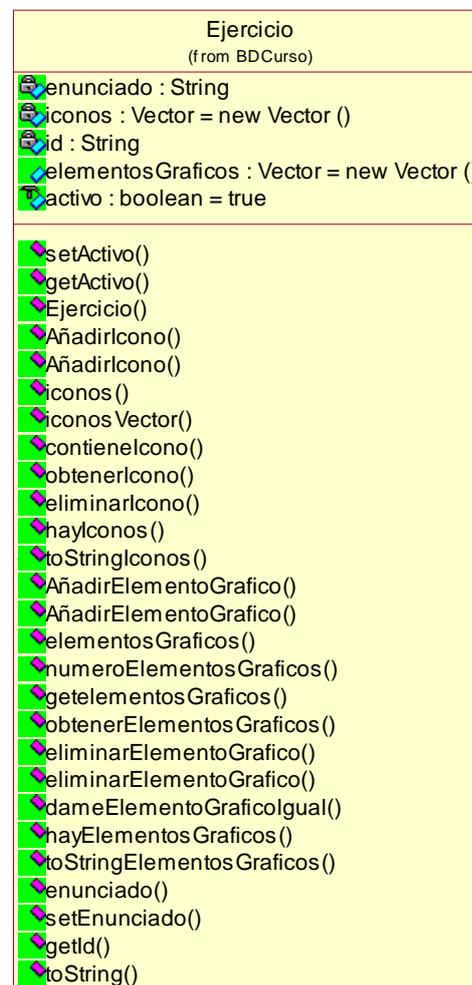


Figura 22: Clase Ejercicio

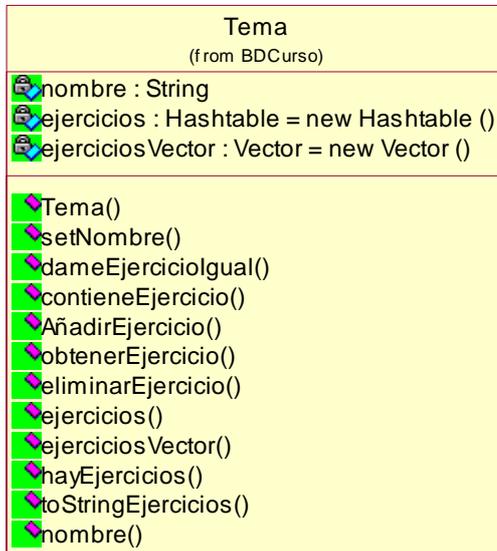


Figura 23: Clase Tema

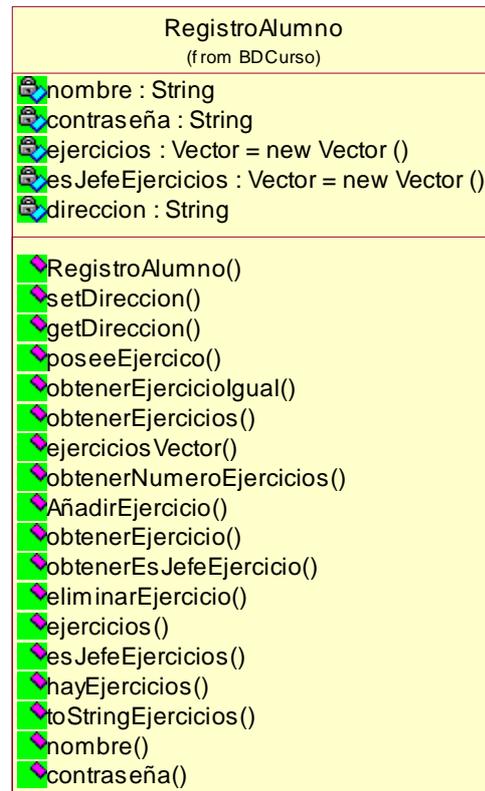


Figura 24: Clase RegistroAlumno

Paquete InformacionAlumnos

La clase 'UserInfo' (ver figura 25) implementa la información del alumno que utiliza el editor gráfico y el servidor. La clase 'UserInfoList' (ver figura 26) sirve para gestionar un conjunto de 'UserInfo'.



Figura 25: Clase UserInfo

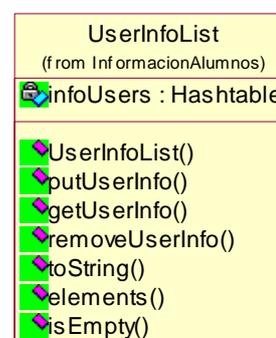


Figura 26: Clase UserInfoList

Paquete Mensajes

Las clases '*MensajeList*' (ver figura 27) y '*MensajeManajer*' (ver figura 28) se encargan de gestionar la entrada y salida de mensajes entre el editor gráfico y el servidor así como entre la herramienta de autor y el servidor.



Figura 27: Clase MessageList



Figura 28: Clase MessageManager

La clase '*Message*' (ver figura 29) implementa un mensaje genérico.

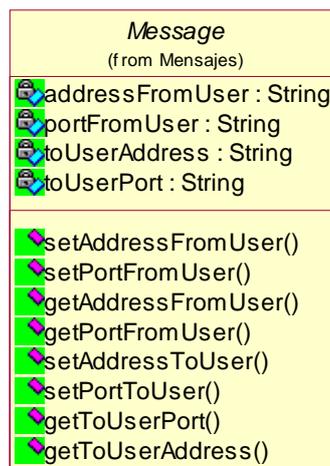


Figura 29: Clase Message

El resto de las clases de este paquete implementan cada uno de los mensajes que se envían (ver figuras 30 – 38).

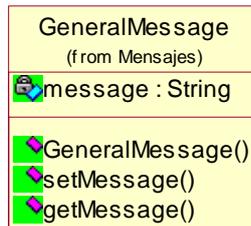


Figura 30: Clase GeneralMessage

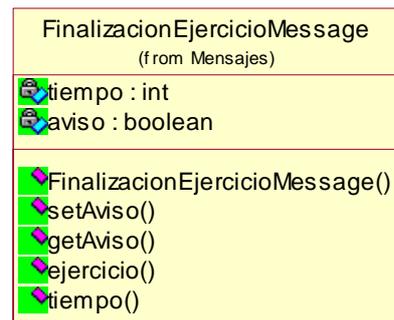


Figura 31: Clase FinalizacionEjercicioMessage

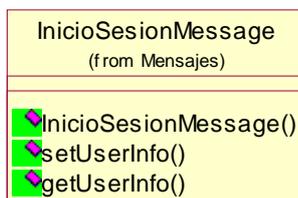


Figura 32: Clase InicioSesionMessage

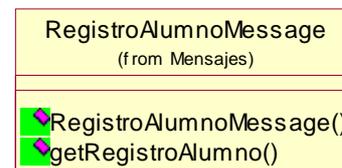


Figura 33: Clase RegistroAlumnoMessage

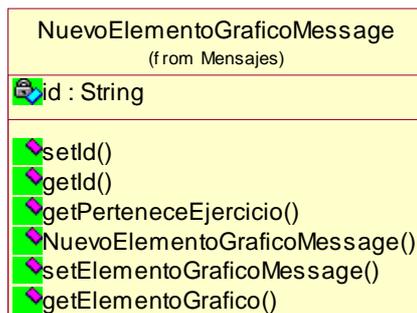


Figura 34: Clase NuevoElementoGraficoMessage

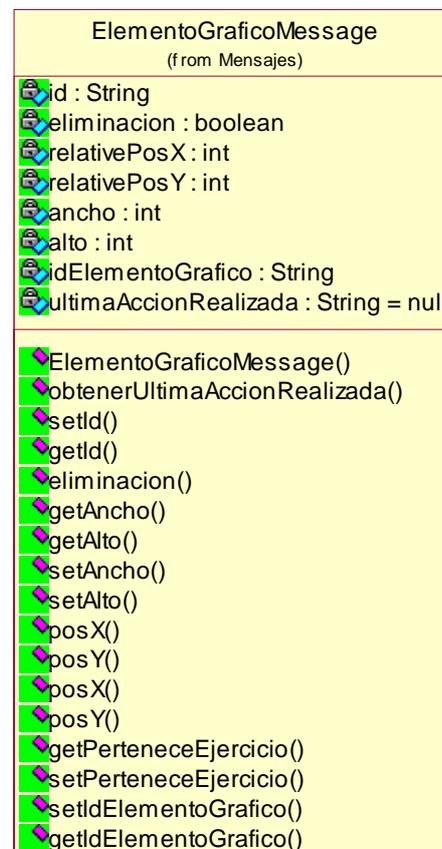


Figura 35: Clase ElementoGraficoMessage

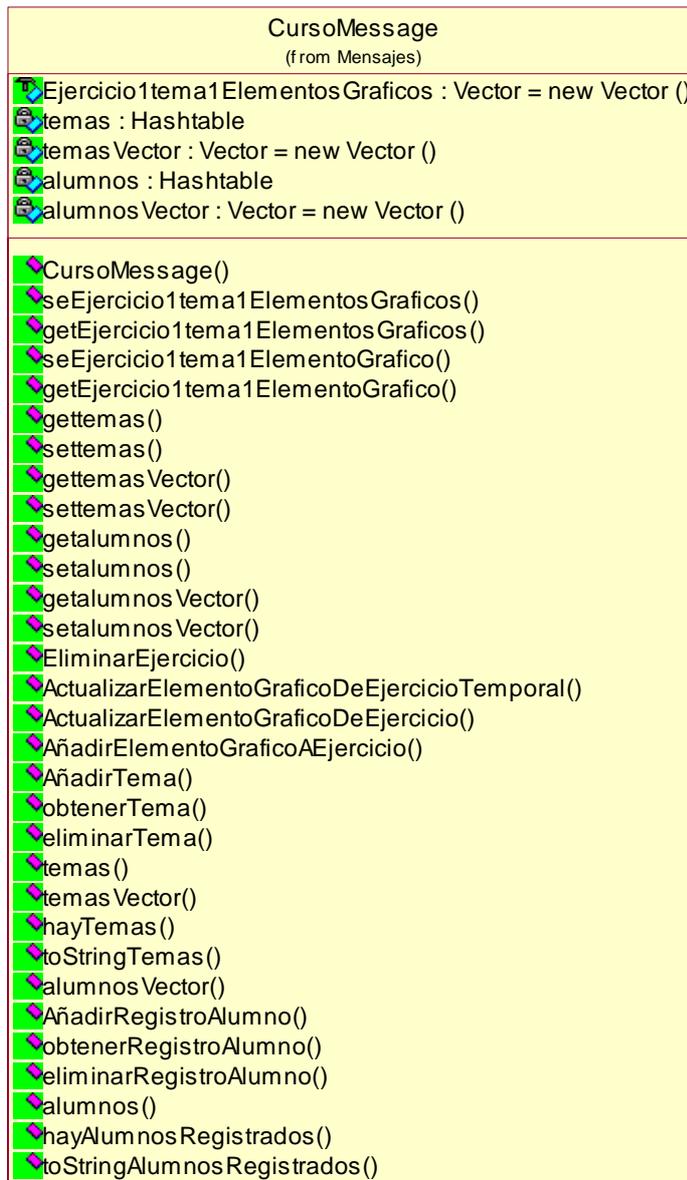


Figura 37: Clase CursoMessage

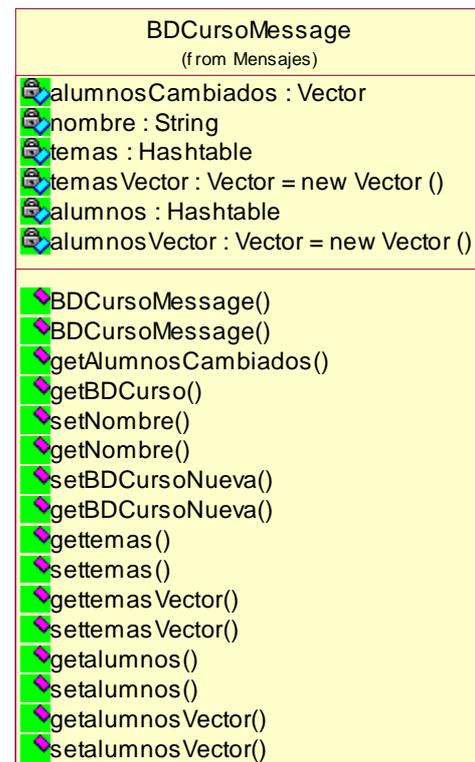


Figura 36: Clase BDCursoMessage

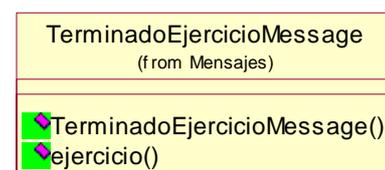


Figura 38: Clase TerminadoEjercicioMessage

Paquete Profesor

La clase 'Profesor' (ver figura 39) se encarga de implementar la comunicación entre la herramienta de autor y el servidor.



La clase '*VentanaProfesor*' (ver figura 40) implementa el interfaz gráfico de la herramienta de autor, por tanto, cuando el alumno la carga esta clase se encarga de mostrar su ventana.

La clase '*ControlTableModelEstandar_Temas*' (ver figura 41) se encarga de implementar la tablas en la que el profesor añade y elimina temas.

VentanaProfesor (from Profesor)
<ul style="list-style-type: none"> alumnosCambiados : Vector = new Vector () contador : int cambioVentana : boolean = true modificarCurso : boolean = false modificadoContador : int = 0 mensajesGuardados : Vector = new Vector ()
<ul style="list-style-type: none"> crearPDF() getVentanaProfesorEventHandler() getBDCurso() getEjercicio() getalumnosCambiados() setalumnosCambiados_cambioEjercicio() setalumnosCambiados_cambioTema() setalumnosCambiados_cambioAlumno() enviarFinalizacionEjercicioMessage() VentanaProfesor() setcambioVentana() getcambioVentana() setModificarCurso() getModificarCurso() procesaElementoGraficoMessage() procesaNuevoElementoGraficoMessage() guardarMensaje() setBDCurso() enviarPeticonBDCurso() ActualizarBD() GuardarCurso() getAreaTrabajo() setAreaTrabajoScrollPane() getCrearPdf() getFinalizarEj() getAñadirIcono() getMenuTemas() getpanelIconosProfesor() getTablaEjercicios() getControlTableModelEstandar_Ejercicios() mostrarIconosEjercicioSeleccionadoEnModificar() setPanelIconosProfesor() mostrarIconosEjercicioSeleccionado() añadirEjercicios() getseleccionEjercicios() getMenuCrear_Alumnos() getTablaAlumnos() getControlTableModelEstandar_Alumnos() mostrarListaEjerciciosSeleccionados() añadirAlumnos() getIrejerciciosTema() getMenuCrear_Temas() getTablaTemas() getControlTableModelEstandar_Temas() añadirTemas() getTemas() getAlumnos() getCrear() getMenuMPP() RecibiendoCursoActual() CrearModificarCurso() getcrearCurso() gestionCursoActual() MPP() GetComponentAt() setCursor() setCursor() setDefaultCursor() actualizarArea() deleteElement() deleteElement() establecerAtributosComponente()

Figura 40: Clase VentanaProfesor

Profesor (from Profesor)
<ul style="list-style-type: none"> nombreProfesor : String contraseñaProfesor : String idMensajeElementoGrafico : int = 1 ejerciciosFinalizados : Vector = new Vector ()
<ul style="list-style-type: none"> Profesor() getIpAddress() getPort() enviarBDCurso() enviarFinalizacionEjercicioMessage() inhabilitaEjercicio() avisaFinalizacionEjercicio() enviarPeticonBDCurso() enviarInicioSesionProfesor() processMessage() procesaElementoGraficoMessage() procesaNuevoElementoGraficoMessage()

Figura 39: Clase Profesor

ControlTableModelEstandar_Temas (from Profesor)
<ul style="list-style-type: none"> tipoContenido : String = null numero : int = 0 tablasEjercicios : Vector = new Vector ()
<ul style="list-style-type: none"> existeTema() ControlTableModelEstandar_Temas() getNombreTemaSeleccionado() getNombreTema() getNumeroTemas() getTablaEjercicios() getControlTableModelEstandar_Ejercicios() getSelectedControlTableModelEstandar_Ejercicios() getSelectedTablaEjercicios() getTablaEjerciciosSize() getTablasEjercicios() seleccionarFila() modificarFila() añadirFila() añadirFila() añadirFila() eliminarFila() botonAñadirFila() botonEliminarFila() setTabla()

Figura 41: Clase ControlTableModelEstandar_Temas

La clase 'ControlTableModelEstandar_Alumnos' (ver figura 42) se encarga de implementar la tablas en la que el profesor añade y elimina alumnos.

La clase 'ControlTableModelEstandar_Ejercicios' (ver figura 43) se encarga de implementar la tablas en la que el profesor añade, modifica y elimina actividades gráficas.

ControlTableModelEstandar_Alumnos (from Profesor)
<ul style="list-style-type: none"> tipoContenido : String = null numero : int = 0 listasEjerciciosSeleccionados : Vector = new Vector () contAlumno : int = 1
<ul style="list-style-type: none"> ControlTableModelEstandar_Alumnos() existeAlumno() getNombreAlumno() getSelectedNombreAlumno() getSelectedContraseñaAlumno() getContraseñaAlumno() getNumeroAlumnos() inicializarListaEjerciciosSeleccionados() getListaEjerciciosAlumnoSeleccionado() getListaEjercicios() seleccionarFila() modificarFila() seleccionarEjercicios() quitarEjercicio() refrescarListaEjercicios() añadirEjerciciosLista() añadirEjerciciosLista() añadirFila() añadirFila() añadirFila() añadirFila() eliminarFila() botonAñadirFila() botonEliminarFila() setTabla()

ControlTableModelEstandar_Ejercicios (from Profesor)
<ul style="list-style-type: none"> tipoContenido : String = null numero : int = 0 panelesIconos : Vector = new Vector () areasDeTrabajo : Vector = new Vector () inicializado : boolean = false
<ul style="list-style-type: none"> ControlTableModelEstandar_Ejercicios() existeEjercicio() procesaElementoGraficoMessage() procesaNuevoElementoGraficoMessage() getSelectedNombreEjercicio() getNombreEjercicio() getEnunciadoEjercicio() getSelectedEnunciadoEjercicio() getNumeroEjercicios() getPanelIconosProfesor() getSelectedPanelIconosProfesor() getareaDeTrabajo() getSelectedareaDeTrabajo() modificarFila() seleccionarFila() añadirFila() setInicializado() inicializado() añadirFila() añadirFila() eliminarFila() botonAñadirFila() botonEliminarFila() setTabla()

Figura 42: Clase ControlTableModelEstandar_Alumnos

Figura 43: Clase ControlTableModelEstandar_Ejercicios

La clase 'PanellconosProfesor' (ver figura 44) se encarga de implementar una tabla en la que cada elemento es un icono. La herramienta de autor lo utiliza para permitir al profesor añadir iconos a las actividades gráficas o mostrarle los que tiene.

La clase 'ElementoLista' (ver figura 45) se encarga de implementar la asignación de una actividad gráfica a un alumno.

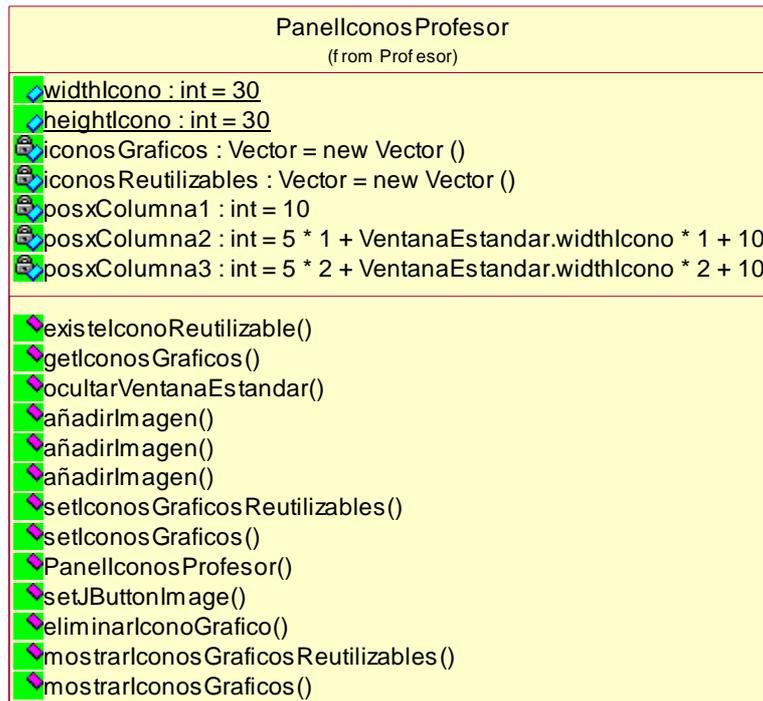


Figura 44: Clase PanellconosProfesor

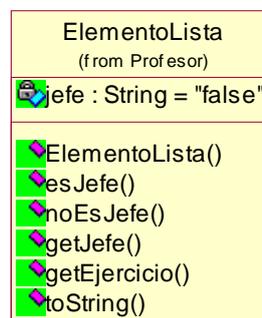


Figura 45: Clase ElementoLista

Paquete ServidorAlumnoProfesor

La clase 'ListClientReader' (ver figura 46) implementa el envío de mensajes unicast y broadcast a los editores gráficos y unicast a la herramienta de autor.

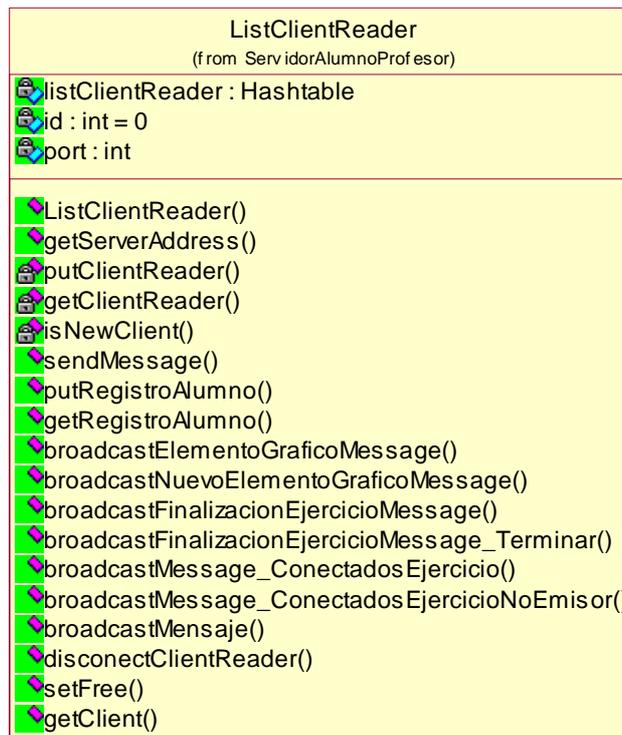


Figura 46: Clase ListClientReader

La clase 'GestorEnvioEjercicioTerminado' (ver figura 47) implementa el control de la finalización de una actividad gráfica. Cuando el periodo de finalización de una actividad gráfica termina, manda un mensaje broadcast a todos los editores gráficos de los alumnos que la poseen.

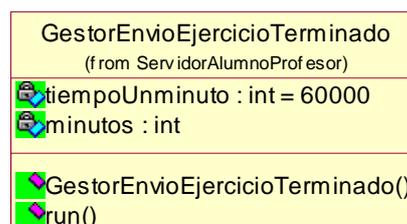


Figura 47: Clase GestorEnvioEjercicioTerminado

Las clases 'Server' (ver figura 48) y 'ServerMessageManager' (ver figura 49) se encargan de implementar la comunicación con el editor gráfico y la herramienta de autor.



Figura 48: Clase Server

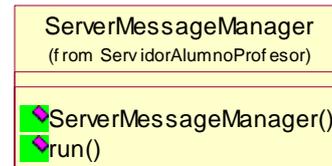


Figura 49: Clase ServerMessageManager

3.5.2 EDITOR TEXTUAL

El editor textual posee gran variedad de clases ya diseñadas en el editor gráfico, por ello en este punto, se limitará a detallar las nuevas.

Paquete Alumno

La clase 'VentanaAlumno' (ver figura 50) implementa el interfaz gráfico del editor textual, por tanto, cuando el alumno lo carga esta clase se encarga de mostrar su ventana.

Las clase 'Alumno' (ver figura 51), se encarga de implementar la comunicación entre el editor textual y el servidor.

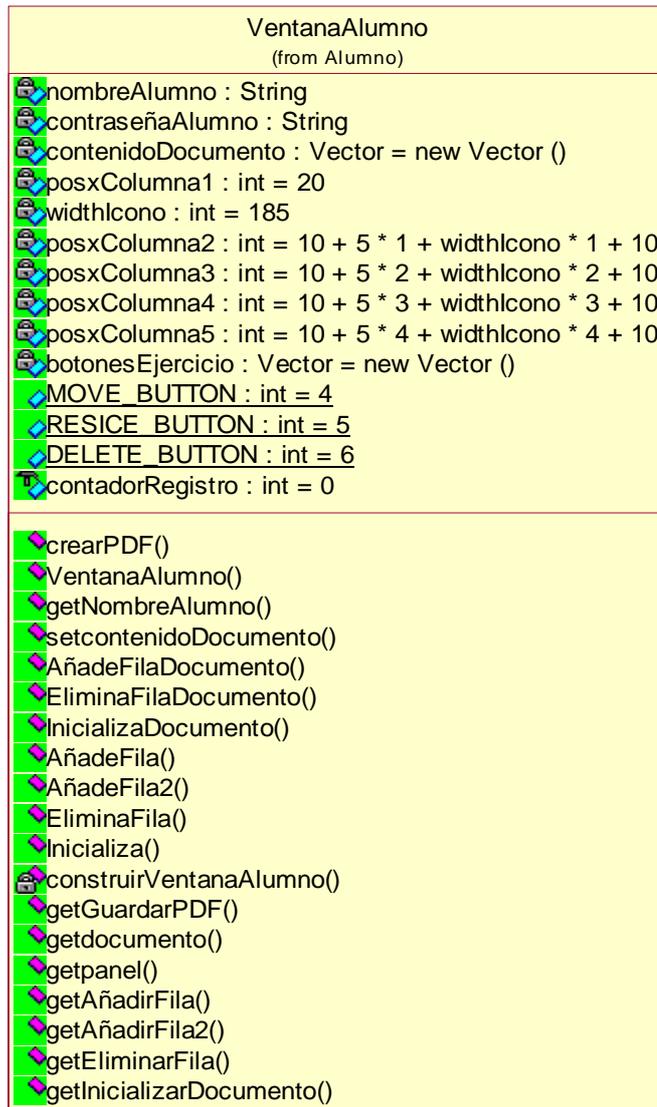


Figura 50: Clase VentanaAlumno del editor textual

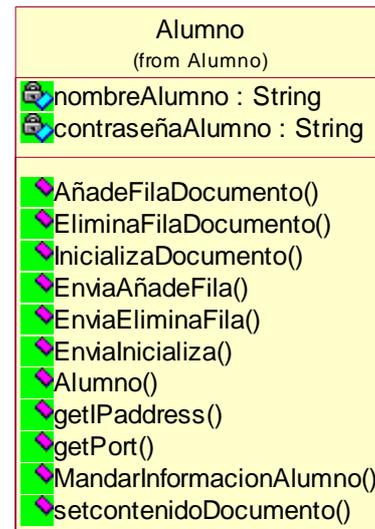


Figura 51: Clase Alumno del editor textual

Paquete Mensajes

Las clases '*MensajeList*' (ver figura 27) y '*MensajeManajer*' (ver figura 28) se encargan de gestionar la entrada y salida de mensajes entre el editor textual y el servidor.

La clase '*Message*' (ver figura 29) implementa un mensaje genérico.

El resto de las clases de este paquete implementan cada uno de los mensajes que se envían (ver figuras 52 y 53).

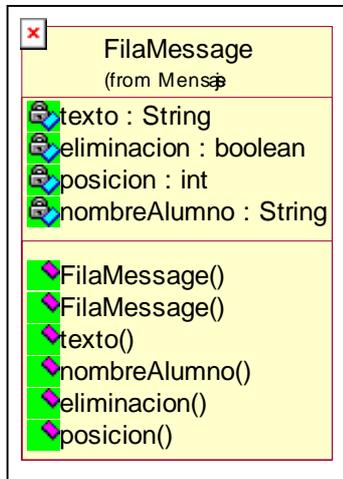


Figura 52: Clase FilaMessage

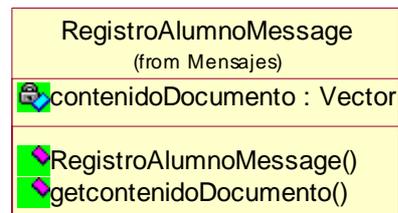


Figura 53: Clase RegistroAlumnoMessage

Paquete ServidorAlumnoProfesor

Las clase 'Server' (ver figura 54) se encarga de implementar la comunicación con el editor textual.

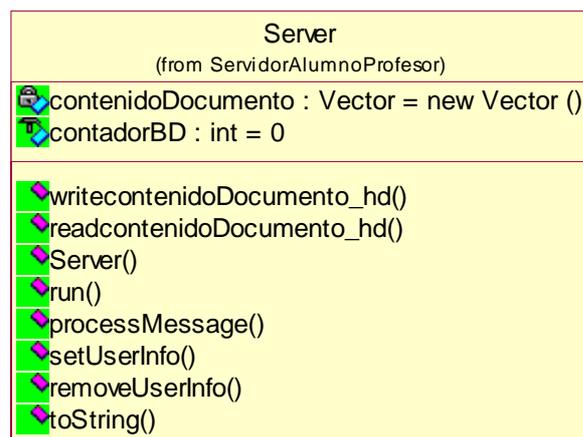


Figura 54: Clase Server

3.5.3 PAQUETES COMUNES

Paquete packLayoutAbsoluto

Las clases 'LayoutAbsoluto' (figura 55) 'LayoutAbsolutoConstraints' (figura 56) y el interfaz 'LayoutManagerAbsoluto' (figura 57) implementan un Layout que permite insertar componentes gráficos en una posición (x, y).

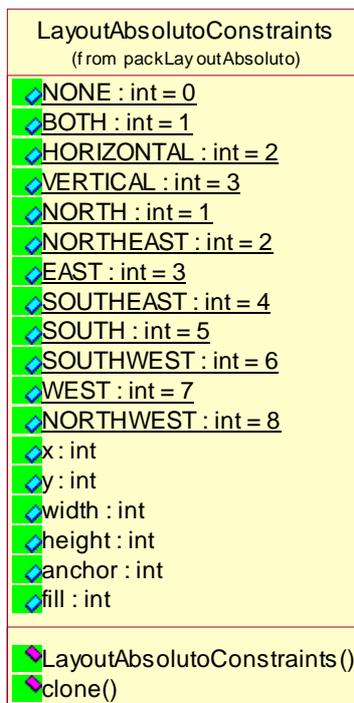


Figura 56: Clase LayoutAbsolutoConstraints



Figura 55: Clase LayoutAbsoluto

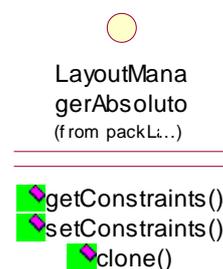


Figura 57: Interfaz LayoutManagerAbsoluto

Paquete VentanaEstandar

La clase 'VentanaEstandar' (ver figura 58) implementa un interfaz gráfico estándar.



Figura 58: Clase VentanaEstandar

Las clases 'TablaEstandar' (ver figura 59), 'ContenidoTabla' (ver figura 60) y 'TableModelEstandar' (ver figura 61) y el interfaz 'ControlTableModel' (ver figura 62) implementan una tabla gráfica. Es una tabla genérica, por tanto no tienen implementado que hacer cuando se añade o elimina un elemento. Esta tabla se puede insertar en 'VentanaEstandar' (ver figura 63).

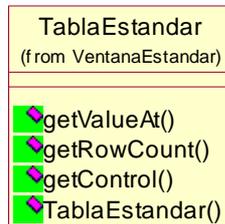


Figura 59: Clase TablaEstandar

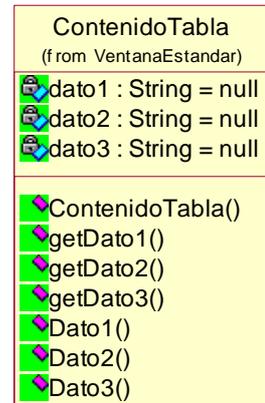


Figura 60: Clase ContenidoTabla

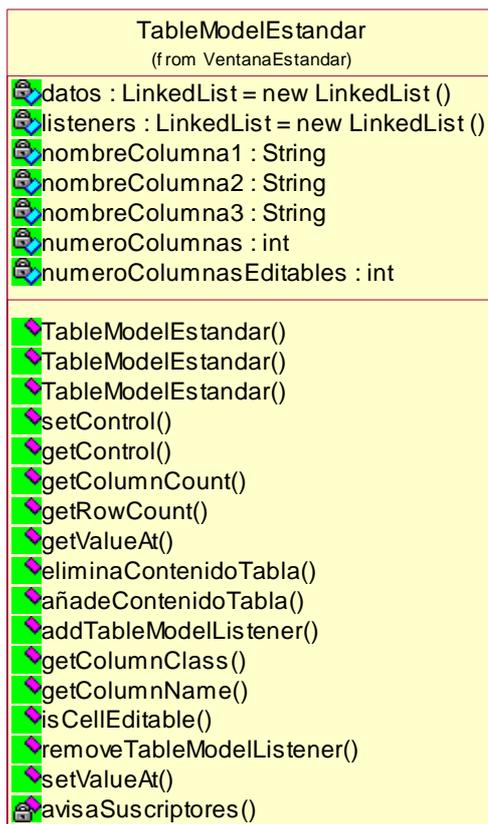


Figura 61: Clase TableModelEstandar

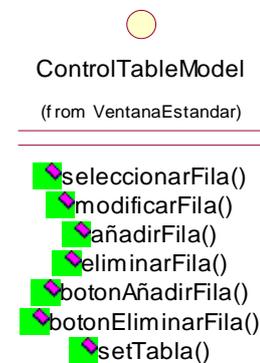


Figura 62: Interfaz ControlTableModel

La clase '*PanelIconos*' (ver figura 63) se encarga de implementar una tabla en la que cada elemento es un icono.

La clase 'IconoGrafico' (ver figura 64) se encarga de implementar un icono gráfico a partir de los bytes de su imagen.

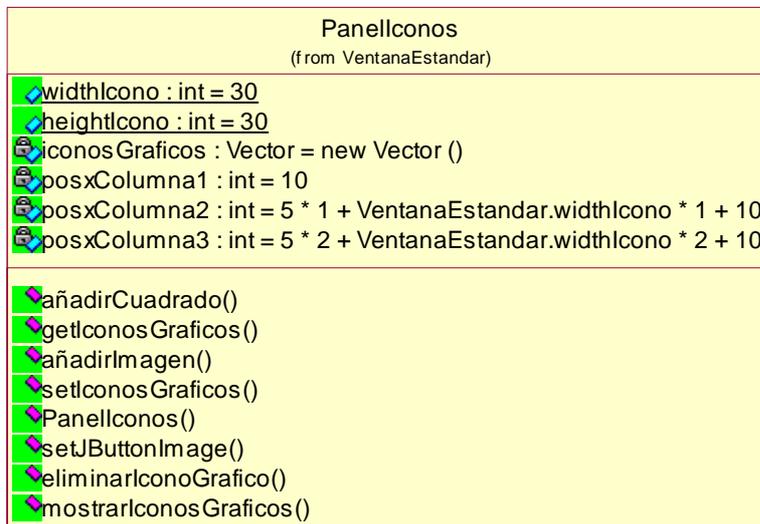


Figura 63: Clase PanellIconos

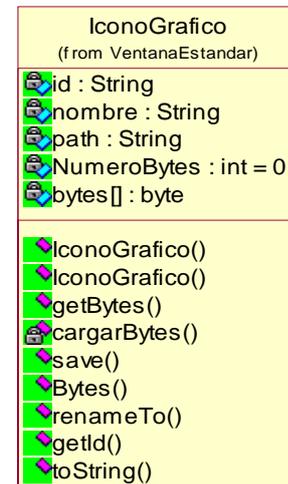


Figura 64: Clase IconoGrafico

Paquete Concurrencia

La clase 'DinamicProducerConsumerMonitor' (ver figura 65) implementa el problema del productor consumidor.

Las clases 'ProtectedObject' (ver figura 66) 'ReadersWriters' (ver figura 67) implementan el problema del lector / escritor en concurrencia.

La clase 'ProtectedObjectList' (ver figura 68) implementa una lista de 'ProtectedObject'.

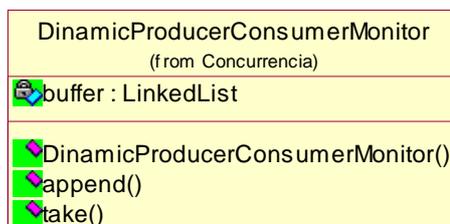


Figura 65: Clase DinamicProducerConsumerMonitor

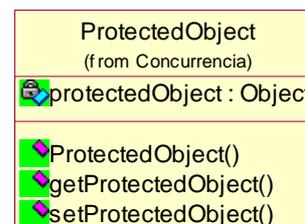


Figura 66: Clase ProtectedObject

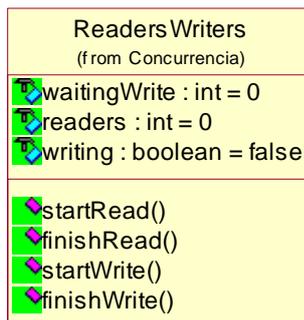


Figura 67: Clase ReadersWriters

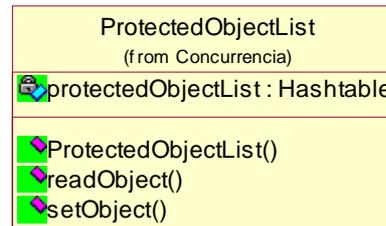


Figura 68: Clase ProtectedObjectList

3.6. FUNCIONAMIENTO Y PRUEBAS

Para comprobar que la aplicación cumple los requisitos especificados en el apartado 3.3, se han realizado una serie de pruebas que lo verifican. En este apartado se detallarán dichas pruebas y a su vez el funcionamiento de la aplicación.

3.6.1 HERRAMIENTA DE AUTOR

Para que funcione correctamente la herramienta de autor es necesario cargar antes el servidor del alumno y el profesor. Una vez cargado el servidor, el profesor carga la herramienta de autor y se le muestra la pantalla principal donde se le ofrece dos posibilidades: 'Crear un nuevo curso' y 'Gestionar el curso actual' (ver figura 69).

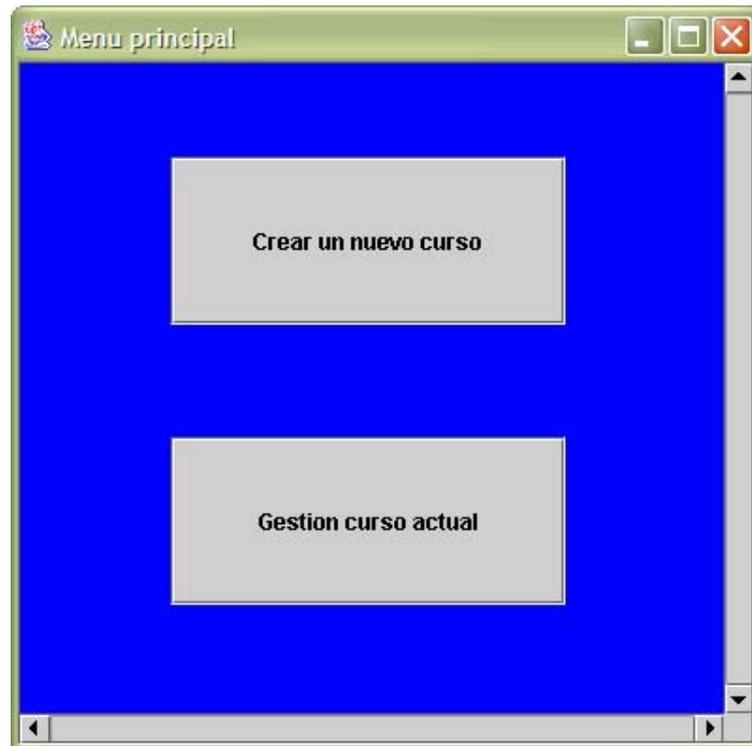


Figura 69: Menú principal herramienta de autor

Estas dos opciones son similares en su funcionamiento y en las pantallas presentadas. La primera sirve para que los profesores creen un nuevo curso de tal manera que puedan especificar los temas en los que se divide el curso, las actividades gráficas asociadas a cada uno de los temas, definición de los alumnos que realizarán el curso, las actividades gráficas que realizarán cada uno de ellos, etc. La segunda opción sirve para modificar las opciones de un curso que previamente se ha creado.

Si el profesor de un curso decide crear uno nuevo, tras pinchar en la opción '*Crear un nuevo curso*' del menú principal podrá crear los temas que contiene el nuevo curso y especificar los alumnos que lo realizarán (ver figura 70). La opción '*Crear el curso*' enviará el curso al servidor para que los alumnos puedan acceder a él desde el editor gráfico.

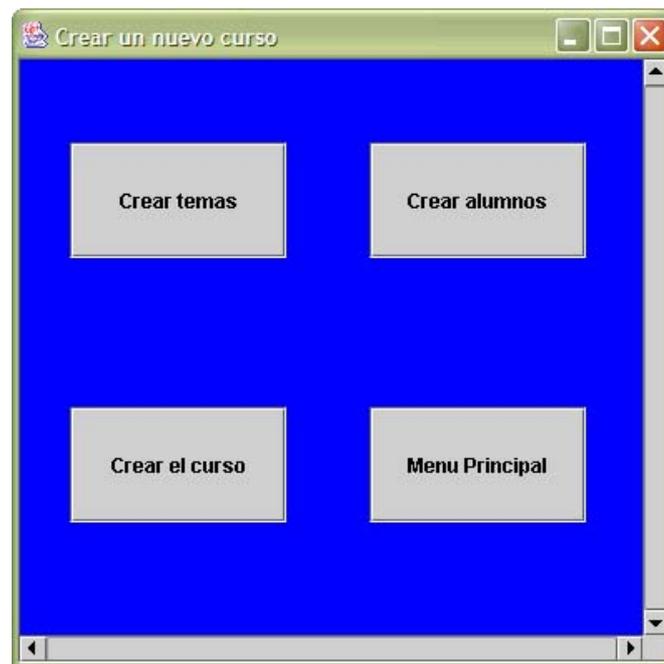


Figura 70: Creación de un nuevo curso

Por el contrario, si decide gestionar un curso ya existente, podrá modificar los temas y los alumnos del curso. Como se puede ver en la figura 71, la pantalla generada para la gestión del curso es análoga a la de creación de un nuevo curso. La opción de la creación del curso es sustituida por la de guardar el curso que además actualiza los editores gráficos de los alumnos que se vean afectados por los cambios y que estén trabajando en ese mismo momento. También comunica la finalización de la sesión de trabajo a los alumnos que hayan sido eliminados de la aplicación.

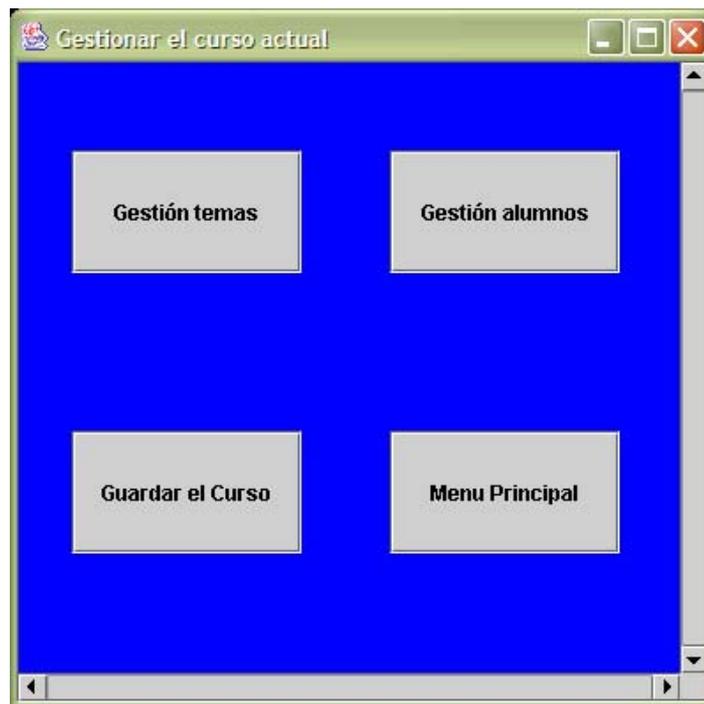


Figura 71: Gestión del curso actual

Al ser tanto el funcionamiento parecido como las pantallas en las dos opciones presentadas al profesor en la pantalla principal, se mostrarán únicamente las pruebas realizadas para la creación de un nuevo curso.

Prueba 1: Creación de los temas y las actividades gráficas del curso

Cuando un profesor quiera crear temas tendrá que pinchar en el botón 'Crear Temas' de la pantalla de creación del curso (ver figura 70). También podrá eliminar los temas ya insertados. Tras pulsar sobre el botón de 'Crear Temas', aparecerá una nueva ventana análoga a la mostrada en la figura 72 en la que podrá realizar la operación deseada.

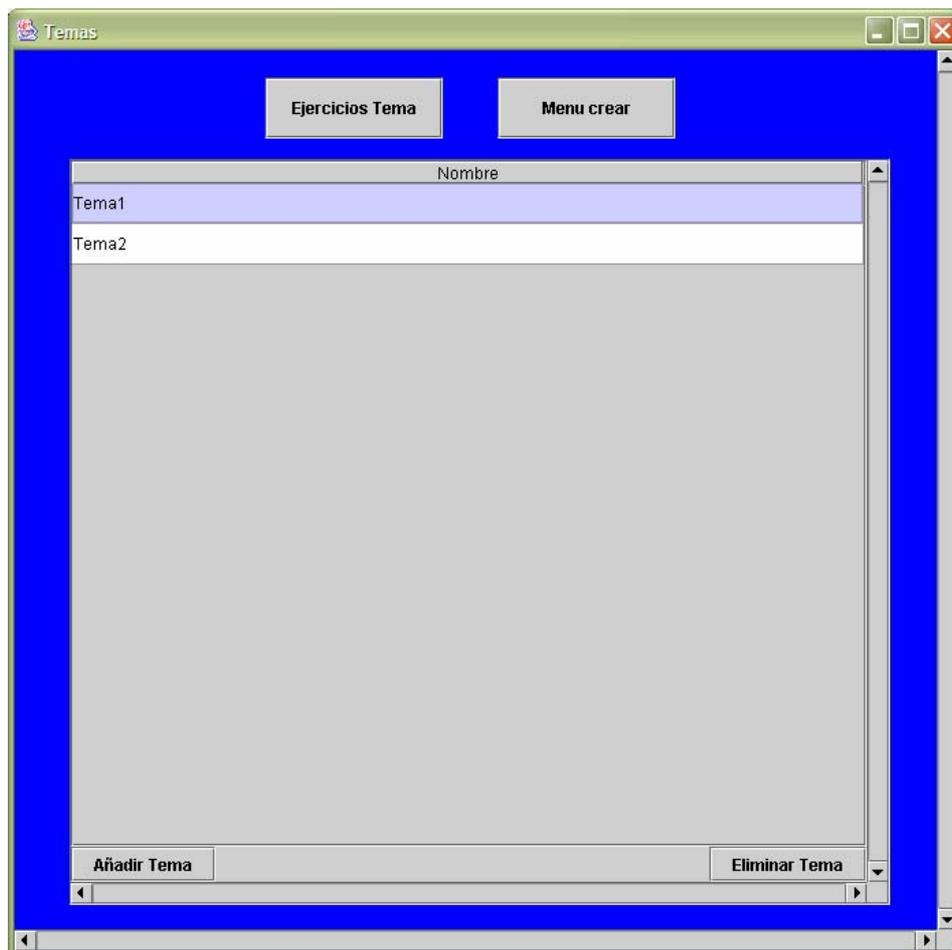


Figura 72: Gestión de temas

Desde esta ventana, el profesor del curso podrá crear un nuevo tema pinchando sobre el botón de '*Añadir Tema*' e introduciendo el nombre del mismo, mientras que para la eliminación de un tema creado tendrá que seleccionar el tema que desea borrar del listado que aparece en pantalla y pinchar en el botón '*Eliminar Tema*'. Se han probado las funcionalidades anteriormente descritas creando los temas 1, 2 y 3 y eliminando posteriormente el último tema (ver figura 72).

Para volver al menú de crear (ver figura 70), el profesor deberá pulsar sobre el botón '*Menu Crear*' (ver figura 72).

Para añadir actividades gráficas a un determinado tema, el profesor tendrá que seleccionar el tema y pinchar en el botón '*Ejercicios Tema*'. También podrá modificar o eliminar las actividades gráficas que añada. Tras esto, aparecerá una nueva ventana en la que podrá efectuar la operación deseada (ver figura 73).

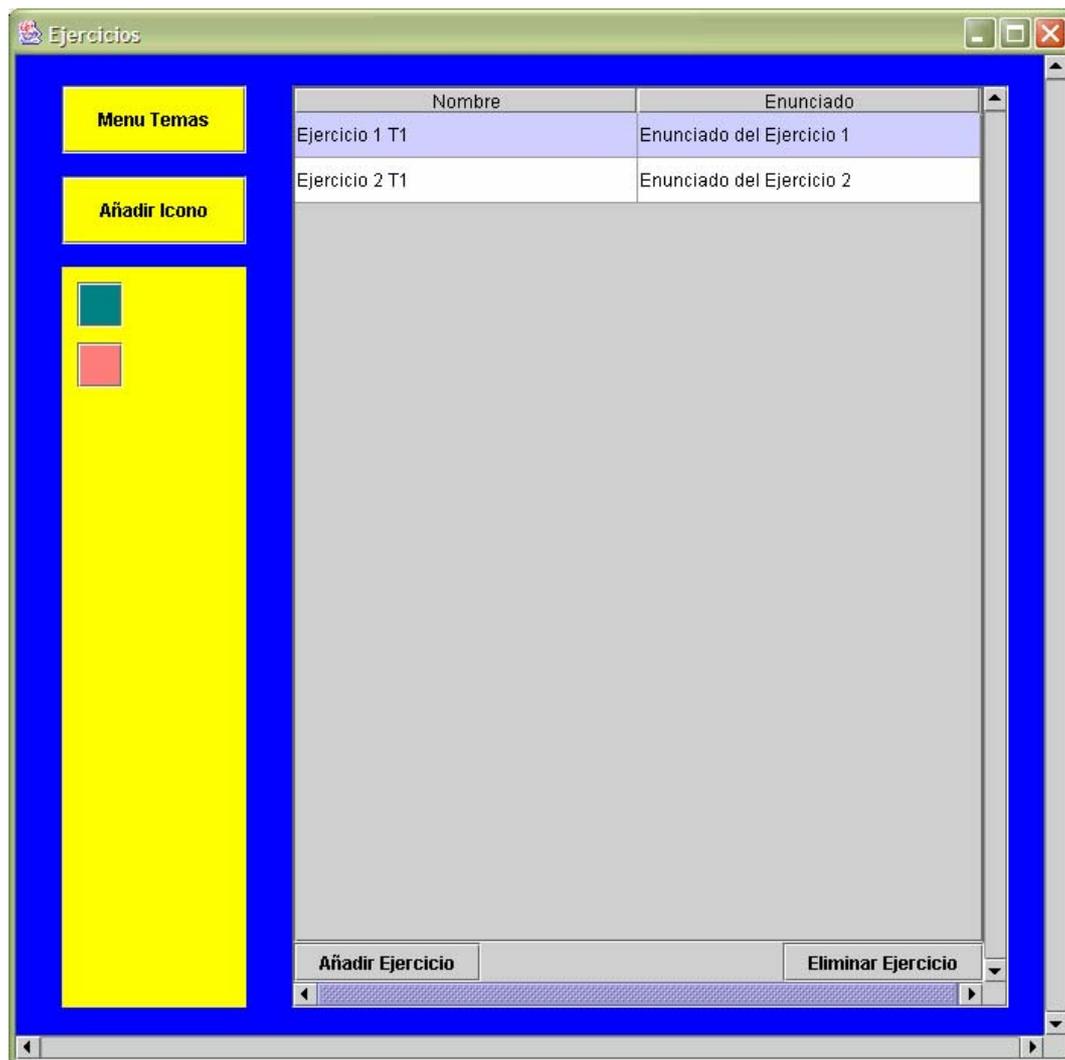


Figura 73: Gestión de actividades gráficas

Si quiere añadir una nueva actividad desde esta ventana, tendrá que pinchar en el botón de 'Añadir Ejercicio', introducir el nombre de la actividad gráfica y su enunciado. Si se desea modificar el enunciado se tendrá que pulsar sobre él y modificar el texto del mismo. Además se deberán introducir los iconos asociados, seleccionando previamente la actividad gráfica y pinchando sobre el botón 'Añadir Icono'. Tras realizar esta operación, aparecerá una nueva ventana desde la que se podrán añadir los iconos (ver figura 74). Estos iconos pueden elegirse de los que ya están almacenados en la aplicación (iconos reutilizables) o bien suministrándolos pulsando sobre el botón 'Cargar imagen de fichero'. Al pulsar sobre este último botón se mostrará un cuadro de diálogo que permitirá buscar por el sistema de ficheros la imagen del icono que se desea insertar.

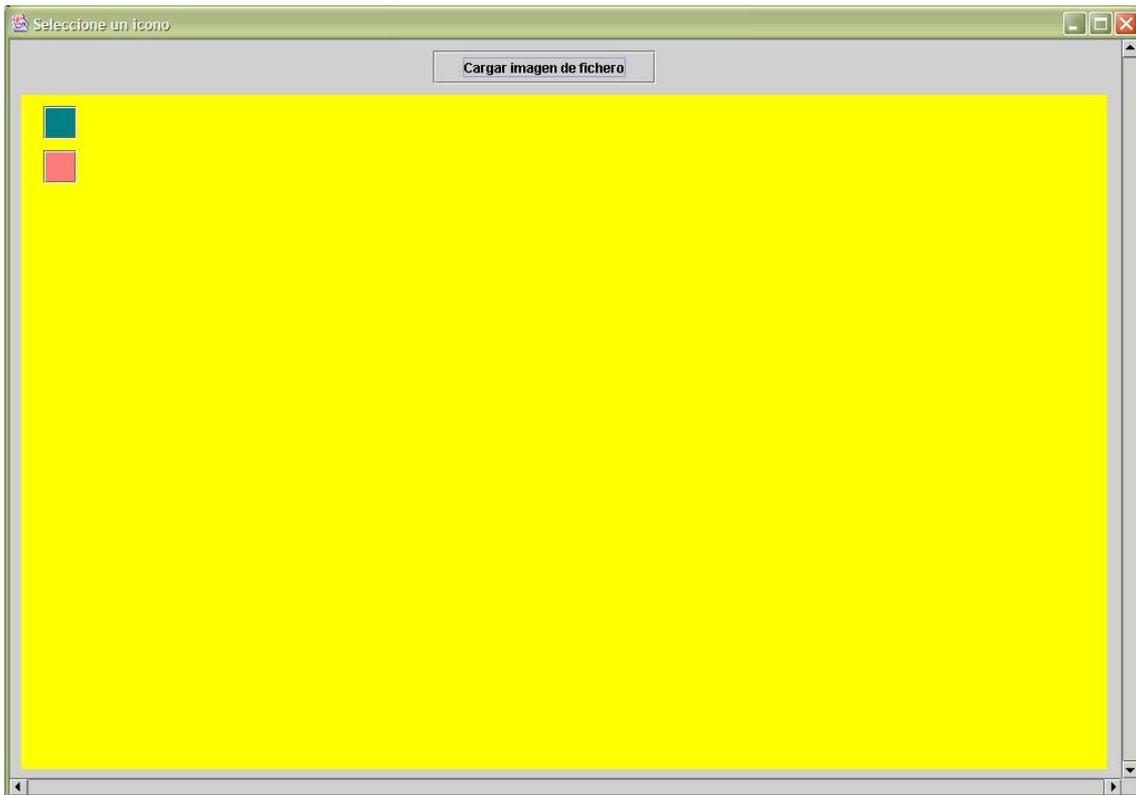


Figura 74: Inserción de iconos

Después de añadir el icono la aplicación mostrará la ventana de Inserción de actividades gráficas (ver figura 73). En la columna de la parte izquierda de esta ventana se reflejará el icono añadido. Para eliminarlo bastará con pinchar en él.

Para eliminar una actividad gráfica añadida, el profesor deberá seleccionar la actividad gráfica a borrar y pulsar sobre el botón '*Eliminar Ejercicio*'.

Se ha probado añadir tres actividades gráficas a los dos temas creados anteriormente y posteriormente se ha procedido a la eliminación de la última. Se han insertado los iconos de las actividades gráficas y eliminado algunos de ellos después de su inserción. También se ha modificado el enunciado de una de las actividades gráficas. En la figura 73 se puede ver la ventana presentada después de realizar estas operaciones sobre el ejercicio 1 del tema 1.

Para volver a la ventana de inserción de temas (ver figura 72), el profesor deberá pulsar sobre el botón '*Menu Temas*' (ver figura 73). Desde allí podrá seleccionar otro tema al que añadirle actividades gráficas.

Prueba 2: Creación de los alumnos y grupos de trabajo de las actividades gráficas

Cuando un profesor quiera crear alumnos y grupos de trabajo tendrá que pinchar en el botón 'Crear Alumnos' de la pantalla de creación del curso (ver figura 70). También podrá eliminar los alumnos que cree así como modificar o eliminar los grupos de trabajo que cree. A continuación, aparecerá una nueva ventana en la que podrá realizar la operación deseada (ver figura 75).

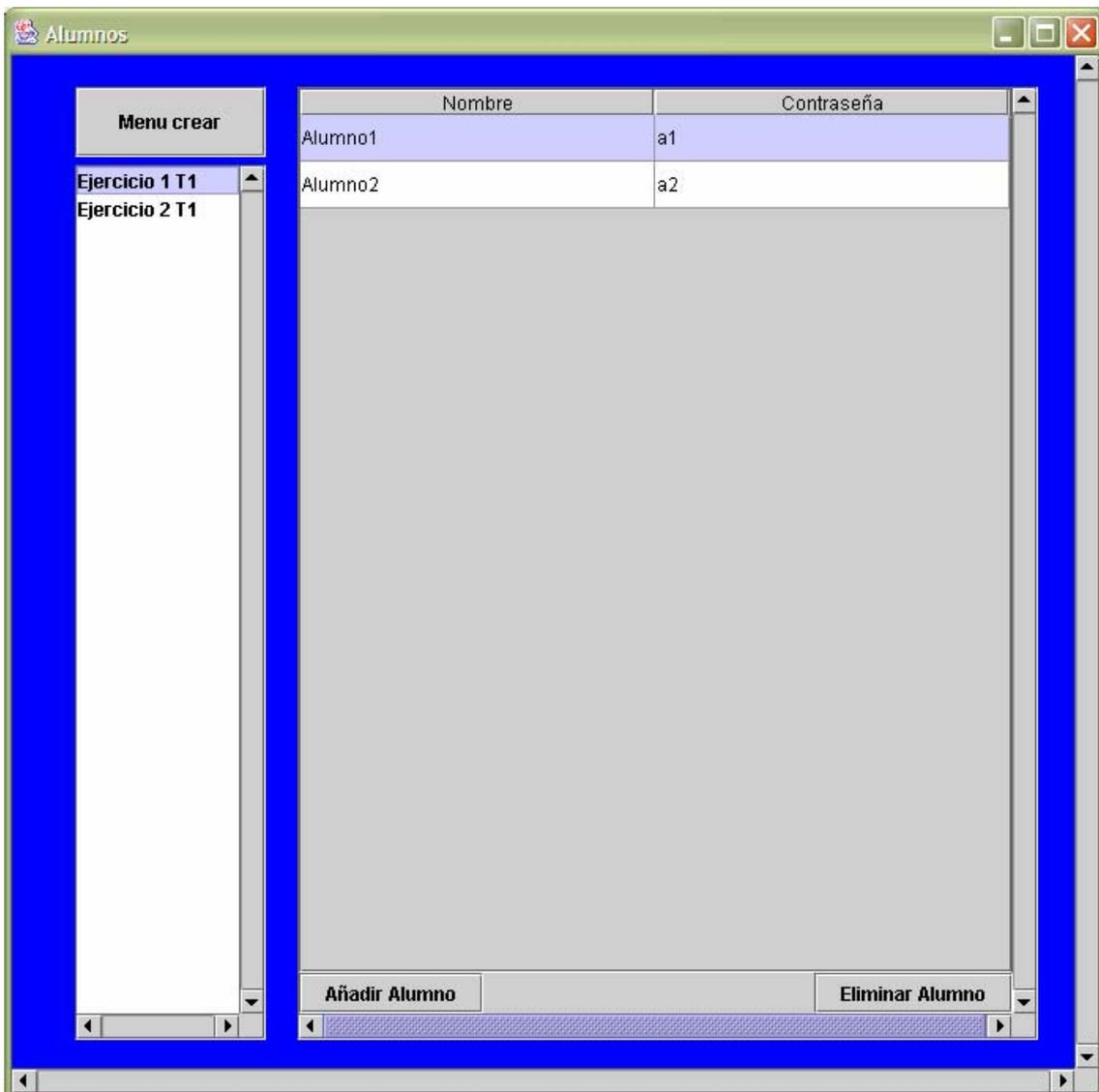


Figura 75: Creación de alumnos y grupos de trabajo

Desde esta ventana, el profesor del curso podrá crear un nuevo alumno pinchando sobre el botón de 'Añadir Alumno' e introduciendo el nombre de usuario y contraseña con el que podrá iniciar sesión en la aplicación desde su editor gráfico. Para la eliminación de un alumno creado tendrá que seleccionar el alumno que desea borrar del listado que aparece en pantalla y pinchar en el botón 'Eliminar Alumno'.

La figura 75 muestra el resultado final después de la creación de tres alumnos (Alumno1, Alumno2 y Alumno3) y del posterior borrado del último alumno creado.

Una vez definidos los alumnos que realizarán alguna de las actividades gráficas del curso, el profesor debe proceder a la formación de los grupos de trabajo de cada una de ellas.

Cuando se selecciona un alumno, en la parte izquierda de la pantalla aparece una lista con las actividades gráficas del curso. Cada una de las actividades gráficas de la lista representa a un grupo de trabajo. Si una actividad gráfica de la lista está seleccionada, indica que el alumno pertenece al grupo de trabajo de la actividad gráfica y que por tanto tiene que realizarla colaborativamente con sus compañeros.

Para incluir a un alumno en un grupo de trabajo, el profesor debe seleccionar al alumno, mantener pulsada la tecla 'ctrl' del teclado y pinchar en la actividad gráfica de la lista que aparece en la parte izquierda de la pantalla, la actividad gráfica quedará seleccionada. A continuación, aparecerá un diálogo en el que se pedirá el rol del alumno en el grupo de trabajo (jefe o normal).

Para eliminar a un alumno de un grupo de trabajo, el profesor debe seleccionar al alumno, mantener pulsada la tecla 'ctrl' del teclado y pinchar en la actividad gráfica de la lista que aparece en la parte izquierda de la pantalla, la actividad gráfica quedará deseleccionada.

Se han realizado pruebas sobre la inclusión de alumnos en varios grupos de trabajo y la eliminación de alumnos.

Para volver al menú de crear de la figura 70, el profesor deberá pulsar sobre el botón 'Menu Crear' (ver figura 75). Desde allí, podrá acceder a las opciones de envío del curso al servidor o de creación de temas.

Prueba 3: Visualización, almacenamiento de resultados y finalización de actividades gráficas

Cuando se está gestionando un curso, desde la ventana de una actividad gráfica se muestra el resultado del trabajo de los alumnos en esa actividad gráfica (ver figura 76). Para guardar el resultado de los alumnos, en un documento pdf, el profesor ha de pulsar sobre el botón 'Crear pdf'.

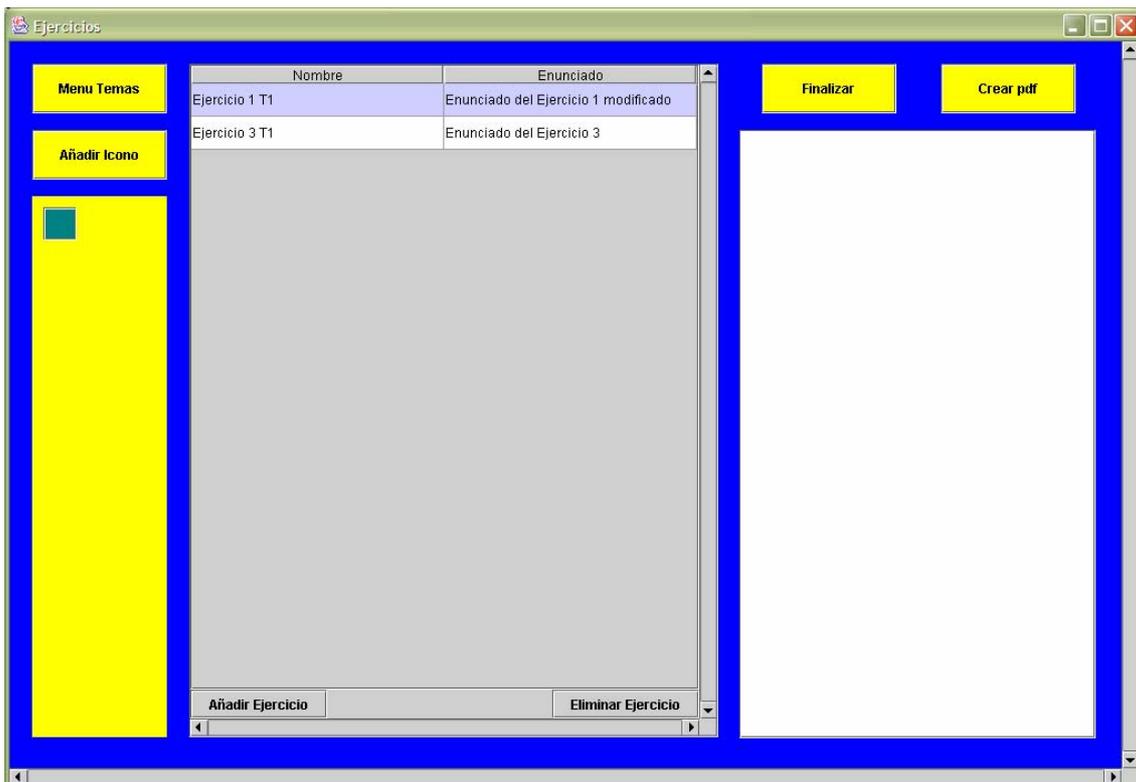


Figura 76: Resultados y finalización de actividades gráficas

Desde esta ventana el profesor del curso también puede finalizar una actividad gráfica pulsando sobre el botón 'Finalizar' e introduciendo los minutos de margen que hay hasta la finalización de la actividad gráfica.

Prueba 4: Actualizar el curso

Cuando el profesor acaba de crear o gestionar los temas y los alumnos de un curso, tiene que actualizarlo. De esta manera, los alumnos podrán acceder con sus editores gráficos al curso actualizado. La forma de hacerlo es pulsando sobre el botón de 'Crear el curso' para el caso de que lo esté creando (ver figura 70) o 'Guardar el curso' para el caso de que lo esté gestionando (ver figura 71).

En el caso de que el profesor esté gestionando un curso y haya cambiado los grupos de trabajo, cada alumno que se vea afectado recibirá un mensaje que le indicará este hecho y accederá a sus nuevos grupos de trabajo actualizados.

En el caso de que el profesor esté gestionando un curso y haya eliminado un alumno que está conectado, este finalizará su sesión. Si el alumno se encuentra trabajando en ese momento con su editor gráfico, recibirá un mensaje indicándole este hecho.

En el caso de que haya finalizado una actividad gráfica, todos los alumnos recibirán una notificación del tiempo restante que les queda para finalizarla.

Se han realizado pruebas sobre estos hechos y los resultados han sido correctos.

3.6.2 EDITOR GRÁFICO

Como se ha expuesto antes, a través del editor gráfico el estudiante podrá realizar actividades colaborativas que contengan iconos o imágenes. Para poder utilizarlo es necesario que esté en funcionamiento el servidor del alumno y el profesor y posteriormente cargar el editor gráfico.

Al iniciar el editor gráfico, la aplicación pide al alumno que se identifique en la aplicación con su nombre de usuario y contraseña. Si la identificación es válida (el alumno forma parte del curso) se le muestra un interfaz gráfico que le permite trabajar con sus compañeros cualquiera de las actividades gráficas de la lista de actividades gráficas pendientes.

Inicialmente, la aplicación carga los datos de la primera actividad gráfica de la lista (ver figura 77). Si el alumno quiere trabajar con otra actividad gráfica deberá seleccionarla de la lista. Tras realizar la selección y si la actividad gráfica no se encuentra finalizada, aparecerá en la parte superior derecha el enunciado, en la parte inferior izquierda los iconos asociados y en la parte inferior derecha el área de trabajo o estado actual de resolución de la actividad gráfica.

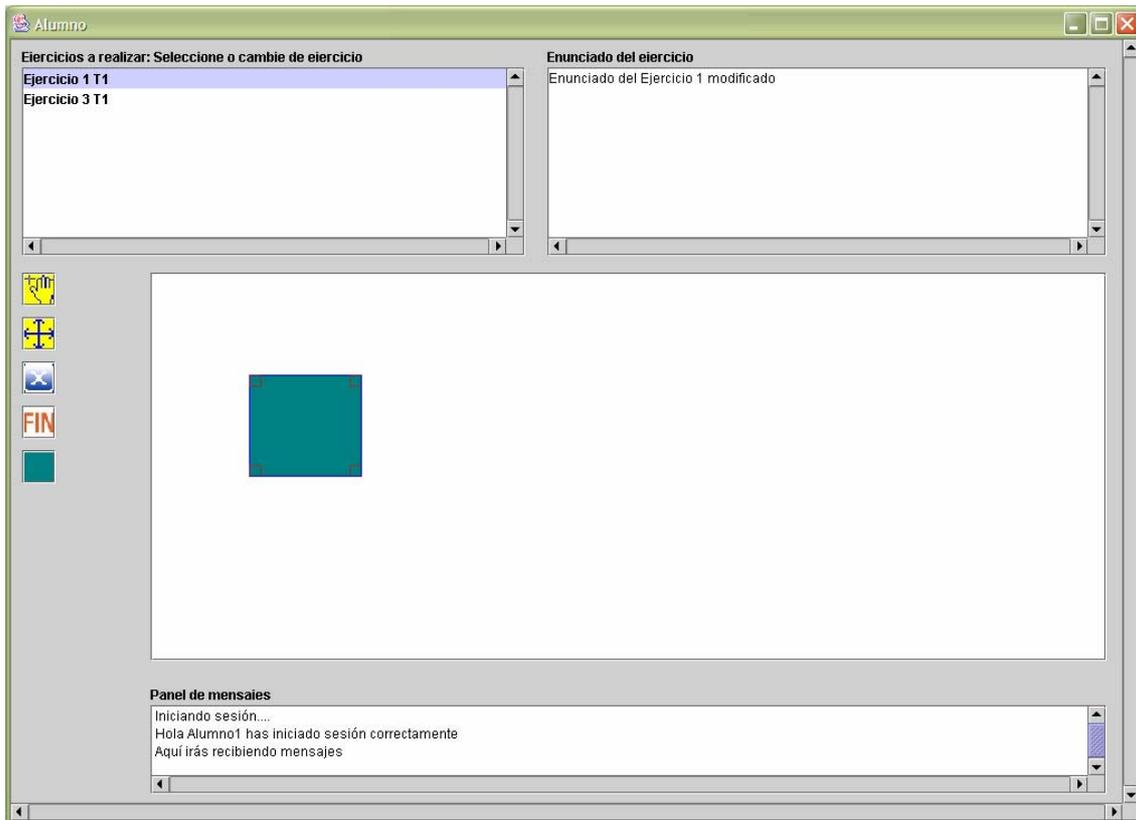


Figura 77: Inicio de sesión de un alumno en el editor gráfico

Prueba 1: Inserción, desplazamiento, modificación del tamaño y eliminación de elementos del área de trabajo de una actividad gráfica

Cada una de estas acciones queda reflejada en el área de trabajo de todos los alumnos del grupo de trabajo de la actividad gráfica (ver figura 77) y en la zona de visualización de la actividad gráfica de la herramienta de autor del profesor (ver figura 76).

Para insertar un elemento, el alumno ha de pinchar en él y pinchar sobre el área de trabajo (ver figura 78).

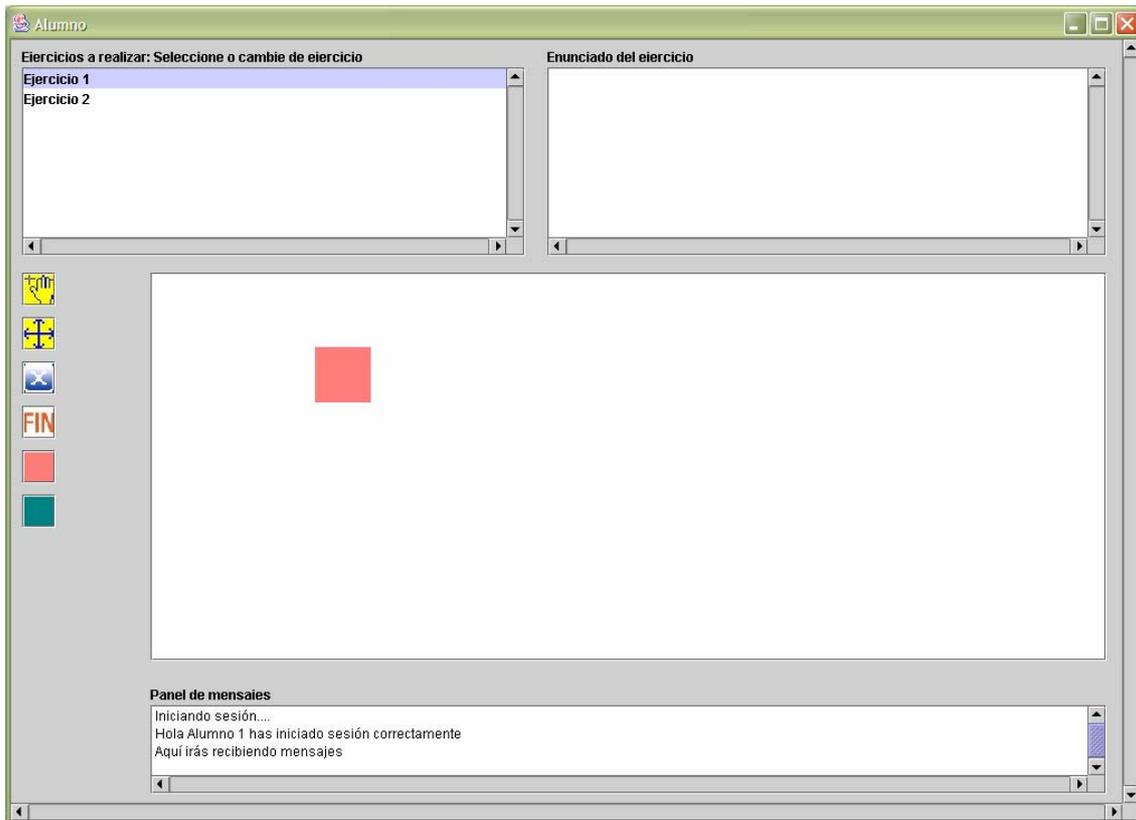


Figura 78: Inserción de un elemento en el área de trabajo

Para mover un elemento del área de trabajo, el alumno ha de pinchar en el icono, de la parte izquierda de la pantalla, que tiene el dibujo de una mano y después realizar las siguientes acciones en el elemento del área de trabajo que se quiere mover (ver figura 79).

1. Pinchar sin soltar
2. Arrastrar
3. Soltar

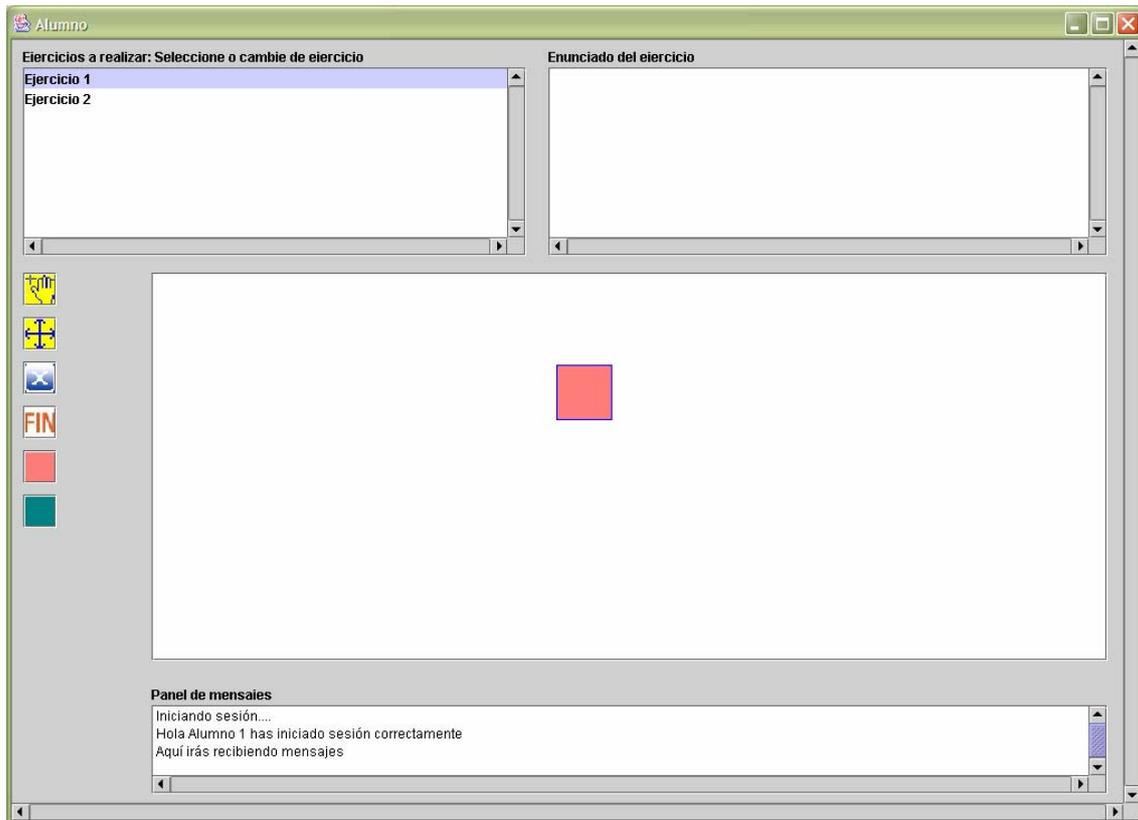


Figura 79: Desplazamiento de un elemento en el área de trabajo

Para modificar el tamaño de un elemento del área de trabajo, el alumno ha de pinchar en el icono, de la parte izquierda de la pantalla, que tiene el dibujo de una cruz y después realizar las siguientes acciones en el elemento del área de trabajo que se quiere mover (ver figura 80).

1. Pinchar sin soltar
2. Arrastrar
3. Soltar

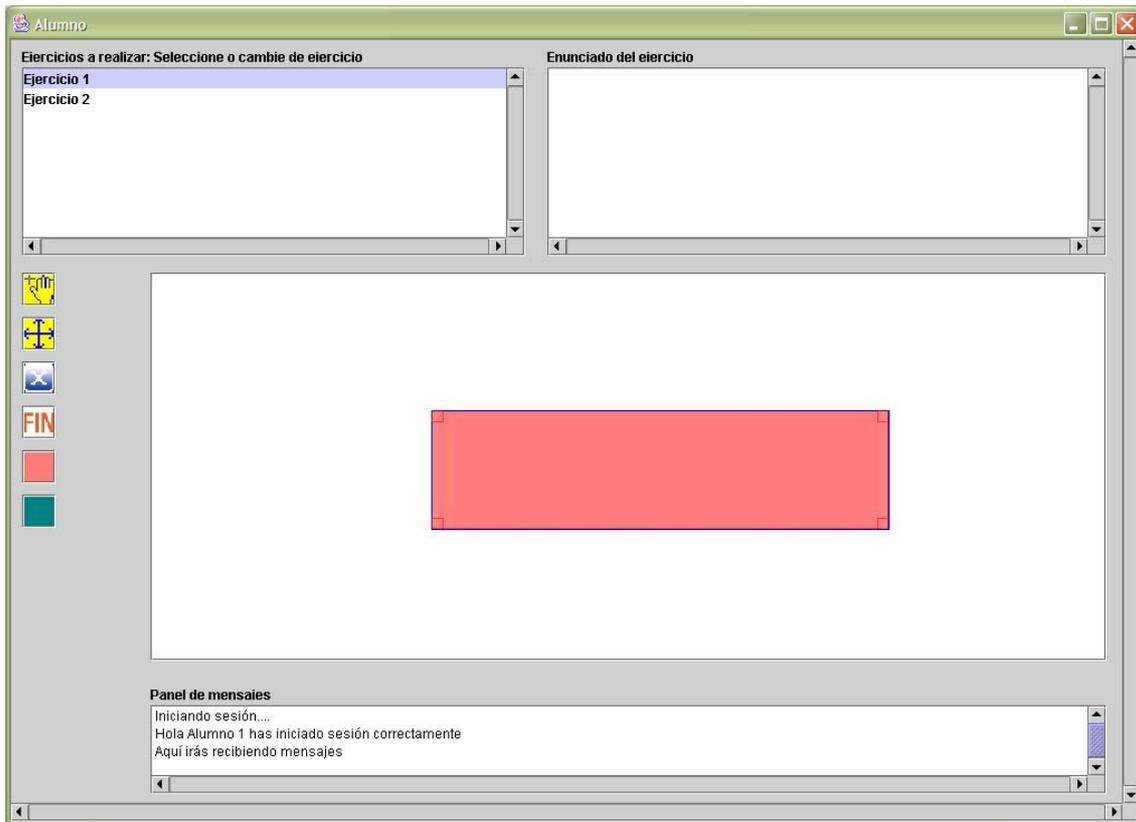


Figura 80: Modificación del tamaño de un elemento del el área de trabajo

Para eliminar un elemento del área de trabajo, el alumno ha de pinchar en el icono, de la parte izquierda de la pantalla, que tiene el dibujo de un aspa y después pinchar en elemento del área de trabajo que se quiere eliminar (ver figura 81).

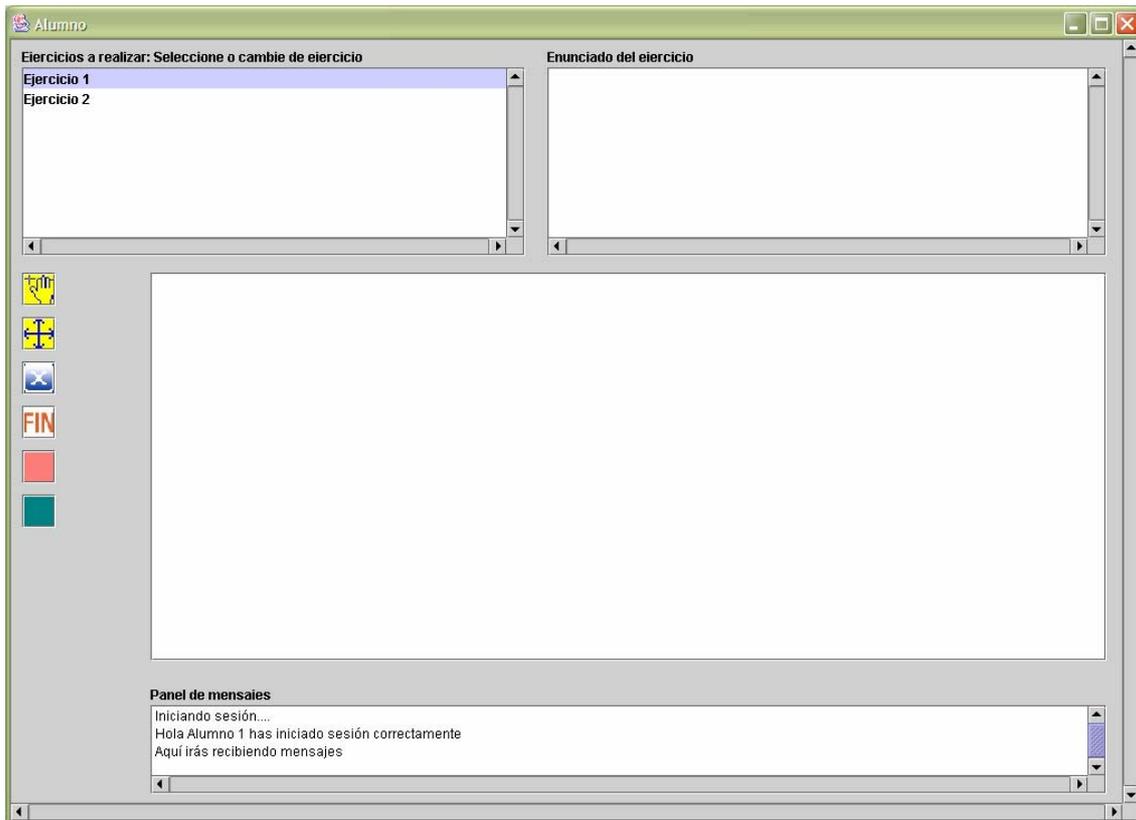


Figura 81: Eliminación de un elemento del área de trabajo

Prueba 2: Finalizar una actividad gráfica

Si en una actividad gráfica, el alumno posee un botón con el título 'FIN', tiene asignado el rol de jefe en esa actividad y por tanto podrá finalizarla. Cuando pulsa sobre este botón, la aplicación le informa a él mediante un mensaje (ver figura 82) y a todos los compañeros del grupo de trabajo (ver figura 83) que quedan 5 minutos para que la actividad gráfica se de cómo concluida. Pasado el tiempo, la actividad gráfica queda finalizada y por tanto ningún miembro del grupo de trabajo podrá modificar el resultado.

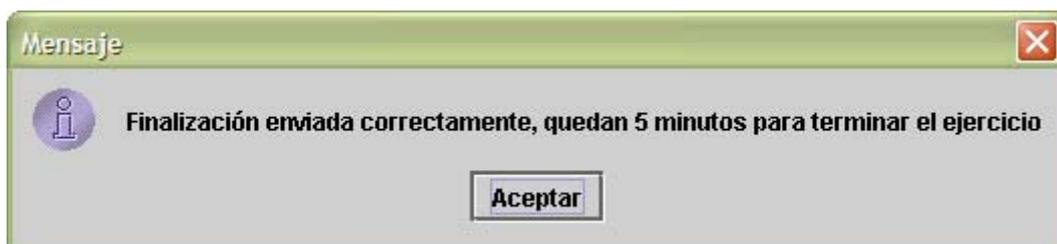


Figura 82: Aviso de envío de finalización y del tiempo de finalización

Del mismo modo si el profesor finalizó una actividad gráfica, todos los alumnos del grupo de trabajo de la misma recibirán un mensaje en el que se indica el tiempo de finalización que el profesor estableció (ver figura 83).

Se ha realizado una prueba en la que un alumno finaliza una actividad gráfica y se ha observado que todos los alumnos del grupo de trabajo reciben la notificación. También se ha probado que cuando la actividad gráfica está finalizada el editor no permite acceder al área de trabajo al seleccionarse de la lista de actividades gráficas.

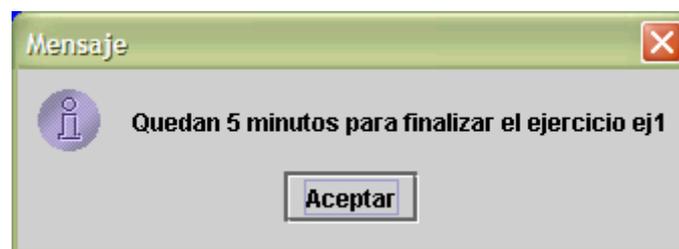


Figura 83: Aviso del tiempo de finalización de una actividad gráfica

Prueba 3: Cambio de grupos de trabajo y de actividades gráficas

Si el profesor ha modificado una actividad gráfica, el alumno recibe un mensaje que le informa de este hecho y ve actualizado el enunciado o los iconos de trabajo de la actividad gráfica (ver figura 84).

Si el profesor ha cambiado los grupos de trabajo a los que pertenece un alumno, éste recibe un mensaje que le informa de este hecho (ver figura 84) y ve actualizada su lista de actividades gráficas.

Se ha probado que lo descrito anteriormente ha funcionado correctamente.

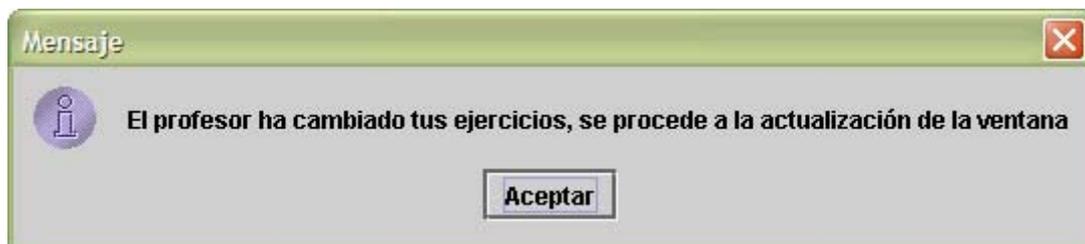


Figura 84: Cambio de actividades gráficas y de grupos de trabajo.

Prueba 4: Finalización de sesión

Si el profesor ha finalizado la sesión de un alumno, este recibe un mensaje y se le informa del hecho. Posteriormente se cierra su aplicación (ver figura 85).

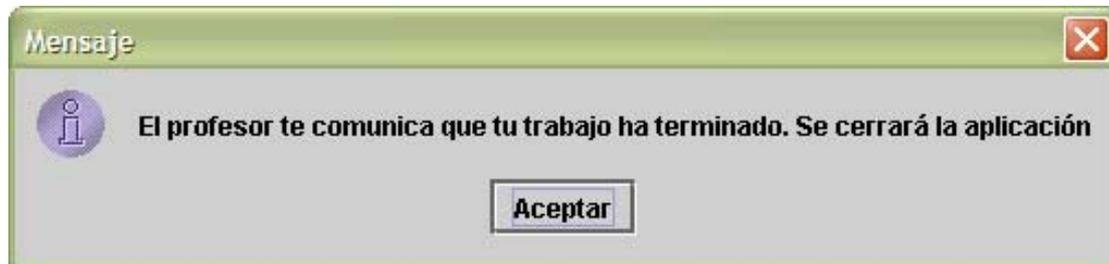


Figura 85: Finalizar sesión

3.6.3 EDITOR TEXTUAL

Lo primero que hay que hacer es cargar el servidor del alumno y después el editor textual. Cuando un alumno inicia el editor textual la aplicación le pide un nombre, después le muestra un interfaz gráfico en el cual, tras finalizar el inicio de sesión, le muestra un mensaje de bienvenida y le permite trabajar con sus compañeros en la actividad textual que han de realizar (ver figura 86).

Inicialmente, la aplicación carga el estado del documento de la actividad textual (ver figura 86). En la parte superior se mostrarán las acciones que se pueden realizar con el documento y en la parte central el documento (este equivale al área de trabajo en el caso de actividades gráficas).

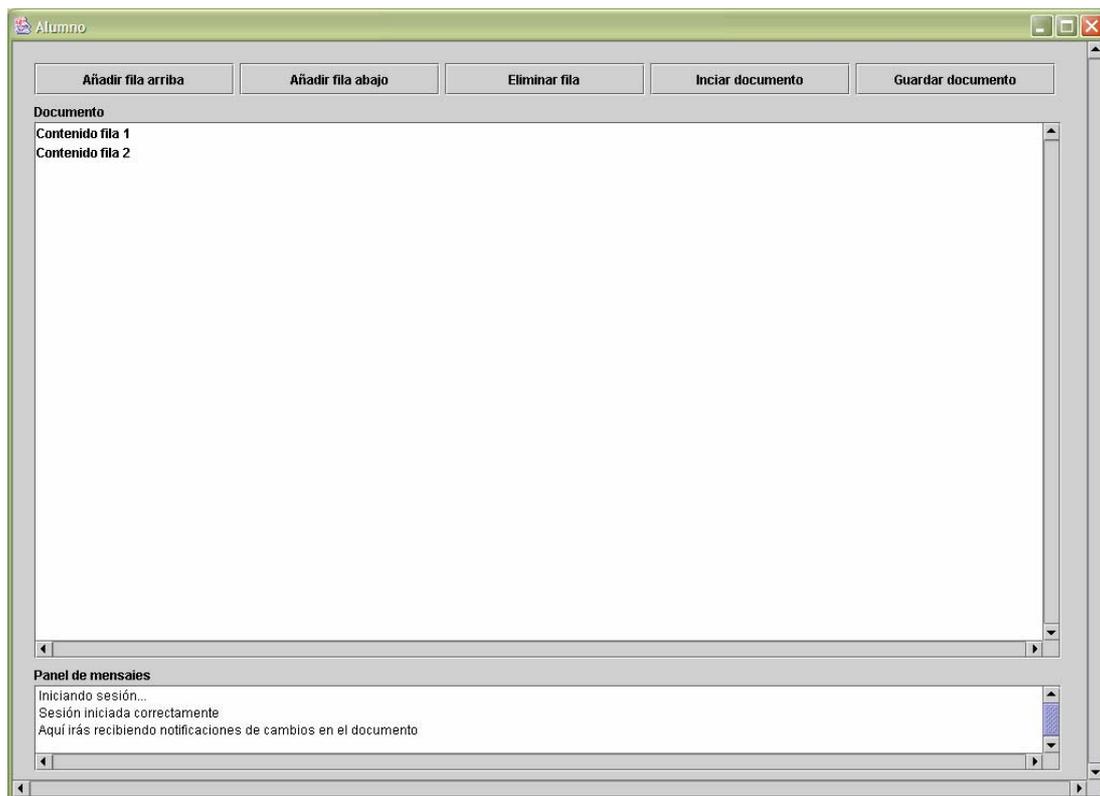


Figura 86: Inicio de sesión en el editor textual

Prueba 1: Inserción de filas en el documento

El alumno puede insertar una fila en el documento seleccionando una fila ya creada y pulsando el botón '*Añadir fila arriba*' para añadirla en una posición superior a la seleccionada o en '*Añadir fila abajo*' para añadirla en una posición inferior a la seleccionada. Si no hubiera ninguna fila en el documento, cualquiera de los botones anteriormente indicados insertaría una fila inicial.

Si hay filas en el documento pero no selecciona ninguna antes de pulsar uno de los botones, la aplicación muestra un mensaje indicando que hay que seleccionar primero una fila. La inserción de la fila queda reflejada en el documento de todos los alumnos del grupo de trabajo.

Se han hecho pruebas de este funcionamiento insertando 7 filas, tres superiores y tres inferiores a una fila inicial, el cambio ha quedado reflejado en el documento de todos los alumnos del grupo de trabajo.

Prueba 2: Eliminación de filas

El alumno puede borrar filas en el documento seleccionando una fila ya creada y pulsando el botón '*Eliminar fila*'.

Si no selecciona ninguna fila, se muestra un mensaje indicando que hay que seleccionar primero una fila.

La eliminación de la fila queda reflejada en el documento de todos los alumnos del mismo grupo de trabajo.

Se han realizado pruebas de este funcionamiento eliminando 5 filas, el cambio ha quedado reflejado en el documento de todos los alumnos del grupo de trabajo.

Prueba 3: Inicialización del documento

El alumno puede inicializar el documento, es decir borrar todas las filas, pulsando el botón '*Iniciar documento*'. La inicialización queda reflejada en el documento de todos los alumnos del grupo de trabajo.

Se han hecho pruebas de este funcionamiento, la inicialización ha quedado reflejada en el documento de todos los alumnos del grupo de trabajo.

Prueba 4: Guardar el documento

El alumno puede guardar el documento pulsando el botón '*Guardar el documento*'. La aplicación muestra un cuadro de diálogo en que introducir un nombre de archivo en el sistema de ficheros, tras ello crea un pdf con el contenido del documento.

Se han hecho pruebas de este funcionamiento y se ha comprobado que el pdf se crea correctamente.

SECCIÓN 4

CONCLUSIONES

La principal ventaja de la aplicación desarrollada en este Proyecto Fin de Carrera es que permite a los estudiantes la realización de actividades colaborativas a través de Internet de tal manera que pueden beneficiarse de las ventajas del aprendizaje colaborativo (desarrollo de ciertas habilidades personales y sociales tales como razonar o aprender a trabajar en grupo). Además permite a los profesores la reutilización de material ya creado, con lo que el tiempo empleado en la ardua tarea de creación de materiales y cursos puede verse reducido.

La aplicación se deberá probar con estudiantes reales y profesores para ver si es útil en la educación a distancia. Se deberá evaluar si la posibilidad de reutilización de materiales multimedia que se da a los profesores ahorra tiempo de desarrollo del curso y si hacen uso de ella o prefieren insertar nuevos contenidos. Además, los estudiantes deberán dar su opinión sobre la funcionalidad y la facilidad de la aplicación.

Se deben realizar experimentos con distintos estudiantes y distintas configuraciones de los grupos de trabajo y de los roles de los alumnos para ver cuál es la mejor organización de los estudiantes (número de estudiantes por grupo y roles) y en qué situaciones se produce el mejor rendimiento de los grupos de trabajo.

El trabajo presentado podría ser integrado dentro de sistemas existentes de educación a distancia que permitan la realización de actividades colaborativas.

Además del campo de la enseñanza, esta aplicación se puede utilizar en el mundo empresarial para la realización de cualquier actividad colaborativa que necesite un editor gráfico compartido (p.e. la elaboración de un circuito electrónico o de un documento por varios empleados de una empresa).

BIBLIOGRAFÍA

Referencias

- [1] Ausbel. Psicología Educativa. 1976 Editorial Trillas
- Barros, B., Verdejo, M.F.: **DEGREE: un sistema para la realización y evaluación de experiencias de aprendizaje colaborativo en enseñanza a distancia**. Revista Iberoamericana de Inteligencia Artificial. Vol 9, 27-37. 2000.
- [2]
- [3] Benkiran, M.A., Ajhoun, R.: **An Adaptive and Cooperative Telelearning System: SMART-Learning**. International Journal on E-Learning, 66-72. Año 2002.
- [4] Benkiran, M.A., Ajhoun, R., M.: **SMART-Learning: Adaptive Telelearning System on the Web**. Workshop on Information Technology. Morocco. 2003.
- [5] Bruner, J. S. (1978): El proceso mental del aprendizaje. Madrid, Narcea.
- [6] Borghoff, U., Schlichter, J.: **Computer-Supported Cooperative Work - Introduction to Distributed Applications**. Springer. 2000.
- [7] Brown, J.S., Collins, A., Duguid, P.: **Situated Cognition and the Culture of Learning**. Educational Researcher; v18 n1, 32-42. 1989.
- [8] Bruner, J.: **Toward a Theory of Instruction**. Cambridge, MA: Harvard University Press. 1966.
- Carro, R.M.: Un mecanismo basado en tareas y reglas para la creación de sistemas hipermedia adaptativos: aplicación a la educación a través de Internet. Tesis doctoral. Departamento de Ingeniería Informática. UAM. 2001.
- [9]
- [10] CROWE M. K. Cooperative work with multimedia. Springer-Verlag, 1994
- [11] DeSantis, G., Gallupe, B.: **A Foundation for the Study of Group Decision Support Systems**. Management Science, Vol. 33, No. 5, 589-609. Mayo 1987.
- [12] Ellis, C. A., Gibbs, S. J., Rein, G. L.: **Groupware: Some issues and experiences**. Communications of the ACM, 34(1), 38-58. 1991. Panitz, T
- [13] Engeström, Y.: **Learning by Expanding: An activity-Theoretical Approach to Development Research**. Helsinki: Orienta-Konsultit Oy. 1997.

- [14] Furugori, N., Sato, H., Ogata, H., Ochi, Y., Yano, Y.: **COALE: Collaborative and Adaptive Learning Environment**. CSCL 2002, 493-494. 2002.
- [15] G. Booch, J. Rumbaugh, I. Jacobson *El Lenguaje Unificado de Modelado*. Addison Wesley. Madrid 2001
- [16] Galán, A.: Aplicación de la Tele-Enseñanza a la docencia interdisciplinar relacionada con los problemas del medio ambiente. Proceedings I Jornada sobre Nuevas Tecnologías en la Innovación Educativa. 2003.
- [17] Gaudioso, E., Boticario, J.G.: **WEBDL Un sistema adaptativo para el aprendizaje cooperativo a distancia a través de Internet**. Jornadas de Ingeniería del Software y Bases de Datos, taller "Sistemas Hipermedia Adaptativos y Colaborativos". El Escorial, Madrid, 18-22. 2002.
- [18] Gottdenker, J.S., Remidez, H., Hong, R., Yoon, S., Amelung, C., Musser, D. R., Laffey, J.M.: **Introduction to the Shadow netWorkspace**. Proceedings CSCL 2002, 527-528.
- [19] Johnson, D.W., Johnson, F.P.: **Joining together: group theory and group skills**. Pearson Allyn & Bacon. 2002
- [20] Johnson, D.W., Johnson, F.P.: **Learning Together: Group Theory and Group Skills**. Pearson Education 1975. ISBN 0-205-36740-2
- [21] Johnson, D. W., Johnson, R.T., Holubec, E.J.: **Cooperation in the Classroom**. Edina, MN: Interaction Book Company. 1984
- [22] Johnson, D. W., Johnson, R.T., Holubec, E.J., Roy, P.: **Circles of Learning: Cooperation in the Classroom**. Alexandria, VA: Association for Supervision and Curriculum Development, 1984.
- [23] Johnson, D.W., Johnson, R.T., Roy, P., Zaidman, B.: **Oral interaction in cooperative learning groups: Speaking, listening and the nature of statements made by high, medium and low-achieving students**. Journal of Psychology 119. 303-321. 1985
- [24] Johnson, D.W., Johnson, F.P.: **Learning together and alone: cooperative, competitive, and individualistic learning**. Pearson Allyn & Bacon. 1998.
- [25] Kaye, A.R.: **Collaborative learning through computer conferencing: the Najaden papers**. Nato Asi Series (Vol F:90). Springer-Verlag. 1992.

-
- [26] Koschmann, T.: **Paradigms shift and instructional technology**. In T. Koschmann (Ed.), CSCL: Theory and practice of an emerging paradigm, 1-23. Mahwah, NJ: Lawrence Erlbaum Associates. 1996.
- [27] Kuutti, K.: **The concept of activity as a basic unit of analysis for CSCW research**. Bannon, Robinson & Schmidt 1991, 249-264. 1991.
- [28] Kuutti, K., Arvonen, T.: **Identify Potential CSCW Applications by means of Activity Theory Concepts: A Case Example**. Proceedings CSCW 1992, ACM Press, 233-240.
- [29] Palincsar, A.S., Brown, A.L.: **Reciprocal teaching of comprehension-fostering and comprehension-monitoring activities**. Cognition and Instruction, 1, 117-175. 1984
- [30] Sharan, Y., Sharan, S.: **Group Investigation in the cooperative classroom**. In Sharan, 97-114. 1994.
- [31] Singer, J., Behrend, S., Roschelle, J.: **Children's collaborative use of a computer microworld**. Proceedings CSCW'88, 271-281. Oregon. 1988.
- [32] Slavin, R.E.: **Cooperative learning**. Review of Educational Research 50, 315-342. 1980.
- [33] Slavin, R.E.: **Cooperative Learning-Theory, Research and Practice**. Englewood Cliffs, NJ: Prentice Hall. 1990
- [34] Slavin, R. E.: **Cooperative learning in middle and secondary schools**. Clearinghouse, 69 (4), 200-204. 1996.
- [35] Soller, A., Cho, K., Lesgold, A.: Adaptive support for collaborative learning on the internet. ITS (Intelligent Tutoring System) conference, Montreal, Canada. 2000
- [36] Soller, A., Lesgold, A.: Knowledge Acquisition for Adaptive Collaborative Learning Environments. AAAI Fall Symposium: Learning How to Do Things, Cape Cod, MA. 2000
- [37] Soller, A.: Supporting Social Interaction in an Intelligent Collaborative Learning System. International Journal of Artificial Intelligence in Education, 12(1), 40-62. 2001.
- [38] Universidad Politécnica de Cataluña: Manual básico del BSCW para los participantes en un programa de acción. Instituto de ciencias de la educación.

- Vizcaíno, A., Contreras, J., Favela, J., Prieto, M.: **An Adaptive, Collaborative Environment to Develop Good Habits in Programming**. In: Gauthier, G., Frasson, C., VanLehn, K. (eds.): Intelligent Tutoring Systems 2000. LNCS 1839, 262-271. Springer. 2000.
- [39]
- Vygotsky, L.S.: **Mind in society: The development of higher psychological processes**. Cambridge MA: Harvard University Press. 1978.
- [40]
- Webb, N.M.: **Group Composition, Group Interaction, and Achievement in small groups**. Journal of Educational Psychology, June, 1982, 74, 4, 475-484.
- [41]
- Webb, N.M.: **Peer interaction and learning in small groups**. International Journal of Educational Research, 13, 21-39. 1989.
- [42]

Referencias HTTP

- [http_ref01] Wana Daphne Lin Hsiao, J.: CSCL Theories.
<http://www.edb.utexas.edu/csclstudent/Dhsiao/theories.html>
<http://www.wou.edu/liberalartsscience/naturalscienceandmath/math/class/cooplist.html>
- [http_ref02] <http://www.phpprojekt.com>
- [http_ref03] <http://www.eecs.umich.edu/distview/userdocs.html>
- [http_ref04] http://www.cnice.mecd.es/recursos2/convivencia_escolar/1_4.htm
<http://www.wcer.wisc.edu/nise/cl1/CL/doingcl/grproles.htm>
- [http_ref05] <http://www.groove.net>
- [http_ref06] <http://sns.internetschools.org/~schools/info/sns2/index.cgi>
- [http_ref07] <http://tornado.dia.fi.upm.es/caepia/numeros/9/barros.pdf>
- [http_ref08] <http://lilt.ics.hawaii.edu/lilt/software/kukakuka/>
- [http_ref09] <http://citeseer.ist.psu.edu/schlichter97lecture.html>

ANEXO 1: MANUAL DE INSTALACIÓN

A continuación se proporciona el manual de instalación de la aplicación para la herramienta de autor y los editores.

APLICACIONES DISPONIBLES:

EditorGrafico: El editor gráfico explicado en esta memoria.

EditorTextual: El editor textual explicado en esta memoria.

HerramientaDeAutor: La herramienta de autor explicada en esta memoria.

ServidorEditorGraficoYHerramientaDeAutor: El servidor para el editor gráfico y la herramienta de autor. Se ha de cargar antes para poder utilizar tanto el editor gráfico como la herramienta de autor.

ServidorEditorTextual: El servidor para el editor textual. Debe estar ejecutándose antes de que los alumnos carguen el editor textual.

CONTENIDO DEL CD

Directorio WebInstallers: En este directorio se encuentran los ficheros de instalación de cada una de las aplicaciones para una gran variedad de plataformas. En el directorio de cada una de las aplicaciones se encuentra el fichero *'install.htm'*. Cargando este fichero se puede descargar el instalador para la plataforma adecuada.

Directorio Ejecutables: En este directorio se encuentran los ejecutables para win32 de cada aplicación. Para ejecutar una aplicación, se ha de seleccionar su directorio y a continuación hacer clic en el archivo bat que se encuentre (está configurado para trabajar en local, para trabajar en red ver los siguientes apartados).

Directorio WindowsInstallers: En este directorio se encuentran los archivos de instalación para win32 de cada aplicación. Para ejecutar una aplicación se ha de pulsar sobre el archivo correspondiente. Tras instalar la aplicación, para ejecutarla hay que pulsar sobre el archivo bat que se encuentra en el directorio de instalación

correspondiente (está configurado para trabajar en local, para trabajar en red ver los siguientes apartados).

ARCHIVOS EJECUABLES Y ARGUMENTOS

En cada directorio de instalación de las aplicaciones disponibles se encuentra un archivo ejecutable. Este archivo es el que se ha de usar para cargar cada aplicación.

Independientemente de la plataforma elegida, para ejecutar cada aplicación se ha de pasar una serie de argumentos al ejecutable comentado anteriormente. A continuación se explican los argumentos requeridos para cada una de las aplicaciones disponibles.

EditorGrafico: Como primer argumento se ha de pasar el puerto del servidor gráfico (ha de ser el mismo que utilice la herramienta de autor) y como segundo argumento la dirección IP del servidor gráfico.

EditorTextual: Como primer argumento se ha de pasar el puerto del servidor textual y como segundo argumento la dirección IP del servidor textual.

HerramientaDeAutor: Como primer argumento se ha de pasar el puerto del servidor de la herramienta de autor (ha de ser el mismo que utilice el editor gráfico) y como segundo argumento la dirección IP de la herramienta de autor.

ServidorEditorGraficoYHerramientaDeAutor: Como único argumento se ha de pasar el puerto del servidor. La dirección IP en la que se encuentre el servidor y el puerto especificado son los argumentos que se han de pasar cargar al editor gráfico y la herramienta de autor.

ServidorEditorTextual: Como único argumento se ha de pasar el puerto del servidor. La dirección IP en la que se encuentre el servidor y el puerto especificado son los argumentos que se han de pasar cargar al editor textual.



FICHEROS .BAT

Para el caso en el que se quiera ejecutar una aplicación en win32 se ha facilitado un archivo .bat para cada una de las aplicaciones disponibles, de esta manera se facilita la especificación de los argumentos anteriormente comentados. Inicialmente están configurados para trabajar en local, para trabajar en red se ha de editar el archivo .bat deseado y cambiar los argumentos.

ANEXO 2: MANUAL DE USUARIO

A continuación se proporciona el manual de usuario para la herramienta de autor y los editores.

Herramienta de autor

Al cargar la herramienta se muestra el menú principal de la aplicación (ver figura 87). Para crear un nuevo curso se ha de pinchar en el botón '*Crear nuevo curso*' y para gestionar el curso actual se ha de pinchar en el botón '*Gestión curso actual*'.

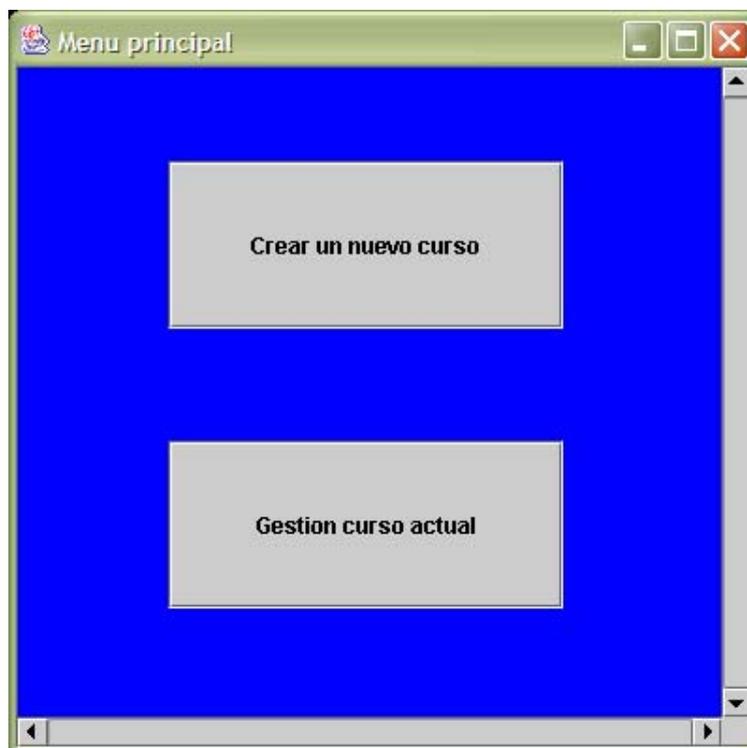


Figura 87: Menú principal

Como se ha comentado en el punto 3.6 estas 2 opciones son similares en funcionamiento y en las pantallas presentadas, por tanto los siguientes párrafos proporcionan la información del manual de usuario para cualquiera de las 2 opciones.

Una vez pinchado el botón '*Crear nuevo curso*' se muestra una pantalla en la que se puede pinchar en los botones, '*Crear Temas*', '*Crear Alumnos*', '*Menú principal*' y '*Crear el curso*' (ver figura 88).

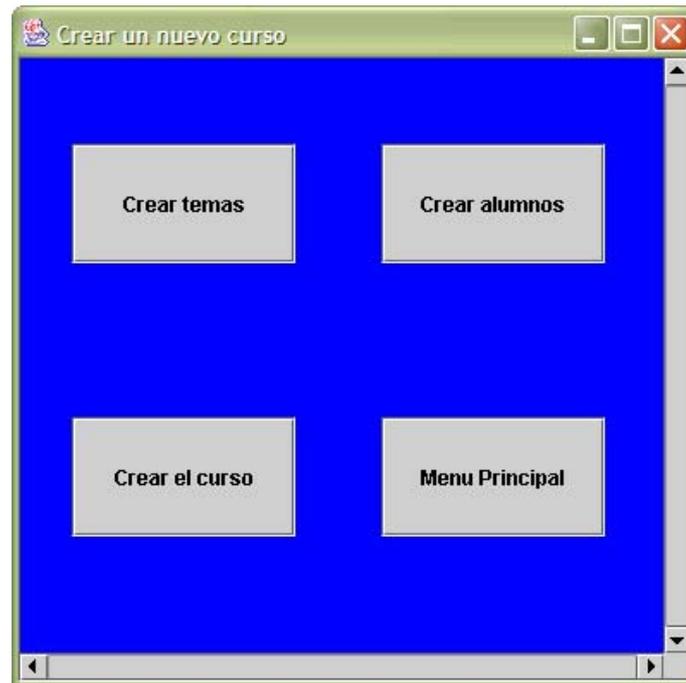


Figura 88: Creación de un nuevo curso

Para volver al menú anterior se ha de pinchar en el botón '*Menu principal*'.

Para mandar el curso al servidor, una vez creado, se ha de pinchar en el botón '*Crear el curso*'.

Botón '*Crear Temas*'

Para poder crear los temas del curso se ha de pinchar en el botón '*Crear Temas*'. Una vez pinchado en el botón, se muestra una pantalla en la que se pueden añadir los temas (ver figura 89).

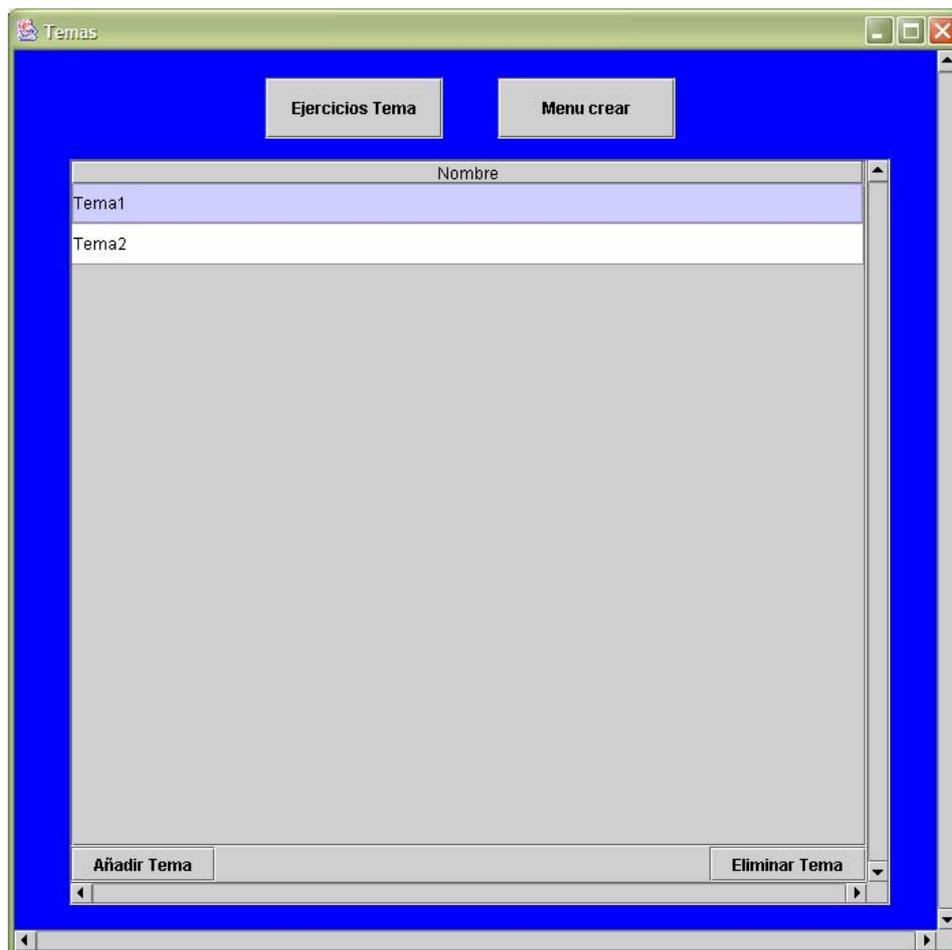


Figura 89: Inserción de temas

Para añadir un tema, se ha de pulsar sobre el botón '*Añadir Tema*', a continuación aparecerá un cuadro de diálogo en el que se pedirá el nombre del tema. Tras introducirlo, se mostrará el nombre en la lista de temas que aparece en pantalla.

Para eliminar un tema se ha de seleccionar su nombre de la lista de temas que aparece en pantalla y pinchar en el botón '*Eliminar Tema*'. A continuación, el tema quedará eliminado de la lista de temas.

Para volver al menú anterior se ha pinchar en el botón '*Menu crear*'.

Para añadir actividades gráficas a un tema se ha de seleccionar el tema de la lista de temas que aparece en pantalla y pinchar en el botón '*Ejercicios Tema*'. Una vez pinchado en el botón, se muestra una pantalla en la que se pueden añadir las actividades gráficas (ver figura 90).

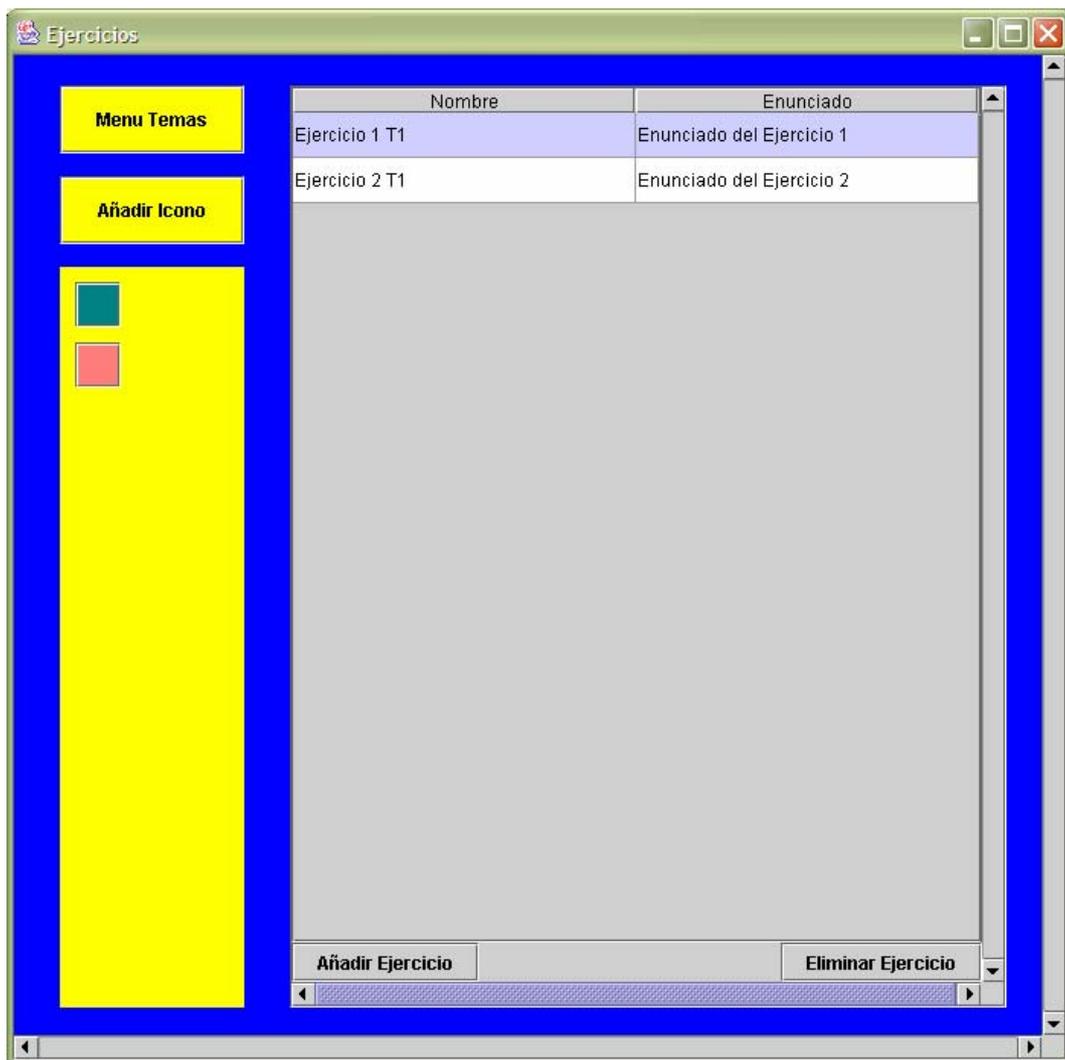


Figura 90: Inserción de actividades gráficas

Para añadir una actividad gráfica se ha de pinchar en el botón 'Añadir Ejercicio', a continuación aparecerá un cuadro de diálogo en el que se pedirá el nombre y el enunciado de la actividad gráfica. Tras introducirlos, se mostrará el nombre y el enunciado en la lista de actividades gráficas que aparece en pantalla.

Para modificar el enunciado de una actividad gráfica se ha de pinchar dos veces en el enunciado de la actividad gráfica que se quiera cambiar e introducir el nuevo enunciado.

Para eliminar una actividad gráfica se ha de seleccionar su nombre de la lista de actividades gráficas que aparece en pantalla y pinchar en el botón 'Eliminar'

Ejercicio'. Tras pinchar en el botón, la actividad gráfica quedará eliminada de la lista de actividades gráficas.

Si se está en la gestión del curso para finalizar una actividad gráfica se ha de pinchar en el botón 'Finalizar' y para guardar el resultado en un pdf se ha de pinchar en el botón '*Crear pdf*' (ver figura 91).

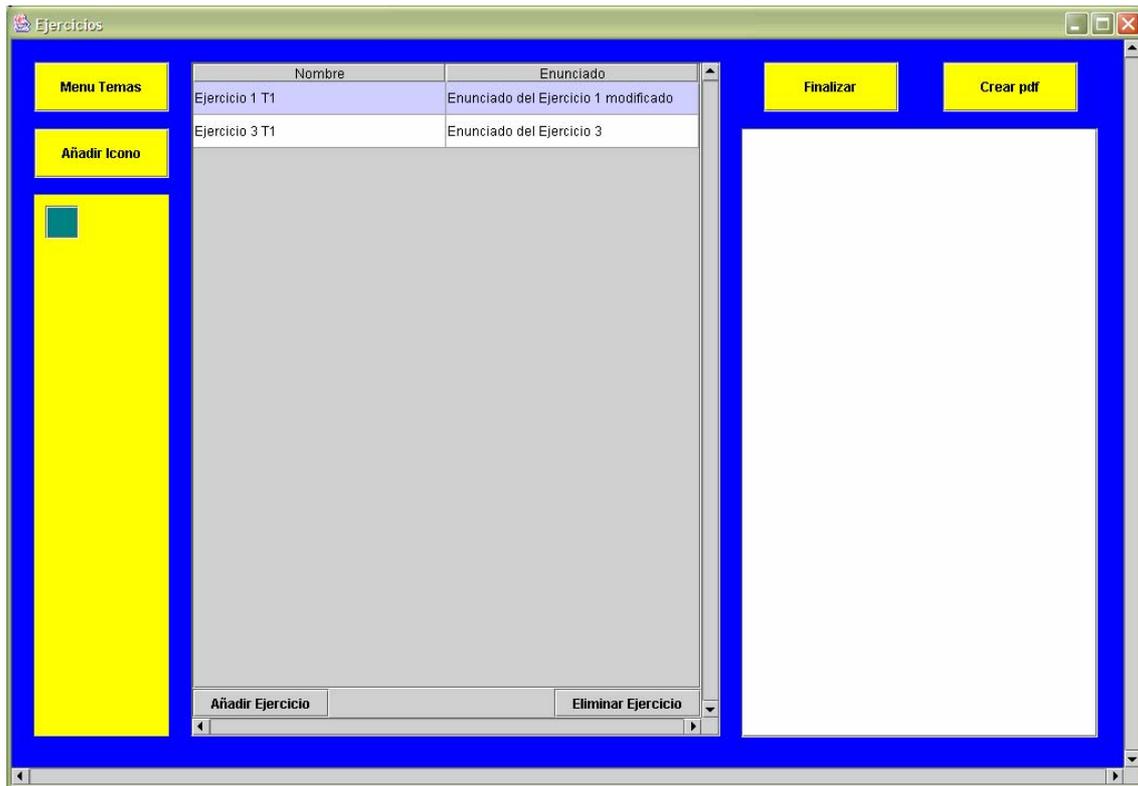


Figura 91: Resultados y finalización de actividades gráficas

Para añadir un icono a una actividad gráfica se ha de seleccionar la actividad gráfica de la lista de actividades gráficas que aparece en pantalla y pinchar en el botón '*Añadir Icono*'. Una vez pinchado el botón, se muestra una pantalla en la que se pueden añadir el icono (ver figura 92).

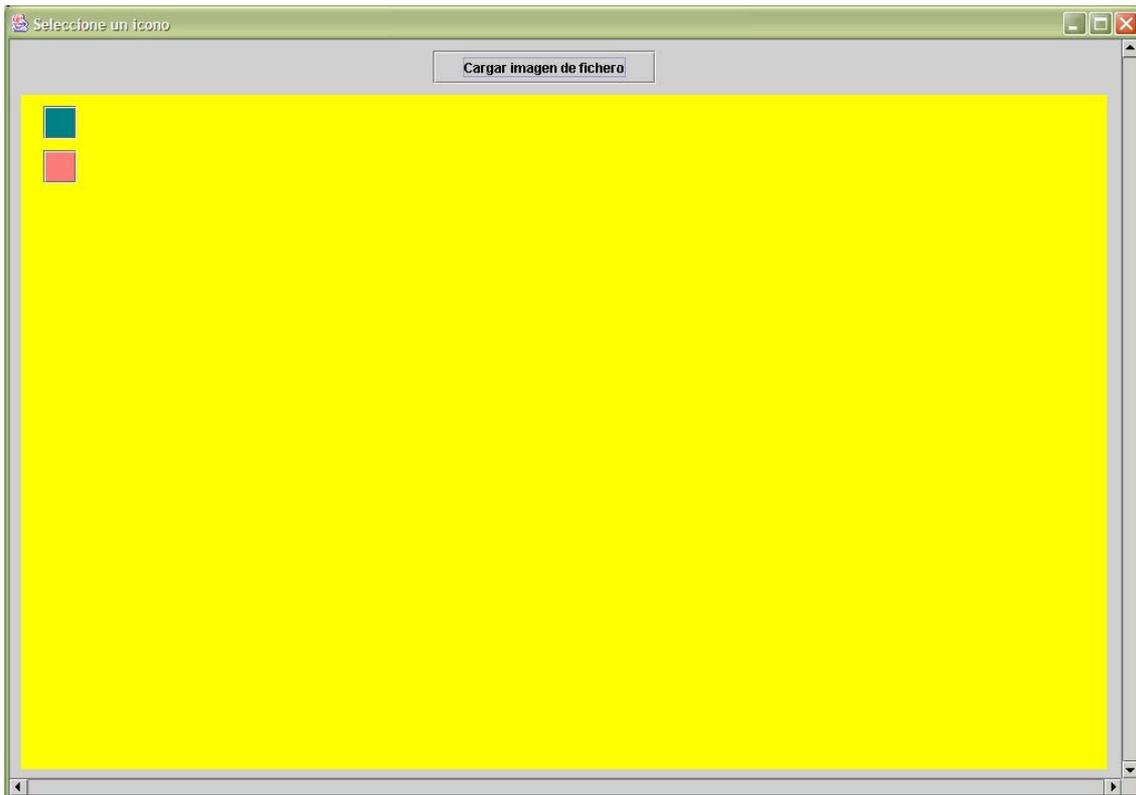


Figura 92: Inserción de iconos

Para añadir el icono se ha de pinchar en uno de los iconos reutilizables que aparecen en la ventana o pinchar en el botón '*Cargar imagen de fichero*' que mostrará un diálogo que permitirá buscar por el sistema de ficheros la imagen del icono. Tras añadirlo el icono quedará fijado en la lista de iconos que aparece en la ventana de actividades gráficas (ver figura 90).

Para eliminar un icono de una actividad gráfica se ha de seleccionar la actividad gráfica de la lista de actividades gráficas que aparece en pantalla y pinchar en el botón del icono de la lista de iconos que aparece en la ventana de actividades gráficas (ver figura 90).

Para volver al menú de temas se ha de pinchar en el botón '*Menu Temas*'.

Botón '*Crear Alumnos*'

Para poder crear los alumnos y los grupos de trabajo del curso se ha de pinchar en el botón '*Crear Alumnos*'. Una vez pinchado en el botón, se muestra una

pantalla en la que se pueden añadir los alumnos y crear los grupos de trabajo (ver figura 93).

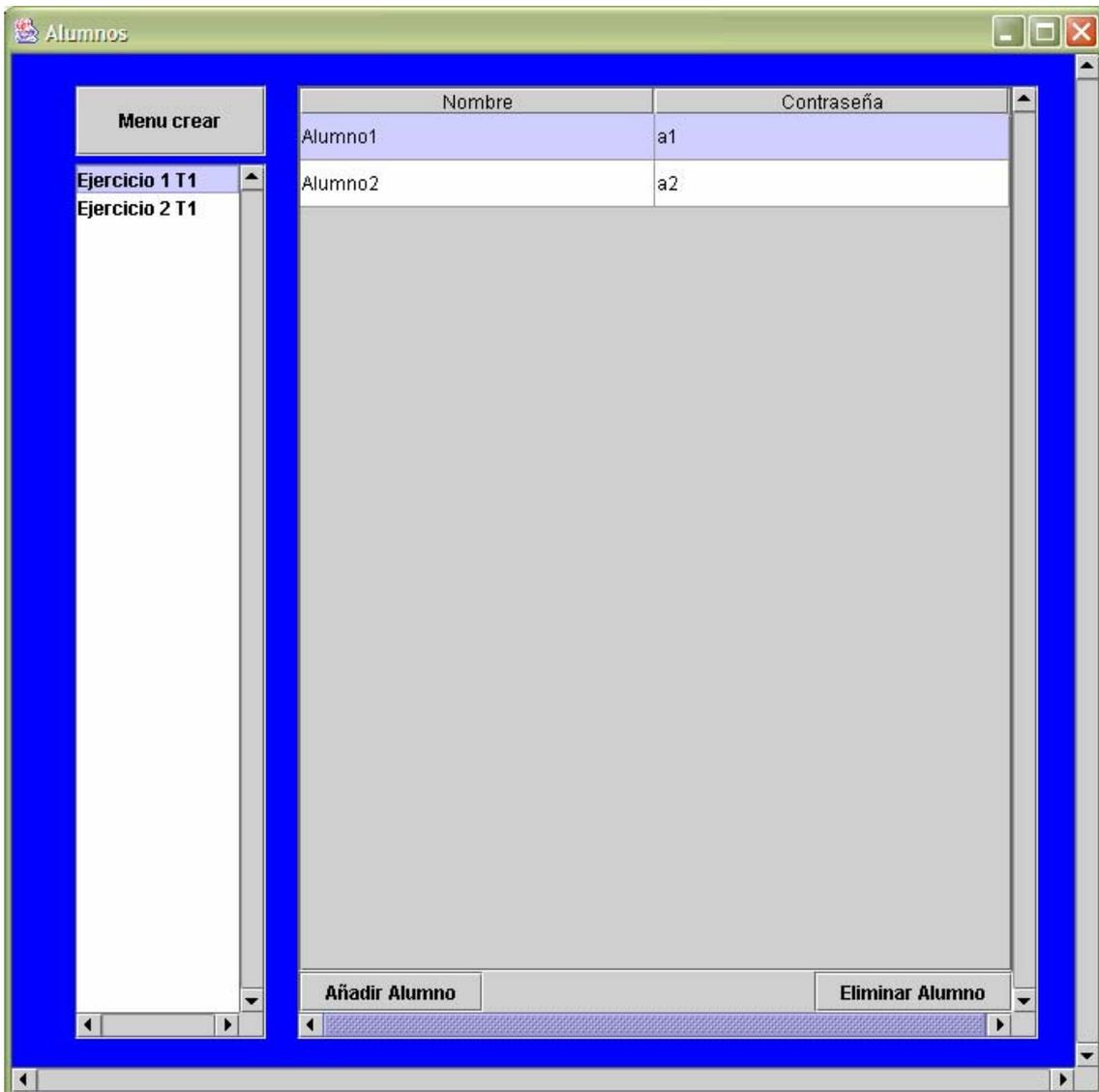


Figura 93: Creación de alumnos y grupos de trabajo

Para añadir un alumno se ha de pinchar en el botón 'Añadir Alumno', a continuación aparecerá un cuadro de diálogo en el que se pedirá el nombre y la contraseña del alumno. Tras introducirlos, se mostrará el nombre y la contraseña en la lista de alumnos que aparece en pantalla.

Para eliminar un alumno se ha de seleccionar su nombre de la lista de alumnos que aparece en pantalla y pinchar en el botón 'Eliminar Alumno'. Tras pinchar en el botón, el alumno quedará eliminado de la lista de alumnos.

Para volver al menú anterior se ha pinchar en el botón '*Menu crear*'.

Para asignar a un alumno una actividad gráfica y por ende un grupo de trabajo se ha de seleccionar el nombre del alumno de la lista de alumnos que aparece en pantalla. A continuación se ha de seleccionar la actividad gráfica de la lista de actividades gráficas que aparece en pantalla con la tecla 'ctrl' pulsada. Tras esto se mostrará un cuadro de diálogo en el que se ha de elegir si el alumno será jefe para la actividad gráfica o normal. Finalmente la actividad gráfica quedará seleccionada en la lista de actividades gráficas de ese alumno.

Para eliminar la asignación a un alumno de una actividad gráfica y por ende un grupo de trabajo se ha de seleccionar el nombre del alumno de la lista de alumnos que aparece en pantalla. A continuación se ha de seleccionar la actividad gráfica de la lista de actividades gráficas que aparece en pantalla con la tecla 'ctrl' pulsada. Finalmente la actividad gráfica quedará no seleccionada en la lista de actividades gráficas de ese alumno.

Editor gráfico

Al cargar el editor gráfico se pide el nombre y la contraseña del alumno para que inicie sesión en el sistema. Si la identificación es válida (el alumno forma parte del curso) se le muestra un interfaz gráfico que le permite trabajar con sus compañeros en cualquiera de las actividades gráficas de la lista de actividades gráficas pendientes (ver figura 94).

Para trabajar con una actividad gráfica se ha de seleccionar esta de la lista de actividades gráficas que el alumno tienen pendiente de realizar. A continuación se mostrará su enunciado y el estado del área de trabajo de dicha actividad.

Para añadir un icono al área de trabajo se ha de pinchar en el botón del icono y a continuación pinchar en un punto del área de trabajo.

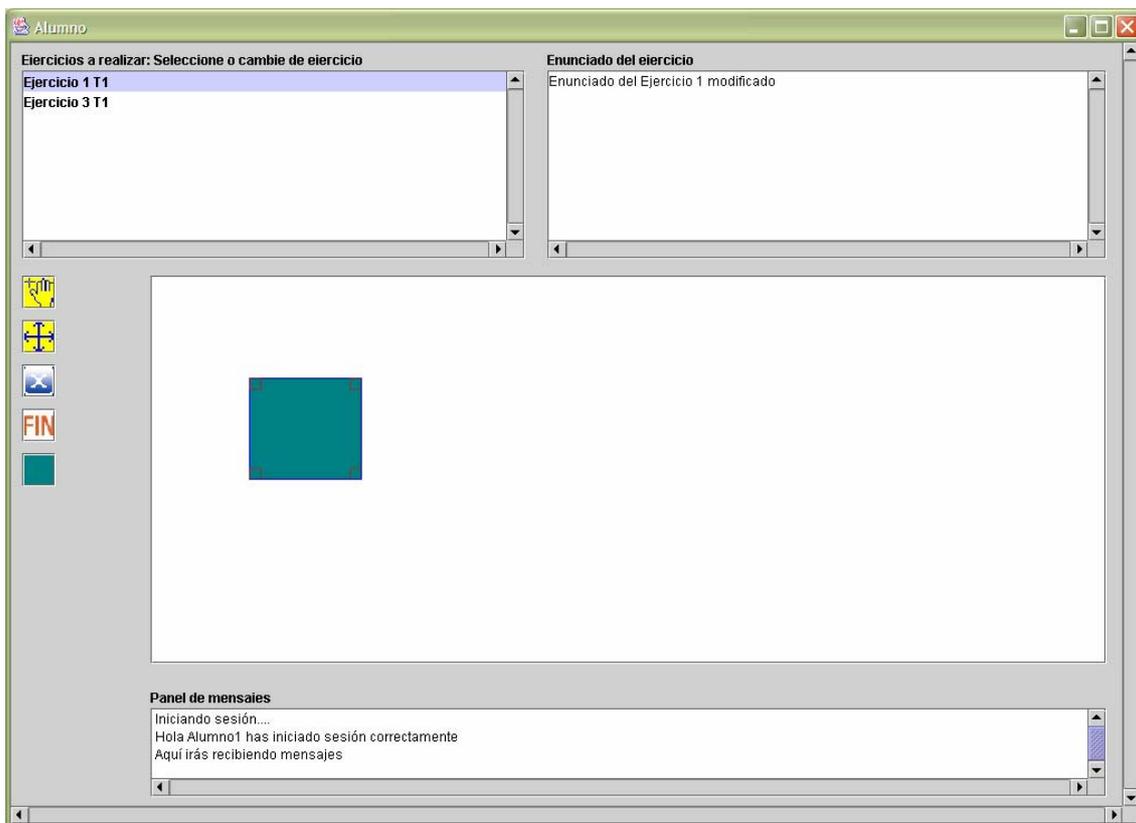


Figura 94: Inicio de sesión de un alumno en el editor gráfico

Para mover un elemento del área de trabajo se ha de pinchar en botón que tiene el dibujo de una cruz, a continuación se ha de pinchar sin soltar en el elemento gráfico, posteriormente se ha de mover el ratón por el área de trabajo y para finalizar se ha de soltar el elemento en el punto del área de trabajo en el que se quiera establecer.

Para cambiar el tamaño de un elemento del área de trabajo se ha de pinchar en el botón que tiene el dibujo de un aspa, a continuación se ha de pinchar sin soltar en una de las esquinas que aparecen en el elemento gráfico, posteriormente se ha de mover el ratón por el área de trabajo para aumentar su tamaño y soltar cuando el elemento tenga el tamaño deseado.

Para eliminar un elemento del área de trabajo se ha de pinchar en botón que tiene el dibujo de un aspa y a continuación se ha de pinchar en el elemento del área de trabajo que se quiere eliminar.

Editor textual

Al cargar el editor textual se pide el nombre del alumno para que inicie sesión en el sistema. Tras ello se le muestra un interfaz gráfico que le permite trabajar con sus compañeros en la actividad textual (ver figura 95).

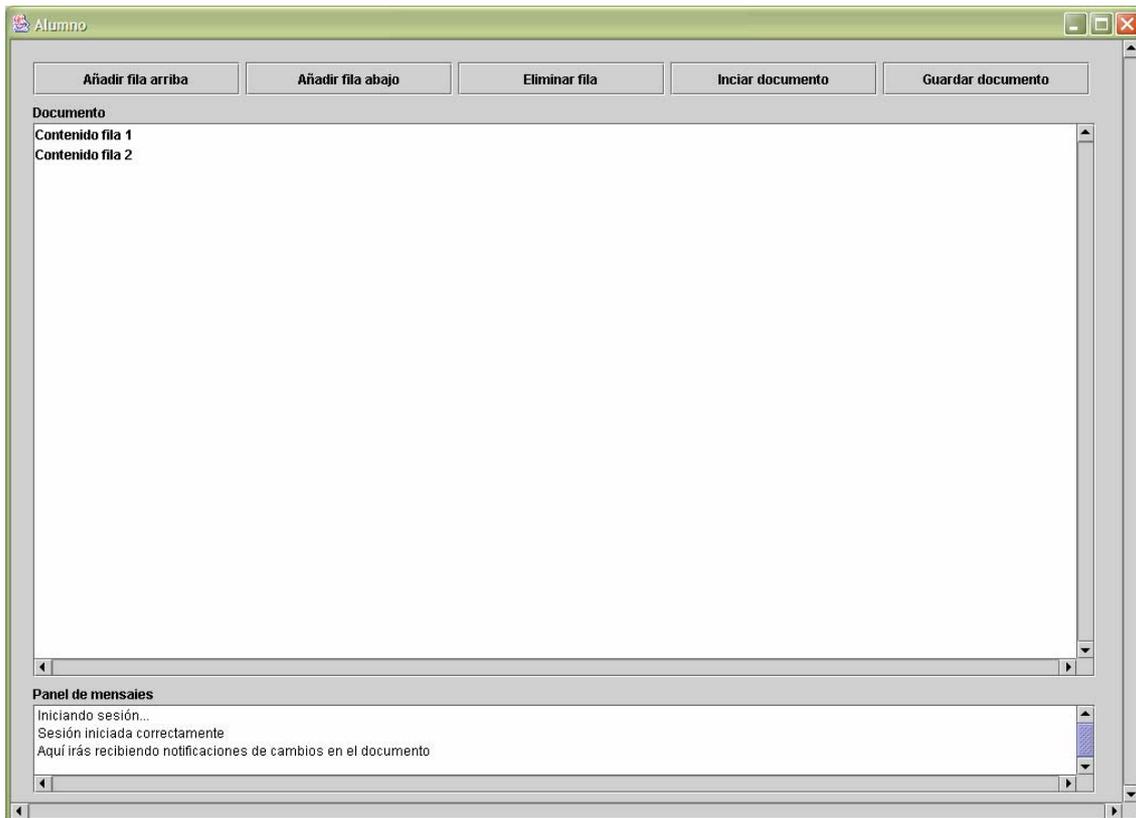


Figura 95: Inicio de sesión en el editor textual

Para añadir la primera fila al documento se ha de pulsar el botón '*Añadir fila arriba*' o bien en el botón '*Añadir fila abajo*' e introducir el texto en un cuadro de diálogo emergente.

Para añadir filas sucesivas se ha de seleccionar una fila existente y a continuación se ha de pinchar en el botón '*Añadir fila arriba*' si se quiere que la fila se inserte por la parte superior o en el botón '*Añadir fila abajo*' si se quiere que la fila se inserte por la parte inferior.

Para eliminar una fila se ha de seleccionar una fila existente y a continuación se ha de pinchar en el botón '*Eliminar fila*'.



Para eliminar todas las filas se ha de pinchar en el botón '*Iniciar documento*'.

Para guardar el documento en un pdf se ha de pinchar en el botón '*Guardar documento*'. A continuación se mostrará un diálogo que permitirá indicar el nombre en el sistema de ficheros, tras indicarlo el documento quedará guardado en el pdf.