

Análisis y diseño del sistema

CAPÍTULO 3

CAPÍTULO 3 – Análisis y diseño del sistema

En este capítulo se pretende realizar un análisis detallado de los requerimientos del software a desarrollar para la creación de un repositorio para programas de Java, incluyendo una descripción general del sistema y los casos de uso.

3.1 Funciones principales del sistema

- Almacenar y recuperar código fuente de java, utilizando la herramienta de ORM *Hibernate* para el mapeo entre los dos modelos
- Compilar código fuente en el momento de subir el programa
- Permitir búsqueda por dos criterios:
 - palabras o fragmentos de código
 - por clasificación
- Generar código fuente coloreado en formato html.
- Control de usuarios y clasificaciones existentes.

3.2 Descripción de interfaces con otros sistemas o dispositivos

A continuación se describen las interacciones con otros sistemas. Por tratarse de una aplicación que maneja datos persistentes se utiliza una base de datos. El contexto web es

proporcionado a través de Apache Tomcat siguiendo una arquitectura cliente-servidor. A continuación se explica más a detalle.

Interacción con el ORM

El software a desarrollar tendrá una interacción constante con el ORM *Hibernate* debido a que a este le será delegado el manejo de la persistencia de los datos para facilitar el desarrollo de la aplicación. Este como se mencionó en el capítulo 2 es un software gratuito y *open source* y cumple con los requerimientos.

Interacción con la base de datos:

El software a desarrollar tendrá interacción constante con una base de datos. La interfaz del programa con la base de datos será el ORM *Hibernate* versión 3.0 y el Sistema Gestor de Base de Datos (SGBD) será MySQL por cumplir las necesidades de requerimientos independiente de ser un software gratuito.

Interacción con servidor de aplicaciones web:

La aplicación a desarrollar está enfocada hacia un ambiente web. Para implementar especificaciones de *servlets* y JSPs es necesario un servidor de aplicaciones y un contenedor de *servlets*. Apache Tomcat es una herramienta gratuita y cumple con los requerimientos.

Interacción con la herramienta de RI:

El programa a desarrollar tendrá interacción con una herramienta de RI. Al momento de subir el código fuente del programa indexará la información necesaria para su posterior recuperación. La herramienta que cumple con las necesidades de los requerimientos es Apache Lucene 1.4.3 independientemente de ser una herramienta gratuita.

Interacción con herramienta para subir archivo al servidor:

Para que el software a desarrollar suba los archivos al servidor tendrá interacción con una herramienta capaz de subir archivos a un servidor. El usuario necesita mandar el programa al servidor para su almacenamiento, indexación y recuperación del mismo. La herramienta gratuita de Apache FileUpload 1.1 cumple con las necesidades de los requerimientos.

Interacción con herramienta para colorear el código:

La aplicación a desarrollar tendrá que ser capaz de generar una página html que contenga el código fuente colorado de cada uno de los programas que se encuentre en el repositorio. La herramienta Java2Html de Markus Gebhard además de ser gratuita cumple con las necesidades de los requerimientos.

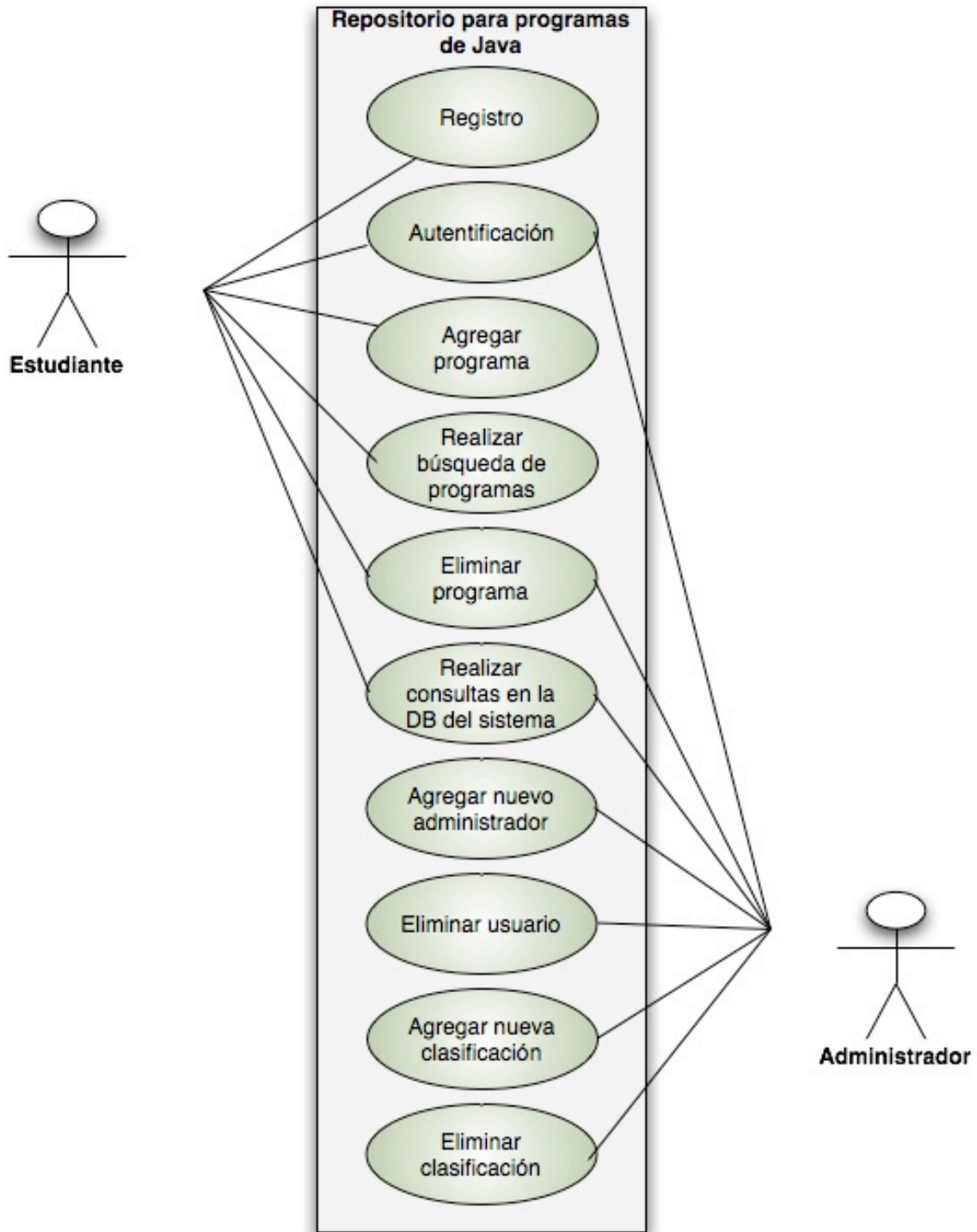
3.3 Especificación de requerimientos

Los casos de uso son una técnica para identificar los diferentes tipos de personas (o dispositivos) que utiliza el sistema o producto [Pressman, 2002]. Describen al sistema, la relación con otros sistemas y su entorno [Barrientos, 2003].

Los casos de uso tiene tres elementos: actores, burbujas de casos de uso y relaciones.

3.3.1 Diagrama casos de uso de alto nivel

Para detallar cada caso de uso se sigue la metodología propuesta por UML y Larman. A continuación (diagrama 3.1) se mostrará el diagrama de casos de uso que involucra a los dos tipos de actores: usuario estudiante y usuario administrador. Posteriormente se mostraran los escenarios para cada burbuja descrita en el diagrama.



3.1 Diagrama general de Casos de Uso

Caso de uso	Registro de nuevo estudiante	
Actores	Estudiante	
Propósito	Agregar un nuevo usuario Estudiante a la base datos del sistema	
Resumen	Cualquier estudiante que ingrese al sitio web puede realizar su registro y necesita llenar 4 campos para completar su registro.	
Tipo	Primario, esencial	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1. Este caso de uso comienza cuando un estudiante ingresa a la aplicación y elige la opción de registro.	2. El sistema entonces muestra una pantalla con una forma en la cual se solicita que complete los siguientes campos: nombre de usuario, contraseña, confirmación de contraseña, nombre(s), apellido(s).	
3. El usuario llena los campos y le da aceptar al sistema	4. El sistema entonces registra al usuario y lo manda a una pantalla donde le muestra el menú principal.	
Cursos Alternos		
<p>- Línea 3. El usuario no llena todos los campos obligatorios de la forma. El sistema muestra un mensaje de error en el que se indican que los campos estaban incompletos y se redirecciona a la página principal de la aplicación.</p> <p>- Línea 4. El sistema no puede realizar el registro del usuario debido a que eligió un nombre de usuario que ya existe. El sistema muestra un mensaje de error en el que se indica que el usuario ya existe que intente con otro y se redirecciona a la página principal de la aplicación.</p> <p>- Línea 4. El sistema no logra tener acceso a la base de datos del sistema o no logra completar el registro satisfactoriamente, se informa al usuario el error y se redirecciona a la página principal de la aplicación.</p>		

Caso de uso	Autenticación	
Actores	Estudiante, Administrador	
Propósito	Validar identidad de los usuarios del sistema.	
Resumen	Todos los usuarios del sistema deben de demostrar de alguna manera que son miembros registrados previamente en el sistema, antes de poder hacer uso del mismo, para esto, deben de realizar un proceso de autenticación.	
Tipo	Primario, esencial	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1. El estudiante o administrador ingresa al sistema. 3. El usuario llena los campos de la forma y manda su información al sistema.	2. El sistema muestra una pantalla de autenticación, en el cual el estudiante o administrador deben de proporcionar su nombre de usuario y contraseña para hacer uso del mismo. 4. Una vez enviada la información, el sistema busca en la base de datos los datos del usuario, una vez encontrados, despliega en la pantalla el menú principal que corresponda al tipo de usuario (estudiante, administrador).	
Cursos Alternos		
<p>- Línea 3. El usuario no llena todos los campos obligatorios de la forma. El sistema muestra una página de error con un mensaje en el cual se indica que los campos deben ser llenados y redirecciona a la página principal de la aplicación.</p> <p>- Línea 4. El sistema no encuentra la información proporcionada y no permite al usuario el acceso al sistema, notificando al usuario que verifique sus datos y lo redirecciona a la página principal de la aplicación.</p>		

Caso de uso	Agregar programa	
Actores	Estudiante	
Propósito	Agregar un nuevo programa a la cuenta de un usuario del sistema.	
Resumen	El estudiante podrá agregar nuevos programas al repositorio una vez que se haya registrado en el sistema.	
Tipo	Primario, esencial	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
<p>1. Este caso de uso comienza cuando un estudiante ya se ha autenticado en el sistema y elige la opción de mis programas.</p> <p>3. El usuario elige el archivo a subir y llena los campos de la forma y manda su información al sistema.</p>	<p>2. El sistema muestra una pantalla en donde le muestra los programas existentes relacionados con la cuenta del usuario, así como la forma necesaria para agregar un nuevo programa.</p> <p>4. El sistema registra la información relacionada con el programa en su base de datos, guarda el archivo y nuevamente muestra la lista de programas existentes y la forma necesaria para agregar un nuevo programa.</p>	
Cursos Alternos		
<p>- Línea 2. El sistema no logra completar las transacciones necesarias con la base de datos para mostrar la información de los programas y muestra un mensaje de error.</p> <p>- Línea 3. El usuario no elige ningún archivo o la extensión de este es diferente a .java. El sistema le notifica del error en el archivo.</p> <p>- Línea 3. El usuario elige un archivo que tiene errores de compilación. El sistema le muestra al usuario de los errores generados.</p> <p>- Línea 3. El usuario elige un archivo de un tamaño mayor a 25KB. El sistema le informa al usuario que no puede subir archivos mayores de 25KB.</p>		

Caso de uso	Realizar búsqueda de programas	
Actores	Estudiante	
Propósito	Realizar búsquedas en todos los programas de repositorio.	
Resumen	El estudiante podrá realizar búsquedas sobre los programas almacenados por todos los usuarios, pudiendo realizar ésta de dos formas: búsqueda en texto y búsqueda por clasificación.	
Tipo	Primario, esencial	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1. Este caso de uso comienza cuando un estudiante ya se ha autenticado en el sistema y elige la opción de buscar.	2. El sistema muestra una pantalla en donde se puede elegir algún criterio de búsqueda, sobre campos específicos o sobre alguna clasificación.	
3. El usuario elige la opción que desee y asigna los datos necesarios.	4. El sistema realiza la búsqueda con los datos asignados, muestra una pantalla con los programas que cumplen con los criterios de la consulta.	
Cursos Alternos		
<p>- Línea 3. El usuario en caso de elegir la búsqueda por texto y dejar en blanco las palabras a buscar. El sistema le notifica del error.</p> <p>- Línea 3. El usuario en caso de elegir la búsqueda por texto no elige en que campos se hará la consulta. El sistema por default busca las palabras en el código fuente de los programas.</p> <p>- Línea 4. El sistema no encuentra programas que coincidan con los datos asignados y notifica al usuario.</p>		

Caso de uso	Eliminar programa	
Actores	Estudiante, Administrador	
Propósito	Eliminar programas no deseados del repositorio.	
Resumen	El estudiante y el administrador podrán eliminar programas no deseados del repositorio. El sistema permite que cada estudiante elimine solo sus programas. Por otra parte el administrador puede eliminar programas de cualquier estudiante.	
Tipo	Primario, esencial	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
<p>1. Este caso de uso comienza cuando un estudiante ya se ha autenticado en el sistema y elige la opción de mis programas.</p> <p>3. El usuario elige los programas que desea eliminar y lo manda al sistema.</p>	<p>2. El sistema muestra una pantalla en donde le muestra los programas existentes relacionados con la cuenta del usuario, y una columna con un checkbox que permite marcar los programas para su eliminación.</p> <p>4. El sistema realiza las operaciones necesarias para la eliminación de los programas. Y le muestra la pantalla en donde se muestran sus programas existentes.</p>	
Cursos Alternos		
<p>- Línea 2. El sistema no es capaz de recuperar la información necesaria para mostrar al usuario sus programas, le notifica de lo ocurrido.</p> <p>- Línea 4. El sistema no pudo terminar el proceso de eliminación le informa al usuario del error.</p> <p>- Línea 4. El sistema nota que el usuario ha borrado todos sus programas. Le notifica que no tiene ningún programa.</p>		

Caso de uso	Realizar consultas de la base de datos del sistema	
Actores	Estudiante, Administrador	
Propósito	Obtener información detallada de los usuarios, clasificaciones y programas.	
Resumen	El estudiante puede obtener información sobre los programas que actualmente tiene y pueden ser visualizados en formato html. Por otra parte el administrador puede obtener información sobre todos los usuarios, las clasificaciones y programas.	
Tipo	Primario, esencial	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1. Este caso de uso comienza cuando un estudiante ya se ha autenticado en el sistema y elige la opción de mis programas o el administrador ya autenticado elige alguna opción en el menú principal (usuarios, programas, clasificaciones).	2. El sistema muestra la pantalla con la información solicitada a manera de lista.	
Cursos Alternos		
- Línea 2. El sistema no es capaz de recuperar la información necesaria para mostrar al usuario la información solicitada (programas, clasificaciones, usuarios). El sistema notifica lo ocurrido.		

Caso de uso	Agregar nuevo administrador	
Actores	Administrador	
Propósito	Agregar un nuevo usuario administrador	
Resumen	Un administrador en el único tipo de usuario capaz de dar de alta un nuevo administrador. Para hacerlo se requieren los datos del nuevo usuario.	
Tipo	Primario, esencial	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1. Este caso de uso comienza cuando un administrador previamente autenticado ha elegido la opción de usuarios desde el menú principal.	2. El sistema muestra entonces una pantalla con los usuarios actuales y una forma para que el administrador ingrese los datos necesarios para agregar al nuevo usuario.	
3. El administrador entonces llena los campos necesarios y manda la información del nuevo usuario a la base de datos del sistema.	4. El sistema realiza las transacciones necesarias con la base de datos para agregar el nuevo usuario a la base de datos. Y muestra una lista con todos los usuarios que actualmente existen.	
Cursos Alternos		
- Línea 3. El administrador no llena todos los campos obligatorios de la forma. El sistema informa al usuario del error en el que se indican los campos que deben ser llenados y le muestra la lista de usuarios existentes.		
- Línea 4. El sistema no logra tener acceso a la base de datos del sistema o no logra completar todas las transacciones necesarias para dar de alta al usuario, muestra un mensaje de error.		

Caso de uso	Eliminar usuario	
Actores	Administrador	
Propósito	Eliminar de la base de datos del sistema a un usuario que no se desee o no requiera usar el sistema. Puede ser administrador o estudiante.	
Resumen	Un administrador es el único tipo de usuario capaz de dar de baja a otros usuarios y con ello borrar todos los programas del mismo.	
Tipo	Primario, esencial	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
<p>1. Este caso de uso comienza cuando un administrador previamente autenticado ha elegido la opción de usuarios desde el menú principal.</p> <p>3. El administrador selecciona los usuarios a eliminar y manda los datos para que el sistema los dé de baja del sistema.</p>	<p>2. El sistema muestra entonces una pantalla con los usuarios actuales y junto de estos un check box que se puede marcar en caso de querer eliminar dicho usuario de la base de datos del sistema.</p> <p>4. El sistema realiza las transacciones necesarias con la base de datos para eliminar al usuario y sus programas de la base de datos, cuando termina muestra una lista con todos los usuarios que actualmente existen.</p>	
Cursos Alternos		
- Línea 4. El sistema no logra tener acceso a la base de datos del sistema o no logra completar todas las transacciones necesarias para eliminar usuario.		

Caso de uso	Agregar nueva clasificación	
Actores	Administrador	
Propósito	Agregar una nueva clasificación a la base de datos.	
Resumen	Un administrador en el único tipo de usuario capaz de dar de alta nuevas clasificaciones en el sistema. Para hacerlos, se requieren los datos de la clasificación a dar de alta	
Tipo	Primario, esencial	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1. Este caso de uso comienza cuando un administrador previamente autenticado ha elegido la opción de clasificaciones desde el menú principal.	2. El sistema muestra entonces una pantalla con las clasificaciones actuales y una forma para que el administrador ingrese los datos necesarios para agregar una nueva clasificación.	
3. El administrador entonces llena los campos necesarios y manda la información de la nueva clasificación a la base de datos del sistema.	4. El sistema realiza las transacciones necesarias con la base de datos para agregar la nueva clasificación a la base de datos. Y muestra una lista con todas las clasificaciones que actualmente existen.	
Cursos Alternos		
- Línea 3. El administrador no llena todos los campos obligatorios de la forma. El sistema informa al usuario del error en el que se indican los campos que deben ser llenados y le muestra la lista de clasificaciones existentes.		
- Línea 4. El sistema no logra tener acceso a la base de datos del sistema o no logra completar todas las transacciones necesarias para dar de alta la clasificación, muestra un mensaje de error.		

Caso de uso	Eliminar clasificación	
Actores	Administrador	
Propósito	Eliminar de la base de datos del sistema una clasificación que no se desee.	
Resumen	Un administrador en el único tipo de usuario capaz de dar de baja clasificaciones y con ello borrar los programas pertenecientes a ésta.	
Tipo	Primario, esencial	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1. Este caso de uso comienza cuando un administrador previamente autenticado ha elegido la opción de clasificaciones desde el menú principal.	2. El sistema muestra entonces una pantalla con las clasificaciones actuales y junto a estas un check box que se puede marcar en caso de querer eliminar dicho clasificación de la base de datos del sistema.	
3. El administrador selecciona las clasificaciones a eliminar y manda los datos para que el sistema los dé de baja del sistema.	4. El sistema realiza las transacciones necesarias con la base de datos para eliminar la clasificación y sus programas de la base de datos, cuando termina muestra una lista con las clasificaciones que actualmente existen.	
Cursos Alternos		
- Línea 4. El sistema no logra tener acceso a la base de datos del sistema o no logra completar todas las transacciones necesarias para eliminar la clasificación.		

3.4 Especificación de proceso.

Los diagramas de flujo de datos (DFD) permite desarrollar los modelos del ámbito de información y funcional al mismo tiempo [Pressman, 2004]. En el Apéndice A se muestra el

nivel 0 mostrando las entradas y salidas principales del sistema. En el nivel 1 se muestran los procesos más importantes.

3.5 Descripción de los objetos de dato

A continuación se da una descripción general de los objetos de datos y de sus atributos que se identificaron:

Clasificación:

Una clasificación ayudará a mantener una organización en los programas. Los siguientes son los atributos de una clasificación:

- Nombre: El nombre de la clasificación.
- Padre: La clasificación superior.
- Hijas: La lista de clasificaciones que dependen de esta clasificación.
- Programas: La lista de programas que pertenecen a esta clasificación.
- Descripción: La descripción de la clasificación

Programa:

Un programa es el principal componente del sistema. Los siguientes son los atributos de un programa:

- Nombre: El nombre del programa.
- Descripción: La descripción del programa.
- Fecha: La fecha de almacenamiento del programa.

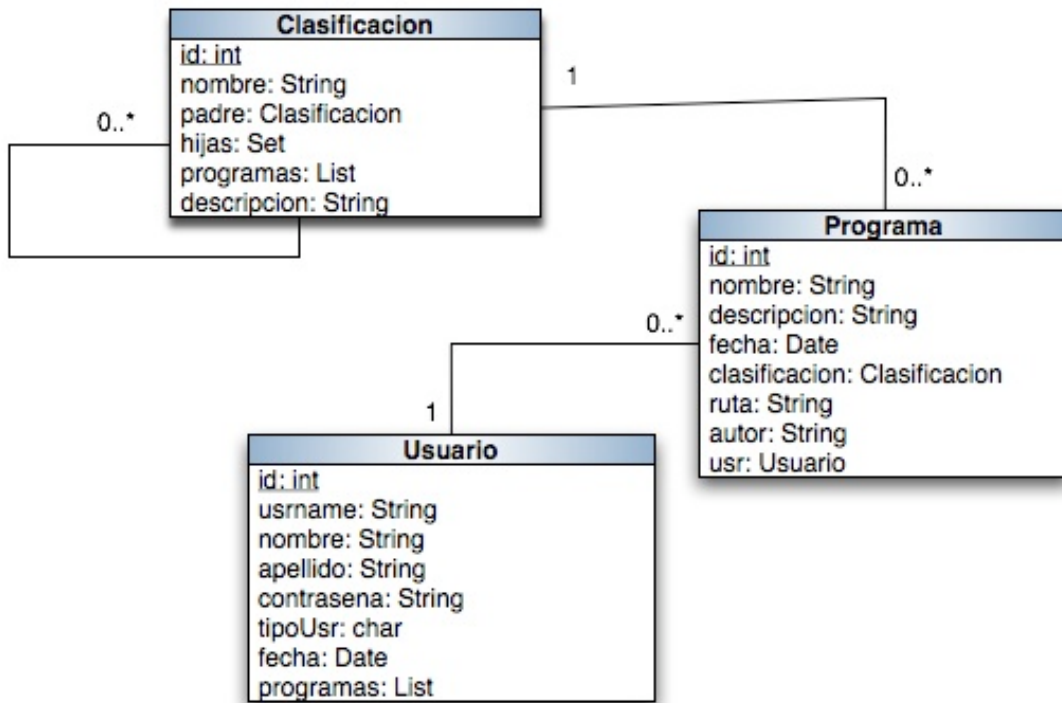
- Clasificación: La clasificación a la que pertenece el programa.
- Ruta: La ruta física en la cual se almacenó el programa.
- Autor: El autor real del programa.
- Usuario: El usuario que almaceno el programa.

Usuario:

Un usuario es la persona que realiza alguna acción sobre el sistema, ya sea guardar, ver o borrar programas, crear, ver y borrar clasificaciones o usuarios. Los siguientes son los atributos de un usuario:

- Nombre de usuario: El nombre que elige el usuario al suscribirse al sistema.
- Nombre: El nombre real del usuario.
- Apellido: El apellido del usuario.
- Contraseña: La contraseña que elige el usuario al suscribirse al sistema.
- Tipo de usuario: El tipo de usuario, puede ser de dos tipos: estudiante y administrador.
- Fecha: La fecha en la que se registra el usuario al sistema.
- Programas: La lista de programas que el usuario ha guardado en el sistema.

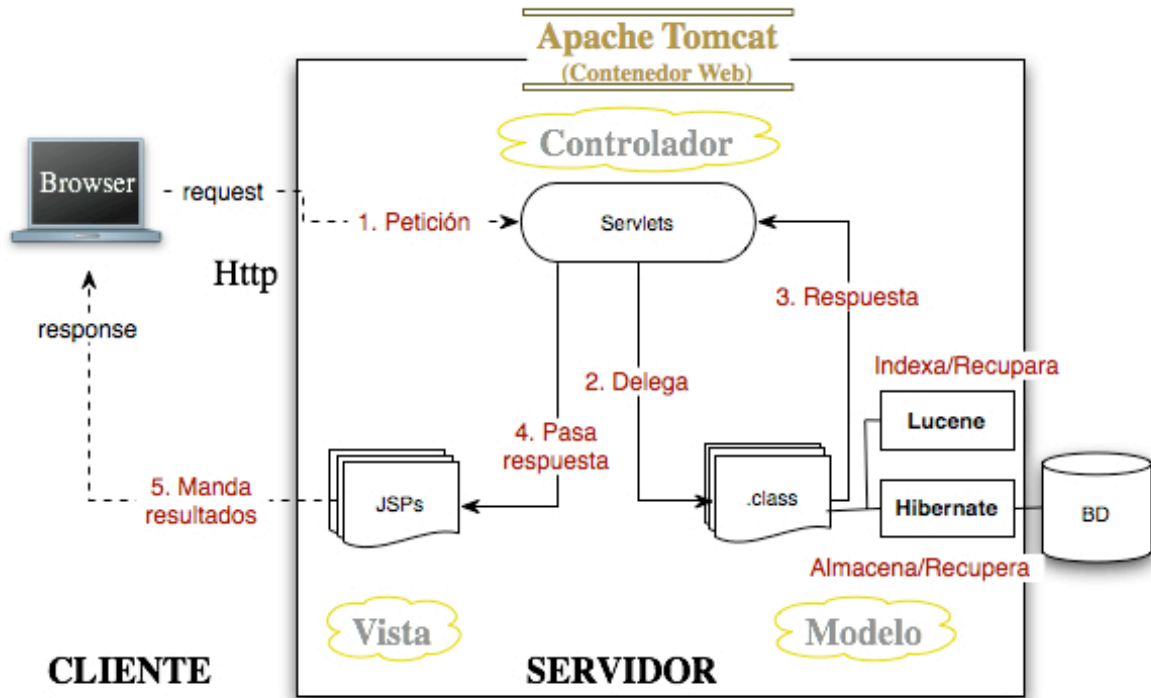
En el diagrama 3.2 se muestra el diagrama de clases obtenido a partir de los tres objetos de datos identificados, los respectivos atributos que le corresponden a cada uno y las relaciones que existen entre ellos. Se muestra la llave primaria subrayada en cada caso.



3.2 Diagrama de clases

3.6 Arquitectura.

La arquitectura que tendrá la aplicación será cliente-servidor (Véase Figura 3.3), siguiendo el patrón de diseño *Model-View-Controller* (MVC). Por lo que el cliente (estudiante, administrador) hará una petición a través de la vista al servidor web, el cual recibirá la petición mediante el controlador, y este último será el encargado de delegar al modelo el trabajo necesario para responder a la petición que hizo el cliente, cuando el modelo tenga la respuesta a la petición delegada. Dentro del modelo se encuentra la el ORM *Hibernate* así como el API *Lucene* encargado del manejo del archivo Java. Por su parte el controlador tomará la respuesta y la mostrará al cliente a través de la vista. El patrón es aplicado estricta y únicamente para las clases que involucran la interacción con el ORM *Hibernate*.



3.3 Diagrama de la arquitectura del sistema

(Adaptado de [Marín, 2004])