 Universidad de La Frontera

Proyecto

Libro de Clases

Alumno: Patricio Gutiérrez

Carrea: Ingeniería civil Informática

Profesor: Víctor Aravena

# Problema

Se desea implementar un sistema que permita emular un libro de clases de un colegio, el cual debe poder ingresar planificaciones, notas, asistencias y anotaciones, además de poder vincular los cursos con cada profesor y alumno y los alumnos con sus respectivos apoderados.

Además, este sistema debe generar reportes de distinta índole y cada uno debe generarse en varios formatos.

# Análisis

Primeramente se deberá crear un archivo inicial el cuál contendrá datos para comenzar a trabajar.

Se debe crear una clase por cada tipo de dato solicitado. Todas las clases deben estar enlazadas de alguna forma (herencia, asociación en sus distintos tipos).

También es necesario crear una interfaz gráfica en la cual se puedan mostrar, acceder y modificar los distintos datos contenidos en el archivo inicial.

En cuanto a los reportes, se deben generar archivos que permitan las transformaciones necesarias para obtener cada uno de estos en los distintos formatos pedidos y almacenarlos separados.

# Diseño

Para comenzar, se crean las clases basándose en los siguientes sustantivos:

* Profesor.
* Alumno.
* Clase.
* Asignatura.
* Anotación.
* Evaluación.
* Apoderado.

Cada una de estas clases se vinculan entre sí pensado de esta forma:

* Un Apoderado tiene un Alumno.
* Una Asignatura la dicta un Profesor.
* Un Alumno puede tener muchas Anotaciones.
* Un Alumno rinde una o más Evaluaciones por Asignatura.
* Una Clase se compone de muchos Alumnos y un Profesor por Asignatura.
* Una Clase tiene 5 Asignaturas.

Debido a la naturaleza de los cursos, se utilizan clases de tipo enum las cuales son NivelCurso y ModuloCurso. La primera clase contiene los nombres de cada curso, los que van desde primero hasta octavo. En cuanto a la clase ModuloCurso, ésta solo contiene los caracteres “A” y “B”.

Para las anotaciones, tenemos la clase enum llamada TipoAnotacion, la cual nos dirá si la anotación escrita a un alumno es positiva o negativa.

En el caso de las Asignaturas, como éstas son comunes para todos los cursos, la clase enum contendrá las siguientes asignaturas: Ingles, lenguaje, matemáticas, historia y ciencias.

Estas clases enum serán útiles a la hora de desplegar la información en las interfaces gráficas.

En cuanto a los reportes, se crearon clases que serán de ayuda para generarlos, puesto que cada una tendrá la información solicitada para su respectivo reporte, por ejemplo, para el informe de planificación por alumno, sólo se necesita extraer los datos principales de éste, tales como nombre, apellido, run, apoderado y obviamente todas sus planificaciones, no siendo necesario las notas, asistencia y los demás datos que posee. Por este motivo se crean clases llamadas *Alumno…*, las cuáles contienen como atributo sólo los datos necesarios para cada reporte.

Para las interfaces gráficas, se necesitan tener 4 ventanas.

La ventana principal mostrará los cursos, módulos, profesores y alumnos junto a su apoderado. En esta ventana se cargarán los enum nombrados anteriormente.

Tendremos 2 JComboBox en los cuales se podrán seleccionar el curso y el módulo. Al cambiar alguno de éstos, se cargarán en los demás campos los alumnos y profesores correspondientes. A la derecha tendremos un botón para instanciar la ventana de asistencia.

Luego tendremos otro JComboBox que poseerá las asignaturas del curso.

Al lado de éste tendremos un botón para abrir la ventana de planificación.

También habrá otro JComboBox que cargará los run de cada alumno del curso seleccionado. Hacia la derecha tendremos campos de textos en los que se mostrará el nombre del alumno y abajo el nombre del apoderado. Además se tendrá un botón para abrir la ventana para agregar anotaciones.

Finalmente se tendrán los siguientes botones: botón para abrir la ventana de reportes, para guardar los cambios realizados y para cerrar el libro de clases.

Para la ventana de asistencia, se seleccionará un alumno por run y habrá un panel para marcar con un tick si es que éste asistió.

En la ventana de planificación se podrá ver el nombre del profesor junto a la asignatura de la cual está encargado, además de un comboBox con la cantidad de planificaciones y abajo un área de texto para anotar el plan.

En la ventana de anotaciones habrá un comboBox que cargará el enum TipoAnotacion explicado anteriormente y un área de texto para anotar el motivo.

En cuanto a la ventana de reportes, ésta contendrá un botón por cada tipo de reporte solicitado, además de un jtextfield para generar el reporte por porcentaje de asistencia.

Para cada reporte es necesario generar un archivo xsl que transforme éstos en los diferentes formatos solicitados, por lo que existe también una clase que toma el archivo xsl y xml de entrada para generar un archivo de salida.

Cada uno de los reportes se separa por carpetas, agrupándolos por tipo de reporte. Los xsl estarán en una carpeta aparte y también agrupados por tipo de reporte.

# Conclusión

Para comenzar a crear el proyecto, es necesario basarse en el modelo UML solicitado, debido a que es necesario abstraerse del problema para ver las relaciones existentes entre las clases que se crearon.

El libro de clases logró llevar a cabo lo solicitado en cuanto a visualiacion, ingreso y modificación de información, guardándola, leyéndola y además generando los distintos formatos para presentar la información necesaria.

En cuanto a los reportes, es necesario generar los respectivos archivos xsl para que las transformaciones sean hechas automáticamente.