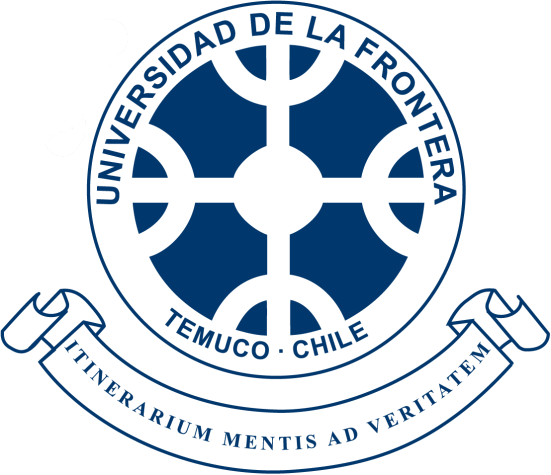
Informe Proyecto

Programación



Nombre: Diego Garrido

Profesor: Victor Aravena Diaz

# Índice

Introducción 3

Planteamiento del problema 4

Análisis del problema 6

Diseño del Software 7

Uml

Datos

Ventanas

Versión Web del proyecto 12

Conclusión / Comentarios finales 20

# Introducción

Los softwares hechos para el manejo de datos se han masificado y son usados en prácticamente todos los rubros y formas de desenvolvimiento de la sociedad, sería imposible pensar en el manejo de gigantes bases de datos de forma manual o la inexistencia siquiera de centros de datos que son el pilar clave de tantas industrias. La educación no es una excepción, ya que cada vez se hace más necesario un sistema de manejo de información que sea rápido, transparente y accesible a todos los interesados en cada uno de los establecimientos que imparten educación.

Por lo anterior es que surgió la necesidad en un colegio de establecer un sistema que les permitiera el manejo de la información cotidiana que se obtiene en el transcurso del periodo académico y, en el presente informe se busca entregar un sistema que dé solución a los múltiples problemas que se les han presentado.

# Planteamiento del problema

Un colegio, que posee una población importante de alumnos, profesores y apoderados presenta un problema para el manejo de los datos del establecimiento. Por lo anterior requiere de un sistema de administración en el cual se simule o se mantenga la información correspondiente al contenido de los libros de clases. Que le permita un manejo fácil y rápido de la información importante para el transcurso de las jornadas académicas en el establecimiento y que le permita informar transparentemente como oportunamente a los apoderados de la situación de cada uno de sus pupilos, no solo la situación final de aprobación o reprobación, sino también que genere informes de las situaciones que requieran de más especial atención.

Para lo cual se han recogido los siguientes requerimientos:

* El colegio está conformado por cursos desde primero hasta octavo básico, divididos en dos letras por nivel, A y B, los cuales tienen una matrícula mínima de 30 cupos por curso.
* Cada uno de los 16 cursos pertenecientes al colegio, tiene 5 asignaturas obligatorias mínimas, y un profesor asignado a cada una de ellas.
* Una asignatura está compuesta por una planificación con 10 actividades y 5 notas en total.
* La asistencia a clases es fundamental, por ello cada uno de los estudiantes tiene una asistencia de 30 registros como mínimo.
* Los apoderados pueden tener más de un pupilo en el establecimiento, y cada uno de los alumnos tiene un apoderado responsable.
* Los alumnos pueden tener anotaciones tanto positivas como negativas que deben ser informadas a los apoderados, junto con el informe de notas y de asistencia a las clases.
* Cada uno de los profesores designados a las asignaturas debe poder ingresar la planificación de las mismas, para lograr tener un orden y una estructura de qué contenidos son los fundamentales que aprendan los niños en el establecimiento.

Se necesita además que se generen distintos informes, para cada uno de los interesados, entre los que se encuentran:

* Apoderados, quienes requieren saber la situación actual de cada uno de sus pupilos, cual es el resultado de las notas obtenidas, si ellos están aprobando o no el curso, cuando están planificadas sus evaluaciones, si ellos poseen anotaciones.
* Profesores, deben conocer el promedio de nota de cada uno de los alumnos perteneciente a los cursos donde impartan una asignatura.
* Un informe que permita conocer el porcentaje de asistencia de cada uno de los alumnos. Y, con ello saber qué estudiante tiene un bajo porcentaje de asistencia.
* Informes que permitan que los estudiantes conozcan su situación de aprobación o reprobación del curso, si es por % de asistencia o por promedio de notas.
* Un informe que liste a los apoderados y que permita conocer cuáles de ellos tiene más de un pupilo en el establecimiento.
* Finalmente, un informe para que los apoderados conozcan la planificación de las actividades que desarrollarán durante el periodo académico sus pupilos.

# Análisis del problema

Para comenzar, se puede extraer del planteamiento del problema las Clases principales que harán el trabajo de simular los “objetos” de la vida real que utilizaremos en el diseño, de acuerdo a la programación Orientada a Objetos.

Estas clases son: Alumno, Apoderado, Profesor, Curso, Asignatura.

Para el desarrollo del proyecto se pide tener un poblamiento de datos, las características de estos datos también nos pueden ayudar al análisis de los datos del problema.

Debemos añadir “características” y “relaciones” de la vida real entre estos objetos, que nos pueden ser útiles para desarrollar y solucionar el problema con el software, a las Clases que establecimos previamente.

Algunas podrían ser los nombres, para el caso de Apoderado, Alumno, Profesor y asignatura.

Para el caso de la clase Curso vendría de ayuda el identificador, es decir el nivel y la letra, si es que hay más cursos del mismo nivel.

La clase alumno debe tener un identificador para el Curso al que pertenece.

Una asignatura debe tener su planificación, por lo que necesita un atributo que le permita almacenar este dato, también debe tener un profesor que la enseñe, estableciendo así la relación Asignatura-Profesor.

Un profesor debe hacer clases por lo menos a un curso, por lo que debe tener una relación con Curso.

Los alumnos también deben tener cada uno sus Notas y registros de Asistencia, además de estar relacionado con un Apoderado, el cual puede tener más de un hijo en el colegio.

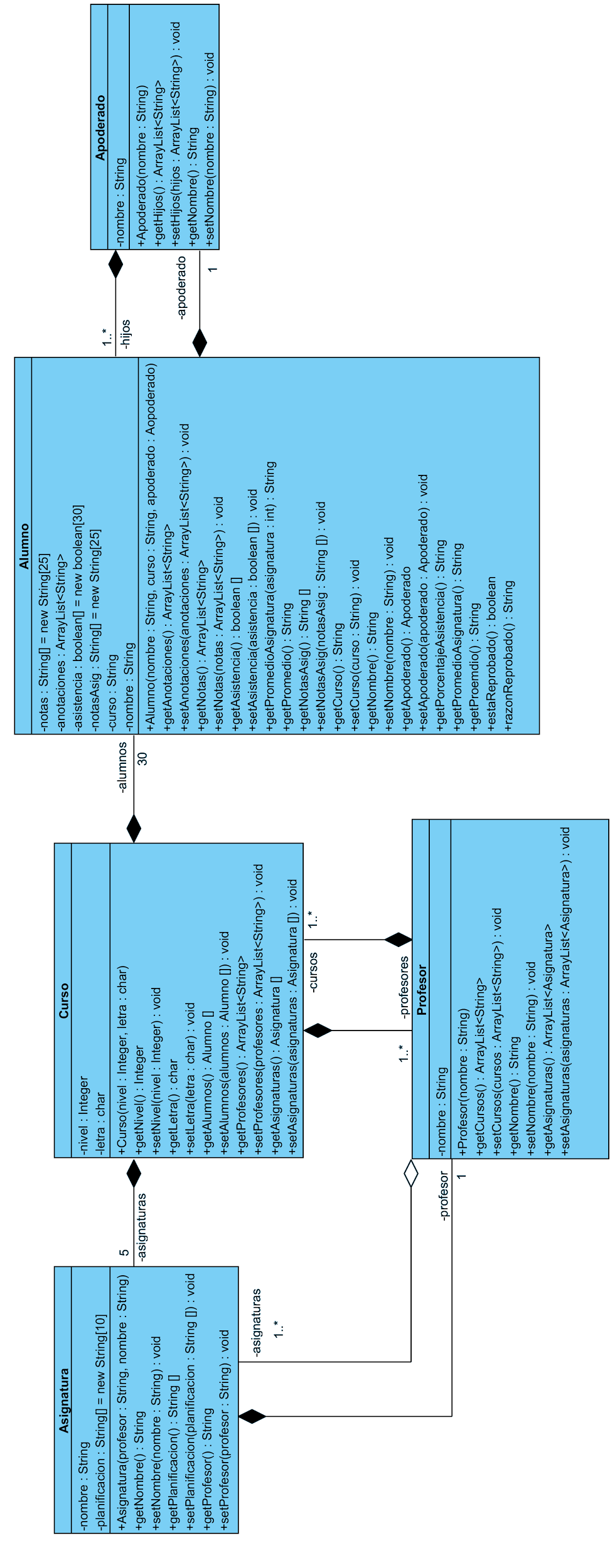
Y por último un Curso necesita tener asignaturas que sean impartidas a sus alumnos.

Se menciona establecer un estado de reprobación de los alumnos, es decir si pasa o no al siguiente curso. Para definir este estado se tomará como requisito para reprobar los mismos utilizados en los colegios básicos de Chile: tener un porcentaje de asistencia menor al 85%. Tener dos asignaturas con promedio rojo y un promedio general menor a 4.5. Tener tres asignaturas con promedio rojo.

También, en las especificaciones del poblamiento de datos se solicita que cada asignatura tenga diez actividades en su planificación, pero no especificaba el detalle de éstas, por lo que se tomará de la siguiente forma: cinco de esas actividades serán las notas que se toman en cuenta para el promedio de la asignatura, estas tienen cada una su ponderación a elección del usuario, de las cinco notas, cuatro son pruebas de la asignatura y la quinta es un promedio de las últimas cinco notas que vendrían a ser talleres o pequeños controles, todos con la misma ponderación.

El promedio general del alumno se toma con la misma ponderación para todos los promedios de cada asignatura.

# Diseño del Software

Luego de haber analizado el problema y los requisitos del poblamiento de datos podemos distribuir las clases y los atributos identificados en un Diagrama de Clases en UML, Poniendo especial énfasis en las relaciones y sus multiplicidades.

Para no perder los datos ingresados debemos realizar un grabado en el disco duro del computador con los atributos de los objetos, para esto crearemos un Package nuevo con clases que se encargarán de la administración de datos y ficheros.

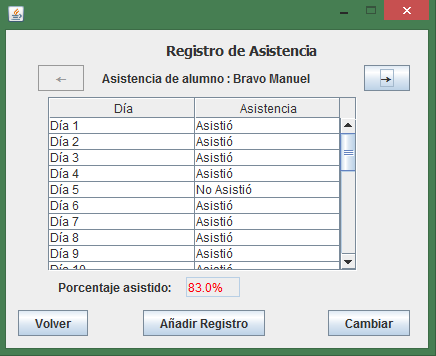
Para la administración de datos se utilizarán archivos en formato Json y Xml, ya que son muy útiles a la hora de administrar y ordenar datos de los atributos de objetos en Programación orientada a Objetos.

Además debemos realizar un poblamiento de datos, para lo que se creará una Clase externa al programa, la cual generará objetos de tipo Curso y los rellenará con Alumnos, Apoderados y Profesores de Nombres aleatorios y a los Alumnos les proveerá de notas y Registros de asistencia, también aleatorios. Los alumnos tienen su apoderado Respectivo, con el mismo apellido, si hay dos alumnos con un mismo apellido son asignados al mismo apoderado.

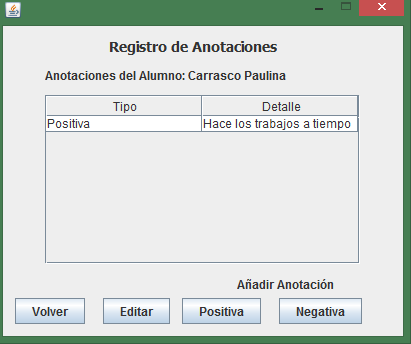
Para finalizar debemos crear un Front End para que el usuario interactúe con los datos y los modifique a su gusto, las Ventanas serán almacenadas en otro Package para mantener un orden entre Front End y Back End.

La ventana Curso es la ventana Principal y es la que mostrará a los alumnos en el curso seleccionado por el usuario y dará el paso a las ventanas que modificarán los datos de éstos. Además tiene los botones para agregar un Nuevo alumno al curso y Generar nuevo colegio, este último borra la base de datos y crea nuevas tablas con información generada aleatoriamente (nombres, notas y asistencia).

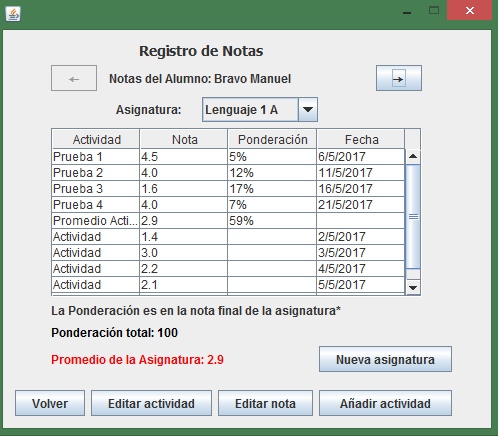
Si se selecciona un alumno de la lista y se presiona en el Botón Asistencia se abrirá la ventana Asistencia, que permitirá ingresar los días que el alumno asistió a clases y permite añadir un día al registro de todo el curso, también muestra el porcentaje de asistencia del alumno, si el porcentaje está en rojo significa que el alumno está en peligro de repitencia (asistencia bajo 85%).



Si en la ventana Curso se selecciona un alumno y se presiona en Anotaciones, se abrirá una ventana que permite añadir una anotación, Positiva o Negativa, muestra las anotaciones actuales y permite Editarlas en caso de que haya habido un error al ingresarla.

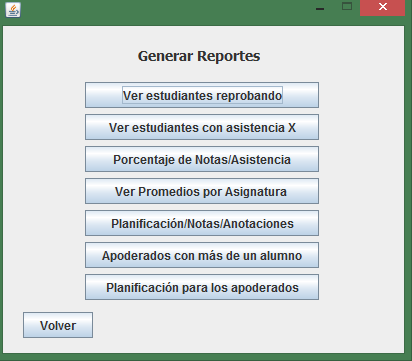


Si en la ventana Curso se Presiona el botón Notas, se abrirá una ventana mostrando las notas del alumno seleccionado. Permitiendo navegar entre las distintas asignaturas que está cursando el alumno, editar las notas y editar las actividades junto con agregar una actividad nueva o incluso una nueva asignatura.

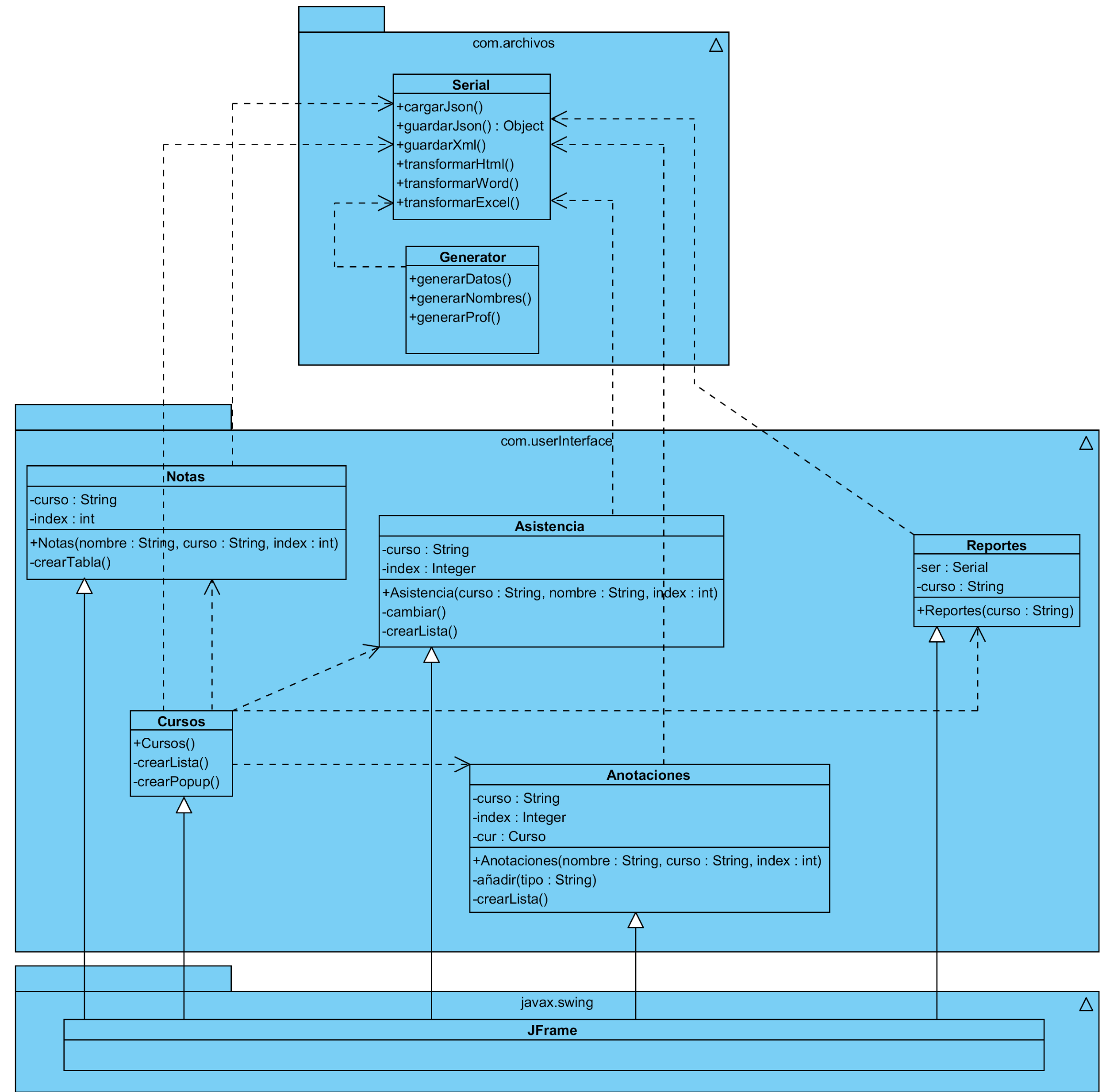


Por último al presionar el botón Reportes se abrirá una ventana con las opciones de guardar los datos solicitados en el planteamiento del problema, en formatos Json, Xml, Word, Excel y Html.

Al seleccionar una opción se guardan automáticamente los cuatro documentos en la Carpeta “reportes” correspondiente al curso anteriormente seleccionado.



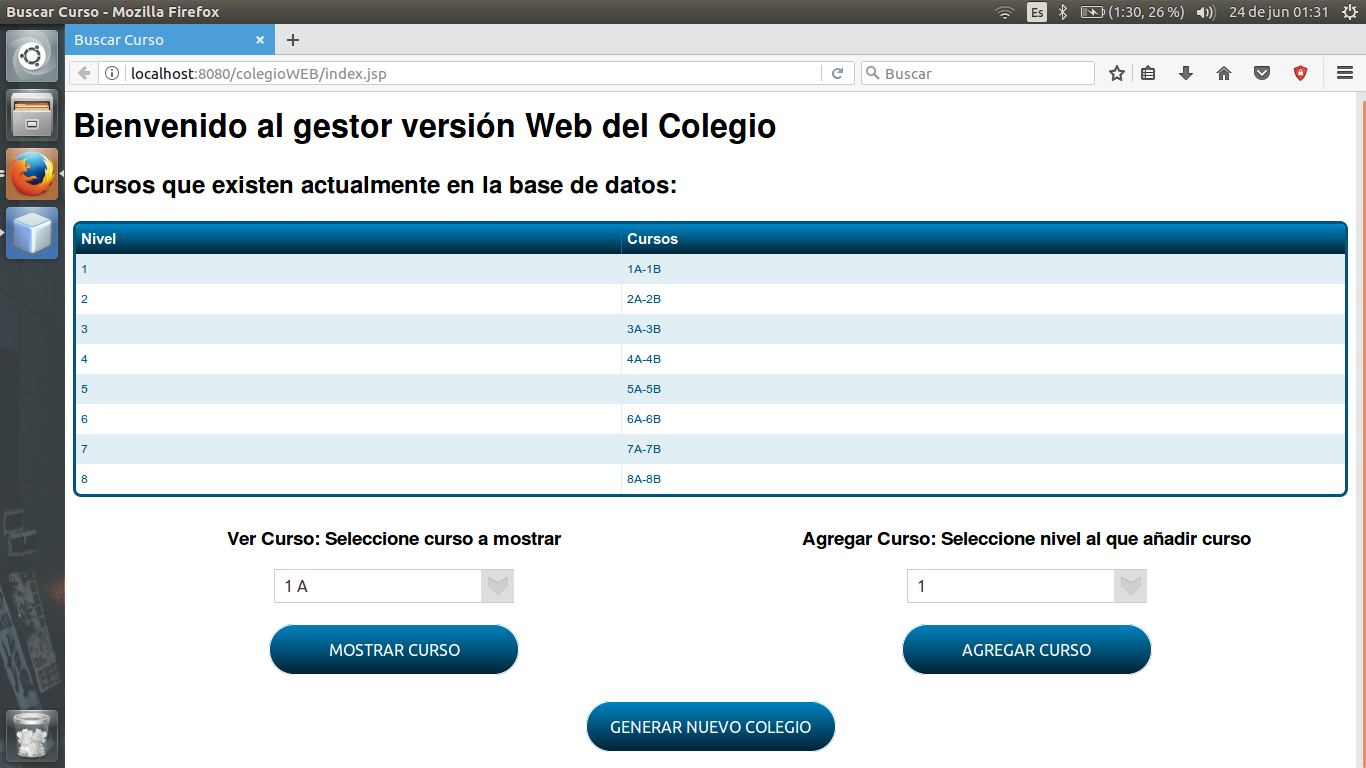
Todas estas ventanas interactúan con la Clase administradora de archivos, es utilizada para guardar automáticamente cualquier cambio en los datos que el usuario realice.



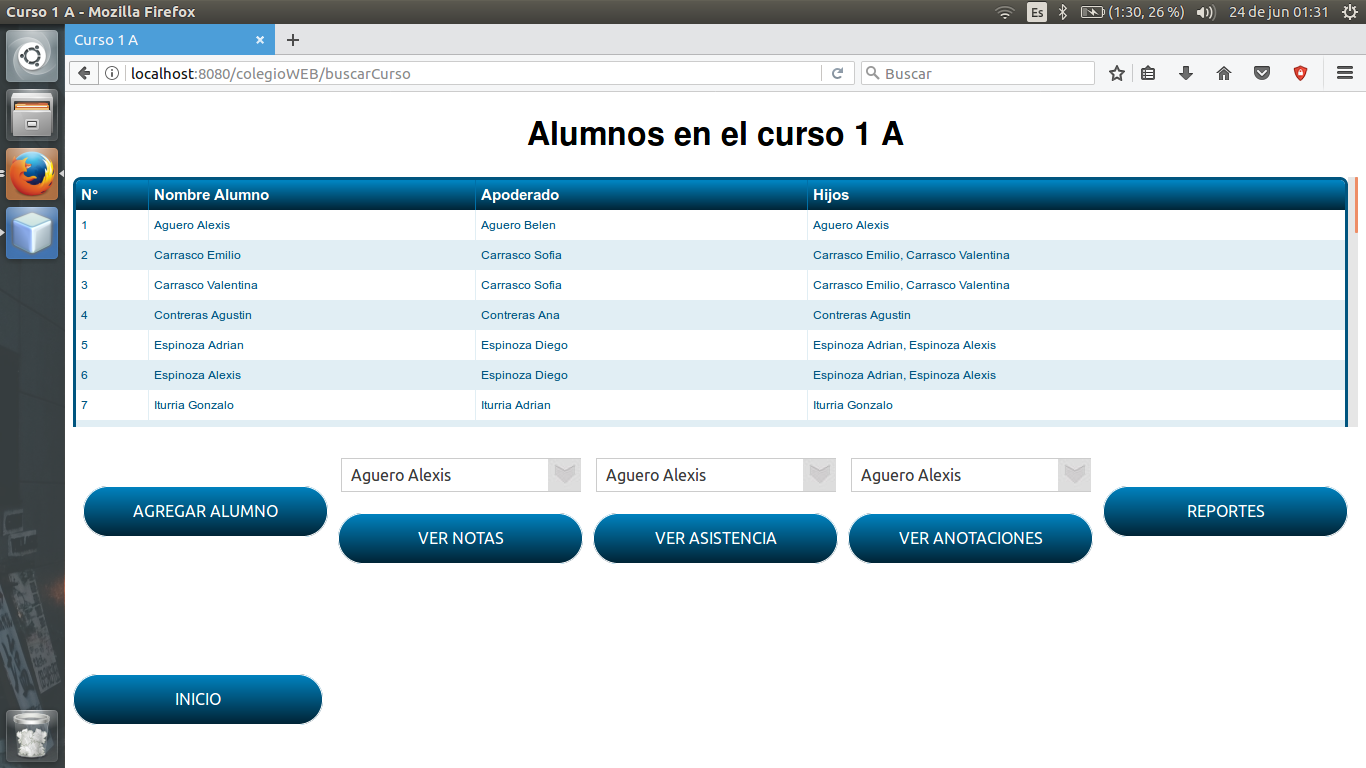
Versión Web

Lo que realizamos fue abordar la problemática mencionada al principio, gracias a la abstracción de las características que debía tener el sistema y las necesidades que debía suplir, para construir ahora un homólogo de la aplicación swing, pero ahora como un sistema Web alojado en un servidor (en nuestro caso GlassFish) aplicando el uso de HTML y la tecnología aportada por los JSP para crear páginas más dinámicas, que cambian según las acciones o solicitudes del usuario y la información que se tenga que desplegar.

Esta es la página principal, en donde se puede revisar en un primer vistazo todos los cursos existentes como también cuántos cursos existen por nivel. Además de eso también se puede añadir un curso, seleccionar un curso a revisar, o en último caso generar un colegio con los parámetros y requerimientos que fueron solicitados como población inicial

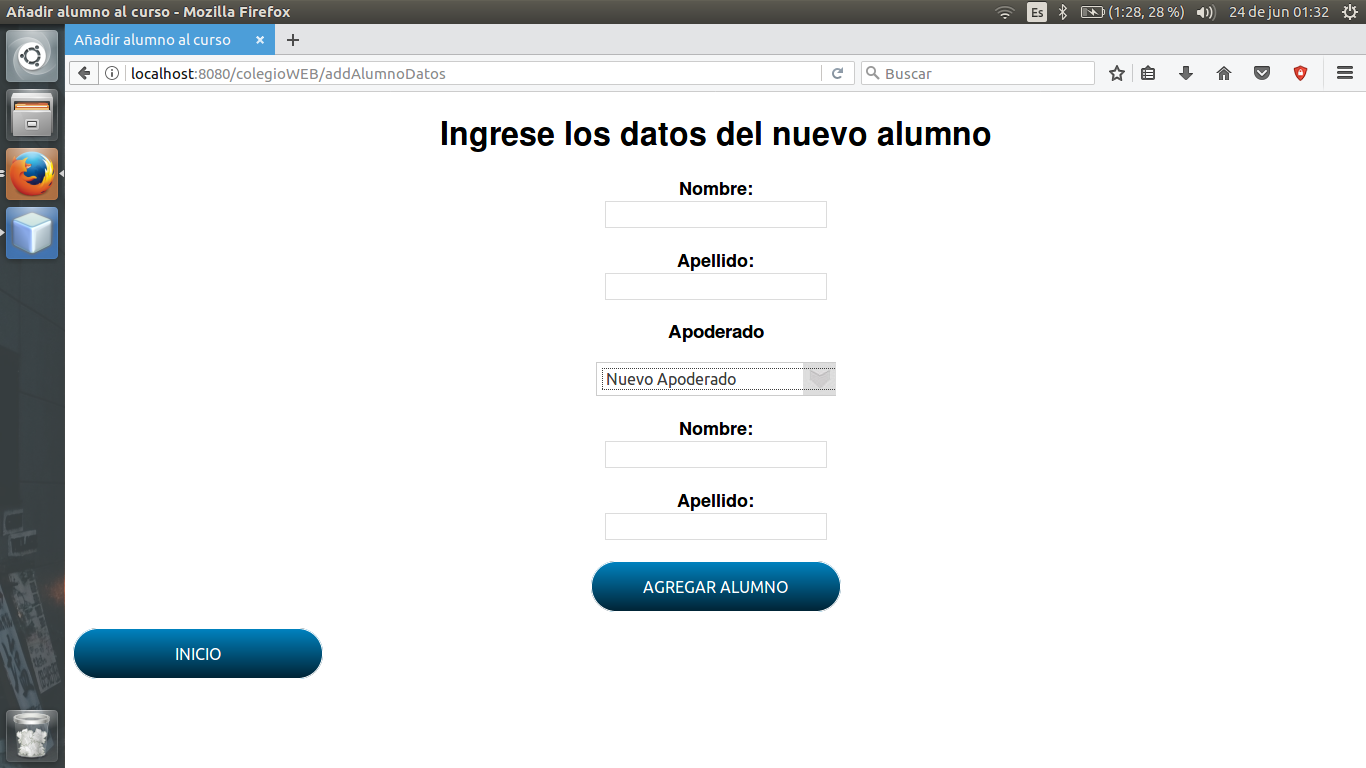


Una vez seleccionado el curso que se desea revisar/mostrar pasamos a la siguiente página, en la cual se puede revisar los alumnos registrados en el curso seleccionado, el nombre del apoderado del alumno registrado, como también los otros pupilos que tiene a su cargo ese mismo apoderado. Además de eso se puede añadir un alumno al curso, seleccionar un alumno para revisar sus notas, asistencia o anotaciones como también se puede acceder al apartado para solicitar reportes desde el botón de esta página.

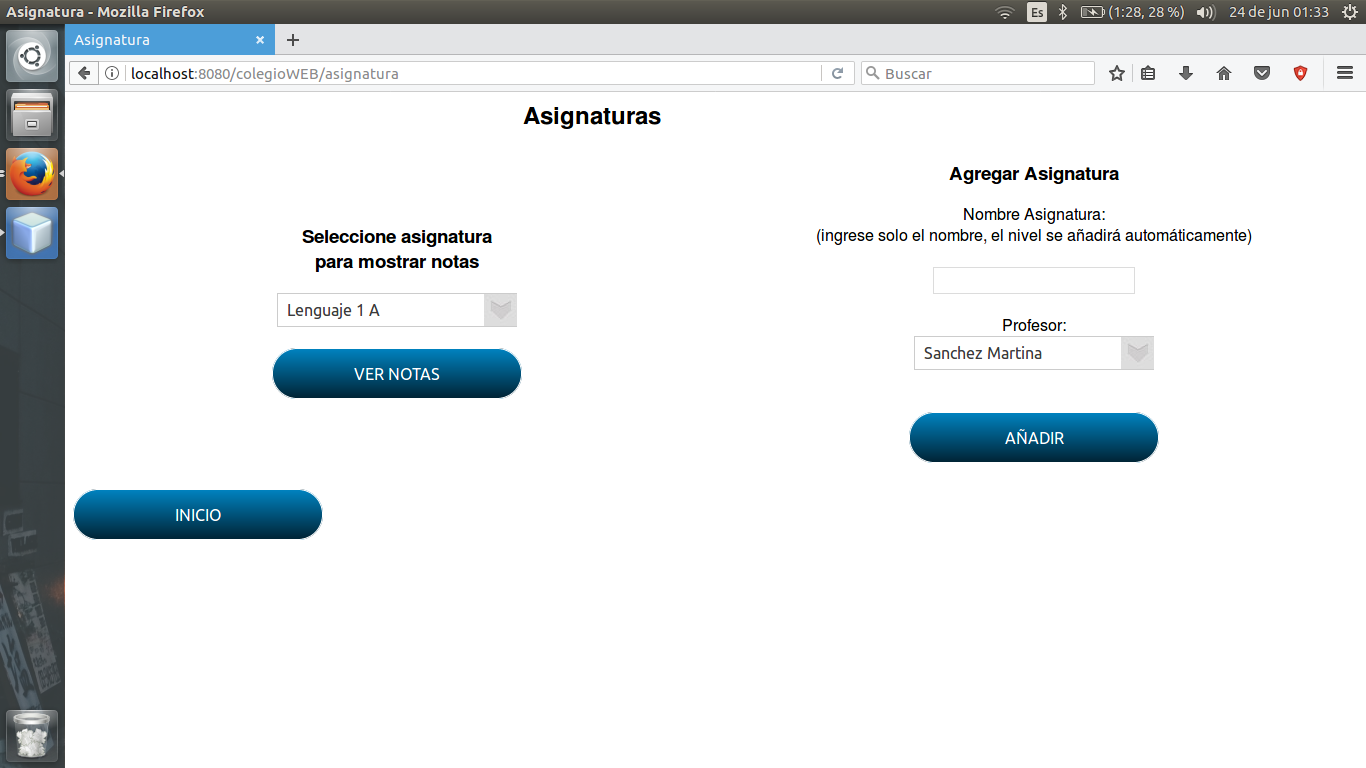


En el caso de apretar “Agregar alumno” se desplegará en una posterior página el formulario en el cual el usuario deberá completar los campos de nombre y apellido para crear al nuevo alumno que desee registrar en el curso, además de eso también deberá elegir el apoderado al que va a ser registrado (si es que ya existe) en caso de que no se encuentre registrado eligiendo la opción de “Nuevo apoderado” dinámicamente gracias a JavaScript se desplegarán los campos que debe rellenar con el nombre y apellido del nuevo apoderado que se desee registrar junto al alumno.

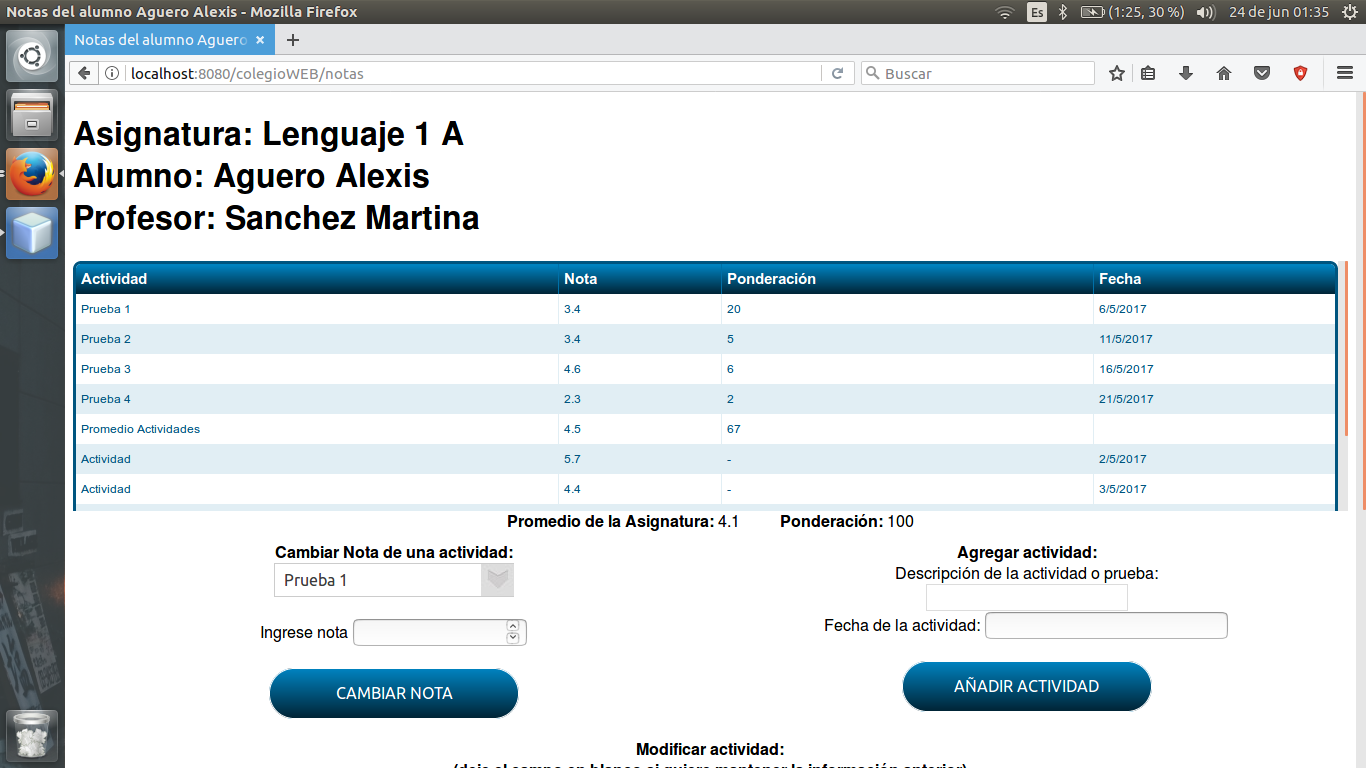


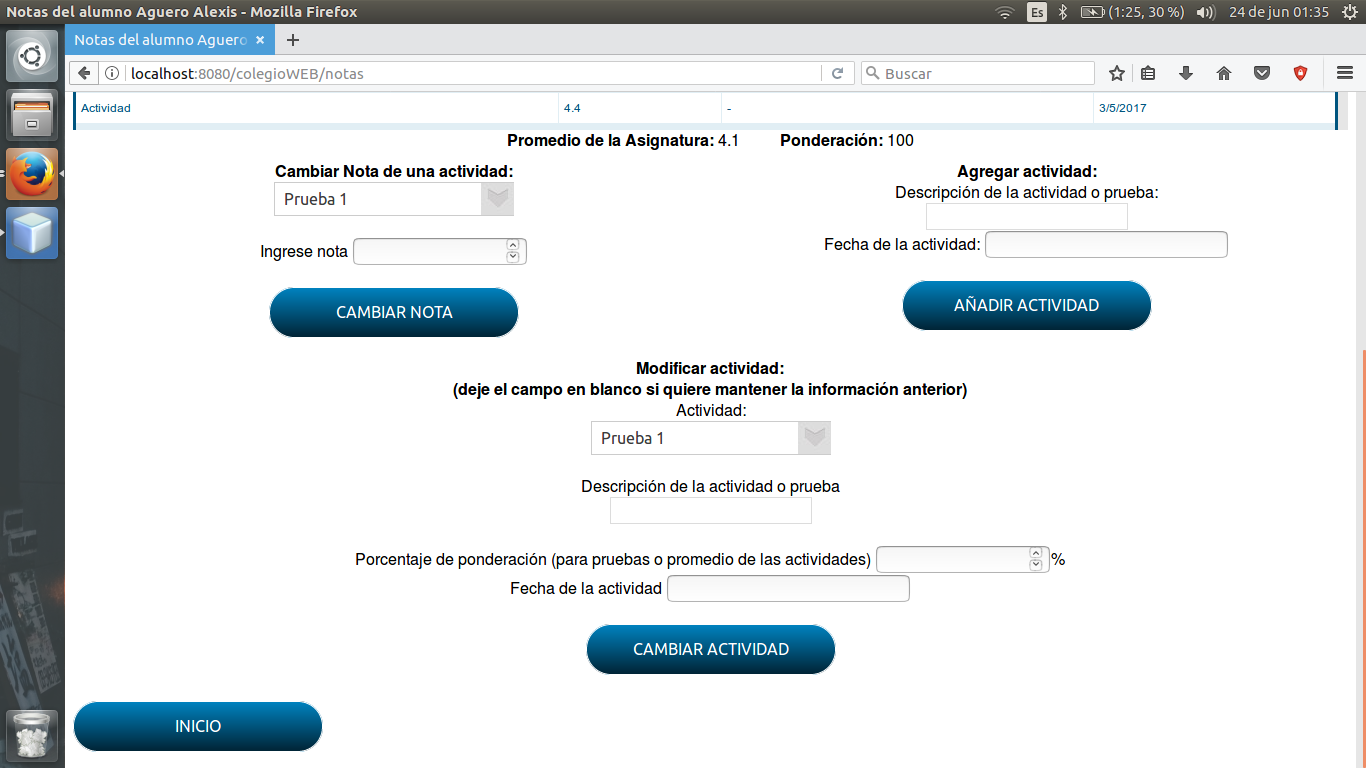


En el caso de seleccionar un alumno y presionar “Ver notas”, se desplegará la siguiente página en la cual el usuario deberá seleccionar la asignatura que desee revisar del alumno anteriormente seleccionado, además de eso, se puede añadir una asignatura completando los campos correspondientes al nombre de la asignatura y eligiendo al profesor que estará a cargo de impartir el ramo, en caso de estar registrado, en caso contrario se desplegará dinámicamente gracias a JavaScript los campos correspondientes a completar para añadirlo justo a la asignatura si se selecciona Nuevo profesor.

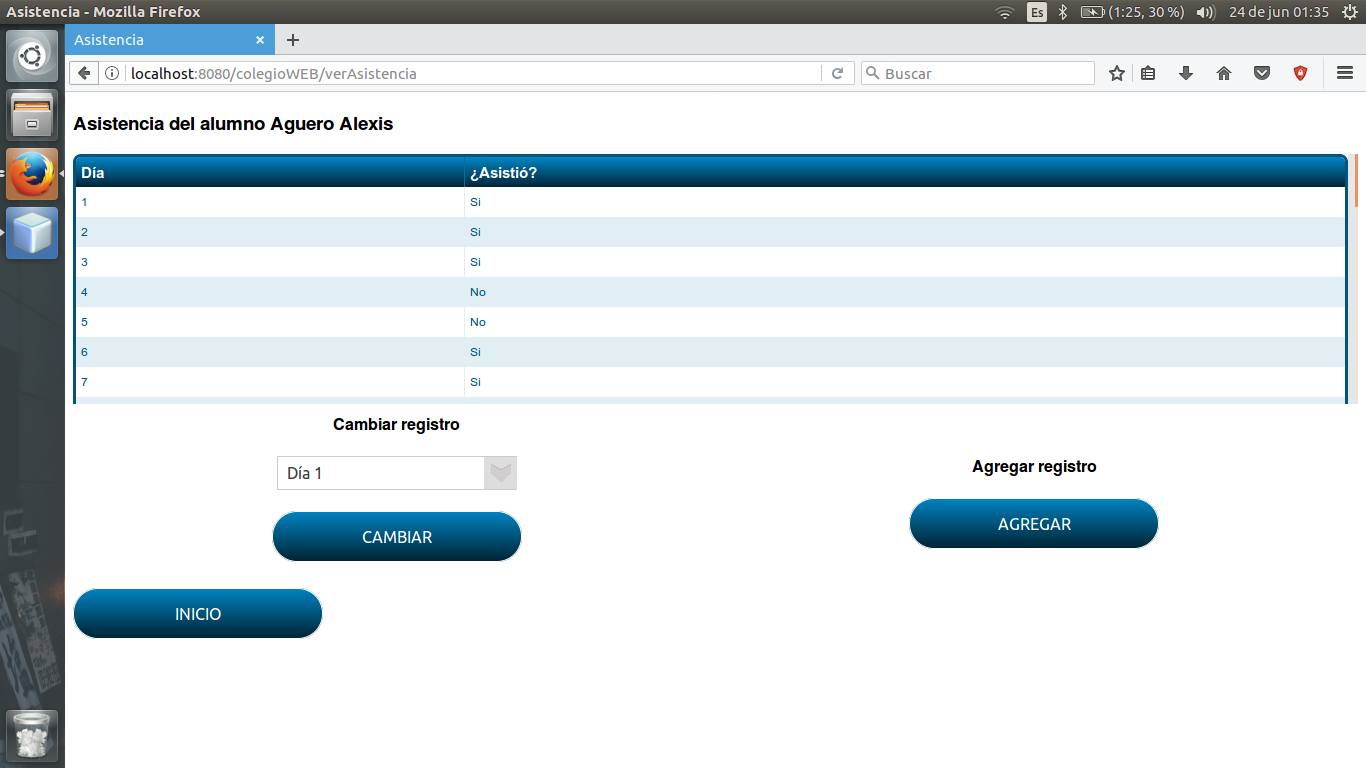


Una vez seleccionada la asignatura y presionado Ver notas en la página anterior, se desplegará la siguiente en la cual el usuario puede revisar las calificaciones del alumno, si es por alguna prueba o actividad, la ponderación de la misma en el promedio general y finalmente la fecha en la que se tomó o se realizó dicha prueba o actividad. Además de lo anterior, el usuario también puede modificar la nota de cualquiera de las actividades seleccionándola y reingresándola en campo correspondiente a Cambiar nota. También se puede agregar una nueva actividad a la planificación de la asignatura simplemente ingresando la descripción de la actividad o prueba que se desee añadir y la respectiva fecha en la que se quiere fijar, por otra parte si lo que se quiere es modificar una actividad, simplemente se selecciona la que se desea cambiar y se rellena lo que se quiera editar.





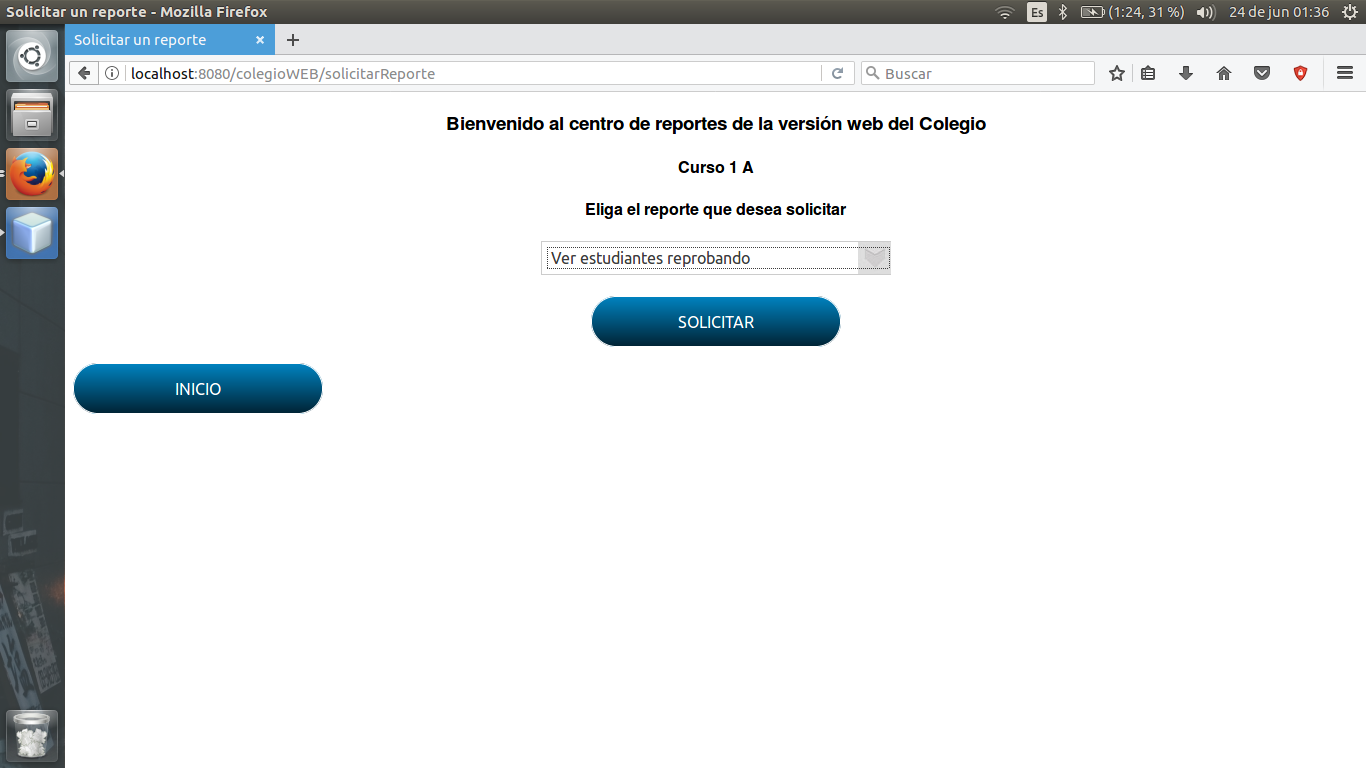
En el caso de seleccionar Ver asistencia, se desplegará la siguiente página en la cual el usuario puede revisar la asistencia del alumno, por día registrado de clases y verificando con un sí o no la asistencia del estudiante, estando aquí se puede cambiar el registro del alumno si es que hubo algún día en el que asistió pero se registró lo contrario, o por otro lado registrar otro día del alumno.

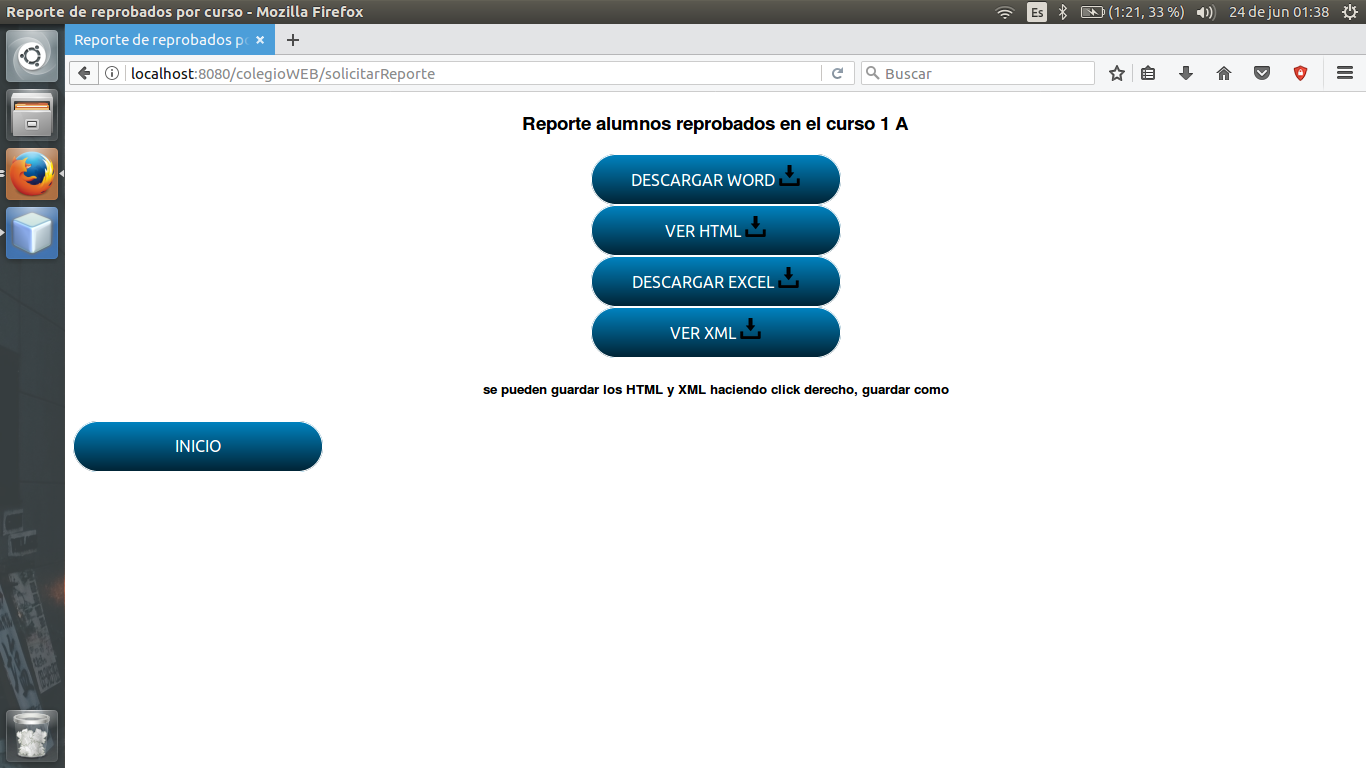


En el caso de seleccionar Ver anotaciones en la página anterior, se desplegará la siguiente página en la cual el usuario puede revisar las anotaciones que tenga registradas el alumno, estando aquí se puede registrar una nueva escribiendo el motivo y poniendo el tipo si es positiva o negativa, o si en caso contrario se quiere modificar alguna, simplemente se selecciona de la lista desplegable y se redacta o se modifica el motivo de la anotación.

# Captura de pantalla de 2017-06-24 01-36-07.png

En el caso de haber seleccionado Reportes, se le mostrará la siguiente página en la cual el usuario podrá solicitar cualquier reporte dentro de la lista desplegable que se encuentra, una vez elegido y presionado Solicitar se le generará los link de descarga del reporte pedido en diferentes formatos como Word, Excel, HTML y XML. En ciertos reportes en particular (insertar reportes) se le solicitará al usuario información adicional si el reporte seleccionado así lo requiere para que se genere.





# Conclusión / Comentarios finales

El objetivo de este proyecto era poner a prueba la materia pasada en la clase de Programación tal como el uso de Xml y Json para guardar objetos instanciados, el uso de Xsl para la transformación de Xml y comprobar la eficacia del desarrollo de Sistemas de Información con enfoque Orientado a Objetos junto y el uso de UML.

Si bien el uso de Xsl es bastante útil a la hora de leer y convertir Xml, puede ser difícil al principio ya que es una especie de “Lenguaje” para programar formatos de texto con enfoque y sintaxis diferentes a los de Java, pero al comprender la estructura de éste se pueden lograr muchos formatos diferentes a gusto.

Se puso a prueba la ventaja del Diagrama de Clases, cuando se necesitó modificar alguna relación de las clases principales y se comprobó que resulta realmente útil utilizar este diagrama para obtener un código, no fue difícil aplicar el cambio en éstas clases. Sin embargo para el caso de las ventanas no se hizo un diseño previo y fue engorroso realizar un cambio estructural en ellas ya que debía analizar todas las ventanas y si tenían relación con la ventana modificada.

Al trabajar en web se aprendió mucho sobre el lenguaje html, css y un poco javascript, investigando y navegando en busca de ejemplos sobre las distintas herramientas y estilos que se pueden utilizar. Estos conocimientos serán muy útiles en un futuro para la carrera de informática.