



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

Proyecto Ingeniería de Software 2015-2

IIC2143 - Ingeniería de Software
Camilo Álvarez L. - Vicente Domínguez M. - Rodrigo Saavedra R.
27 de Agosto del 2015

Objetivo del Proyecto

El objetivo del proyecto grupal es que los alumnos logren tener un acercamiento real a lo que es el desarrollo de software en la actualidad. Se busca que los alumnos puedan aplicar los conocimientos adquiridos en el curso durante el semestre, de tal forma que una vez finalizado el proyecto tengan una concepción propia de los factores que ayudan a desarrollar un software desde el punto de vista ingenieril (Levantamiento de requisitos, diseño, codificación y testing).

Desarrollo general

El proyecto se realizará en grupos de tres personas. En este semestre existirán dos proyectos diferentes a desarrollar. Cada grupo tendrá la opción de expresar su preferencia, la cual será considerada al momento de distribuir los grupos. Además cada grupo trabajará con un ayudante en específico durante todo el semestre, el cual será asignado al momento de distribuir los proyectos y este actuará a modo de cliente del software.

Cada proyecto contará con un enunciado donde se especificará la idea que el cliente tiene para su producto, explicitando las funcionalidades que se consideran mínimas para el software. Cabe mencionar que los requisitos de los proyectos pueden ir cambiando a lo largo del semestre, según las necesidades observadas por el ayudante.

El proyecto se dividirá en seis entregas, más una presentación a final de semestre. Las fechas de los entregables se indican en una sección posterior. Cada entrega contará con requisitos a evaluar según la materia vista en el ramo (no confundir con los requisitos del software). El enunciado de cada entrega será dado en la fecha de término de la entrega anterior, a excepción de la entrega 1, que será publicada el lunes 31 de agosto.

Es importante que planifiquen un buen diseño del código desde el comienzo del proyecto, así les será más fácil agregar funcionalidades a futuro.

Requisitos formales y generales para el proyecto

- Uso del lenguaje **JAVA** para el desarrollo del software
- Uso de un repositorio en GitHub
- Cualquier implementación extra en el software debe ser informada al ayudante y contar con una explicación en el repositorio del proyecto (dónde descargar, forma de instalación, etc)

Recomendaciones

- Para la implementación de la GUI se recomienda utilizar JAVA FX, para aquellos que hayan programado con WPF o PyQT esta se les hará más simple de utilizar.
- Como IDE recomendamos usar las otorgadas por Eclipse o NetBeans.

Fechas de entrega

- Entrega 1 (E_1): Viernes 11 de septiembre. Relatos de usuario, modelo de dominio y casos de uso.
- Entrega 2 (E_2): Viernes 25 de septiembre. Diagramas UML.
- Entrega 3 (E_3): Viernes 9 de octubre. Diseño de código en base a diagrama UML.
- Entrega 4 (E_4): Viernes 23 de octubre. Diseño GUI y programa funcional.
- Entrega 5 (E_5): Viernes 6 de noviembre. Plan de pruebas.
- Entrega 6 (E_6): Viernes 20 de noviembre. Programa terminado y testing integrado.
- Presentación final (P): Viernes 4 de diciembre.

Estas fechas son definitivas e impostergables. El atraso en alguna de ellas implica obtener nota **1.0** en dicha entrega.

Nota de proyecto(NP)

El cálculo de la nota de proyecto se calcula de la siguiente manera:

$$NP = 0,15E_1 + 0,1E_2 + 0,1E_3 + 0,15E_4 + 0,15E_5 + 0,15E_6 + 0,2P$$

Observación

Los aspectos formales del proyecto están sujetos a modificaciones según el desarrollo que se observe durante el semestre.

Proyectos

Academic Manager

Para muchos universitarios es importante poder contar con una herramienta computacional que les ayude a llevar su registro académico. De esta manera, les permitirá planificar su estadía en la universidad y priorizar aquellos ramos que son más importantes para ellos. Es por esto que se le ha pedido que cree un software capaz de llevar el registro académico de un alumno universitario, donde podrá anotar los semestres que ha cursado junto con los ramos que ha tomado en cada uno de ellos.

El software maneja el concepto de semestres. Cada semestre tendrá los datos de los ramos que ha cursado el alumno. Al iniciar el programa por primera vez, este debe pedirle al alumno sus datos personales: nombre, edad, sexo, carrera, RUT. Luego, en base a su carrera, elige una malla curricular y se le pide crear su primer semestre, tomando los ramos que corresponda. Una vez terminado el semestre, el alumno debe poder calificarlo, es decir, asignarle a cada ramo su nota obtenida. Luego el semestre se guarda y se le pide al alumno crear un nuevo semestre.

El alumno puede consultar a cada momento su historial académico. Este debe mostrar los ramos aprobados y reprobados, correspondientes a los distintos semestres cursados. Además debe ser posible incorporar una vista que muestre el avance curricular del alumno de acuerdo a sus ramos aprobados, con respecto a la malla. Se debe tener en cuenta que el alumno puede querer consultar su avance curricular para distintas mallas.

El alumno puede querer planificar su próximo semestre con anticipación. Es por esto que el software debe incorporar un buscador de cursos que entregue información acerca de los ramos que se dictan: Sigla, profesor, horario, sala, facultad, requisitos, créditos. Al momento de crear un nuevo semestre, el software debe ser capaz de controlar requisitos, toques de horario y máximo de créditos.

Este software está pensado para la gestión personal de alumnos de cualquier universidad. Es por esto que debe haber una cuenta de administrador académico. Esta cuenta debe proveer las funcionalidades de crear ramos con sus características (sigla, profesor, etc.), y crear las mallas curriculares de las distintas carreras impartidas por la universidad. Debe tenerse en cuenta de que un profesor no puede dictar dos ramos a la misma hora, que dos ramos no pueden dictarse en la misma sala y a la misma hora, y que pueden haber ramos que se dicten una vez al año. Se debe poder alternar entre las funcionalidades del modo alumno y modo administrador.

Tenga en consideración que cuando un alumno haya reprobado una determinada cantidad de ramos, el software no debe permitirle el ingreso. Este límite de ramos reprobados lo fija el administrador. El administrador también fija la regla para la cantidad máxima de créditos que se pueden tomar por semestre.

Servicio de encomiendas ChilExplox

El gerente de informática de la reconocida empresa de servicios de encomiendas *ChilExplox*, ha observado el sorprendente crecimiento que está teniendo la empresa. Preocupado por la estabilidad y buen funcionamiento del negocio, ha decidido mejorar el sistema informático que controla el servicio de encomiendas. En su intento se vio enfrentado a un sistema legado, mal diseñado, mal documentado y con ausencia total de **refactory**, ante tal descubrimiento ha decidido desarrollar un nuevo software, dado a que no cuenta con el equipo de trabajo adecuado, ha decidió externalizar, pidiendo la ayuda a los insignes alumnos de la PUC.

A continuación se procede a detallar el giro de la empresa y elementos que el gerente considera importantes:

La empresa cuenta con diversas sucursales en todo el país, en cada sucursal se reciben a diario decenas de encargos para enviar a otra sucursal en el país, por lo que en cada oficina debe existir un computador con acceso a la información propia de las encomiendas del lugar donde se está. Será necesario entonces, que el usuario tenga la opción de elegir la sucursal que está administrando en determinado momento. (O sea, en un mismo computador se puede tener acceso a la información de todas las sucursales existentes).

Diariamente llegan varios clientes con el objetivo de enviar encomiendas a todo Chile. Cuando son atendidos, el usuario del sistema debe poder crear un nuevo pedido, donde se ponen los datos del cliente (Nombre, número de teléfono, dirección) y además se deben colocar los datos del pedido (Peso, volumen, sucursal de destino, dirección de destino y prioridad). Tras llenar los datos de la encomienda (no necesariamente llenados los del cliente), el usuario tiene acceso a un presupuesto del costo del envío, calculado en función de las características de la encomienda. Para un mismo cliente se pueden asociar diversas encomiendas antes de cerrar el pedido (solo se puede cerrar el pedido si se llenan los datos del cliente). Una vez se tiene listo el pedido (la suma de encomiendas) se ve el monto final que debe pagar el cliente.

Lo primero que debería suceder es que los pedidos se mantengan ordenados según la urgencia que tienen. Cada pedido tiene un estado, el cual debe comenzar con el valor *“En origen”*, indicando que aún está en la sucursal desde donde se envía.

La empresa cuenta con una flota de camiones distribuidos en las diversas sucursales. El usuario del sistema puede ver la disponibilidad de camiones y puede escoger enviar algún pedido (a su elección). Cuando se envía un camión con un pedido (considere que un camión puede llevar un número fijo pedidos, pero a una misma sucursal. Ese número es determinado por ustedes), el estado de este último debe cambiar a *“En tránsito”*. Por otro lado, cada sucursal tiene una lista donde aparecen los camiones que han llegado a ella y que esperan ser descargados (Puede asumir que los camiones llegan inmediatamente a su destino). El usuario debe tener la opción de seleccionar un camión o todos, y recepcionar los pedidos que contienen. Con esta última acción el estado cambia a *“En destino”*.

Cuando se recibe un pedido, se deben revisar los datos de manera manual para confirmar la correcta recepción. Si todos los datos son correctos, el estado cambia a *“Entregado”*. A partir de este punto puede asumir que el sistema no debe preocuparse por lo que suceda con las encomiendas. En cualquier caso, al recibirse el pedido, el camión vuelve a la sucursal de origen.

El gerente de informática ha entrevistado a algunos miembros de la empresa para poder encontrar puntos que mejorar en el sistema, las observaciones más importantes se ponen a continuación:

Marcel Sandy, Supervisor de recepción:

- *“Hay veces en que el que envía una encomienda escribe mal la sucursal o la dirección de destino, en esos casos la política de la empresa dice que el operario que recibe la encomienda con el error, puede solicitar a la sucursal de origen que arregle la dirección ... hoy en día todo eso se hace llamando por teléfono y muchas veces no se genera un registro adecuado.”*

Ante esta observación, el gerente solicita explícitamente que exista la posibilidad de enviar mensajes de todo tipo a otra sucursal, y que existan mensajes especiales donde se avisa sobre un error en los datos y sobre la medida escogida por el operario. El gerente ha sido enfático en aclarar que por protocolo solo la sucursal de origen puede cambiar la dirección de un pedido.

Chris Rivers, Supervisor de envíos:

- *“Como es el operario quien escoge qué pedidos enviar, muchas veces se nos quedan pedidos importantes por más tiempo del debido, lo que obviamente enoja a los clientes”*

El gerente de informática cree que este problema se puede solucionar haciendo que el programa avise cuando alguna encomienda de alta prioridad aún no sea enviada.

Observación

Se espera que los problemas planteados por los supervisores encuentren una solución factible y eficiente en el software que ustedes desarrollen, respetando como mínimo lo expuesto por el gerente.