

Desarrollo de sistema de facturación y cobro para una institución educativa privada

Juan Carlos Rocafuerte Guanoluiza⁽¹⁾, Ing. Rayner Stalyn Durango Espinoza⁽¹⁾
Facultad de Ingeniería en Electricidad y Computación⁽¹⁾
Escuela Superior Politécnica del Litoral (ESPOL)⁽¹⁾
Campus Gustavo Galindo, Km 30.5 vía Perimetral
Apartado 09-01-5863. Guayaquil-Ecuador
jrocafue@espol.edu.ec, rdurango@espol.edu.ec

Resumen

El presente proyecto muestra el análisis, diseño, desarrollo e implementación de un Sistema de Facturación, realizado para una institución educativa, la cual tenía la necesidad de integrar un software que cumpla con los requerimientos de su modelo de negocio para la ejecución de un proceso de cobro eficaz, además de regirse de las normativas que exige el Servicio de Rentas Internas. Esta solución sirve de apoyo para los departamentos que se ven involucrados directamente con el proceso de cobro, siendo estos: Contabilidad, Financiero y Sistemas, disminuyendo la carga de trabajo que representaba la ejecución manual de varios procesos como: cheques protestados, gestión de saldos a favor, descuentos a estudiantes, informe de liquidez financiera, entre otros.

Palabras Claves: facturación, universidad, cobro, sistema, programación, FIEC, LICISIS.

Abstract

This project shows the analysis, design, development and implementation of a Billing System, made to an educational institution, which had the need to integrate software that complies with the requirements of its business model to the execution of a process effective collection plus governed by the regulations required by the SRI. This solution provides support for departments that are directly involved in the collection process, these being: Accounting, Finance and Systems, decreasing the workload that represented the manual execution of several processes such as: protested checks, credit balances, discounts to students, report financial liquidity, among others.

Keywords: billing, university, charging, system, programming, FIEC, LICISIS.

1. Introducción

La empresa para la cual se desarrolló este proyecto es una empresa privada que brinda servicios de educación superior y capacitaciones para obtener la licencia de conducir profesional.

Debido a la alta demanda en los últimos años, se ha creado otro campus en el cantón Samborondón de la provincia del Guayas.

Cuando un estudiante desea matricularse en la institución, le hacen la toma de datos personales y académicos, seguidos por el cobro de algunos rubros tales como el valor de la matrícula, valor de las materias, transporte, seguro de vida, entre otros.

Se brindan diferentes clases de apoyo y facilidades, tales como: el valor del semestre (número de asignaturas a tomar por el valor de cada asignatura) se los difiere en varias cuotas, se hacen descuentos por el pronto pago, se otorgan becas a los estudiantes destacados académicamente o por convenios con otras instituciones, se puede pagar la carrera por adelantado, y además aceptan diversas formas de pago.

Utilizan un sistema de cobro ineficiente que no se adapta al modelo de negocio de la empresa.

La metodología de desarrollo de software implementada fue Cascada, ya que el proyecto estuvo dividido en varias etapas que fueron: análisis de requerimientos, diseño del sistema, desarrollo del sistema, pruebas e integración del sistema, funcionamiento y mantenimiento del sistema. [1]

2. Procesos críticos del negocio

Los procesos que representaban más del 50% de los problemas, eran dos: cobro de materias a estudiantes y el cierre de caja.

2.1. Cobro de materias a estudiantes. El estudiante se acerca a efectuar el pago de sus deudas a caja, el cajero consulta las deudas le pregunta cuáles desea cancelar y solicita el valor a cancelar. El estudiante entrega el valor solicitado en cualquiera de las formas de pagos que ellos aceptan. En caso de que el estudiante quiera emitir un cheque posfechado, primero tiene que

consultarlo con la directora financiera para que ella de la orden para que lo acepten en caja, de acuerdo a un previo análisis de la situación financiera por la que esté atravesando el estudiante.

Una vez verificadas las formas de pago, el cajero procede a registrar el pago en el sistema y a generar la respectiva factura, para luego en caso de que exista un descuento crear una nota de crédito con el valor del descuento sobre aquella factura.

2.2. Cierre de caja. Cuando el cajero finaliza su jornada de trabajo, realiza el arqueo de caja que consiste en comparar los documentos como facturas, vouchers, cheques, papeletas de depósitos, transferencias, notas de crédito, comprobantes de ingreso contra el informe de caja que arroja el sistema para ver si cuadran los valores.

Luego procede a depositar en el banco el dinero en efectivo y los cheques al día; entrega las papeletas del depósito al supervisor, quien las registra en sistema y cierra la caja.

3. Diseño del sistema

En el diseño del sistema, se realizaron las siguientes actividades: estructuración y estandarización de la base de datos, definición de los modelos del sistema a desarrollar, diagramas de los procesos del negocio, diseño de interfaz de usuario para la entrada, procesamiento y presentación de los datos, iconografía de la aplicación.

4. Requerimientos técnicos

Los requerimientos técnicos para el desarrollo como funcionamiento del sistema son los siguientes:

Tabla 1. Requerimientos no funcionales

Visual Studio 2012	Diseño y desarrollo de interface de usuario.
SQL Server 2012	Gestor de la base de datos del sistema.
Windows Server 2012	Sistema Operativo donde reside la base de datos del sistema.
Crystal Reports 11	Herramienta para diseño y desarrollo de la reportería del sistema.
ODBC PostgreSQL	Driver utilizado para la comunicación entre bases de datos SQL Server y PostgreSQL.

5. Arquitectura del sistema

La arquitectura del sistema está dividida en dos capas: capa de datos y capa de aplicación.

Además de la elaboración y entrega de los respectivos manuales de usuarios y manuales técnicos.

La capa de datos se lleva a cabo en dos servidores con sistemas operativos y bases de datos diferentes, creando un entorno de interacción heterogéneo. La comunicación entre estas dos plataformas es llevada a cabo gracias al driver ODBC de PostgreSQL.

La capa de aplicación está alojada en cada una de las máquinas clientes, en las cuales también se encuentra las reglas del negocio.

La data es enviada a través del formato XML para garantizar la atomicidad de las transacciones.

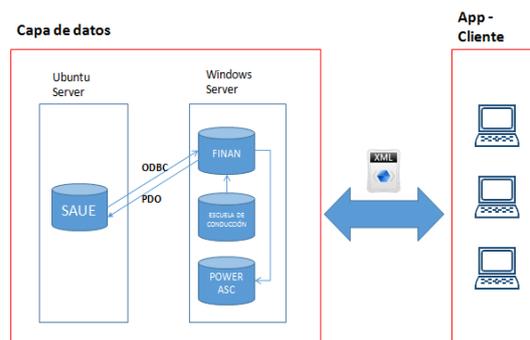


Figura 1. Arquitectura del sistema implementado

6. Resultados obtenidos

En cualquier parte que se especifique “Times”, usted también puede usar “Times Roman” o “New Times Roman”.

7. Pruebas del sistema

Las pruebas se realizaron en paralelo, es decir el sistema nuevo con el sistema que iba a ser reemplazado.

Se registraban cobros en ambos sistemas y al final del día cuando se realizaba el cierre de caja se comparaban los resultados de los reportes que arrojaban ambos sistemas para sacar conclusiones y proceder a corregir en caso que lo requiera. Cuando se migraban las transacciones del día al sistema de contabilidad, la contadora era la encargada de revisar si los asientos contables y los valores eran los correctos.

La prueba tuvo una duración de aproximadamente 3 semanas.

8. Capacitaciones

Las capacitaciones se dividieron en dos fases: usuarios que manejan el sistema y usuarios del departamento de sistemas.

Conclusiones

1. Los requerimientos de información iniciales, no se realizaron con los usuarios directos del sistema, sino que fueron supuestos por parte del jefe de proyecto.
2. No se llevó un debido análisis y diseño de los procesos del negocio que intervienen en la solución de los requerimientos encontrados.
3. En base a la mala definición de los requerimientos previamente mencionados, se realizó una errónea estimación de los tiempos en el cronograma del proyecto.
4. No se tomó en cuenta la migración de los datos del sistema actual tanto en su contexto técnico como logístico.
5. Entorno inestable de trabajo en los departamentos de sistemas y contabilidad del cliente, los cuales son claves para el diseño, desarrollo e implementación de la aplicación.
6. No hubo un alcance definido, por ende los requerimientos seguían llegando después de haber concluido los que se habían definido en la fase de análisis.
7. El sistema que utilizaban anteriormente, generó una base de datos inconsistente, lo cual se convirtió en un desafío al momento de migrar la información antigua.

Referencias

- [1] Kendal y Kendal, Análisis y diseño de sistemas Segunda Edición, México, 2005.