

# UNIVERSIDAD TÉCNICA DEL NORTE

## FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

### ESCUELA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

#### INFORME TÉCNICO

TESIS PREVIA LA OBTENCIÓN DEL TÍTULO DE INGENIERA EN  
SISTEMAS COMPUTACIONALES



#### TEMA:

**Desarrollo de Aplicaciones Web con el Framework Java  
Server Faces (JSF).**

#### APLICATIVO:

**Sistema de Administración Estudiantil para los Colegios  
Fiscales de la provincia de Imbabura con Herramientas de  
Software Libre.**

**Autor:** Dayce Gabriela Tulcán Suquitana

**Director:** Ing. Miguel Orquera

Ibarra, Marzo de 2010

# CAPITULO I

## APLICACIONES WEB

Con la introducción de Internet y de la Web en concreto, se han abierto infinidad de posibilidades en cuanto al acceso a la información desde casi cualquier sitio. Esto representa un desafío a los desarrolladores de aplicaciones, ya que los avances en tecnología demandan cada vez aplicaciones más rápidas, ligeras y robustas que permitan utilizar la Web.


La creación de aplicaciones web requiere la existencia de software ejecutándose en el servidor, que genere automáticamente los ficheros HTML que se visualizan en el navegador del usuario. La comunicación entre el cliente y el servidor se realiza a través del protocolo HTTP y éste delega a otros módulos la generación dinámica de páginas HTML que se envían al cliente, el navegador cliente es independiente de la tecnología que se utilice en el servidor para generar páginas de forma dinámica.

Las Aplicaciones Web pueden considerarse como un sitio web al que se accede mediante un navegador o browser, y que está dotado de: interactividad, una base de datos, comunicaciones encriptadas y contraseñas de acceso.

En las Aplicaciones Web tanto el software como los datos se encuentran en una máquina remota, centralizada que permite trabajar de manera estable, continua e ininterrumpida. El usuario se conecta a este servidor a través de un navegador web.

La plataforma Web aporta múltiples ventajas frente al software de escritorio lo que hace de ésta una interesante alternativa:

- ✚ **Multiusuario.**- en las aplicaciones web puede haber varios usuarios conectados al sistema simultáneamente, cada uno a través de un computador diferente y en distintas ubicaciones geográficas.
- ✚ **Independencia de software.**- para acceder a la aplicación sólo es requisito un navegador web, sin necesidad de instalar en cada equipo otro software especial.
- ✚ **Seguridad.**- al encontrarse la aplicación en un servidor remoto, el funcionamiento de la misma y los valiosos datos que contiene son totalmente independientes del ordenador utilizado para acceder a la aplicación.

 **Multiplataforma e Interoperabilidad.**- las aplicaciones web son multiplataforma por diseño; esto significa que igual podrá ingresar a la aplicación desde un ordenador con un Sistema Operativo Windows, Linux u otro.

## CAPITULO II

### FRAMEWORKS MVC

#### ARQUITECTURA MODELO VISTA CONTROLADOR

MVC es un patrón de arquitectura de software; fue diseñado para reducir el esfuerzo de programación necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos; donde las responsabilidades estén bien distribuidas entre las distintas partes del diseño que son: modelo, vista y controlador.

El Modelo representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico del Controlador o de las Vistas, ni siquiera contiene referencias a ellos.

La Vista está representada por un conjunto de pantallas que representan la interfaz de usuario, es decir lo que se muestra a los ojos del usuario.

El controlador es el responsable de recibir los eventos, determinar el procesador del evento, invocar al procesador y finalmente provocar la generación de la vista apropiada.

#### STRUTS

Struts está construido básicamente sobre las tecnologías de Servlets y JSP. Struts combina Servlets, JSPs, custom tags propios y recursos de la aplicación en un único Framework. Struts fue creado por Craig McClanahan y forma parte del proyecto Apache Struts, de Apache Software Foundation.

Struts oculta al programador los detalles del protocolo HTTP, JSP, Servlets, etc. Un programador Struts puede desconocer estos nombres, sin embargo tener

conocimiento de las tecnologías de base de Struts hace que se puedan hacer soluciones creativas.

## **TAPESTRY**

Tapestry es un Framework para el desarrollo de aplicaciones web, basado en componentes. En Tapestry todo son componentes y los componentes se pueden anidar unos dentro de otros para formar componentes más complejos. Tapestry se encarga de manejar por nosotros las propiedades persistentes, la validación de formularios, la localización, etc., permitiendo al desarrollador centrarse en la lógica del negocio.

## **JAVASERVER FACES**

JavaServer Faces es un Framework para desarrollar aplicaciones web en plataforma Java. JSF es el primer estándar de Java para aplicaciones web esto es muy importante ya que podemos encontrar implementaciones libres y comerciales de este Framework; además que los IDEs más conocidos lo soportan. JSF provee de todos los componentes necesarios para crear interfaces de usuario como son botones, cajas de texto, tablas, grids, combos entre otros, éstos componentes son manejados como objetos con estado en el lado del servidor; también provee al desarrollador de una serie de validadores y conversores estándar además de permitir al usuario crear los suyos; también permite asociar eventos a componentes de forma individual; la navegación es manejada de forma declarativa en un archivo de configuración XML, mediante reglas y casos de navegación.

El ciclo de vida de una aplicación JSF consta de los siguientes pasos:

- **Reconstruir el árbol de componentes.-** JSF construye el árbol de componentes y conecta los manejadores de eventos y los validadores y guarda el estado en el FacesContext; éste contiene toda la información necesaria para procesar una petición.
- **Aplicar valores de la petición.-** Una vez construido el árbol de componentes, cada componente del árbol extrae su nuevo valor desde los parámetros de la petición y el valor es almacenado localmente en el componente. Si falla la conversión del

valor, se genera un mensaje de error asociado con el componente; este mensaje se mostrará durante la fase de renderizar la respuesta, junto con cualquier error de validación resultante de la fase Procesar validaciones.

Si en esta fase se produce un evento, JSF emite el evento al oyente correspondiente. Si algunos componentes de la página tienen atributos inmediatos, entonces la validación, conversión y los eventos asociados a estos componentes son procesados en esta fase.

- **Procesar validaciones.**- en esta fase se procesan todas las validaciones registradas en los diferentes componentes.
- **Actualizar los valores del modelo.**- cuando JSF determina que un dato es válido puede pasar por el árbol de componentes y actualizar los valores del objeto del modelo correspondiente con los valores locales del componente.
- **Invocar la aplicación.**- Si la vista que se está procesando ya existe se reconstruirá con información del estado de una petición anterior y si un componente ha disparado un evento, estos eventos se emiten a los oyentes interesados. A continuación JSF, configura el árbol de componentes de la respuesta a esa nueva página.
- **Renderizar la respuesta.**- JSF invoca las propiedades de codificación de los componentes y dibuja los componentes del árbol grabado en el FacesContext. Si en fases anteriores se encontraron errores éstos son mostrados en componentes message.

## CAPITULO III

### DESARROLLO DE APLICATIVO

#### FRAMEWORKS UTILIZADOS

- JSF para la vista y el controlador.
- Hibernate para la persistencia de datos.

#### ENTORNO DE DESARROLLO

El IDE utilizado es JDeveloper de Oracle, cuenta con excelente documentación y se lo puede descargar gratuitamente desde la página oficial de Oracle.

## **SERVIDOR DE APLICACIONES**

El Servidor de Aplicaciones utilizado es GlassFish Server V2.1, este servidor no necesita mayores requerimientos de hardware, es libre y lo podemos descargar gratuitamente de su página oficial.

GlassFish incluye las últimas versiones de JavaServer Faces, JavaServer Pages, Enterprise Java Beans entre otros.

## **BASE DE DATOS**

PostgreSQL es un sistema de base de datos relacional que es robusto, escalable y cumple con los estándares SQL.

Soporta transacciones seguras, uniones, triggers, procedimientos almacenados; además incluye una gran variedad de tipos de datos.

## **PROBLEMA**

Hoy en día, los sistemas informáticos están presentes en la mayoría de las actividades laborales que desempeña el ser humano, facilitando dichas actividades al usuario del sistema, permitiéndole hacer su trabajo en menor tiempo y de manera eficiente.

Existen sistemas informáticos para entidades bancarias, para llevar la contabilidad de una empresa, para comercio electrónico, y para todas las áreas que podamos imaginar las cuales han automatizado sus procesos, sin ser una excepción el área educativa.

En nuestro país el estado asigna a la educación un bajo porcentaje del presupuesto general, lo que no ha permitido a todos los colegios adquirir un sistema informático para realizar las actividades propias de estas instituciones como son matriculación de estudiantes, ingreso de notas, registro de asistencia de docentes y estudiantes entre otros. Y las instituciones que lo han logrado adquirir en muchos casos lo han hecho por autogestión.

## DEFINICION DE REQUERIMIENTOS

El sistema cuenta con los siguientes tipos de usuario, los mismos que podrán acceder a las diferentes páginas de la aplicación según su perfil:






- **Secretaria.-** puede realizar las siguientes operaciones, impresión de libretas, promociones, nóminas, actas de grado, horarios por curso entre otras.
- **Administrador.-** puede ingresar datos que sirven como parámetros del sistema como son: fechas de vigencia para ingreso de notas que pueden ser trimestrales, de supletorios o de grado; crear y eliminar usuarios, crear nuevas materias, cursos, especialidades en caso de que sea necesario, ingresar los horarios de cada curso entre otras.
- **Inspector.-** este tipo de usuario puede registrar y justificar las faltas de estudiantes y docentes. Obtener reportes de asistencia, calificar la conducta de los diferentes cursos a su cargo.
- **Docente.-** este usuario puede ingresar notas parciales, obtener un reporte de promedios, imprimir su horario personal de clases entre otros.

## DISEÑO MEDIANTE UML

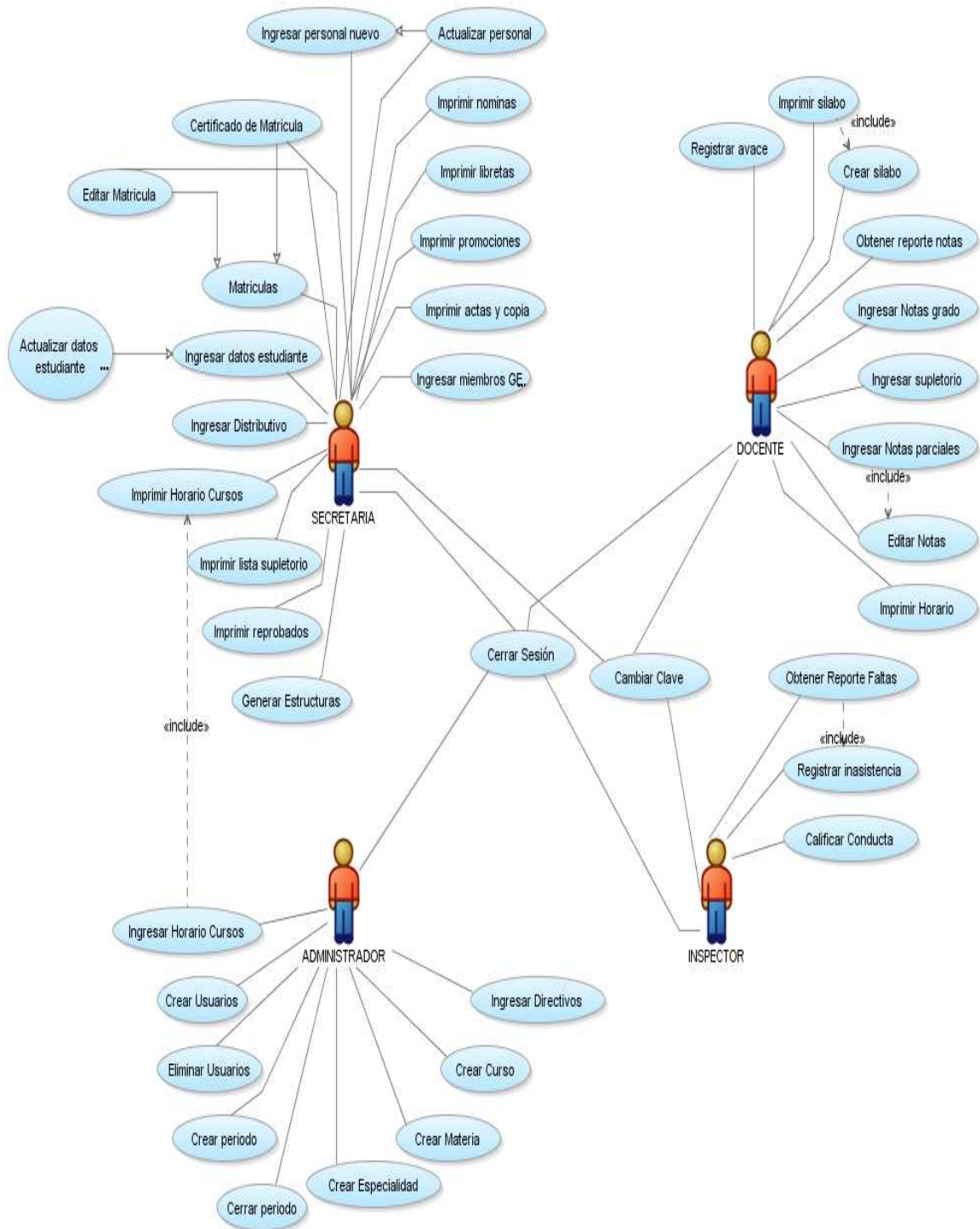
Lenguaje Unificado de Modelado, es un lenguaje gráfico que permite visualizar, construir y documentar un sistema.

Los diagramas UML son un conjunto de herramientas que permiten modelar sistemas orientados a objetos.

Los diagramas utilizados para modelar el sistema son los siguientes:

-  Diagrama de casos de uso
-  Diagrama de secuencias
-  Diagrama de colaboración
-  Diagrama de actividades
-  Diagrama de clases

## DIAGRAMA DE CASOS DE USO DE LA APLICACION



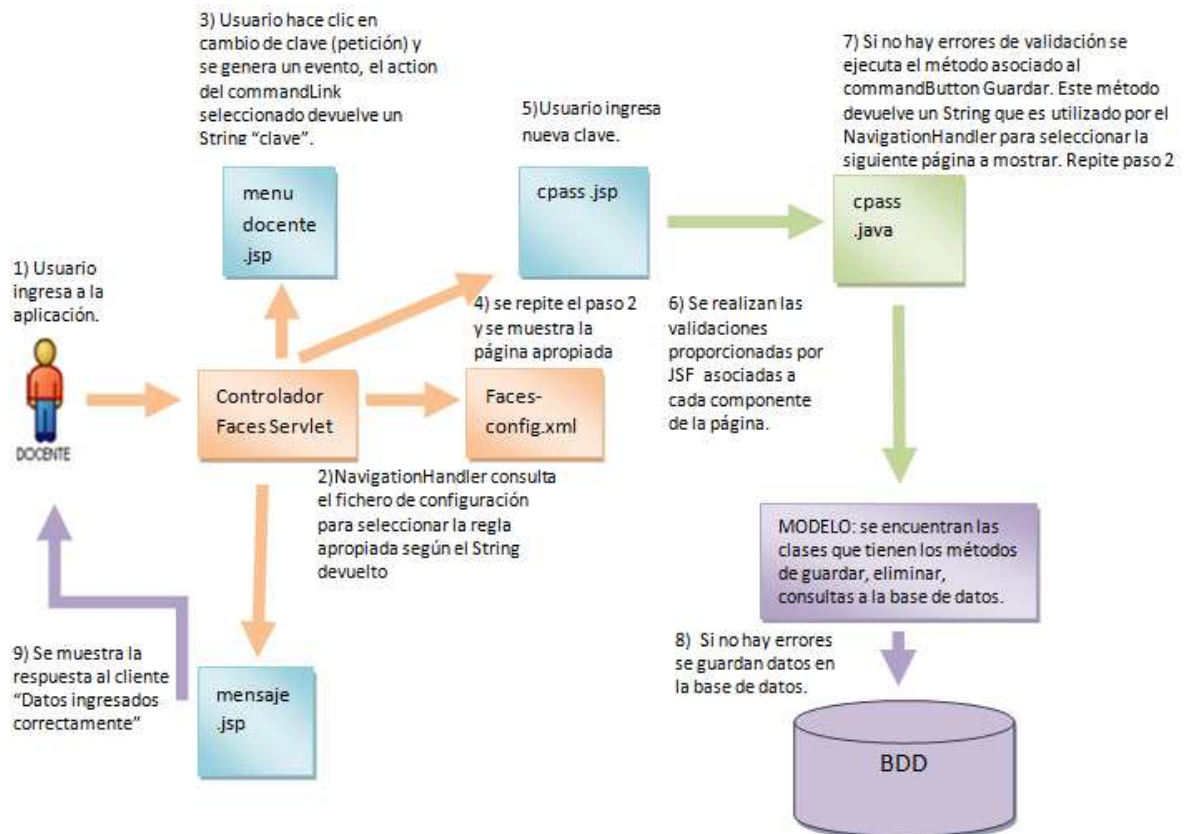
## DIAGRAMA DE CLASES







## ARQUITECTURA DE LA APLICACIÓN



## CONCLUSIONES

- La evolución de Frameworks Web marca una tendencia en la abstracción del protocolo HTTP, para beneficiarse de los modelos basados en controles y componentes como en la programación de aplicaciones de escritorio.
- JSF se acerca más a la definición del patrón MVC; ya que a pesar de trabajar sobre el protocolo sin estado HTTP mantiene el estado de los componentes de UI en el servidor, lo que hace posible la creación de aplicaciones web orientadas a eventos.
- Al ser JavaServer Faces un estándar podemos encontrar implementaciones de JSF tanto libres como comerciales lo que nos permite escoger la que más nos convenga.
- Usar herramientas gratuitas para desarrollar aplicaciones permite reducir costos en el desarrollo de aplicaciones.
- Manejar la navegación mediante reglas que se configuran en un archivo XML facilita el mantenimiento de una aplicación.

- ✚ Para una institución educativa tener un sistema que automatice sus procesos es de gran ayuda para brindar un mejor y eficaz servicio a sus estudiantes, además de evitar errores, y tener toda su valiosa información en una base de datos.
- ✚ Es importante indicar que los requerimientos para el desarrollo del sistema se los tomó de un colegio fiscal al azar y en base a éstos se elaboró el sistema, mismo que podría ser instalado en otros colegios que cumplan con los mismos parámetros bajo los que fue desarrollado.

## RECOMENDACIONES

- ✚ Para reducir costos en el desarrollo de una aplicación lo más recomendable es utilizar herramientas de software libre.
- ✚ Instalar el sistema antes de terminar de desarrollarlo permite que los usuarios puedan hacer sugerencias acerca de que funcionalidades se pueden incorporar al sistema. Además permite a los usuarios familiarizarse poco a poco con el sistema y se les facilite utilizarlo una vez que éste listo para ponerlo en funcionamiento.
- ✚ Realizar la capacitación a los usuarios del sistema tratando de no utilizar términos técnicos, hacerlo lo más comprensible para ellos y tomar en cuenta sus opiniones sobre el sistema. Explicándoles que el uso del sistema les facilitará su trabajo y que éste es necesario para la institución.

# CHAPTER I

## WEB APPLICATIONS

With the introduction of Internet and Web in particular have opened up many possibilities in terms of access to information from almost anywhere. This represents a challenge to application developers, as advances in technology are increasingly demanding applications faster, lightweight and robust for using the Web.

The creation of web applications requires the existence of software running on the server, which automatically generates HTML files that are displayed in the browser. Communication between the client and the server is done via the HTTP protocol and delegated to other modules this dynamic generation of HTML pages that are sent to the client, the client browser is independent of the technology used on the server to generate pages dynamically.

Web Applications can be considered as a website which is accessed through a browser, and is equipped with: interactivity, database, communications and encrypted passwords.

Web Applications in both the software and the data is in a remote, centralized, which allows working in a stable, continuous and uninterrupted. The user connects to this server through a web browser.

The Web platform provides multiple advantages over desktop software that makes this an interesting alternative:

- ✚ **Multiusuer.-** in web applications can have multiple users connected to the system simultaneously, each through a different computer and geographical locations.
- ✚ **Independence of software.-** to access the application requirement only a web browser without installing other software on each computer special.
- ✚ **Security.-** to find the application on a remote server, the operation of it and the valuable data it contains are completely independent of the computer used to access the application.
- ✚ **Multiplatform and Interoperability.-** Web applications are cross-platform by design, meaning that as you logging into the application from a computer with a Windows Operating System, Linux or another.

## CHAPTER II

### FRAMEWORKS MVC

#### MODEL VIEW CONTROLLER ARCHITECTURE

MVC is a pattern of software architecture was designed to reduce the programming effort required to implement multiple and synchronized systems of the same data, where responsibilities are well distributed among different parts of the design are: model, view and controller.

The Model represents the information in the program. Manages and controls all data transformations. The model has no specific knowledge of the controller or views, even contains references to them. The view is represented by a set of screens that represent the user interface, what is displayed to the user's eyes.

The controller is responsible for receiving events, determine the processor of the event, invoking the processor and eventually cause the generation of the appropriate view.

#### STRUTS

Struts is basically built on Servlet and JSP technologies. Combines Struts Servlets, JSPs, custom tags themselves and application resources into a single framework. Struts was created by Craig McClanahan and is part of the Apache Struts, Apache Software Foundation.

Struts hidden from the programmer the details of the HTTP protocol, JSP, Servlets, etc. A Struts developer can ignore these names, however be aware of Struts-based technologies that can be done to creative solutions.

#### TAPESTRY

Tapestry is a framework for developing Web applications based on components. In all Tapestry components are components and can be nested within one another to form more complex components. Tapestry is responsible for managing the persistent properties for us, form validation, location, etc., Allowing the developer to focus on business logic.

## JAVASERVER FACES

JavaServer Faces is a framework for developing web applications in Java platform. JSF is the first standard Java Web application that is very important because we can find free and commercial implementations of this Framework; well known IDEs that support it. JSF provides all the necessary components for creating user interfaces such as buttons, text boxes, tables, grids, combos, among others, these components are handled as stateful objects on the server side, the developer also provides a series of standard validators and converters also allow the user to create their own, and allows events to be associated with individual components, the navigation is handled declaratively in an XML configuration file, using navigation rules and cases.

The life cycle of a JSF application consists of the following steps:

- ✚ **Rebuild the JSF component tree.-** builds the component tree and connect event handlers and validators and saves the state in the FacesContext, which shall contain all information necessary to process a request.
- ✚ **Apply request values.-** Once built, the tree of components, each component of the tree extracts its new value from the request parameters and the value is stored locally on the component. If the value conversion fails, it generates an error message associated with the component, this message will be displayed during the render response phase, along with any validation error resulting from the Process Validations phase. If at this stage an event occurs, JSF gives the listener for the event. If some components have attributes page immediately, then the validation, conversion, and events associated with these components are processed in this phase.
- ✚ **Process validation.-** At this stage all validations are processed in the different components registered.
- ✚ **Update model values.-** when JSF determines that a valid data may pass through the component tree and update the values of the model object corresponding to local values of the component.
- ✚ **Update model values.-** when JSF determines that a valid data may pass through the component tree and update the values of the model object corresponding to local values of the component.



- ✚ **Rendering the response.**- JSF invokes the encoding properties of the components and draw the tree components recorded in the FacesContext. If upstream errors were found they are displayed in message components.

## **CHAPTER III**

### **DESARROLLO DE APLICATIVO**

#### **FRAMEWORKS USED**

- JSF for the view and controller.
- Hibernate for data persistence.

#### **DEVELOPMENT ENVIRONMENT**

The IDE used is Oracle JDeveloper, has excellent documentation and can be downloaded for free from the official page of Oracle.

#### **APPLICATION SERVER**

Server GlassFish Application Server V2.1 is used, this server does not need further hardware requirements, is free and you can download for free from their official website.

GlassFish includes the latest versions of JavaServer Faces, JavaServer Pages, Enterprise Java Beans and others.

#### **DATABASE**

PostgreSQL is a relational database that is robust, scalable and meets the standard SQL.

Supports secure transactions, joins, triggers, stored procedures, also includes a variety of data types.



## PROBLEM

Today, computer systems are present in most work activities performed by the human being, enabling these activities to the user of the system, allowing you to work in less time and efficiently.

There are computer systems for banks to keep accounts of a company, to electronic commerce, and for all imaginable areas which have automated their processes, without exception the educational field.

En nuestro país el estado asigna a la educación un bajo porcentaje del presupuesto general, lo que no ha permitido a todos los colegios adquirir un sistema informático para realizar las actividades propias de estas instituciones como son matriculación de estudiantes, ingreso de notas, registro de asistencia de docentes y estudiantes entre otros. Y las instituciones que lo han logrado adquirir en muchos casos lo han hecho por autogestión.

## DEFINITION OF REQUIREMENTS

The system has the following types of user, the same may access the various pages in the application according to your profile:

- ✚ **Secretary.-** can perform the following operations, printing of booklets, promotions, salary, grade records, course schedules by, among others.
- ✚ **Administrator.-** may enter data that serve as system parameters such as: effective dates for entry of notes that can be quarterly, to provide or grade; create and delete users, create new materials, courses, specialties where necessary, enter each course schedules and more.
- ✚ **Inspector.-** This type of user can record and justify the misconduct of students and teachers. Obtain reports of attendance, describing the behavior of different courses in their care.
- ✚ **Professor.-** the user can enter notes partial get a report of averages, print your personal schedule of classes among others.

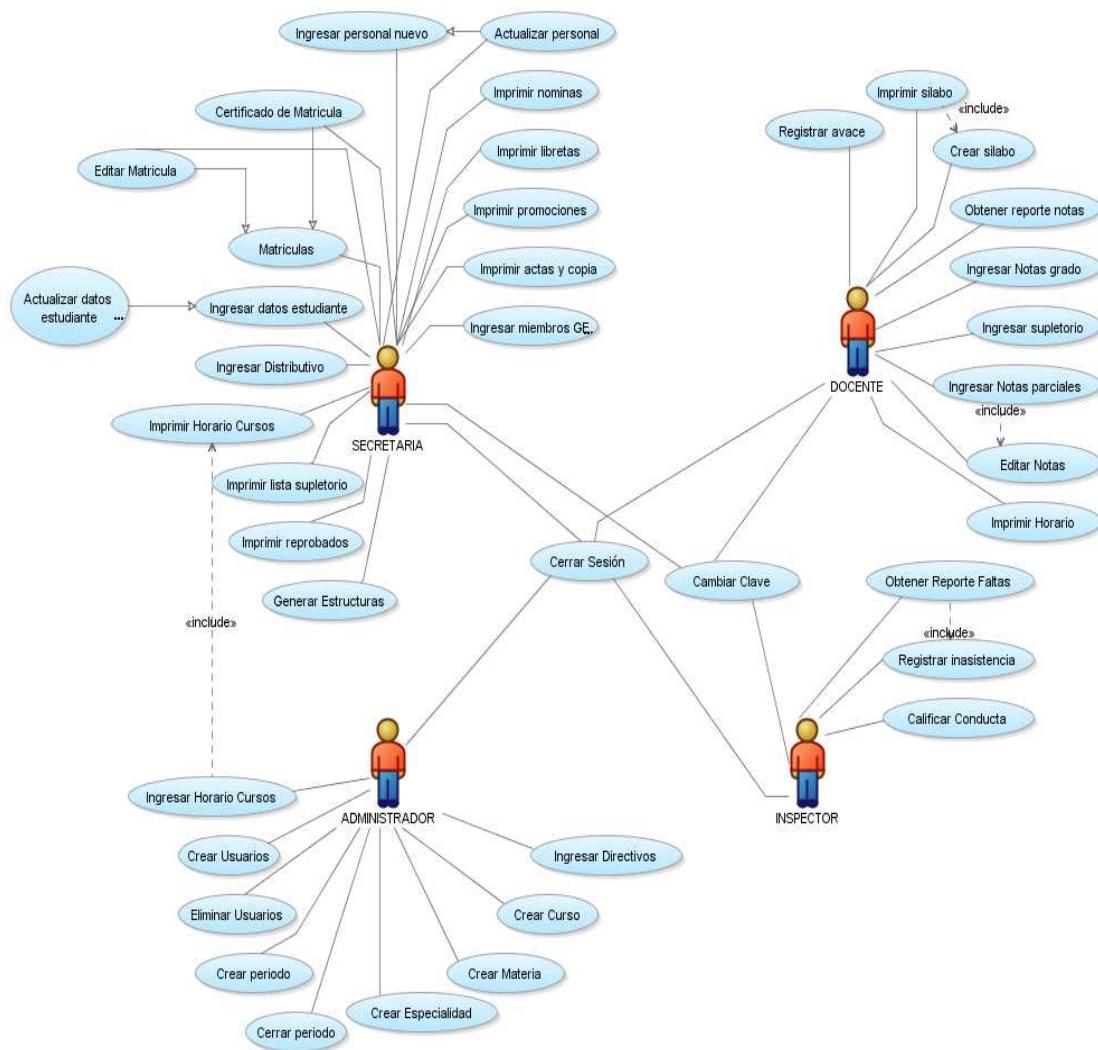
## DESIGN USING UML

Unified Modeling Language, is a graphic language to visualize, build and document a system.

UML diagrams are a set of tools for modeling object-oriented systems.

- Use Case Diagram
- Sequence Diagram
- Collaboration diagram
- Activity diagrams
- Class Diagram

## USE CASE DIAGRAM

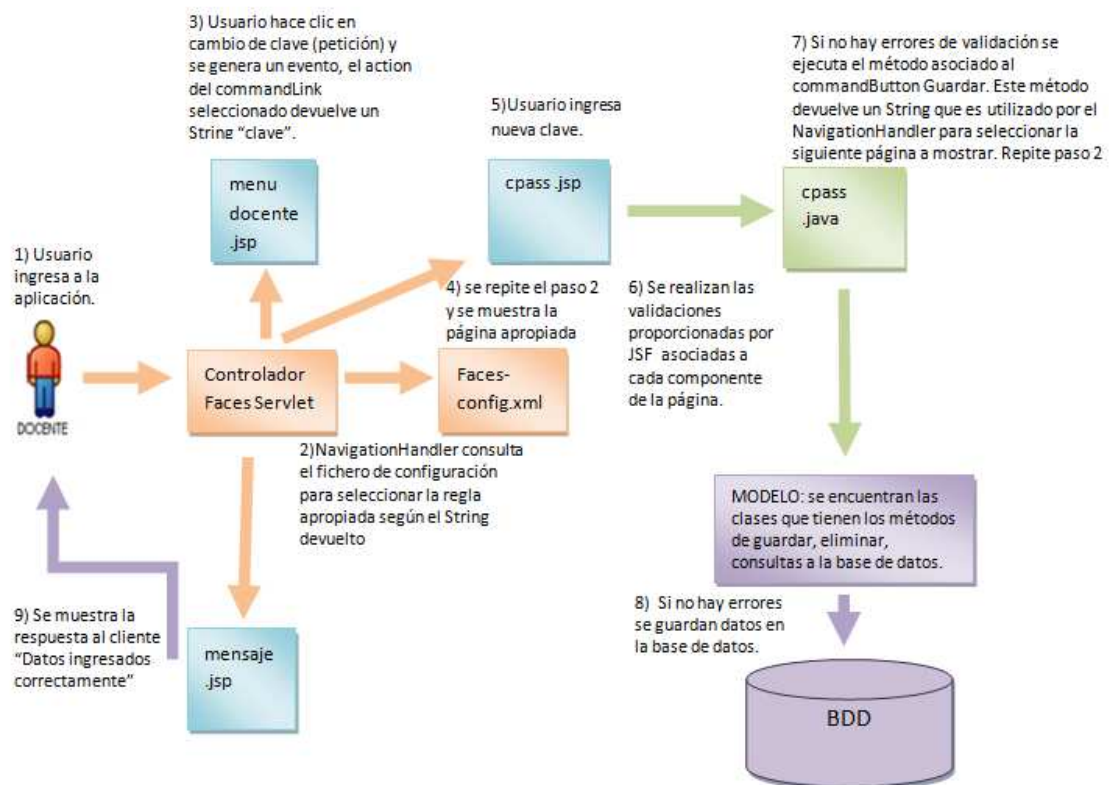


## CLASS DIAGRAM





## APPLICATION ARCHITECTURE



## CONCLUSIONS

- ✚ Web Frameworks developments mark a trend in the abstraction of the HTTP protocol, to benefit from control-based models and components and the programming of desktop applications.
- ✚ JSF is closer to the definition of the MVC pattern, and that despite working on the stateless protocol HTTP stateful UI components on the server, making it possible to create event-driven Web applications.
- ✚ As a standard JavaServer Faces JSF implementations can find both free trade which allows us to choose the most appropriate version.
- ✚ Use free tools to develop applications to reduce development costs of applications.
- ✚ Manage navigation by rules that are configured in an XML file facilitates the maintenance of an application.

- ✚ For an educational institution have a system to automate its processes is of great help to provide better and efficient service to its students along with mistakes and have all your valuable information in a database.
- ✚ Importantly, the requirements for development of the system is taken from a random school tax and based on the system they developed, it could be installed at other schools that meet the same standards under which it was developed.

## **RECOMMENDATIONS**

- ✚ To reduce costs in developing an application it is best to use free software tools.
- ✚ Installing the system before the end of development enables users to make suggestions about which features can be incorporated into the system. It also allows users to gradually become familiar with the system and be provided to use once it ready to put it into operation.
- ✚ Conduct training for system users trying not to use technical terms, make it as understandable to them and take their views into account on the system. Explaining that the use of the system will facilitate their work and that it is necessary for the institution.