

Facultad de Ingeniería de Sistemas y Electrónica

Ingeniería de Sistemas e Informática

Trabajo de Suficiencia Profesional:

“Automatización del Sistema de Matrículas de la institución Grupo Educativa con una Aplicación Web”

Bachilleres:

Darwin Bruno Quispe Cutipa

Horry Yhonatan Cabana Quispe

Carlos Alberto Pauca Ramos

Para optar el Título profesional de Ingeniero de Sistemas e Informática

Arequipa – Perú

2017

Todos los derechos reservados. Es prohibida la reproducción total o parcial del trabajo sin la autorización de la Universidad, de los autores y de los asesores.

**Darwin Bruno Quispe Cutipa**

**Horry Yhonatan Cabana Quispe**

**Carlos Alberto Pauca Ramos**

Graduados en la Universidad Tecnológica del Perú, Carrera Profesional de Sistemas e Informática.

**AGRADECIMIENTOS**

Agradecemos profundamente a nuestro asesor Ing. Bruno Jesús Juárez Valdivia por su esfuerzo, apoyo y dedicación en su asesoramiento. Agradecer su orientación y su paciencia el cual permitió la culminación de la tesis, agradecer de igual forma su responsabilidad, seriedad académica los que nos permitió una formación completa.

A nuestra Universidad Tecnológica del Perú, nuestra alma mater, y a todos los Ingenieros que nos compartieron sus conocimientos en el transcurso de la realización de nuestra tesis, gracias por prepararnos para un futuro competitivo.

Del mismo modo a nuestras familias que nos impulsaron cada día para no rendirnos jamás ante ninguna situación, en cada etapa de nuestras vidas nos han proporcionado fortaleza para seguir adelante siempre,

A todos ustedes, ¡Gracias!

**RESUMEN**

Actualmente las entidades del rubro educativo tanto públicas como como privadas tienden a utilizar los avances tecnológicos, la informática se ha vuelto una herramienta indispensable para el avance y desarrollo de entidades al servicio de la comunidad y para este fin es necesario adoptar ciertas tecnologías en instituciones educativas automatizando y optimizando los procesos y recursos mediante herramientas de software libre, es así que conociendo la gran importancia que brinda un sistema informático se presenta una aplicación web que automatiza el proceso de matrículas, registro de pagos, asistencias y actividades académicas de la institución “Grupo Educativa”, el cual llega a convertirse en un aliado eficaz para la administración de la información en lo que respecta básicamente a matrículas, registro de pagos por concepto tanto de matrícula como de pensiones, generación de fichas de matrícula, registro y reporte de asistencias de los alumnos, gestión de recuperación de clases de los alumnos, registro de temas académicos, reportes dinámicos con la opcionalidad de ser exportados en formato Excel y/o PDF, dando así al personal una experiencia satisfactoria en el uso de este sistema web desarrollado en lenguaje JAVA con motor de base de datos MySQL y a la entidad educativa un mejor panorama mejorando su imagen corporativa frente a diferentes entidades educativas.

**Palabras claves**: Automatización, proceso, sistema de matrícula, web, registro, gestión.

**ABSTRACT**

At present, public and private educational entities tend to use technological advances, computing has become an indispensable tool for the advancement and development of entities serving the community and for this purpose it is necessary to adopt certain technologies in educational institutions automating and optimizing processes and resources using free software tools, so knowing the great importance of a computer system is presented a web application that automates the process of enrollment, registration of payments, attendance and academic activities of an educational entity, which becomes an effective ally for the administration of information in terms of enrollment, registration of payments for both enrollment and pension, generation of enrollment forms, registration and reporting of student attendance, class recovery management of students, registration of academic subjects, dynamic reports with the possibility of being exported in Excel and/or PDF format, thus giving the staff a satisfactory experience in the use of this web system developed in JAVA language with MySQL database engine and the educational entity a better panorama improving its corporate image in front of different educational entities.

**Keywords**: Automation, process, system of registration, web, registry, management.

CONTENIDO

[1.INTRODUCCIÓN 1](#_Toc485656584)

[1.1.Planteamiento del Problema 1](#_Toc485656585)

[1.2.Justificación 1](#_Toc485656586)

[1.3.Objetivos 4](#_Toc485656587)

[1.4.Alcances y limitaciones 5](#_Toc485656588)

[2.MARCO TEÓRICO 8](#_Toc485656589)

[2.1Antecedentes 8](#_Toc485656592)

[2.2Bases Teóricas 8](#_Toc485656593)

[2.2.1Sistemas de Información 8](#_Toc485656594)

[2.2.2JAVA 9](#_Toc485656595)

[2.2.3MySQL 9](#_Toc485656596)

[2.2.4UML 10](#_Toc485656597)

[2.2.5Diccionario de Datos 11](#_Toc485656598)

[2.2.6Programación Orientada Objetos 11](#_Toc485656599)

[2.2.7Primefaces 11](#_Toc485656600)

[2.2.8DAO (Patrón de Diseño) 12](#_Toc485656601)

[2.2.9Facade (Patrón de Diseño) 12](#_Toc485656602)

[2.2.10Singleton (Patrón de Diseño) 13](#_Toc485656603)

[2.2.11Java Server Face 13](#_Toc485656604)

[3.PROPUESTA DE APLICACIÓN PROFESIONAL 14](#_Toc485656605)

[3.1.Descripción de la propuesta 14](#_Toc485656606)

[3.2.Recursos 14](#_Toc485656607)

[3.2.1.Recursos Humanos 14](#_Toc485656608)

[3.2.2.Recursos operacionales 15](#_Toc485656609)

[3.3.Estimación 16](#_Toc485656610)

[3.4.Planificación 18](#_Toc485656611)

[4.METODOLOGÍA DE DESARROLLO DEL PROYECTO 19](#_Toc485656612)

[5.ANÁLISIS Y DISEÑO 21](#_Toc485656613)

[5.1.Análisis del Sistema 21](#_Toc485656614)

[5.1.1.Especificación de Requerimientos 21](#_Toc485656615)

[5.2.Diseño del Aplicativo 38](#_Toc485656616)

[5.2.1.Arquitectura del Aplicativo 38](#_Toc485656617)

[5.2.2.Casos de Uso 43](#_Toc485656618)

[5.2.3.Diagramas de Clases 63](#_Toc485656619)

[5.2.4.Diagrama de Secuencia 69](#_Toc485656620)

[5.2.5.Diagrama de Componentes 80](#_Toc485656621)

[5.2.6.Diseño de interfaces 83](#_Toc485656622)

[5.2.7.MER……………… 84](#_Toc485656623)

[6.ASEGURAMIENTO DE LA CALIDAD 85](#_Toc485656624)

[6.1.Plan de Pruebas 85](#_Toc485656625)

[6.2.Cheklist de pruebas 87](#_Toc485656626)

[7.RESULTADOS 89](#_Toc485656627)

[7.1.Cuadro estadístico 89](#_Toc485656628)

[7.2.Encuestas de satisfacción 90](#_Toc485656629)

[CONCLUSIONES 94](#_Toc485656630)

[Referencias Bibliográficas 96](#_Toc485656631)

[ANEXOS 98](#_Toc485656632)

[1.MANUAL DEL SISTEMA 98](#_Toc485656633)

[2.MANUAL DE USUARIO 98](#_Toc485656634)

[3.DOCUMENTO DE PRUEBAS 98](#_Toc485656635)

[4.GUÍA DE INSTALACIÓN 98](#_Toc485656636)

[5.DICCIONARIO DE DATOS 98](#_Toc485656637)

[6.ENCUESTAS REALIZADAS 98](#_Toc485656638)

[7.DOCUMENTO DE ARQUITECTURA DE SOFTWARE 98](#_Toc485656639)

[8.DOCUMENTO DE ESPECIFICACIÓN DE REQUISITOS 98](#_Toc485656640)

**INDICE DE TABLAS**

[Tabla 1 Recursos Operacionales (FUENTE PROPIA) 15](#_Toc486507513)

[Tabla 2 Recurso Operacional de Pruebas (FUENTE PROPIA) 15](#_Toc486507514)

[Tabla 3 Funcionalidades del Sistema (FUENTE PROPIA) 16](#_Toc486507515)

[Tabla 4 Horas de trabajo (FUENTE PROPIA) 17](#_Toc486507516)

[Tabla 5 Diagrama de Gantt (FUENTE PROPIA) 18](#_Toc486507517)

[Tabla 6 Definiciones, Acrónimos y Definiciones (FUENTE PROPIA) 22](#_Toc486507518)

[Tabla 7 Características del usuario: Administrador de Sistema (FUENTE PROPIA) 22](#_Toc486507519)

[Tabla 8 Características del usuario: Director (FUENTE PROPIA) 22](#_Toc486507520)

[Tabla 9 Características del usuario: Secretaria (FUENTE PROPIA) 23](#_Toc486507521)

[Tabla 10 Características del usuario: Profesor (FUENTE PROPIA) 23](#_Toc486507522)

[Tabla 11 Requerimiento Funcional 01 (FUENTE PROPIA) 24](#_Toc486507523)

[Tabla 12 Requerimiento Funcional 02 (FUENTE PROPIA) 26](#_Toc486507524)

[Tabla 13 Requerimiento funcional 03 (FUENTE PROPIA) 27](#_Toc486507525)

[Tabla 14 Requerimiento funcional 04 (FUENTE PROPIA) 28](#_Toc486507526)

[Tabla 15 Requerimiento funcional 05 (FUENTE PROPIA) 29](#_Toc486507527)

[Tabla 16 Requerimiento funcional 06 (FUENTE PROPIA) 29](#_Toc486507528)

[Tabla 17 Requerimiento funcional 07 (FUENTE PROPIA) 30](#_Toc486507529)

[Tabla 18 Requerimiento funcional 08 (FUENTE PROPIA) 31](#_Toc486507530)

[Tabla 19 Requerimiento funcional 09 (FUENTE PROPIA) 32](#_Toc486507531)

[Tabla 20 Requerimiento funcional 10 (FUENTE PROPIA) 33](#_Toc486507532)

[Tabla 21 Requerimiento funcional 11 (FUENTE PROPIA) 33](#_Toc486507533)

[Tabla 22 Requerimiento No funcional 01 (FUENTE PROPIA) 35](#_Toc486507534)

[Tabla 23 Requerimiento No funcional 02 (FUENTE PROPIA) 36](#_Toc486507535)

[Tabla 24 Requerimiento No funcional 03 (FUENTE PROPIA) 36](#_Toc486507536)

[Tabla 25 Requerimiento No funcional 04 (FUENTE PROPIA) 37](#_Toc486507537)

[Tabla 26 Requerimiento No funcional 05 (FUENTE PROPIA) 37](#_Toc486507538)

[Tabla 27 Requerimiento No funcional 06 (FUENTE PROPIA) 38](#_Toc486507539)

[Tabla 28 Definiciones, acrónimo y abreviaciones de Arquitectura (FUENTE PROPIA) 38](#_Toc486507540)

[Tabla 29 Características del usuario: Administrador de Sistema (FUENTE PROPIA) 41](#_Toc486507541)

[Tabla 30 Características del usuario: Director (FUENTE PROPIA) 41](#_Toc486507542)

[Tabla 31 Características del usuario: Secretaria (FUENTE PROPIA) 42](#_Toc486507543)

[Tabla 32 Características del usuario: Profesor (FUENTE PROPIA) 42](#_Toc486507544)

[Tabla 33 Vistas de Arquitectura (FUENTE PROPIA) 42](#_Toc486507545)

[Tabla 34 Descripción de Casos de Uso: Gestionar Usuarios (FUENTE PROPIA) 43](#_Toc486507546)

[Tabla 35 Descripción de Casos de Uso: Asignar Roles (FUENTE PROPIA) 45](#_Toc486507547)

[Tabla 36 Descripción de Casos de Uso: Gestionar Actividades (FUENTE PROPIA) 46](#_Toc486507548)

[Tabla 37 Descripción de Casos de Uso: Gestionar Horarios de Recuperación (FUENTE PROPIA) 48](#_Toc486507549)

[Tabla 38 Descripción de Casos de Uso: Asignar Horarios (FUENTE PROPIA) 50](#_Toc486507550)

[Tabla 39 Descripción de Casos de Uso: Gestionar Montos (FUENTE PROPIA) 51](#_Toc486507551)

[Tabla 40 Descripción de Casos de Uso: Asignar Profesores (FUENTE PROPIA) 53](#_Toc486507552)

[Tabla 41 Descripción de Casos de Uso: Gestionar Matriculas (FUENTE PROPIA) 54](#_Toc486507553)

[Tabla 42 Descripción de Casos de Uso: Gestionar Estados de Pago (FUENTE PROPIA) 56](#_Toc486507554)

[Tabla 43 Descripción de Casos de Uso: Gestionar Asistencia de Alumnos (FUENTE PROPIA) 57](#_Toc486507555)

[Tabla 44 Descripción de Casos de Uso: Gestionar Reportes (FUENTE PROPIA) 59](#_Toc486507556)

[Tabla 45 Descripción de Casos de Uso: Consultar Alumno (FUENTE PROPIA) 61](#_Toc486507557)

[Tabla 46 Cuadro de Descripción de Módulos (FUENTE PROPIA) 81](#_Toc486507558)

[Tabla 47 Descripción de Componentes (FUENTE PROPIA) 82](#_Toc486507559)

[Tabla 48 Recurso del Sistema de Pruebas (FUENTE PROPIA) 86](#_Toc486507560)

[Tabla 49 Configuración de Ambiente de Prueba (FUENTE PROPIA) 86](#_Toc486507561)

[Tabla 50 Plantilla de Caso de Pruebas (FUENTE PROPIA) 87](#_Toc486507562)

[Tabla 51 Modelo de encuesta de Satisfacción 90](#_Toc486507563)

**INDICE DE ILUSTRACIONES**

[Ilustración 1 Arquitectura Modelo Vista Controlador 2](#_Toc486507564)

[Ilustración 2 Fases de metodología en cascada 19](#_Toc486507565)

[Ilustración 3 Modelo de Negocio (FUENTE PROPIA) 40](#_Toc486507566)

[Ilustración 4 Modelo Vista Controlador 41](#_Toc486507567)

[Ilustración 5 Diagrama de Casos de uso(FUENTE PROPIA) 43](#_Toc486507568)

[Ilustración 6 Diagrama de Clases: Modulo Alumno (FUENTE PROPIA) 63](#_Toc486507569)

[Ilustración 7 Diagrama de Clases: Modulo Usuarios (FUENTE PROPIA) 64](#_Toc486507570)

[Ilustración 8 Diagrama de Clases: Modulo Montos (FUENTE PROPIA) 65](#_Toc486507571)

[Ilustración 9 Diagrama de Clases: Modulo Matricula (FUENTE PROPIA) 66](#_Toc486507572)

[Ilustración 10 Diagrama de Clases: Modulo Asistencia (FUENTE PROPIA) 67](#_Toc486507573)

[Ilustración 11 Diagrama de Clases: Modulo Actividad (FUENTE PROPIA) 68](#_Toc486507574)

[Ilustración 12 Diagrama de Clases: Modulo Pagos (FUENTE PROPIA) 68](#_Toc486507575)

[Ilustración 13 Diagrama de Secuencia: Gestión de Alumnos (FUENTE PROPIA) 69](#_Toc486507576)

[Ilustración 14 Diagrama de Secuencia: Gestión de Matriculas (FUENTE PROPIA) 70](#_Toc486507577)

[Ilustración 15 Diagrama de Secuencia: Gestión Matriculas (FUENTE PROPIA) 71](#_Toc486507578)

[Ilustración 16 Diagrama de Secuencia gestión de usuarios (FUENTE PROPIA) 72](#_Toc486507579)

[Ilustración 17 Diagrama de Secuencia gestión de usuarios (FUENTE PROPIA) 73](#_Toc486507580)

[Ilustración 18 Diagrama de Secuencia gestión de montos-descuentos (FUENTE PROPIA) 74](#_Toc486507581)

[Ilustración 19 Diagrama de Secuencia gestión de montos-establecer pagos (FUENTE PROPIA ) 75](#_Toc486507582)

[Ilustración 20 Diagrama de Componentes (FUENTE PROPIA) 80](#_Toc486507583)

[Ilustración 21 Diagrama Entidad Relación (FUENTE PROPIA) 84](#_Toc486507584)

[Ilustración 22 Cuadro Estadístico (FUENTE PROPIA) 89](#_Toc486507585)

# INTRODUCCIÓN

## Planteamiento del Problema

La institución “Grupo Educativa” cuenta con un sistema de matrículas manual apoyado en algunos casos por programas de ofimática (Excel y Word) para el registro de matrículas, pagos, asistencias y actividades académicas, esto dificulta las actividades tanto administrativas como académicas, genera excesivo papeleo y no se tienen muchas garantías de que estos registros puedan mantener y ofrecer una disponibilidad inmediata cuando se les requiera y debido al gran aumento de alumnado en Grupo Educativa con el transcurso de los años resulta obsoleto continuar con dichos procesos de forma manual puesto que son de gran importancia siendo los principales en esta institución.

## Justificación

* + 1. **Justificación Funcional**

Debido a la problemática planteada surge la necesidad de automatizar estos procedimientos para el progreso y desarrollo de la entidad, es importante que se haga uso de los medios tecnológicos (sistemas informáticos) en el ámbito educativo para agilizar sus procesos, así poder tener fluidez en sus actividades y también ayudará a almacenar la información para realizar consultas cuando el usuario requiera verificar la data guardada.

* + 1. **Justificación Técnica**
       1. **Arquitectura**

Respecto a la arquitectura de software de nuestro sistema se optó por el patrón “Modelo Vista Controlador” ya que es el más adecuado para el tipo de proyecto a implementar tomando en cuenta los beneficios y ventajas que nos otorga tales como separar los componentes de nuestra aplicación dependiendo de la responsabilidad que tienen, esto significa que, al realizar un cambio en alguna parte de nuestro código, esto no afecte otra parte del mismo. Por ejemplo, si modificamos nuestra Base de Datos, sólo deberíamos modificar el modelo encargado de los datos y el resto de la aplicación debería permanecer intacta, por otro lado permite entender la aplicación de manera ligera ya que la convierte en un paquete modular y sencilla de mantener mejorando la rapidez del desarrollo, dicho diseño modular también nos permite a los desarrolladores y diseñadores trabajar simultáneamente, a continuación, mostramos un esquema de cómo estaría nuestro sistema basado en esta arquitectura.

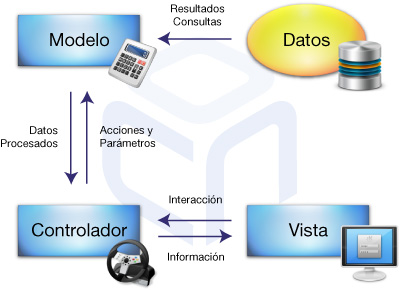


Ilustración 1 Arquitectura Modelo Vista Controlador

* + - 1. **Lenguaje de Programación**

El que vamos emplear para el desarrollo de nuestro sistema de Matriculas es JAVA, y como entorno en el que desarrollaremos será NetBeans IDE 8.2.

Netbeans proporciona el acceso a base de datos desde su propio entorno conectándonos a diversos gestores de bases de datos, entre las +cuales tenemos Oracle, MySql y demás, con estas bases de datos podemos manejarlas viendo las tablas, realizar consultas y modificaciones, y todo ello integrado en el propio IDE. También proporciona asistentes para la creación de proyectos, así como su configuración, de igual forma puedes elegir el frameworks más adecuado, y también proporciona un buen editor de código, multilenguaje y sugerencias en el código, también permite el acceso a clases digitando en el código fuente, de igual forma permite el control de versiones, una fácil ubicación de diversas clases que compone todo el código.

* + - 1. **Base de Datos**

Para nuestro proyecto se propone emplear MySQL como gestor de base de datos.

Ya que MySQL en la actualidad es una de las bases de datos más utilizadas, como usuarios tenemos a Facebook, Twitter, LinkedIn, Yahoo!, *Amazon Web Services, pues MySQL nos permite manejar grandes* volúmenes de datos, claro ejemplo tenemos los datos controlados por las redes sociales.

También debemos añadir que podemos ubicar fácilmente proveedores de hosting para albergar una base de datos MySQL a diferencia de otros gestores de base de datos (MariaDB) que son poco comunes en el mercado local. MySQL puede cumplir con las expectativas de rendimiento en la Entidad Educativa y para el tipo de disposición que se requiere es más que suficiente.

Se propone emplear también Workbench 6.3.8 como herramienta de diseño para el modelado de nuestra base de datos.

## Objetivos

* + 1. **Objetivo General**

Automatizar el proceso manual de matrículas, registro de pagos, asistencias y actividades académicas de la institución “Grupo Educativa” con una aplicación WEB.

* + 1. **Objetivos Específicos**
* Migrar la información actual de la Institución almacenada en una hoja de Excel a una base de datos en MySQL con el fin de utilizar esta información al realizar el proceso de matrículas ya con la aplicación web.
* Implementar un módulo de matrículas basándonos en la ficha física de inscripción con la que cuenta la institución educativa.
* Implementar un módulo de registro de pagos para el control adecuado de descuentos y deudas respecto a las matrículas y pensiones.
* Implementar un módulo de actividades, donde el profesor pueda gestionar las asistencias de los alumnos y las recuperaciones de las mismas.
* Registrar temas académicos de las clases a dictar que se programen para cada nivel.
* Implementar un control de accesos para cada trabajador que acceda a la aplicación web.
* Generar reportes dinámicos con filtros de búsqueda de información útil y relevante para los usuarios que pueda contar con la funcionalidad de exportarlos en archivos tanto en formato Excel como en formato PDF.
* Diseñar una Interfaz gráfica agradable la cual permita al usuario interactuar con el aplicativo de manera intuitiva.

## Alcances y limitaciones

* + 1. **Dentro de Alcance**
* La aplicación a implementar tiene como alcance la puesta en producción de este software únicamente en dicha entidad educativa y sus sedes.
* La aplicación permitirá gestionar las Matrículas propiamente dichas.
* La aplicación permitirá gestionar la Asistencia de alumnos.
* La aplicación permitirá gestionar las Actividades (Temas y Proyectos a realizar durante clases).
* La aplicación permitirá gestionar la Recuperación de clases.
* La aplicación permitirá gestionar los Estados de Pagos que se realicen tanto por Matrícula como Pensión.
* La aplicación emitirá un reporte el cual será denominado Ficha de Matrícula que vendría a reemplazar la ficha física actual que se genera al momento de realizar la matrícula de un alumno.
* La aplicación contempla la funcionalidad de tomar una fotografía instantánea al alumno durante el proceso de matriculación con el fin de anexar a su ficha de matrícula en el sistema.
* La aplicación también contempla la funcionalidad de consultas dinámicas a la base de datos respecto a información relevante para el usuario.
* Se entregará un manual de inducción al sistema.
* Se entregará la documentación del diccionario de datos.
* Dentro del alcance está el brindar las capacitaciones correspondientes a los usuarios finales.
* Optimizar los tiempos que se llevan con los actuales procedimientos manuales respecto a las matriculaciones.
* Implementar indirectamente una política de cero papeles dentro de la entidad educativa contribuyendo así con medio ambiente.
* A mediano/largo plazo indirectamente disminuir los costos operacionales fijos anuales por compras de papel, ganchos, folder, archivadores y otros ítems.
* Incrementar la satisfacción de cliente, usuario final e interno.
* Contribuir con el mejoramiento de la imagen corporativa de la empresa innovando en cuanto a procedimientos administrativos y tecnología y en el cuidado del medio ambiente.
* Está contemplada la entrega de un manual del sistema.
* Del mismo modo está contemplada la entrega de una guía de ayuda al usuario.
  + 1. **Limitaciones**
* El sistema no contempla la emisión de comprobantes tales como boletas y/o facturas.
* La implantación del sistema no implica la compra de nuevos equipos de oficina ni estudios para cambios en la estructura organizacional de la entidad educativa.
* El sistema a implementar será aplicado únicamente a Grupo Educativa en sus dos sedes debido a la particularidad de sus procedimientos por lo cual no podría ser implantado en otras entidades educativas de otro ámbito u denominación.
* Respecto a la migración de data, el proyecto no contempla el llenado de información que no se tenga de forma digital.
* La aplicación se basa en el modelo de negocio de la Institución educativa, y no propone nuevos procesos o reglas de negocio.
* El proceso de registro de pagos no está sujeto a la generación de comprobantes, así como la autogeneración de número de serie.
* El aplicativo no contempla la ejecución automática de backup’s el cual se puede realizar de forma manual.

# MARCO TEÓRICO



## Antecedentes

Grupo Educativa está ubicada en la ciudad de Arequipa contando con dos sedes, la primera en la Urb. Las Orquídeas L-3 (ASVEA) - Cercado, y la segunda en la Urb. San Pedro A-6 - Yanahuara, en conjunto ambas sedes poseen aproximadamente 260 alumnos en total.

Grupo Educativa actualmente no tiene un sistema que automatice sus procesos administrativos y académicos, por tal motivo nuestro proyecto será benéfico para la institución.

A raíz del aumento del alumnado a través del tiempo y en un afán de mejorar los procedimientos tanto administrativos como académicos se observó la necesidad de implementar una aplicación la cual permita automatizar y agilizar todo el manejo de la información en lo que respecta a los procesos principales mencionados en el objetivo general.

## Bases Teóricas

### Sistemas de Información

Es un conjunto de elementos que nos permite interactuar entre sí, para así poder apoyar en diversas actividades de un negocio u empresa contando con el apoyo de equipos computacionales que permita al sistema acceder a la información, interactuando el recurso humano con el sistema planteado, con la finalidad de operar con coordinación y eficiencia. (JOSÉ MANUEL RODRÍGUEZ RODRÍGUEZ, MARÍA JOSÉ DAUREO CAMPILLO, 2003)

### JAVA

*“Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems.”* [[1]](#footnote-1)(JAVA TM, 2014)

*Java es un lenguaje muy útil debido a la opción multiplataforma que provee (desde PC, Linux, Windows, hasta MAC, teléfonos, pocket PCs, etc.) y veremos en el transcurso de este texto como se puede optimizar la ejecución de una aplicación Java para que se aproxime a la ejecución de un binario nativo como los que se compilan con gcc*. (Pérez, 2008)

### MySQL

*“MySQL es un sistema de administración de bases de datos relacional (RDBMS). MySQL compite con sistemas RDBMS propietarios conocidos, como Oracle, SQL Server y DB2.”* [[2]](#footnote-2)(Glifillan, (2003))

Esta base de datos en particular es capaz de almacenar una enorme cantidad de datos de gran variedad y no solamente eso también puede distribuirlos y así cubrir las expectativas de cualquier tipo de organización y/o usuario en particular.

*“MySQL es la base de datos de código abierto más popular del mercado. Además, es una elección muy popular como base de datos integrada, distribuida por miles de ISV y OEM.”*[[3]](#footnote-3)(Oracle Corporation, 2014)

MySQL nos ofrece un rendimiento de calidad y que decir de su facilidad de uso que proporciona, esta base de datos se ha convertido en una empresa líder de las aplicaciones basadas en web de perfil alto, como Facebook, Twitter, YouTube y entre otros sitios web de gran importancia.

### UML

*“El Lenguaje Unificado de Modelado prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan.”* [[4]](#footnote-4)(Watson, 1998)

*Con el UML podemos modelar distintos tipos de sistemas: Sistemas de Hardware, así como los del Software, y todo el entorno de Sistemas. UML proporciona 9 diagramas que permite modelar sistemas*. (John S. Dean, Raymond H. Dean, 2009)

* Lógica de Negocio - Diagramas de Casos de Uso.
* Lógica de procesos - Diagramas de Secuencias.
* Interacción entre objetos – Diagrama de Colaboración.
* Modelo del sistema – Diagrama de Estado.
* Comportamiento Casos Uso – Diagrama de Actividades.
* Lógica del sistema- Diagrama de Clases.
* Desarrollo Completo del Sistema- Diagramas de Componentes

### Diccionario de Datos

*“El diccionario de datos almacena información sobre los datos relativos al origen de éstos, descripción, relación con otros datos, uso, responsabilidad y formato. Es la misma Base de Datos que almacena “datos sobre los datos”. El diccionario de datos es una guía y contiene el mapa de la ruta hacia la BD.”*[[5]](#footnote-5)(López, (2009))

### Programación Orientada Objetos

*“La POO es un paradigma de la programación de computadores; esto hace referencia al conjunto de teorías, estándares, modelos y métodos que permiten organizar el conocimiento, proporcionando un medio bien definido para visualizar el dominio del problema e implementar en un lenguaje de programación la solución a ese problema.”* [[6]](#footnote-6)(Microsoft, 2017)

*La particularidad de POO, es que el elemento principal es el OBJETO, pues este elemento contiene varias características y comportamientos, que nos permite verlo como una unidad y nos facilita el manejo en programación pues podemos interrelacionarlo con otros objetos de su misma clase u otras clases, tal y cual vemos en la vida real.* (John S. Dean, Raymond H. Dean, 2009)

### Primefaces

*“PrimeFaces una librería de componentes visuales de código abierto para el conjunto Java Server Faces 2.0 desarrollada y mantenida por Prime Technology.”* [[7]](#footnote-7)( Fernando Pech-May-Mario A. Gomez-Rodriguez- Luis A. de la Cruz-Diaz , (2014))

Así como otros componentes de JSF tiene con finalidad brindar un conjunto de componentes para facilitar la creación y diseño de aplicaciones web. Los componentes de PrimeFaces contiene un soporte nativo de Ajax, así podemos tener un mayor control en cada evento. Además, cuenta con un módulo adicional TouchFaces para dispositivos móviles que puedan usar navegadores basados en WebKit. (Çivici, 2014)

### DAO (Patrón de Diseño)

El patrón de diseño DAO es un componente de SW que permite generar una interfaz de recopilación de datos de algún sitio y almacenarlos en la interfaz y a la espera que alguien lo utilice.

Además, DAO encapsula la tecnología que permite acceder a los datos y separa la lógica de negocio de la de datos. Así se garantiza una portabilidad del sistema, el cual nos permite trabajar diferentes motores de base de datos.

### Facade (Patrón de Diseño)

El patrón de diseño FACADE permite simplificar la comunicación entre dos objetos como por ejemplo el objeto A y el objeto B, así su interacción sea más sencilla.

También FACADE se usa cuando queremos dividir nuestros subsistemas en subcapas, así logramos simplificar las dependencias entre las mismas y solo se comuniquen con sus fachadas.

### Singleton (Patrón de Diseño)

El Patrón de diseño SINGLETON se está diseñado para controlar que se cree una sola vez una instancia de una clase en toda la aplicación, y así tengamos un solo punto de acceso.

### Java Server Face

JSF (Java Server Faces) es un framework de desarrollo basado en el patrón MVC (Modelo Vista Controlador).

La tecnología JSP (Java Server Pages) es una especificación abierta desarrollada por Sun Microsystems como una alternativa a Active Server Pages (ASP) de Microsoft, y son un componente dominante de la especificación de Java 2 Enterprise Edition (J2EE). Muchos de los servidores de aplicaciones comercialmente disponibles (como BEA WebLogic, IBM WebSphere, Live JRun, Orion, etcétera) ya utilizan tecnología JSP. (Macario Polo,Daniel Villafranca , 2008)

# PROPUESTA DE APLICACIÓN PROFESIONAL

## Descripción de la propuesta

Se propone automatizar el proceso de matrículas, registro de pagos, asistencias y actividades académicas, los cuales actualmente se manejan de forma manual.

La automatización agilizara el manejo de la información de dichos procesos y así también aumentara el nivel tecnológico que caracteriza a la institución Grupo Educativa por impartir la enseñanza en temas de “robótica y programación” mejorando de esta manera su imagen corporativa.

## Recursos

### Recursos Humanos

* Darwin Bruno Quispe Cutipa.

Actualmente laborando en ITSC General Services - SEDAPAR como Analista Programador. Por otro lado, ocupando también el puesto de Docente Instructor de Robótica Educativa en Grupo Educativa.

* Horry Yhonatan Cabana Quispe.

Actualmente laborando en la Municipalidad Distrital de Cayarani como Asistente Administrativo y encargado del Área de Sistemas de la Municipalidad - Auditando toda la documentación financiera de la Municipalidad del periodo actual, y dando soporte al software con el que trabajan en la Municipalidad.

* Carlos Alberto Pauca Ramos.

Actualmente laborando en la empresa LAYCONSA como Administrador de Servidores y Comunicaciones, experiencia en gestión de base de datos ORACLE, SQL SERVER y MYSQL, manejo de servidores LINUX, WINDOWS SERVER. Experiencia en desarrollo de aplicaciones en PHP utilizando AJAX y BOOTSTRAP como herramienta de diseño.

### Recursos operacionales

Tabla 1 Recursos Operacionales (FUENTE PROPIA)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Recurso** | **Características** | **Aplicaciones** | **Cant.** | **Costo** |
| **Servidor de desarrollo** | Amazon AWS: Windows Server 2008 R2 64-Bit | Java 8.0  Framework JSF  NetBeans IDE 8.2  PrimeFaces 6.0  WampServer 3.0.6 64-Bit  Workbench 6.3 | 1 | S/0.00 |

Tabla 2 Recurso Operacional de Pruebas (FUENTE PROPIA)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Recurso** | **Características** | **Aplicaciones** | **Cant.** | **Costo** |
| **Servidor de pruebas** | Amazon AWS: Ubuntu Server 14.04 64-Bit | Apache Service  MySql Service  GlassFish Server  FTP | 1 | S/0.00 |

## Estimación

Estudiando los diferentes métodos de estimación nos basamos y tomamos como referencias la técnica del método de Puntos de Función, así como la estimación de tiempos basados en una metodología “ágil” generando una especie de combinación entre ambos, así generamos la siguiente estimación de nuestro proyecto de software para la Institución Grupo Educativa.

Tabla 3 Funcionalidades del Sistema (FUENTE PROPIA)

|  |  |  |  |
| --- | --- | --- | --- |
| **Funcionalidad** | **Descripción** | **Nro. F4** | **Horas** |
| F1 | Ingresar usuarios, de acuerdo a perfil y rol. | 4 | 20 |
| F2 | Buscar usuarios, de acuerdo a perfil y rol. | 3 | 15 |
| F3 | Mantener usuarios, de acuerdo a perfil y rol. | 4 | 20 |
| F4 | Controlar accesos de usuarios, de acuerdo a perfil y rol. |  | 5 |
| F5 | Emitir reportes de usuarios que muestran su estado. | 3 | 15 |
| F6 | Ingresar datos del alumnado de la institución. | 5 | 25 |
| F7 | Buscar datos del alumnado de la institución. | 4 | 20 |
| F8 | Mantener datos del alumnado de la institución. | 5 | 25 |
| F9 | Buscar sede de procedencia del alumno matriculado. | 4 | 20 |
| F10 | Ingresar matrículas del alumnado. | 5 | 25 |
| F11 | Buscar matrículas del alumnado. | 4 | 20 |
| F12 | Mantener matrículas del alumnado. | 5 | 25 |
| F13 | Registrar actividades académicas. | 5 | 25 |
| F14 | Buscar actividades académicas. | 4 | 20 |
| F15 | Mantener actividades académicas. | 6 | 30 |
| F16 | Registrar asistencia de los alumnos. | 5 | 25 |
| F17 | Buscar asistencia. de los alumnos | 4 | 20 |
| F18 | Mantener asistencia de los alumnos. | 6 | 30 |
| F19 | Emitir reporte de asistencias por alumno. | 4 | 20 |
| F20 | Registrar estado de pagos. | 3 | 15 |
| F21 | Verificar estado de pagos. | 1 | 5 |
| F22 | Registrar profesores de la institución. | 3 | 15 |
| F23 | Buscar usuarios de la aplicación. | 2 | 10 |
| F24 | Mantener usuarios de la aplicación. | 3 | 15 |
| F25 | Asignar profesores a niveles académicos. | 2 | 10 |
|  | **TOTAL** |  | **485** |

El sistema está proyectado con una duración de 5 meses y un esfuerzo de 485 horas. Teniendo como recursos 3 personas con una disponibilidad aproximada de 6 horas semanales se asignan un aproximado de 162hrs. por persona.

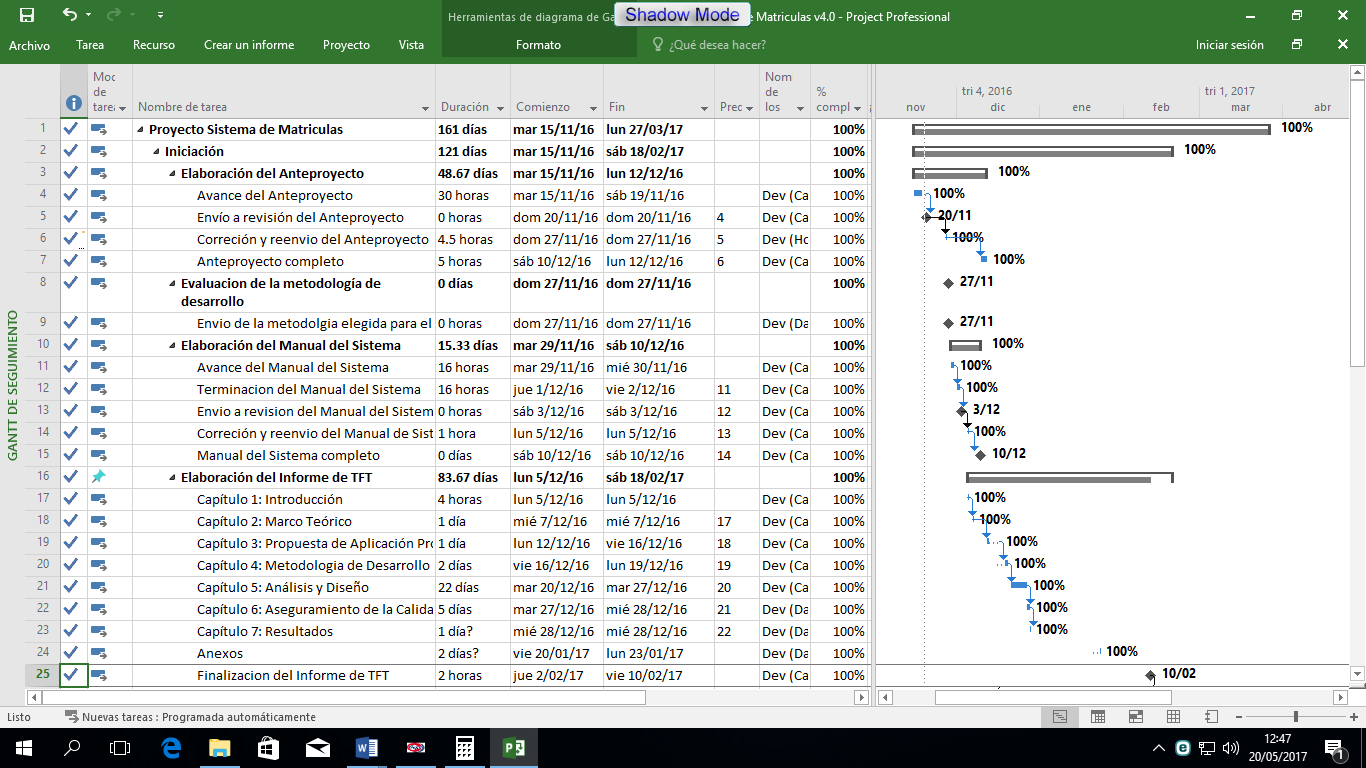
Tabla 4 Horas de trabajo (FUENTE PROPIA)

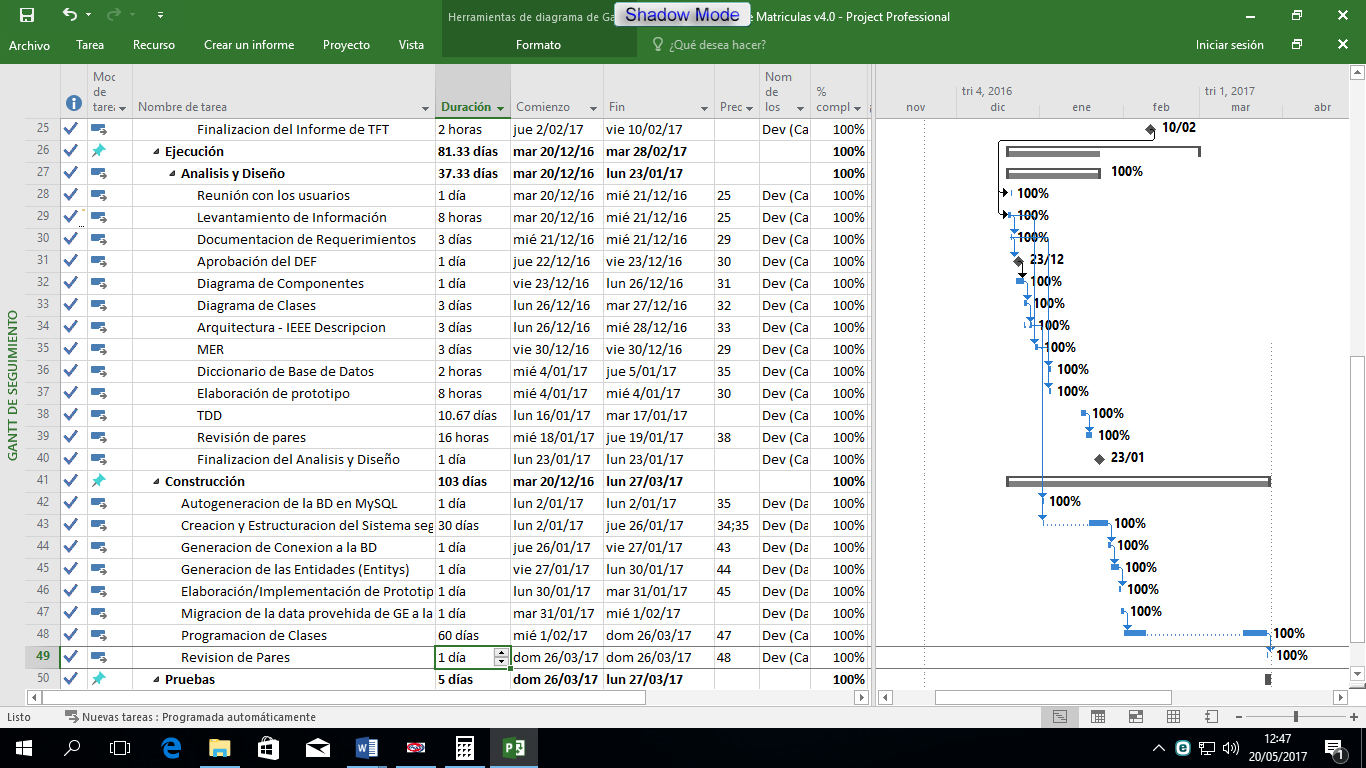
|  |  |  |
| --- | --- | --- |
|  | **Horas Mensuales** | **Duración** |
| **Recurso 1** | 162Hrs. | 5 Meses |
| **Recurso 2** | 162Hrs. |
| **Recurso 3** | 162Hrs. |
| **Total horas (Esfuerzo):** | **486Hrs.** |

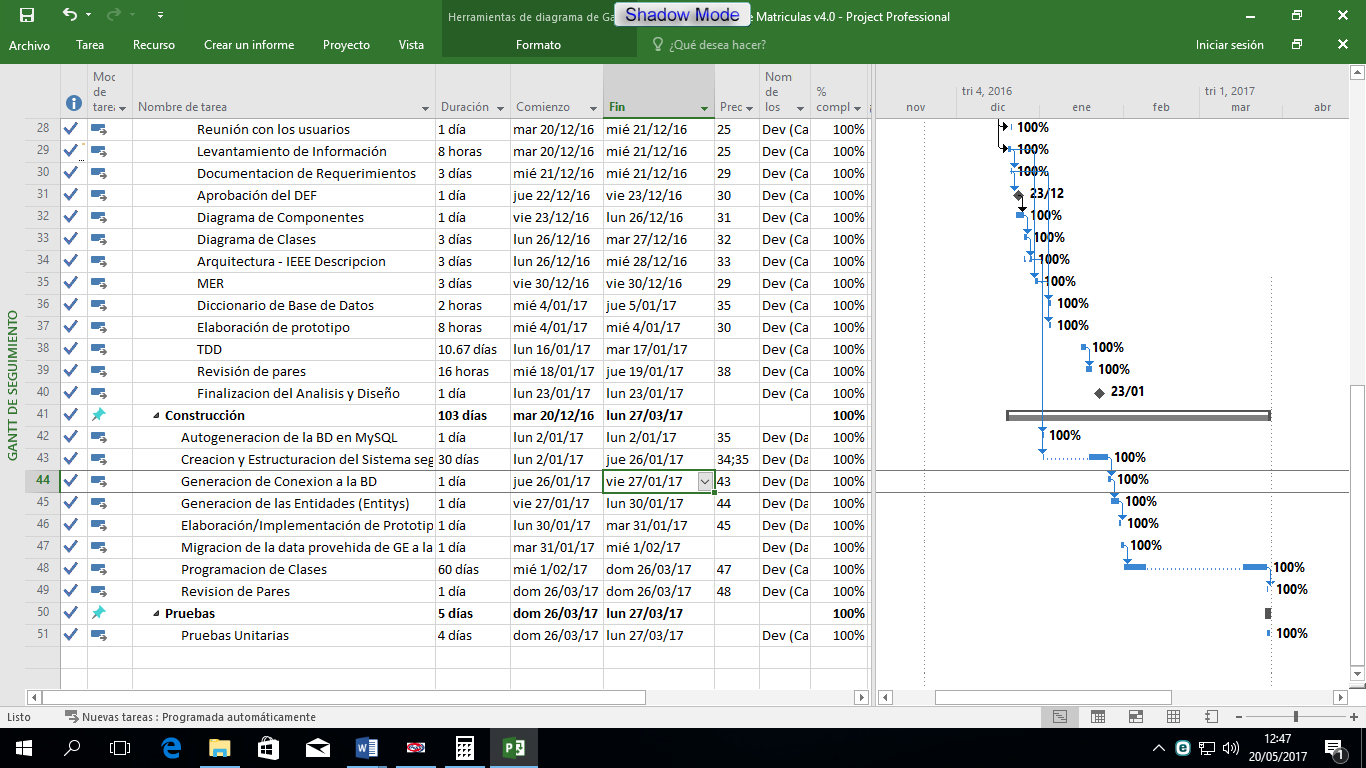
|  |  |
| --- | --- |
|  | **Costo por Hora** |
| **Recurso 1** | 1620.00 |
| **Recurso 2** | 1620.00 |
| **Recurso 3** | 1620.00 |
| **Total Soles:** | 4860.00 |

## Planificación

Tabla 5 Diagrama de Gantt (FUENTE PROPIA)







# METODOLOGÍA DE DESARROLLO DEL PROYECTO

La modelo cascada, es la metodología más tradicional para el desarrollo de proyectos, ha sido probado con el tiempo y es de fácil comprensión pero algo más importante tomando en cuenta el momento de la elección de una metodología de desarrollo para nuestro proyecto fue precisamente el tipo de proyecto a implementar debido a que se tiene proyectado aproximadamente tres meses para el desarrollo lo cual significa que es bastante compatible con esta metodología de desarrollo ya que en nuestra experiencia se tiene entendido que esta metodología se aplica y adapta mucho mejor en proyectos relativamente cortos en tiempos de desarrollo.

Como sabemos este es un modelo estático el cual nos permite gestionar los desarrollos de sistemas de forma lineal y secuencial, completando una actividad antes que la otra, de allí proviene su nombre “Cascada”. Se compone de las siguientes 5 fases: (Observe la Figura 2: Fase de metodología en Cascada).



Ilustración 2 Fases de metodología en cascada

Como se puede observar en la imagen previa, cuando una fase es desarrollada por completo se procede a continuar con la siguiente pero ya no existe la posibilidad de regresar a una fase anterior, lo que requiere indispensablemente que el proyecto tenga un enfoque estructurado y orientado a los procesos.

Este modelo es muy útil en desarrollo de sistemas estructurados donde la alteración del software después de la codificación está prohibida.

También, los datos y los procesos están usualmente separados, de manera que, si se tiene que modificar la data, el código también tiene que ser modificado. Esto hace que el software no sea reutilizable y que el sistema sea difícil de actualizar, ya que si se desea hacer algún ajuste se tiene que modificar el proceso completo lo que puede ser muy complicado y costoso.

# ANÁLISIS Y DISEÑO

## Análisis del Sistema

### Especificación de Requerimientos

* + - 1. **Propósito**

El documento de Especificación de Requerimientos tiene como propósito definir las especificaciones funcionales, no funcionales para el desarrollo del sistema de matrícula web, el cual permitirá realizar toda la lógica de negocio de la Institución.

* + - 1. **Alcances**

El presente documento está orientado al usuario del sistema, para realizar el desarrollo del sistema Web de la Institución “Grupo Educativa” automatizando el proceso de matrículas, la cual tiene por objetivo principal el gestionar los distintos procesos administrativos y académicos como son:

* Gestionar las Matrículas en base a la ficha de matrícula física.
* Gestionar la Asistencia de alumnos.
* Gestionar las Actividades (Temas y Proyectos a realizar durante clases).
* Gestionar la Recuperación de clases.
* Gestionar los Estados de Pagos que se realicen tanto por matrícula como Pensión.

Tabla 6 Definiciones, Acrónimos y Definiciones (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Usuario | Persona que usará el sistema para gestionar procesos. |
| SISMAT | Automatización del sistema de Matrículas para Grupo Educativa |
| RF | Requisito Funcional. |
| RNF | Requisito No Funcional. |
| ERS | Especificación de requisitos de software |
| Gestionar | Consultar, insertar o modificar información de la base de datos. |
| Cliente | Padre de familia del alumno o interesado en llevar las asignaturas. |

* + - 1. **Características de los Usuarios**

Tabla 7 Características del usuario: Administrador de Sistema (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Tipo de Usuario** | Administrador de Sistema |
| **Formación** | Ing. de Sistemas |
| **Actividades** | Gestión de usuarios, genera reporte de usuarios. |

Tabla 8 Características del usuario: Director (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Tipo de Usuario** | Director |
| **Formación** | Educador |
| **Actividades** | Registra pagos y matrículas, edita datos de los alumnos, establece montos, establece descuentos, asigna alumnos a los profesores y genera reportes de ficha de matrícula, alumnos, deudores, alumnos inactivos, cumpleaños de alumnos y asistencias. |

Tabla 9 Características del usuario: Secretaria (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Tipo de Usuario** | Secretaria |
| **Formación** | Administrativa |
| **Actividades** | Registra pagos y matrículas, edita datos de los alumnos, establece descuentos, asigna alumnos a los profesores y genera reportes de ficha de matrícula, alumnos, deudores, alumnos inactivos, cumpleaños de alumnos y asistencias. |

Tabla 10 Características del usuario: Profesor (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Tipo de Usuario** | Profesor |
| **Formación** | Educador |
| **Actividades** | Registra asistencias, registrar recuperación de clases, registrar actividades, genera reporte de inasistencias. |

* + - 1. **Restricciones**
* Es obligatorio contar con una conexión a internet para proceder al uso del sistema.
* La velocidad del sistema se verá afectada de acuerdo al ancho de banda que posea el usuario.
* El sistema será diseñado bajo el modelo vista controlador.
* El servidor Web debe ser capaz de soportar tecnologías JAVA (PrimeFaces, JEE, JSF) HTML.
* El servidor Web debe permitir el funcionamiento de una base de datos en MySql.
* El servidor Web debe soportar consultas concurrentes a la base de datos.
* El servidor web o cuenta web hosting deben soportar SSL.

#### Requerimientos funcionales

Tabla 11 Requerimiento Funcional 01 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF01 |
| **Nombre del requerimiento** | Gestionar usuarios |
| **Características** | Solo el administrador del sistema podrá gestionar usuarios. |
| **Descripción del requerimiento** | Permitirá al administrador del sistema registrar y/o modificar el nombre de usuario que deberá seguir los siguientes parámetros:   * Para usuarios del tipo director. - Sera de la forma “director\_correlativo/local”, donde el correlativo es la cantidad más uno de usuarios tipo director y local es la letra inicial de la sede al que corresponde el usuario (y=Yanahuara, a=Asvea). Ejemplo: directort\_1y. * Para usuarios del tipo secretaria. - Sera de la forma “secretaria\_correlativo/local”, donde el correlativo es la cantidad más uno de usuarios tipo secretaria y local es la letra inicial de la sede al que corresponde el usuario (y=Yanahuara, a=Asvea). Ejemplo: secretaria\_3y. * Para usuarios del tipo profesor. - Sera de la forma “categoría\_correlativo/local”, donde categoría es el nivel de enseñanza que impartirá el profesor (early, beginner, elementary, intermediate), el correlativo es la cantidad más uno de usuarios tipo profesor y local es la letra inicial de la sede al que corresponde el usuario (y=Yanahuara, a=Asvea). Ejemplo: early\_1a, early\_2a. * Para usuarios del tipo administrador. - Sera de la forma “admin\_correlativo/local”, donde el correlativo es la cantidad más uno de usuarios tipo administrador y local es la letra inicial de la sede al que corresponde el usuario (y=Yanahuara, a=Asvea). Ejemplo: admin\_1a.   Además ingresará el nombre completo del usuario, contraseña de acceso y seleccionara el tipo de usuario. |
| **Dependencias** | RF02  RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Alta |

Tabla 12 Requerimiento Funcional 02 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF02 |
| **Nombre del requerimiento** | Permitir acceso de acuerdo a rol |
| **Características** | Solo el administrador del sistema gestionara los accesos a los usuarios de acuerdo al rol. |
| **Descripción del requerimiento** | El aplicativo debe permitir el ingreso a los usuarios de acuerdo al rol de usuario. |
| **Dependencias** | RNF01  RNF03  RNF04  RNF05  RNF06 |
| **Prioridad de Requerimiento** | Alta |

Tabla 13 Requerimiento funcional 03 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF03 |
| **Nombre del requerimiento** | Gestionar matriculas |
| **Características** | Los usuarios con roles específicos (director y secretaria) podrán gestionar las matrículas de los alumnos. |
| **Descripción del requerimiento** | Permitirá al director y secretaria registrar y/o modificar la matrícula de un alumno, donde deberá ingresar datos como: DNI, nombre, apellidos, fecha de nacimiento, edad, colegio de procedencia, domicilio, distrito, provincia, departamento, datos del padre o apoderado, datos de responsable que recogerá al alumno en caso de que tenga movilidad, nivel al que ingresara, sede, fecha de comienzo y fin del curso, el día en que tomara clases y el turno, foto del alumno que será realizada por el aplicativo en el momento que se haga la matrícula y detalle del pago de matrícula. |
| **Dependencias** | RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Alta |

Tabla 14 Requerimiento funcional 04 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF04 |
| **Nombre del requerimiento** | Asignar horarios |
| **Características** | Los usuarios con roles específicos (secretaria y director) podrán asignar horarios de estudio para los alumnos |
| **Descripción del requerimiento** | Permitirá a la secretaria y director asignar los horarios para los alumnos así como la modificación de los mismos, ingresando datos como: sede, fecha de inicio, fecha fin, día y turno. |
| **Dependencias** | RF03  RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Alta |

Tabla 15 Requerimiento funcional 05 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF05 |
| **Nombre del requerimiento** | Gestionar horario de recuperación de estudio. |
| **Características** | Los usuarios con roles específicos (profesor) podrán gestionar el horario de recuperación de estudio. |
| **Descripción del requerimiento** | Permitirá al profesor asignar un horario de recuperación de estudio a los alumnos que tuvieron problemas para asistir a sus clases. |
| **Dependencias** | RF04  RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Media |

Tabla 16 Requerimiento funcional 06 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF06 |
| **Nombre del requerimiento** | Gestionar actividades |
| **Características** | Los usuarios con roles específicos (profesor) podrán gestionar las actividades de los alumnos. |
| **Descripción del requerimiento** | Permitirá al profesor asignar a cada alumno: talleres, temas del dictado de clases y tiempo de duración de la actividad así como la modificación de las actividades. |
| **Dependencias** | RF03  RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Alta |

Tabla 17 Requerimiento funcional 07 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF07 |
| **Nombre del requerimiento** | Gestionar asistencia e inasistencias de alumnos |
| **Características** | Los usuarios con roles específicos (profesor) podrán gestionar las asistencias e inasistencias de los alumnos. |
| **Descripción del requerimiento** | Permitirá a los profesores registrar la asistencia de los alumnos, así como la modificación de la misma. Además de poder consultar los alumnos que tienen inasistencias con la finalidad de que estos recuperen clases otro día. |
| **Dependencias** | RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Alta |

Tabla 18 Requerimiento funcional 08 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF08 |
| **Nombre del requerimiento** | Consultar alumnos |
| **Características** | Los usuarios con roles específicos (secretaria y director) podrán buscar los datos de los alumnos. |
| **Descripción del requerimiento** | Permitirá a la secretaria y/o director realizar búsquedas de los alumnos por DNI, colegio de procedencia, nivel, edad, inasistencias, deuda, por fecha de matrícula y por onomástico del mes en el que se ejecute la búsqueda. |
| **Dependencias** | RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Alta |

Tabla 19 Requerimiento funcional 09 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF09 |
| **Nombre del requerimiento** | Gestionar montos |
| **Características** | Los usuarios con roles específicos (director) podrán gestionar los montos de matrícula y pensión. |
| **Descripción del requerimiento** | Permitirá a la secretaria y/o director registrar los montos de pago y/o descuento, este último puede ser en porcentaje o en soles, así como la modificación de los mismos. |
| **Dependencias** | RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Alta |

Tabla 20 Requerimiento funcional 10 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF10 |
| **Nombre del requerimiento** | Asignar profesores |
| **Características** | Los usuarios con roles específicos (secretaria y director) podrán asignar profesores. |
| **Descripción del requerimiento** | Permitirá a la secretaria y/o director asignar profesores a determinados días, turnos y cursos. |
| **Dependencias** | RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Alta |

Tabla 21 Requerimiento funcional 11 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF11 |
| **Nombre del requerimiento** | Generar reportes |
| **Características** | Los usuarios con roles específicos (administrador del sistema, secretaria y director) podrán generar reportes. |
| **Descripción del requerimiento** | El sistema permitirá generar reportes de asistencia por alumno, profesores, estados de pago, lista de usuarios y ficha de matrícula. |
| **Dependencias** | RNF01  RNF03  RNF04  RNF05  RF03 |
| **Prioridad de Requerimiento** | Alta |

Tabla 22 Requerimiento funcional 12 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF12 |
| **Nombre del requerimiento** | Generar ficha de matricula |
| **Características** | Los usuarios con roles específicos (director y secretaria) podrán generar una ficha de matrícula. |
| **Descripción del requerimiento** | Permitirá al director y secretaria generar una ficha de matrícula, previamente a ello tendrá la opción de ingresar observaciones que se verán reflejadas en la ficha. La mencionada se generará en formato PDF. |
| **Dependencias** | RF03  RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Alta |

Tabla 23 Requerimiento funcional 13 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RF13 |
| **Nombre del requerimiento** | Registrar pagos |
| **Características** | Los usuarios con roles específicos (director y secretaria) podrán registrar pagos. |
| **Descripción del requerimiento** | Permitirá al director y/o secretaria registrar los pagos de pensión y matricula de cada alumno, ingresando de manera manual el número de comprobante y la posibilidad de aplicar descuentos en soles o en porcentaje. |
| **Dependencias** | RNF01  RNF03  RNF04  RNF05 |
| **Prioridad de Requerimiento** | Alta |

#### Requerimientos no funcionales

Tabla 22 Requerimiento No funcional 01 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RNF01 |
| **Nombre del requerimiento** | Interfaz de la aplicación |
| **Características** | El aplicativo presentara una interfaz para que el usuario interactúe con el mismo. |
| **Descripción del requerimiento** | El aplicativo presentara una interfaz sencilla e intuitiva para que los usuarios puedan hacer un fácil manejo del mencionado. |
| **Prioridad de Requerimiento** | Alta |

Tabla 23 Requerimiento No funcional 02 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RNF02 |
| **Nombre del requerimiento** | Mantenimiento |
| **Características** | El aplicativo contara con un manual de instalación y un manual de usuario que podrán ser usados para su correcto uso. |
| **Descripción del requerimiento** | El aplicativo contara con documentación detallada de cada operación que pueda realizarse en el mismo. |
| **Prioridad de Requerimiento** | Alta |

Tabla 24 Requerimiento No funcional 03 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RNF03 |
| **Nombre del requerimiento** | Control de accesos |
| **Características** | Cada usuario tendrá acceso y/o restricción a determinados módulos del aplicativo. |
| **Descripción del requerimiento** | El aplicativo debe permitir grupos de acceso de acuerdo al rol de usuario |
| **Prioridad de Requerimiento** | Alta |

Tabla 25 Requerimiento No funcional 04 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RNF04 |
| **Nombre del requerimiento** | Manejo de Errores |
| **Características** | Informará Errores |
| **Descripción del requerimiento** | El aplicativo debe emitir mensajes de errores. |
| **Prioridad de requerimiento** | Alta |

Tabla 26 Requerimiento No funcional 05 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RNF05 |
| **Nombre del requerimiento** | Comprobaciones de integridad |
| **Características** | El aplicativo evaluara la integridad de datos ingresados |
| **Descripción del requerimiento** | El aplicativo evaluara la integridad de datos ingresados. |
| **Prioridad de Requerimiento** | Alta |

Tabla 27 Requerimiento No funcional 06 (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Identificación del requerimiento** | RNF06 |
| **Nombre del requerimiento** | Multiplataforma |
| **Características** | El aplicativo podrá ser ejecutado en cualquier sistema operativo |
| **Descripción del requerimiento** | El aplicativo podrá ser ejecutado en cualquier sistema operativo. |
| **Prioridad de Requerimiento** | Alta |

## Diseño del Aplicativo

### Arquitectura del Aplicativo

* + - 1. **Definiciones, acrónimos y abreviaciones.**

Tabla 28 Definiciones, acrónimo y abreviaciones de Arquitectura (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| DAS | Documento de Arquitectura de Software |
| Usuario | Persona que usará el aplicativo para gestionar procesos |
| SISMAT | Automatización del sistema de Matrículas para Grupo Educativa |
| Gestionar | Consultar, insertar o modificar información de la base de datos. |
| Cliente | Padre de familia del alumno o interesado en llevar las asignaturas. |
| Vistas | Es una representación de un área de interés o perspectiva del aplicativo en alto nivel. |
| JAVA EE | Plataforma Java Edición Empresarial (Plataforma de Programación) |
| API | Interfaz de Programación de Aplicaciones (para ser utilizado por otro software) |
| Stakeholder | Individuo, equipo u organización con intereses relativos al aplicativo. |
| Modulo o Componente: | Cualquier elemento estructural abstracto, visible, externo, de alto nivel, analizable, que pueda constituir una funcionalidad de la solución del aplicativo. |
| Escenario | Especifica el comportamiento y limita el interés de un área específica del aplicativo para uno o varios Stakeholder. |
| Arquitectura de Software | Es el conjunto de elementos estáticos, propios del diseño intelectual del aplicativo, que definen y dan forma tanto al código fuente, como al comportamiento del software en tiempo de ejecución. Naturalmente este diseño arquitectónico ha de ajustarse a las necesidades y requisitos del proyecto. |

* + - 1. **Framework Conceptual****- Descripción de la arquitectura en contexto**

Este documento presenta la Arquitectura como una serie de vistas enfocadas a la Arquitectura de Software. Para el cual se realizó un escenario el cual muestra la funcionalidad del modelo de negocio de la institución GRUPO EDUCATIVA.

* + - 1. **Stakeholder y sus roles**

Este diagrama representa la identificación de Stakeholder y sus roles (Modelo del Negocio).

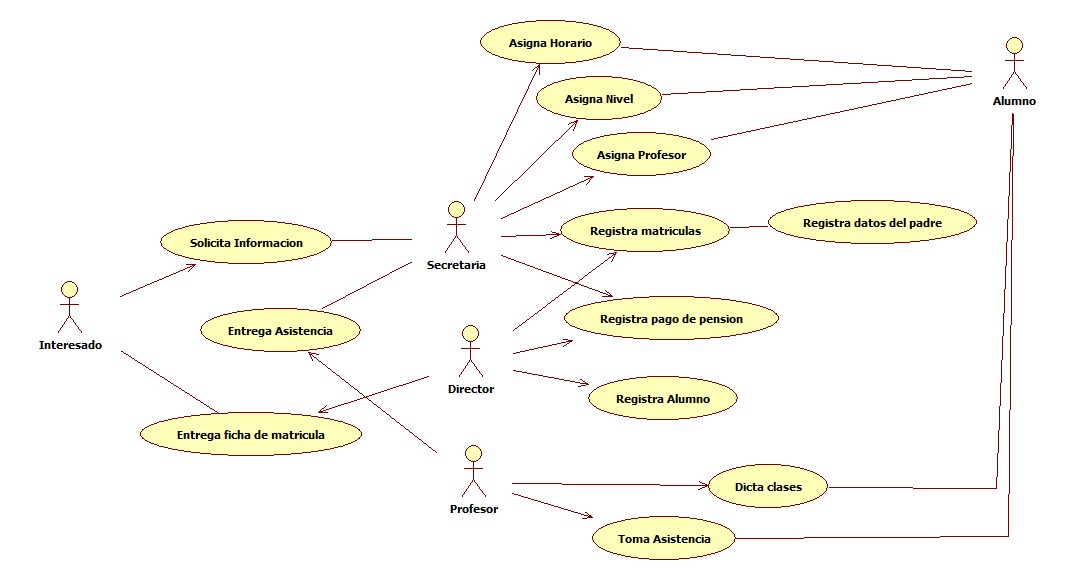


Ilustración 3 Modelo de Negocio (FUENTE PROPIA)

* + - 1. **Actividades de arquitectura en el ciclo de vida**

El arquitecto contribuye al desarrollo, operación y mantenimiento de un sistema desde su concepción inicial hasta su retiro del uso. Como tal, la arquitectura se entiende mejor en un contexto de ciclo de vida, no simplemente como una sola actividad en un punto en ese ciclo de vida.

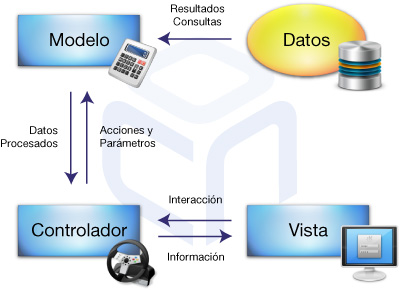


Ilustración 4 Modelo Vista Controlador

* + - 1. **Stakeholder y sus responsabilidades**

Tabla 29 Características del usuario: Administrador de Sistema (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Tipo de Usuario** | Administrador de Sistema |
| **Formación** | Ing. de Sistemas |
| **Actividades** | Gestión de usuarios, genera reporte de usuarios. |

Tabla 30 Características del usuario: Director (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Tipo de Usuario** | Director |
| **Formación** | Educador |
| **Actividades** | Registra pagos y matrículas, edita datos de los alumnos, establece montos, establece descuentos, asigna alumnos a los profesores y genera reportes de ficha de matrícula, alumnos, deudores, alumnos inactivos, cumpleaños de alumnos y asistencias. |

Tabla 31 Características del usuario: Secretaria (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Tipo de Usuario** | Secretaria |
| **Formación** | Administrativa |
| **Actividades** | Registra pagos y matrículas, edita datos de los alumnos, establece descuentos, asigna alumnos a los profesores y genera reportes de ficha de matrícula, alumnos, deudores, alumnos inactivos, cumpleaños de alumnos y asistencias. |

Tabla 32 Características del usuario: Profesor (FUENTE PROPIA)

|  |  |
| --- | --- |
| **Tipo de Usuario** | Profesor |
| **Formación** | Educador |
| **Actividades** | Registra asistencias, registrar recuperación de clases, registrar actividades, genera reporte de inasistencias. |

* + - 1. **Selección de puntos de vista de la arquitectura**

Tabla 33 Vistas de Arquitectura (FUENTE PROPIA)

|  |  |
| --- | --- |
| **VISTAS** | **UML** |
| Escenarios | Casos de Uso |
| Lógica | Clases |
| Desarrollo | Componentes |
| Procesos | Secuencia |

### Casos de Uso

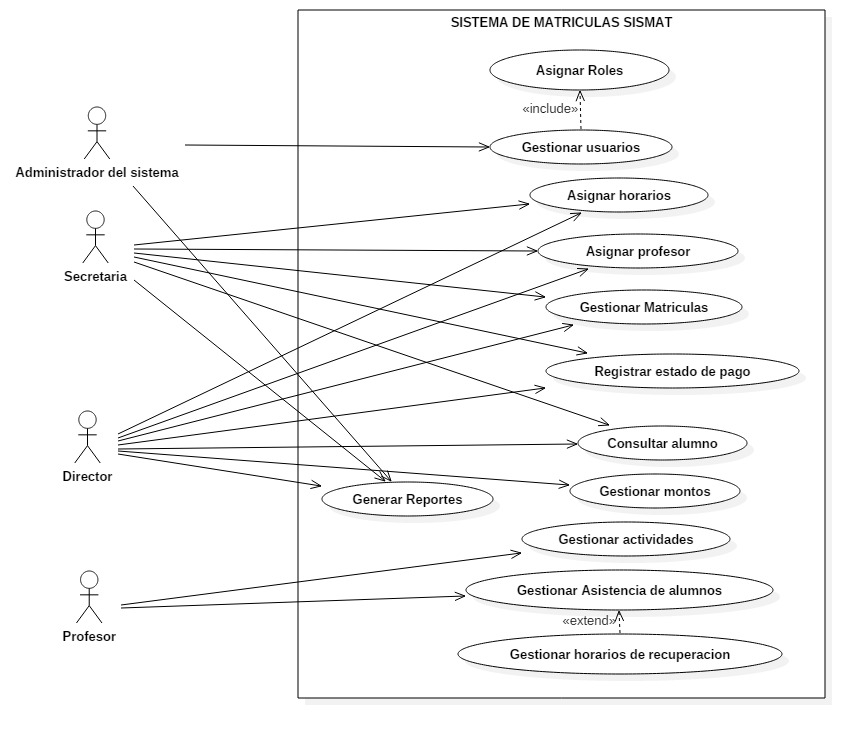
****

Ilustración 5 Diagrama de Casos de uso(FUENTE PROPIA)

* + - 1. **Descripción de casos de uso**

Tabla 34 Descripción de Casos de Uso: Gestionar Usuarios (FUENTE PROPIA)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador** | **GESTIONAR USUARIOS** | | | |
| **Descripción** | El aplicativo deberá permitir al administrador gestionar los usuarios del aplicativo como se describe a continuación: | | | |
| **Precondición** | Ninguno | | | |
| **Actores** | Administrador de Aplicativo | | | |
| **Secuencia Normal** | **Paso** | **Acción** | | |
| 1 | El administrador elige entre una lista de opciones (**agregar**, **buscar**) | | |
|  | 1a | Si el administrador elige **agregar**, el aplicativo mostrará nuevos campos que permitan el ingreso de datos del usuario. | |
|  | 2a | El administrador procederá a ingresar los datos correspondientes. | |
|  | 3a | El aplicativo comprueba la validez de los datos ingresados y los almacena. | |
|  | 1b | Si el administrador elige **buscar**, el aplicativo solicitara los parámetros de búsqueda. | |
|  | 2b | El administrador procederá a ingresar los parámetros de búsqueda. | |
|  | 3b | El aplicativo solicitará un listado de resultados a la base de datos. | |
|  | 4b | El aplicativo habilitará las opción **modificar** para la lista de resultados. | |
|  |  | 4.1a | Si el administrador elige **modificar**, el aplicativo habilitará la edición para el usuario seleccionado. |
|  |  | 4.2a | El administrador procederá a realizar los cambios necesarios. |
|  |  | 4.3a | El aplicativo comprueba la validez de los datos modificados y los almacena |
| **Post-Condición** | Asignar roles | | | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 3a, 1 al 4.3a, en un mínimo de 1 minuto el cual variará de acuerdo al ancho de banda con que cuente el usuario. | | | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida mínima de 10 veces al mes. | | | |
| **Importancia** | Vital | | | |
| **Requerimiento**  **Asociado** | RF01, RNF03 | | | |

Tabla 35 Descripción de Casos de Uso: Asignar Roles (FUENTE PROPIA)

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | **ASIGNAR ROLES** | | |
| **Descripción** | El aplicativo deberá permitir al administrador gestionar roles al aplicativo como se describe a continuación: | | |
| **Precondición** | Gestionar Usuarios | | |
| **Actores** | Administrador de Aplicativo | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | El administrador elegirá la opción **agregar** | |
|  | 1a | Si el administrador elige **agregar**, el aplicativo mostrará nuevos campos que permitan el ingreso de datos del usuario. |
|  | 2a | El administrador procederá a ingresar los datos correspondientes. |
| 2 | El administrador asignara un rol al usuario ingresado | |
| 3 | El aplicativo comprueba la validez de los datos ingresados y los almacena. | |
| **Post-Condición** | Ninguno | | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 2a, 1 al 3, en un mínimo de 1 minuto el cual variará de acuerdo al ancho de banda con que cuente el usuario. | | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida mínima de 10 veces al mes. | | |
| **Importancia** | Vital | | |
| **Requerimiento**  **Asociado** | RF02 | | |

Tabla 36 Descripción de Casos de Uso: Gestionar Actividades (FUENTE PROPIA)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador** | **GESTIONAR ACTIVIDADES** | | | |
| **Descripción** | El aplicativo deberá permitir al usuario (profesor) gestionar las actividades del estudiante como se describe a continuación: | | | |
| **Precondición** | Ninguno | | | |
| **Actores** | Profesor | | | |
| **Secuencia Normal** | **Paso** | **Acción** | | |
| 1 | El usuario elige entre una lista de opciones (**actualizar, buscar**) | | |
|  | 1a | Si el usuario elige **actualizar**, el aplicativo mostrará todas las actividades en el calendario del mes referidos a los estudiantes. | |
|  | 2a | El usuario procederá a seleccionar el día donde requiera insertar una actividad. | |
|  | 3a | El aplicativo mostrara una ventana con campos (Tema, fecha de inicio, fecha de fin) vacíos a ingresar. | |
|  | 4a | El usuario ingresara los datos requeridos y seleccionara la opción **guardar** | |
|  | 1b | Si el usuario elige **buscar**, el aplicativo solicitara los parámetros de búsqueda. | |
|  | 2b | El usuario procederá a seleccionar los parámetros de búsqueda. | |
|  | 3b | El aplicativo solicitará un listado de resultados a la base de datos. | |
|  | 4b | El aplicativo habilitará la edición de la actividad que se seleccione del día escogido. | |
|  |  | 4.1a | Si el usuario elige **Eliminar**, el aplicativo eliminara la actividad seleccionada |
|  |  | 4.2a | El aplicativo comprueba la validez de los datos modificados y los almacena al seleccionar **guardar** |
| **Post-Condición** | Ninguno | | | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 4a, 1 al 4.2a, en un mínimo de 1 minuto el cual variará de acuerdo al ancho de banda con que cuente el usuario. | | | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida mínima de 100 veces al mes. | | | |
| **Importancia** | Vital | | | |
| **Requerimiento**  **Asociado** | RF06 | | | |

Tabla 37 Descripción de Casos de Uso: Gestionar Horarios de Recuperación (FUENTE PROPIA)

|  |  |  |
| --- | --- | --- |
| **Identificador** | **GESTIONAR HORARIOS DE RECUPERACIÓN** | |
| **Descripción** | El aplicativo deberá permitir al usuario gestionar los horarios de recuperación del estudiante como se describe a continuación: | |
| **Precondición** | Ninguno | |
| **Actores** | Profesor | |
| **Secuencia Normal** | **Paso** | **Acción** |
| 1 | El usuario elige entre una lista de opciones (**inasistencias**) y selecciona **actualizar** |
| 2 | El aplicativo devolverá una lista de alumnos que tengan inasistencias |
| 3 | El usuario seleccionara el alumno que desea asignar una recuperación de clases. |
| 4 | El usuario insertará un horario de recuperación de clases. |
| 5 | El aplicativo comprueba la validez de los datos ingresados y los almacena. |
| **Post-Condición** | Ninguno | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 5, en un mínimo de 1 minuto el cual variará de acuerdo al ancho de banda con que cuente el usuario. | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida mínima de 100 veces al mes. | |
| **Importancia** | Vital | |
| **Requerimiento Asociado** | RF05 | |

Tabla 38 Descripción de Casos de Uso: Asignar Horarios (FUENTE PROPIA)

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | **ASIGNAR HORARIO** | | |
| **Descripción** | El aplicativo deberá permitir al usuario gestionar los horarios del estudiante como se describe a continuación: | | |
| **Precondición** | Ninguno | | |
| **Actores** | Secretaria, Director | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | El usuario elige la opción (**registrar matricula**) | |
|  | 1a | El usuario ingresara todos los datos solicitados por el aplicativo hasta llegar a la opción **programa**. |
|  | 2a | El usuario procederá a ingresar el día, turno, la sede y las fechas de inicio y fin de la matrícula. |
|  | 3a | El usuario seleccionara la opción **siguiente**. |
|  | 4a | El aplicativo validara lo datos y continuara con el proceso de matrícula. |
|  | 5a | Una vez ingresado todos los datos de matrícula el usuario generara la matrícula y el aplicativo guardara el horario asignado. |
| **Post-Condición** | Ninguno | | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 5a, en un mínimo de 1 minuto el cual variará de acuerdo al ancho de banda con que cuente el usuario. | | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida mínima de 5 veces al mes. | | |
| **Importancia** | Vital | | |
| **Requerimiento Asociado** | RF04 | | |

Tabla 39 Descripción de Casos de Uso: Gestionar Montos (FUENTE PROPIA)

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | **GESTIONAR MONTOS** | | |
| **Descripción** | El aplicativo deberá permitir al usuario gestionar los montos de pago y descuentos del estudiante. | | |
| **Precondición** | Ninguno | | |
| **Actores** | Director | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | El usuario elige entre una lista de opciones (**Establecer montos, descuentos**) | |
|  | 1a | Si el usuario elige **establecer montos**, el aplicativo mostrará nuevos campos que permitan el ingreso de datos del monto de matrícula y pensión. |
|  | 2a | El usuario procederá a ingresar los datos correspondientes. |
|  | 3a | El aplicativo comprueba la validez de los datos ingresados y los almacena. |
|  | 1b | Si el usuario elige **descuentos**, el aplicativo mostrara todos los estudiantes a los cuales se les puede aplicar descuentos. |
|  | 2b | El usuario procederá a ingresar el descuento a los alumnos correspondientes. |
|  | 3b | El aplicativo guardara los datos ingresados. |
|  | 4b | El usuario seleccionara la opción **actualizar** para verificar que los cambios se hicieron efectivos. |
| **Post-Condición** | Ninguno | | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 3a, 1b al 4b, en un mínimo de 1 minuto el cual variará de acuerdo al ancho de banda con que cuente el usuario. | | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida mínima de 5 veces al mes. | | |
| **Importancia** | Vital | | |
| **Requerimiento Asociado** | RF09 | | |

Tabla 40 Descripción de Casos de Uso: Asignar Profesores (FUENTE PROPIA)

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | **ASIGNAR PROFESORES** | | |
| **Descripción** | El aplicativo deberá permitir al usuario asignar los profesores encargados del dictado de clases como se describe a continuación: | | |
| **Precondición** | Ninguno | | |
| **Actores** | Secretaria, director | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | El usuario elige la opción (**administrar alumno**) | |
| 2 | El usuario elige entre una lista de opciones los siguientes datos (**horarios, días, cursos, profesores**), para luego seleccionar la opción **revisar** | |
|  | 2a | Si el usuario elige **revisar**, el aplicativo mostrará los resultados de búsqueda de la cantidad de alumnos aptos para ser asignados. |
|  | 3a | El usuario ingresar la cantidad de alumnos que desea asignar al profesor. |
|  | 4a | El aplicativo comprueba la validez de los datos ingresados y los almacena. |
| **Post-Condición** | Ninguno | | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 2, 2 al 4a, en un mínimo de 1 minuto el cual variará de acuerdo al ancho de banda con que cuente el usuario. | | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida mínima de 10 veces al mes. | | |
| **Importancia** | Vital | | |
| **Requerimiento Asociado** | RF10 | | |

Tabla 41 Descripción de Casos de Uso: Gestionar Matriculas (FUENTE PROPIA)

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | **GESTIONAR MATRICULAS** | | |
| **Descripción** | El aplicativo deberá permitir al usuario gestionar las matriculas del estudiante como se describe a continuación: | | |
| **Precondición** | Ninguno | | |
| **Actores** | Secretaria, Director | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | El usuario elige entre una lista de opciones (**registrar**) | |
|  | 1a | Si el usuario elige **nueva matricula**, el aplicativo mostrará nuevos campos que permitan el ingreso de datos de las matrículas. |
|  | 2a | El usuario procederá a ingresar los datos correspondientes. |
|  | 3a | El aplicativo comprueba la validez de los datos ingresados y los almacena. |
|  | 1b | Si el aplicativo detecta que el estudiante ya existe solicitara que se elimine el alumno para poder matricularlo. |
|  | 2b | El usuario eliminara al estudiante e ingresara solo su DNI para que los campos se completen automáticamente y podrá modificar los datos necesarios. |
|  | 4b | El aplicativo comprueba la validez de los datos ingresados y los almacena. |
|  |  | 5b | Al final del ingreso de datos el usuario podrá escoger la opción de **Administrar/Alumno/FichaMatricula** para imprimir la ficha de matrícula. |
| **Post-Condición** | Ninguno | | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 3a, 1 al 5b, en un mínimo de 1 minuto el cual variará de acuerdo al ancho de banda con que cuente el usuario. | | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida mínima de 200 veces al mes. | | |
| **Importancia** | Vital | | |
| **Requerimiento Asociado** | RF03, RF12 | | |

Tabla 42 Descripción de Casos de Uso: Gestionar Estados de Pago (FUENTE PROPIA)

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | **REGISTRAR ESTADOS DE PAGO** | | |
| **Descripción** | El aplicativo deberá permitir al usuario registrar los estados de pago de los estudiantes como se describe a continuación: | | |
| **Precondición** | Ninguno | | |
| **Actores** | Secretaria, Director | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | El usuario elige entre una lista de opciones (**registrar / pagos**) | |
|  | 1a | Si el usuario elige **registrar / pagos**, el aplicativo mostrará una lista de alumnos matriculados. |
|  | 2a | El usuario procederá a seleccionar el alumno del cual desea realizar un registro de pago y seleccionara una opción de vista. |
|  | 3a | El aplicativo mostrara un lista de campos para que pueda ser llenada por el usuario. |
|  | 4a | El usuario seleccionara la opción pagar y el aplicativo validara y almacenara los datos. |
| **Post-Condición** | Ninguno | | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 4a, en un mínimo de 1 minuto el cual variará de acuerdo al ancho de banda con que cuente el usuario. | | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida de 200 veces al mes. | | |
| **Importancia** | Vital | | |
| **Requerimiento**  **Asociado** | RF13 | | |

Tabla 43 Descripción de Casos de Uso: Gestionar Asistencia de Alumnos (FUENTE PROPIA)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador** | **GESTIONAR ASISTENCIA DE ALUMNOS** | | | |
| **Descripción** | El aplicativo deberá permitir al usuario gestionar la asistencia de alumnos como se describe a continuación: | | | |
| **Precondición** | Ninguno | | | |
| **Actores** | Profesor | | | |
| **Secuencia Normal** | **Paso** | **Acción** | | |
| 1 | El usuario elige entre una lista de opciones (**mañana 1, mañana 2, tarde, recuperación**) | | |
|  | 1a | Si el usuario elige **mañana 1**, el aplicativo mostrará una lista de alumnos asignados al profesor en el turno antes mencionado. | |
|  | 2a | El usuario procederá a ingresar los datos de asistencia. | |
|  | 3a | El aplicativo comprueba la validez de los datos ingresados y los almacena. | |
|  | 1b | Si el usuario elige **mañana 2**, el aplicativo tendrá el mismo comportamiento descrito desde el punto 1a al 3a, pero serán alumnos propios del turno “mañana 2”. | |
|  | 1c | Si el usuario elige **tarde**, el aplicativo tendrá el mismo comportamiento descrito desde el punto 1a al 3a, pero serán alumnos propios del turno “mañana 2”. | |
|  | 1d | Si el usuario elige **recuperación**, el aplicativo tendrá el mismo comportamiento descrito desde el punto 1a al 3a, pero serán alumnos propios del turno “mañana 2”. | |
|  | 1e | El aplicativo habilitará las opción **modificar** para la lista de resultados. | |
|  |  | 1.1e | El usuario podrá modificar la asistencia de los alumnos mientras no termine el día en que se tome su asistencia. |
|  |  | 1.2e | El usuario procederá a realizar los cambios necesarios. |
|  |  | 1.3e | El aplicativo comprueba la validez de los datos modificados y los almacena, el usuario selecciona la opción **actualizar** para verificar los cambios. |
| **Post-Condición** | Gestionar horarios de recuperación | | | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 3a, 1 al 1b, 1 al 1c, 1 al 1d, 1 al 1.3e, en un mínimo de 1 minuto el cual variará de acuerdo al ancho de banda con que cuente el usuario. | | | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida de 1000 veces al mes. | | | |
| **Importancia** | Vital | | | |
| **Requerimiento Asociado** | RF07 | | | |

Tabla 44 Descripción de Casos de Uso: Gestionar Reportes (FUENTE PROPIA)

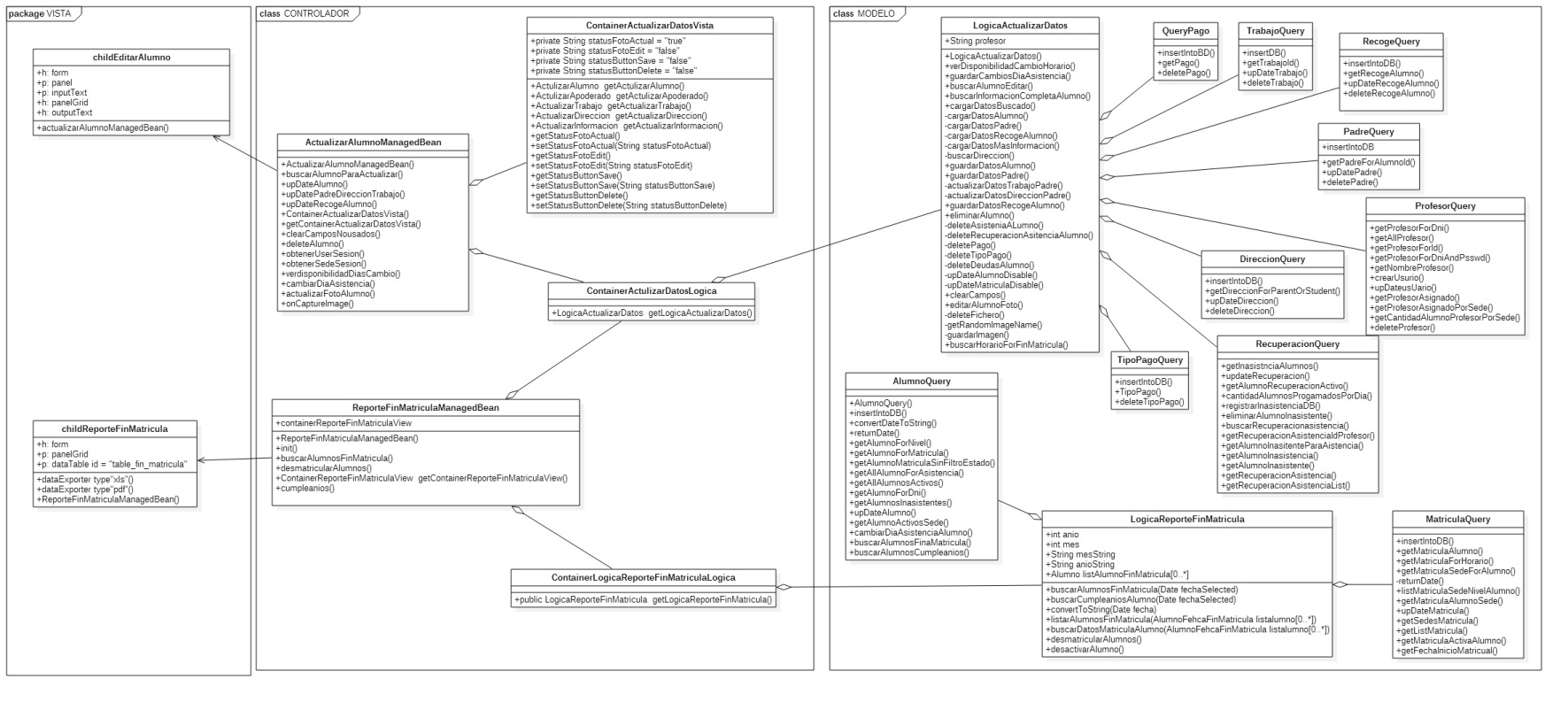
|  |  |  |
| --- | --- | --- |
| **Identificador** | **GENERAR REPORTES** | |
| **Descripción** | El aplicativo deberá permitir al usuario gestionar los reportes como se describe a continuación: | |
| **Precondición** | Ninguno | |
| **Actores** | Administrador del aplicativo, Secretaria, Director | |
| **Secuencia Normal** | **Paso** | **Acción** |
| 1 | El usuario seleccionara la opción **reportes** y elegirá la opción de reporte que desee (reporte alumnos, reporte deudores, alumnos inactivos, reporte fin de matrícula, reporte de cumpleaños, reporte de asistencia, asentar deudores. |
| 2 | El usuario procederá a ingresar los parámetros de búsqueda. |
| 3 | El aplicativo mostrará un listado de resultados al usuario. |
| 4 | El aplicativo habilitará las opciones **imprimir/exportar** para la lista de resultados. |
| **Post-Condición** | Ninguno | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 4, en un mínimo de 5 minutos el cual variará de acuerdo al ancho de banda con que cuente el usuario y la cantidad de registros del reporte. | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida de 100 veces al mes. | |
| **Importancia** | Vital | |
| **Requerimiento**  **Asociado** | RF11 | |

Tabla 45 Descripción de Casos de Uso: Consultar Alumno (FUENTE PROPIA)

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | **CONSULTAR ALUMNO** | | |
| **Descripción** | El aplicativo deberá permitir al usuario consultar alumnos para poder editarlos como se describe a continuación: | | |
| **Precondición** | Ninguno | | |
| **Actores** | Secretaria, Director | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | El usuario seleccionara la opción **administrar / alumno** y elegirá la opción de editar | |
| 2 | El aplicativo solicitara al usuario el ingreso del número de DNI del alumno a consultar. | |
| 3 | El usuario ingresara los datos solicitados y elegirá la opción **buscar**. | |
| 4 | El aplicativo mostrara un panel con los datos del alumno, donde el usuario podrá **modificar** o **eliminar** la información. | |
|  | 1a | Si el usuario desea modificar solo deberá editar los datos mostrados (alumno, padre, responsable de recojo, día de asistencia. |
|  | 2a | El usuario seleccionara la opción guardar para almacenar la información modificada. |
|  | 1b | Si el usuario desea eliminar solo deberá escoger la opción **eliminar**. |
|  | 2b | El aplicativo muestra mensaje de advertencia. |
|  | 3c | El usuario confirma la eliminación. |
|  | 4d | El aplicativo guarda los cambios. |
| **Post-Condición** | Ninguno | | |
| **Rendimiento** | El aplicativo deberá realizar las acciones descritas en los pasos 1 al 4, 1 al 2a, 1 al 4d, en un mínimo de 5 minutos el cual variará de acuerdo al ancho de banda con que cuente el usuario y la cantidad de registros del reporte. | | |
| **Frecuencia** | Este caso de uso se espera que se lleve a cabo en una medida mínima de 100 veces al mes. | | |
| **Importancia** | Vital | | |
| **Req. Asociado** | RF08 | | |

### Diagramas de Clases

* + - 1. **Modulo Alumno**

****Ilustración 6 Diagrama de Clases: Modulo Alumno (FUENTE PROPIA)

63

* + - 1. **Modulo Usuarios**

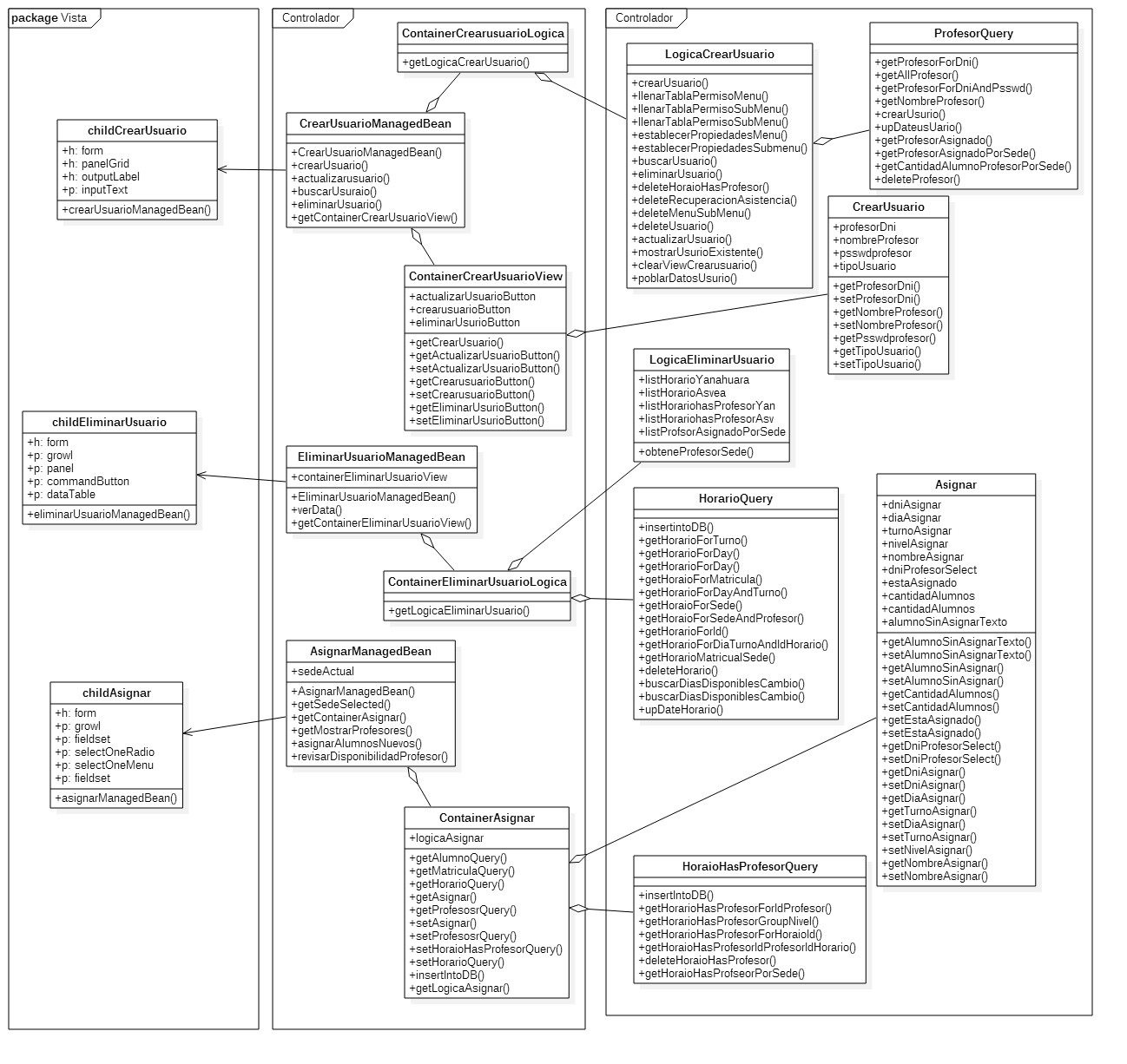


Ilustración 7 Diagrama de Clases: Modulo Usuarios (FUENTE PROPIA)

* + - 1. **Modulo Montos**

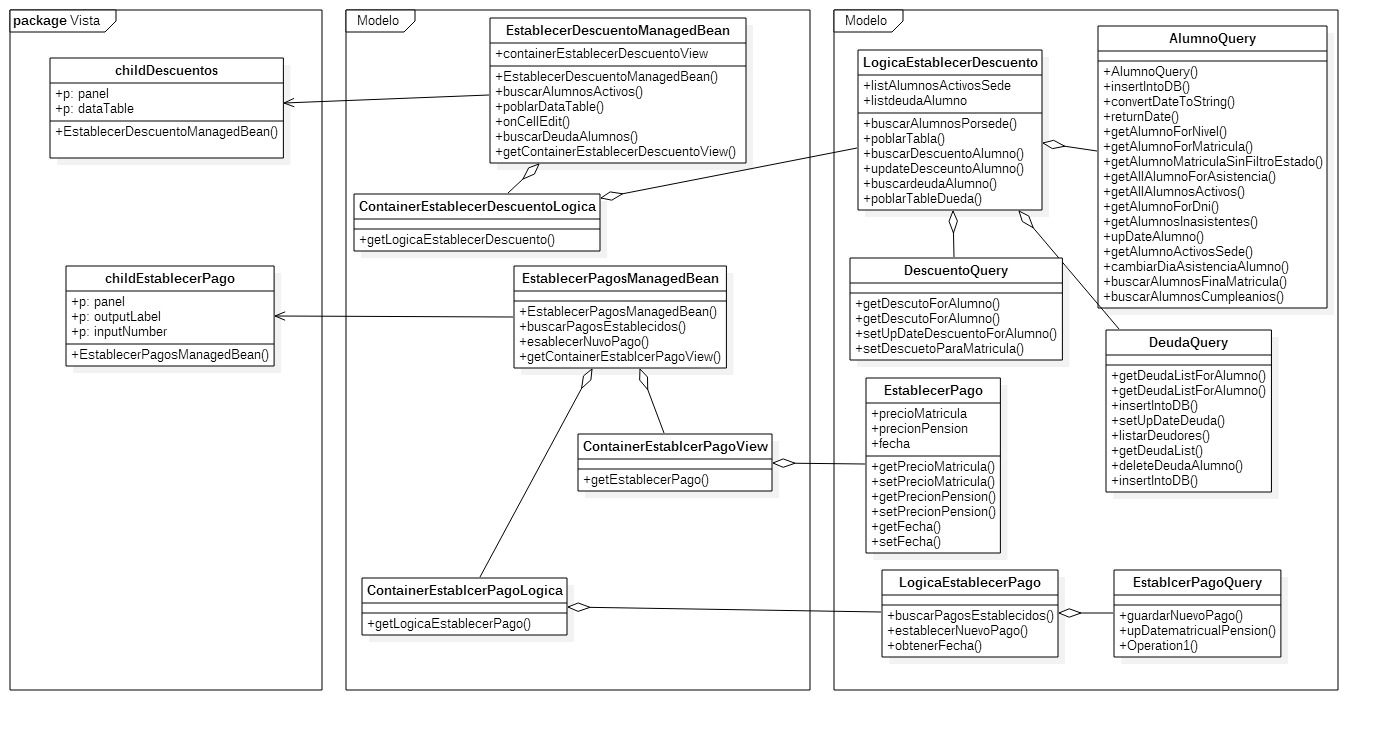
****

Ilustración 8 Diagrama de Clases: Modulo Montos (FUENTE PROPIA)

65

* + - 1. **Modulo Matricula**

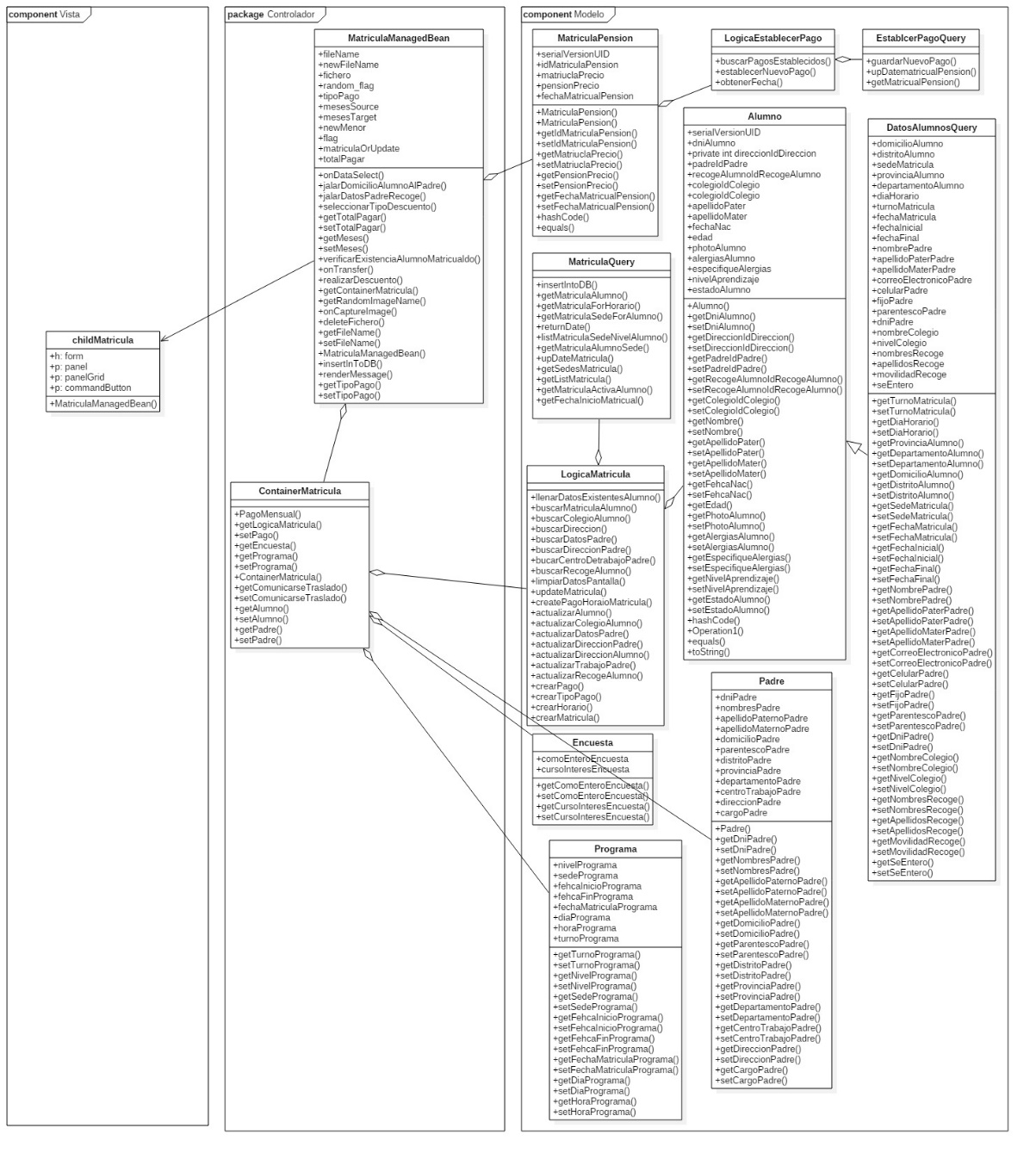
****

Ilustración 9 Diagrama de Clases: Modulo Matricula (FUENTE PROPIA)

* + - 1. **Modulo Asistencia**

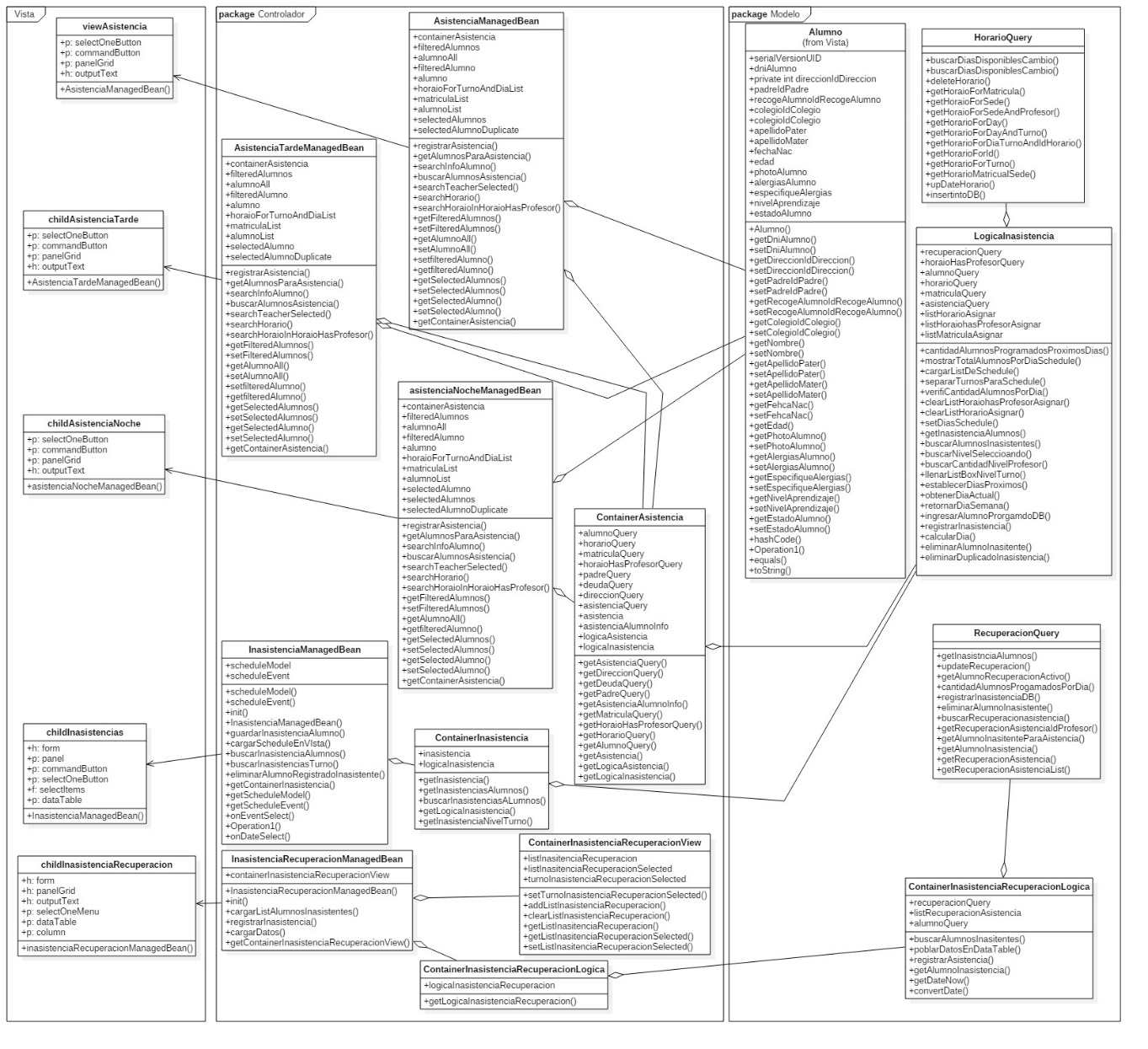
****

Ilustración 10 Diagrama de Clases: Modulo Asistencia (FUENTE PROPIA)

* + - 1. **Modulo Actividad**

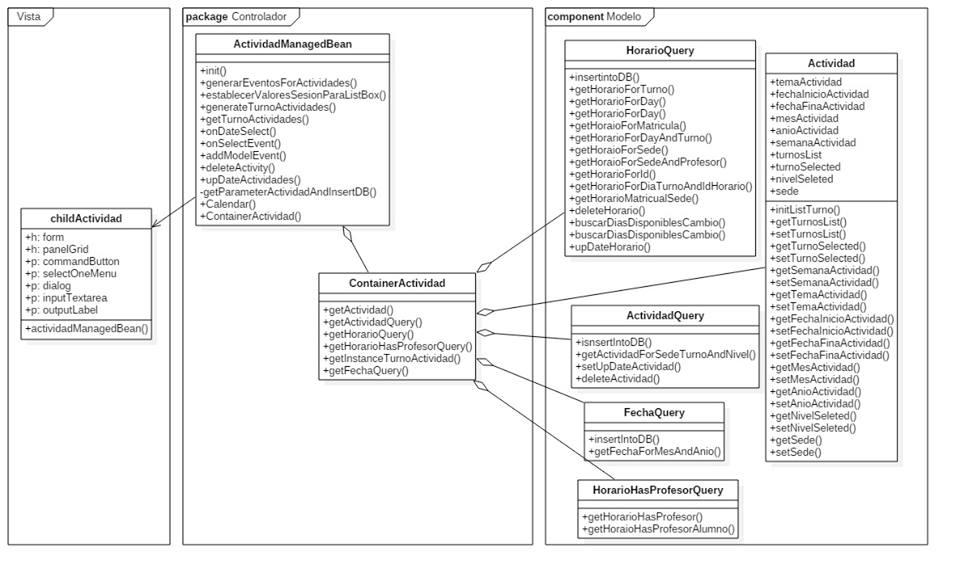


Ilustración 11 Diagrama de Clases: Modulo Actividad (FUENTE PROPIA)

* + - 1. **Modulo Pagos**

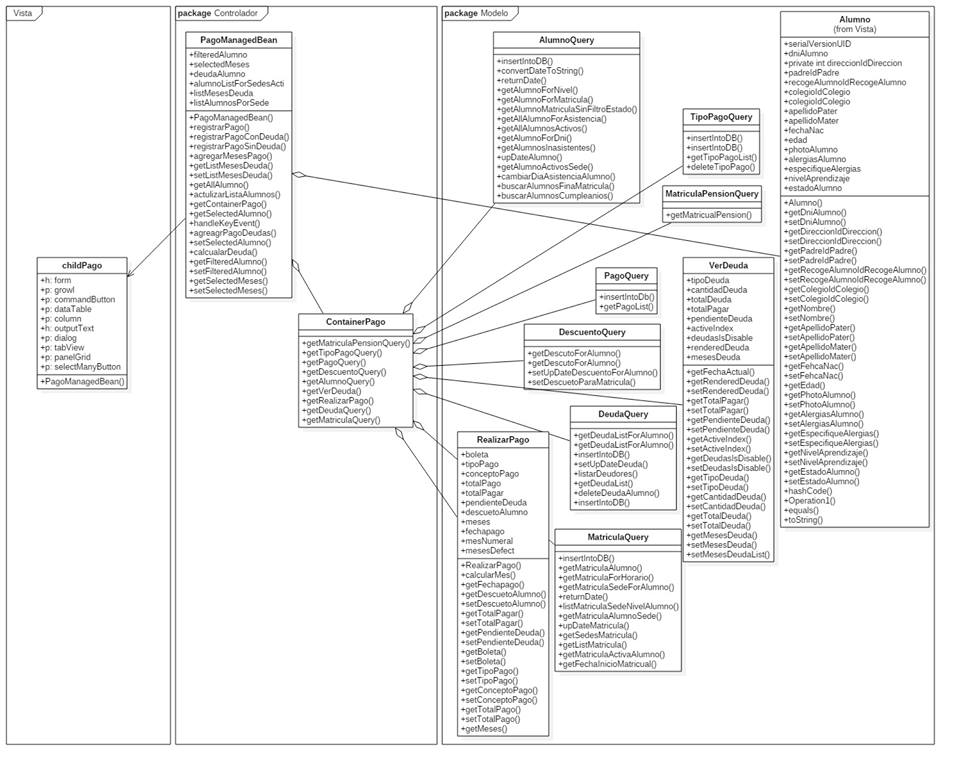
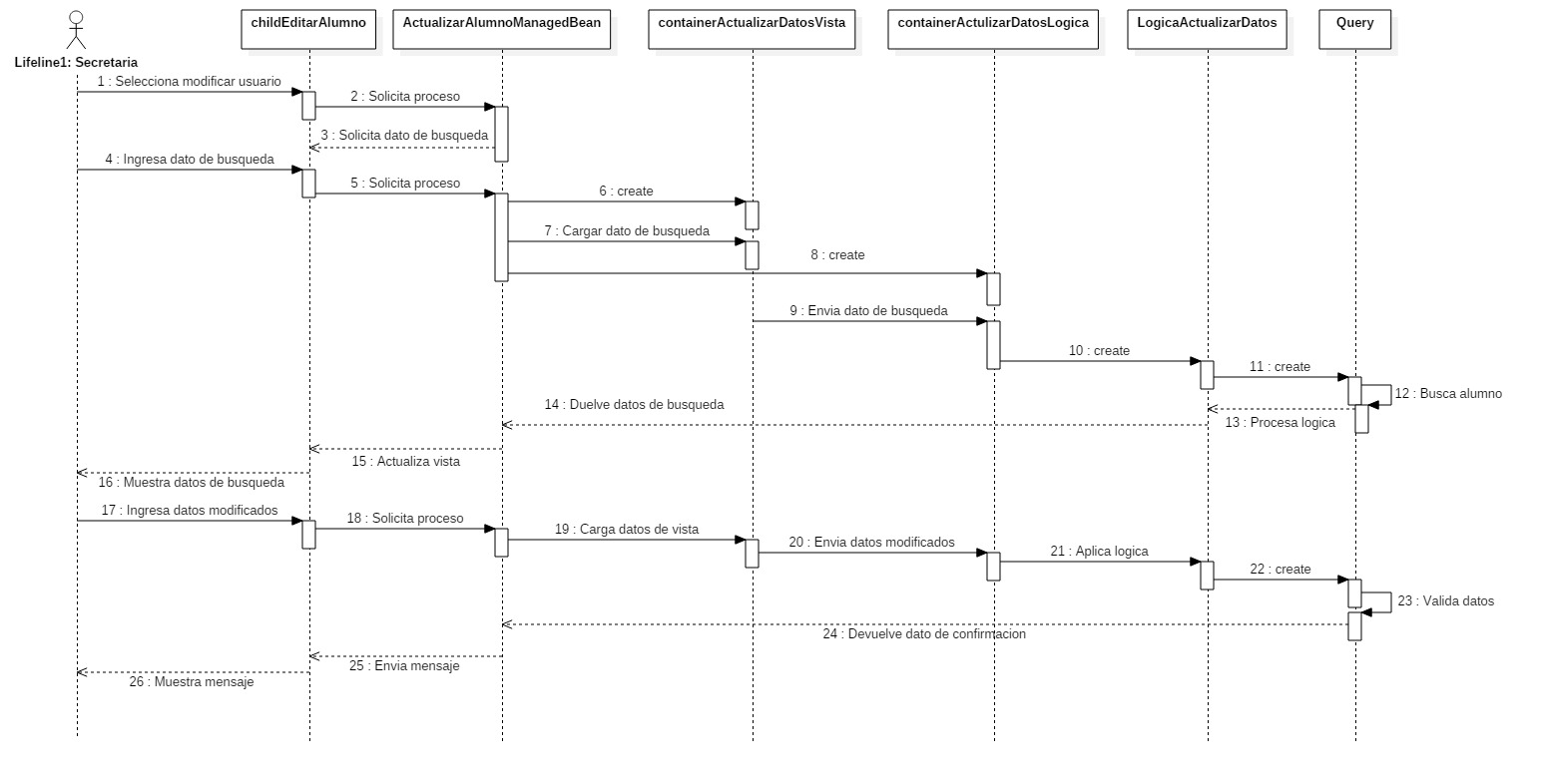


Ilustración 12 Diagrama de Clases: Modulo Pagos (FUENTE PROPIA)

### Diagrama de Secuencia

* + - 1. **Diagrama de Secuencia: Gestión de Alumnos**

Alumno: modificar / buscar

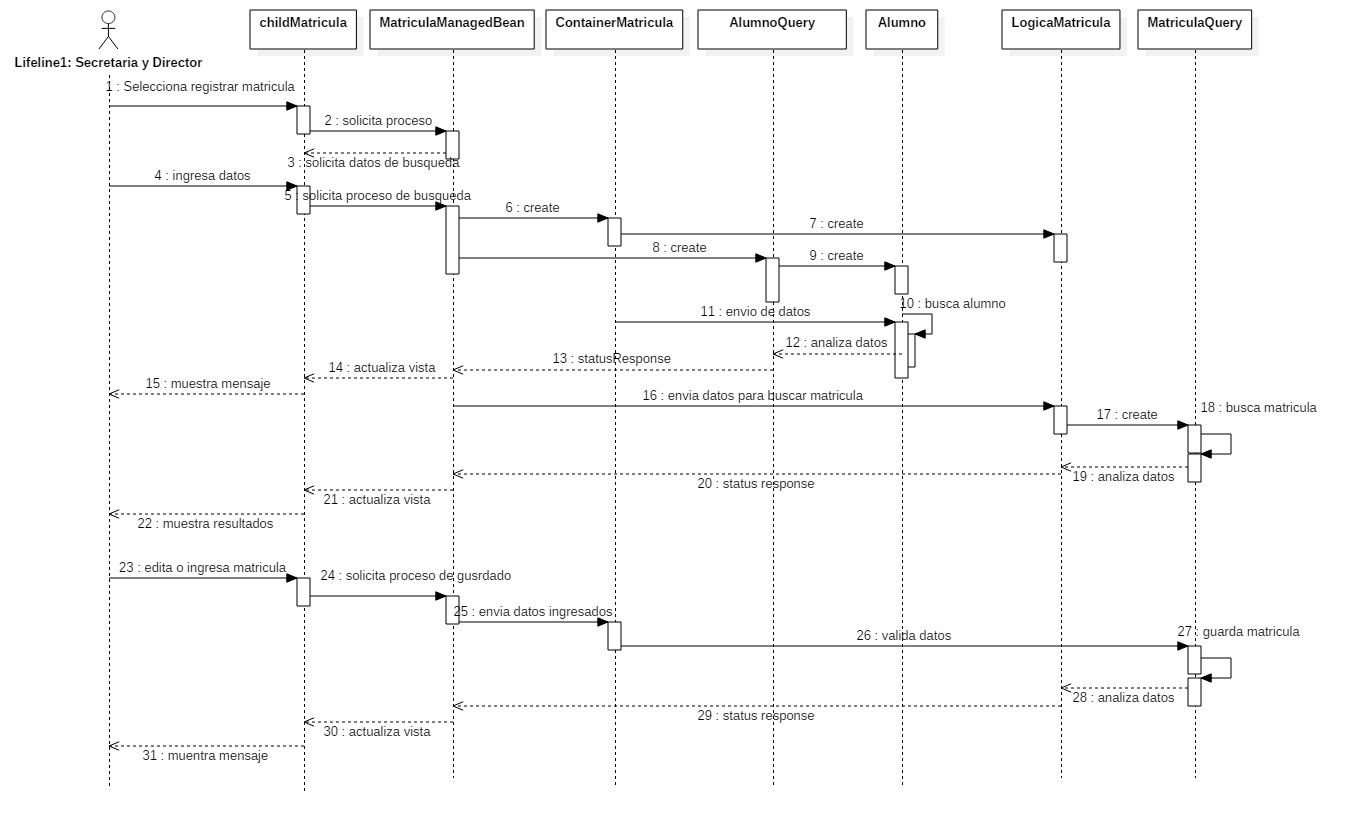
**

69

Ilustración 13 Diagrama de Secuencia: Gestión de Alumnos (FUENTE PROPIA)

* + - 1. **Diagrama de Secuencia: Gestión de matriculas**

Matriculas: Buscar / Modificar / Crear



70

Ilustración 14 Diagrama de Secuencia: Gestión de Matriculas (FUENTE PROPIA)

* + - 1. **Diagrama de Secuencia: Gestión de matriculas**

Matriculas: Buscar / Reporte

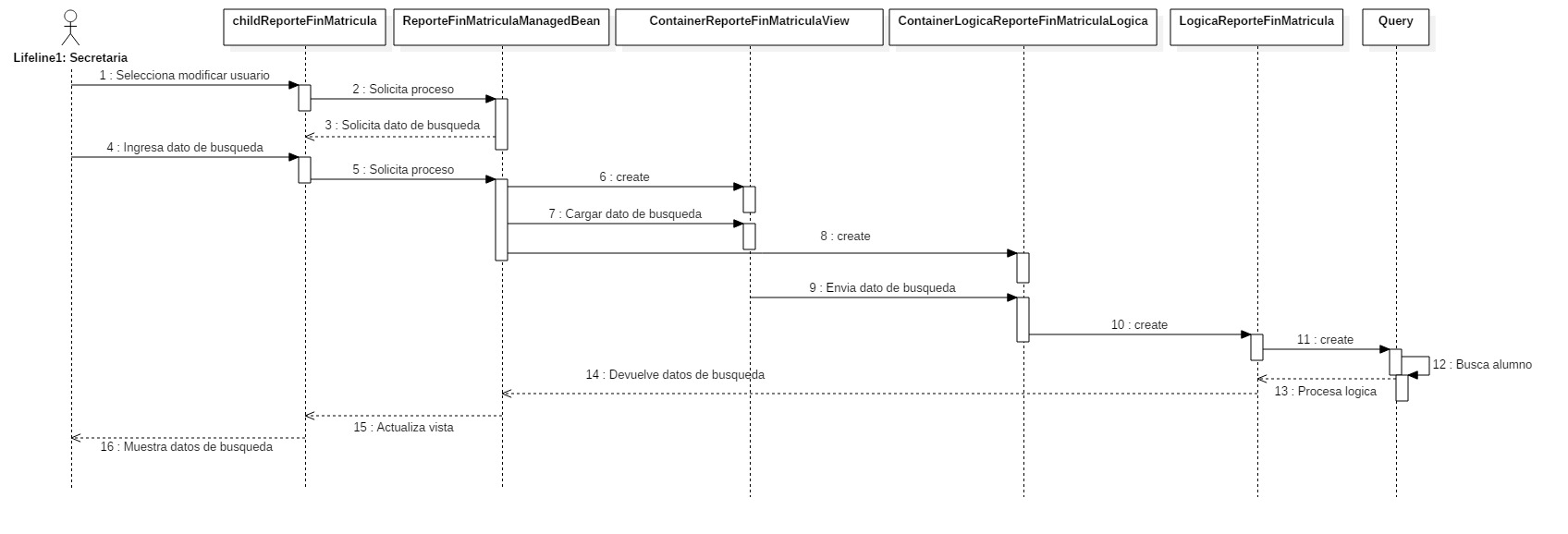


Ilustración 15 Diagrama de Secuencia: Gestión Matriculas (FUENTE PROPIA)

71

* + - 1. **Diagrama de Secuencia: Gestión de usuarios**

Usuario: crear / buscar

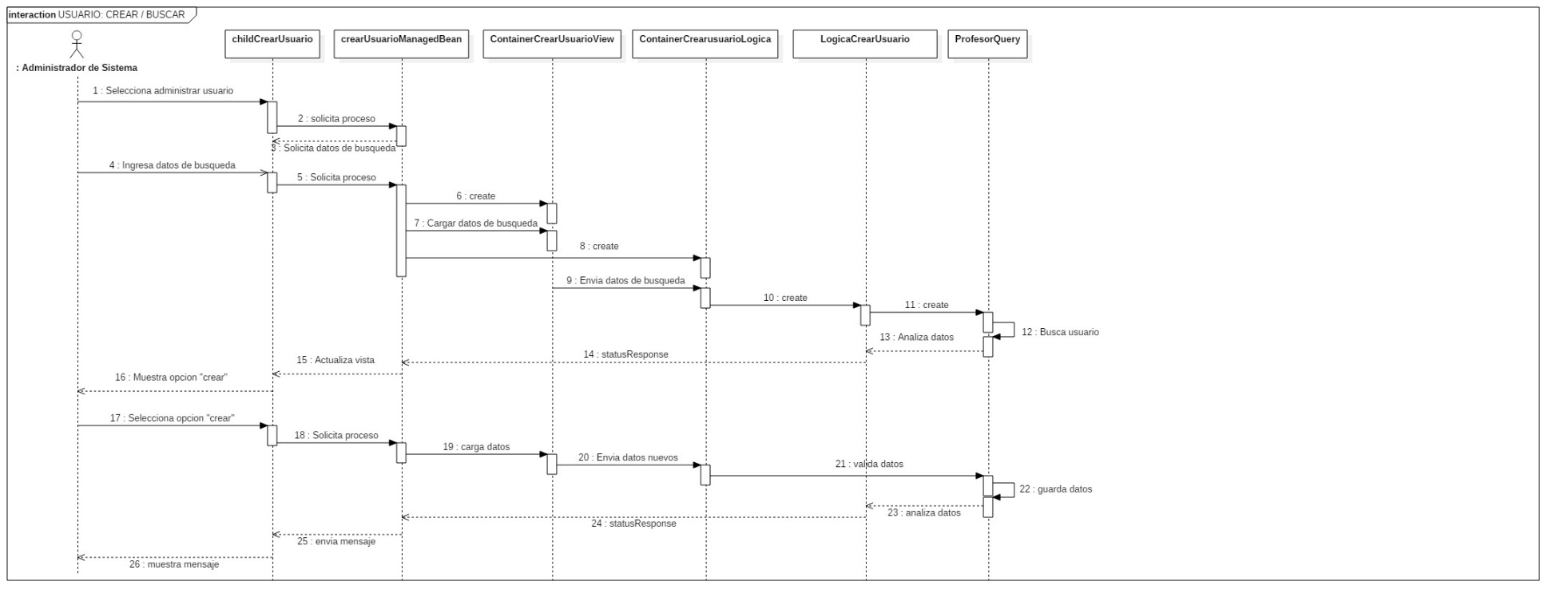
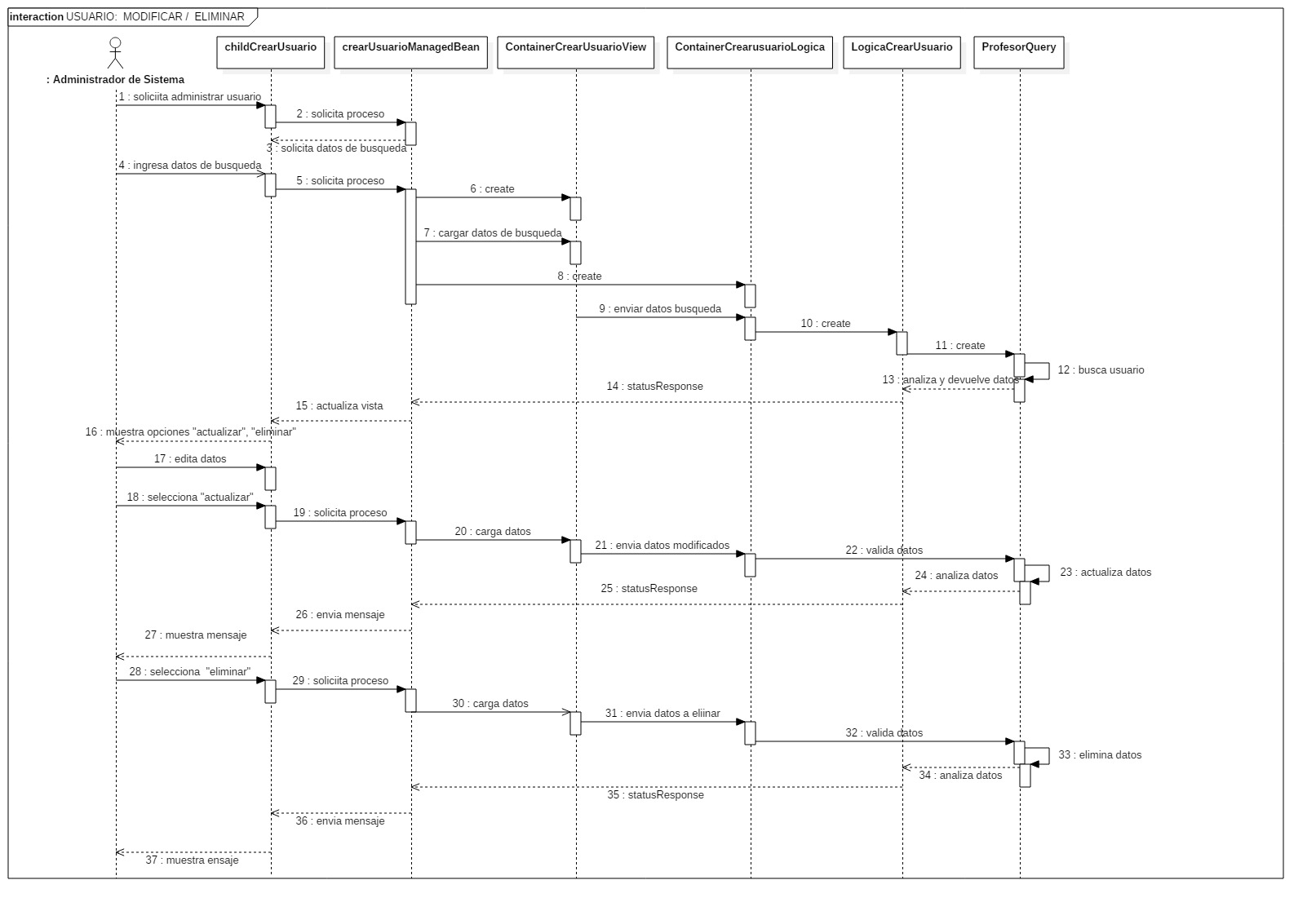


Ilustración 16 Diagrama de Secuencia gestión de usuarios (FUENTE PROPIA)

72

Usuario: modificar / eliminar

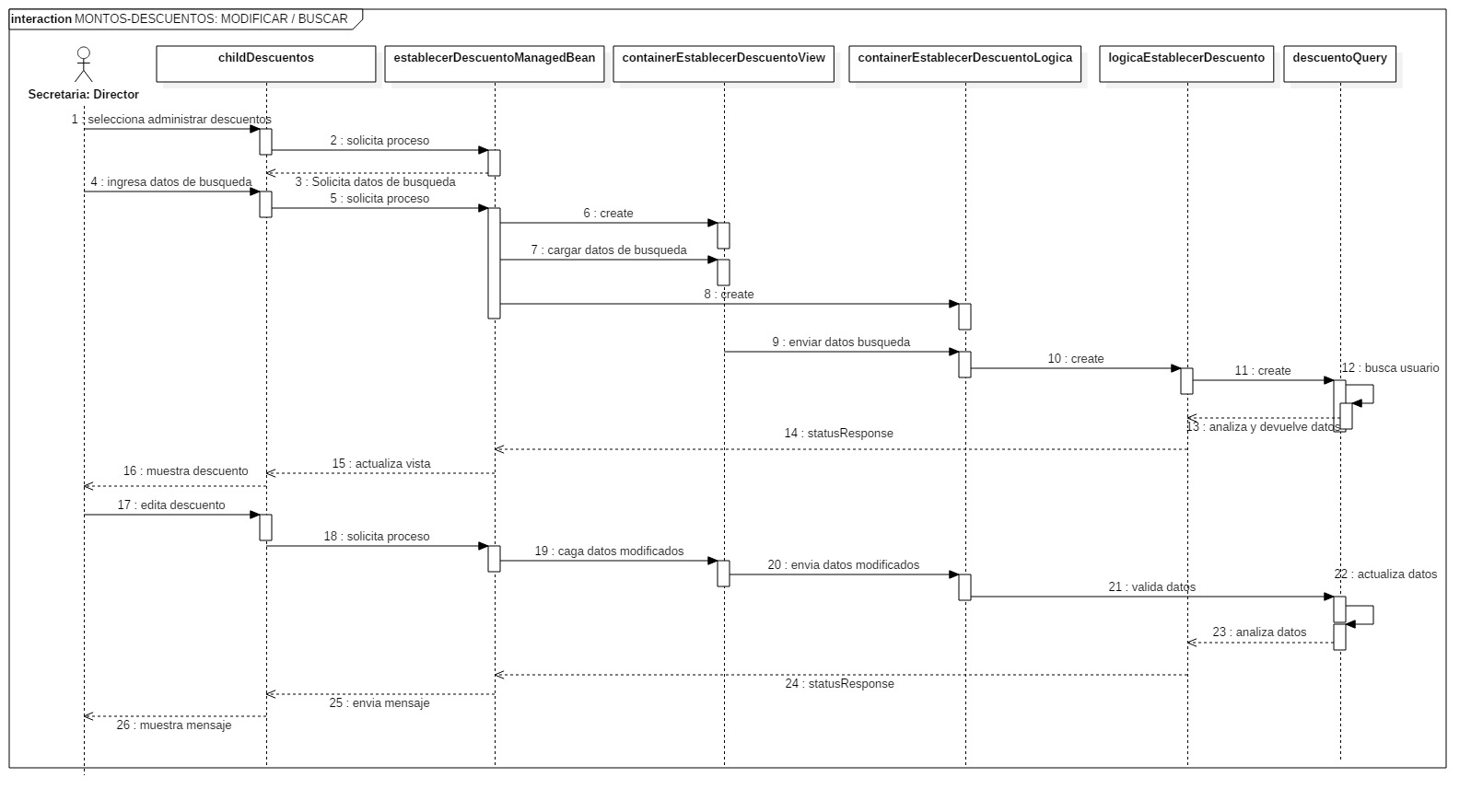


73

Ilustración 17 Diagrama de Secuencia gestión de usuarios (FUENTE PROPIA)

* + - 1. **Diagrama de Secuencia: Gestión de montos**

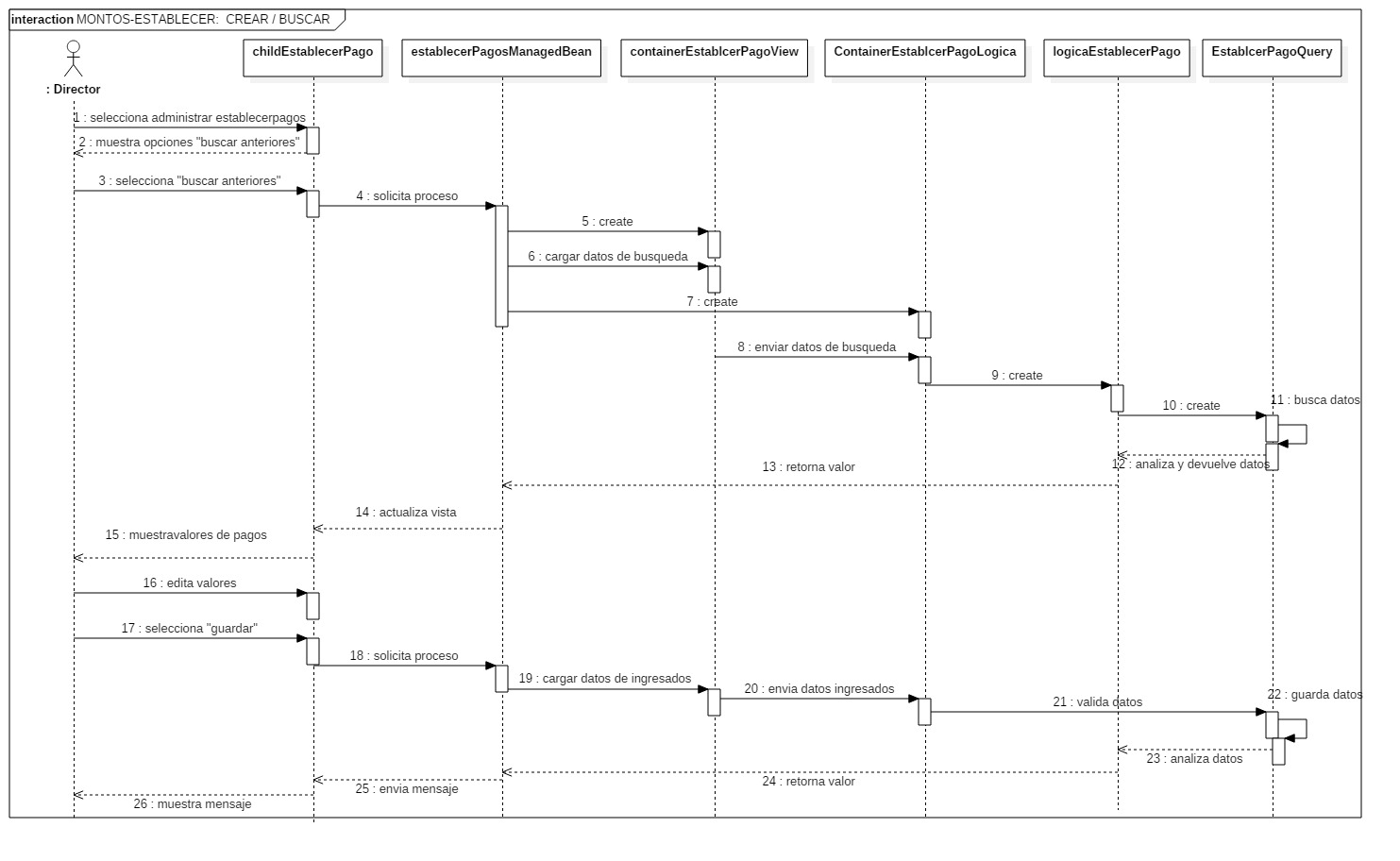
Montos - descuentos: modificar / buscar



74

Ilustración 18 Diagrama de Secuencia gestión de montos-descuentos (FUENTE PROPIA)

Montos - establecer: crear / buscar

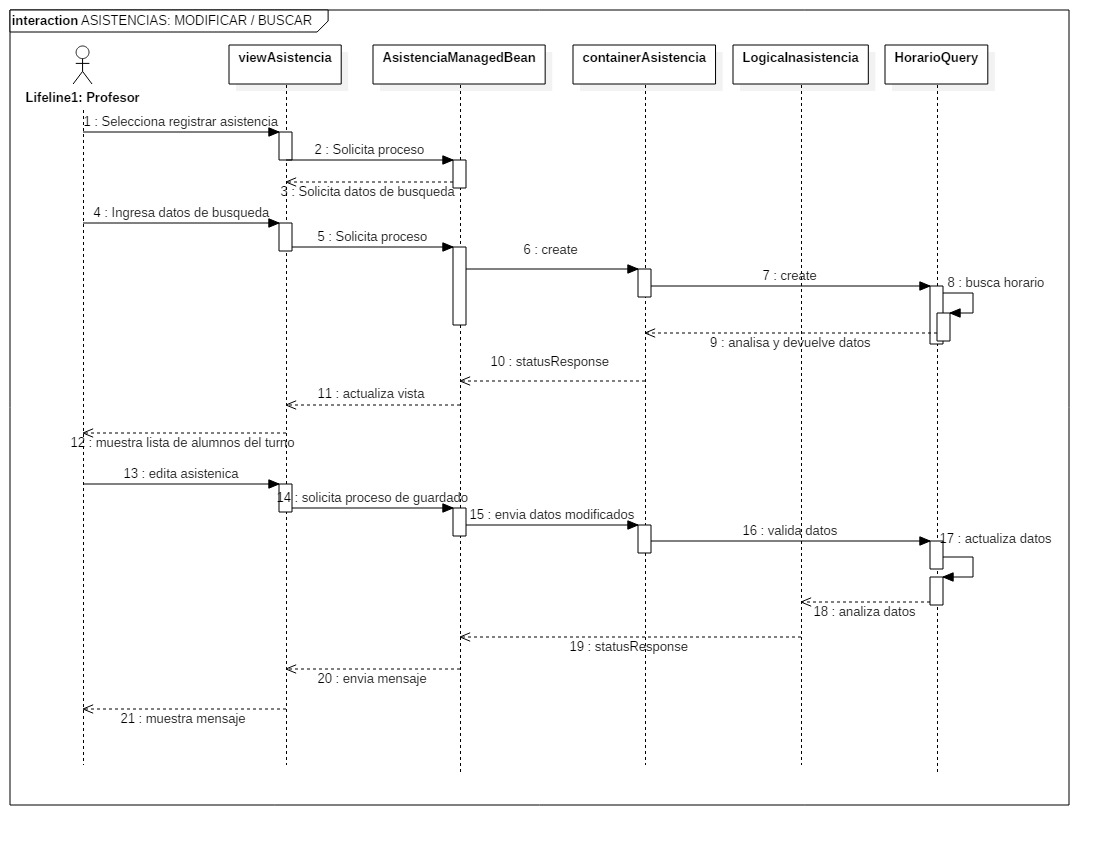


75

Ilustración 19 Diagrama de Secuencia gestión de montos-establecer pagos (FUENTE PROPIA )

* + - 1. **Diagrama de Secuencia: Gestión de asistencias**

Asistencias (Mañana1, mañana2, tarde): buscar / modificar

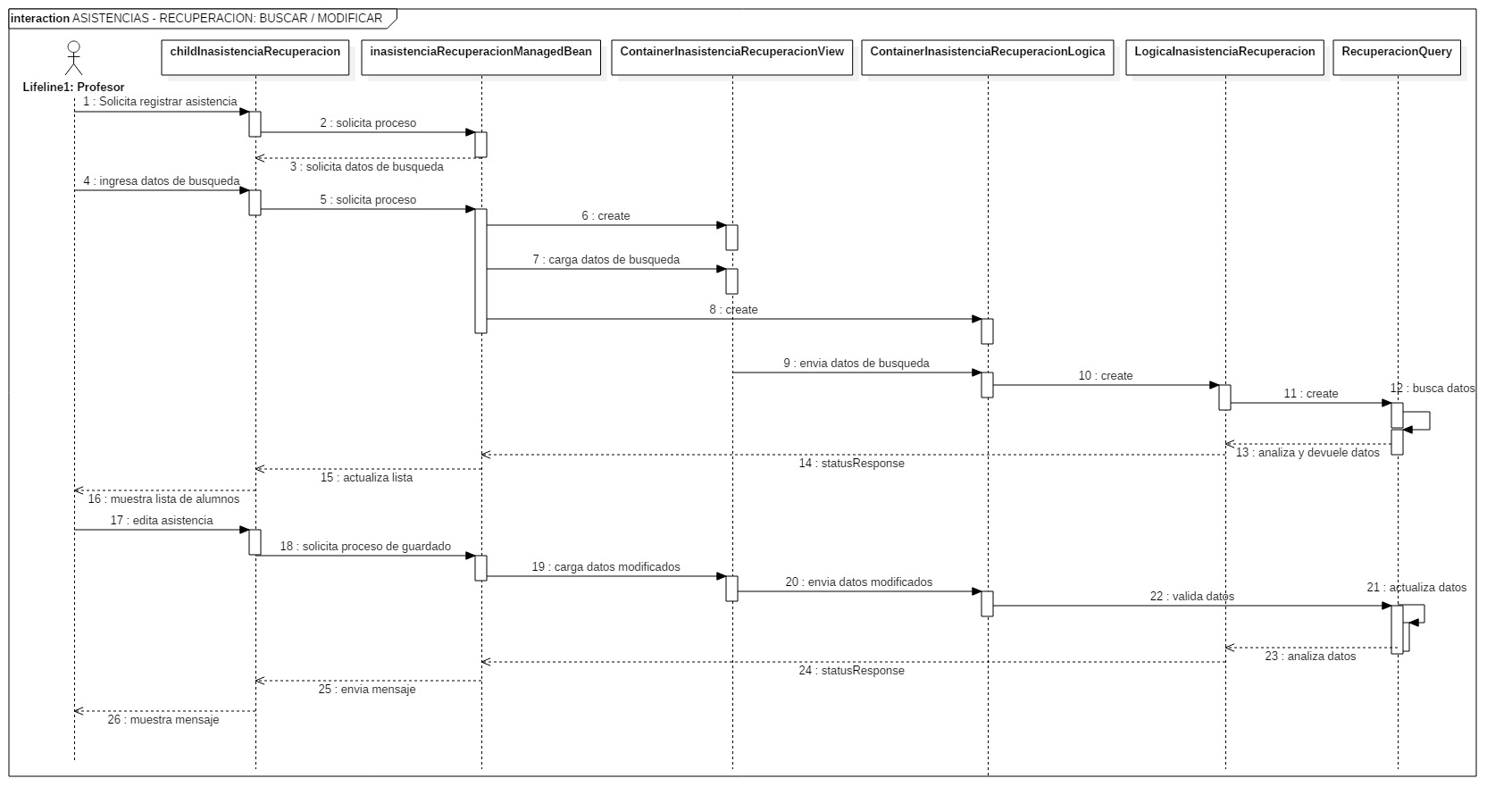


76

Ilustración 20 Diagrama de Secuencia gestión de asistencias (FUENTE PROPIA)

* + - 1. **Diagrama de Secuencia: Gestión de asistencias**

Asistencias (Recuperación): buscar / modificar



77

Ilustración 21 Diagrama de Secuencia gestión de asistencias (FUENTE PROPIA)

* + - 1. **Diagrama de Secuencia: Gestión de actividades**

Actividades: buscar / modificar

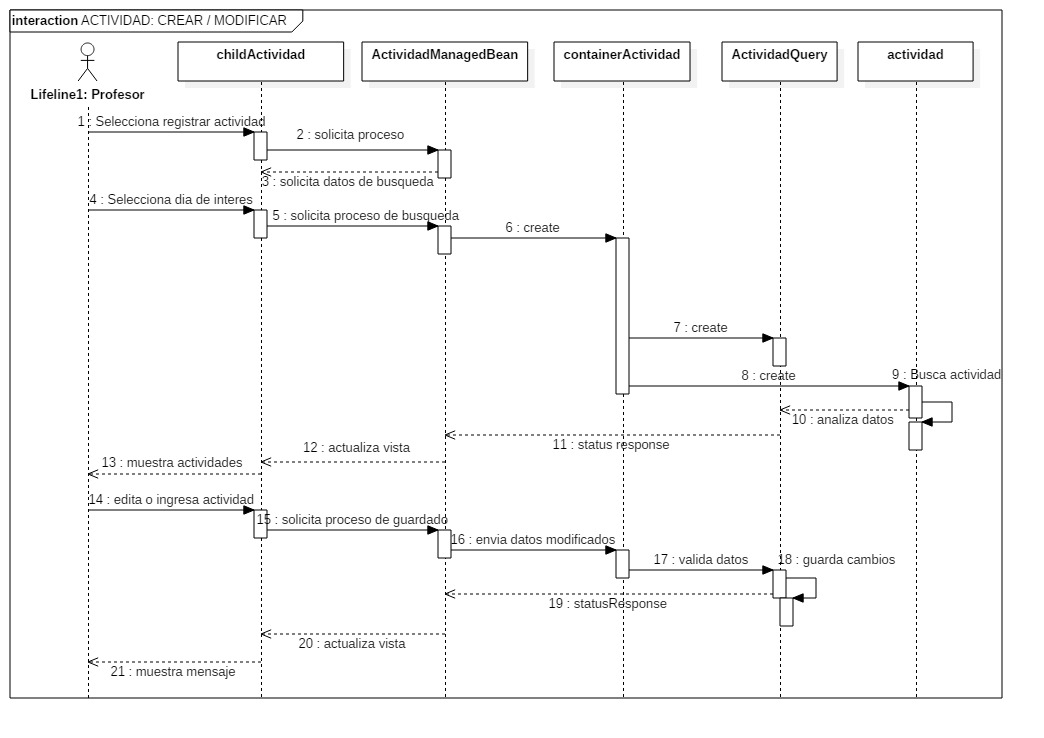
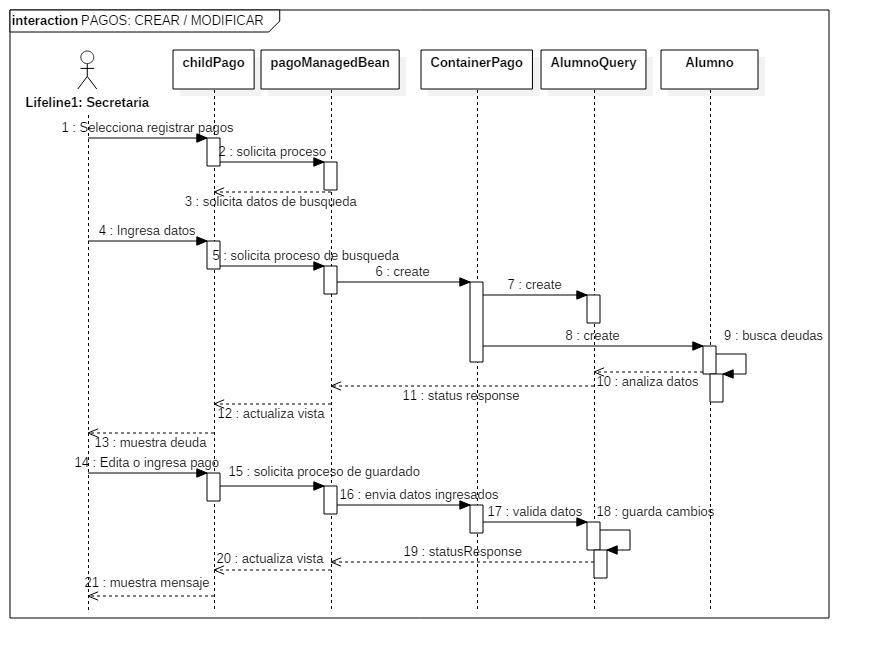


Ilustración 22 Diagrama de Secuencia gestión de actividades (FUENTE PROPIA)

78

* + - 1. **Diagrama de Secuencia: Gestión de pagos**

Pagos: buscar / modificar



79

Ilustración 23 Diagrama de Secuencia gestión de pagos (FUENTE PROPIA)

### Diagrama de Componentes

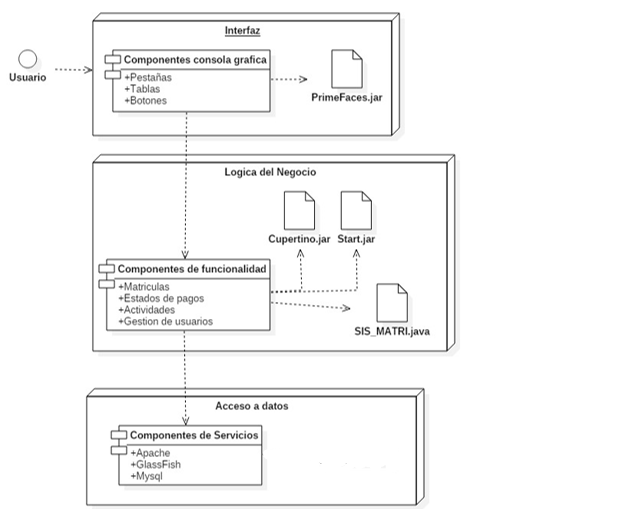


Ilustración 20 Diagrama de Componentes (FUENTE PROPIA)

* + - 1. **Consistencia en la cantidad de vistas de la arquitectura**

Tabla 46 Cuadro de Descripción de Módulos (FUENTE PROPIA)

|  |  |
| --- | --- |
| **NOMBRE DEL MODULO** | **DESCRIPCION** |
|  | Este módulo representa la INTERFAZ a la cual el usuario interactuara directamente gracias a nuestro framewok Prime Faces y así agrupar funcionalidades que se acercan más al negocio. |
| **NOMBRE DEL MODULO** | **DESCRIPCION** |
|  | Este módulo representa la LÓGICA DEL NEGOCIO pues se desarrollara las funcionalidades que presenta el sistema como Matricular, Estado de pagos, Actividades y gestor de usuarios. |
| **NOMBRE DEL MODULO** | **DESCRIPCION** |
|  | Este módulo ACCESO A DATOS contiene a las clases que hacen persistencia a la base de datos expresada en el modelo de datos. |

Tabla 47 Descripción de Componentes (FUENTE PROPIA)

|  |  |
| --- | --- |
| **NOMBRE DEL COMPONENTE** | **DESCRIPCION** |
|  | Contiene la lógica de consola gráfica de pestañas , tablas y los botones que permitirá la interacción con el usuario y su facilidad de uso. |
| **NOMBRE DEL COMPONENTE** | **DESCRIPCION** |
|  | Contiene la lógica de desarrollo de matrículas, estado de pagos, actividades y la gestión de usuarios. Que permitirá comprender la funcionalidad. |
| **NOMBRE DEL COMPONENTE** | **DESCRIPCION** |
|  | Contiene la lógica de servicios con el cual se trabaja los cuales son Apache, GlassFish y MySql. |

### Diseño de interfaces

* + - 1. **Interfaces de Usuario**

La interfaz gráfica con la que el usuario final hará interacción consistirá en un conjunto de ventanas con botones, opciones, listas y campos de texto. Esta deberá ser intuitiva de tal manera que el usuario pueda identificar rápidamente la funcionalidad del sistema, además de ser compatible con cualquier sistema operativo y navegadores comunes tales como Internet Explorer, Firefox y Chrome.

* + - 1. **Interfaces de Hardware**

Para una correcta ejecución del sistema se deberá disponer de equipos de cómputo en perfecto estado con las siguientes características mínimas:

* Tarjeta de red Ethernet o Inalámbrica dependiendo del tipo de conexión a internet.
* Memoria RAM mínima de 2GB.
* Procesador de 2.33 GHz en adelante.
* Mouse, teclado, monitor
  + - 1. **Interfaces de Software**

Navegador recomendado Firefox, pero es compatible con Chrome, Internet Explorer, Microsoft Edge y otros.

### MER

*Ilustración 21 Diagrama Entidad Relación (FUENTE PROPIA)*

84

# ASEGURAMIENTO DE LA CALIDAD

## Plan de Pruebas

Algunos documentos del proyecto AUTOMATIZACIÓN DE UN SISTEMA DE MATRICULAS DE GRUPO EDUCATIVA CON UNA APLICACIÓN WEB, son base fundamental de este documento inicial de plan de pruebas, que a continuación se relacionan:

* Metodología de Pruebas.
* Documento de Especificación de Requerimientos IEEE 830.
* Documento de Arquitectura IEEE 1471
* Términos de referencia del proceso de licitación de la plataforma de interoperabilidad.
  + 1. **Recursos del plan de pruebas**
       1. **Recurso del sistema**

Las pruebas se realizarán en un ambiente controlado y administrado por la UT; a continuación, se describen las características de la infraestructura del ambiente de pruebas.

Tabla 48 Recurso del Sistema de Pruebas (FUENTE PROPIA)

|  |  |  |
| --- | --- | --- |
| **DESCRIPCIÓN** | **FUNCIONALIDAD** | **CANTIDAD** |
| Servidor | Montar ambiente de Pruebas con la solución en proceso de desarrollo | 1 |
| * Estaciones de Trabajo | Con acceso al Servidor de Pruebas a través de internet. | 1 |
| * Herramientas de pruebas de sistemas | JUNIT | 1 |

* + - 1. **Configuración del ambiente de pruebas**

El ambiente de pruebas estará implementado en las instalaciones de la UT y su configuración deberá ser la siguiente.

Tabla 49 Configuración de Ambiente de Prueba (FUENTE PROPIA)

|  |  |  |  |
| --- | --- | --- | --- |
| **COMPONENTE** | **CONFIGURACIÓN** | **SOFTWARE INSTALADO Y CONFIGURADO** | **CANTIDAD** |
| Servidor | **Procesador:**1 CPU, 2.5 GHz, Intel Xeon Familia  **Memoria:**1 GB  **Disco Duro:** 30 GB  **Red:** AMAZON AWS | * Java NetBeans IDE 8.1 * PrimeFaces * MySql * PHPMyAdmin   UBUNTU SERVER 14.0 | 1 |

## Cheklist de pruebas

Plantilla de Caso de Pruebas, para más información consultar el ANEXO 3 DOCUMENTO DE PLAN DE PRUEBAS.

Tabla 50 Plantilla de Caso de Pruebas (FUENTE PROPIA)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **INFORMACIÓN GLOBAL DEL CASO DE PRUEBA** | | | | | | |
| **Tipo de Prueba:** | <Descripción del tipo de prueba: Carga, Volumen, Estrés, ETC> | **Código de la prueba** | | <Codificación de la prueba> | | | |
| **Descripción de la prueba:** | <Descripción del objetivo de la prueba> | | | | | |
| **Versión de Ejecución** | <Versión o iteración de ejecución de la prueba> | **Fecha de Ejecución** | | <Fecha de ejecución en formato AAAA/MM/DD diligenciado por el analista de pruebas al momento de su ejecución> | | |
| 1. **Prerrequisitos de la prueba** | | | | | | |
| <Lista de los prerrequisitos a tener en cuenta antes de ejecutar la prueba> | | | | | | |
| 1. **Insumos de la prueba** | | | | | | |
| <Lista de Insumos necesarios para ejecutar la prueba> | | | | | | |
| 1. **Lista de chequeo de la prueba** | | | | | | |
| **Pasos a Seguir** | | | **Prueba satisfactoria** | **Observaciones** | | |
| **SI** | | **NO** |  |
| <Pasos numerados en orden lógico para la ejecución de la prueba> | | |  | |  |  |
|  | | |  | |  |  |
|  | | |  | |  |  |
| 1. **Resultados de la prueba** | | | | | | |
| **Defectos y desviaciones** | | | | **Veredicto** | | |
| <Lista de defectos o desviaciones encontrados por el analista o usuario al ejecutar la prueba> | | | |  | | |
| **Observaciones** | | | | **Probador** | | |
| <Observaciones generales del analista o usuario sobre la ejecución de la prueba> | | | |  | | |
| **Firma:**  **Nombre:**  **Fecha:** | | |

# RESULTADOS

## Cuadro estadístico

89

Ilustración 22 Cuadro Estadístico (FUENTE PROPIA)

## Encuestas de satisfacción

Las pruebas de usabilidad se guiarán por la siguiente estructura de encuesta:

Tabla 51 Modelo de encuesta de Satisfacción

|  |  |
| --- | --- |
| **PREGUNTAS** | **ITEMS** |
| **1. ¿Hay términos en idiomas diferentes mezclados?** | 1 = Se encuentran en todo el sistema  2= Se encuentra en algunas partes del sistema.  3 = No se encuentran en ninguna parte del sistema. |
| **2. ¿Es simple el vocabulario utilizado?** | 1 = El vocabulario es demasiado técnico.  2=El vocabulario presenta algunas dificultades de comprensión.  3=El vocabulario es completamente comprensible. |
| **3. ¿Se proporciona tiempo suficiente para realizar las entradas por teclado?** | 1 = El tiempo es muy limitado.  2 = El tiempo es limitado para algunas funcionalidades.  3 = El tiempo es completamente suficiente. |
| **4. ¿Hay algún tipo de asistencia para los usuarios que hacen uso del sistema por primera vez?** | 1 = No existe ninguna ayuda.  2 = Se encuentra ayuda en algunas partes.  3 = Existen ayudas en todo el sistema. |
| **3. ¿El sistema es fácil de operar para alguien que no recibió capacitación en su operación?** | 1 = El sistema es de difícil comprensión.  2 = El sistema es fácil de operar en algunas de sus funcionalidades.  3 = El sistema es completamente fácil de operar. |
| **6. ¿Se entienden la interfaz y su contenido?** | 1 = No se entiende su interfaz.  2 = La interfaz se entiende en algunas partes.  3 = La interfaz es completamente entendible. |
| **7. ¿Resulta fácil identificar un objeto o una acción?** | 1 = Es difícil identificar los objetos o acciones.  2 = Se pueden identificar los objetos y acciones en algunas partes del sistema.  3 = Todos los objetos y acciones son fácilmente identificables. |
| **8. ¿Resulta fácil entender el resultado de una acción?** | 1= Los resultados de las acciones no son entendibles.  2= Los resultados de las acciones son entendibles en algunas partes o la mayor parte del sistema.  3 = Todos los resultados de las acciones son entendibles. |
| **9. ¿Está diseñada la interfaz para facilitar la realización eficiente de las tareas de la mejor forma posible?** | 1 = La interfaz es difícil de usar.  2 = La interfaz es difícil de usar en algunas partes del sistema.  3 = La interfaz es completamente sencilla de usar. |
| **10. ¿Son apropiados los mensajes presentado por el sistema?** | 1 = Los mensajes non son apropiados.  2 = Los mensajes son apropiados en algunas partes del sistema.  3 = Todos los mensajes son apropiados y fáciles de comprender. |
| **11. ¿Actúa el sistema en la prevención de errores?** | 1 = El sistema no previene errores del usuario.  2 = El sistema previene algunos o la mayoría de los errores del usuario.  3 = El sistema previene cualquier error que pueda cometer el usuario. |
| **12. ¿El sistema informa claramente sobre los errores presentados?** | 1 = El sistema no informa de manera adecuada sobre los errores cometidos.  2 = El sistema informa de manera adecuada algunos o la mayoría de los errores cometidos por el usuario.  3 = El sistema informa de forma adecuada todos los errores cometidos por el usuario. |
| **13. ¿Se utiliza mensajes y textos descriptivos?** | 1 = Los mensajes de texto no son descriptivos.  2 = La mayoría de los textos son descriptivos o fáciles de interpretar  3 = Todos los textos son descriptivos o fáciles de interpretar. |
| **14. ¿Permite una cómoda navegación dentro del producto y una fácil salida de éste?** | 1 = La navegación no es sencilla.  2 = La navegación presenta algunas dificultades.  3 = La navegación es sencilla, requiere de pocos vínculos para accedes a las funcionalidades del sistema. |
| **13. ¿Se permite al usuario personalizar la interfaz?** | 1 = La interfaz no es personalizable.  2 = La interfaz es personalizable con algunas restricciones.  3 = La interfaz es completamente personalizable. |
| **16. ¿Se proporciona información visual de dónde está el usuario, qué está haciendo y qué puede hacer a continuación?** | 1 = No se presenta ninguna información visual ni otro tipo de ayuda.  2 = Presenta ayudas en algunas partes del sistema.  3 = Las ayudas son apropiadas y están distribuidas a los largo del sistema. |
| **17. ¿Existe atajos del teclado bien hechos?** | 1 = No existen atajos por teclado.  2 = Existen algunos atajos por teclado.  3 = Todas las opciones presentan atajos por teclado. |
| **18. ¿Se presenta al usuario la información que sólo necesita?** | 1 = La información presentada es más de la que necesita y tiende a ser confusa.  2 = En algunas partes se presenta mayor información a la necesaria.  3 = La información es estrictamente la necesaria según el perfil. |

# CONCLUSIONES

* + Se migró la información de la Institución de forma satisfactoria la cual se encuentra almacenada en una base de datos MySQL así mismo se hace uso de ella en la aplicación web implementada.
  + Se implementó el módulo de matrículas planteado el cual además permite emitir un reporte ficha de inscripción digital el cual remplaza a la ficha física de inscripción con la que contaba la institución educativa.
  + Se implementó el módulo de registro de pagos el cual permite gestionar descuentos y deudas respecto a las matrículas y pensiones.
  + Se Implementó el Modulo de actividades el cual permite gestionar las asistencias de los alumnos y las recuperaciones de las mismas.
  + Asimismo, el aplicativo permite registrar temas académicos de las clases a dictar que se programan para cada nivel.
  + Se implementó un control de accesos en forma de roles para cada trabajador que acceda a la aplicación web.
  + El aplicativo permite generar reportes dinámicos con filtros de búsqueda información útil y relevante para los usuarios que pueda contar con la funcionalidad de exportarlos en archivos tanto en formato Excel como en formato PDF.
  + El aplicativo cuenta con una interfaz gráfica agradable la cual permite al usuario interactuar con el aplicativo de manera intuitiva.
  + Gracias a los esfuerzos y tiempo invertidos en el desarrollo de la solución se logró implementar los requerimientos que el usuario nos hizo saber, no obstante, la solución esta para ser complementada con nuevos módulos y/o requerimientos a petición de los usuarios, por ejemplo, se nos solicitó implementar un módulo de inscripciones online.
  + Económicamente el proyecto fue viable ya que las herramientas empleadas para el desarrollo fueron de software libre en su totalidad sin embargo para el despliegue de la aplicación web en producción es necesario adquirir un hosting.
  + Finalmente mencionar que este proyecto se encuentra en constante optimización, mejora y desarrollo de nuevos requerimientos puesto que el aplicativo se encuentra en producción real y los usuarios demandan más funcionalidades, asimismo se está ofreciendo el soporte correspondiente a la institución.

# Referencias Bibliográficas

* Fernando Pech-May-Mario A. Gomez-Rodriguez- Luis A. de la Cruz-Diaz . ((2014)). *Desarrollo de Aplicaciones web con JPA, EJB, JSF y PrimeFaces.* Instituto Tecnológico Superior de los Ríos.
* Çivici, Ç. (2014). *PRIMEFACES - USER GUIDE 5.0.* EE.UU.: First Edition.
* Glifillan, I. ( (2003)). *La Biblia de MySQL.* ANAYA MULTIMEDIA.
* *https://www.java.com/es/download/faq/whatis\_java.xml*. (s.f.). Obtenido de https://www.java.com/es/download/faq/whatis\_java.xml.
* https://www.mysql.com/why-mysql/white-papers/10-razones-para-elegir-mysql-para-las-aplicaciones-web-de-la-proxima-generacion/. (s.f.). *https://www.mysql.com/why-mysql/white-papers/10-razones-para-elegir-mysql-para-las-aplicaciones-web-de-la-proxima-generacion/*. Obtenido de https://www.mysql.com/why-mysql/white-papers/10-razones-para-elegir-mysql-para-las-aplicaciones-web-de-la-proxima-generacion/.
* *https://www.oracle.com/es/mysql/index.html*. (s.f.). Obtenido de https://www.oracle.com/es/mysql/index.html.
* JAVA TM. (8 de Agosto de 2014). *www.java.com*. Obtenido de https://www.java.com/es/download/faq/whatis\_java.xml
* John S. Dean, Raymond H. Dean. (2009). *Intoduccion a la Programacion con Java.* Mexico: McGRAW-HILL/INTERAMERICANA EDITORES, S.A. DE C.V.
* JOSÉ MANUEL RODRÍGUEZ RODRÍGUEZ, MARÍA JOSÉ DAUREO CAMPILLO. (2003). *SISTEMAS DE INFORMACION: ASPECTOS TÉCNICOS Y LEGALES.* ALMERIA: ALMERIA.
* López, M. G. ((2009)). *Diccionario/ Directorio y Seguridad de datos.* Centro de investigación en Sistemas de Información CISI.
* Macario Polo,Daniel Villafranca . (2008). *Introducción a las aplicaciones Web con JAVA.* Expaña: ESI-UCLM.
* Microsoft. (20 de 3 de 2017). *msdn.microsoft.com*. Obtenido de msdn.microsoft.com: https://msdn.microsoft.com/es-es/library/bb972232.aspx
* Oracle Corporation. (3 de Marzo de 2014). *www.oracle.com*. Obtenido de www.oracle.com: https://www.oracle.com/es/mysql/index.html
* Oracle Corporation. (3 de Marzo de 2014). *www.oracle.com*. Obtenido de www.oracle.com: https://www.oracle.com/es/mysql/index.html
* Pérez, G. G. (11 de Febrero de 2008). *Aprendiendo Java y Programación Orientada a Objetos.* Obtenido de www.compunauta.com: http://compunauta.com/forums/linux/programacion/java/ebook.html
* Watson, P. H. (1998). *La guía del desarrollador, con una aplicación java basada en web .* Morgan Kauffman Publishers, Inc.

# ANEXOS

## [MANUAL DEL SISTEMA](K:\\CD PET\\ANEXOS\\ANEXO 1 DOCUMENTO DE MANUAL DEL SISTEMA.docx)

## [MANUAL DE USUARIO](K:\\CD PET\\ANEXOS\\ANEXO 2 MANUAL DE USUARIO.docx)

* 1. [**MANUAL DE USUARIO U\_ADMINISTRADORDAD**](file:///K:\CD%20PET\ANEXOS\ANEXO%202.1%20MANUAL%20DE%20USUARIO%20-%20U_ADMINISTRADOR.pdf)
  2. [**MANUAL DE USUARIO - U\_DIRECTOSECRETARIA**](file:///K:\CD%20PET\ANEXOS\ANEXO%202.2%20MANUAL%20DE%20USUARIO%20-%20U_DIRECTOR-SECRETARIA.pdf)
  3. [**MANUAL DE USUARIO - U\_PROFESOR**](file:///K:\CD%20PET\ANEXOS\ANEXO%202.3%20MANUAL%20DE%20USUARIO%20-%20U_PROFESOR.pdf)

## [DOCUMENTO DE PRUEBAS](K:\\CD PET\\ANEXOS\\ANEXO 3 DOCUMENTO DE PLAN DE PRUEBAS.docx)

## [GUÍA DE INSTALACIÓN](K:\\CD PET\\ANEXOS\\ANEXO 4 DOCUMENTO GUIA DE INSTALACIÓN.docx)

## [DICCIONARIO DE DATOS](K:\\CD PET\\ANEXOS\\ANEXO 5 DICCIONARIO DE DATOS.docx)

## [ENCUESTAS REALIZADAS](K:\\CD PET\\ANEXOS\\ANEXO 6 ENCUESTAS REALIZADAS.docx)

## [DOCUMENTO DE ARQUITECTURA DE SOFTWARE](K:\\CD PET\\ANEXOS\\ANEXO 7 DOCUMENTO DE ARQUITECTURA DEL SISTEMA.docx)

## [DOCUMENTO DE ESPECIFICACIÓN DE REQUISITOS](K:\\CD PET\\ANEXOS\\ANEXO 8 Especificaciónes de Requisitos según el estándar de IEEE 830 25-03-2017.docx)

1. https://www.java.com/es/download/faq/whatis\_java.xml [↑](#footnote-ref-1)
2. Ian Glifillan (2003). La Biblia de MySQL, ANAYA MULTIMEDIA [↑](#footnote-ref-2)
3. https://www.oracle.com/es/mysql/index.html [↑](#footnote-ref-3)
4. La guía del desarrollador, con una aplicación java basada en web, por Paul Harmon y Mark Watson; Morgan Kauffman Publishers, Inc., 1998 [↑](#footnote-ref-4)
5. Diccionario/ Directorio y Seguridad de datos, María Gertrudis López (2009), Centro de investigación en Sistemas de Información CISI. [↑](#footnote-ref-5)
6. https://msdn.microsoft.com/es-es/library/bb972232.aspx [↑](#footnote-ref-6)
7. Desarrollo de Aplicaciones web con JPA, EJB, JSF y PrimeFaces, Fernando Pech-May-Mario A. Gomez-Rodriguez- Luis A. de la Cruz-Díaz (2014) Instituto Tecnológico Superior de los Ríos [↑](#footnote-ref-7)