

**JAVA-UNIMINUTO-OVA  
SOFTWARE OBJETO VIRTUAL DE APRENDIZAJE – APRENDIZAJE DE  
PROGRAMACIÓN ORIENTADA A OBJETOS**

**INTEGRANTES.  
MIGUEL ÁNGEL ROJAS OCAMPO.  
YEISSON ALBERTO OSORIO GUARÍN.**

**CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS.  
FACULTAD DE INGENIERÍA.  
TECNOLOGÍA EN INFORMÁTICA.  
BOGOTÁ, COLOMBIA.  
2012.**

**JAVA-UNIMINUTO-OVA  
SOFTWARE OBJETO VIRTUAL DE APRENDIZAJE – APRENDIZAJE DE  
PROGRAMACIÓN ORIENTADA A OBJETOS**

**INTEGRANTES  
MIGUEL ÁNGEL ROJAS OCAMPO.  
YEISSON ALBERTO OSORIO GUARÍN.**

**PROYECTO DE GRADO.  
TECNOLOGÍA EN INFORMÁTICA.**

**DIRECTOR: ING. CARLOS CHARRY MORA.  
ASESOR: ING. ANA MARÍA OBANDO NATES.**

**CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS.  
FACULTAD DE INGENIERÍA.  
TECNOLOGÍA EN INFORMÁTICA.  
BOGOTÁ, COLOMBIA.  
2012**

Nota de aceptación:

---

---

---

---

---

---

---

---

---

---

---

Firma del presidente del jurado

---

Firma del jurado

---

Firma del jurado

Bogotá 5 de julio 2012.

## **DEDICATORIA**

Dedicado a quienes nos apoyaron a seguir esta  
Iniciativa de estudiar con ganas, esfuerzo, sacrificio y amor.

Para nuestros padres y Docentes académicos de la facultad.

Nuestros mayores agradecimientos y esta dedicatoria es para ellos.

## **AGRADECIMIENTOS**

A nuestras familias que nos han dado todo  
El apoyo y fuerza para llegar cumplir  
Nuestros objetivos propuestos.

A la Corporación Universitaria Minuto de Dios Regional Soacha  
Por darnos la oportunidad de lograr las metas que nos proponemos.

A todos los profesores y  
Demás integrantes de esta institución  
Por guiarnos al cumplimiento de todas las tareas propuestas.

Y gracias a Dios por llevarnos por el camino debido todos los días,  
Y darnos fuerzas para lograr cada acción paso a paso  
Buscando el bien de nuestras vidas y de nuestras familias.

Para todo efecto, aclaramos que el presente trabajo es original y de nuestra  
total autoría; en aquellos casos en los cuales hemos requerido del trabajo de  
otros autores o investigadores, hemos dado los respectivos créditos.

Atentamente,

Miguel Ángel Rojas Ocampo.  
Yeisson Alberto Osorio Guarín.

## CONTENIDO

Pág.

### INTRODUCCIÓN

<b>1.</b>	<b>TEMA</b> .....	<b>20</b>
<b>2.</b>	<b>PROBLEMA</b> .....	<b>21</b>
2.1	DESCRIPCION.....	21
2.2	FORMULACIÓN DEL PROBLEMA. ....	21
2.3	ANTECEDENTES.....	21
<b>3.</b>	<b>OBJETIVOS</b> .....	<b>23</b>
3.1	OBJETIVO GENERAL.....	23
3.2	OBJETIVOS ESPECÍFICOS .....	23
<b>4.</b>	<b>MARCO REFERENCIAL</b> .....	<b>24</b>
4.1	<b>MARCO TEÓRICO</b> .....	24
4.1.1	Los sistemas de información (tics). ....	26
4.1.2	El aprendizaje de la programación orientada a objetos a través de un objeto virtual de aprendizaje (OVA). ....	26
4.1.3	El constructivismo y el aprendizaje significativo en Ausubel. ....	27
4.2	<b>MARCO CONCEPTUAL</b> .....	31
4.2.1	Inicio de la aplicación.....	32
4.3	<b>MARCO ORGANIZACIONAL</b> .....	35
4.3.1	Características de una ova .....	35
4.3.2	Factibilidad técnica .....	35
4.3.3	Factibilidad económica .....	38
4.3.4	Factibilidad legal.....	39
4.4	<b>ESTADO DEL ARTE</b> .....	41
4.4.1	Investigación.....	41
4.4.2	Frecuencias trabajos en clase .....	42
4.4.3	Notas finales.....	44
4.4.4	El aprendizaje de la programación orientada a objetos. ....	48
4.4.5	Metodologías. ....	49
4.4.6	Modelo espiral: .....	49
4.4.7	Ovas y su usos. ....	51
4.4.8	Ventajas y desventajas del software.....	52
<b>5.</b>	<b>ALCANCES</b> .....	<b>54</b>
5.1	DELIMITACIONES .....	55
<b>6.</b>	<b>SISTEMA ACTUAL</b> .....	<b>57</b>
6.1	<b>DIAGRAMAS DE CASOS DE USO.</b> .....	57
6.1.1	Actores. ....	57

6.1.2	Usuario registrados.....	58
6.1.3	Alumno. ....	58
6.1.4	Tutor. ....	59
6.2	<b>DIAGRAMA DE SECUENCIA</b> .....	60
6.2.1	Usuario registrados.....	60
6.2.2	Alumnos.....	61
6.3	<b>DIAGRAMA ENTIDAD RELACIÓN</b> .....	61
6.4	<b>DIAGRAMA DE CLASES</b> .....	61
7.	<b>SISTEMA PROPUESTO</b> .....	62
7.1	<b>DIAGRAMAS DE CASOS DE USO</b> .....	62
7.1.1	Actores. ....	62
7.1.2	Usuarios registrados.....	63
7.1.3	Administrador. ....	63
7.1.4	Alumno .....	64
7.1.5	Tutor. ....	64
7.2	<b>DIAGRAMA DE SECUENCIA</b> .....	65
7.2.1	Usuario registrados.....	65
7.2.2	Administrador – auditoria.....	66
7.2.3	Administrador – recuperación.....	66
7.2.4	Administrador – autorización. ....	67
7.2.5	Alumno – ejercicios.....	67
7.2.6	Alumno – datos personales. ....	68
7.2.7	Alumno – resultados.....	68
7.2.8	Tutor – ejercicios. ....	69
7.2.9	Tutor – datos personales.....	69
7.3	<b>DIAGRAMA DE ENTIDAD RELACION</b> .....	70
7.4	<b>DIAGRAMA DE CLASES</b> .....	70
8.	<b>CRONOGRAMA DE ACTIVIDADES</b> .....	71
8.1	FASE 1 ANÁLISIS Y PLANEACIÓN.....	71
8.2	FASE 2 IMPLEMENTACIÓN DE OBJETIVOS.....	71
8.3	FASE 3 EJECUCIÓN DE PROCESOS. ....	71
8.4	FASE 4 CIERRE.....	71
9.	<b>ARQUITECTURA DE LA SOLUCIÓN DEL SOFTWARE</b> .....	72
9.1	MODELO .....	72
10.	<b>ANÁLISIS Y DISEÑO DE LA SOLUCIÓN DEL SOFTWARE</b> .....	75
10.1	ESTRUCTURA DE ALMACENAMIENTO.....	75
10.2	INTERFAZ GRAFICA DE USUARIO .....	75
10.2.1	Ventajas.....	76
10.3	INTERFAZ DE PROCESAMIENTO DE DATOS .....	77
11.	<b>DESARROLLO Y PRUEBAS</b> .....	78
11.1	PRUEBA 1:.....	78
11.2	PRUEBA 2:.....	78
11.3	PRUEBA 3:.....	78

11.4	PRUEBA 4:.....	79
12.	<b>CONCLUSIONES .....</b>	<b>80</b>
13.	<b>RECOMENDACIONES.....</b>	<b>81</b>
14.	<b>BIBLIOGRAFÍA E INFOGRAFÍA.....</b>	<b>83</b>
15.	<b>ANEXOS.....</b>	<b>85</b>



## LISTA DE TABLAS

	Pág.
Tabla 1: Costos del software utilizado. ....	95
Tabla 2: Costos y beneficios .....	96
Tabla 3: Resultados encuesta Tec. Inf. 1.....	97
Tabla 4: Notas parciales .....	97
Tabla 5: Notas finales .....	98
Tabla 6: Resultados de conceptos básicos en programación.....	100
Tabla 7: Torta de resultados en conceptos básicos en programación. ....	100
Tabla 8: Resultados de los tipos de datos en programación. ....	101
Tabla 9: Torta de resultados de los tipos de datos en programación.....	101
Tabla 10: Resultados de ciclos en programación. ....	102
Tabla 11: Torta de resultados de ciclos en programación. ....	102
Tabla 12: Resultados de manejo de cadenas en programación. ....	103
Tabla 13: Torta de resultados de manejo de cadenas en programación. ...	103
Tabla 14: Resultados de arreglos en programación. ....	104
Tabla 15: Torta de resultados de arreglos en programación. ....	104
Tabla 16: Resultados de poco entusiasmo en los estudiantes. ....	105
Tabla 17: Torta de Resultados de poco entusiasmo en los estudiantes. ...	105
Tabla 18: Resultados de complejidad en los estudiantes. ....	106
Tabla 19: Torta de resultados de complejidad en los estudiantes. ....	106
Tabla 20: Resultados de poca claridad en los estudiantes. ....	107
Tabla 21: Torta de resultados de poca claridad en los estudiantes. ....	107
Tabla 22: Resultados de estrategias de enseñanzas en los estudiantes....	108
Tabla 23: Torta de resultados de estrategias de enseñanza en los estudiantes.....	108
Tabla 24: Resultados de herramientas para estudiar en los estudiantes....	109
Tabla 25: Torta de resultados de herramientas para estudiar en los estudiantes.....	109

## ILUSTRACIONES

	Pág.
Ilustración 1: Encuesta notas de trabajos en clases. ....	43
Ilustración 2: Grafica torta de la notas parciales de los estudiantes. ....	44
Ilustración 3: Porcentajes Nota final del periodo. ....	46
Ilustración 4: Modelo Espiral. ....	49
Ilustración 5: Mapa de actores sistema actual. ....	57
Ilustración 6: Funciones del usuario registrado en el sistema actual. ....	58
Ilustración 7: Funciones del usuario administrador en el sistema actual. ....	58
Ilustración 8: Funciones del usuario Tutor sistema actual. ....	59
Ilustración 9: Proceso de inicio sesión de usuarios registrados en el sistema actual. ....	60
Ilustración 10: Proceso de evaluación de ejercicios sistema actual. ....	61
Ilustración 11: Actores del sistema propuesto. ....	62
Ilustración 12: Funciones del usuario registrado en el sistema propuesto. ...	63
Ilustración 13: Alumno acciones sistema propuesto. ....	63
Ilustración 14: Funciones del usuario alumno en el sistema propuesto. ....	64
Ilustración 15: Funciones del usuario tutor en el sistema propuesto. ....	64
Ilustración 16: Proceso de inicio sesión de usuarios registrados en el sistema propuesto. ....	65
Ilustración 17: Alumno ejercicios sistema propuesto. ....	66
Ilustración 18: Proceso de recuperación de cuenta de usuarios en el sistema propuesto. ....	66
Ilustración 19: Proceso de autorización de alumnos en el sistema propuesto. ....	67
Ilustración 20: Proceso de alumno con ejercicios en el sistema propuesto. .	67
Ilustración 21: Proceso De actualización de datos personales de alumno en el sistema propuesto. ....	68
Ilustración 22: Proceso de visualización de notas del alumno en el sistema propuesto. ....	68
Ilustración 23: Proceso de creación y eliminación de ejercicios del tutor en el sistema propuesto. ....	69
Ilustración 24: Proceso De actualización de datos personales de tutor en el sistema propuesto. ....	69
Ilustración 25: Formulario registro tutor. ....	75
Ilustración 26: Página de inicio. ....	76
Ilustración 27: Consulta de registros. ....	77
Ilustración 28: Modelo entidad relación sistema actual. ....	85
Ilustración 29: Diagrama de Clases sistema actual. ....	86

Ilustración 30: Diagrama de entidad relación sistema propuesto.....	87
Ilustración 31: Diagrama de clases sistema propuesto.....	88
Ilustración 32: Clase opción lista.....	89
Ilustración 33: Clase ejercicios.....	89
Ilustración 34: Clase alumno.....	90
Ilustración 35: Clase consulta. ....	90
Ilustración 36: Clase conexión. ....	91
Ilustración 37: Clase registros.....	91
Ilustración 38: Clase controlacceso. ....	92
Ilustración 39: Clase Administrador. ....	92
Ilustración 40: Clase tutor. ....	93

## LISTA DE ANEXOS

	Pág.
ANEXO I. DIAGRAMA ENTIDAD RELACIÓN SISTEMA ACTUAL.	85
ANEXO II. DIAGRAMA DE CLASES SISTEMA ACTUAL.	86
ANEXO III. DIAGRAMA ENTIDAD RELACION SISTEMA PROPUESTO.	87
ANEXO IV. DIAGRAMA DE CLASE SISTEMA PROPUESTO.	88
ANEXO V. CRONOGRAMA DE ACTIVIDADES.	94
ANEXO VI. COSTOS.	95
ANEXO VII. COSTOS Y BENEFICIOS.	96
ANEXO VIII. ENCUESTAS RESULTADOS.	97
ANEXO IX. ENCUESTA REALIZADA.	99

## GLOSARIO

**AMBIENTE VIRTUAL DE APRENDIZAJE:** Es la organización del espacio y la disposición y distribución de los recursos didácticos mediados por la tecnología que transforma la relación educativa ya que la tecnología facilita la gestión, comunicación y el procesamiento manejo de tiempo y las disposiciones manejo de las interacciones y el tiempo. Es un entorno dinámico con determinadas condiciones, que posibilitan y favorecen el aprendizaje.

**APRENDIZAJE:** Es el proceso en el cual se adquieren o se modifican habilidades, destrezas, conocimientos valores y conductas como resultado de la experiencia, instrucción, razonamiento, observación y del estudio. Es una de las funciones más importantes en humanos, animales y sistemas artificiales.

El proceso principal del aprendizaje es la imitación, la repetición de un proceso observado, que implica tiempo habilidades, espacio, y otros recursos. Según lo define Isabel García: *“El aprendizaje es todo aquel conocimiento que se adquiere a partir de las cosas que nos suceden en la vida diaria, de este modo se adquieren conocimientos, habilidades etc”*.

**APRENDIZAJE SIGNIFICATIVO:** Se refiere al tipo de aprendizaje en que un aprendiz o estudiante relaciona la información nueva con la que ya posee, reajustando y reconstruyendo ambas informaciones en este proceso. Dicho de otro modo, la estructura de los conocimientos previos condiciona los nuevos conocimientos y experiencias, y éstos, a su vez, modifican y reestructuran aquellos. El aprendizaje es recíproco tanto por parte del estudiante o el alumno en otras palabras existe una retroalimentación.

**CONSTRUCTIVISMO:** Es una corriente de la didáctica que se basa en la teoría del conocimiento constructivista. Postula la necesidad de entregar al alumno herramientas que le permitan crear sus propios procedimientos para resolver una situación problemática, lo cual implica que sus ideas se modifiquen y siga aprendiendo.

**ENSAYO:** Consiste en la interpretación o explicación de un tema, sin que sea necesariamente obligado usar un aparato documental, es decir, de manera libre, asistemática y con voluntad de estilo. Se trata de un acto de habla elocutivo.

Un ensayo es una obra literaria breve, de reflexión subjetiva, en la que el autor trata de una manera personal, no exhaustiva, y en la que muestra. Esto último propone crear una obra literaria, no simplemente informativa.

**INGENIERÍA DE SOFTWARE:** Aplicación de un enfoque sistémico, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software, y el estudio de estos enfoques, es decir, la aplicación de la ingeniería al software, ya que integra matemáticas, ciencias de la computación y prácticas cuyos orígenes se encuentran en la ingeniería.

**MAPAS MENTALES:** Un mapa mental es un diagrama usado para representar las palabras, ideas, tareas, u otros conceptos ligados y dispuestos radialmente alrededor de una palabra clave o de una idea central. Se utiliza para la generación, visualización, estructura, y clasificación taxonómica de las ideas, y como ayuda interna para el estudio, planificación, organización, resolución de problemas, toma de decisiones y escritura.

**MAPAS CONCEPTUALES:** Mapa conceptual es una técnica usada para la representación gráfica del conocimiento. Como se ve, un mapa conceptual es una red de conceptos. En la red, los nodos representan los conceptos, y los enlaces las relaciones entre los conceptos.

**OBJETO VIRTUAL DE APRENDIZAJE:** Es un conjunto de recursos digitales que pueden ser rehusados y que puede contener:

- Actividades del aprendizaje a tratar.
- Los elementos que llevan la contextualización de la herramienta.
- Usabilidad.
- Didáctica.
- Interacción.
- Accesibilidad.
- Atemporalidad.

Al tener en cuenta estos puntos el aplicativo será un OVA. Sin estas herramientas, será un objeto informativo.

Es un medidor de aprendizaje que se ha diseñado con la intención de dar una enseñanza y reforzar los conocimientos del aprendizaje.

**PLATAFORMA MOODLE:** "moodle" es una plataforma virtual interactiva, adaptada a la formación y empleada como complemento o apoyo a la tarea docente en multitud de centros de enseñanza en todo el mundo. La difusión

de esta plataforma entre institutos y universidades de todo el mundo va en aumento, estimándose en más de dos millones los usuarios a escala mundial.

**PROYECTO:** Del latín proiectu y podría definirse a un proyecto como el conjunto de las actividades que desarrolla una persona o una entidad para alcanzar un determinado objetivo. Es una planificación que consiste en un conjunto de actividades que se encuentran interrelacionadas y coordinadas. La razón de un proyecto es alcanzar objetivos específicos dentro de los límites que imponen un presupuesto, calidades establecidas previamente y un lapso de tiempo previamente definido. La gestión de proyectos es la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades de un proyecto para satisfacer los requisitos del proyecto. Surge como respuesta a una necesidad, acorde con la visión de la organización, aunque ésta puede desviarse en función del interés. El proyecto finaliza cuando se obtiene el resultado deseado, y se puede decir que colapsa cuando desaparece la necesidad inicial o se agotan los recursos disponibles.

**UML:** (LUM o UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. Ofrece un estándar para describir un "plano" del sistema, incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados.

**V EPISTEMOLÓGICA:** V epistemológica de Gowin Constituye un método para ayudar a estudiantes y educadores a profundizar en la estructura y el significado del conocimiento que tratan de entender (meta conocimiento) y posibilita la incorporación de nuevos conocimientos a la estructura teórico-conceptual que posee el alumno (aprendizaje significativo).

**SERVIDOR:** Es un tipo de software que realiza ciertas tareas en nombre de los usuarios. El término también se utiliza para referirse al ordenador físico en el cual funciona ese software, una máquina cuyo propósito es proveer datos de modo que otras máquinas puedan utilizar esos datos.

Servidor web podría referirse al software, como el servidor de http de Apache, que funciona en la máquina y maneja la entrega de los componentes de las páginas web como respuesta a peticiones de los navegadores de los clientes.

Sistema operativo. Programa o conjunto de programas que en un sistema informático gestiona los recursos de hardware y provee servicios a los programas de aplicación, y se ejecuta en modo privilegiado respecto de los restantes.

Es el software encargado de ejercer el control y coordinar el uso del hardware entre diferentes programas de aplicación y los diferentes usuarios. Es un administrador de los recursos de hardware del sistema.



## **RESUMEN**

### **DESCRIPCIÓN**

Este proyecto va destinado a mejorar los aspectos de la programación orientada a objetos en los estudiantes de informática o practicantes de esta rama. El objetivo es lograr que los estudiantes obtengan aprendizajes significativos por medio de este software, el cual será un tutor virtual, que presentara una serie de ejercicios guiados y otros aspectos de la programación orientada a objetos, por medio de un objeto virtual de aprendizaje.

### **CONTENIDOS**

El proyecto contiene: bases de datos, software, instalador de compilación del software, manuales: usuario y programador, documentación del software y del proyecto.

### **METODOLOGÍA**

En este software se utilizará la metodología uml, basado en el análisis y diseño de sistemas, la plataforma será orientada a objetos con conexión a una base de datos.

### **PALABRAS CLAVES:**

- Programación.
- Objetos Virtuales de Aprendizaje.
- Sistema Gestor De Bases de Datos.
- Servidor Web.
- Ofimática.
- Redes.
- Informática.

## CONCLUSIONES

En el proyecto estos temas que se tomaron son la referencia que van a orientar al proyecto dentro de lo que se quiere desarrollar dentro de las características que se necesitan.

Para aplicar la información suministrada se tuvieron en cuenta los diferentes términos que aplican en el diseño y desarrollo de un software.

Fecha Elaboración resumen    Día  Mes  Año

## ABSTRACT

This project seeks to improve and optimize the learning of object-oriented programming to students of the faculty of engineering and technology in computer systems.

The goal is that students gain meaningful learning through this software, which will be a virtual tutor, who will present a series of guided exercises and other aspects of object-oriented programming, an object through virtual learn.

## INTRODUCCIÓN

Este proyecto busca mejorar y optimizar el aprendizaje de la programación orientada a objetos en los estudiantes de la facultad de ingeniería de sistemas y tecnología en informática.

Para llevar a cabo lo anterior, se requiere la construcción de un software, el cual va ser capaz de dar un soporte de ayuda para los estudiantes y profesores de la facultad de ingeniería de sistemas, tecnología en informática y técnicos en sistemas, que estudien o practiquen la programación orientada a objetos. Por medio de una serie de ejercicios guiados, temas y evaluaciones que aportará un refuerzo en el aprendizaje del mismo.

En principio se tomarán los conceptos básicos, tipos de datos, ciclos, manejo de cadenas y arreglos, en el sistema, que evalúa los conocimientos que el estudiante tiene en el momento y corrige los errores que pueda tener al principio el estudiante en la programación orientada a objetos.

Este sistema lleva una serie de instrucciones y procedimientos para hacer más efectivo el aprendizaje de la programación orientada a objetos, basados en los objetos virtuales de aprendizaje.

Los usuarios para poder acceder al sistema necesitan estar registrados, para tomar sus datos y en caso de algún mal procedimiento por parte de los usuarios, comunicarles su estado como tal en tiempo real por medio de un correo electrónico.

Para este proyecto se quiere implementar el sistema por vía web, para un mayor acceso y cobertura de lo que se quiere aplicar y enseñar en la plataforma, lo cual hace que sea un sistema independiente y se convierta en un ambiente de aprendizaje significativo.

Se utilizarán los diferentes diagramas uml y diseños del software basados en el análisis y diseño de sistemas. Se mostrarán los procesos con los cuales llevamos a cabo este proyecto.

Se hará una base de datos que alojara toda la información soportada en este software para los respectivos ejercicios, registro de usuarios con su respectivo control de acceso, ejercicios suministrados por el software para su desarrollo y los promedios de calificación de los ejercicios desarrollados por los estudiantes más el control de los datos por parte del administrador del software.

## **1. TEMA**

Software para la apropiación del conocimiento de los conceptos básicos, tipos de datos, ciclos, manejo de cadenas y arreglos en programación orientada a objetos, usando objetos virtuales de aprendizaje.

## **2. PROBLEMA**

### **2.1 DESCRIPCION.**

Dentro de muchas instituciones de educación superior incluyendo esta universidad se producen problemas en el aprendizaje de la programación básica y orientada a objetos. Esto se debe a la baja comprensión de los conceptos dados por los tutores en las clases presenciales.

Sus causas pueden ser variadas, por ejemplo falta de conceptos más profundos y mejor descripción de ese tema, una desconcentración en el momento de recibir la información para realizar su análisis y su respectiva ejecución de la aplicación a ese concepto, entre otros casos.

También a la hora de realizar un ejercicio con las herramientas y conceptos que se explicaron en clase, no lo entienden los estudiantes muchas veces porque es muy extenso lo que les toca hacer y no saben cómo desarrollar ese tema.

Se comprende que la programación básica y orientada a objetos no es fácil de aplicaren muchos de sus conceptos, pero en un estudiante de sistemas no se toma con mucha rapidez, por eso la aplicación de estos conocimientos se hace tan difícil. El efecto que produce en los estudiantes de la facultad de ingeniería de sistemas, tecnología en informática, y técnicos en sistemas, es molesto, ya que no puede hallar soluciones y no entiende lo que se le describió para que servía ese concepto y como lo aplicaba. Por eso algunas veces los estudiantes pierden materias y llegan al caso de hasta no continuar estudiando porque se les hace muy complicado estudiar la programación.

### **2.2 FORMULACIÓN DEL PROBLEMA.**

¿Por qué los estudiantes experimentan problemas en el aprendizaje de la programación orientada a objetos?

### **2.3 ANTECEDENTES.**

En la web hay muchos programas videos, libros y tutoriales que hablan acerca de la programación orientada a objetos, pero no dan una definición muy entendible en el estudiante, puede que la solución que se muestra en la web tenga la respuesta internamente, pero su terminología y los ejercicios que se muestran son muy engorrosos y complicados al desarrollarla, también

es porque algunas personas poco inquieren o no interpretan, ni manipulan estas herramientas.

Debido a esto se realizó un estudio dentro de la institución educativa y se tomaron los resultados de una muestra hecha en el semillero de investigación de la Corporación Universitaria Minuto de Dios Regional Soacha donde demuestran que muchos estudiantes no les gusta programar porque hace falta más definición en los términos técnicos, en la forma que se realizan los ejercicios y la poca explicación de los mismos, no solo eso genera la falta de interés en la materia sino que también la deserción del estudio de la programación en la materia, carrera, facultad y se deciden por hacer otras asignaturas o carreras justificando la dificultad de aprender la programación.

Para ello el semillero de investigación ideó la forma de mejorar esos índices de los resultados de esas muestras para ello, buscaron la forma de diseñar un software de aprendizaje significativo basado en la programación orientada a objetos, siendo este un mecanismo de enseñanza que promoverá una mejor experiencia en el estudiante, así obtendrá aprendizajes significativos y no memorísticos. Además será un soporte para mantenerse a nivel en el estudio de la programación; con este software ya no será una tarea difícil de mirar, pasará a ser algo más práctico, más eficiente, pero sobretodo significativo en el aprendizaje y la aplicación de la programación orientada a objetos.

El software se realizó como objetivo de cumplir unos requisitos en la práctica profesional, además de estar en todas las reuniones que se realizaron en el semillero de investigación de la época se desarrolló el software básico solo con los el ingreso del usuario o sesiones y los ejercicios guiados que se montaron dentro de la aplicación, su programación interna fue totalmente estructurada con una base de datos.

Tiempo después se mejoró la propuesta para ser realizada como un proyecto de grado y algo más serio para que quedara al servicio de la comunidad en ese punto se presentaron las respectivas propuestas de proyecto, agregándole más módulos de interacción y más ejercicios, además también se le modificó la programación interna se proponía programación orientada a objetos con interfaz web y colgando los servicios a la web, claro está que se le modificó la base de datos y se hace más grande el proyecto. El proyecto se convierte en un objetivo de graduación y objetivo de suministrarles una herramienta de ayuda a los estudiantes y tutores de la Corporación Universitaria Minuto de Dios Regional Soacha.

### **3. OBJETIVOS**

#### **3.1 OBJETIVO GENERAL**

Construir un software que permita acceder a un portal Web, mediante unas pruebas test que faciliten el conocimiento en los temas: conceptos básicos, tipos de datos, ciclos, manejo de cadenas y arreglos. En la programación orientada a objetos, para los estudiantes de tecnología en informática de la Corporación Universitaria Minuto de Dios Regional Soacha.

#### **3.2 OBJETIVOS ESPECÍFICOS**

- Diseñar una base de datos para la conexión de la información y el alojamiento de las variables, temas e información de registro.
- Elaborar los respectivos modelos de entidad relación y diagramas de clases para el diseño del software.
- Diseñar la interfaz gráfica del software basado en la metodología a utilizarse.
- Identificar actividades de aprendizaje para ser reproducidos en el software a partir que contengan información sobre la programación orientada a objetos.
- Establecer una serie de ejercicios guiados y preguntas para los estudiantes, teniendo en cuenta los conocimientos básicos que un alumno debe tener.
- Desarrollar instrumentos de investigación que permitan la visualización de problemas de aprendizaje.

## **4. MARCO REFERENCIAL**

### **4.1 MARCO TEÓRICO.**

Para implementar este proyecto se toma la Corporación Universitaria Minuto de Dios Regional Soacha. Para la facultad de ingeniería, Tecnología en informática y redes y seguridad informática.

Desde qué punto se podrá utilizar esta aplicación para los estudiantes de primer semestre en delante de las carreras ya descritas.

La aplicación inicio después de tomarse la idea de crear un objeto virtual de aprendizaje que logrará enseñar la programación orientada a objetos de manera fácil y didáctica.

El software se desarrolló en un ambiente grafico básico con colores muy poco agradables al usuario ya que era una presentación beta de dicha aplicación.

El tipo de programación que se utilizó en esa época para desarrollar la aplicación y diseñar el ambiente grafico fue en php estructurado con una conexión a bases de datos, sin el manejo de hojas de estilo css, con división por medio de frames, utilizando software externos como: Dreamweaver versión cs3, creador de menús flash llamado 123 flash menú, y creador de imágenes y animaciones 3d xara 3d.

También se le implemento una base de datos con mysql que tuvo 10 tablas con sus registros respectivos y la conexión de las aplicaciones para la interacción de la aplicación.

Se le desarrollaron unas pruebas iniciales para ver su funcionamiento y como desarrollaba los procedimientos en tiempo real. Para ello se tomaron ejercicios básicos para determinar los resultados finales del proceso de prueba del proyecto.

La aplicación tuvo como virtud las sesiones de usuario, el cual solo ingresaba el usuario a la aplicación y realiza su inicio de sesión y revisaba el contenido de su temario y las evaluaciones como el tipo de preguntas que necesita realizar.



En los ejercicios guiados el estudiante después de ingresar a su sesión de usuario revisa que tipo de temas que hay en la aplicación y desarrollaba los ejercicios que estaban incluidos en la aplicación, revisaba los temas le evalúa el proceso de desarrollo de la aplicación y al final le mostraba un resultado final con las respuestas correctas.

El desarrollo de validación de las preguntas realizadas el estudiante registra sus ejercicios y le mostraba una respuesta para indicar como puede llegar a la respuesta correcta, por medio de una serie de pistas. Claro está que cada pista baja la puntuación de la respuesta contestada.

Finaliza el usuario revisando su calificación final y las respuestas correctas que obtuvo durante la evolución y la experiencia que obtuvo con la aplicación, se cierra la sesión del usuario y finaliza la interacción del software del usuario.

Después de que las pruebas se realizaron con buen rendimiento en la aplicación se reestructuro el sistema cambiándolo todo, mejorando la seguridad en el código y la interfaz con cambios de color que se ajustan a la institucionalidad del aprendizaje.

La programación se cambió de estructurada a programación orientada a objetos y se utilizaron hojas de estilo css para el diseño y la interfaz de la aplicación.

El sentido del software sigue siendo el mismo, pero se le han hecho modificaciones en, lo que corresponde a las sesiones, ejercicios, calificaciones.

Las mejoras más relevantes se producen en el código cuando captura los datos ya que toda la programación de conexión fue cambiada a programación orientada a objetos.

Su contenido es más contextualizado y mejorado en el comportamiento de las evaluaciones y las notas, ya que el resultado no se muestra de forma textual, sino grafica para ver el resultado global de todos los ejercicios realizados, muestra cada tema con su valor total en promedio que obtuvo, para ver el rendimiento del estudiante.

La sesión de maestros puede subir sus ejercicios y los temas que crea que necesita el estudiante, puede ver los resultados de todos los estudiantes, su rendimiento y como fue la participación en el software.

La sesión de administrador monitorea las actividades de los distintos usuarios, sus procedimientos, además es el que realiza las autorizaciones de control de acceso de usuarios y modificación de la información por medio de esta cuenta se obtiene todo el control del software en caso de cualquier inconveniente que se presente a la hora de estar ejecutando la aplicación.

Para que se cumpla con lo estipulado se quiere tener el software implementado en un servidor web o hosting para ejecutar las funciones del aplicativo.

#### **4.1.1 Los sistemas de información (tics).**

Las tecnologías de la información y la comunicación, agrupan los elementos y las técnicas usados en el tratamiento y la transmisión de la información.

El uso de las TICS ayudará a disminuir la brecha digital para optimizar el uso de esas herramientas y sea un medio tecnológico para el desarrollo de sus actividades.

¿Qué relación tienen las tics con la programación y con la ova?

Tienen mucha relación, ya que gracias al desarrollo constante de la informática y las tecnologías derivadas de ella, más las que están relacionadas como lo son la televisión la telefonía y otras. Hacen que esto mejore la forma de contribuir al desarrollo de mejores actores que rodean el entorno.

Las ovas son parte de las tics porque relacionan el aprendizaje dinámico con los estudiantes, la programación vendría a ser la materia prima para conectar estos actores y hacen la mejor ayuda para cualquier persona natural que necesite de él.

#### **4.1.2 El aprendizaje de la programación orientada a objetos a través de un objeto virtual de aprendizaje (OVA).**

La programación orientada a objetos es muy importante en estos momentos ya que todos los programas utilizan este tipo de programación en sus aplicaciones y software de cualquier tipo informático. Esto hace que sea de gran importancia que los estudiantes que tengan conocimientos informáticos y aprendan a manejar este tipo de lenguajes de programación, con esto estamos mejorando la forma de capacitar los estudiantes y programadores que trabajen con este tipo de herramientas.

Los objetos virtuales de aprendizaje son una entidad virtual usada o re usada para el aprendizaje soportado por las tecnologías actuales. Su intención principal es dar soporte y ayuda a los profesionales, estudiantes, maestros y entre otros que quieran interactuar en el circuito de la aplicación, para elaborar y re-estructurar todo tipo de materiales educativos dirigidos a la formación, actualización y actividades de autoestudio.

Un software educativo es mejor ejemplo de cómo aplicar una ova por ejemplo se quiere crear un objeto virtual de aprendizaje para la programación orientada a objetos, que requiere: que sea dinámico, didáctico, que sea comprensivo y que tenga el poder de cautivar al aprendiz por medio de la aplicación, con todo lo necesario para aplicarlo.

Se requieren los temas que debe llevar el aplicativo, ya sea, textos, ejercicios preguntas, audios, videos entre otros. Los cuales sirvan para fomentar un ambiente de aprendizaje, también que el diseño de la aplicación sea amigable con lo que se va manejar o tratar en el OVA.

Para la programación orientada a objetos es mucho mejor poder aplicar los conocimientos en un OVA, ya que en un medio más didáctico el estudiante tendrá forma de conocer más los conceptos que adquirió en modo presencial. Le aportará más información y le mejorara la practicidad a la hora de desarrollar aplicaciones con la programación, le enseñará a mejorar sus capacidades motoras para obtener los conocimientos fundamentales y necesarios en la programación orientada a objetos.

La relación del objeto virtual de aprendizaje con el estudiante y la programación orientada a objetos será más dinámica y sencilla porque le ayudará al estudiante sentirse más atraído a aprender y manejar las herramientas, conocer más conceptos de cada dato y variable, además le dejará la información necesaria para aplicar sus capacidades actuales y mejorarlas.

### **4.1.3 El constructivismo y el aprendizaje significativo en Ausubel.**

En este punto se explicará los conceptos que llevaron a desarrollar el proceso por el cual se está desarrollando el software. Los siguientes aspectos a detallar son:

#### **4.1.3.1 Biografía.**

Nacido en nueva york el 25 de octubre de 1918, Estudió medicina y psicología en la Universidad de Pennsylvania y Middlesex, también estudio

en la New York University, en 1950 acepto trabajo en proyectos de investigación en la universidad de Illinois. En 1976 fue premiado por la Asociación Americana de psicología por su contribución distinguida a la psicología de la educación. Falleció a los 89 años el 9 de julio de 2008; fue seguidor de Jean Piaget, su mayor aporte fue en el campo del aprendizaje y la psicología.

Ausubel considero que el aprendizaje por descubrimiento no debe ser presentado como opuesto al aprendizaje por exposición. Así, el aprendizaje escolar puede darse por recepción o descubrimiento, como estrategia de enseñanza, y puede lograr un aprendizaje significativo o memorístico y repetitivo.

Esto se logra cuando el estudiante relaciona los nuevos conocimientos con los anteriormente adquiridos, es muy necesario que el alumno se interese por aprender lo que se está mostrando.

#### **4.1.3.2 Aprendizaje significativo.**

Se tiene entendido que cuando el docente crea un entorno de instrucción en el que el aprendiz relaciona la información nueva con la que ya posee, reajustando y reconstruyendo ambas informaciones en este proceso. De este modo el Aprendizaje significativo se opone al aprendizaje mecanicista.

Este aprendizaje se da mediante dos factores, el conocimiento previo que se tenía de algún tema y la llegada de nueva información, la cual complementa a la información anterior, para enriquecerla.

En resumen el aprendizaje significativo es aquel que:

Permanente: el aprendizaje que adquirimos es a largo plazo. Produce un cambio cognitivo, se pasa de una situación de no saber a saber. Está basado sobre la experiencia, depende de los conocimientos previos.

La teoría fue postulada en los sesentas por David Ausubel, y propone 4 procesos mediante los cuales puede ocurrir el aprendizaje significativo:

##### **4.1.3.2.1 Subsunción derivada.**

Se adquiere un concepto básico como “árbol” y sus características. Ahora se aprende sobre un tipo de árbol diferente, que se ajusta al concepto anterior de árbol, sin alterar el concepto.

#### **4.1.3.2.2 Subsunción correlativa.**

Ahora se encuentra una nueva clase de árbol con diferentes características en color y tamaño. Para acomodar la nueva información, tengo que alterar o ampliar la información de mi concepto de árbol. En cierto modo se puede decir que este aprendizaje es más valioso que el de subsunción derivado.

#### **4.1.3.2.3 Aprendizaje de supe ordinal.**

Ya que estamos familiarizados con el concepto de árbol y sus diferentes tipos y clases. Hasta que me enseñaron, que estos son todos ejemplos de árboles caducifolios. En este caso ya conocía ya a muchos ejemplos del concepto, pero no sabía el concepto del mismo hasta que me fue enseñado.

#### **4.1.3.2.4 Aprendizaje combinatorio.**

Los primeros tres procesos de aprendizaje implican que nueva información se agrega en una jerarquía de un nivel debajo sobre el concepto ya adquirido. Este aprendizaje es diferente; describe un proceso por el cual la nueva idea sea derivada de otra idea que no sea ni más alta ni más baja en la jerarquía, pero del mismo nivel.

Áreas donde se trabaja el aprendizaje significativo. Se ha trabajado desde el punto cognitivo y afectivo, para las teorías cognitivas, es un proceso mental por el cual se adquieren o se reestructuran los conocimientos; es el resultado de un proceso de reorganizaciones que hacemos sobre nuestro conocimiento con el fin de alcanzar la comprensión de un fenómeno.

Por otro lado, si solo se trabaja el componente afectivo, resultan personas emocionales, intuitivas, pero que carecen de una sustentación teórica que justifique las acciones que realizan.

El aprendizaje también puede definirse como un cambio de actitud. Para que se produzca deben participar en el proceso los tres componentes de la actitud: el cognoscitivo (saber), el afectivo (ser) y el comporta mental (saber hacer).

#### **4.1.3.2.5 Constructivismo.**

Es una corriente de la didáctica que se basa en la teoría del conocimiento constructivista. Postula la necesidad de entregar al alumno herramientas que

le permitan resolver y crear sus propios procedimientos para resolver una problemática.

#### **4.1.3.2.6 El mayor referente es Jean Piaget.**

Otras definiciones son: en el arte es un movimiento de vanguardia que se interesa por la organización de los planos y la expresión del volumen utilizando materiales de la época industrial. Este movimiento nació en Rusia en 1914. En la filosofía es una corriente de pensamiento surgida a mediados del siglo xx. En las matemáticas requiere, para la prueba de la existencia de un objeto matemático, que el mismo pueda ser “construido”. Consiste en suponer que en un objeto “x” no existe “y”, y partiendo de esta premisa, derivar una contradicción no es suficiente.

#### **4.1.3.2.7 Constructivismo en lo educativo.**

Propone un paradigma del proceso enseñanza / aprendizaje y se percibe como un proceso dinámico, de modo que el conocimiento sea una auténtica construcción operada por la persona que aprende.

Se considera al alumno como poseedor de los conocimientos que le pertenecen, con base en eso habrá que construir nuevos conocimientos. A partir de los conocimientos de los educandos previos de los educandos, el docente guía a los estudiantes que logren construir conocimientos nuevos y significativos. Un sistema educativo que adopta el constructivismo como línea psicopedagógica se orienta a llevar a cabo un cambio educativo en todos los niveles.

#### **4.1.3.2.8 Constructivismo y aprendizaje significativo en la informática.**

Basado en un conocimiento previo del alumno y de las teorías ya planteadas por Ausubel, los sistemas y las herramientas que tenemos, ahora se pueden diseñar software o aplicativos de enseñanza y de refuerzo académico, como también sistemas para aprendizaje virtuales que complementen los conocimientos previos del alumno sea cualquier área en la que necesite reforzar sus técnicas y conocimientos del tema a estudiar, puede tomar como instrumentos las herramientas de la informática y sus derivados. Además de crear estímulos y nuevos aprendizajes se fusionan técnicas nuevas de reforzamiento actitudinales y psicopedagógicas como académicas que ponen al alumno en un punto alto de aprendizaje.

## 4.2 MARCO CONCEPTUAL

Aplicación fue diseñada con los con el lenguaje de la programación orientada a objetos, la interfaz trae un diseño suave que ayuda al usuario sentirse amigable con la interfaz y la realización de sus procedimientos normales.

Se utilizaron las hojas de estilo css que llevan sus atributos y clases orientadas a objetos, esto hace que la interfaz sea más robusta, siendo a su vez que se vea mucho más amigable. Cada parte de esta aplicación está construida bajo la programación orientada a objetos.

Se diseñó y creó un sistema gestor de bases de datos para el alojamiento de los datos y archivos que sean subidos el sistema, que hará la ejecución de los datos más ordenada y separada en cada una de sus tablas.

Herencia (En la programación orientada a objetos) es el mecanismo fundamental para implementar la reutilización y extensibilidad del software. A través de ella los diseñadores pueden construir nuevas clases partiendo de una jerarquía de clases ya existente (comprobadas y verificadas) evitando con ello el rediseño, la modificación y verificación de la parte ya implementada. La herencia facilita la creación de objetos a partir de otros ya existentes, obteniendo características (métodos y atributos) similares a los ya existentes.

Paradigma de la programación orientada a objetos. Utiliza objetos como elementos fundamentales en la construcción de la solución. Surge en los años 70. Un objeto es una abstracción de algún hecho o ente del mundo real que tiene atributos que representan sus características o propiedades y métodos que representan su comportamiento o acciones que realizan. Todas las propiedades y métodos comunes a los objetos se encapsulan o se agrupan en clases. Una clase es una plantilla o un prototipo para crear objetos, por eso se dice que los objetos son instancias de clases.

Análisis y diseño de sistemas se refiere al proceso de examinar la situación de una empresa con el propósito de mejorar con métodos y procedimientos más adecuados. El desarrollo de sistemas tiene dos componentes.

JAVA es el lenguaje de programación orientado a objetos, desarrollado por Sun Microsystems . El lenguaje toma de la sintaxis de C y C++, tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. La memoria es gestionada mediante un recolector de basura.

Programación orientada a objetos es el paradigma de programación que usa objetos y sus interacciones, para diseñar aplicaciones y software informáticos. Está constituido en varias técnicas, que incluyen herencia, abstracción, polimorfismo y encapsulamiento. Su uso se popularizó a principios de la década de los 90. En la actualidad, existe variedad de lenguajes de programación que soportan la programación orientada a objetos.

Hojas de estilo en cascada. Proveen del inglés Cascading Style Sheets (CSS). Es un lenguaje usado para definir la presentación de un documento escrito en HTML o XML2 (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

Hosting. Alojamiento o también conocido como hospedaje web, alojamiento web, web site hosting, web hosting o webhosting, es un negocio que consiste en alojar, servir, y mantener archivos para uno o más sitios web. Más importante que el espacio del ordenador que se proporciona para los archivos del sitio web es la conexión rápida a Internet.

Hosting gratuito. El hosting gratuito es extremadamente limitado comparado al hosting de pago. Los proveedores de alojamiento gratuito normalmente requieren sus propios anuncios en el sitio alojado de forma gratuita y tienen límites muy grandes de espacio y de tráfico.

#### **4.2.1 Inicio de la aplicación.**

Para crear este software se tuvo en cuenta los estudios hechos por el semillero de investigación en el año de 2011. Se pidió que fuera un objeto virtual de aprendizaje que pueda enseñar los conceptos básicos de la programación orientada a objetos.

Se diseñó la aplicación del código php estructurado y HTML, se hicieron las investigaciones del autor del aprendizaje significativo y del constructivismo para la realización del software.

Se entregó el software como avance del desarrollo de la aplicación funcionando.



En febrero de 2012 se retomó la idea para implantarla como proyecto de grado, se presentaron las ideas, luego se presentó una pre-propuesta de proyecto de grado basado en objetos virtuales de aprendizaje dando como conceptos de aprendizaje los conceptos básicos de la programación orientada a objetos.

Se presentaron modificaciones a la propuesta y se aprobaron con los lineamientos que se dieron en la propuesta de proyecto, se comenzó a diseñar el sistema con la metodología en espiral con sus respectivas fases de ciclos de vida desarrollo del software descrito.

Realizaron cambios la programación estructurada de la programación orientada a objetos utilizando hojas de estilo en cascada para el diseño de la interfaz, se modificaron los conceptos para utilizar del software.

La implementación de las distintas sesiones de usuarios y su respectivo acceso de control a la aplicación, los ejercicios guiados, temarios, glosario, y resultados para mostrar en la aplicación.

Se crearan manuales de usuario y programador para que se vean las distintas funcionalidades del software y sus interfaces de control.

El software se entrega finalizado con las funcionalidades completamente finalizadas y funcionales dentro de lo establecido en la propuesta inicial.

La experiencia que ofrece este software es para que sea interactivo y obtenga unas competencias en la programación, el software es un objeto virtual de aprendizaje, el cual dará una ayuda al tutor y al estudiante para mejorar el rendimiento tanto académico como aplicativo en su profesión laboral con la carrera que esté desarrollando.

Se utilizará la metodología de Gestión de Proyectos ya que se ajusta a las necesidades de desarrollo del proyecto, tiene como finalidad la planificación, el seguimiento, control de las actividades de los recursos les que intervienen en el desarrollo del proyecto. Como consecuencia de este control, es posible conocer en todo momento los problemas que se producen y resolverlos o paliarlos de manera inmediata.

En el software se ejecutaran una serie de evaluaciones tipo Test. Las cuales arrojarán resultados por cada tema visto o desarrollado. Si el estudiante es capaz de desarrollar la serie de temas, avanzará al siguiente y así hasta terminar el aprendizaje por la herramienta.

Para ello se piensa utilizar un hosting gratuito para aplicar el servicio que se está ofreciendo en el software, en tiempo real y se vean los resultados en tiempo real.

Dentro de los conceptos que tendrá el software se manejarán:

Objetos virtuales de aprendizaje es cualquier entidad, digital o no digital, la cual puede ser usada, re-usada o referenciada durante el aprendizaje apoyado por tecnología. Ejemplos de aprendizajes apoyados por tecnologías incluyen sistemas de entrenamiento basados en computador, ambientes de aprendizaje interactivos, sistemas inteligentes de instrucción apoyada por computador, sistemas de aprendizaje a distancia y ambientes de aprendizaje colaborativo.

Ambientes virtuales de aprendizaje (AVA) o Virtual learning environment (VLE) es un sistema de software diseñado para facilitar a profesores la gestión de cursos virtuales para sus estudiantes, especialmente ayudándolos en la administración y desarrollo del curso. El sistema puede seguir a menudo el progreso de los principiantes, puede ser controlado por los profesores y los mismos estudiantes. Originalmente diseñados para el desarrollo de cursos a distancia, vienen siendo utilizados como suplementos para cursos presenciales.

Estos sistemas funcionan generalmente en el servidor, para facilitar el acceso de los estudiantes a través de Internet.

LMS Learning Manager System es una aplicación de software para la administración, documentación, seguimiento y presentación de informes de programas de capacitación, eventos de la clase y en línea, e-learning programas y contenidos de formación.

LMS van desde sistemas de gestión de la formación y los registros educativos, con el software para la distribución de cursos a través de Internet con características de colaboración en línea. Corporativos LMS uso formación para automatizar el registro y la afiliación de los empleados. Estudiante de auto-servicio (por ejemplo, de auto-registro de un instructor de formación), la formación de flujo de trabajo (por ejemplo, la notificación del usuario, la aprobación del administrador, la lista de espera de gestión), la provisión de aprendizaje en línea (por ejemplo, entrenamiento basado en computadora, leer y entender), evaluaciones en línea, gestión de la educación profesional continua (CPE), el aprendizaje colaborativo (por ejemplo, uso compartido de aplicaciones, foros de discusión), y la gestión de

los recursos de formación (por ejemplo, los instructores, instalaciones, equipos), son las dimensiones de los sistemas de gestión de aprendizaje .

### **4.3 MARCO ORGANIZACIONAL**

A continuación se dará la explicación de toda la organización del proyecto y sus procesos de jerarquía, como también las factibilidades del software.

#### **4.3.1 Características de una ova**

Dentro de este punto se destacaran todo lo que refiere a la organización del proyecto, factibilidades, costos y legales, como también la puesta en marcha y su funcionamiento.

#### **4.3.2 Factibilidad técnica**

Para iniciar este proyecto, se han tenido en cuenta los equipos y el software que se necesitan como requisito para programar y para ejecutar la aplicación dentro de lo que se especifica en este documento.

El software realizó pruebas lo que hacen que sea autónomo y que no tenga problemas durante la ejecución del mismo. El número de pruebas desarrolladas han sido 15 para verificar que el sistema interno de la aplicación funcione correctamente dentro de unas fechas específicas. Que fueron del 12 al 30 mayo de 2012.

Para que el usuario se mantenga en el software se le han desarrollado una serie de ítems que están declarados por criterio y tipo de usuario. Dentro de los ítems están:

- Profesores
- Alumnos
- Administradores

Para lo cual se hizo esta aplicación utilizando la programación orientada a objetos con php y una base de datos, para lo cual se necesitó de unos compiladores de los lenguajes y de la base de datos para ejecutar todo y ver la funcionalidad de la aplicación.

No se requiere mucho recurso humano para este proyecto solo 2 ó 3 programadores que tengan conocimiento de las herramientas que se utilizan en este proyecto. Las personas que son responsables del normal desarrollo

del proyecto y lograr que se obtengan los resultados propuestos, teniendo un líder al frente del proyecto.

Personas a cargo: Miguel Ángel Rojas Ocampo, Yeisson Alberto Osorio Guarín. Diseñadores del software.

#### **4.3.2.1 Requerimientos.**

Debido a que es una aplicación ejecutada en php orientado a objetos, no es necesaria la instalación de otros programas complejos y fuera del alcance de nuestro bolsillo.

##### **4.3.2.1.1 Hardware.**

El software debe ser agregado a un alojamiento que tenga estándares de calidad como XHTML 1.0 y CSS 2.1, teniendo en cuenta que el alojamiento debe tener compatibilidad a cualquier navegador y resoluciones de pantalla.

Equipos portátiles: entradas con puertos USB 2.0 (de alta velocidad), Disco Duro de 320GigaBytes de capacidad (máximo), Tarjeta de Red Ethernet 10/100, Procesador Intel dual core y amd dual core de 2.0 Giga hertzios, Memoria RAM 2 Gigabytes (para mejor manejo de datos y transacciones).

Un equipo tipo desktop con la torre de generación ATX, con puertos USB 2.0 (de alta velocidad), Disco Duro de 320GigaBytes de capacidad (máximo), Tarjeta de Red Ethernet 10/100, Procesador Intel dual core y amd dual core de 2.0 Giga hertzios, Memoria RAM 2 Gigabytes (para mejor manejo de datos y transacciones).

- Monitor **VGA** (Si es LCD, o LED para una mejor experiencia).
- Teclado y Mouse óptico (USB 2.0).

##### **4.3.2.1.2 Red.**

Conexión a Internet (**Banda Ancha**) mínimo, de 512 Kilo Bytes por segundo. Recomendado, Banda ancha de 2 Mega Bytes por segundo.

Módem (**ISP** – Proveedor de Servicios de Internet – Movistar, ETB, UNE, TELMEX, etc.) (Entrada Ethernet RJ45 y RJ11 – para teléfono).

##### **4.3.2.1.3 Software.**

Debido a que el aplicativo web soporta todas las plataformas de Sistemas Operativos (Windows XP, Vista, 7 – Linux – Unix – Mac, etc. En versiones de 32 y 64 bits), es compatible con cualquier clase de hardware y software.

Tener la licencia original y Registrada del Sistema Operativo (si es bajo Licenciamiento, de lo contrario hacer caso omiso a este requerimiento).

Tener instalado Navegadores soportados: google chrome versiones de 15 en adelante. Mozilla Firefox versiones 10 en adelante. (Para una mayor comodidad recomendado para el uso del software). NO recomendamos internet Explorer y opera, ya que afecta su funcionamiento normal y no se verá el diseño de la interfaz bien definido, además presentará algunos bugs ya que no manejan los lenguajes soportados en el software.

La versión más actualizada de los plugins **Adobe Flash Player** (si es posible, para que al momento de ejecutar la página, no se encuentren problemas de cargas en la misma). Para actualizar este software solo ingresar a la página principal de adobe.

Anti-Virus recomendamos un anti-virus que sea de versión gratuita como el Microsoft esencial (solo ejecuta en sistema operativo Windows) u otra marca pero que pueda utilizarse con licencia gratuita.

Tener una versión del plugin de **Adobe Reader X** para la documentación que se pueda leer en pdf. En este caso utilizar la versión gratuita de adobe Reader. No importa la versión ya que el software toma estos archivos y los carga en el sistema de manera más rápida y confiable. Para actualizar este software solo ingresar a la página principal de adobe.

Sistema gestor de bases de datos **MYSQL**.

Compilador de sistema gestor de bases de datos. **Appserver** versión **appserv-2.5.10** arquitectura de **32** bits (si hay uno de 64 bits realizar la acción que se describe) para los sistemas operativos Windows, Linux, mac.

[\(Ver anexos VI\).](#)

### **4.3.3 Factibilidad económica**

Se han propuesto los costos de software utilizados, licencias, versiones y costos - beneficios en este documento. A continuación se darán los respectivos costos y valores de este proyecto en la siguiente tabla de valores.

Para que este software sea factible económicamente se tuvo en cuenta los valores de las aplicaciones usadas los costos fijos de lo utilizado y costos variables del proyecto:

#### **4.3.3.1 Costos fijos.**

Son aquellos costos cuyo importe permanece constante, independiente del nivel de actividad de la empresa. De manera tal que se realice o no la producción, se venda o no la mercadería o servicio, dichos costos igual deben ser solventados por la empresa.

- Alquileres \$0
- Seguros \$30.000 cada uno
- Servicios Públicos (Luz, TE., Gas, etc.).
- Luz \$ 21.000 por mes (6) = 126.000
- Internet 50.000 por mes (6) = 300.000
- Sueldo y cargas sociales de encargados
- Carrera valor 941.000 (2) = 1'882.000
- Salario de los programadores \$ 0
- Manutención y transporte \$ 280.000

(Ver anexos VII).

Para encontrar la información con más detalles en el archivo con extensión [.xlsx] costos.xlsx (documento de Excel formato 2007-2010. No compatible con office 2003).

#### **4.3.3.1.1 Software.**

- Valor del hosting: Gratuito.
- Sistema gestor de bases de datos: \$0 pesos colombianos. GNU gratuito MYSQL.
- Software de diseño de páginas web \$0 pesos tipo de licencia shareware.
- Licencia de sistema operativo: \$450.000 Windows 7 Professional.

- Vida útil estimada 4 años.
- Equipos valores: 850.000 cada equipo con las características ya definidas.
- Vida útil estimas 6 años.

#### 4.3.3.2 Costos variables.

Son aquellos costos que varían en forma proporcional, de acuerdo al nivel de producción o actividad de la empresa. Son los costos por "producir" o "vender".

- Mano de obra directa.
- Mano de obra \$ 0.
- Valor del hosting: Gratuito
- Valor del programa de creación de menús flash Java Script: \$20 dólares aproximadamente \$55.000 pesos colombianos (el software va licenciado por un año).
- Sistema gestor de bases de datos: \$0 pesos colombianos. GNU gratuito MYSQL.
- Licencia de sistema operativo: \$450.000 Windows 7 Professional.
- Equipos valores: 850.000 cada equipo con las características ya definidas.
- Costo del software: se estima que el costo sea de **\$ 6.000.000** estos costos se derivan en el desarrollo y construcción de la aplicación y gastos varios de los programadores derivados en los salarios (tener en cuenta que el salario del programador es un salario básico), transporte y papelería.

#### 4.3.4 Factibilidad legal

En este punto. Este software se rige bajo las normas de la Corporación Universitaria Minuto de Dios, el cual no modificara en ningún aspecto de normatividad interna o externa de la institución. Este proyecto va dirigido a la ayuda de los estudiantes, en ningún momento se modificara el sistema para realizar los normales ejercicios académicos o de la universidad Minuto de Dios.

Será licenciado a la corporación que utilizara el software y se le darán los permisos de uso y reutilización del código para su propia elaboración y construcción.

No se permite que este código sea redistribuido para otras entidades o instituciones con la licencia de la institución que tiene los permisos de explotación del software. Para ello debe adquirir los servicios por individual para que se puedan utilizar las propiedades del código y de la aplicación.

El producto fue creado en código abierto y software libre para el uso del titular, mas no de otros terceros o instituciones y empresas no pertenecientes a la marca titular que utiliza el software en el proceso.

El software es un producto de uso académico, para la Corporación Universitaria Minuto de Dios. Su uso estará vigilado bajo las normas y requerimientos de la universidad, para cualquier modificación y reproducción del mismo.

Está protegido por las leyes de software de dominio público, puede ser modificado y distribuido bajo los derechos del autor principal de la aplicación o los dueños del software.

Se construirá bajo Software libre y código abierto para el desarrollo del lenguaje y los compiladores que se utilizaran para reproducir la aplicación. También es desarrollado bajo las normas del copyleft.

Los navegadores web que se utilizan para compilar este software son gratuitos y en código abierto para que se miren las respectivas modificaciones.

Se utilizarán productos con licenciamientos shareware es denominado como software de evaluación, para desarrollar y utilizar mejoras parciales en el producto desarrollado lo cual no afecta para nada los costos y valores del proyecto.

Gnu General License. Esta licencia es creada para la libre distribución, uso y modificación del software. Esta licencia protege al producto de ser licenciado y ser tomado en autoría por terceros

Hosting Gratuito. Es el servicio que permite alojar a cualquier usuario sus páginas web de manera gratuita, pero a cambio de este servicio el usuario debe dejar publicitar productos y servicios del hosting. En algunos casos el hosting no deja publicitar servicios de otros, sino del hosting para que otras personas o usuarios obtengan sus servicios y se alojen en sus hostings.



#### **4.4 ESTADO DEL ARTE**

El proyecto se pensó desde la fecha de 21 de octubre del 2011 como un proyecto de práctica profesional para ayudar a los estudiantes y tenerlo como un punto de referencia hacia los que continuaran hacer la idea como proyecto de grado.

Desde febrero 11 de 2012 en este momento se ha desarrollado el sistema sesiones y otros aspectos que contiene el software de manera cronológicamente.

Debido a esto se han tomado los resultados de una muestra hechas en el semillero de investigación de la Corporación Universitaria Minuto de Dios Regional Soacha donde se muestra que muchos estudiantes no les gusta programar porque hace falta más definición en los términos técnicos, en la forma que se realizan los ejercicios y la poca explicación de los mismos, no solo eso genera la falta de interés en la materia sino que también la deserción del estudio de la misma y se deciden por hacer otras asignaturas o carreras justificando la dificultad de aprender la programación.

Se tomó como fuente de la investigación las notas de trabajos en clase, parciales y notas finales por semestre de todos los estudiantes registrados al curso de programación durante los años 2007 a 2010. En la Universidad Minuto de Dios; regional Soacha jornada de la mañana, se estableció como universo 1500 estudiantes.

Este proyecto se presentara finalmente en la fecha dispuesta de la universidad y se tendrá listo para esa fecha.

Se quiere diseñar un software de aprendizaje significativo en la programación orientada a objetos, siendo éste un mecanismo de enseñanza que promoverá una mejor experiencia en el estudiante, así obtendrá aprendizajes significativos y no memorísticos. Además será un soporte para mantenerse a nivel en el estudio de la programación; con este software ya no será una tarea difícil de mirar, pasará a ser algo más práctico, más eficiente, pero sobretodo significativo en el aprendizaje y la aplicación de la programación orientada a objetos.

##### **4.4.1 Investigación.**

Basado los análisis desarrollado por la corporación universitaria minuto de Dios Regional Soacha. Muestras tomadas por el docente encargado

Ingeniero Ricardo Bernal Jiménez. Se hizo una valoración de la población de estudiantes en la programación básica que afecta también a la programación orientada a objetos desde el punto de vista del nivel obtenido por el aprendizaje de los estudiantes en las épocas del 2007 al 2010.

Se tomó como población del universo alrededor de 1.500 estudiantes de tecnología en informática. Su fuente para obtener los resultados propuestos son las calificaciones en los corridos de sus semestres con notas parciales y finales de cada estudiante de informática.

Los resultados de esas muestras se verán en las tablas que se muestran en este documento.

[\(Ver anexo VIII\).](#)

#### **4.4.2 Frecuencias trabajos en clase**

El segundo índice calificado más significativo está ubicado en el rango 33,2-41,6 en el cual muestra que son 210 estudiantes los tuvieron buenas calificaciones, se observa que este número es porcentualmente parecido al número de estudiantes con calificaciones regulares el cual corresponde al 38% de la comunidad educativa.

El estudio nos indica que 55 estudiantes fueron evaluados como excelentes, lo cual corresponde al 10% de todos los estudiantes inscritos en la materia programación básica para el periodo comprendido entre 2007 al 2010.

De acuerdo a lo anterior los estudiantes que requirieron apoyo extra por parte del docente en la comprensión de los temas tratados y evaluados como trabajo presencial en clase fueron de 290 los cuales representan algo más del 50% de todos pues quienes no requirieron apoyo ascendieron a 265 según la tabla de frecuencias.

En estas condiciones se puede concluir que es necesario disminuir el número de estudiantes que se catalogan como regulares, insuficiente y deficiente los cuales para el caso tratado son el 52%.

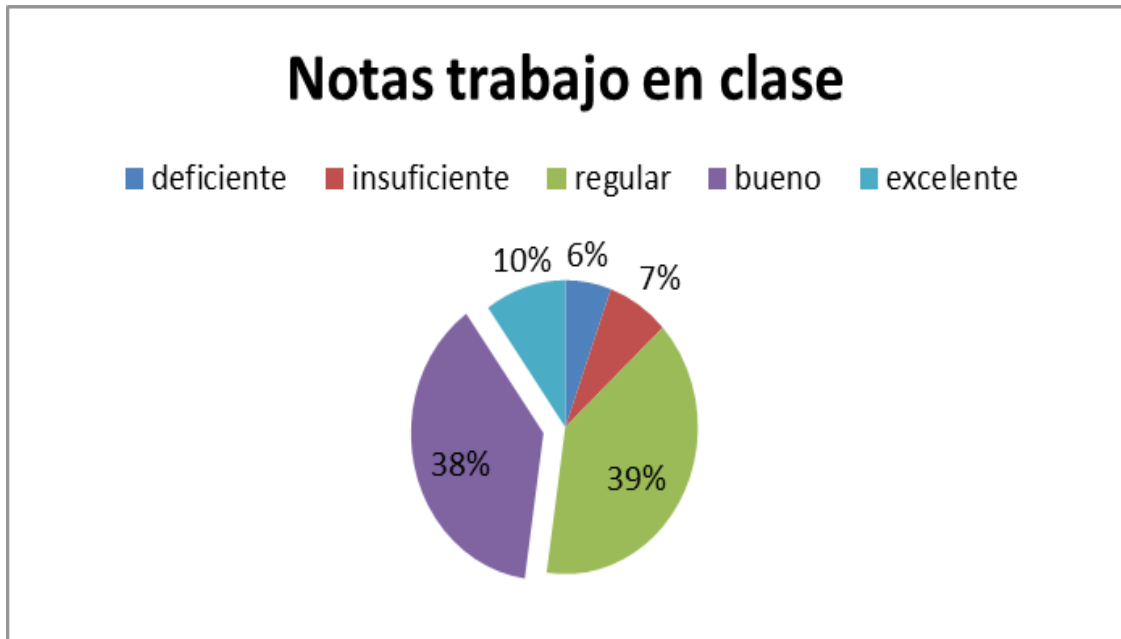


Ilustración 1: Encuesta notas de trabajos en clases.

Para ver la tabulación de los datos de esta ilustración (ver en anexos) la tabla 3: Resultados encuesta Tec. Inf.

Muestra que 20 estudiantes obtuvieron un índice calificado como deficiente, el cual corresponde al 3% de la población de estudiantes bajo estudio, en este caso son estudiantes que por lo general no han tenido un comportamiento adecuado en la realización de sus tareas en casa y en las actividades presenciales del curso.

Con las notas obtenidas en las evaluaciones nos podemos dar cuenta que la cantidad de estudiantes que necesitan ayuda es de  $(20+107+154)$  281 del total de la muestra; esto quiere decir que el 45.2% los cuales están son catalogados por este documento con capacidades regulares, insuficientes e deficientes para tomar el curso de programación básica desarrollada por contenidos propios del lenguaje.

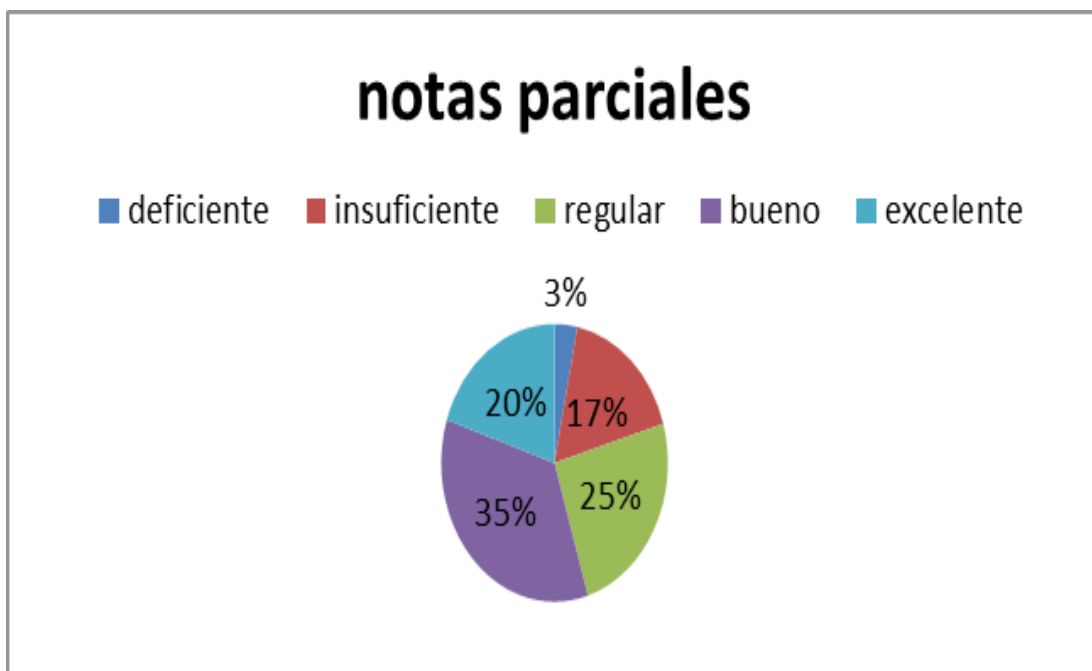


Ilustración 2: Grafica torta de la notas parciales de los estudiantes.

Para ver la tabulación de los datos de esta ilustración (ver en anexos) la tabla 4: Notas parciales.

#### 4.4.3 Notas finales.

Recordando que la nota final es la sumatoria de las notas de trabajos en clase y en casa con la nota del examen parcial según las cuales las primeras tienen un peso del 40% y 60% respectivamente se desarrolló la Tabla No.3 la cual se muestra a continuación. En este caso se tomó una muestra de 561 por ser el número total de registros del universo de la investigación para esta variable.

Según la tabla y siguiendo el análisis que se ha venido dando tenemos que 9 estudiantes obtuvieron un índice calificado deficiente, el cual corresponde al 1% en este caso y al tratarse de la nota final la apropiación del conocimiento fue nulo.

Los estudiantes que mostraron muy bajo rendimiento pero que no se puede decir que no aprendieron nada fue de 35; representando el 6%. Este grupo puede considerarse como no adecuado para la comprensión de los contenidos planteados por el plan de trabajo propuesto.

Aunque la universidad estima que el curso es superado con notas superiores a 30; en este caso no se trata de observar a partir de ese límite; sino dentro de los rangos establecidos para plantear una estrategia sobre la cual se pueda planear aumentar la cantidad de estudiantes que pueden considerarse buenos y excelentes.

De la muestra 268 estudiantes obtuvieron un índice calificado como regular, es decir; con notas entre 24,8 y 33,2 el cual corresponde al 48% (ver gráfico n° 3). La cantidad nos indica que se puede trabajar proponiendo estrategias de retroalimentación en las nuevas generaciones para disminuir el impacto que representa esto en los niveles superiores. 195 estudiantes obtuvieron un índice según el cual; este estudio los toma como de buen nivel de apropiación del conocimiento, corresponde al 35% y se estima que en este grupo están los estudiantes que están bien ubicados y cuyo futuro estará en función de propuestas de tipo informático

La muestra indicó que 54 estudiantes obtuvieron un excelente resultado esto corresponde al 10% del total de los estudiantes que se matricularon al curso de programación básica durante los años 2007, 2008, 2009 y 2010.

De acuerdo a lo anteriormente planteado 312 estudiantes fueron susceptibles de apoyo en el curso. Su desempeño no es el adecuado con lo esperado por la universidad y permite observar que en gran medida los índices de adaptación a los lenguajes de programación estructurada implica la necesidad de proponer estrategias nuevas que puedan ser puestas en funcionamiento en las nuevas promociones con el fin de reducir este número que representa el 55.6%, El 44.4% de los estudiantes matriculados al curso de programación básica supero satisfactoriamente los contenidos y representa la verdadera producción del curso durante el periodo.

Como se observa en la Grafica No. 3 el área más representativa es de estudiantes regulares; es decir aquellos que pueden ser recuperados y que en algún momento pueden superar los logros propuestos.

Solo el 7% de la muestra no se adaptó adecuadamente; lo que permite proponer como una proposición válida que aunque la Universidad no tiene como política realizar exámenes de ingreso a ninguna de sus carreras, esto realmente no es importante a la hora de establecer si los estudiantes están o no adecuadamente ubicados en lo que realmente quieren estudiar.

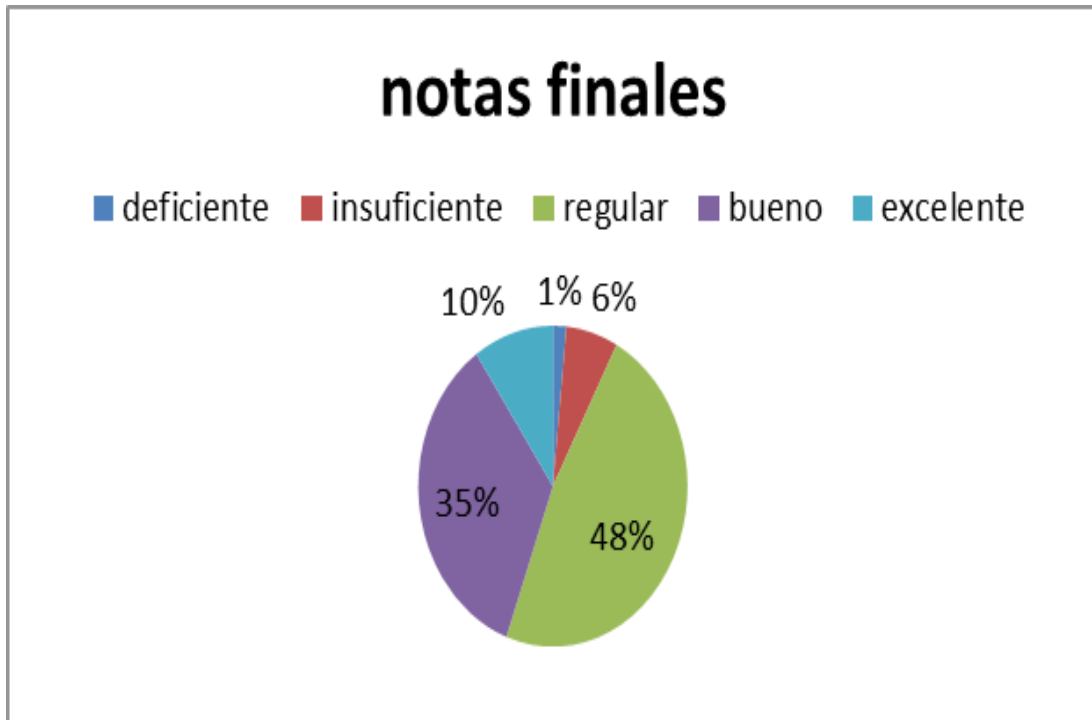


Ilustración 3: Porcentajes Nota final del periodo.

Para ver la tabulación de los datos de esta ilustración (ver en anexos) la tabla 5: Notas finales.

#### 4.4.3.1 Conclusiones.

De acuerdo a lo observado solo el 7% de los estudiantes que la Universidad Minuto de Dios Regional Soacha recibió en su curso de programación básica durante los últimos 4 años mostraron deficiencias de comprensión y adaptación a los lenguajes de programación estructurada.

En promedio el 45% de todos los estudiantes superaron adecuadamente el curso y se adaptaron a las estructuras de programación que se propusieron desde los contenidos del programa académico para la materia desde los años 2007, 2008, 2009 y 2010.

Se observa que el nivel académico de los estudiantes de primer semestre que tienen contacto con la programación en C; se pueden apoyar a partir de nuevas estrategias que permitan reducir el porcentaje de estudiantes regulares que en el periodo resulto ser de un 48%.

Una estrategia valida es la implementación de ambientes V\_learning a través de los cuales las clases presenciales puedan ser retroalimentadas a partir de

ambientes E\_learning usando plataformas de ambientes virtuales de aprendizaje como moodle.

Lo que queda claro es que no necesariamente es importante proponer exámenes de ingreso ya que según este estudio solo el 7% mostro desadaptación al programa en una materia que es el soporte epistemológico del programa.

Con el apoyo de las TIC como herramientas se puede disminuir el porcentaje de estudiantes que en este estudio fueron catalogados como regulares con el fin de entregar a la sociedad un mayor número de tecnólogos adecuadamente preparados.

Anexo el documento del cual se tomaron las muestras (instrumentos.docx), este documento se encuentra dentro de la tesis digital (CD-ROM) del software de este proyecto.

#### **4.4.4 El aprendizaje de la programación orientada a objetos.**

La programación orientada a objetos es utilizada por muchos programadores de informática; ya que es el lenguaje de programación más usado y por el que optan varios a desarrollar sus aplicaciones, o el distinto software de uso para cualquier empresa o ente gubernamental.

Por eso se hace de gran importancia que el aprendizaje de la programación orientada a objetos se instruya desde la programación básica, para que sus fundamentos y características se conozcan con mayor fluidez y amplitud en los conceptos y manejo de las distintas herramientas que lo componen, como también los compiladores que lo pueden utilizar.

La programación orientada a objetos no se debe ver como una barrera más, al contrario debe tomarse como un lenguaje práctico, dinámico y mucho mejor de lo que se trata en la seguridad para el diseño de aplicaciones.

Ahora bien los estudiantes de facultades en ingeniería, tecnólogos, técnicos u otros apasionados por los lenguajes de programación y la informática, deben tener en cuenta que no es fácil aprender este tipo de programación; pero con constancia y tiempo pasará de un lenguaje complicado a algo más divertido y mucho más simple. Pero aquí no viene el problema, la dificultad de los estudiantes o aprendices no tomen la programación como un punto fuerte, es por la falta de interés y las pocas herramientas que se les dan, además de que algunas veces no los instruyen bien en los fundamentos básicos de programación.

Las dificultades no solo son las que se describen antes, también están por el no conocimiento de terminologías aplicadas en la programación, lo otro es que falta más conceptualización o descripciones más entendibles que le ayuden a los aprendices de programación orientada a objetos conocer más sus herramientas y sus usos posibles.

En el transcurso del tiempo, han aportado diferentes alternativas de aprendizaje de la programación orientada a objetos y otros tipos de lenguajes de programación, como también otras áreas de aprendizaje desde otras vertientes ya sea física, matemáticas, entre otros, además eso ha sido importante en el aprendizaje de los estudiantes ya que el aporte ha sido fundamental para la mejora de los conocimientos en los estudiantes y tener mucha más conceptualización en los diferentes temas de estudio.



Las tecnologías de la información dieron su grano de arena para este gran avance en la enseñanza y la aplicación del aprendizaje visto desde cualquier óptica incluso en la propia informática y los lenguajes de programación para la obtención de conceptos y aprendizajes académicos, que servirán en un presente – futuro para tener mejores profesionales con más capacidad y retentiva de información.

#### 4.4.5 Metodologías.

Se utilizara el lenguaje unificado de modelado (UML) para realizar sus respectivas funciones de los usuarios en el sistema. Los diagramas de casos de uso establecerán el comportamiento de los siguientes usuarios: tutor, alumno, administrador del sistema e invitados en el portal Web.

Este software se realizara por prototipos y desde ahí se empezara a manejar el ciclo de vida espiral, obteniendo los fallos existentes en el software propuesto.

#### 4.4.6 Modelo espiral:

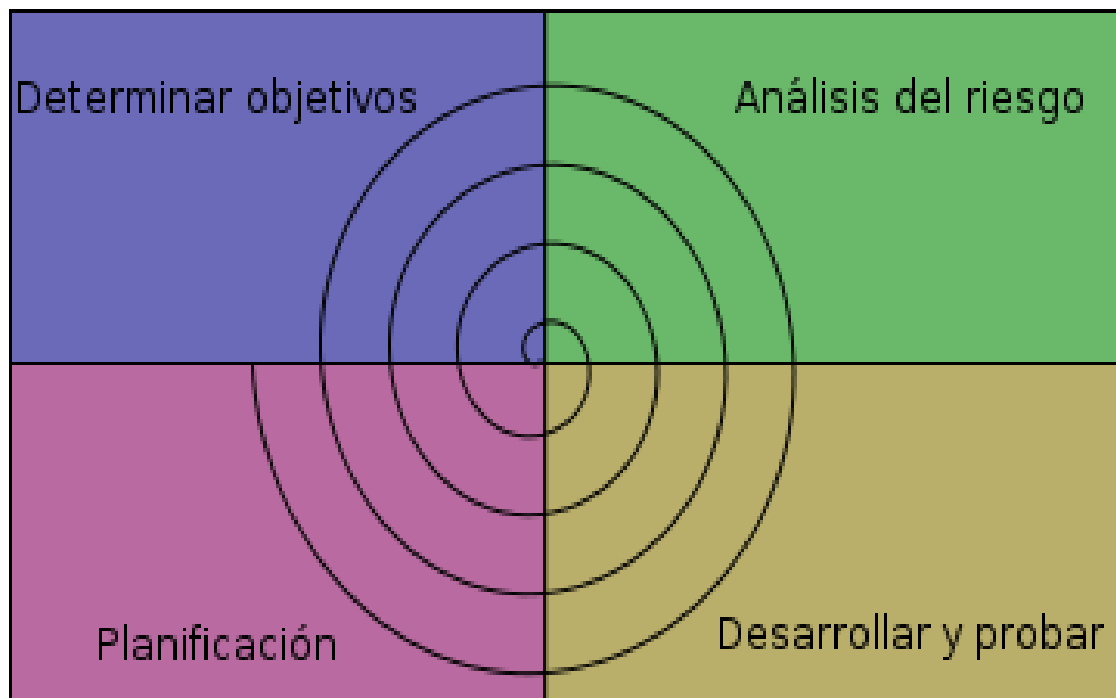


Ilustración 4: Modelo Espiral.

#### **4.4.6.1 Determinación de objetivos.**

Estableceremos los requerimientos y las factibilidades posibles en el sistema como la arquitectura del computador, exploradores de Internet que puede utilizar el sistema, sistema gestor de base de datos y lenguaje de programación compatible a utilizar. También los castros valores que tendrá el software y los riesgos legales que podría tener o que factores legales estará expuesto el software.

Dentro de la restricciones se deberá llevar el proceso de modela miento del software por módulos, teniendo en cuenta si implementado a un portal web proporcionarle más seguridad web para evitar el hacheó del software. Como sabemos el software posee una seguridad básica.

#### **4.4.6.2 Análisis de riesgos.**

Realizaremos recolección de información para determinar los posibles problemas existentes, amenazas y daños del sistema al interactuar los usuarios se revisaran los posibles errores que se puedan generar en el momento de uso de la aplicación. Adicionando que el software es orientado a objetos (php) dentro de su desarrollo produjo una serie de errores, tomando como alternativa la metodología mosler que implica mucho en los procesos llevados en el software, sabiendo que debemos basarnos en evaluaciones para poder generar un modulo que tenga buen rendimiento.

Basados en unos criterios de riesgo partimos que en el software la interacción del HTML conjunto con php tiene muy pocas restricciones de uso y de manejo para los usuarios, para ello se observo la serie de errores que podrían ser comunes a la hora de estar en tiempo real.

Siendo claros de que en este punto abarco el encuentro de generar más errores en las funciones como:

- Creación de un ejercicio.
- Validación de un ejercicio.

Siendo estas funciones desarrolladas en tiempo real en navegadores de poca estabilidad debidamente se proporcionaba un porcentaje de estabilidad de la función tenido en cuenta las ventajas como el iniciar una nueva fase desarrollo para la estabilidad de los módulos.

Se proporciono una estabilidad para el cliente por medio de validaciones estrictas de cada modulo del software, Llevando este proceso a su vez a la corrección de errores básicos del software.

#### **4.4.6.3 Planificación.**

Desde el modificaremos que proceso debe seguir y cual proceso se puede mejorar tanto para los estudiantes y tutores se determinaran los usos que tendrá el software, que aplicaciones ejecuta y donde será ejecutado.

Para el software tuvimos una estrategia de auditorías en los proceso del tutor y del alumno desde este modulo se llevo el seguimiento a cada función del software y observando el comportamiento del motor gestor de base de datos al trabajar en conjunto con el software, se tuvo encuentra que mysql tiene buena estabilidad pero el navegador como mozilla firefox genera error de duplicidad en los datos. Para este error planificamos que debíamos manejar navegadores con una estabilidad y que un navegador beta en tiempo real con el servidor local no es lo más estable.

Dentro de algunas tareas que se tuvieron más en cuenta son:

- **Entrada de datos:** Donde en esta parte debía ser implementada en una clase con un mayor rendimiento en los datos y proporcionando una mejor ocultación de los datos recibidos de un formulario en tiempo real.
- **Cuerpo HTML:** Se incluyo en una clase cada función HTML que era utilizada en el software, Se utilizo para mayor estabilidad del software en el manejo de cajas de textos, botón, span, formularios y etc.

#### **4.4.6.4 Desarrollar y probar.**

Se ejecutara desde el servidor que contenga un sistema gestor de bases de datos y un sistema apache, se utilizaran para estar probando los módulos realizados en DreamWeaver con formato PHP orientado a objetos.

#### **4.4.7 Ovas y su usos.**

El objeto virtual de aprendizaje es una unidad digital cuyos elementos constitutivos están íntimamente relacionados entre sí y tienen un propósito didáctico, del cual podemos reconocer como elementos constitutivos

mínimos: objetivos, contenido, contextualización, evaluación y datos que permitan su almacenamiento, identificación y recuperación.

Su mayor uso es para explicar un tema del cual no tenga nociones la persona o el usuario a utilizar la aplicación se integran los complementos para que el usuario conozca la forma adecuada de utilizar estos conceptos u objetos y temas que se le den el objeto virtual de aprendizaje.

Los gobiernos han habilitado el uso de las tecnologías de la información para estos fines académicos y poder dar más elementos tanto para el docente como para el alumno o el usuario que esté dispuesto a aprender a conocer los temas conceptos y otras formas de aprendizaje bajo estas condiciones, y los objetos virtuales de aprendizaje dan valor al uso de las tecnologías de la información y su respectivo acceso.

Con esto las personas que utilizan estas herramientas mejoran su conocimientos y su actividades de manera más eficaz y dinámica, además se garantiza que los conocimientos adquiridos mediante estas plataformas no se perderá, por eso se crearon para mejorar la retentiva de cada actor que usa un objeto virtual de aprendizaje y tiene mejor conocimientos del uso de las tecnologías de la información.

#### **4.4.8 Ventajas y desventajas del software.**

Dentro de este punto se conocerán las ventajas y desventajas del software, así como también sus descripciones que se mostraran a continuación:

##### **4.4.8.1 Ventajas.**

Dentro de las ventajas que destacamos en este software es el diseño de la interfaz que es muy amigable para el usuario en el momento de interactuar con la aplicación.

El sistema permite que el maestro suba los ejercicios y los temas que el estudiante quiere ver y que pueden ayudarle a mejorar sus resultados.

Muestra los resultados en gráficas para los maestros y los estudiantes de las pruebas realizadas por parte de los estudiantes que ingresaron a la aplicación, muestra resultado por cada tema y en general sus procedimientos.

El costo de este software fue bajo porque las aplicaciones manejadas son de software libre y el código es abierto.

#### 4.4.8.2 Desventajas.

Falta agregar más componentes de manejo de diseño en el sistema Bugs encontrados por el diseño de la interfaz gráfica para el usuario. La no compatibilidad con algunos navegadores web.

- **Cosas por mejorar respecto a otros software similares:**
  - El diseño de la interfaz gráfica.
  - La compatibilidad con los navegadores web.
  
- **Cosas favorables respecto a otros software similares:**
  - Su manejo de sesiones con respecto a otro software.
  
  - Su bajo costo en uso de elementos para el desarrollo de la aplicación.
  
  - A comparación con la plataforma moodle es su desarrollo en los tipos de ejercicios por la variedad y que promueve mayor aprendizaje dentro de la aplicación.
  
  - El manejo de sus estadísticas y resultados en tiempo real, la realización de ejercicios dentro del mismo software, las sesiones, las auditorias y la autorización para el ingreso al software.

## 5. ALCANCES

Este sistema se diseña para darle solución a los problemas que presentan los estudiantes de informática y de la facultad de ingeniería de sistemas y o afines, respecto a la programación orientada a objetos de la universidad, también puede estar disponible para las distintas universidades del país, ya sea con el consentimiento de la Corporación Universitaria Minuto de Dios.

Se toma como punto de partida la Universidad Minuto de Dios, siendo esta la prioridad en la mejora de las falencias que los estudiantes presentan en la programación orientada a objetos, y les proporcionen conocimientos significativos de información.

Para ello se implementará el software, para mejorar las capacidades del estudiante de la facultad de ingeniería, tecnología en informática, técnico en sistemas y/o afines a la carrera. Este software realiza evaluaciones tipo test para obtener un rendimiento de los estudiantes, se verán los avances del mismo en el tiempo que realice las pruebas, mejorará sus conocimientos básicos y obtendrá más información que le servirá de base para su aprendizaje, tiene manejo de sesiones para saber qué rol ocupa cada actor, por ejemplo si es estudiante, si es maestro o si es administrador.

Otra función que puede tener este software, no solo se planifica para que enseñe lo debidamente marcado en el documento, sino también cuando otros desarrolladores puedan tomar posesión de la actualización del mismo pueden agregarle más plataformas de enseñanza para las distintas áreas de aprendizaje de la carrera, sin embargo deben estar orientados bajo las ideas de esta aplicación y manejar las ideas planteadas desde el momento que se creó este software.

Dentro de las fases que se utilizaron para lograr construir este software se han dado unas pautas de desarrollo para el avance y construcción del proyecto.

En la fase inicial se abarcaron los diseños del sistema propuesto, recolección de la información para desarrollar los ejercicios, fuentes relacionadas, construcción de la base de datos, diseño de la interfaz gráfica y presentación de la propuesta.

Luego se implementaron los objetivos a realizarse dentro de los cuales se encuentran partes de la información de la fase inicial, constan de los análisis previos de requerimientos de costos y software que se utilizará en el software.

Se desarrollaron ejecuciones y se revisaron los procedimientos de la aplicación como también del documento o afines para que el continuo avance no se vea estropeado en ningún momento, se revisaron todos los procedimientos de aplicación para que no se vea afectada y en caso dado se pueda mejorar la aplicación en base de las ejecuciones realizadas y los procedimientos dentro de su codificación interna, se revisaron los objetivos del documento para ver si se han cumplido hasta el punto de todo lo realizado.

Por último se verán los resultados que se quieren obtener por todo lo realizado se entregara el software terminado hasta el punto delimitado con los requerimientos escritos e investigados, se tendrá en cuenta que el ciclo de vida sea cumplido hasta el punto de cierre.

Para ello se tuvo en cuenta un presupuesto inicial (el cual se encuentra en la Tabla 1: Costos del software utilizado.), el tiempo estimado del proyecto fue aproximadamente de 6 meses contando los días laborales únicamente.

El software será controlado y utilizado por la corporación universitaria minuto de Dios regional Soacha con fines académicos y mejorar el rendimiento de los estudiantes. Con ello se quiere que la aplicación de cobertura al aprendizaje de todos los estudiantes de la carrera descrita.

## **5.1 DELIMITACIONES**

El proyecto, pretende crecer e implementar nuevas tecnologías que con el tiempo que se vayan acoplando. Durante este tiempo se pretende realizar el levantamiento de información, antecedentes, bases teóricas de la aplicación y las teorías basadas en la Tecnología en informática. Se desarrolló este software para los estudiantes de la Corporación Universitaria Minuto de Dios que tengan dificultades durante el aprendizaje presencial de la programación orientada a objetos.

Éste software es de apoyo y de refuerzo para los estudiantes de informática ya sea técnicos, tecnólogos, ingenieros o afines de la corporación Universitaria Minuto de Dios Regional Soacha. Solamente servirá para evaluar y medir el nivel de programación del estudiante.

El software contiene evaluaciones temas y conceptos que se manejan en la aplicación, será un medidor de aptitudes del aprendizaje será de apoyo para maestros y estudiantes, está diseñado para mejorar el rendimiento del estudiante y agilizar los temarios del maestro en la enseñanza del mismo.

Los temas que se manejan dentro del software son: conceptos básicos, tipos de datos, ciclos, manejo de cadenas y arreglos, en programación orientada a objetos. Este software será implementado en un Portal Web para la accesibilidad de los contenidos y resultados generados por el alumno que se verán en tiempo real.

El presente proyecto de estará determinado por la metodología a trabajar que permita la identificación de las necesidades de los usuarios del sistema el cual una vez realizado se actualizará con la información que se vaya produciendo, de modo que pueda usarse indefinidamente en el tiempo dado el carácter universal de su metodología.



## 6. SISTEMA ACTUAL

Se presenta la metodología, en la cual se definen las actividades, se organizan cronológicamente de acuerdo a sus requerimientos de precedencia y aprovechamiento de recursos. Como resultado se obtiene unas actividades que refleja lo que se va a hacer en el proyecto.

### 6.1 DIAGRAMAS DE CASOS DE USO.

Utilizaremos los diagramas de casos de uso ya que se ajusta a las necesidades de desarrollo del estudio, está tiene como finalidad principal la planificación, el seguimiento y control de las actividades y de los recursos humanos y materiales que intervienen en el desarrollo del proyecto. Como consecuencia de este control es posible conocer en todo momento qué problemas se producen y resolverlos o paliarlos de manera inmediata. Veremos lo que se tiene actualmente, con esto se tendrá como base para aplicar las modificaciones pertinentes en el proyecto.

#### 6.1.1 Actores.

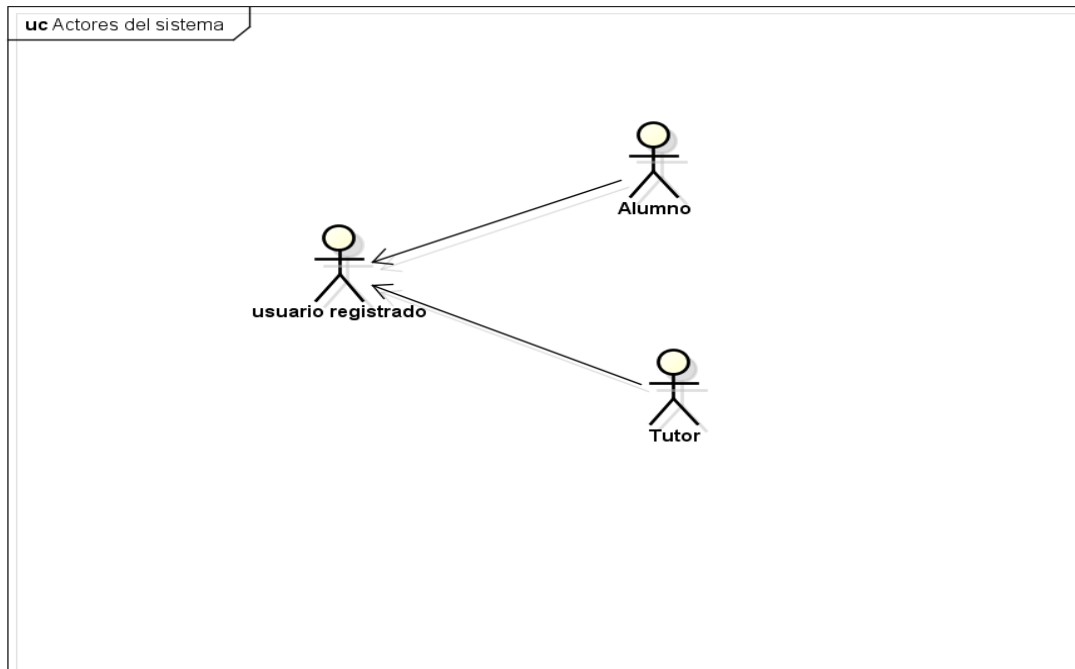


Ilustración 5: Mapa de actores sistema actual.

### 6.1.2 Usuario registrados.

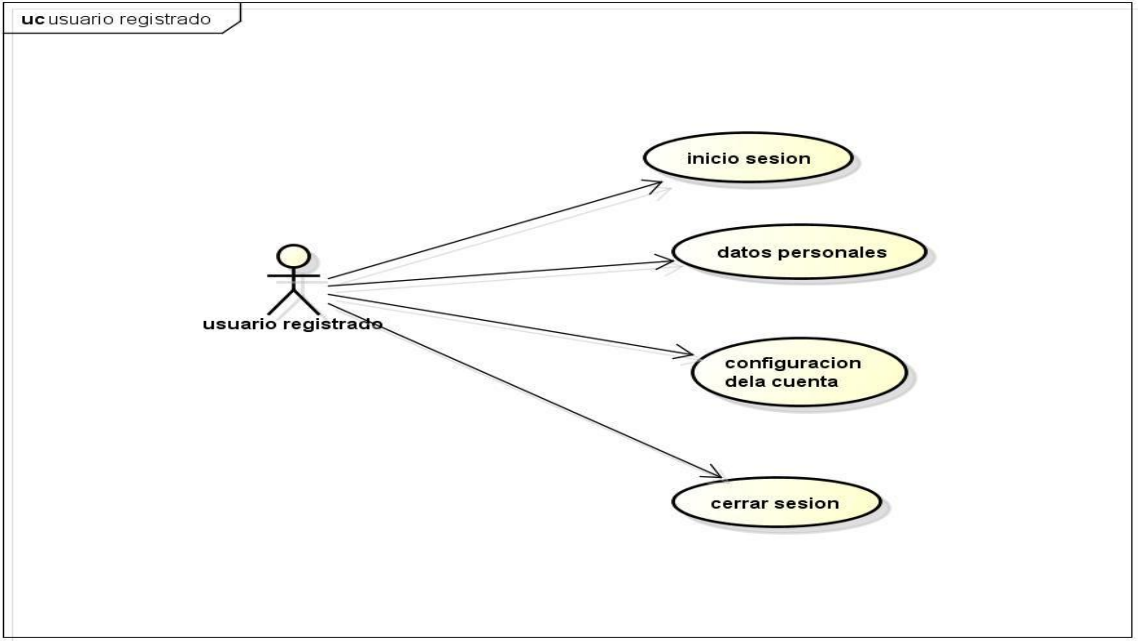


Ilustración 6: Funciones del usuario registrado en el sistema actual.

### 6.1.3 Alumno.

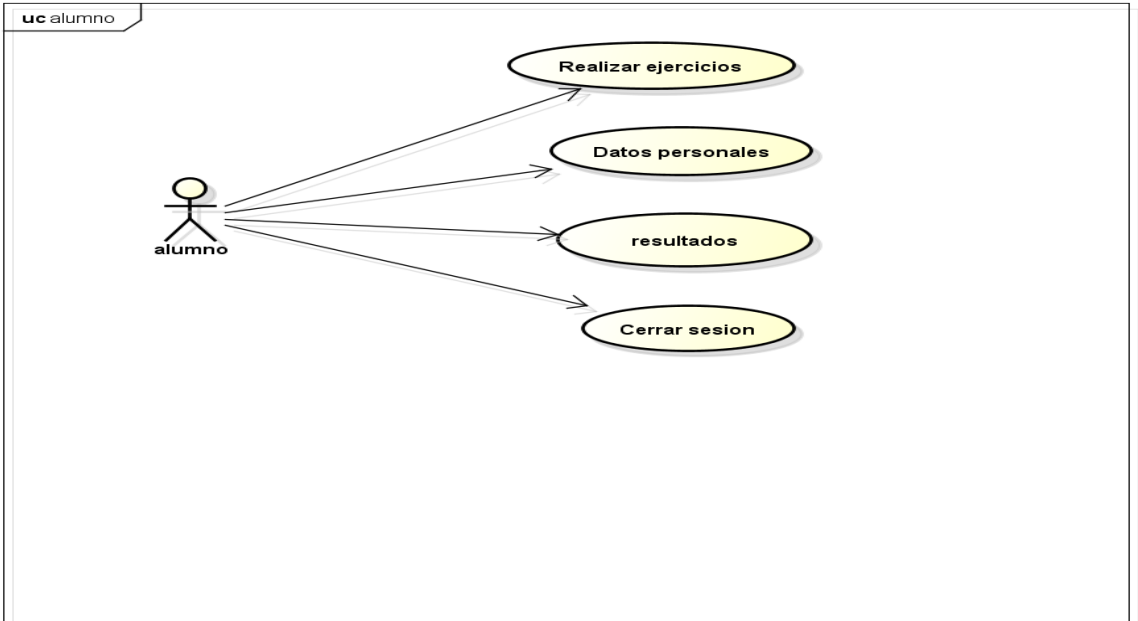


Ilustración 7: Funciones del usuario administrador en el sistema actual.

**6.1.4 Tutor.**

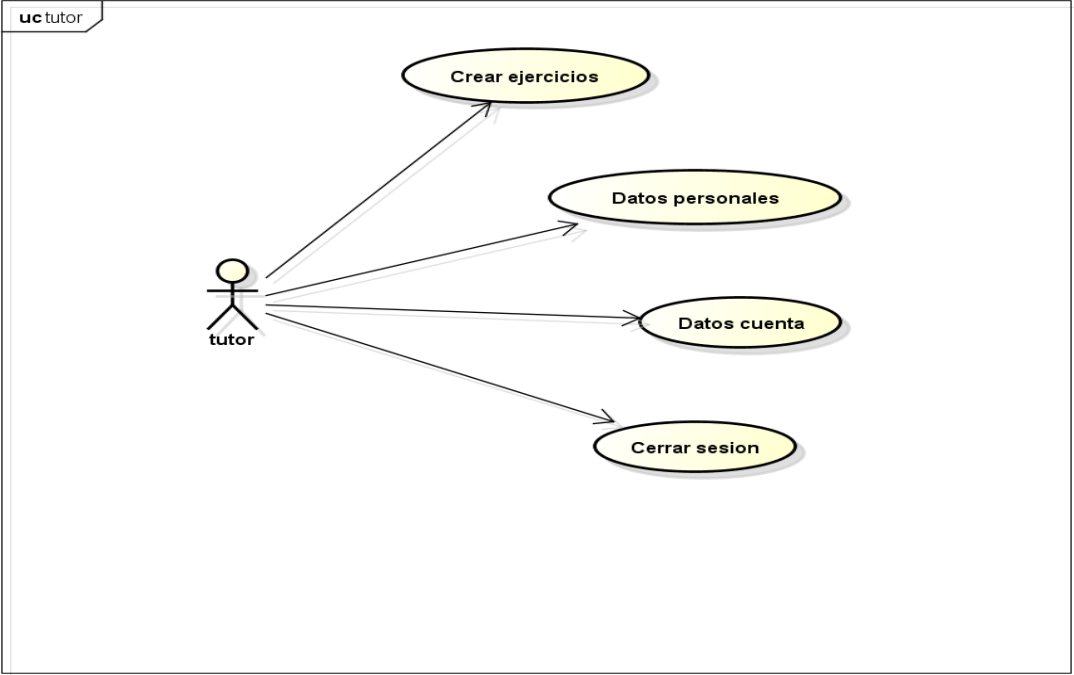


Ilustración 8: Funciones del usuario Tutor sistema actual.

## 6.2 DIAGRAMA DE SECUENCIA

En esta etapa se deben llevar a cabo las actividades de manera secuencial en el momento que se estaba actuando sin la propuesta del proyecto, lo que permitirá dejar muy claro lo que se quiere lograr con el proyecto.

### 6.2.1 Usuario registrados.

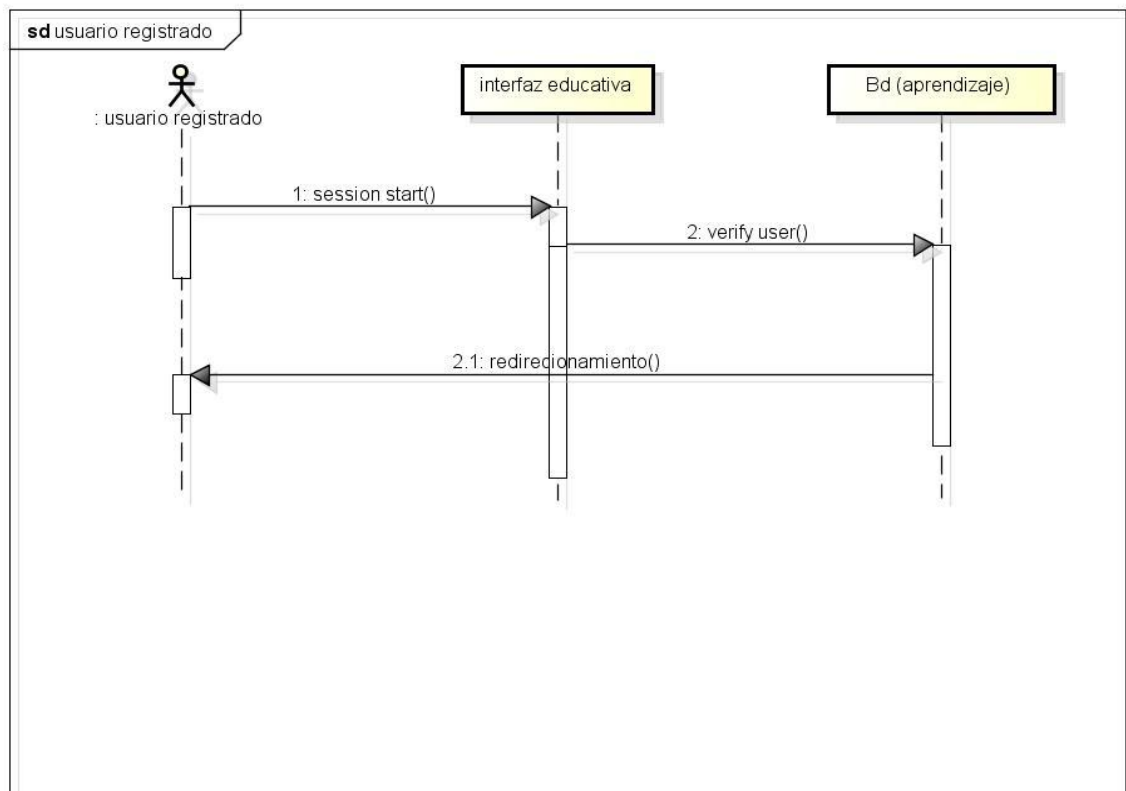


Ilustración 9: Proceso de inicio sesión de usuarios registrados en el sistema actual.

## 6.2.2 Alumnos.

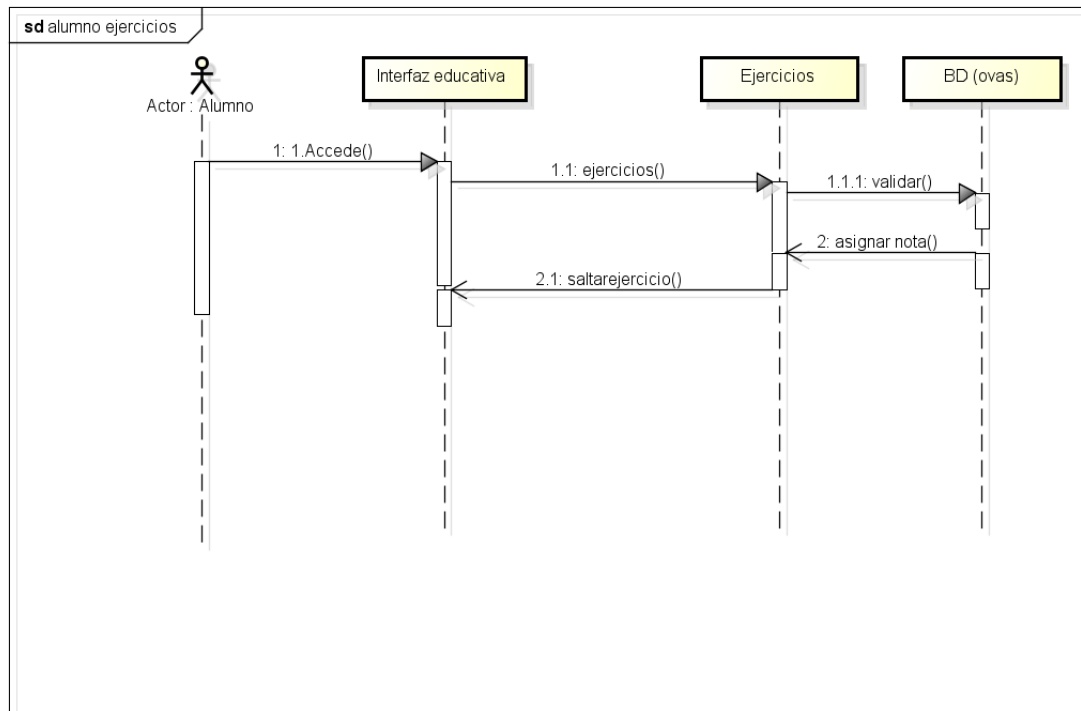


Ilustración 10: Proceso de evaluación de ejercicios sistema actual.

## 6.3 DIAGRAMA ENTIDAD RELACIÓN.

[\(Ver anexo I\).](#)

## 6.4 DIAGRAMA DE CLASES.

[\(Ver anexos II\).](#)

## 7. SISTEMA PROPUESTO

Se presenta la metodología, en la cual se definen las actividades, se organizan cronológicamente de acuerdo a sus requerimientos de precedencia y aprovechamiento de recursos. Como resultado se obtiene unas actividades que reflejan lo que se está haciendo en el proyecto.

### 7.1 DIAGRAMAS DE CASOS DE USO

Utilizaremos los diagramas de casos de uso ya que se ajusta a las necesidades de desarrollo del estudio, está tiene como finalidad principal la planificación, el seguimiento y control de las actividades y de los recursos humanos y materiales que intervienen en el desarrollo del proyecto. Como consecuencia de este control es posible conocer en todo momento qué problemas se producen y resolverlos o paliarlos de manera inmediata.

#### 7.1.1 Actores.

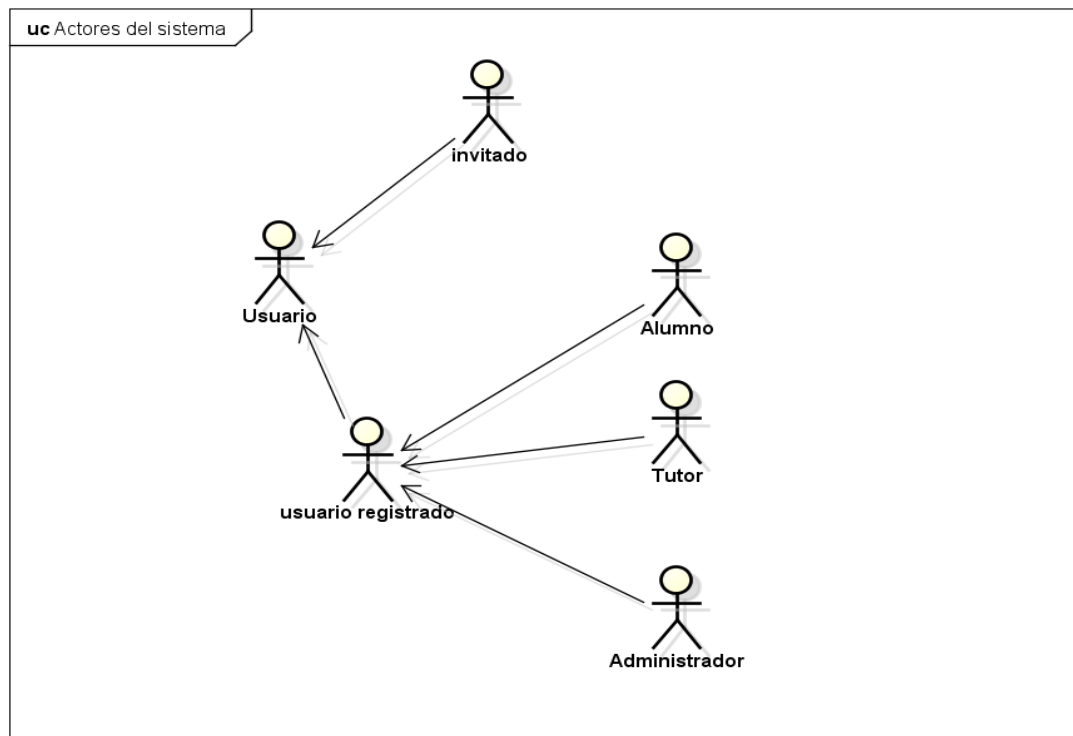


Ilustración 11: Actores del sistema propuesto.

## 7.1.2 Usuarios registrados

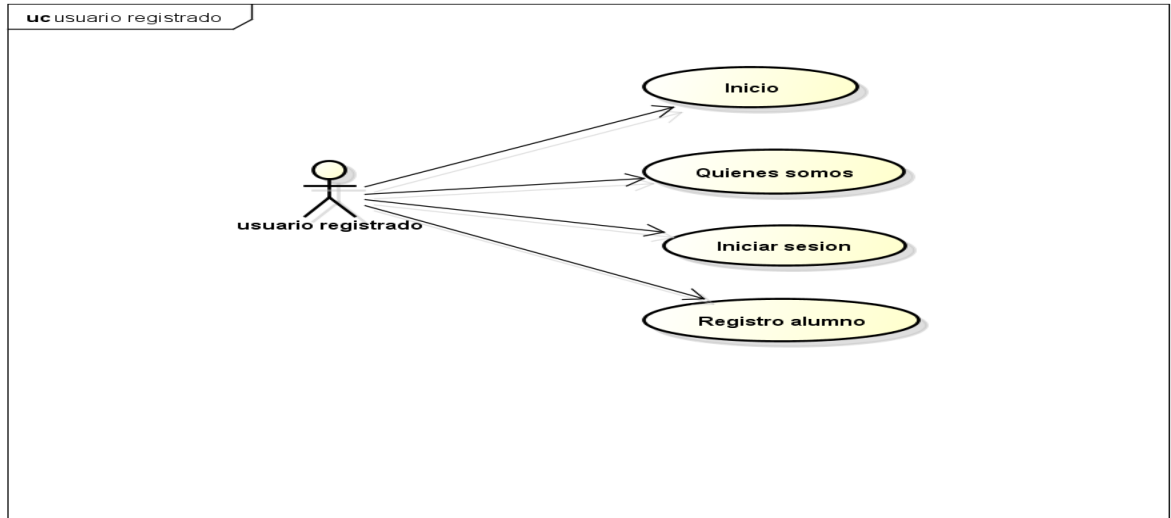


Ilustración 12: Funciones del usuario registrado en el sistema propuesto.

## 7.1.3 Administrador.



Ilustración 13: Alumno acciones sistema propuesto.

### 7.1.4 Alumno

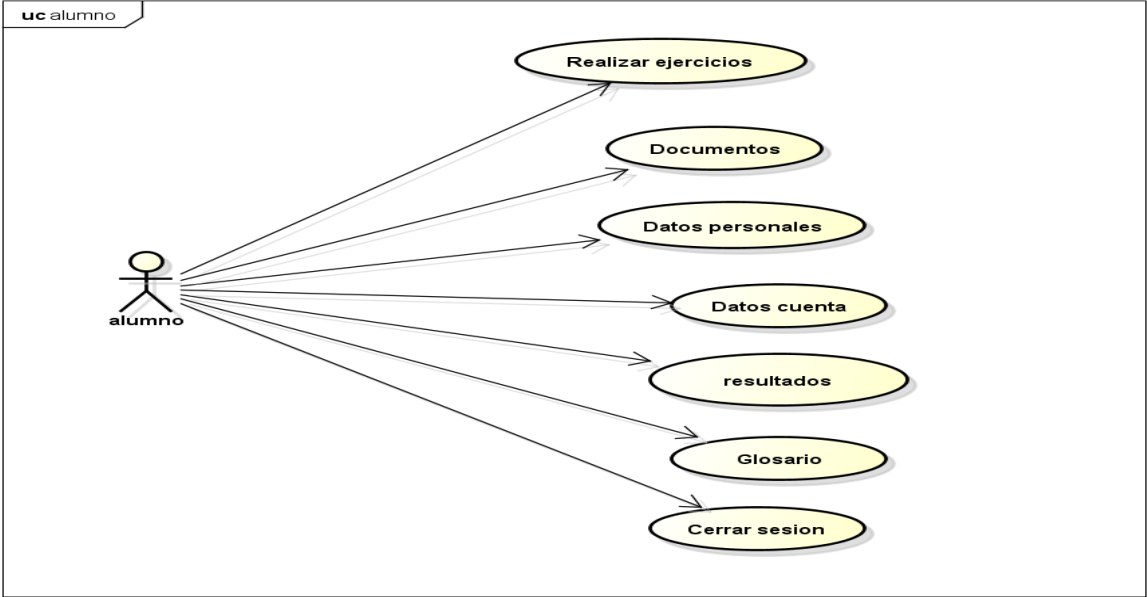


Ilustración 14: Funciones del usuario alumno en el sistema propuesto.

### 7.1.5 Tutor.

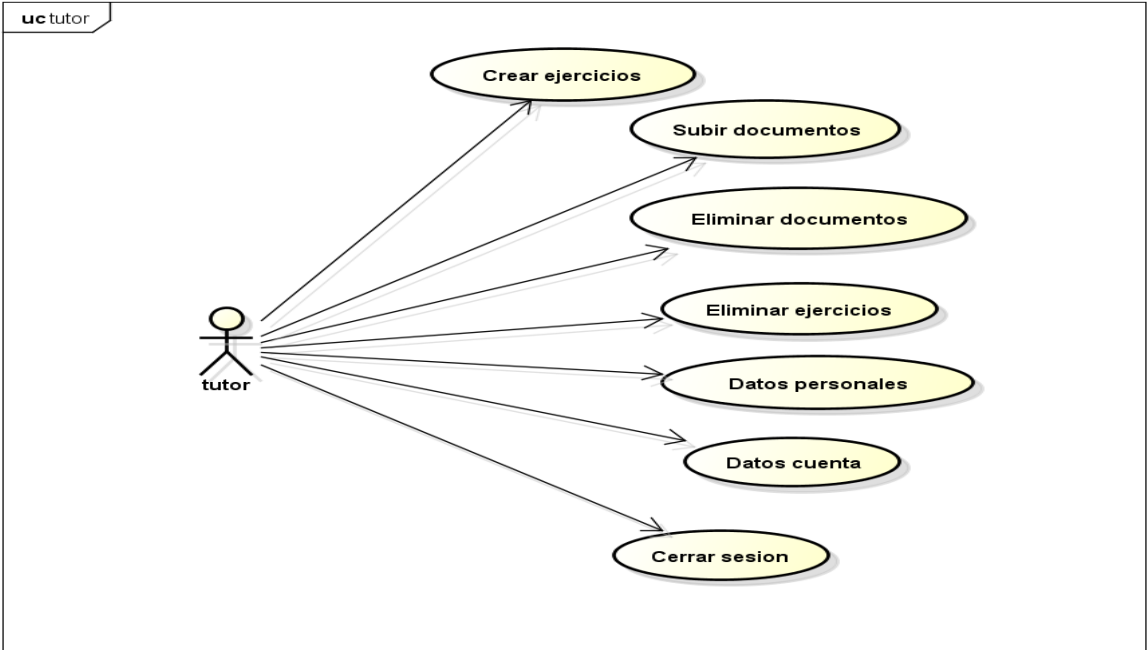


Ilustración 15: Funciones del usuario tutor en el sistema propuesto.



## 7.2 DIAGRAMA DE SECUENCIA.

En esta etapa se deben llevar a cabo las actividades de manera secuencial a lo largo de todo el proyecto, lo que permitirá dejar muy claro lo que se quiere lograr con el proyecto. Muestra la interacción de los objetos en una aplicación a través del tiempo y se modela para cada caso de uso. Contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes intercambiados entre los objetos.

### 7.2.1 Usuario registrados.

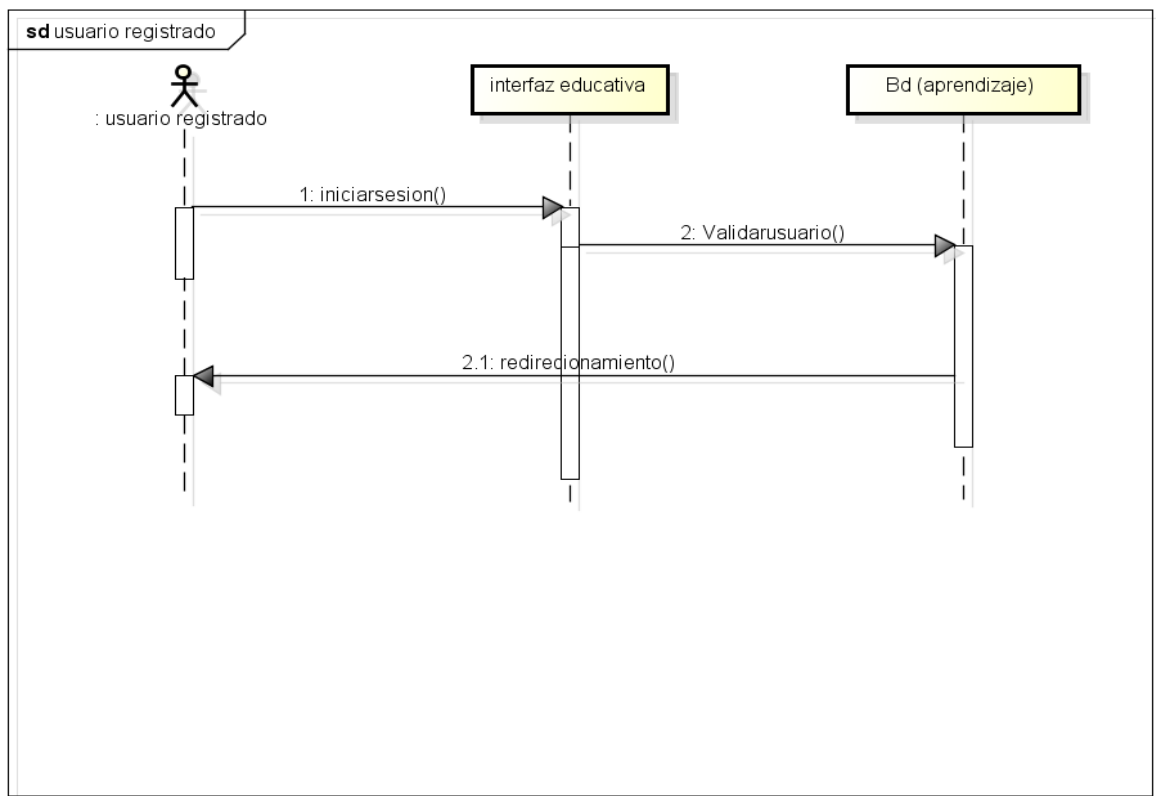


Ilustración 16: Proceso de inicio sesión de usuarios registrados en el sistema propuesto.

**7.2.2 Administrador – auditoria.**

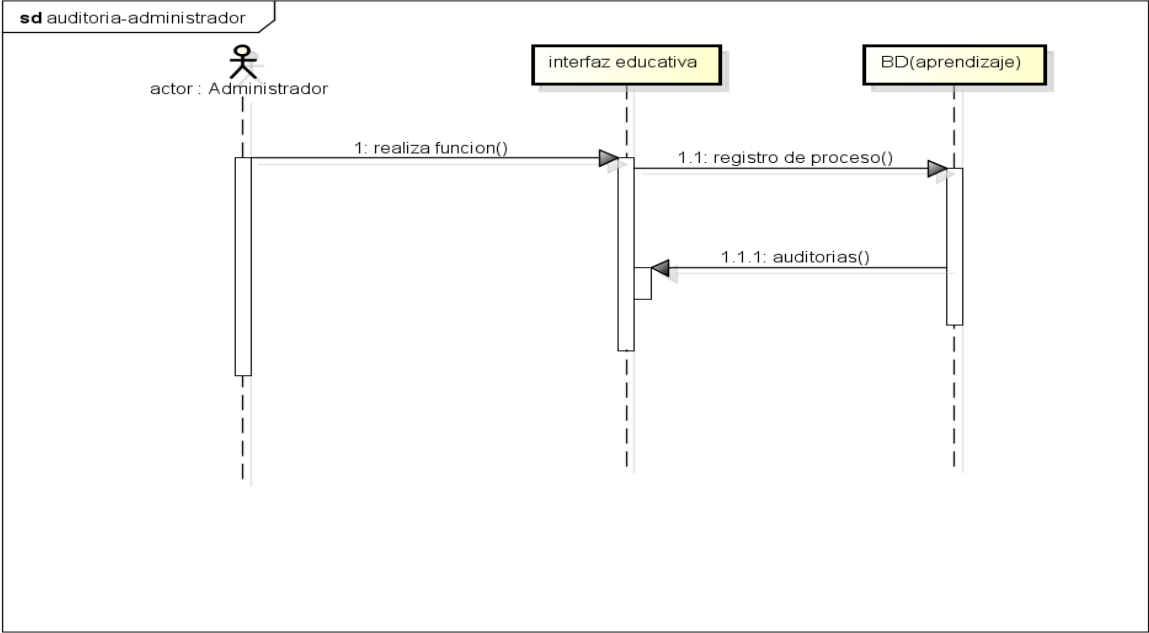


Ilustración 17: Alumno ejercicios sistema propuesto.

**7.2.3 Administrador – recuperación.**

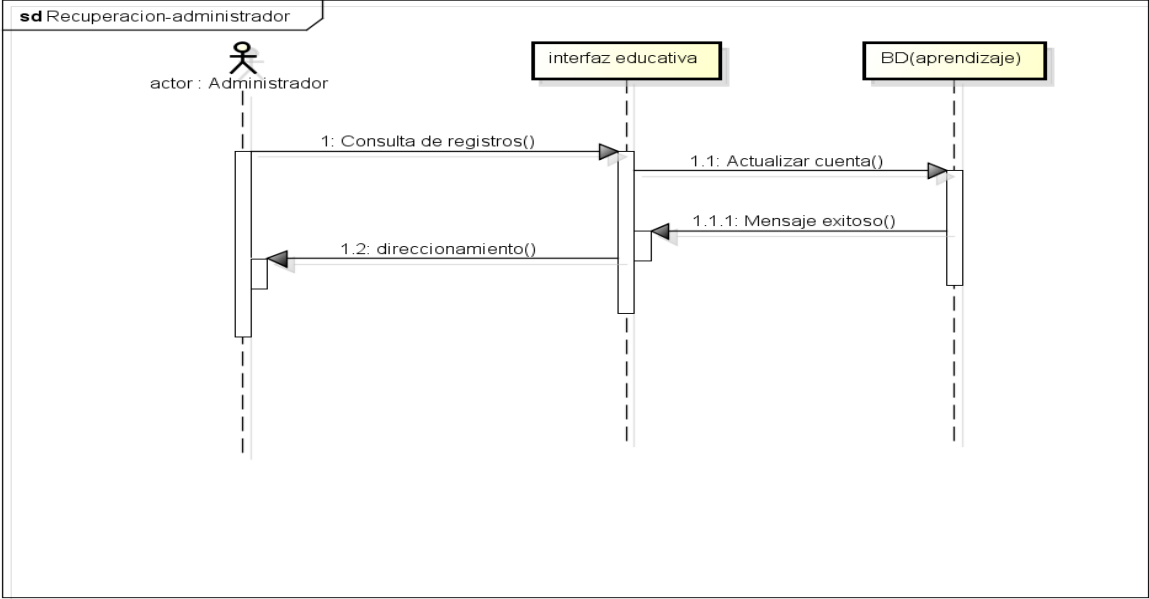


Ilustración 18: Proceso de recuperación de cuenta de usuarios en el sistema propuesto.

## 7.2.4 Administrador – autorización.

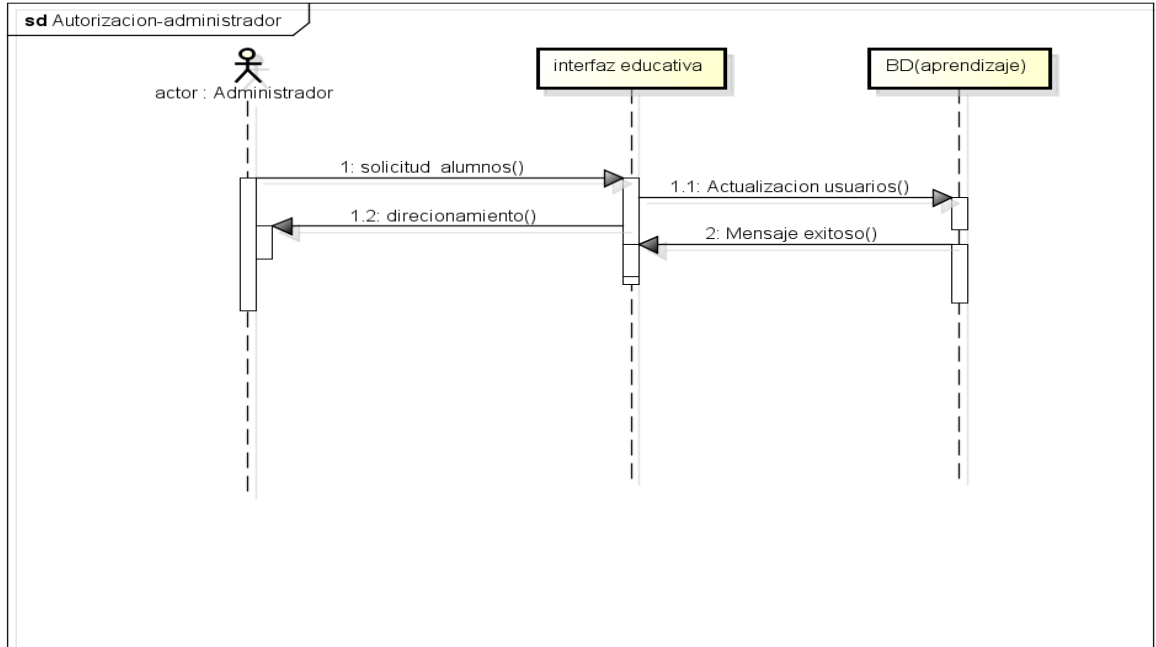


Ilustración 19: Proceso de autorización de alumnos en el sistema propuesto.

## 7.2.5 Alumno – ejercicios.

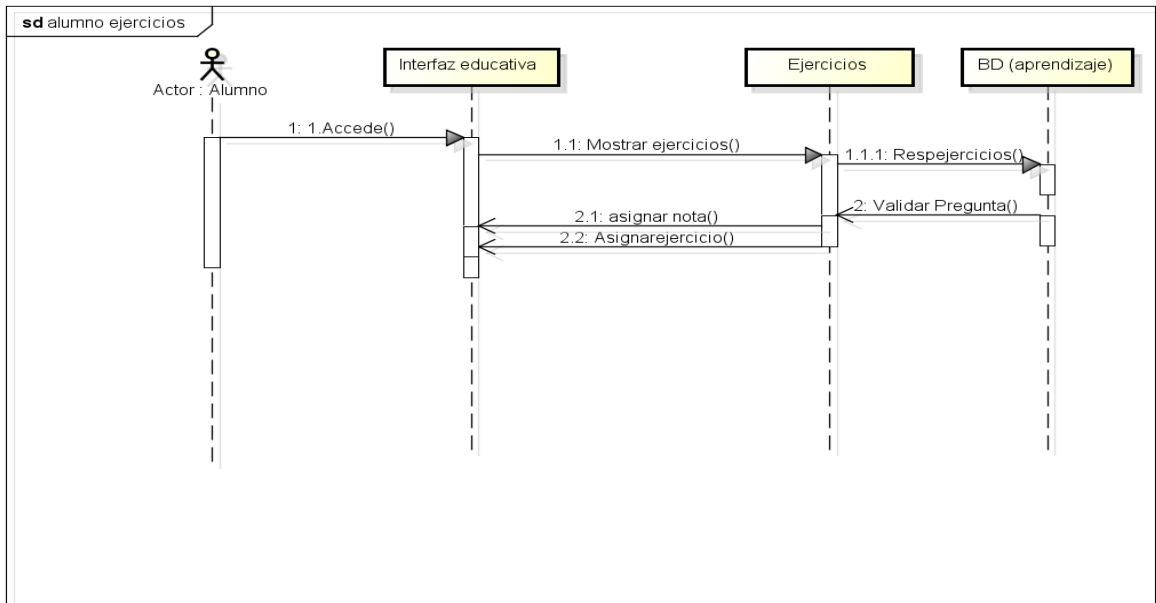


Ilustración 20: Proceso de alumno con ejercicios en el sistema propuesto.

## 7.2.6 Alumno – datos personales.

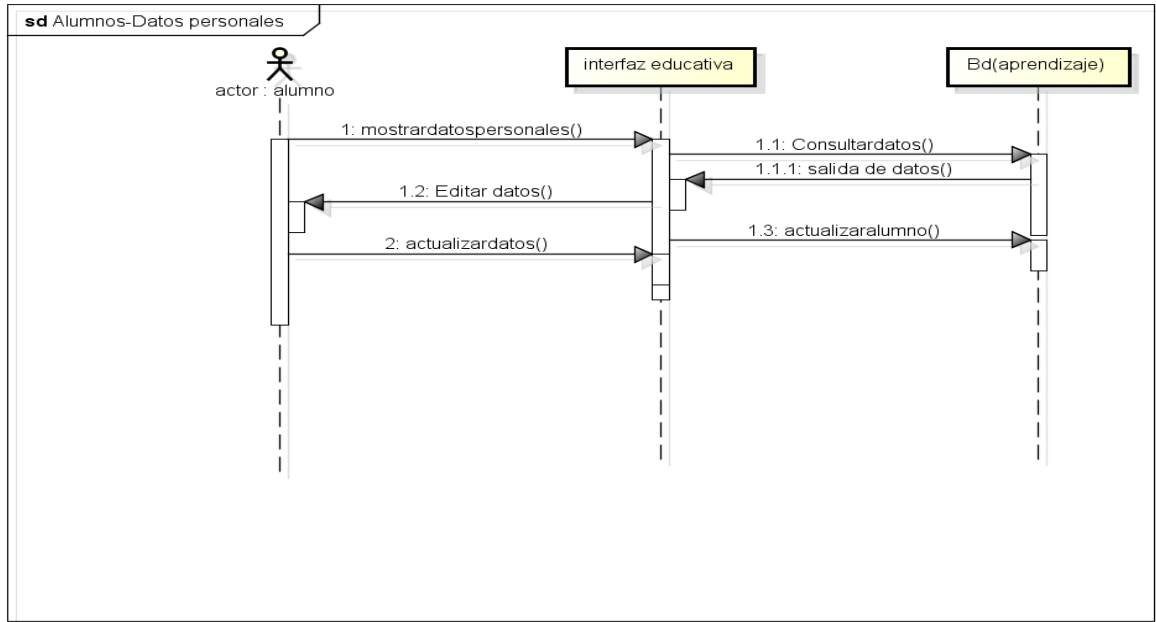


Ilustración 21: Proceso De actualización de datos personales de alumno en el sistema propuesto.

## 7.2.7 Alumno – resultados

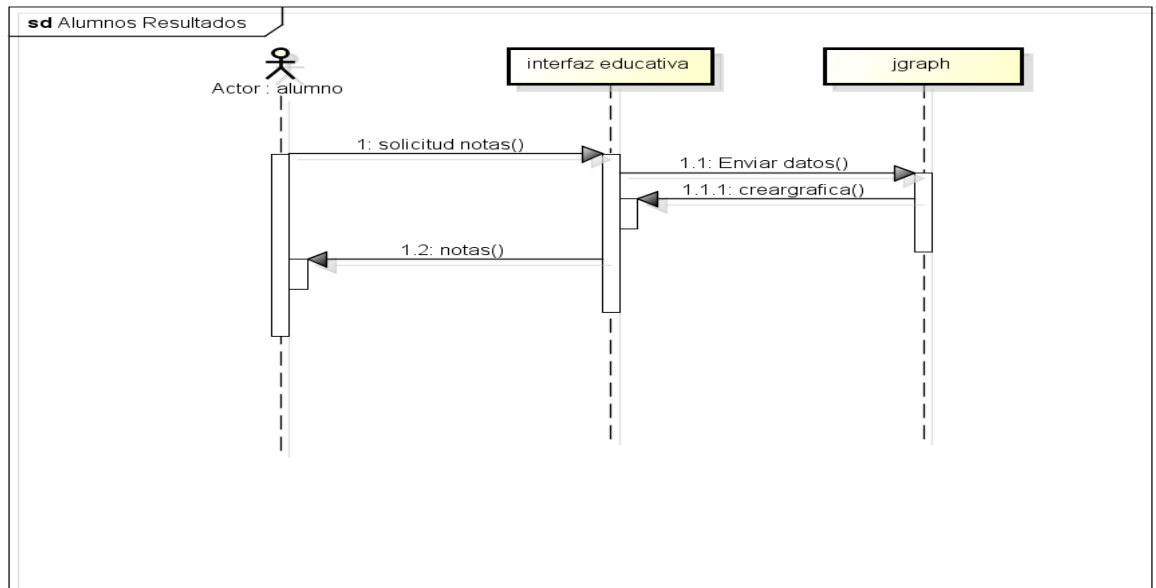


Ilustración 22: Proceso de visualización de notas del alumno en el sistema propuesto.

## 7.2.8 Tutor – ejercicios.

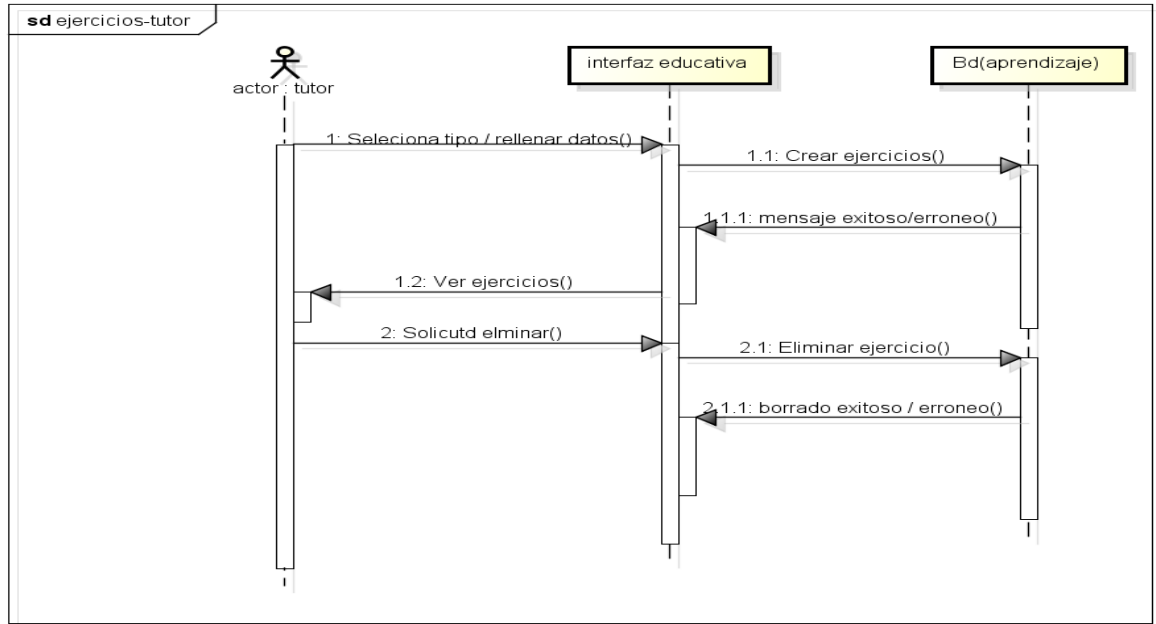


Ilustración 23: Proceso de creación y eliminación de ejercicios del tutor en el sistema propuesto.

## 7.2.9 Tutor – datos personales.

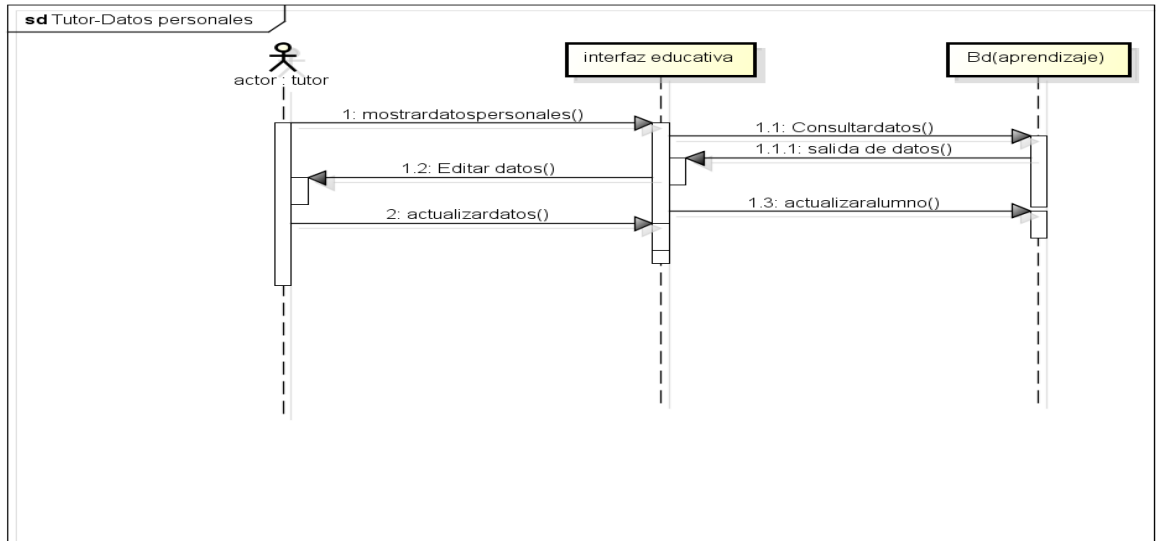


Ilustración 24: Proceso De actualización de datos personales de tutor en el sistema propuesto.

### **7.3 DIAGRAMA DE ENTIDAD RELACION**

Para este punto tendremos en cuenta los diagramas de clases que describen la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Son utilizados durante el proceso de análisis y diseño de sistemas, donde se crea un diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.

[\(Ver anexos III\).](#)

### **7.4 DIAGRAMA DE CLASES**

Este es el modelo entidad relación del software propuesto con los procedimientos y acciones que se describen a continuación.

[\(Ver anexos IV\).](#)

## **8. CRONOGRAMA DE ACTIVIDADES**

En esta etapa se definen las actividades, se organizan cronológicamente de acuerdo a sus requerimientos de precedencia y aprovechamiento de recursos. Como resultado se obtiene un Cronograma de actividades que refleja lo que se va a hacer en el tiempo y con los recursos solicitados para lograr la determinación del proyecto.

### **8.1 FASE 1 ANÁLISIS Y PLANEACIÓN.**

Realizaremos recolección de información para determinar los posibles problemas existentes, amenazas y daños del sistema al interactuar los usuarios.

### **8.2 FASE 2 IMPLEMENTACIÓN DE OBJETIVOS.**

Estableceremos los requerimientos posibles en el sistema como la arquitectura del computador, exploradores de Internet, sistema gestor de base de datos y lenguaje de programación compatibles a utilizar.

### **8.3 FASE 3 EJECUCIÓN DE PROCESOS.**

Desde el modificaremos que proceso debe seguir y cual proceso se puede mejorar tanto para los estudiantes y tutores.

### **8.4 FASE 4 CIERRE.**

Se ejecutara desde el servidor apache (**AppServer**), probando los módulos realizados en DreamWeaver con formato **PHP** orientado a objetos.

[\(Ver anexos V\).](#)

## 9. ARQUITECTURA DE LA SOLUCIÓN DEL SOFTWARE

### 9.1 MODELO

Los módulos de Ejercicios, Usuarios, Resultados, Auditorias y Sesiones para la administración del sistema, están encadenados por medio del proyecto llamado **Aprendizaje web** que estará definido por el patrón MVC (Modelo Vista Controlador). En esta carpeta se encuentran:

- La conexión a la base de datos **aprendizaje** mediante el script “**conexion.php**”.
- Están todos los scripts que afectaran la base de datos directamente, sentencias SQL como Insert, Update y las clases como pdf.
- Se encuentran todos los formularios y los scripts de consultas de cada módulo, los divisiones, las imágenes y los menús hechos en hojas de estilo css, los cuales se encuentran en carpetas separadas, para que puedan ser editadas cuando así se requiera.
- Para el proceso del software se determina en los usuarios que accederán al sistema, dentro de los usuarios tenemos: administrador, tutor, estudiante, y el usuario no registrado.

En el usuario **administrador** contiene:

- **Inicio:** Donde podrá ver el contenido inicial de la página, nuestros patrocinadores, y el objetivo con el que se hizo el software.
- **Auditorias:** Dentro de esta área, el administrador podrá ver los procesos que los usuarios, harán en cada inserción modificación o eliminación ya sea de datos archivos, o usuarios eliminados.
- **Consulta registros:** El administrador consulta cuantos usuarios registrados hay actualmente en la aplicación. También examina el tipo de usuario que se encuentra registros, si es un profesor o un estudiante, como sea el caso, él le asignara el valor de cuenta y revisará si el usuario realmente es el que está manejando.
- **Recuperación:** En esta opción el administrador recupera el usuario y lo modifica en ingreso por la contraseña o el tipo de pregunta que le



asigna no el usuario.

- **Autorización:** El administrador desde este punto toma el control de los usuarios para autorizar o desautorizar el acceso al sistema. Esto hace que el software sea seguro y no cualquiera pueda crear una cuenta y posicionarse en el lugar correcto.
- **Eliminar usuarios:** En este punto el administrador elimina los usuarios que estén inactivos o no hayan hecho un registro bien elaborado, los que ya hayan elaborado el proceso de evaluación pasado el tiempo del mismo. Con esto se libera espacio en la bases de datos.
- **Registro Tutores:** Desde esta opción. El administrador registra los maestros dentro un formulario diseñado para el registro de este tipo de usuarios con toda su información personal.
- **Cerrar sesión:** Sale del sistema y cierra la cuenta de administrador.

En el usuario **tutor** contiene:

- **Inicio:** Donde podrá ver el contenido inicial de la página, nuestros patrocinadores, y el objetivo con el que se hizo el software.
- **Crear ejercicios:** El tutor puede subir y crear los ejercicios para los estudiantes, cada ejercicio lleva un tema en específico y se sube para que el estudiante realice los respectivos ejercicios.
- **Subir documentos:** El tutor puede subir los documentos que estén en extensión pdf para que el alumno vea como desarrollar estos ejercicios subidos al sistema, además de eso tendrá que escoger el tema específico que va tratar este documento y dejarlo como apoyo para el estudiante.
- **Eliminar documentos:** En esta sección el tutor elimina los documentos que haya subido por error o mal y también los ejercicios que ya fueron realizados.
- **Eliminación ejercicios:** El tutor desde este punto podrá eliminar los ejercicios realizados o los que haya subido mal o por error.

- **Datos personales:** El tutor podrá modificar sus datos personales y actualizarlos cuando sea necesario.
- **Datos cuenta:** El usuario podrá modificar su contraseña y la pregunta de seguridad para actualizarlo periódicamente.
- **Cerrar sesión:** Sale del sistema y cierra la cuenta de tutor.

En el usuario **tutor** contiene:

- **Inicio:** Donde podrá ver el contenido inicial de la página, nuestros patrocinadores, y el objetivo con el que se hizo el software.
- **Ejercicios:** El alumno puede ingresar y desarrollar los ejercicios que se dejaron para el tema propuesto con el documento propuesto para ese ejercicio.
- **Documentos:** En esta opción el alumno podrá ver los documentos que se dejaron para los temas y ejercicios que debe desarrollar en la aplicación, esto le servirá de soporte para el proceso que debe seguir el alumno en el programa.
- **Datos personales:** El alumno podrá modificar sus datos personales y actualizarlos cuando sea necesario.
- **Datos cuenta:** El usuario podrá modificar su contraseña y la pregunta de seguridad para actualizarlo periódicamente.
- **Resultados:** El alumno en este punto puede ver el resultado de las pruebas realizadas en un promedio general por cada tema resuelto y un promedio global para todos los temas en general. Esto le dará una calificación promedio si obtuvo capacidades en el desarrollo de las competencias.
- **Glosario:** El alumno podrá visualizar los tipos de variables, ciclos y las estructuras de ellos con una caja de texto entrelazada con el link especificando un ejemplo de declaración.
- **Cerrar sesión:** Sale del sistema y cierra la cuenta de alumno.

## 10. ANÁLISIS Y DISEÑO DE LA SOLUCIÓN DEL SOFTWARE

### 10.1 ESTRUCTURA DE ALMACENAMIENTO

Dentro de esta estructura se verá como los procesos de ingreso de registro y de desarrollo de las competencias aplicadas al software se realizan con sus respectivos usuarios y categorías donde mostraran que los datos están almacenados

Registro De tutores

**Datos Personales**

Primer Nombre  Segundo nombre

Primer apellido  Segundo apellido

Genero  Direccion

Telefono (fijo)  T. Celular

**Datos De la Cuenta**

Nombre Usuario

Contraseña  Repetir Contraseña

PreguntasDe Seguridad

Respuesta

Limpiar

Ilustración 25: Formulario registro tutor.

### 10.2 INTERFAZ GRAFICA DE USUARIO

Dentro de esta estructura se verá como los procesos de ingreso de registro y de desarrollo de las competencias aplicadas al software se realizan con sus respectivos usuarios y categorías donde mostraran que los datos están almacenados.

### 10.2.1 Ventajas.

El diseño se basó en los colores institucionales los cuales fueron 3 (blanco, gris, azul) para que el usuario no se canse y se sienta desgastado al usar la aplicación. Con este diseño se adquieren los conocimientos más rápido y entiende su uso de manera rápida y sencilla.

El usuario puede retener más conocimientos sin temor a perder la información adquirida al utilizar la interfaz.



Ilustración 26: Página de inicio.

### 10.3 INTERFAZ DE PROCESAMIENTO DE DATOS

Dentro de esta estructura se verá como los procesos del software que se realizan con los usuarios y categorías donde mostraran que los datos están almacenados

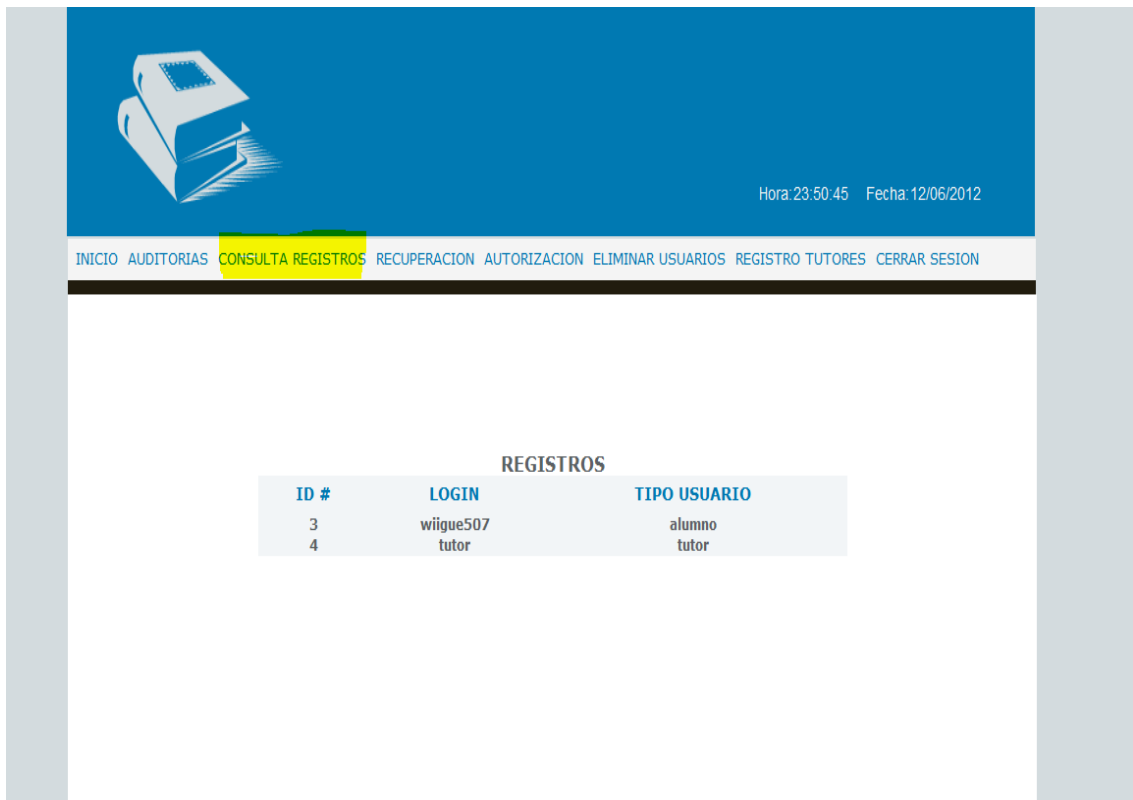


Ilustración 27: Consulta de registros.

## **11. DESARROLLO Y PRUEBAS**

En este punto se quiere ver los resultados de las pruebas desarrolladas a la aplicación a continuación se describirán las pruebas realizadas y el desarrollo llevado a cabo.

Se hicieron 4 pruebas en fechas definidas para ver la funcionalidad del software:

### **11.1 PRUEBA 1:**

Abril 23 de 2012, en este punto se probó la funcionalidad de la interfaz y la base de datos y las sesiones de usuarios. Se comprobaron errores en el diseño por los colores que se estaban manejando en esa época, se registró que las sesiones tenían errores en la ejecución del acceso al usuario ya que no registra los ingresos debidamente y la seguridad, se revisaron los error arrojados en las bases de datos por medio de los diagramas y se tuvieron en cuenta las conexiones y registros. Se mejoraron a raíz de esta prueba el acceso de las sesiones de usuarios se mejora la base de datos y los colores de la interfaz como los diseños.

### **11.2 PRUEBA 2:**

Mayo 10 de 2012, se le agregan las siguientes características, subir ejercicios, crear ejercicios, como su modificación y eliminación así respectivamente en cada acción para las sesiones.

Se comprobaron los errores en la ejecución para la mejora en esta fase y se corrigieron, lo cual deja el sistema con las características agregadas ya implementadas

### **11.3 PRUEBA 3:**

Mayo 10 de 2012, solo se arreglaron los detalles faltantes por arreglar para mejorar el sistema de y la aplicación en la ejecución se modificaron partes de las cuales presentaron errores y se corrigieron al punto de lo que se espera revisar en la otra fase, se le agregaron los menús las gráficas y no presentaron problemas durante la ejecución.

#### **11.4 PRUEBA 4:**

Mayo 10 de 2012, en este punto se revisó que el software esté funcionando correctamente y haciendo las conexiones bien en los registros de las bases de datos y manejo y reconocimiento de los resultados.

En este punto se da por terminado el software ya que las pruebas realizadas mejoraron el óptimo funcionamiento, se da por finalizado el ciclo de pruebas realizadas al software.

[\(Ver anexos IX\).](#)

## 12. CONCLUSIONES

- La idea desde el inicio es dejar un software para la universidad como apoyo de los estudiantes que desean tener más conocimiento en la programación.
- Los objetos virtuales de aprendizaje son un medio de apoyo para los estudiantes, es una de las mejores formas de estrategia de enseñanza.
- Al implemento de php orientado a objetos, es una de las nuevas formas de crear software más robusto para un mayor uso.
- Al tener software asistidos por computador puede mejorar las enseñanza, teniendo en cuenta que los integrantes de las carrera de informática y sistemas debe ser primordial el uso del computador para su conocimiento de la programación.



### 13. RECOMENDACIONES

- Recomendamos no hacer actualizaciones del software utilizados para desarrollar este software. De hacerlo probablemente presente problemas, de lo contrario omita esa recomendación.
- Este software debe llevar las ideas pactadas que se encuentran en este documento, como es para donde está orientado el desarrollo y para qué origen va dirigido.
- Ejecutar el software de desarrollo con las instrucciones aplicadas en el manual del programador que se encuentra dentro de la información digital de este software.
- Las personas que quieran dar continuidad a este proyecto, para esta plataforma le puedan mejorar el diseño de la interfaz, pero sin modificarle el tipo de programación que se utilizo para hacer este proyecto.
- Que se le agreguen mas temas, utilizando los que lleva: conceptos básicos, debe llevar, la herencia es para tener más conceptualización a la hora de enseñar en la plataforma toda la conceptualización de los temas que ejecuta la aplicación.
- La expectativa de este software es que pueda ser publicado en la plataforma de uniminuto como refuerzo a los problemas que se presentan en el aprendizaje de los estudiantes.
- Hacer investigaciones de como se ha mejorado la experiencia con el uso de la plataforma si es que se está usando, y si es el caso actualizarla a los estándares que se manejen.

- Las personas que tomen el reto de continuar este proyecto, que sean estudiantes que tengan los conocimientos bien definidos de los tipos de programación que se utilizaron en la plataforma.

## 14. BIBLIOGRAFÍA E INFOGRAFÍA

- Como Programar en Java, Deitel (Prentice Hall) Prentice Hall ISBN: 9702605180 Spines PDF 1268 páginas Deitel Harvey M, Deitel Paul J. 2004
- E. KENDALL, KENNETH y E. KENDALL, JULIE. Análisis y diseño de sistemas. Sexta edición. PEARSON EDUCACIÓN, México, 2005. ISBN: 970-26-0577-6. Área: Computación. Formato: 21 x 27 cm Páginas: 752.
- Metodos orientados a objetos. Segunda edición. Ian Graham. Editorial: Addison-Wesley Iberoamericana S.A .y ediciones deiaz de santos s.a.
- java para estudiantes Mayo de 2012. Se encuentra disponible. Internet: ([http://books.google.com.co/books/about/Java\\_Para\\_Estudiantes.html?id=TRUdyfwdaSoC&redir\\_esc](http://books.google.com.co/books/about/Java_Para_Estudiantes.html?id=TRUdyfwdaSoC&redir_esc) <[http://books.google.com.co/books/about/Java\\_Para\\_Estudiantes.html?id=TRUdyfwdaSoC&redir\\_esc](http://books.google.com.co/books/about/Java_Para_Estudiantes.html?id=TRUdyfwdaSoC&redir_esc)>=)
- objetos ova descripción Mayo de 2012. Se encuentra disponible. Internet: <[http://issuu.com/andresquim22/docs/presentacion\\_tics\\_objetos\\_virtuales\\_de\\_aprendizaje](http://issuu.com/andresquim22/docs/presentacion_tics_objetos_virtuales_de_aprendizaje)>
- datos para el marco legal “porque Linux es mejor” Mayo de 2012. Se encuentra disponible. Internet: ([http://www.whylinuxisbetter.net/index\\_es.php?lang=es](http://www.whylinuxisbetter.net/index_es.php?lang=es) <[http://www.whylinuxisbetter.net/index\\_es.php?lang=es](http://www.whylinuxisbetter.net/index_es.php?lang=es)>).
- Que es, formas y tipos de licenciamiento de software Mayo de 2012. [Se encuentra disponible. Internet: <<http://www.wilkinsonpc.com.co/free/articulos/softwarelibre.html>>
- Imagen del ciclo de vida modelo espiral tomada Mayo de 2012. Se encuentra disponible. Internet: <<http://commons.wikimedia.org/wiki/File:ModeloEspiral.svg?uselang=es>>
- Ciclo de vida del software modelo en espiral tomado MAYO DE 2012 [Se encuentra disponible. Internet: <[http://es.wikipedia.org/wiki/Desarrollo\\_en\\_espiral](http://es.wikipedia.org/wiki/Desarrollo_en_espiral)>

- Java conceptos Mayo de 2012. Se encuentra disponible. Internet:  
< [http://es.wikipedia.org/wiki/Java\\_\(lenguaje\\_de\\_programaci%C3%B3n\)](http://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n))>
- PHP conceptos Mayo de 2012. Se encuentra disponible. Internet:  
< <http://es.wikipedia.org/wiki/Php>>
- Objeto conceptos Mayo de 2012. Se encuentra disponible. Internet:  
<[http://es.wikipedia.org/wiki/Objetos\\_\(programaci%C3%B3n\\_orientada\\_a\\_objetos\)](http://es.wikipedia.org/wiki/Objetos_(programaci%C3%B3n_orientada_a_objetos))>
- Clase conceptos Mayo de 2012. Se encuentra disponible. Internet:  
< [http://es.wikipedia.org/wiki/Clase\\_\(inform%C3%A1tica\)](http://es.wikipedia.org/wiki/Clase_(inform%C3%A1tica)) >
- Cadena conceptos Mayo de 2012. Se encuentra disponible. Internet:  
< [http://es.wikipedia.org/wiki/Cadena\\_\(inform%C3%A1tica\)](http://es.wikipedia.org/wiki/Cadena_(inform%C3%A1tica)) >

## 15. ANEXOS

### ANEXO I. DIAGRAMA ENTIDAD RELACIÓN SISTEMA ACTUAL.

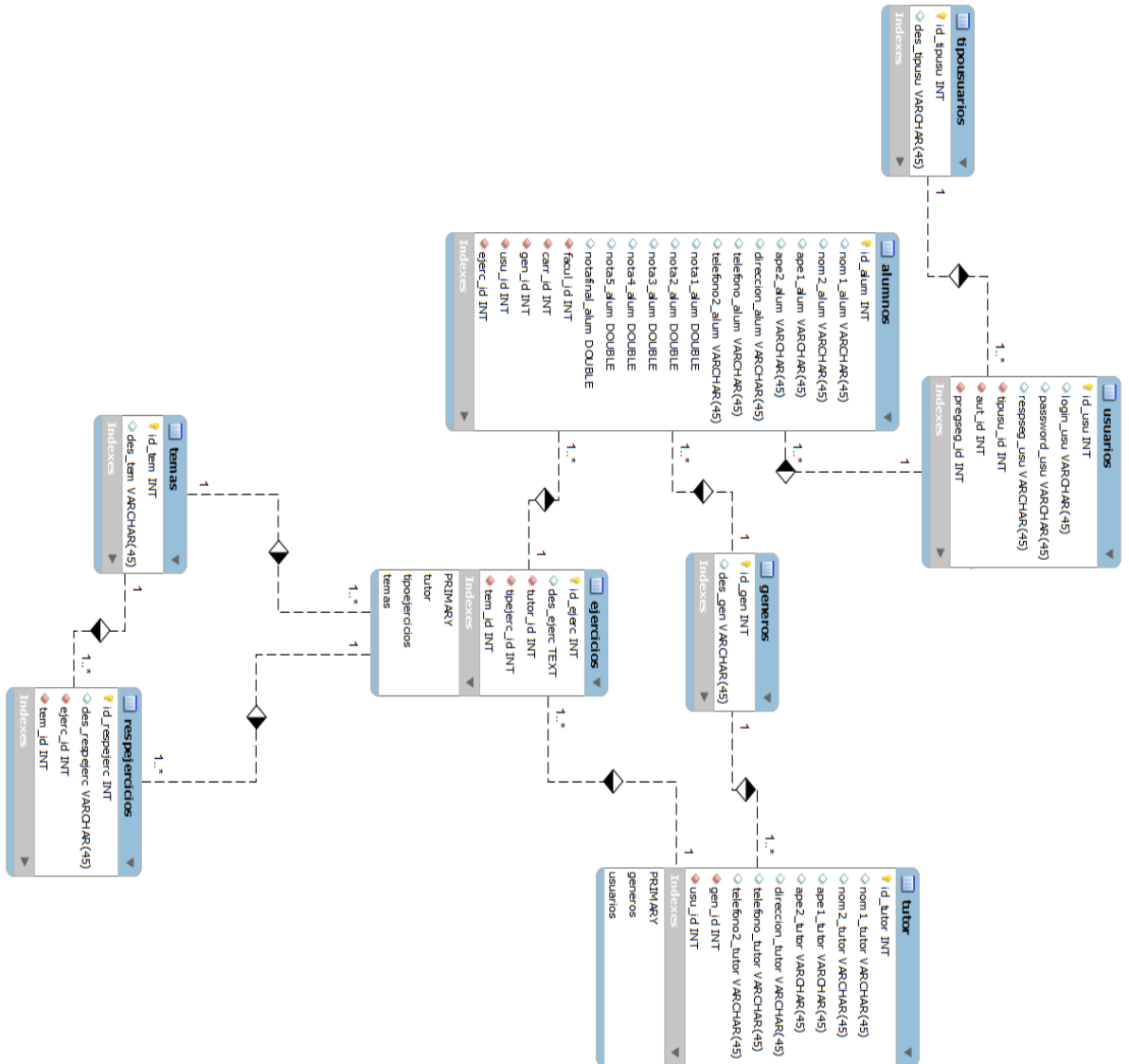


Ilustración 28: Modelo entidad relación sistema actual.

## ANEXO II. DIAGRAMA DE CLASES SISTEMA ACTUAL.

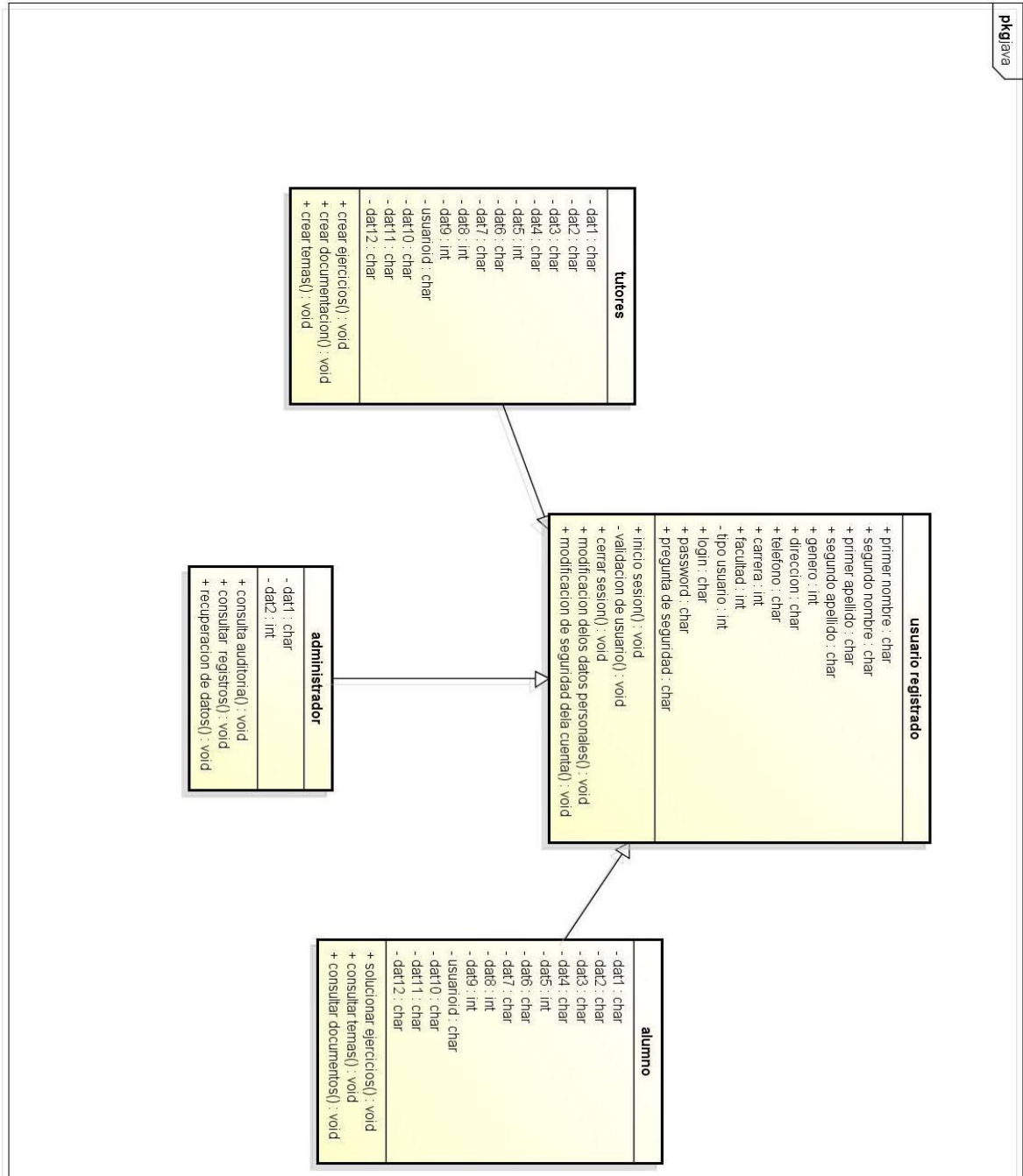


Ilustración 29: Diagrama de Clases sistema actual.

### ANEXO III. DIAGRAMA ENTIDAD RELACION SISTEMA PROPUESTO.

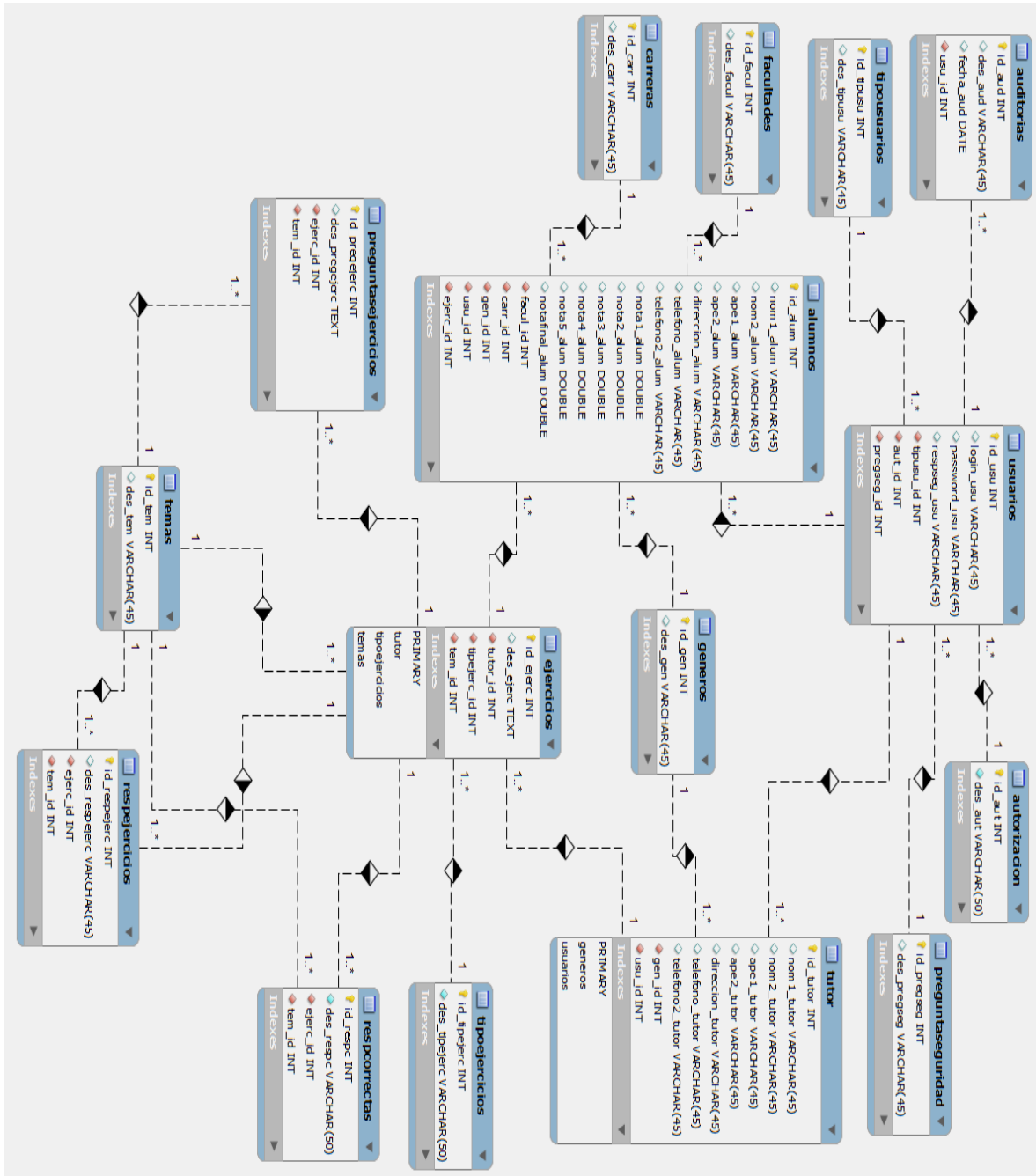


Ilustración 30: Diagrama de entidad relación sistema propuesto.

## ANEXO IV. DIAGRAMA DE CLASE SISTEMA PROPUESTO.

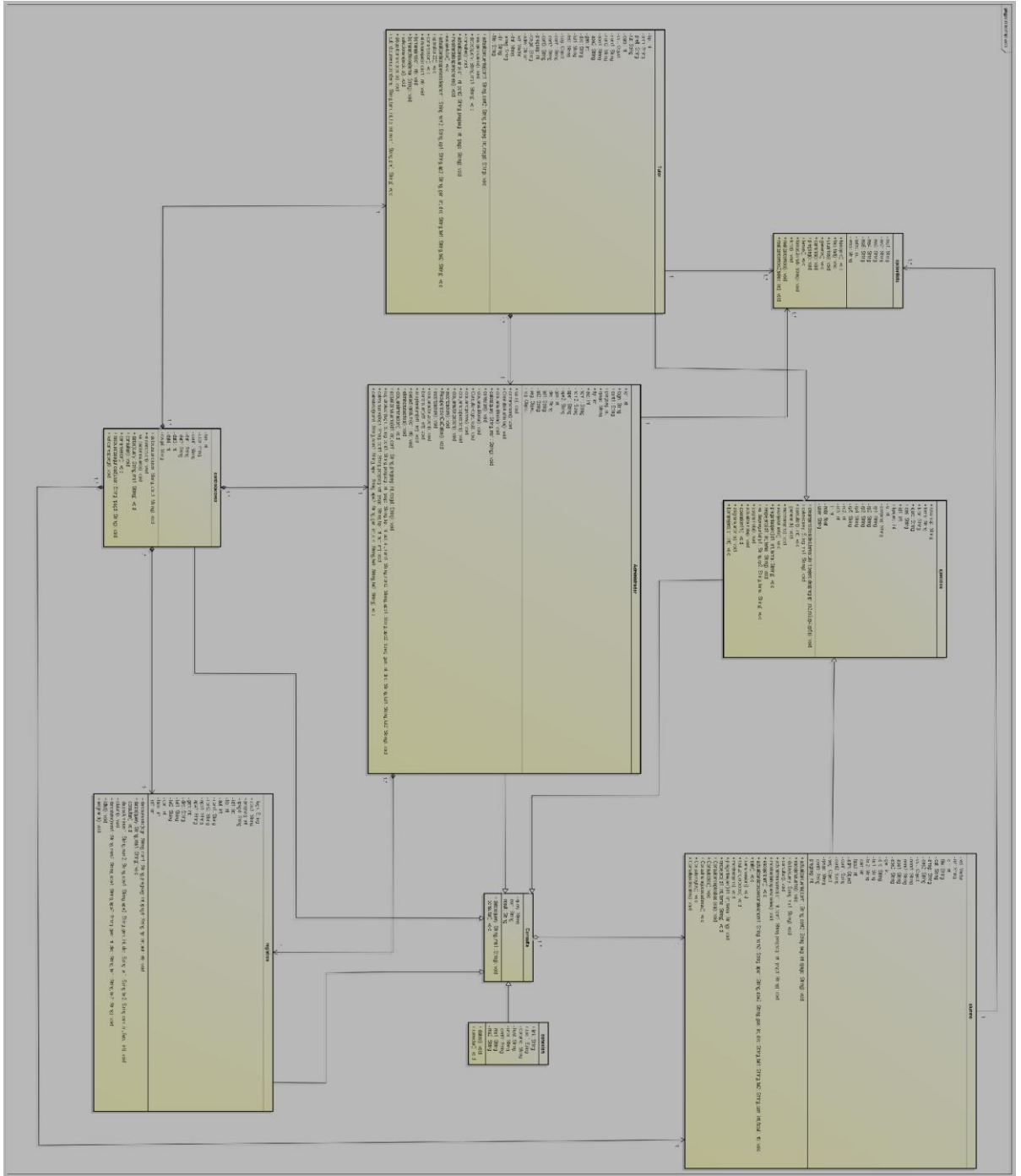


Ilustración 31: Diagrama de clases sistema propuesto.



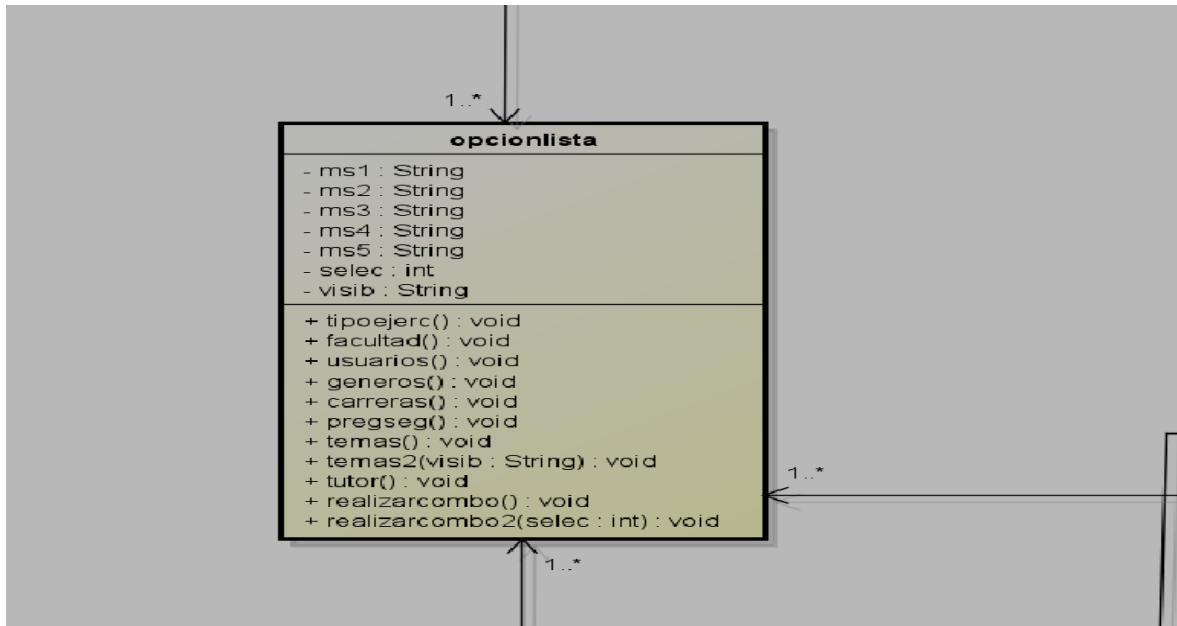


Ilustración 32: Clase opción lista.

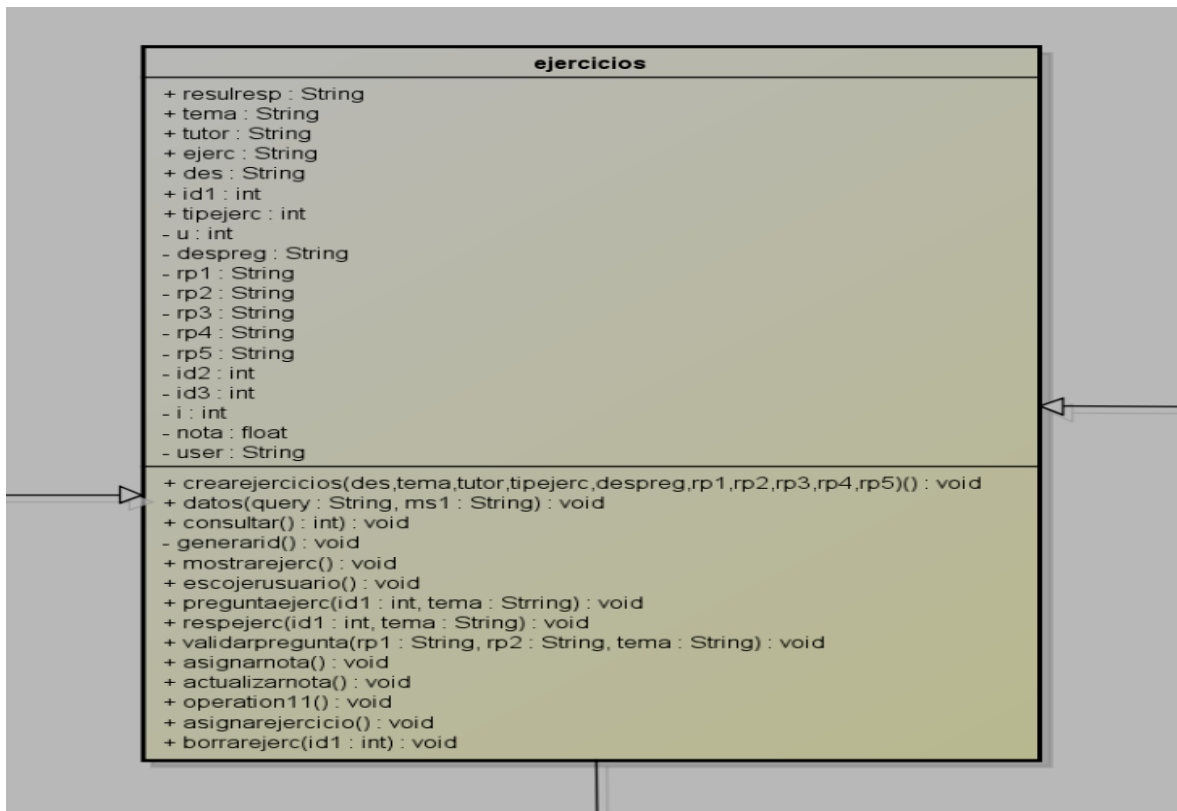


Ilustración 33: Clase ejercicios.

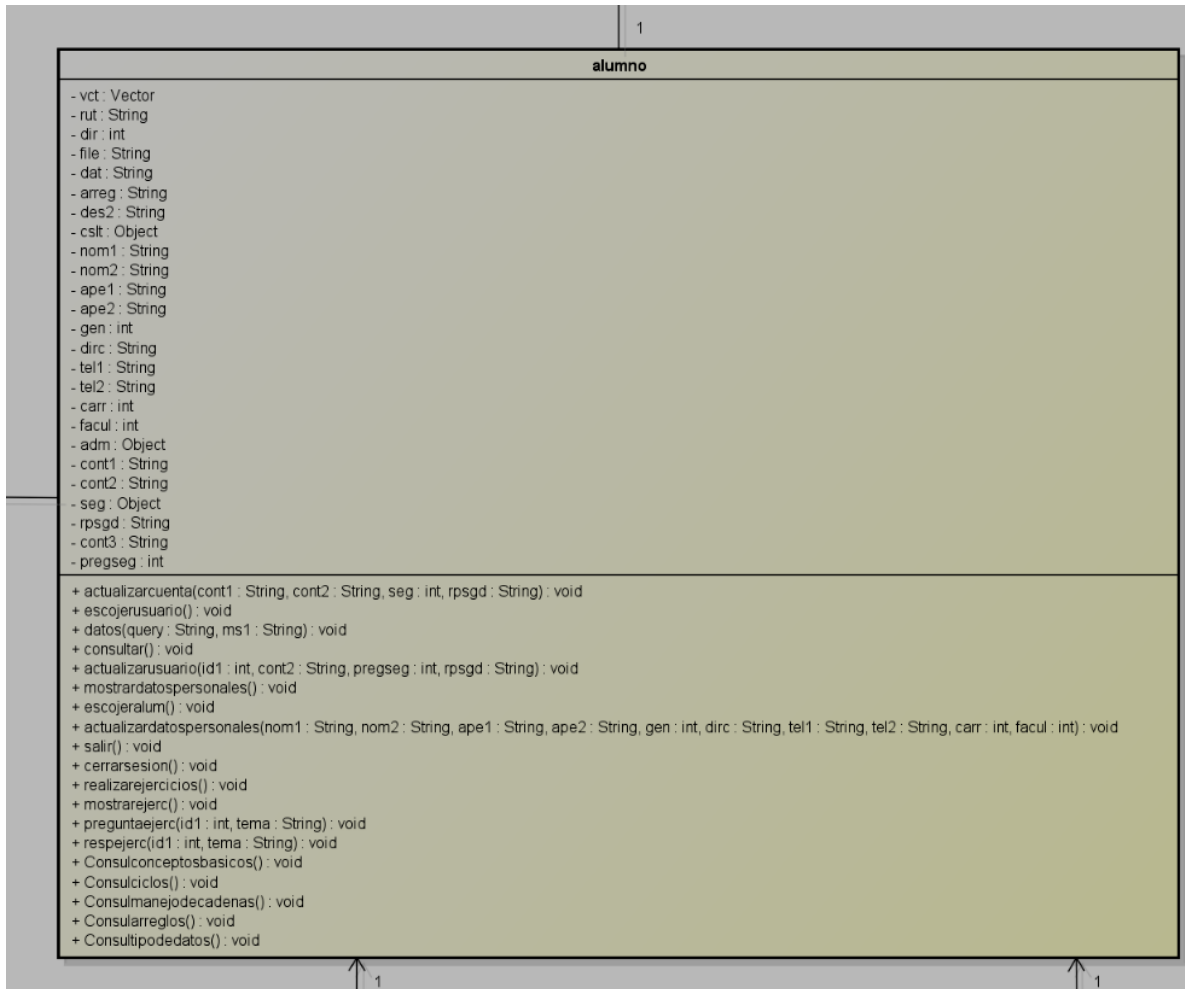


Ilustración 34: Clase alumno.

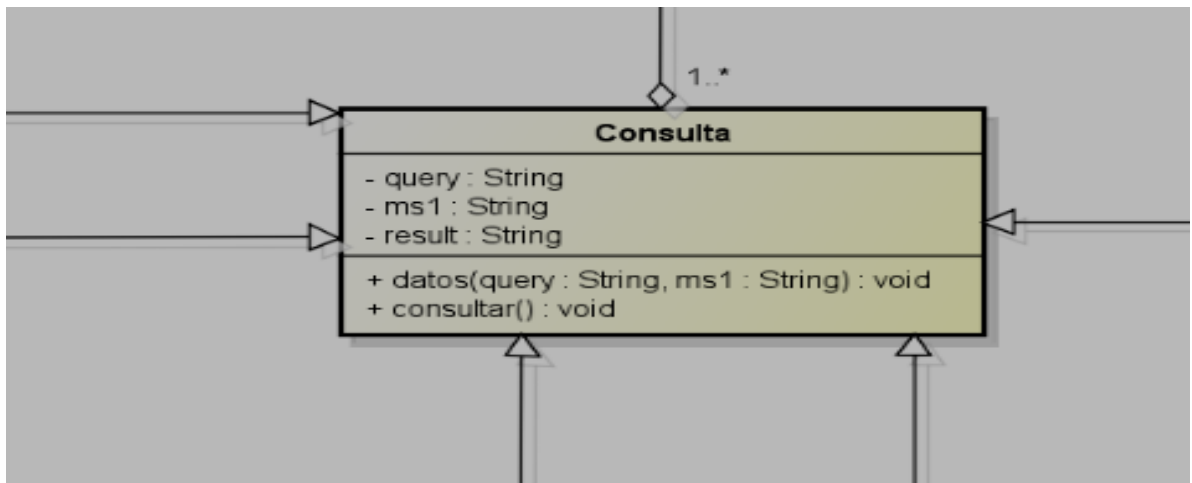


Ilustración 35: Clase consulta.

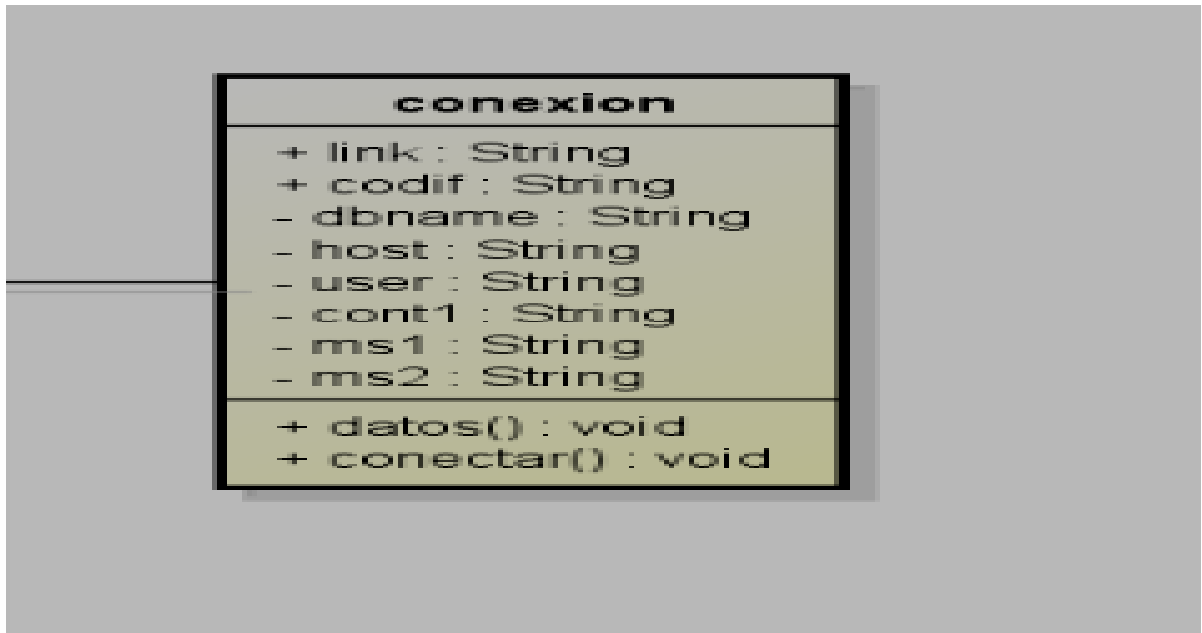


Ilustración 36: Clase conexión.

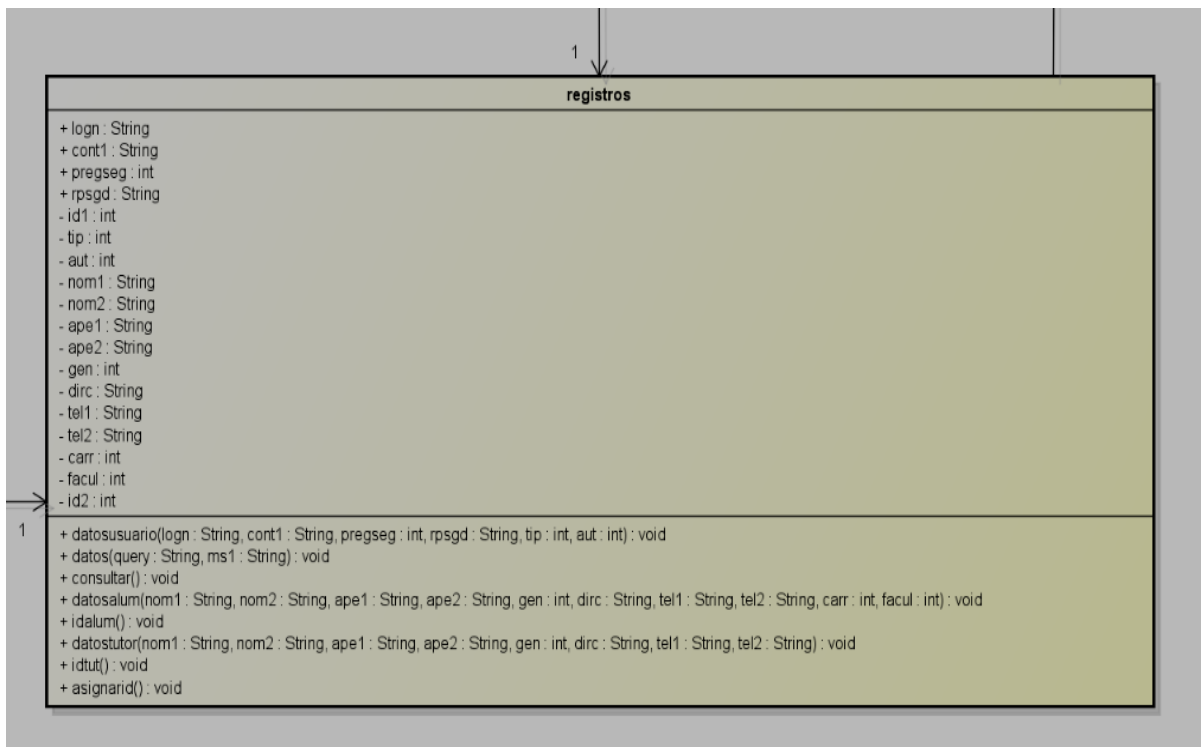


Ilustración 37: Clase registros.

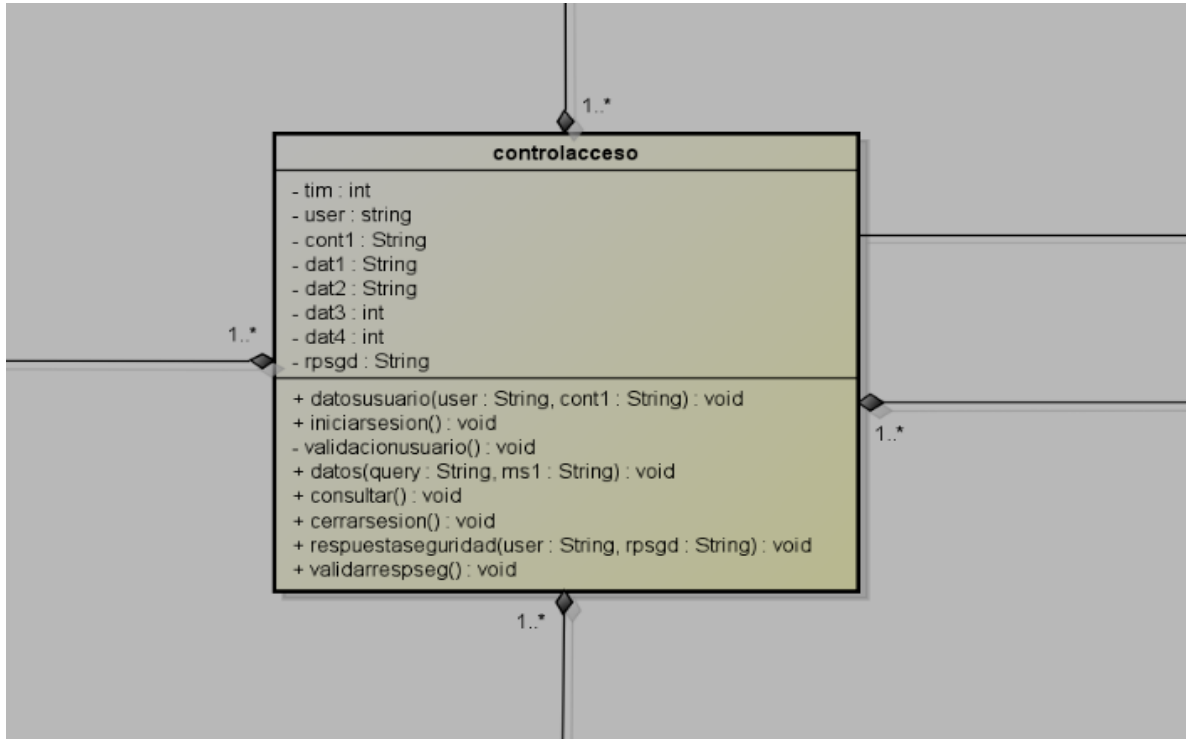


Ilustración 38: Clase controlacceso.

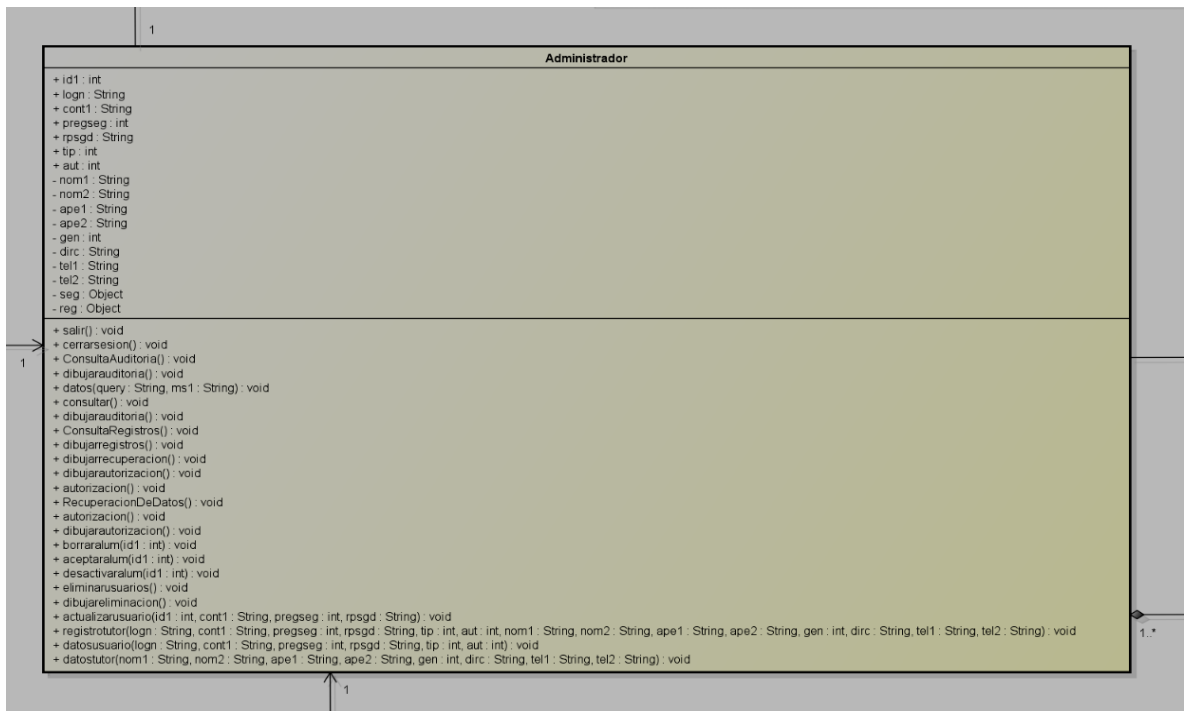


Ilustración 39: Clase Administrador.

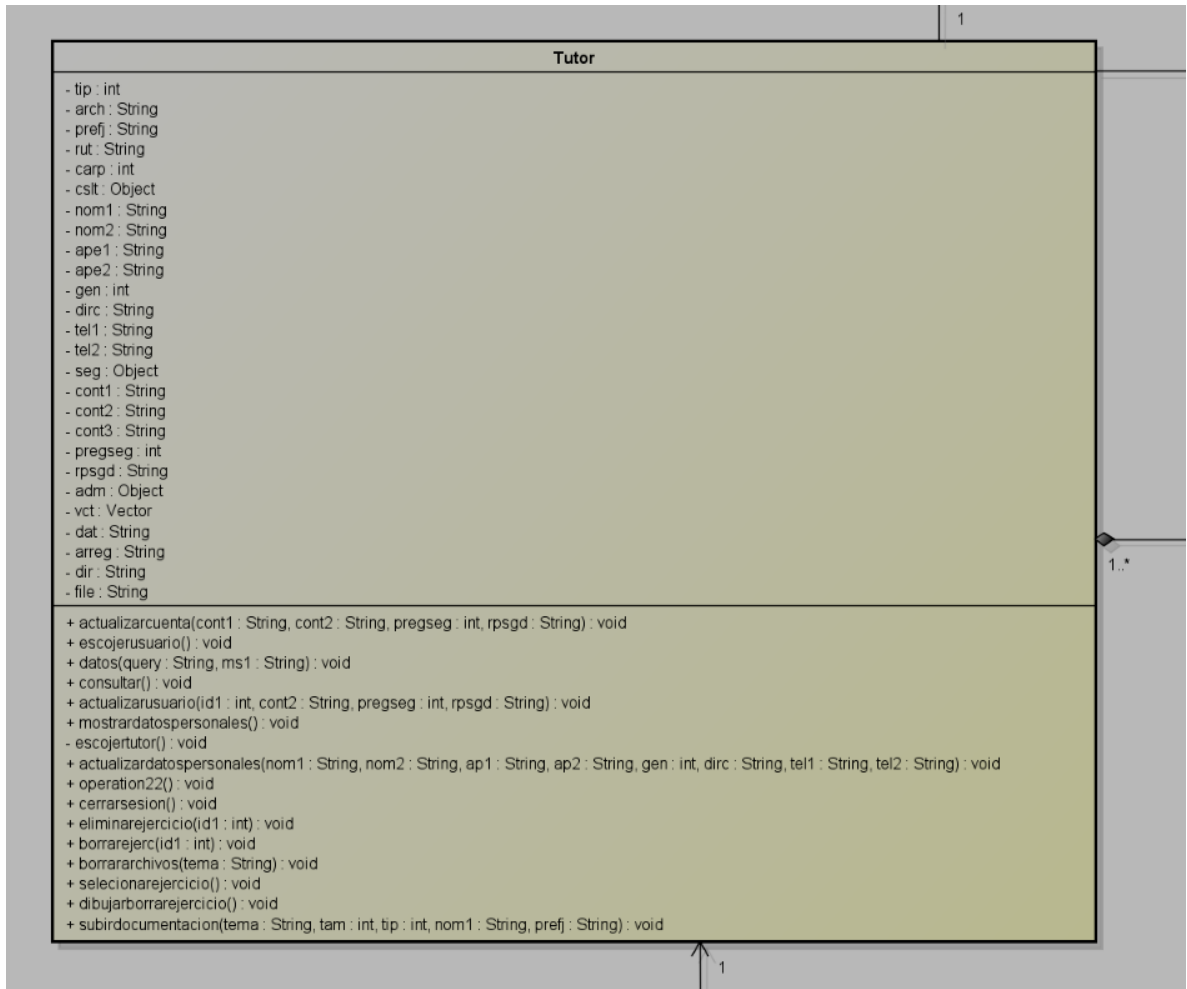
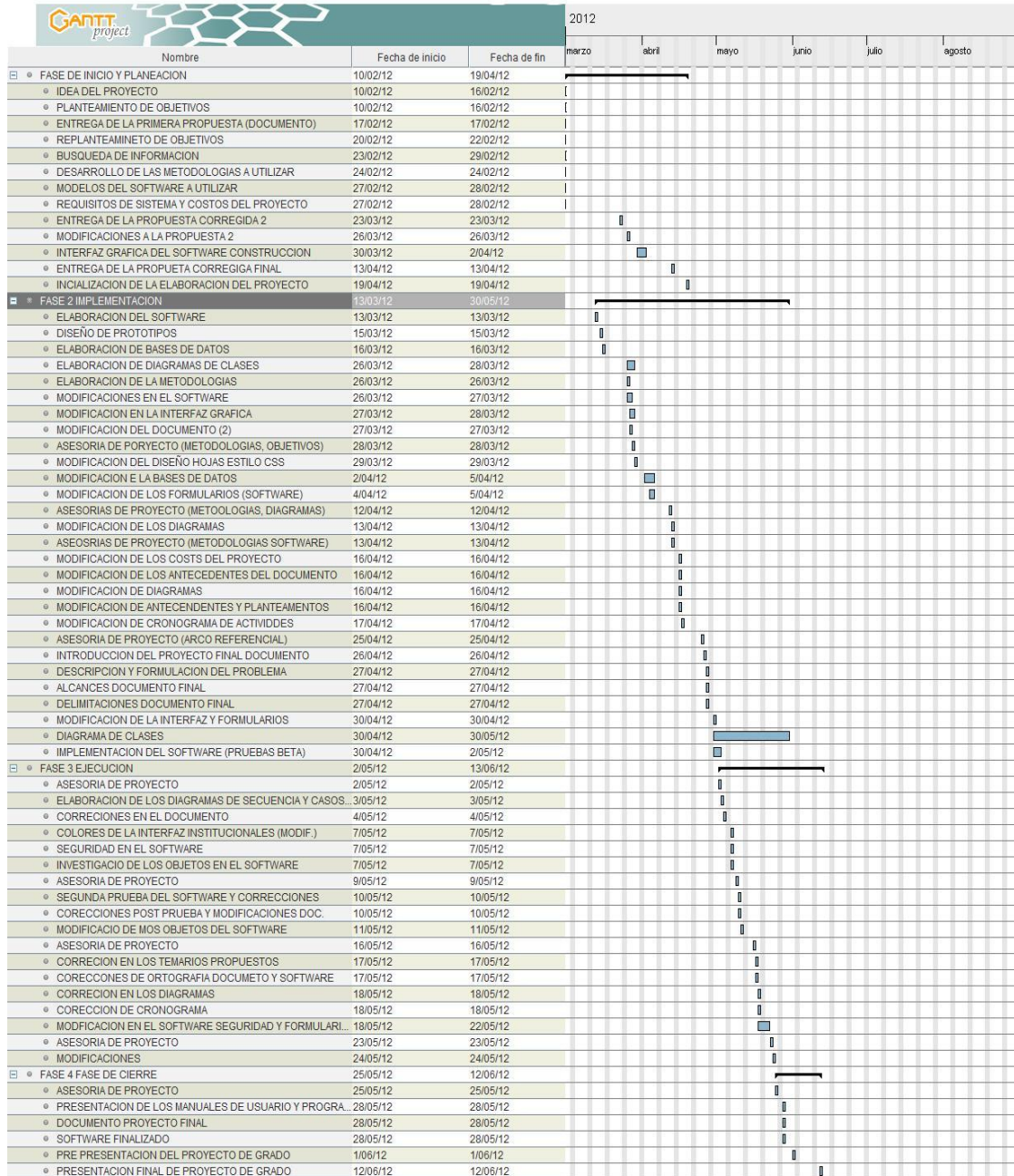


Ilustración 40: Clase tutor.

## ANEXO V. CRONOGRAMA DE ACTIVIDADES.



## ANEXO VI. COSTOS.

ARTÍCULOS	VALOR	REQUISITOS DE SISTEMA OPERATIVO	TIPO DE LICENCIA	DURACIÓN DE LA DE LICENCIA	VERSIÓN	NAVEGADORES COMPATIBLES	ARQUITECTURA EQUIPO
Star UML	\$ -	windows,xp,vista, 7,linux y mac	GNU	gnu libre licencia	5.02	iexplorer9 o sup. Chrome15 o Sup, Firefox 10 o Sup.	
Adobe Dreamweaver	\$ 250.000,00	windows,xp,vista, 7,linux y mac	Shareware	30 días	CS 5.5	iexplorer9 o sup. Chrome15 o Sup, Firefox 10 o Sup.	
Mysql workbench	\$ -	windows,xp,vista, 7,linux y mac	GNU	gnu libre licencia	5.2	iexplorer9 o sup. Chrome15 o Sup, Firefox 10 o Sup.	
Sothink logo maker	\$ 50.000,00	windows,xp,vista, 7,linux y mac	Shareware	30 días	3.1	iexplorer9 o sup. Chrome15 o Sup, Firefox 10 o Sup.	
Appserver	\$ -	windows,xp,vista, 7,linux y mac	GNU	gnu libre licencia	1.777 y 5.8	iexplorer9 o sup. Chrome15 o Sup, Firefox 10 o Sup.	
	\$ -					iexplorer9 o sup. Chrome15 o Sup, Firefox 10 o Sup.	
HOSTING	\$ -		Gratis	gnu libre licencia	N/P	iexplorer9 o sup. Chrome15 o Sup, Firefox 10 o Sup.	Gratis
programa de creación de menús flash	\$ 60.000,00	Windows, xp vista 7	Shareware	Shareware	N/P	iexplorer9 o sup. Chrome15 o Sup, Firefox 10 o Sup.	M.RAM 512MB, PROC. 1GHZ
Sistema Gestor de Bases de datos	\$ -	win, xp vista 7 mb	GNU	gnu libre licencia	N/P	iexplorer9 o sup. Chrome15 o Sup, Firefox 10 o Sup.	M.RAM 512MB, PROC. 1GHZ
Licencia de sistema operativo W7 Home	\$ 540.000,00	Windows	Copyrigh	duración de tipo propietario	6.1 (7601)	iexplorer9 o sup. Chrome15 o Sup, Firefox 10 o Sup.	M.RAM 512MB, PROC. 1GHZ
Papelería	\$ 30.000,00						
Transporte	\$ 50.000,00						
Otros gastos	\$ -						
	\$ -						
	\$ -						
<b>TOTAL</b>	<b>\$ 980.000,00</b>						

Tabla 1: Costos del software utilizado.

## ANEXO VII. COSTOS Y BENEFICIOS.

COSTOS		BENEFICIOS	
EQUIPOS	\$ 1.000.000,00	INFORMACIÓN SEGURA	\$ 2.000.000,00
ADECUACIÓN	\$ 200.000,00	MAS ESPACIO DE MANEJO	\$ 1.000.000,00
INFRAESTRUCTURA	\$ 100.000,00		
LICENCIAMIENTO SOFTW.	\$ 1.000.000,00	SISTEMA LICENCIADO 100%	\$ 1.000.000,00
LICENCIA DE SOFT. APLIC.	\$ 1.000.000,00		
LEVANTAMIENTO DE INFO Y DOC.	\$ 450.000,00	DATOS Y DOC. DIGITALES	\$ 1.000.000,00
PUESTA EN MARCHA Y SEG.	\$ 400.000,00	SEGUIMIENTO PERIÓDICO	\$ 400.000,00
ACTUALIZACIÓN Y ADECUACIÓN	\$ 200.000,00	DATOS A LA FECHA	\$ 500.000,00
ADMINISTRACIÓN	\$ 200.000,00	MANEJO DE LA BASE DE DATOS	\$ 500.000,00
OTROS GASTOS	\$ 200.000,00	AHORRO EN OTROS GASTOS	\$ 2.500.000,00
TOTAL COSTOS	\$ 4.750.000,00	TOTAL BENEFICIOS	\$ 8.900.000,00

Tabla 2: Costos y beneficios



## ANEXO VIII. ENCUESTAS RESULTADOS.

Orden de la clase	intervalo de clase	marca de clase	frecuencia absoluta	frecuencia relativa	Acumuladas			
					Absolutas		Relativas	
J	$Y_{j-1} - Y_j$	$Y_j$	$N_j$	$h_j$	$N_j$	$N_j$	$H_j$	$H_j$
1	8 – 16,4	12,2	31	0,0558559	31	555	0,055856	1
2	16,4 – 24,8	20,6	42	0,0756757	73	524	0,131532	0,944144
3	24,8 – 33,2	29	217	0,390991	290	482	0,522523	0,868468
4	33,2 – 41,6	37,4	210	0,3783784	500	265	0,900901	0,477477
5	41,6 – 50	45,8	55	0,0990991	555	55	1	0,099099
			555	1				

Tabla 3: Resultados encuesta Tec. Inf. 1.

En esta tabla se encuentra la tabulación de los datos que se muestran en la Ilustración 1: encuestas de notas de trabajos en clase.

Orden de la clase	intervalo de clase	marca de clase	frecuencia absoluta	frecuencia relativa	acumuladas			
					Absolutas		Relativas	
J	$Y_{j-1} - Y_j$	$Y_j$	$n_j$	$h_j$	$N_j$	$N_j$	$H_j$	$H_j$
1	9 – 17,2	13,1	20	0,032206	20	621	0,032206	1
2	17,2 – 25,4	20,6	107	0,172303	127	601	0,204509	0,967794
3	25,4 – 33,6	28,8	154	0,247987	281	494	0,452496	0,795491
4	33,6 – 41,8	37	216	0,347826	497	340	0,800322	0,547504
5	41,8 – 50	45,2	124	0,199678	621	124	1	0,199678
	total		621	1				

Tabla 4: Notas parciales

En esta tabla se encuentra la tabulación de los datos que se muestran en la Ilustración 2: Grafica torta de la notas parciales de los estudiantes.

Orden de la clase	intervalo de clase	marca de clase	frecuencia absoluta	frecuencia relativa	acumuladas			
					Absolutas		Relativas	
J	$Y_{j-1} - Y_j$	$Y_j$	$n_j$	$H_j$	$N_j$	$N_j$	$H_j$	$H_j$
1	8 – 16,4	12,2	9	0,016043	9	561	0,016043	1
2	16,4 – 24,8	20,6	35	0,062389	44	552	0,078431	0,983957
3	24,8 – 33,2	29	268	0,477718	312	517	0,55615	0,921569
4	33,2 – 41,6	37,4	195	0,347594	507	249	0,903743	0,44385
5	41,6 – 50	45,8	54	0,096257	561	54	1	0,096257
			561	1	A	D	A	D

Tabla 5: Notas finales

En esta tabla se encuentra la tabulación de los datos que se muestran en la Ilustración 3: Porcentajes Nota final del periodo.

## **ANEXO IX. ENCUESTA REALIZADA.**

Encuesta para determinar los temas que presentan mayores dificultades en la programación orientada a objetos en los estudiantes de la Facultad de Ingeniería. Tecnología en Informática. Corporación universitaria minuto de dios. Regional Soacha.

**1.** Califique de 1 a 6 el grado de dificultad que usted ha encontrado en los siguientes temas. Donde 1 representa que no es difícil, y 6 muy difícil.

- a.** Conceptos básicos.
- b.** Tipos de datos.
- c.** Ciclos.
- d.** Manejo de cadenas.
- e.** Arreglos.

**2.** De acuerdo con su percepción personal a que se deben las dificultades en la programación en la Corporación Universitaria Minuto de Dios Regional Soacha. Califique de 1 a 4 el grado de dificultad que usted ha encontrado en los siguientes temas. Donde 1 representa que no es difícil, y 4 muy difícil

- a.** Complejidad de los temas.
- b.** Poco entusiasmo en los estudiantes.
- c.** Poca claridad en la enseñanza.
- d.** Las estrategias de enseñanza que necesita la programación son especiales.
- e.** Faltan herramientas para estudiar la programación orientada a objetos.

Esta encuesta es totalmente libre y responde a la necesidad de observar la población de estudiantes con dificultad en el aprendizaje en la programación.

Agradecemos su información.

## CONCEPTOS BASICOS.

Tabulación de los datos conceptos básicos en programación de la encuesta realizada a los estudiantes de la programación orientada a objetos. En la Facultad de Ingeniería Tecnología en informática. De la Corporación Universitaria Minuto de Dios. Regional Soacha.

<b>Concepts basicos</b>			
<b>Respuesta</b>	<b>Código</b>	<b>Frecuencia</b>	<b>Porcentaje</b>
Fácil	1	1	1%
Está un poco fácil	2	6	6%
Medio fácil	3	4	4%
Medio	4	11	11%
difícil	5	14	14%
Muy difícil	6	64	64%
<b>TOTAL</b>		<b>100</b>	<b>100%</b>

Tabla 6: Resultados de conceptos básicos en programación.

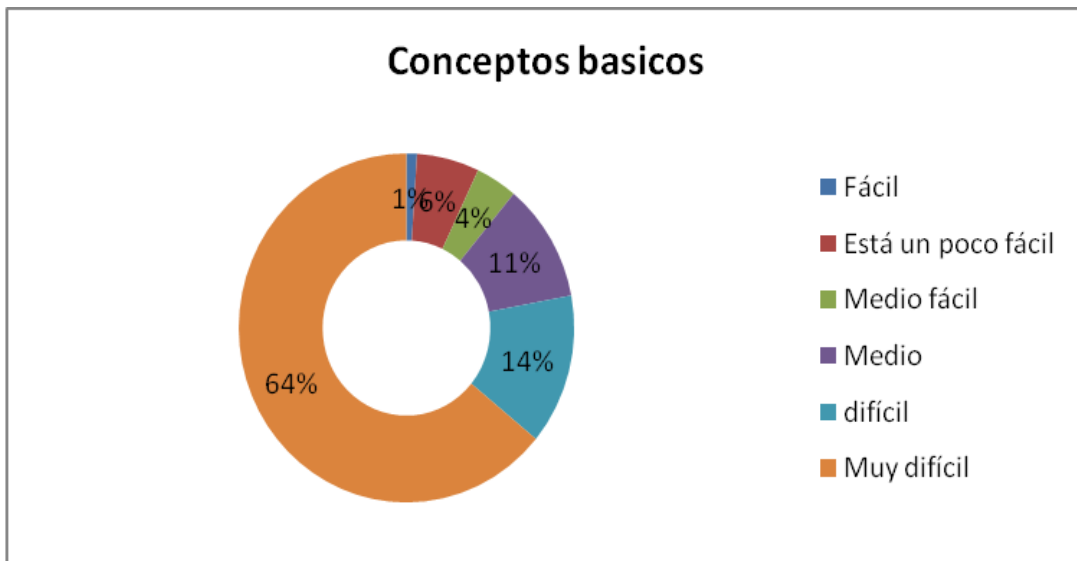


Tabla 7: Torta de resultados en conceptos básicos en programación.

## TIPOS DE DATOS

Tabulación de los datos tipos de datos en programación de la encuesta realizada a los estudiantes de la programación orientada a objetos. En la Facultad de Ingeniería Tecnología en informática. De la Corporación Universitaria Minuto de Dios. Regional Soacha.

<b>Tipos de datos</b>			
<b>Respuesta</b>	<b>Código</b>	<b>Frecuencia</b>	<b>Porcentaje</b>
Fácil	1	5	5%
Está un poco fácil	2	6	6%
Medio fácil	3	7	7%
Medio	4	13	13%
difícil	5	12	12%
Muy difícil	6	57	57%
<b>TOTAL</b>		100	100%

Tabla 8: Resultados de los tipos de datos en programación.

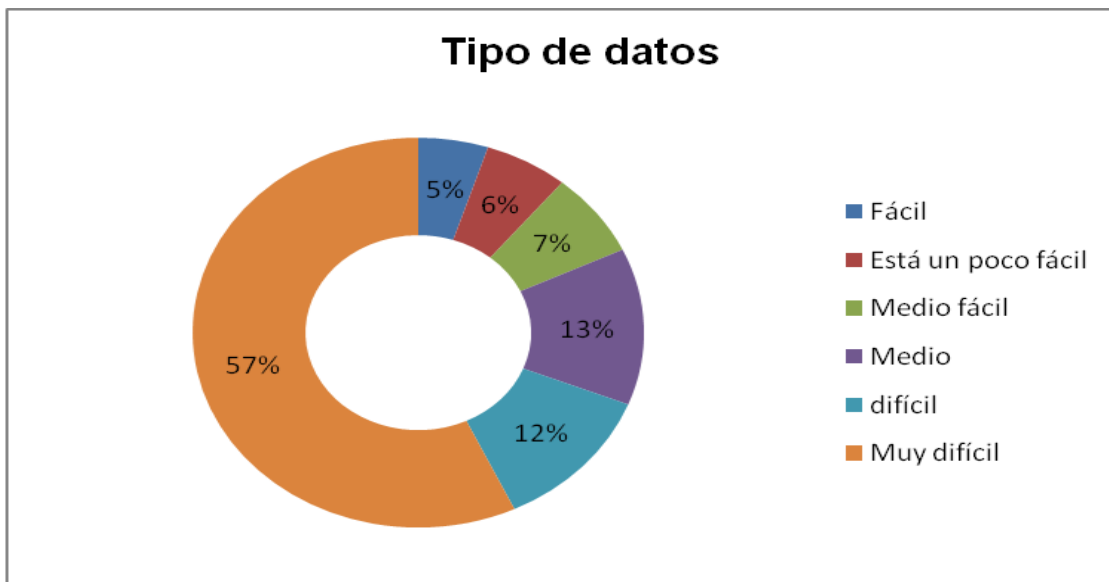


Tabla 9: Torta de resultados de los tipos de datos en programación.

## CICLOS

Tabulación de los datos ciclos en programación de la encuesta realizada a los estudiantes de la programación orientada a objetos. En la Facultad de Ingeniería Tecnología en informática. De la Corporación Universitaria Minuto de Dios. Regional Soacha.

<b>Ciclos</b>			
<b>Respuesta</b>	<b>Código</b>	<b>Frecuencia</b>	<b>Porcentaje</b>
Fácil	1	4	4%
Está un poco fácil	2	8	8%
Medio fácil	3	6	6%
Medio	4	12	12%
difícil	5	19	19%
Muy difícil	6	51	51%
<b>TOTAL</b>		100	100%

Tabla 10: Resultados de ciclos en programación.

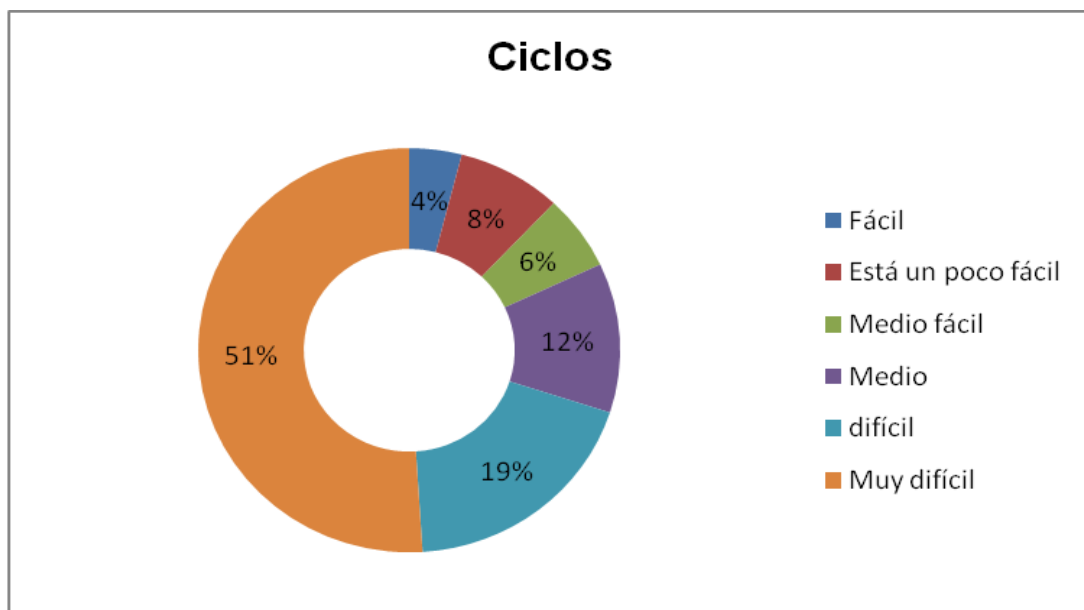


Tabla 11: Torta de resultados de ciclos en programación.

## MANEJO DE CADENAS

Tabulación de los datos manejo de cadenas en programación de la encuesta realizada a los estudiantes de la programación orientada a objetos. En la Facultad de Ingeniería Tecnología en informática. De la Corporación Universitaria Minuto de Dios. Regional Soacha.

<b>Manejo de cadenas</b>			
<b>Respuesta</b>	<b>Código</b>	<b>Frecuencia</b>	<b>Porcentaje</b>
Fácil	1	2	2%
Está un poco fácil	2	8	8%
Medio fácil	3	5	5%
Medio	4	15	15%
difícil	5	12	12%
Muy difícil	6	58	58%
<b>TOTAL</b>		100	100%

Tabla 12: Resultados de manejo de cadenas en programación.

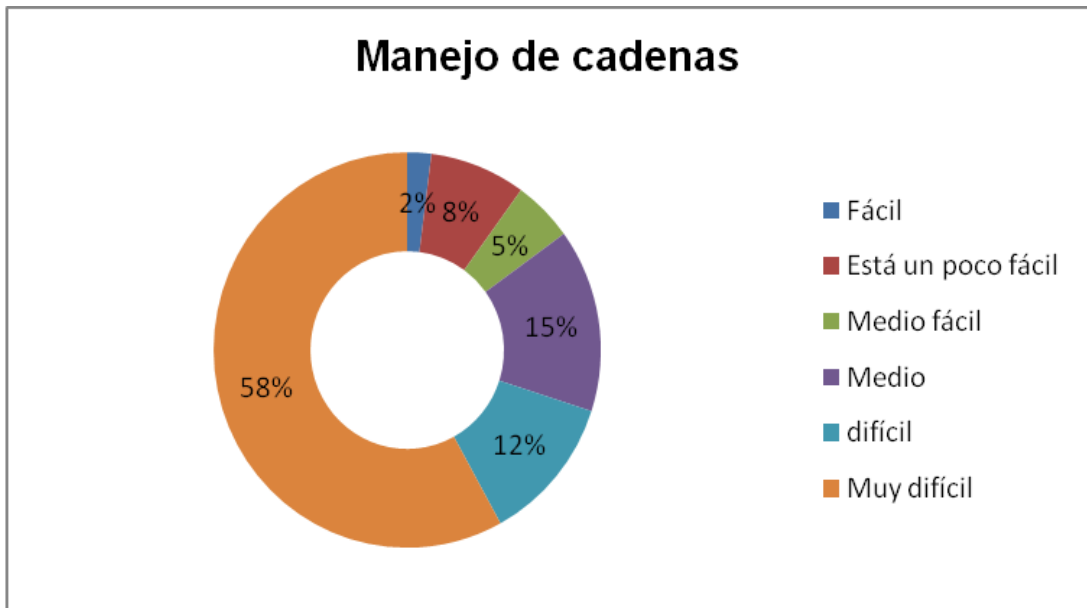


Tabla 13: Torta de resultados de manejo de cadenas en programación.

## ARREGLOS

Tabulación de los datos de arreglos en programación de la encuesta realizada a los estudiantes de la programación orientada a objetos. En la Facultad de Ingeniería Tecnología en informática. De la Corporación Universitaria Minuto de Dios. Regional Soacha.

<b>Herencia y polimorfismo</b>			
<b>Respuesta</b>	<b>Código</b>	<b>Frecuencia</b>	<b>Porcentaje</b>
Fácil	1	1	1%
Está un poco fácil	2	9	9%
Medio fácil	3	6	6%
Medio	4	9	9%
difícil	5	16	16%
Muy difícil	6	59	59%
<b>TOTAL</b>		100	100%

Tabla 14: Resultados de arreglos en programación.

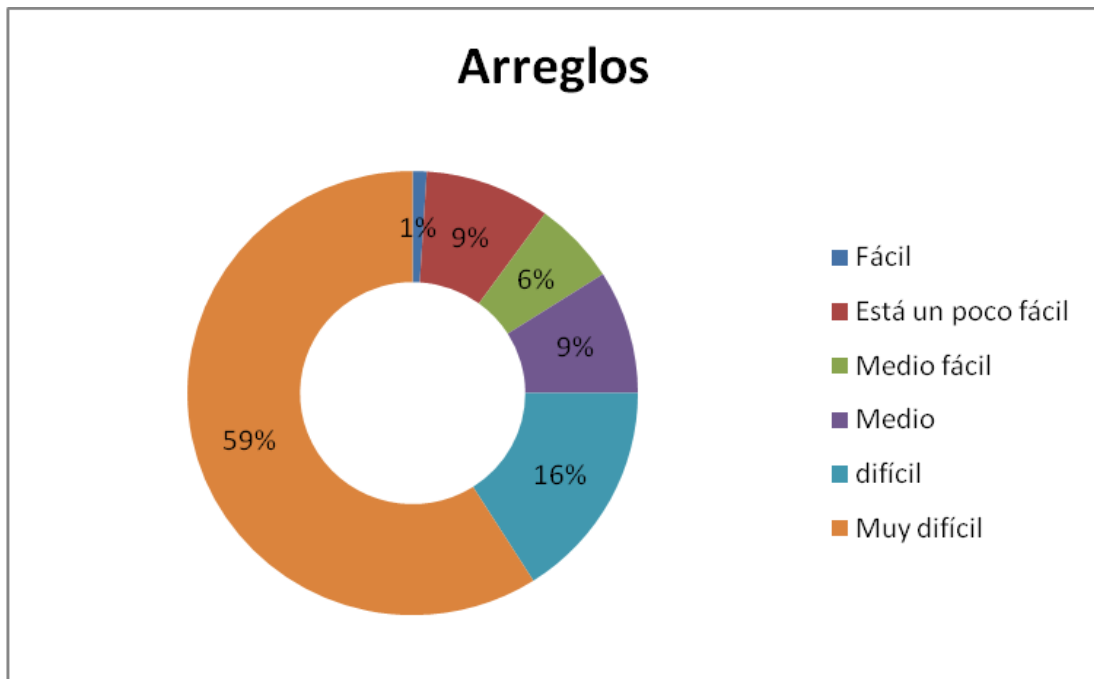


Tabla 15: Torta de resultados de arreglos en programación.



## POCO ENTUSIAMO.

Tabulación de los datos de poco entusiasmo en programación de la encuesta realizada a los estudiantes de la programación orientada a objetos. En la Facultad de Ingeniería Tecnología en informática. De la Corporación Universitaria Minuto de Dios. Regional Soacha.

Poco entusiasmo en los estudiantes			
Respuesta	Código	Frecuencia	Porcentaje
Facil	1	8	8%
Medio facil	2	9	9%
Medio dificil	3	34	34%
Muy dificil	4	49	49%
<b>TOTAL</b>		100	100%

Tabla 16: Resultados de poco entusiasmo en los estudiantes.

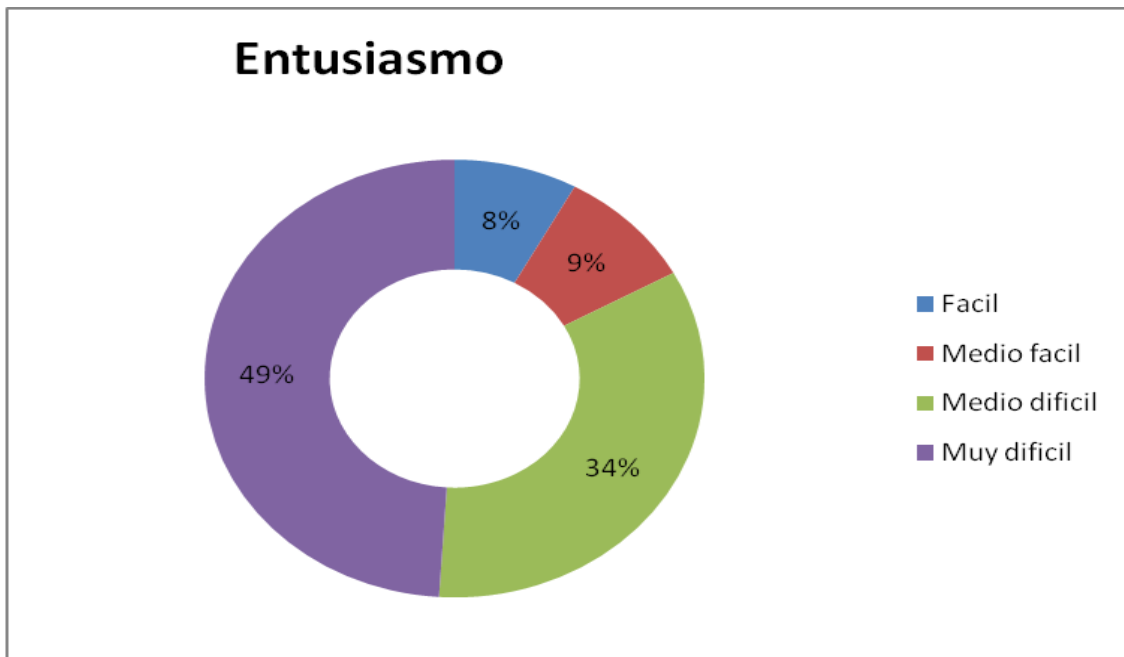


Tabla 17: Torta de Resultados de poco entusiasmo en los estudiantes.

## COMPLEJIDAD DE LOS TEMAS

Tabulación de los datos de complejidad de los temas en programación de la encuesta realizada a los estudiantes de la programación orientada a objetos. En la Facultad de Ingeniería Tecnología en informática. De la Corporación Universitaria Minuto de Dios. Regional Soacha.

<b>Complejidad de los temas</b>			
<b>Respuesta</b>	<b>Código</b>	<b>Frecuencia</b>	<b>Porcentaje</b>
Facil	1	10	10%
Medio facil	2	10	10%
Medio dificil	3	32	32%
Muy dificil	4	48	48%
<b>TOTAL</b>		100	100%

Tabla 18: Resultados de complejidad en los estudiantes.

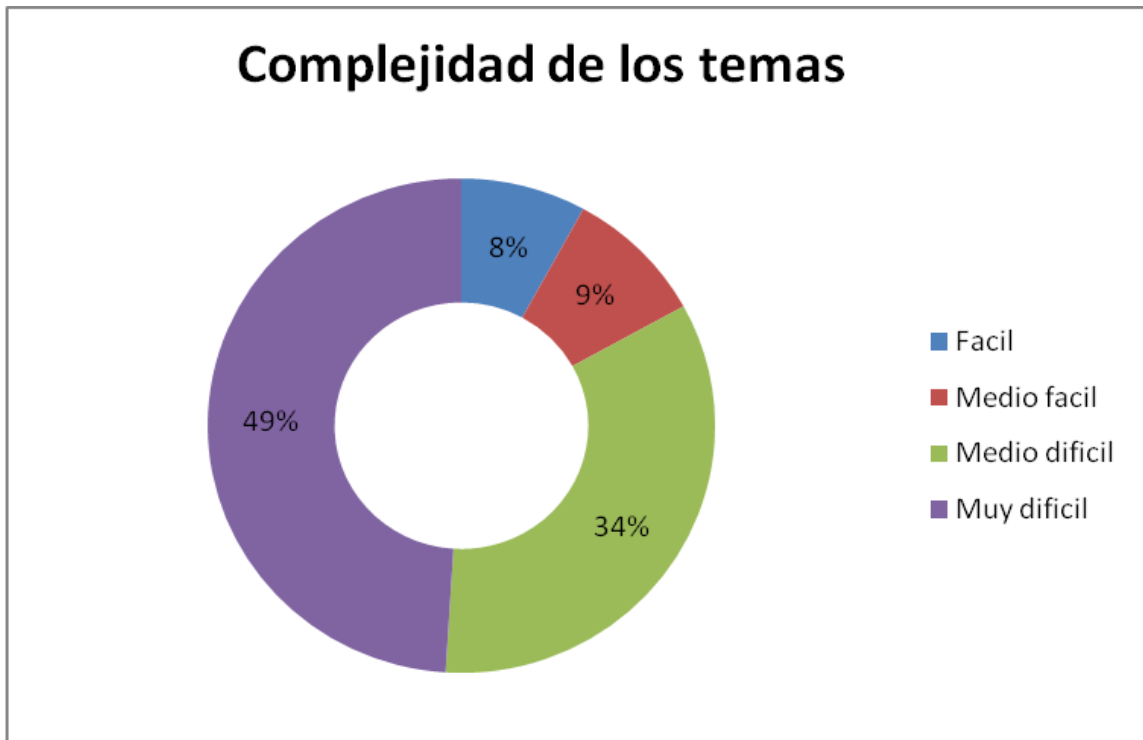


Tabla 19: Torta de resultados de complejidad en los estudiantes.

## POCA CLARIDAD EN LA ENSEÑANZA

Tabulación de los datos de poca claridad en la enseñanza en programación de la encuesta realizada a los estudiantes de la programación orientada a objetos. En la Facultad de Ingeniería Tecnología en informática. De la Corporación Universitaria Minuto de Dios. Regional Soacha.

<b>Poca claridad en la enseñanza</b>			
<b>Respuesta</b>	<b>Código</b>	<b>Frecuencia</b>	<b>Porcentaje</b>
Facil	1	8	8%
Medio facil	2	8	8%
Medio dificil	3	39	39%
Muy dificil	4	45	45%
<b>TOTAL</b>		100	100%

Tabla 20: Resultados de poca claridad en los estudiantes.

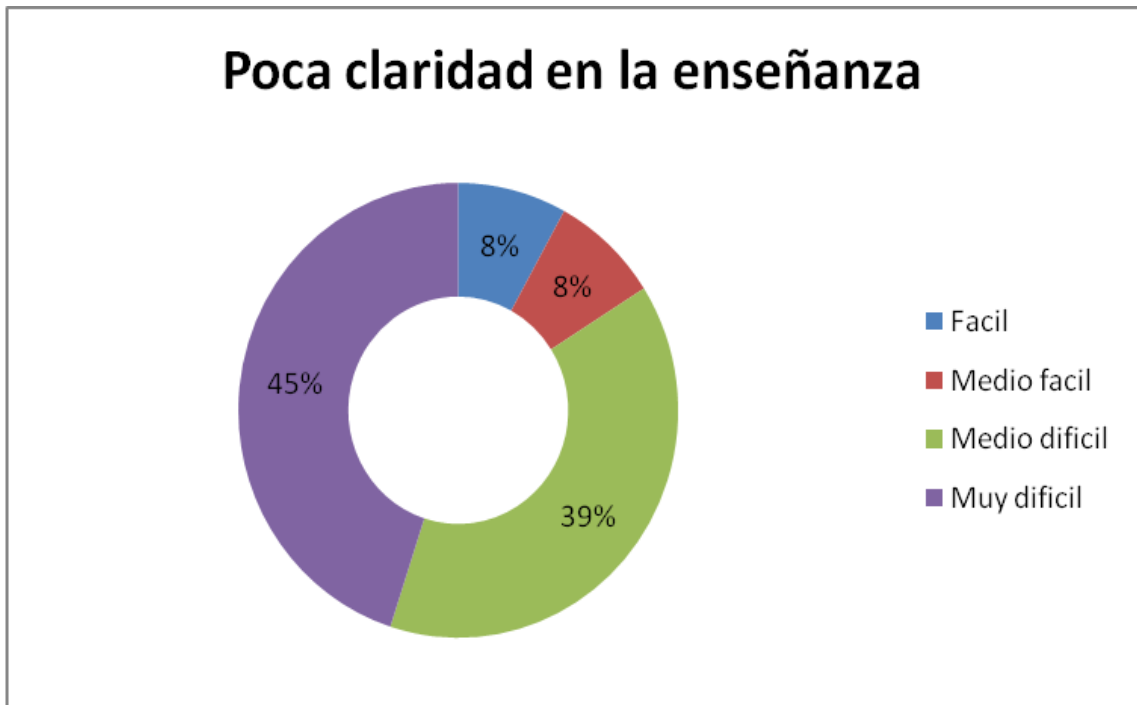


Tabla 21: Torta de resultados de poca claridad en los estudiantes.

## LAS ESTRATEGIAS DE ENSEÑANZA QUE NECESITA LA PROGRAMACIÓN SON ESPECIALES

Tabulación de los datos de Las estrategias de enseñanza que necesita la programación son especiales en programación de la encuesta realizada a los estudiantes de la programación orientada a objetos. En la Facultad de Ingeniería Tecnología en informática. De la Corporación Universitaria Minuto de Dios. Regional Soacha.

Las estrategias de enseñanza que necesita la programación son especiales			
Respuesta	Código	Frecuencia	Porcentaje
Facil	1	8	8%
Medio facil	2	12	12%
Medio dificil	3	36	36%
Muy dificil	4	44	44%
<b>TOTAL</b>		100	100%

Tabla 22: Resultados de estrategias de enseñanzas en los estudiantes.



Tabla 23: Torta de resultados de estrategias de enseñanza en los estudiantes.

## FALTAN HERRAMIENTAS PARA ESTUDIAR LA PROGRAMACIÓN ORIENTADA A OBJETOS

Tabulación de los datos de faltan herramientas para estudiar la programación orientada a objetos en programación de la encuesta realizada a los estudiantes de la programación orientada a objetos. En la Facultad de Ingeniería Tecnología en informática. De la Corporación Universitaria Minuto de Dios. Regional Soacha.

Faltan herramientas para estudiar la programación orientada a objetos			
Respuesta	Código	Frecuencia	Porcentaje
Facil	1	9	9%
Medio facil	2	10	10%
Medio dificil	3	39	39%
Muy dificil	4	42	42%
<b>TOTAL</b>		100	100%

Tabla 24: Resultados de herramientas para estudiar en los estudiantes.



Tabla 25: Torta de resultados de herramientas para estudiar en los estudiantes.