

**Universidad Internacional de La Rioja
Máster universitario en Ingeniería de Software y
Sistemas Informáticos**

Desarrollo de un sistema de
control de asistencia estudiantil
mediante reconocimiento facial.

Trabajo Fin de Máster

Tipo de trabajo: Desarrollo práctico

Presentado por: Salvatierra Tumbaco, Gabriel

Directora: Fuentes Lorenzo, Damaris

Ciudad: Portoviejo

Fecha: 18 de Septiembre de 2018

Resumen

El control de asistencias, es un proceso que toda institución, empresa u organización debe realizar diariamente para vigilar la correcta labor del personal. Sin embargo, este se convierte en problema cuando el número de personas a controlar es grande y no se dispone de mecanismos que ayuden a agilizar el proceso de toma de asistencias. El presente trabajo de fin de master, muestra el desarrollo de un sistema de control de asistencias utilizando el reconocimiento facial, a través de la integración de servicios Cloud de Amazon Web Services.

El objetivo principal, es controlar el registro de asistencias de los estudiantes de la Unidad Educativa “Julia Navarrete Mendoza”, de forma masiva, es decir, que mediante fotografías tomadas a los estudiantes del curso, se reconozcan los rostros e identifique a las personas que se encuentran presentes.

Para controlar el proceso de desarrollo del software, se utilizó la metodología RUP (Proceso Racional Unificado), teniendo en cuenta cada una de las actividades y las disciplinas que garantiza la creación de un sistema de calidad. Además, la metodología está dirigida por casos de uso, centrado en la arquitectura de componentes y adaptable a la necesidad del proyecto y requisitos cambiantes de usuario.

Para validar los requisitos del software, se estableció un plan de pruebas por cada requisito funcional y no funcional; además, para garantizar la usabilidad, integridad y disponibilidad del sistema, se utilizó el test SUMI centrado desde la perspectiva del usuario final.

Al final se obtuvo un producto funcional, que permite el registro de la información de los involucrados en las asistencias estudiantiles y comunicaciones a padres de familia, logrando con éxito el objetivo de este trabajo, que es la utilización de servicios Cloud para el reconocimiento facial masivo.

Palabras Clave: Reconocimiento facial, asistencia masiva, test usabilidad, proceso racional unificado.

Abstract

The control of assistance, is a process that every institution, company or organization must perform daily to monitor the proper work of the staff. However, this becomes a problem when the number of people to be controlled is large and there are no mechanisms available to help speed up the process of taking care. The present work of end of master, shows the development of a system of control of assistances using the facial recognition, through the integration of services Cloud of Amazon Web Services.

The main objective is to control the attendance record of the students to de High School "Julia Navarrete Mendoza", in a massive way, that is to say, that by means of photographs taken to the students of the course, the faces are recognized and identify to the people presents.

To control the software development process, the RUP (Unified Rational Process) methodology was used, taking into account each of the activities and disciplines that guarantee the creation of a quality system. In addition, the methodology is directed by use cases, focused on the architecture of components and adaptable to the need of the project and changing user requirements.

To validate the software requirements, a test plan was established for each functional and non-functional requirement; In addition, to ensure the usability, integrity and availability of the system, the SUMI test was used from the perspective of the end user.

In the end a functional product was obtained, which allows the registration of the information of those involved in student assistance and communications to parents, successfully achieving the objective of this work, which is the use of Cloud services for mass facial recognition.

Keywords: Facial recognition, massive assistance, usability test, Rational Unified Process.

Índice de contenidos

1.	Introducción	12
1.1.	Justificación	13
1.2.	Planteamiento del trabajo.....	13
1.3.	Estructura de la memoria	14
2.	Contexto y estado del arte	16
2.1.	Reconocimiento facial	16
2.1.1.	Adquisición de la imagen.....	17
2.2.	Amazon Rekognition	17
2.2.1.	Funcionamiento	18
2.3.	Antecedentes	20
2.4.	Conclusión del estado del arte	21
3.	Objetivos concretos y metodología de trabajo	23
3.1.	Objetivo general	23
3.2.	Objetivos específicos	23
3.3.	Metodología de desarrollo RUP (Proceso Unificado Racional).....	24
3.3.1.	Fases del ciclo de vida	24
4.	Inicio	27
4.1.	Introducción	27
4.1.1.	Alcance	27
4.1.2.	Definición de acrónimos y abreviaturas.....	28
4.1.3.	Referencias.....	28
4.2.	Descripción general del sistema.....	28
4.2.1.	Perspectiva del producto.....	29
4.2.2.	Funcionalidad del producto	29
4.2.3.	Características de los usuarios del sistema.....	29
4.2.4.	Restricciones	30

4.2.5.	Suposiciones y dependencias	30
4.2.6.	Requisitos futuros	31
4.3.	Especificación de requisitos	31
4.3.1.	Requisitos funcionales	31
4.3.2.	Requisitos no funcionales	34
4.4.	Estudio de factibilidad	36
4.4.1.	Humana	36
4.4.2.	Legal	36
4.4.3.	De sistema	37
4.4.3.1.	Técnica.....	37
4.4.3.2.	Operativa.....	37
4.4.3.3.	Tecnológica	37
4.4.3.4.	Operacional	38
5.	Elaboración	39
5.1.	Entorno del software	39
5.2.	Arquitectura candidata	39
5.3.	Herramientas tecnológicas a utilizar en el desarrollo.....	40
5.3.1.	Vaadin framework.....	40
5.3.2.	Eclipse IDE	40
5.3.3.	Power Designer	40
5.3.4.	Apache Tomcat	41
5.3.5.	MySQL	41
5.3.6.	Amazon web services.....	41
5.3.6.1.	Elastic cloud computing EC2.....	41
5.3.6.2.	Amazon Simple Storage Service S3.....	41
5.3.6.3.	Amazon Rekognition	42
5.4.	Participantes en el desarrollo	42
5.5.	Riesgos técnicos	43

5.6.	Plan de pruebas	43
5.7.	Arquitectura a implementar	44
5.8.	Vista de casos de uso	45
5.8.1.	Diagrama de casos de uso	45
5.9.	Representación gráfica de procesos	49
5.10.	Especificación de casos de uso	50
6.	Construcción	60
6.1.	Vista lógica.....	60
6.1.1.	Diagrama de secuencias	60
6.1.2.	Diagrama de clases	68
6.1.3.	Diseño de la base de datos	73
6.2.	Vista de procesos.....	76
6.2.1.	Situación actual del proceso	76
6.2.2.	Identificación de procesos	77
6.3.	Vista de despliegue.....	78
6.4.	Vista de implementación	79
6.5.	Diccionario de datos.....	79
6.6.	Diseño de modelo en capas.....	80
6.7.	Diseño de interfaces del sistema.....	81
6.8.	Estándares de codificación	82
6.8.1.	Generales.....	82
6.8.2.	Indentación.....	82
6.8.3.	Nombrado de símbolos.....	82
6.8.4.	Espaciado dentro del código.....	83
6.8.5.	Comentarios dentro del código	83
7.	Transición	84
7.1.	Implementación.....	84
	Configuración de Instancia EC2.....	84

Compilación y despliegue de aplicación	85
Implementación de base de datos	86
Imágenes de entrenamiento en S3 y colección de rostros en Rekognition	86
7.2. Evaluación del sistema.....	88
7.2.1. Usabilidad.....	88
7.2.1.1. Sujetos experimentales.....	88
7.2.1.2. Formato de test.....	88
7.2.1.3. Aplicación del test.....	89
7.2.1.4. Resultado del test	90
7.2.2. Trazabilidad de requisitos.....	93
7.2.2.1. Requisitos funcionales	93
7.2.2.2. Requisitos no funcionales	104
8. Conclusiones y trabajo futuro	107
8.1. Conclusiones	107
8.2. Líneas de trabajo futuro	108
Bibliografía.....	109
Anexos.....	111
Anexo I. Artículo	111
Anexo II. Interfaces del sistema.....	117
Anexo III. Ficha de aplicación de test SUMI.....	120
Anexo IV. Evidencias.....	122
Anexo V. Diccionario de datos.....	124

Índice de tablas

Tabla 1: Características de Amazon Rekognition	17
Tabla 2: Glosario de términos.....	28
Tabla 3: Referencia de estándar para desarrollo de software.....	28
Tabla 4: Característica de usuario administrador.....	29
Tabla 5: Característica de usuario secretaria.....	30
Tabla 6: Característica de usuario docente.....	30
Tabla 7: Requisito funcional gestionar usuario.....	31
Tabla 8: Requisito funcional autenticar y autorizar usuario	31
Tabla 9: Requisito funcional gestionar estudiante.....	32
Tabla 10: Requisito funcional gestionar padre de familia	32
Tabla 11: Requisito funcional gestionar cursos.....	32
Tabla 12: Requisito funcional gestionar asistencia estudiantil.....	33
Tabla 13: Requisito funcional gestionar reportes	33
Tabla 14: Requisito funcional gestionar comunicaciones.....	33
Tabla 15: Requisito funcional gestionar docente.....	33
Tabla 16: Requisito no funcional espacio y rendimiento	34
Tabla 17: Requisito no funcional usabilidad	35
Tabla 18: Requisito no funcional seguridad	35
Tabla 19: Requisito no funcional portabilidad.....	36
Tabla 20: Requisito no funcional operacional.....	36
Tabla 21: Matriz de roles y responsabilidades	42
Tabla 22: Simbología de diagrama de flujo.....	49
Tabla 23: Descripción de caso de uso gestionar usuario	50
Tabla 24: Descripción de caso de uso autenticar y autorizar	51
Tabla 25: Descripción de caso de uso gestionar estudiante	52
Tabla 23: Descripción de caso de uso gestionar padre de familia.....	53

Tabla 27: Descripción de caso de uso gestionar curso	54
Tabla 23: Descripción de caso de uso gestionar asistencia estudiantil	55
Tabla 29: Descripción de caso de uso gestionar comunicaciones	56
Tabla 30: Descripción de caso de uso gestionar informes	57
Tabla 31: Descripción de caso de uso gestionar docente	58
Tabla 32: Descripción de caso de uso gestionar docente	89
Tabla 33: Caso de prueba RF01 Gestionar usuario	93
Tabla 34: Caso de prueba RF02 autenticar y autorizar usuario	94
Tabla 35: Caso de prueba RF03 Gestionar estudiante	95
Tabla 36: Caso de prueba RF04 gestionar padre de familia	96
Tabla 37: Caso de prueba RF05 gestionar cursos	97
Tabla 38: Caso de prueba RF06 gestionar asistencia estudiantil	101
Tabla 39: Caso de prueba RF07 gestionar reportes	102
Tabla 40: Caso de prueba RF08 gestionar comunicaciones	102
Tabla 41: Caso de prueba RF09 Gestionar docente	103
Tabla 42: Caso de prueba RNF01 espacio y rendimiento	104
Tabla 43: Caso de prueba RNF02 usabilidad	105
Tabla 44: Caso de prueba RNF03 seguridad	105
Tabla 45: Caso de prueba RNF04 portabilidad	105
Tabla 46: Caso de prueba RNF05 operacional	106

Índice de figuras

Figura 1: Tipos de reconocimiento Amazon Rekognition	18
Figura 2: Tipos de detección y reconocimiento.....	19
Figura 3: Operaciones API de almacenamiento y sin almacenamiento	19
Figura 4: Fases y disciplinas de RUP	25
Figura 5: Funcionalidad general de SISCAE	29
Figura 6: Tipos de requerimientos no funcionales	34
Figura 7: Arquitectura en capas.....	39
Figura 8: Arquitectura genérica en 4 capas	44
Figura 9: Diseño arquitectónico del sistema SISCAE	44
Figura 10: Caso de uso, diagrama general.....	45
Figura 11: Caso de uso, gestionar usuario	45
Figura 12: Caso de uso, autenticar y autorizar usuario.....	46
Figura 13: Caso de uso, gestionar estudiantes.....	46
Figura 14: Caso de uso, gestionar padre de familia.....	46
Figura 15: Caso de uso, gestionar curso	47
Figura 16: Caso de uso, gestionar asistencia estudiantil	47
Figura 17: Caso de uso, gestionar comunicaciones	48
Figura 18: Caso de uso, gestionar informes	48
Figura 19: Caso de uso, gestionar docente	49
Figura 20: Diagrama de flujos de procesos del sistema SISCAE.....	50
Figura 21: Diagrama de secuencia, gestionar usuario	60
Figura 22: Diagrama de secuencia, gestionar autenticar y autorizar	61
Figura 23: Diagrama de secuencia, gestionar estudiante	61
Figura 24: Diagrama de secuencia, gestionar padre de familia	62
Figura 25: Diagrama de secuencia, gestionar curso.....	63
Figura 26: Diagrama de secuencia, gestionar asistencia estudiantil.....	64

Figura 27: Diagrama de secuencia, gestionar comunicaciones	65
Figura 28: Diagrama de secuencia, gestionar informes	66
Figura 29: Diagrama de secuencia, gestionar docente	67
Figura 30: Diagrama de clases, SISCAE	68
Figura 31: Diseño lógico de la base de datos del sistema	73
Figura 32: Diseño físico de la base de datos del sistema	74
Figura 33: Proceso actual de registro de asistencias masivas o individuales	76
Figura 32: Gestión de proceso matriculación estudiante curso	77
Figura 35: Gestión de proceso asignación docente curso	77
Figura 36: Gestión de proceso asistencia estudiantil	78
Figura 37: Gestión de proceso envío de comunicaciones	78
Figura 37: Diagrama de despliegue	78
Figura 39: Diagrama de componentes del sistema	79
Figura 39: Diagrama de componentes del sistema	80
Figura 41: Diseño final de la base de datos	81
Figura 42: Instancia EC2, Windows Server 2012 R2	84
Figura 43: Datos de configuración para AWS CLI	84
Figura 44: Configuración de variables de entorno	84
Figura 45: Verificación de ejecución de servicios en el Windows Server	85
Figura 46: Compilación de aplicación a .war	85
Figura 47: Administrador de MySQL	86
Figura 48: Bucket de imágenes con las fotografías de cada estudiante	86
Figura 49: Funcionamiento de Rekognition	87
Figura 50: Prueba del sistema, desplegado en una instancia de AWS	87
Figura 51: Manipulación del sistema y aplicación de test SUMI	88
Figura 52: Resultado pregunta 2 SUMI	91
Figura 53: Resultado pregunta 8 SUMI	91
Figura 54: Resultado pregunta 13 SUMI	91

Figura 55: Resultado pregunta 25 SUMI92

Figura 56: Resultado pregunta 37 SUMI92

1. Introducción

El control de asistencias, es un proceso que toda institución, empresa u organización debe realizar diariamente para vigilar la correcta labor del personal. Sin embargo, este se convierte en problema cuando el número de personas a controlar es grande y no se dispone de mecanismos que ayuden a agilizar el proceso de toma de asistencias. Además, la generación de reportes al final de un periodo es necesaria para validar el trabajo realizado por el personal.

El control de asistencia es tan importante como necesario en cualquier ámbito; por ejemplo, en eventos masivos donde se escapa de las manos este control, se pueden implementar herramientas tecnológicas que ayuden a realizar este fin; y es ahí donde aparecen los sistemas biométricos (huellas, reconocimiento facial, reconocimiento de voz, entre otros) que además de proporcionar agilidad es una forma segura de identificar que un usuario es quien dice ser.

En el campo educativo, la asistencia estudiantil es fundamental para controlar el proceso de estudio de un alumno, es decir, se deben controlar los días asistidos, atrasos y justificaciones. Estos datos, deben ser presentados en informes al terminar un parcial con las evidencias del caso.

Para solventar los requisitos de la Unidad Educativa “Julia Navarrete Mendoza” respecto al manejo de asistencias, se desea desarrollar una aplicación que permita llevar fácilmente el control de las asistencias de estudiantes en la Institución, utilizando tecnologías de Cloud para el reconocimiento facial, almacenamiento y consulta de información.

Con este sistema, el control de asistencias estudiantiles será más eficiente para los docentes y personal administrativo de la Unidad Educativa. Además, los padres estarán informados sobre las inasistencias de sus representados mediante correos electrónicos.

Se debe mencionar que este trabajo es de tipo piloto experimental, ya que se basa en el estudio y uso de varias tecnologías del Cloud Computing aplicadas en el desarrollo de la aplicación, para el uso en una sola aula de clases.

Para el desarrollo de la aplicación se utilizará la metodología de desarrollo de software RUP (Rational Unified Process), la cual guiará la construcción del sistema en todas sus fases. La

ventaja de RUP, es una metodología de carácter adaptativo a las necesidades de la Institución correspondiente a los requisitos del software. Para la documentación referente a los requisitos de software, se tomará como guía una parte del estándar IEEE 830-1998, mientras que, para la codificación y almacenamiento de datos se usará JAVA con el Framework VAADIN y el sistema gestor de base de datos (SGDB) MySQL respectivamente.

Lo mencionado anteriormente, estará desplegado sobre una infraestructura en la nube utilizando los servicios de Amazon Web Services (AWS). Además, se utilizará Amazon Rekognition para el proceso de reconocimiento facial y otras tecnologías que se irán mencionando en este trabajo de fin de máster.

1.1. Justificación

La iniciativa para la investigación del presente trabajo de fin de máster surgió observando la necesidad de un sistema que controle la asistencia de los estudiantes de la Unidad Educativa “Julia Navarrete Mendoza”. Actualmente los docentes registran la asistencia de los alumnos en hojas de cálculo de Excel y en algunos casos cuadernos físicos lo que provoca minutos de retraso para impartir la clase planificada. Actualmente la Unidad Educativa cuenta con 275 estudiantes, pero al ser una aplicación piloto experimental solo se delimitará a un aula de clase.

Cuando se desea evaluar la presencia de las personas en eventos masivos, este control se escapa de las manos, por ende esta aplicación es relevante y a la vez novedosa porque solucionaría en gran parte el proceso de registro de los asistentes a eventos (clases, reuniones). Además, utilizar técnicas de reconocimiento facial a través de servicios Cloud, proporcionan agilidad, seguridad y confianza de que la persona que se identifica es la persona correcta y reconocida por una base de datos de caras extraídas de las imágenes de entrenamiento.

Se pretende disponer de una aplicación web que agilice el tiempo de registro de las asistencias mediante servicios de Cloud proporcionados por Amazon Web Services.

1.2. Planteamiento del trabajo

Para solventar la problemática se desarrollará una aplicación que permita llevar el control de las asistencias de los estudiantes de la Unidad Educativa “Julia Navarrete Mendoza” cambiando la forma de registro actual de cómo se lleva este proceso (papel, hojas de Excel)

a una aplicación en entorno Web desplegada en un modelo Cloud IAAS, y que además lo realice de forma automática mediante los servicios de reconocimiento facial otorgados por Amazon Web Services; lo cual mejorará tanto el control de asistencias de los alumnos como la confianza hacia los sistemas que utilizan las tecnologías de reconocimiento facial. Además, se informará al padre de familia en caso de existir una o más faltas a clases de su representado.

1.3. Estructura de la memoria

La estructura de la presente memoria está distribuida de la siguiente manera:

Capítulo 1: Introducción al trabajo de fin de máster. En este capítulo se detalla una breve explicación del trabajo realizado, además, esta introducción está comprendida por la justificación del tema a tratar, el planteamiento de la problemática y la estructura del documento o memoria del TFM.

Capítulo 2: Este apartado lo conforma el contexto y estado del arte, en donde se analizan los estudios previos realizados por otros investigadores y las herramientas tecnológicas utilizadas en el desarrollo del aplicativo. También, se incluye una conclusión del estado del arte para demostrar la posibilidad del estudio.

Capítulo 3: Objetivos concretos y metodologías del trabajo, en esta sección se da a conocer la finalidad del estudio y el porqué de su realización, tomando en cuenta que se especifican los objetivos, tanto general como los específicos. Así mismo, se describe la metodología de trabajo, que servirá como guía durante todo el proceso de desarrollo del TFM.

En los capítulos 4, 5, 6 y 7 se describe el proceso de desarrollo específico de la contribución, en el cual intervienen cada una las fases que indica la metodología de desarrollo de software Rational Unified Process (RUP), dichas fases son: inicio, elaboración, construcción y transición.

Capítulo 4: Fase **inicio**, describe el alcance que la aplicación a desarrollar va a tener, también, una descripción general del software donde se detalla una perspectiva del producto, la funcionalidad, características de usuarios, restricciones, suposiciones y dependencias. Así mismo, se incluye la especificación de requerimientos utilizando una parte del estándar IEEE 830-1998 como guía para la captación de requisitos funcionales y no funcionales. Por último, se toma en cuenta estudios de factibilidad.

Capítulo 5: Fase de **elaboración**, contiene apartados como el entorno del software, la arquitectura a utilizar, la descripción de los participantes, riesgos y un plan de pruebas a realizar sobre la aplicación, así como, la vista del software utilizando los casos de uso para realizar la representación gráfica de los procesos. También, se incluyen las fichas de la descripción de los casos de uso.

Capítulo 6: Fase de **construcción**, incluye el modelado del software desde el punto de vista lógico, procesos, despliegue e implementación. Además, se detallan los diccionarios de datos y el diseño de las interfaces. En esta fase, se realiza el modelado y la codificación de estos modelos, estableciendo estándares de programación de acuerdo a la arquitectura seleccionada.

Capítulo 7: Fase de **transición**, especifica el proceso de implementación de la aplicación y los requisitos necesarios del entorno de despliegue. También, se presenta la evaluación del sistema respecto a: usabilidad y la precisión de reconocimiento facial. Del mismo modo, se presenta la trazabilidad de requisitos funcionales y no funcionales.

Capítulo 8: Conclusiones, recomendaciones y trabajo futuro. En este último capítulo se presentan las conclusiones y recomendaciones para trabajos futuros.

Bibliografía: Referencias bibliográficas utilizados para el desarrollo de la memoria.

Anexos: Recursos utilizados para cada capítulo de la memoria.

2. Contexto y estado del arte

2.1. Reconocimiento facial

El reconocimiento facial es un sistema utilizado para la identificación de personas por medio de imágenes, las cuales pueden ser tomadas y almacenadas, o adquiridas en un sistema de tiempo real. Para el reconocimiento facial se deben de conocer principalmente algunos elementos, ya que existen variables externas que afectan exponencialmente el procesamiento de la imagen; al igual se puede realizar un proceso más efectivo dependiendo de la técnica de reconocimiento que se desea trabajar. El reconocimiento facial tiene un proceso estándar de desarrollo el cual puede ser muy explícito en la siguiente imagen. (Bertino Salazar & Orozco Alzate)

Para llevar a cabo el reconocimiento facial se deben considerar dos tareas básicas, la detección que permite ubicar uno o varios rostros dentro de una imagen (fija o secuencia de video) y el reconocimiento el cual compara la cara detectada con otras almacenadas previamente en una base de datos. Para poder realizar el presente trabajo se consideró el uso de un dispositivo de detección (cámara) y un servidor de reconocimiento (Amazon Rekognition)

El **proceso** consta de lo siguiente:

- Capturar la imagen
- Detectar las caras
- Enviar la región seleccionada a un servidor de reconocimiento
- Comparar la caras encontradas con las almacenadas
- Esperar respuesta del servidor con los rostros identificados

Finalmente, se espera recibir datos provenientes de los dispositivos que ejecutan la aplicación cliente e interactúa con los servidores de reconocimiento para satisfacer las necesidades oportunas.

Tareas realizadas en el servidor de reconocimiento

- Pre procesar la imagen que ha recibido del cliente
- Realizar el reconocimiento
- Obtener información extra
- Mejorar las características del clasificador
- Dar respuesta al cliente con los rostros encontrados

Tareas en el cliente

Son tareas livianas que podrá realizar sin consumir excesivos recursos del sistema, como la captura y subida de la imagen al servidor de reconocimiento. (Gonzales Marin, 2016)

2.1.1. Adquisición de la imagen

Según (Bertino Salazar & Orozco Alzate) el proceso de adquisición de la imagen desde el objeto hasta su ingreso en el procesador, se logra a través de un dispositivo que debe ser sensible a los cambios de la luz, el cual es una cámara compuesta por elementos fotosensibles que transforman los fotones de la luz en una corriente eléctrica. En la adquisición de la imagen se tienen dos parámetros: resolución y formato.

La resolución, debe ser óptima y de buena calidad para poder conseguir un mejor resultado en el procesamiento de la imagen, ya que una baja calidad puede distorsionar los rostros y suprimir objetos. Así mismo, el formato recomendable es TIF y PNG, sin embargo, estas imágenes tienen un tamaño de almacenamiento más alto que otros formatos.

2.2. Amazon Rekognition

Como se ha mencionado en el apartado 2.1.1 existen servidores de reconocimiento, confiables que alivianan el trabajo del programador de sistemas y también el procesamiento en el cliente, ya que ofrecen APIs que permiten utilizar herramientas de reconocimiento facial a través de algoritmos desarrollados y probados por expertos en la materia.

Uno de los proveedores que brinda un servicio de servidor de reconocimiento facial, es Amazon Web Services con su producto Amazon Rekognition, el cual según (Amazon, Amazon Rekognition, 2018) permite identificar texto, objetos, personas y actividades mediante archivos de video o imagen que se encuentren almacenados en Amazon S3, y a su vez identificar contenido inapropiado.

Amazon Rekognition, hace uso de redes neuronales profundas, que se utilizan para detectar y etiquetar miles de objetos y escenas en las imágenes proporcionadas. Las características se detallan en la siguiente tabla:

Tabla 1: Características de Amazon Rekognition

Análisis de videos e imágenes eficientes.	Permiten solicitar una variedad de características, incluida la detección de objetos, el reconocimiento facial y el seguimiento de personas.
Aprendizaje continuo.	El servicio se entrena continuamente con datos nuevos para ampliar su capacidad de

	reconocimiento.
Completamente administrado.	Proporciona tiempos de respuesta estables independientemente del volumen de las solicitudes que realice.
Análisis en lote y en tiempo real.	Análisis en tiempo real en videos a partir de Amazon Kinesis Video Streams o analizar imágenes a medida que se carguen en Amazon S3.
Bajo costo.	Solo se paga por el número de imágenes o minutos de video que se analice y los datos faciales que se almacene para el reconocimiento facial.

Fuente: <https://aws.amazon.com/es/rekognition/>

2.2.1. Funcionamiento

Amazon Rekognition, realiza un análisis de imágenes de aprendizaje profundo fácil de usar y puede ser de dos tipos: imagen y video.

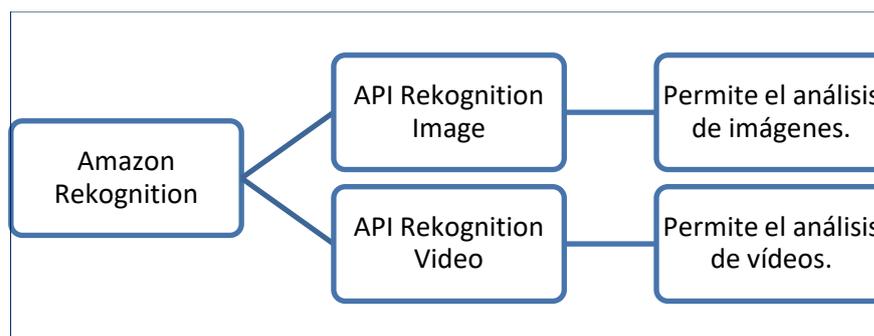


Figura 1: Tipos de reconocimiento Amazon Rekognition

Fuente: https://docs.aws.amazon.com/es_es/rekognition/latest/dg/how-it-works.html

Tipos de detección y reconocimiento

La detección y reconocimiento se puede realizar de varios tipos, estos son: etiquetas, rostros, búsqueda de rostros, personas, famosos y contenido no seguro. Para el desarrollo de la aplicación de este TFM se ha considerado el uso de rostros y búsqueda de rostros, que permiten identificar mediante una comparación en una base de datos de imágenes en Amazon S3, los rostros que son indicados en los parámetros de la búsqueda.

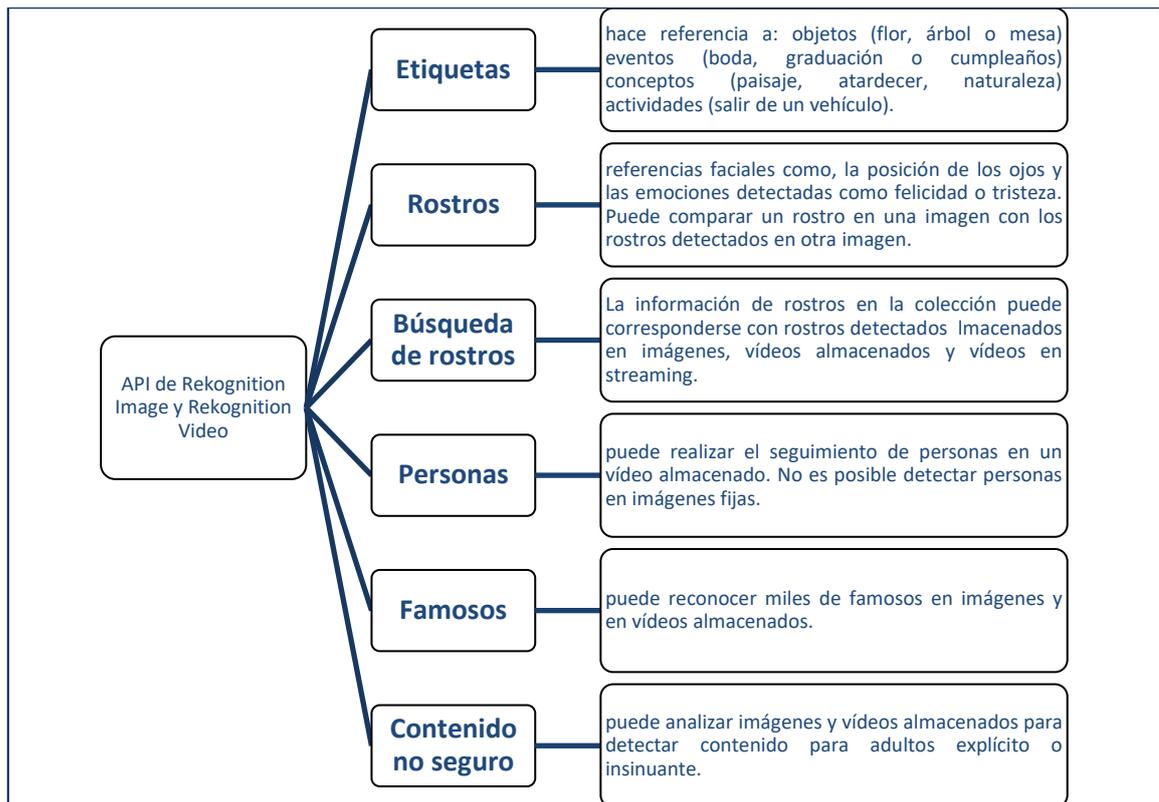


Figura 2: Tipos de detección y reconocimiento

Fuente: https://docs.aws.amazon.com/es_es/rekognition/latest/dg/how-it-works-types.html

Operaciones de Rekognition Image

Las operaciones de Amazon Rekognition Image son síncronas. La entrada y la respuesta se encuentran en formato JSON. Las operaciones de Amazon Rekognition Image analizan una imagen de entrada que está en formato de imagen .jpg o .png. La imagen se transfiere a una operación de Rekognition Image que puede almacenarse en un bucket de Amazon S3. Si no se utiliza la AWS CLI¹, también se puede transferir bytes de imágenes cifrados en base64 directamente a una operación de Amazon Rekognition. (Amazon, Amazon Rekognition, 2018)

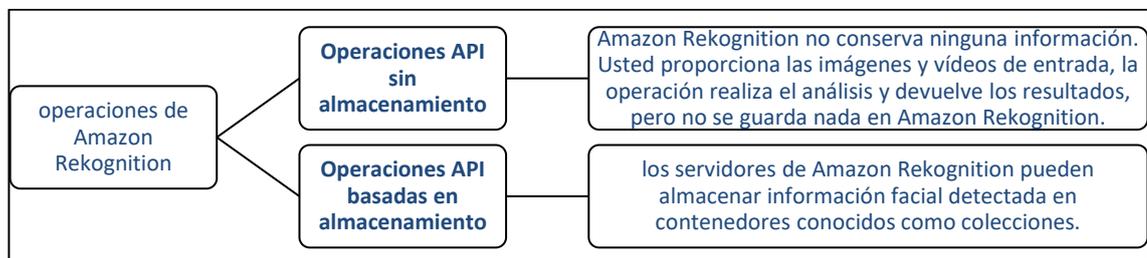


Figura 3: Operaciones API de almacenamiento y sin almacenamiento

Fuente: https://docs.aws.amazon.com/es_es/rekognition/latest/dg/how-it-works-storage-non-storage.html

¹ CLI: Command line interface, es un método que permite a los usuarios dar instrucciones a algún programa informático por medio de una línea de texto simple

2.3. Antecedentes

Existen en la actualidad sistemas que permiten realizar el reconocimiento facial para la toma de asistencia estudiantil. Los casos analizados a continuación utilizan diferentes formas para cumplir con dicho objetivo.

Estudiantes de la Universidad Veracruzana de México, propusieron mediante la ayuda de Matlab un registro de asistencia para estudiante utilizando reconocimiento facial, donde según esta investigación, los aspectos importantes que se deben considerar son: 1) el equipo se debe encontrar a una altura de 1.50 m.; 2) el estudiante se debe colocar atrás de una línea marcada en el suelo frente a la cámara, aproximadamente a unos 60 cm. de distancia; 3) los estudiantes deben ir pasando conforme van llegando, con la finalidad que el registro de asistencia se diera de forma natural y; 4) cada estudiante debe ver detenidamente a la cámara durante tres segundos, indicando mediante dos parpadeos de color verde que la fotografía fue tomada de forma correcta para su posterior análisis. Se obtiene la información que puede ser mostrada de diferentes maneras y a su vez generar reportes sobre los movimientos que ha realizado el usuario.

Para obtener resultados óptimos al momento de realizar el proceso de reconocimiento facial, se deben tener en consideración ciertos factores tales como:

- ✓ Luz apropiada al momento de realizar la fotografía, ya que tanto la ausencia como el exceso de la misma, puede provocar resultados incorrectos
- ✓ La altura y la distancia adecuada entre la cámara y las personas, la cual debe permitir obtener el área de interés que en este caso sería el rostro y
- ✓ El fondo, siendo el color blanco el más óptimo debido que permite una mejor extracción del rostro y la eliminación de ruido.

Un proyecto similar fue realizado en la Pontificia Universidad Católica de Valparaíso Chile, donde se desarrolló un sistema de reconocimiento facial mediante el uso de las librerías OpenCV/EmguCV para optimizar su eficiencia y rendimiento. Dispone a su vez de una base de datos local, la cual permite almacenar las fotografías tomadas a los usuarios para su posterior uso al momento de reconocer a una persona mediante la utilización de diferentes algoritmos y poder identificar así cual es el más eficiente. (Espinoza Olguin & Jorquera Guillen, 2015)

En el trabajo mencionado, se realizaron tres pruebas diferentes, la primera utilizando fotos tamaño carnet de 100x125, la segunda con fotos tomadas mediante una cámara digital con tres imágenes cada uno y la tercera con tres fotos tomadas (1 frontal y 2 con pequeñas inclinaciones a los costados), para evaluar la precisión de los algoritmos.

La Universidad de China en Pekín, cuenta con un sistema de reconocimiento facial que permite controlar la asistencia de estudiantes. El funcionamiento de este software consiste, en que antes de iniciar la clase, el docente activa el sistema instalado en una tableta y toma fotos de los estudiantes que pasan frente de la cámara, “uno por uno” para después comparar las fotos con las almacenadas en una base de datos. Dicho sistema se basa en la tecnología de reconocimiento facial de la plataforma de inteligencia artificial del gigante chino de Internet, Baidu. (EFE, 2017)

En el Ecuador, Estudiantes de la Universidad de Guayaquil realizaron una propuesta de un prototipo de sistema para el control de asistencia estudiantil de un paralelo mediante procesamiento digital de imágenes. Utilizaron el lenguaje de programación Python, la librería OpenCV, y para poder alimentar la base de datos realizaron 20 capturas por cada estudiante, para de esta forma obtener expresiones diferentes, considerando dos ambientes (mucha y poca iluminación) para poder comprobar la efectividad del algoritmo utilizado. (Hidalgo Veliz, 2017)

Mediante una laptop incorporada con cámara web activa, se captura la imagen del aula para que pueda identificar los rostros de los estudiantes y cuando el algoritmo detecte los rostros, se aplica técnicas de PDI² para mejorar la imagen transformándola a escala de grises y recortando la cara del alumno para poder ser comparada con la base de datos y lograr el control de asistencia.

Según el autor (Hidalgo Véliz, 2017) menciona que se recomienda un ambiente controlado, una correcta orientación de la cámara y la posición de la cara de los estudiantes en el salón, para lograr mayor precisión.

2.4. Conclusión del estado del arte

Como se puede evidenciar en los apartados anteriores existen en el mercado sistemas de este tipo, sin embargo, hacen uso de tecnologías de reconocimiento facial como

² PDI: Procesamiento de imágenes

OpenCV/EmguCV, y realizan la toma de asistencias “uno a uno”, es decir, que para marcar una asistencia, se debe posar frente de la cámara por un tiempo determinado y así cumplir con la funcionalidad del programa.

La idea del sistema a desarrollar, es evitar el registro “uno a uno” como se mencionó en párrafo anterior, y poder realizarlo de forma masiva; en otras palabras, el sistema solo deberá capturar una foto del salón de clases, identificar los rostros encontrados y registrar en la base de datos la asistencia de las personas encontradas en dicha imagen.

Para lograr la funcionalidad de control masivo de asistencias en el sistema a desarrollar, se hará uso de tecnologías de Cloud Computing y específicamente de Amazon Web Services con las herramientas: 1) Elastic Cloud computing EC2 (Servidor Tomcat, Base de Datos); 2) Simple Storage Service S3 (Almacenamiento de objetos) y 3) Amazon Rekognition (Procesamiento de imágenes); este último, evitará codificar un algoritmo de reconocimiento facial y tan solo se utilizará el servicio proporcionado, en otras palabras, se contará con un servidor de procesamiento facial, por medio del proveedor Amazon.

Este sistema también puede estar sujeto a inconvenientes, como la resolución de la cámara, factores naturales como exposiciones de luz, así mismo, la precisión y confianza del algoritmo utilizado para el reconocimiento de rostros, aunque pertenece a Amazon Web Services, el usarlo mediante la API requiere un sin número de configuraciones. Además, la fotografía tomada con la cámara mediante el sistema, debe ser subida a los servicios del Cloud para su posterior tratamiento, lo cual implica un retardo en el tiempo de registro. Para mitigar esto se puede disponer de una conexión a Internet con un ancho de banda apropiado exclusivo para el sistema, que serviría solo para subir la imagen de forma rápida, ya que el sistema estará desplegado en una instancia EC2 y el proceso de reconocimiento facial se realizaría directamente en los servidores e infraestructura de Amazon Web Services.

Finalmente, en el Ecuador este tipo de sistemas aún se encuentran en desarrollo y otros en fase de pruebas. En determinados casos, estos sistemas no ayudan al proceso sino que ralentiza actividades, ya que el uso de algoritmos desarrollados por los mismos programadores con herramientas de procesamiento de imágenes, tienden a ser lentos y en algunos casos con una precisión muy pobre, lo cual no permite cumplir con el objetivo de esta línea de sistemas. Es por eso que se apuesta a los beneficios de las herramientas de Cloud Computing, que actualmente están en auge y en constante desarrollo.

3. Objetivos concretos y metodología de trabajo

3.1. Objetivo general

Modelar e implementar una aplicación web que permita el control de asistencia estudiantil mediante reconocimiento facial, aplicado a la Unidad Educativa “Julia Navarrete Mendoza”, utilizando herramientas de Cloud Computing Amazon Web Services.

3.2. Objetivos específicos

- Optimizar el proceso de toma de asistencia estudiantil mediante reconocimiento facial utilizando servicios Cloud.
- Garantizar el acceso a la información de asistencia a través de una aplicación Web que sea intuitiva y fácil de usar, determinado por pruebas de cumplimiento de requisitos.
- Gestionar el registro de la información de los involucrados en el proceso de asistencia estudiantil y comunicaciones a padres de familia de la Unidad Educativa.
- Generar informes que ayuden a la toma de decisiones en los procesos educativos.

3.3. Metodología de desarrollo RUP (Proceso Unificado Racional)

Según (Rational, 2005), el Proceso Unificado Racional, es un proceso de ingeniería de software, que provee una disciplina aprovechada para asignar tareas y responsabilidades con una organización en el desarrollo. Su objetivo es garantizar la producción de un software de alta calidad que cumple con las necesidades de los usuarios finales, dentro de un cronograma y presupuesto predecible.

El RUP utiliza el lenguaje Unificado de Modelado (UML), para preparar todos los esquemas de un sistema software. De hecho, UML es una parte esencial del proceso unificado, sus desarrollos fueron en paralelo. (Jacobson, Booch, & Rumbaugh, 2000)

Características del RUP

- ✓ Dirigido por casos de uso
- ✓ Proceso centrado en la arquitectura
- ✓ Iterativo e incremental
- ✓ La captura de requisitos utiliza artefactos³, roles⁴ y flujos de trabajo

3.3.1. Fases del ciclo de vida

El ciclo de vida de RUP, está comprendido de cuatro fases de desarrollo, dentro de cada fase se realizan iteraciones que son variables según sea el alcance del proyecto. Además, las fases del RUP se encuentran asociadas a disciplinas que corresponden al flujo de proceso del trabajo; cada fase termina con un hito en el cual se verifican los objetivos que se han tenido que realizar. Cuando se finaliza cada hito, los directores deben tomar la decisión de si es conveniente seguir a la siguiente fase no avanzar y continuar con una nueva iteración en la fase actual. (Jacobson, Booch, & Rumbaugh, 2000)

Las fases de RUP son las siguientes:

- ✓ Inicio
- ✓ Elaboración
- ✓ Construcción
- ✓ Transición

³ Artefactos: casos de uso, actores, prototipos de interfaz de usuario

⁴ Roles: Funciones de personas en determinada área de una entidad

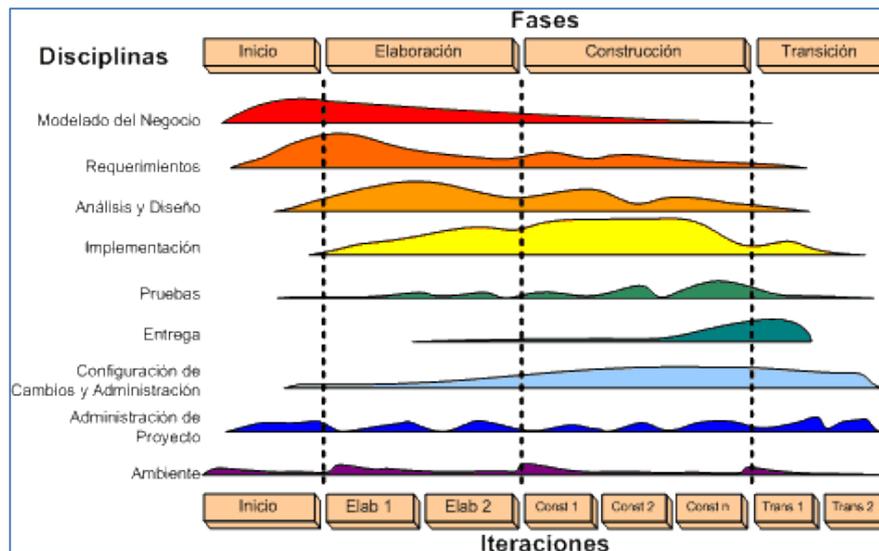


Figura 4: Fases y disciplinas de RUP

Fuente: El proceso unificado de software, Jacobson, Booch, & Rumbaugh, 2000

Inicio

En esta fase se desarrolla una descripción del producto final a partir de una buena idea presentando como resultado el análisis de negocio del producto. Se identifican los principales casos de uso y riesgo. (Jacobson, Booch, & Rumbaugh, 2000)

Como instrumento de desarrollo, se realizarán reuniones periódicas para la obtención de los requisitos con el interesado del sistema, y así, lograr una mejor comprensión y entendimiento de los objetivos que debe cumplir el producto.

Elaboración

En esta fase se recopilan la mayor parte de los requisitos que aún queden pendientes, formulando los requisitos funcionales como casos de uso, se establece una base de la arquitectura sólida que es la línea base para guiar el trabajo durante las fases de construcción y transición. Además, se eliminan los riesgos claves del proyecto. (Jacobson, Booch, & Rumbaugh, 2000)

Para el desarrollo de las actividades de esta fase, se tomará como base la especificación de requisitos según el estándar IEEE 830.

Construcción

La fase de construcción incluye diseño, programación y pruebas del sistema. Partes del sistema se desarrollan en paralelo y se integran durante esta fase. Al completar ésta, debe tenerse un sistema de software funcionando y la documentación relacionada y lista para entregarse al usuario. (Sommerville, 2011)

Para esta fase, se utilizarán las herramientas: 1) Vaadin que permitirá a creación de interfaces web mediante programación en el lenguaje Java; 2) MySQL para el almacenamiento de la información.

Transición

Fase en la cual ya se cuenta con un producto en versión beta. En esta versión del sistema varios usuarios prueban el producto e informan sobre fallos e inconsistencias. El desarrollador corrige los problemas encontrados e incorpora funcionalidades sugeridas. Los defectos que puede tener el producto se pueden clasificar en dos grupos: los que tienen suficiente impacto en la operación y los que pueden corregirse en una nueva versión normal (Jacobson, Booch, & Rumbaugh, 2000). Como punto final, se entrega el producto en su última versión con su respectivo testeo.

Para comprobar la validez del sistema, se implementará un test de usabilidad que permitirá conocer si hay fallas respecto a las interfaces de usuario. También, se evaluará el cumplimiento de los requisitos para verificar el correcto funcionamiento del sistema.

4. Inicio

En este apartado correspondiente a la fase de inicio de la metodología RUP, se procederá a definir los requisitos del sistema a desarrollar, estableciendo el alcance, la descripción general y el estudio de factibilidad, recordando que el software a desarrollar es un sistema de control de asistencia estudiantil mediante reconocimiento facial, para la Unidad Educativa Julia Navarrete Mendoza.

4.1. Introducción

La fase de inicio, permite establecer hasta donde va a llegar el sistema y que procesos va a cubrir; para este proyecto se tomará en cuenta el análisis de los requisitos funcionales y no funcionales mediante el estándar IEEE 830-1998. Además, se pretende establecer el alcance y verificar si SISCAE como se nombrará de ahora en adelante al sistema de control de asistencias estudiantil mediante reconocimiento facial, es factible en su realización, de acuerdo a los recursos y a las herramientas disponibles para su desarrollo.

4.1.1. Alcance

El “sistema de control de asistencia estudiantil mediante reconocimiento facial”, permitirá llevar el registro y administración de las asistencias de los estudiantes automáticamente utilizando las técnicas de reconocimiento facial. Además, se permitirá el registro de los estudiantes, docentes, padres de familia y documentos (fichas de atraso, comunicaciones) necesarios para el cumplimiento del proceso de asistencias. También el usuario, podrá generar los reportes necesarios para su posterior legalización.

El proceso finalizará con el despliegue de la aplicación web, en un servicio de Cloud llamado Elastic Cloud Computed (EC2) de Amazon Web Services (AWS), incluyendo los siguientes entregables:

- ✓ Requerimientos funcionales
- ✓ Requerimientos no funcionales
- ✓ Diagramas y plantillas de casos de uso
- ✓ Diagrama de base de datos
- ✓ Diagrama de secuencia
- ✓ Interfaces de la aplicación
- ✓ Pequeña guía del uso del aplicativo

4.1.2. Definición de acrónimos y abreviaturas

Tabla 2: Glosario de términos.

Nombre	Descripción
UE JUNAME	Unidad Educativa “Julia Navarrete Mendoza”, institución donde se implementará la aplicación
SISCAE	Sistema de control de asistencia estudiantil
ERS	Especificación de requisitos del software
RF	Requisito funcional
RFN	Requisito no funcional
HTTP	Protocolo de transporte de hipertexto
AWS	Amazon Web Services
S3	Simple Storage Services
EC2	Elastic Cloud Computing
Usuario	Persona que usará el sistema para gestionar procesos
CLI	Command line interface
HIBERNATE	Herramienta de Mapeo objeto-relacional (ORM) para la plataforma Java
JPA	API de persistencia desarrollada para la plataforma Java EE
MySql	Base de datos
SGDB	Sistema gestor de base de datos

Fuente: Elaboración propia

4.1.3. Referencias

Tabla 3: Referencia de estándar para desarrollo de software.

Título del documento	Referencia
Standard IEEE 830-1992	IEEE

Fuente: Elaboración propia

4.2. Descripción general del sistema

El sistema a desarrollar, tiene como finalidad el control de asistencia estudiantil mediante reconocimiento facial; para conocer en detalle la funcionalidad del mismo, se indica una descripción general con sus respectivas restricciones, factores supuestos y dependencias que pueden afectar el desarrollo, todo esto sin entrar al mínimo detalle. Además, se especifican detalladamente los requisitos necesarios para que el sistema cumpla con su objetivo principal.

	administración de usuarios, gestión de cursos y paralelos, periodos lectivos. Control total de SISCAE
--	---

Fuente: Elaboración propia

Tabla 5: Característica de usuario secretaria

Tipo de usuario	Secretaria
Formación	Administración de procesos educativos
Actividades	Autenticarse en el sistema, generación de reportes de asistencias, registro de estudiantes y padres de familia, envío de notificaciones a correos, registro de nuevos docentes y gestión de cursos.

Fuente: Elaboración propia

Tabla 6: Característica de usuario docente

Tipo de usuario	Docente
Formación	Conocimiento básico en informática, manejo de Internet y aplicaciones.
Actividades	Autenticarse en el sistema, realizar el registro de asistencias estudiantiles, sea manual o automáticamente (Reconocimiento facial),

Fuente: Elaboración propia

4.2.4. Restricciones

- El framework de desarrollo VAADIN.
- Lenguajes y tecnologías en uso: JAVA.
- El sistema se diseñará según un modelo cliente/servidor.
- Dificultades para el reconocimiento facial.
- El sistema deberá ser desarrollado de una manera sencilla y de fácil uso por parte de la secretaria y docentes.
- El sistema será accedido a través de la Web mediante un navegador.
- Herramientas de Cloud.
- Cada usuario del sistema dispondrá de una contraseña única que permitirá la autenticación y autorización.

4.2.5. Suposiciones y dependencias

- Se asume que los requisitos establecidos por los stakeholders involucrados durante el desarrollo del sistema, sean estables y no muy cambiantes en su contexto.

- La infraestructura cloud donde se desplegará el sistema, deberá ofrecer toda su capacidad técnica y operativa para garantizar la disponibilidad de la información cuando se solicite.
- El algoritmo de reconocimiento facial proporcionado por el proveedor de AWS es seguro y confiable.
- El equipo fotográfico toma fotos de buena calidad.
- Disponibilidad de tiempo para el desarrollo e implementación.
- El sistema SISCAE, depende del servidor de reconocimiento facial proporcionado por AWS.

4.2.6. Requisitos futuros

Como desarrollo futuro, se tiene pensado implementar una aplicación móvil para el uso de cada docente, en la cual se incluya el proceso de toma de asistencias mediante reconocimiento facial, y así aprovechar el hardware de estos dispositivos, con la finalidad de que si no hay una computadora y cámara en cada salón de clases, exista otra alternativa de hacerlo.

4.3. Especificación de requisitos

4.3.1. Requisitos funcionales

Tabla 7: Requisito funcional gestionar usuario

Identificación del requisito	RF01
Nombre del requisito	Gestionar usuario
Descripción del requisito	El sistema permitirá al usuario administrador dar de alta a los usuarios que harán uso del sistema. Además, permitirá consultar información de dichos usuarios.
Requisito no funcional	RNF01 - RNF02 - RNF03 - RNF04
Prioridad del requisito	Alta

Fuente: Elaboración propia

Tabla 8: Requisito funcional autenticar y autorizar usuario

Identificación del requisito	RF02
Nombre del requisito	Autenticar y autorizar usuario
Descripción del requisito	El sistema permitirá a los usuarios identificarse para dar autorización a las

	opciones o módulos del sistema que le corresponden.
Requisito no funcional	RNF01 - RNF02 - RNF03 - RNF04
Prioridad del requisito	Alta

Fuente: Elaboración propia

Tabla 9: Requisito funcional gestionar estudiante

Identificación del requisito	RF03
Nombre del requisito	Gestionar estudiante
Descripción del requisito	El sistema permitirá al usuario (secretaria y administrador) registrar, modificar y eliminar la información de los estudiantes en la base de datos.
Requisito no funcional	RNF01 - RNF02 - RNF03 - RNF04
Prioridad del requisito	Media

Fuente: Elaboración propia

Tabla 10: Requisito funcional gestionar padre de familia

Identificación del requisito	RF04
Nombre del requisito	Gestionar padre de familia
Descripción del requisito	El sistema permitirá al usuario (secretaria y administrador) registrar, modificar y eliminar la información de los padres de familia en la base de datos.
Requisito no funcional	RNF01 - RNF02 - RNF03 - RNF04
Prioridad del requisito	Media

Fuente: Elaboración propia

Tabla 11: Requisito funcional gestionar cursos

Identificación del requisito	RF05
Nombre del requisito	Gestionar cursos
Descripción del requisito	El administrador del sistema, podrá definir los cursos y los respectivos paralelos correspondientes a esos cursos, esto es de importancia para que los estudiantes puedan ser inscritos en un paralelo
Requisito no funcional	RNF01 - RNF02 - RNF03 - RNF04
Prioridad del requisito	Alta

Fuente: Elaboración propia

Tabla 12: Requisito funcional gestionar asistencia estudiantil

Identificación del requisito	RF06
Nombre del requisito	Gestionar asistencia estudiantil
Descripción del requisito	El sistema permitirá al usuario (docente), registrar la respectiva asistencia a clases de los estudiantes de un determinado curso. Este registro se puede realizar de forma manual o automática (reconocimiento facial)
Requisito no funcional	RNF01 - RNF02 - RNF03 - RNF04 – RNF05
Prioridad del requisito	Media

Fuente: Elaboración propia

Tabla 13: Requisito funcional gestionar reportes

Identificación del requisito	RF07
Nombre del requisito	Gestionar reportes
Descripción del requisito	El usuario secretaria, podrá generar los reportes de asistencias parciales para posteriormente ser validados y legalizados
Requisito no funcional	RNF02 - RNF03 - RNF04
Prioridad del requisito	Media

Fuente: Elaboración propia

Tabla 14: Requisito funcional gestionar comunicaciones

Identificación del requisito	RF08
Nombre del requisito	Gestionar comunicaciones
Descripción del requisito	El sistema permitirá, al usuario (secretaria), el envío de comunicaciones a los padres de familia en caso de que el estudiante tenga faltas por inasistencia a clases.
Requisito no funcional	RNF02 - RNF03 - RNF04
Prioridad del requisito	Media

Fuente: Elaboración propia

Tabla 15: Requisito funcional gestionar docente

Identificación del requisito	RF09
Nombre del requisito	Gestionar docente
Descripción del requisito	El sistema permitirá, al usuario (administrador, secretaria), el registro de los

	docentes con sus respectivos cursos y materias.
Requisito no funcional	RNF02 - RNF03 - RNF04
Prioridad del requisito	Media

Fuente: *Elaboración propia*

4.3.2. Requisitos no funcionales

Para realizar el análisis de los requisitos no funcionales se tomarán los requisitos indicados en el libro de Ingeniería de Software de Sommerville, que se muestran en la siguiente figura.

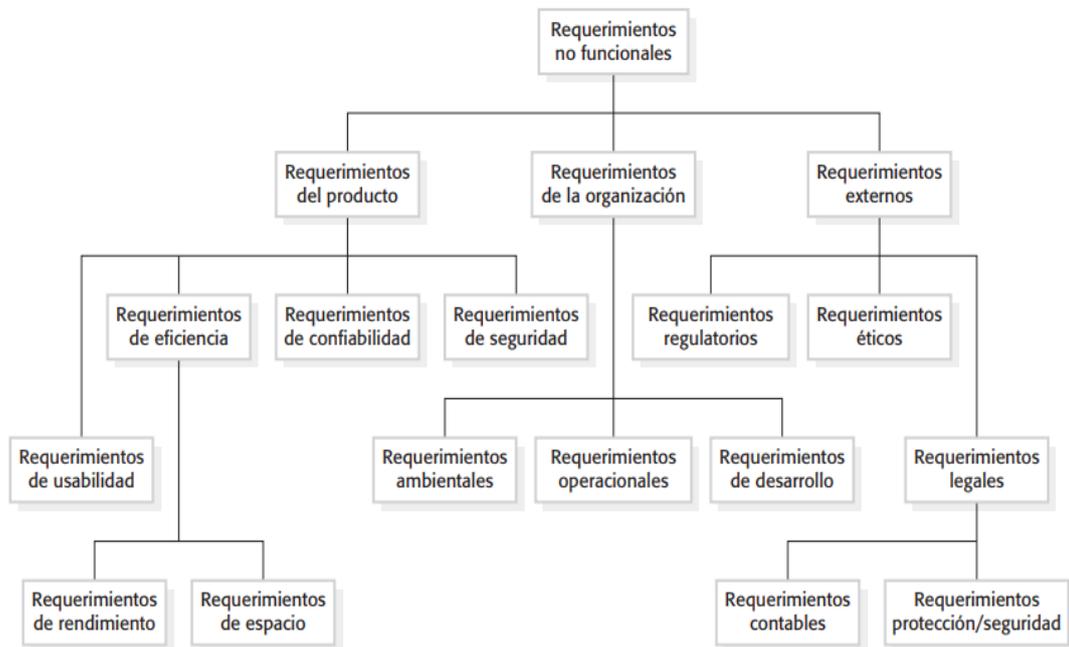


Figura 6: Tipos de requerimientos no funcionales

Fuente: Somerville 2011

Tabla 16: Requisito no funcional espacio y rendimiento

Identificación del requisito	RNF01
Nombre del requisito	Espacio y rendimiento
Descripción del requisito	El tamaño total del sistema a desplegar en el servidor Tomcat es de 20 Mb aproximadamente y la base de datos es de 1 Mega. Además, se contabilizan las fotos tomadas a los estudiantes para el entrenamiento de los algoritmos de reconocimiento facial, estas imágenes van a estar almacenadas en Amazon S3. Cada vez que el sistema tome la fotografía para reconocer los rostros encontrados, se subirá al

	Cloud para su posterior análisis. Respecto al rendimiento, se espera un tiempo de respuesta del sistema de 3 segundos.
Prioridad del requisito	Alta

Fuente: Elaboración propia

Tabla 17: Requisito no funcional usabilidad

Identificación del requisito	RNF02
Nombre del requisito	Usabilidad
Descripción del requisito	El framework Vaadin, permite que los componentes gráficos del sistema sean de tipo responsive. Sin embargo, se recomienda el uso de navegadores como Google Chrome y Mozilla Firefox
Prioridad del requisito	Alta

Fuente: Elaboración propia

Tabla 18: Requisito no funcional seguridad

Identificación del requisito	RNF03
Nombre del requisito	Seguridad
Descripción del requisito	<p>El sistema deberá autenticar y autorizar a los usuarios en las opciones que le corresponden, mediante un nombre de usuario y contraseña.</p> <p>Se necesitará mecanismos de encriptación para cifrar el contenido de usuario y clave, además, la autenticación será mediante un formulario web utilizando un cifrado SHA – 256</p> <p>Para salvaguardar la integridad de información de la base de datos, se realizarán backups diarios.</p> <p>Los padres de familia, serán informados de la asistencia a clases de sus hijos mediante correos electrónicos.</p>
Prioridad del requisito	Alta

Fuente: Elaboración propia

Tabla 19: Requisito no funcional portabilidad

Identificación del requisito	RNF04
Nombre del requisito	Portabilidad
Descripción del requisito	El sistema solo necesitará un navegador web para su ejecución, y será accesible desde cualquier dispositivo que cumpla con este requerimiento.
Prioridad del requisito	Alta

Fuente: Elaboración propia

Tabla 20: Requisito no funcional operacional

Identificación del requisito	RNF05
Nombre del requisito	Operacional
Descripción del requisito	Para desplegar el sistema es necesario un servidor de aplicaciones Java Tomcat, la máquina virtual de Java, un servidor de reconocimiento facial(Amazon Rekognition) y una base de datos de objetos(Amazon S3)
Prioridad del requisito	Alta

Fuente: Elaboración propia

4.4. Estudio de factibilidad

4.4.1. Humana

El usuario administrador, secretaria y docente, cuentan con conocimientos básicos en informático y manejo del Internet, por lo cual el uso del sistema no se complicará. Además, se dispondrá de un pequeño manual en línea donde se explica el uso del sistema.

4.4.2. Legal

La Unidad Educativa, dispone de servicios de aula virtual y siempre está innovando tecnológicamente, por lo cual este sistema a implementar dará valor agregado a los procesos educativos realizados por el personal docente y no hay ninguna restricción legal que impida su realización.

4.4.3. De sistema

Para llevar a cabo el desarrollo e implementación del sistema SISCAE, es necesario realizar el estudio de viabilidad del sistema, lo cual permitirá determinar la capacidad técnica, operativa, económica, tecnológica y operacional del mismo, con el propósito de conocer si es o no factible realizar este desarrollo.

4.4.3.1. Técnica

Los conocimientos necesarios para la implementación de esta aplicación son:

- ✓ Programación Orientada a Objetos.
- ✓ Lenguaje de programación Java.
- ✓ Framework de desarrollo web Vaadin
- ✓ IDE Eclipse oxygen.
- ✓ SGDB MySql.
- ✓ Conocimientos en Cloud Computing.
- ✓ Manejo de Amazon Web services.
- ✓ APIs de herramientas de cloud (Amazon EC2, Rekognition, S3)

4.4.3.2. Operativa

La aplicación estará disponible para todas las personas involucradas. También, será sencilla de utilizar, gracias a la interfaz de usuario intuitiva que lo guiará durante su uso, obteniendo una rápida comprensión que cubra todas las expectativas de los usuarios.

4.4.3.3. Tecnológica

Se realizó un análisis del equipo y la tecnología necesaria para el desarrollo e implementación del sistema SISCAE y se los clasificó en dos enfoques hardware y software.

El hardware con el que se dispone para el desarrollo de esta aplicación es:

- ✓ 1 Computador marca Toshiba, cuyas características son
 - **Marca:** Toshiba, Satellite
 - **Procesador:** Intel Core I5 – 64 bits
 - **Velocidad:** 2.40 Ghz
 - **Memoria RAM:** 6 GB
 - **Disco:** 750 GB

El software que se usará para el desarrollo de la aplicación es:

- ✓ IDE Eclipse Oxygen
- ✓ SGDB MySql
- ✓ JDK 7.0
- ✓ SDK Amazon Rekognition y S3
- ✓ Framework Vaadin
- ✓ Servidor Web Apache Tomcat versión 7.0
- ✓ Servicios Cloud AWS

4.4.3.4. Operacional

De acuerdo con la Factibilidad Operativa y Tecnológica, la Unidad Educativa facilitará el acceso a la información y los datos que disponen, los mismos que serán necesarios durante el proceso de desarrollo del sistema de control de asistencia estudiantil mediante reconocimiento facial.

5. Elaboración

5.1. Entorno del software

Se debe de tomar en cuenta que la institución es una Unidad Educativa privada cuya ubicación es en Manabí – Portoviejo específicamente en la calle 12 de Marzo y Av. Rocafuerte. El sistema a implementar estará desarrollado en entorno web, y desplegado en una instancia de EC2 de AWS, para que pueda ser accedido desde cualquier ubicación y el único requisito para su funcionamiento, es que se cuente con acceso a Internet y un navegador web.

Aunque el sistema es un prototipo a implementar, el equipo de desarrolladores debe de tener la mayor comunicación posible con las personas involucradas de la institución (director, docentes, secretaria, estudiantes, otros), para así recabar la mayoría de los requisitos necesarios y así evitar problemas en el desarrollo y validación del sistema.

5.2. Arquitectura candidata

Para el desarrollo del “Sistema de Control de Asistencia Estudiantil mediante Reconocimiento Facial”, se ha considerado utilizar la arquitectura cliente/servidor considerando el modelo de **cuatro capas**, además, de que es un estándar de las aplicaciones Webs desarrolladas utilizando como lenguaje de programación JAVA.

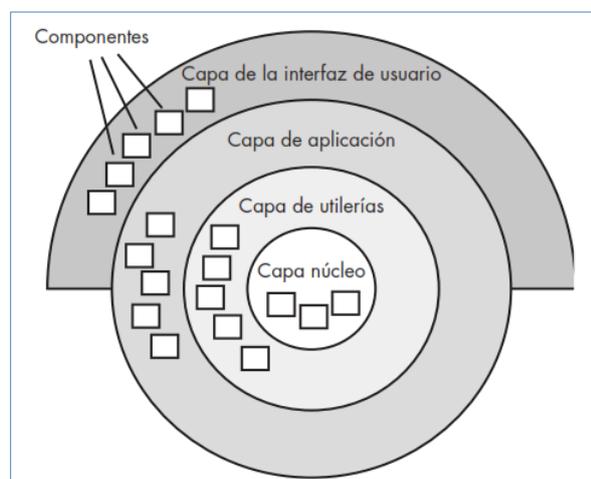


Figura 7: Arquitectura en capas

Fuente: Ingeniería de software, Pressman 2010

Según (Pressman, 2010), cada capa ejecuta operaciones diferentes que se aproximan progresivamente al conjunto de instrucciones de la máquina (sistema operativo). En la capa externa, los componentes atienden las operaciones de la interfaz de usuario. En la interna, los componentes realizan la interfaz con el sistema operativo. Las capas intermedias proveen servicios de utilerías y funciones de software de aplicación.

Para tener una visión más amplia del diseño arquitectónico, en el apartado 5.6 se presentará la arquitectura general y específica del sistema a desarrollar.

5.3. Herramientas tecnológicas a utilizar en el desarrollo

Las herramientas presentadas y descritas a continuación, fueron utilizadas para el desarrollo del sistema de control de asistencia estudiantil usando reconocimiento facial. Se seleccionaron dichas herramientas, porque se adaptan a los requisitos del Software para la necesidad del problema planteado, por la facilidad de aprendizaje, confianza y alto nivel de uso y demanda en el mercado, también, por recomendación de docentes especializados en el tema y el conocimiento adquirido durante el curso del máster.

5.3.1. Vaadin framework

Vaadin Framework, es un Framework UI de Java, que simplifica el desarrollo de aplicaciones web. El código es escrito en Java y ejecutado en un servidor JVM (Tomcat, glassfish), mientras la UI se renderiza como HTML5 en el navegador web. El framework también automatiza la comunicación entre el cliente (navegador) y servidor. (Vaadin, 2018)

5.3.2. Eclipse IDE

Eclipse es un entorno de desarrollo integrado, que permite desarrollar aplicaciones en Java, mismo lenguaje en el que se encuentra desarrollado. Además, permite la integración de lenguajes como: C/C++ y php. Está basado en plugins instalables desde su Marketplace que proporciona la implementación de otras funcionalidades.

5.3.3. Power Designer

Es una herramienta de modelado de datos de software, que permite crear modelos de arquitectura mientras se visualiza el impacto del cambio de requisitos antes de que suceda. Además, se puede especificar automáticamente las capas y requisitos arquitectónicos;

cuenta con un potente repositorio de metadatos alimentado por una comunidad empresarial dedicada al desarrollo de software. (SAP, s.f.)

5.3.4. Apache Tomcat

El software Apache Tomcat, es un servidor de aplicaciones, que ejecuta servlets y los representa como páginas webs que incluyen la codificación de Java. Descrito por Apache Software Foundation (2017), como una implementación de código abierto de Java Servlets, Java Server Pages, Java Expression Language y Java WebSocket que se encuentran desarrolladas bajo Java Community Process.

5.3.5. MySQL

MySQL, es el sistema de administración de base de datos SQL de código abierto más popular, desarrollado, distribuido y respaldado por Oracle Corporation. Permite la creación de base de datos relacionales, funciona en sistemas cliente/servidor o integrados, además, es multiplataforma y muy sencillo de utilizar. (Mysql, 2018)

5.3.6. Amazon web services

Amazon Web Services (AWS) es una plataforma de servicios en la nube que ofrece potencia de cómputo, almacenamiento de bases de datos, entrega de contenido y otras funcionalidades para ayudar a las empresas a escalar y crecer. Permite el despliegue de infraestructura, crear aplicaciones sofisticadas flexibles, escalables y fiables. (Amazon, Cloud Computing con Amazon Web Services, 2018)

5.3.6.1. Elastic cloud computing EC2

Amazon (2018), define a EC2 como un servicio web que proporciona capacidad informática en la nube segura y de tamaño modificable. Está diseñado para facilitar a los desarrolladores el uso de la informática en la nube a escala de la Web.

5.3.6.2. Amazon Simple Storage Service S3

Amazon S3 es un servicio de almacenamiento de objetos creado para almacenar y recuperar cualquier volumen de datos desde cualquier ubicación: sitios web y aplicaciones móviles, aplicaciones corporativas y datos de sensores o dispositivos IoT. Está diseñado para ofrecer una durabilidad del 99,999999999% y almacena datos para millones de

aplicaciones utilizadas por líderes de mercados de todas las industrias. (Amazon, Amazon S3, 2018)

5.3.6.3. Amazon Rekognition

Amazon Rekognition facilita la incorporación del análisis de imágenes y videos a sus aplicaciones. Suministrando una imagen o video a la API de Rekognition, el servicio identificará objetos, personas, texto, escenas y actividades, además de detectar contenido inapropiado. Amazon Rekognition también ofrece reconocimiento y análisis facial con un alto nivel de precisión. Se puede detectar, analizar y comparar rostros para una amplia variedad de casos de uso de verificación de usuarios, catalogación, contabilización de personas y seguridad pública. (Amazon, Amazon Rekognition, 2018)

5.4. Participantes en el desarrollo

Roles y responsabilidades:

Tabla 21: Matriz de roles y responsabilidades

CARGO	DESCRIPCIÓN	RESPONSABLE
Jefe del proyecto	Asigna los recursos, gestiona tareas prioritarias, coordina las interacciones con los stakeholders, mantiene al equipo del proyecto enfocado en los objetivos.	Ing. Gabriel Salvatierra
Analista de Sistemas	Captura, especifica y valida requisitos interactuando con el usuario mediante entrevistas. Elabora un modelo de análisis y diseño y colabora en la elaboración de las pruebas funcionales y modelos de datos.	Ing. Gabriel Salvatierra
Desarrollador	Se encarga del desarrollo del sistema de gestión, de acuerdo a los requerimientos funcionales y a validaciones con el usuario.	Ing. Gabriel Salvatierra
Ingeniero de Software	Gestiona los requisitos, elabora el modelo de datos, prepara las pruebas funcionales y elabora la documentación.	Ing. Gabriel Salvatierra
DBA	Tiene la responsabilidad de mantener y operar la base de datos de la Unidad Educativa Julia Navarrete Mendoza	Ing. Gabriel Salvatierra
Tester	Encargado de realizar la pruebas	Ing. Katherine Mendoza Docentes y secretaria de la

		Unidad Educativa
--	--	------------------

Fuente: Elaboración propia

Todas las pruebas serán realizadas una vez finalizada cada iteración de la fase de construcción de acuerdo a: el plan de pruebas, la descripción de los casos de uso y los diagramas de secuencias; estos artefactos permitirán establecer si se ha cumplido con los requisitos del sistema.

Las pruebas iniciales, se realizarán en paralelo cuando se finalice la codificación de componentes y se obtenga módulos funcionales de la aplicación y posteriormente serán testeados por los usuarios que usarán el sistema.

5.5. Riesgos técnicos

Los riesgos técnicos amenazan la calidad del software, así mismo, se ve afectada la planificación temporal y a largo plazo del ciclo de desarrollo. Los principales riesgos identificados en el análisis y despliegue del sistema SISCAE son:

- ✓ **Diseño:** La interpretación incorrecta de los requisitos pueden ocasionar que el modelado no sea el adecuado y se tenga inconvenientes en la fase de construcción.
- ✓ **Implementación:** La configuración por defecto no segura, puede provocar vulnerabilidades que afecten la integridad de los datos.
- ✓ **Interfaz de usuario:** La UI se puede tornar compleja y poco amigable ante los usuarios.

5.6. Plan de pruebas

El plan de pruebas establecido para el sistema de control de asistencias por reconocimiento facial a desarrollar, está indicado y evaluado por cada caso de uso, el cual enuncia los requisitos funcionales de cada módulo y componente del sistema. Por lo tanto, se creará una ficha de prueba por cada caso de uso que indique un proceso funcional.

Además, se evaluará el tiempo de subida y análisis de la imagen utilizada para reconocer los rostros de los estudiantes, de la misma forma, la calidad de servicio del servidor de reconocimiento facial.

5.7. Arquitectura a implementar

Como se especificó en el apartado 5.2, Java posee un estándar para la arquitectura de aplicaciones web, por tal motivo y tomando en cuenta este estándar la arquitectura a implementar será la de cuatro capas. Además, (Sommerville, 2011) señala que, las nociones de separación e independencia son fundamentales para el diseño arquitectónico porque permiten localizar cambios.

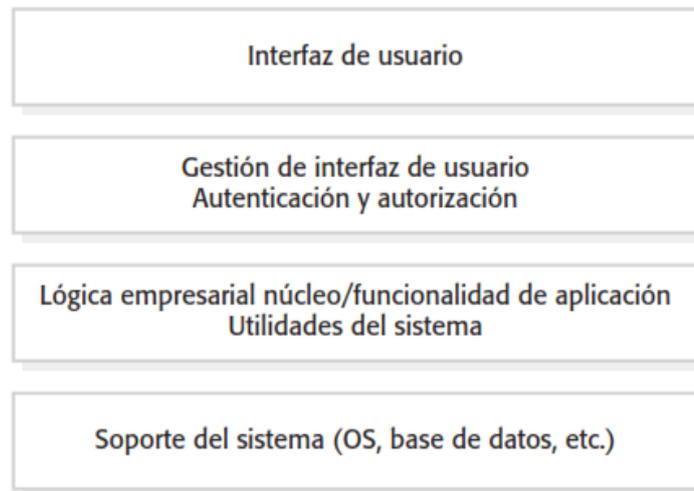


Figura 8: Arquitectura genérica en 4 capas
Fuente: Ingeniería de software, Sommerville 2011

Para conocer el funcionamiento de forma general de la aplicación, se presenta un diseño arquitectónico de los procesos que realizaría el sistema.

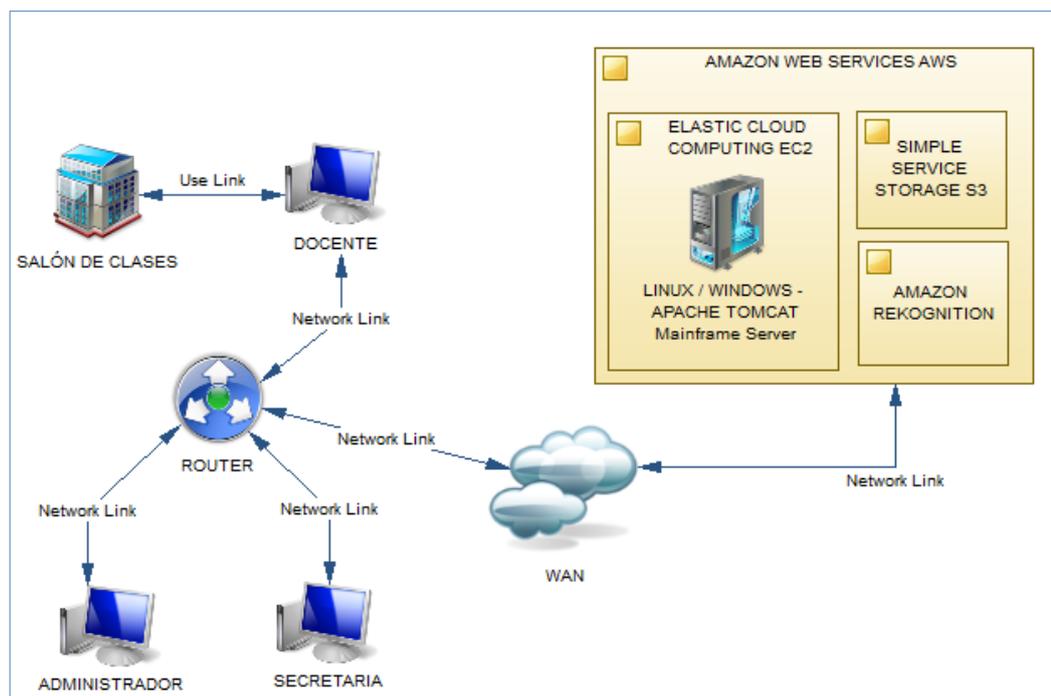


Figura 9: Diseño arquitectónico del sistema SISCAE

Fuente: Elaboración propia

5.8. Vista de casos de uso

5.8.1. Diagrama de casos de uso

Diagrama general

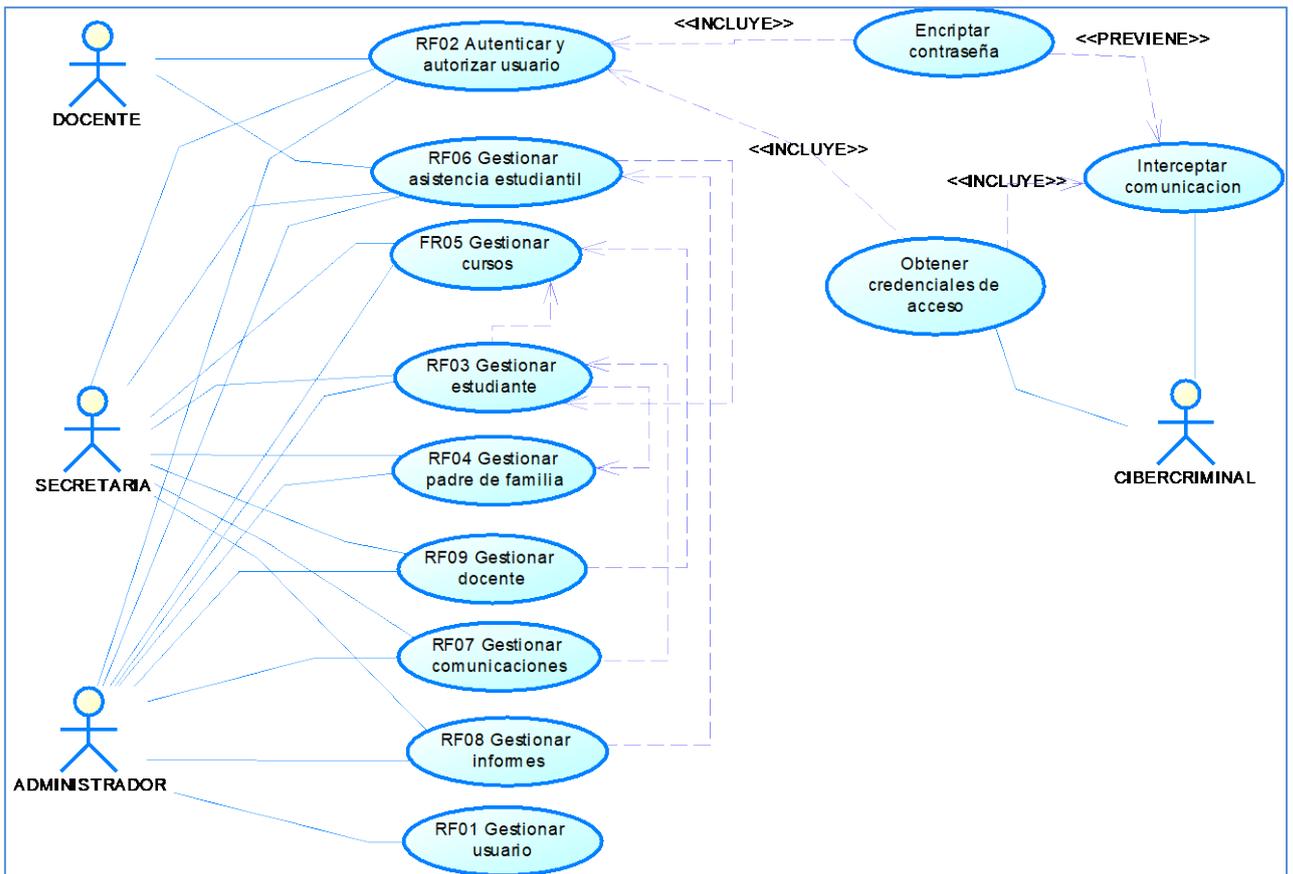


Figura 10: Caso de uso, diagrama general
Fuente: Elaboración propia

Gestionar usuario

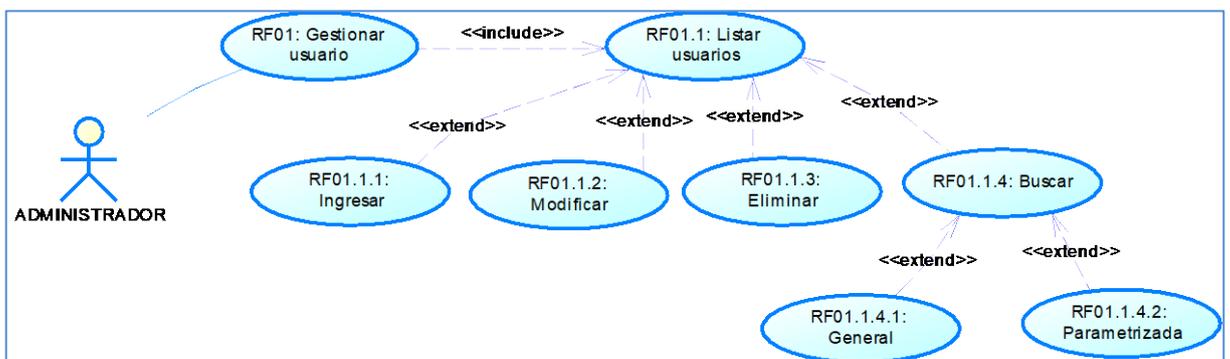


Figura 11: Caso de uso, gestionar usuario
Fuente: Elaboración propia

Autenticar y autorizar usuario

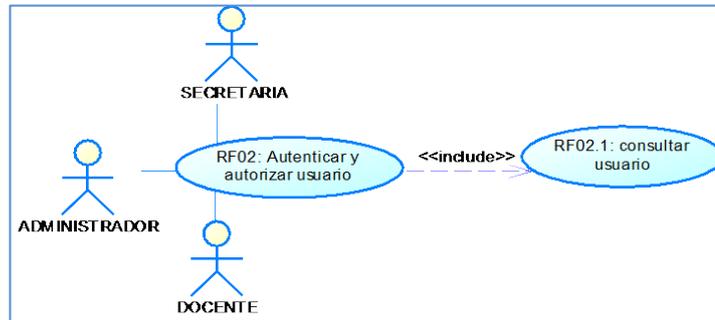


Figura 12: Caso de uso, autenticar y autorizar usuario
Fuente: Elaboración propia

Gestionar estudiante

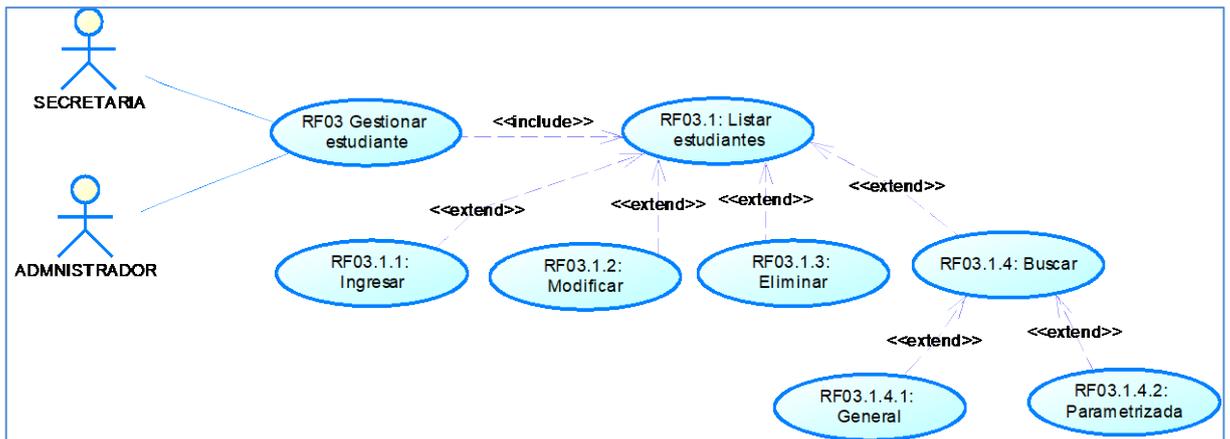


Figura 13: Caso de uso, gestionar estudiantes
Fuente: Elaboración propia

Gestionar padre de familia

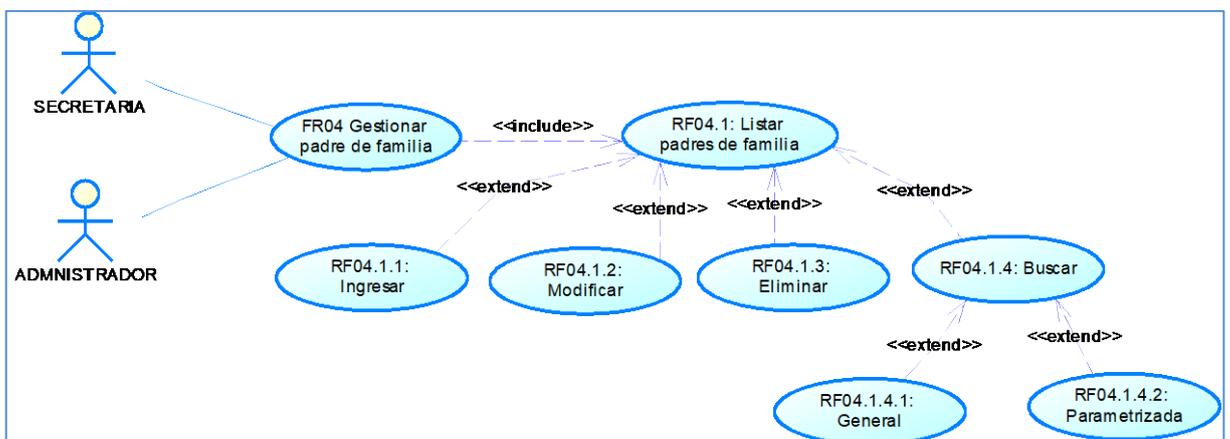


Figura 14: Caso de uso, gestionar padre de familia
Fuente: Elaboración propia

Gestionar curso

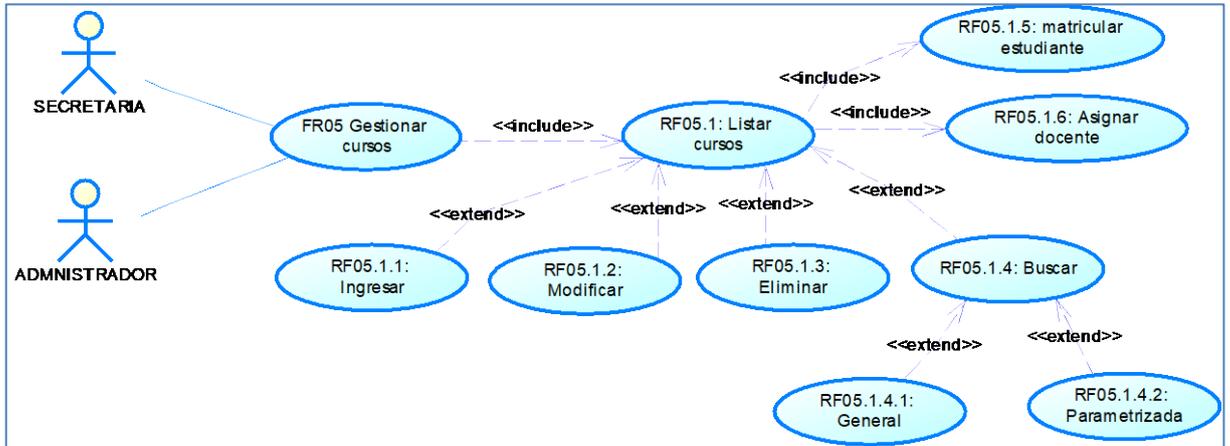


Figura 15: Caso de uso, gestionar curso
Fuente: Elaboración propia

Gestionar asistencia estudiantil

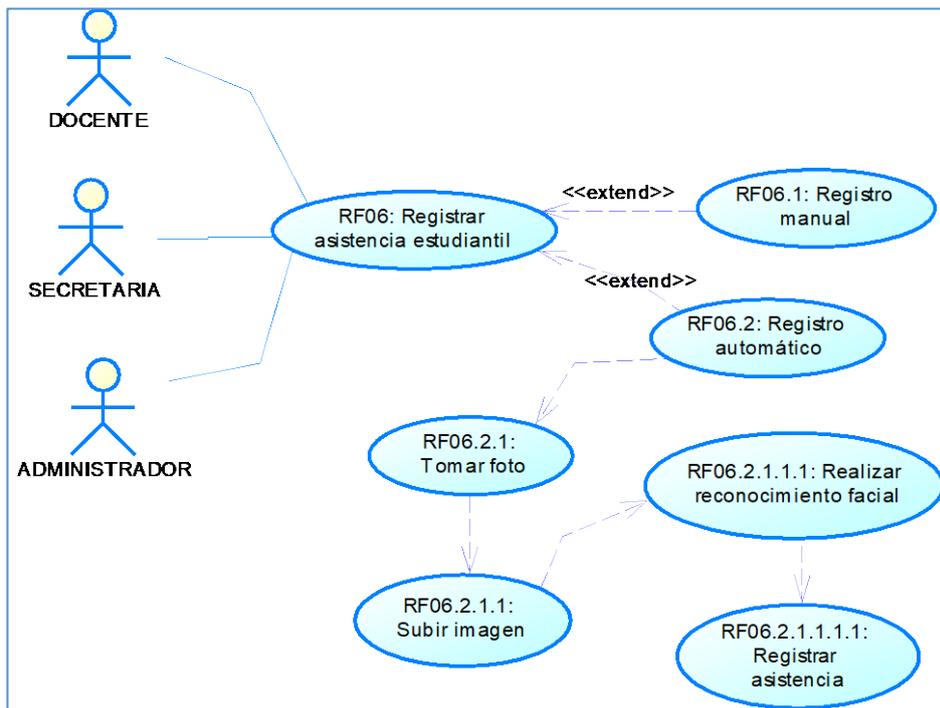


Figura 16: Caso de uso, gestionar asistencia estudiantil
Fuente: Elaboración propia

Gestionar comunicaciones

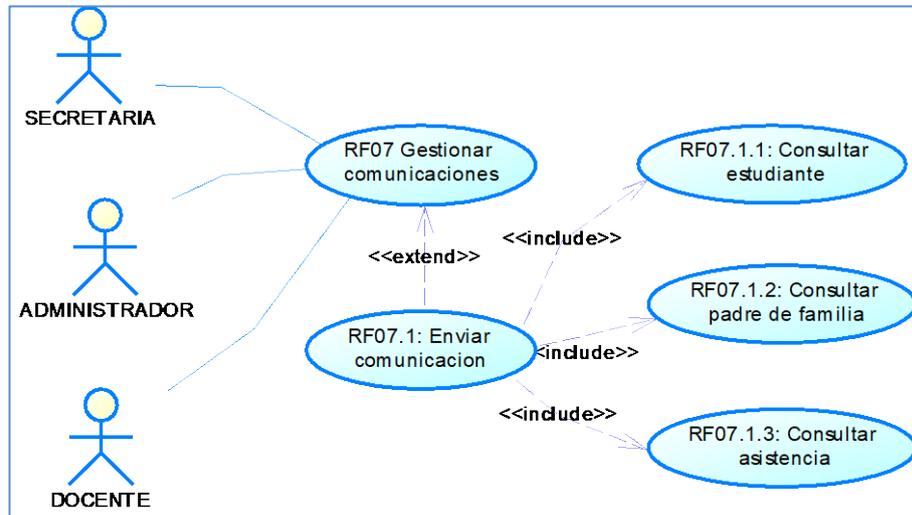


Figura 17: Caso de uso, gestionar comunicaciones
Fuente: Elaboración propia

Gestionar informes

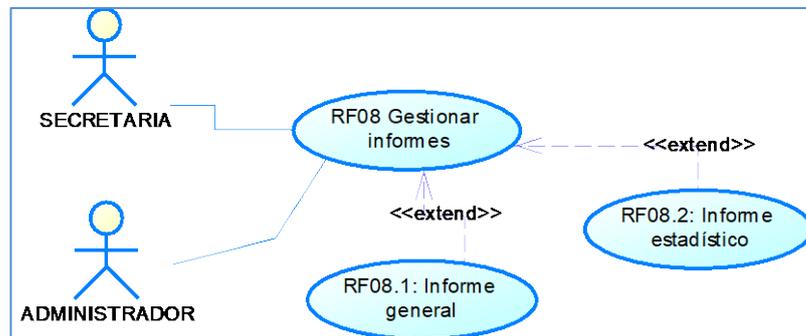


Figura 18: Caso de uso, gestionar informes
Fuente: Elaboración propia

Gestionar docente

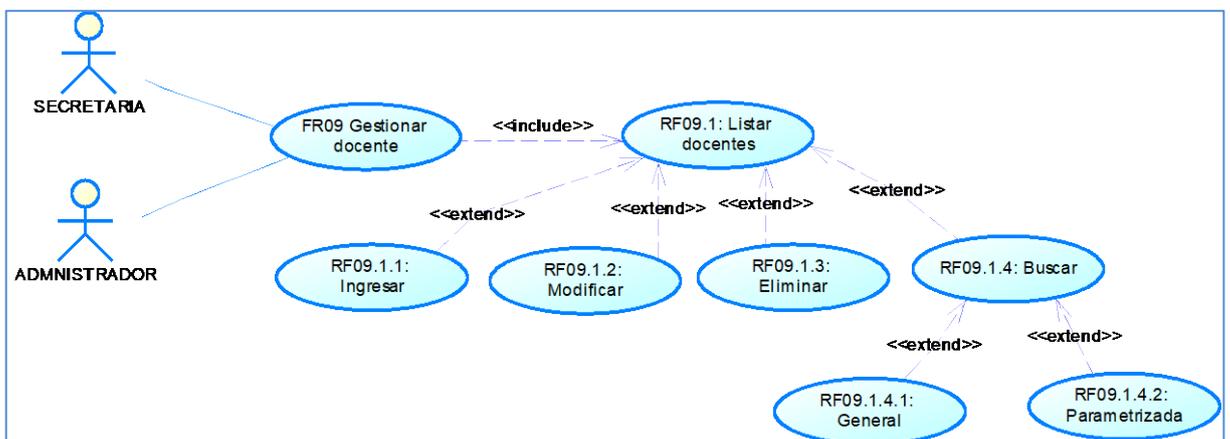


Figura 19: Caso de uso, gestionar docente
Fuente: Elaboración propia

5.9. Representación gráfica de procesos

A continuación se presenta un flujograma de los procesos con los requerimientos iniciales para el desarrollo del sistema de control de asistencia estudiantil.

Símbolo	Descripción
	Inicio/Final Determina el inicio o fin de una operación.
	Proceso Se utiliza para procesos, verificar calidad o cantidad
	Decision Indica una decisión sobre una actividad o proceso
	Dirección de Flujo Muestra hacia donde va el flujo

Tabla 22: Simbología de diagrama de flujo
Fuente: Elaboración propia

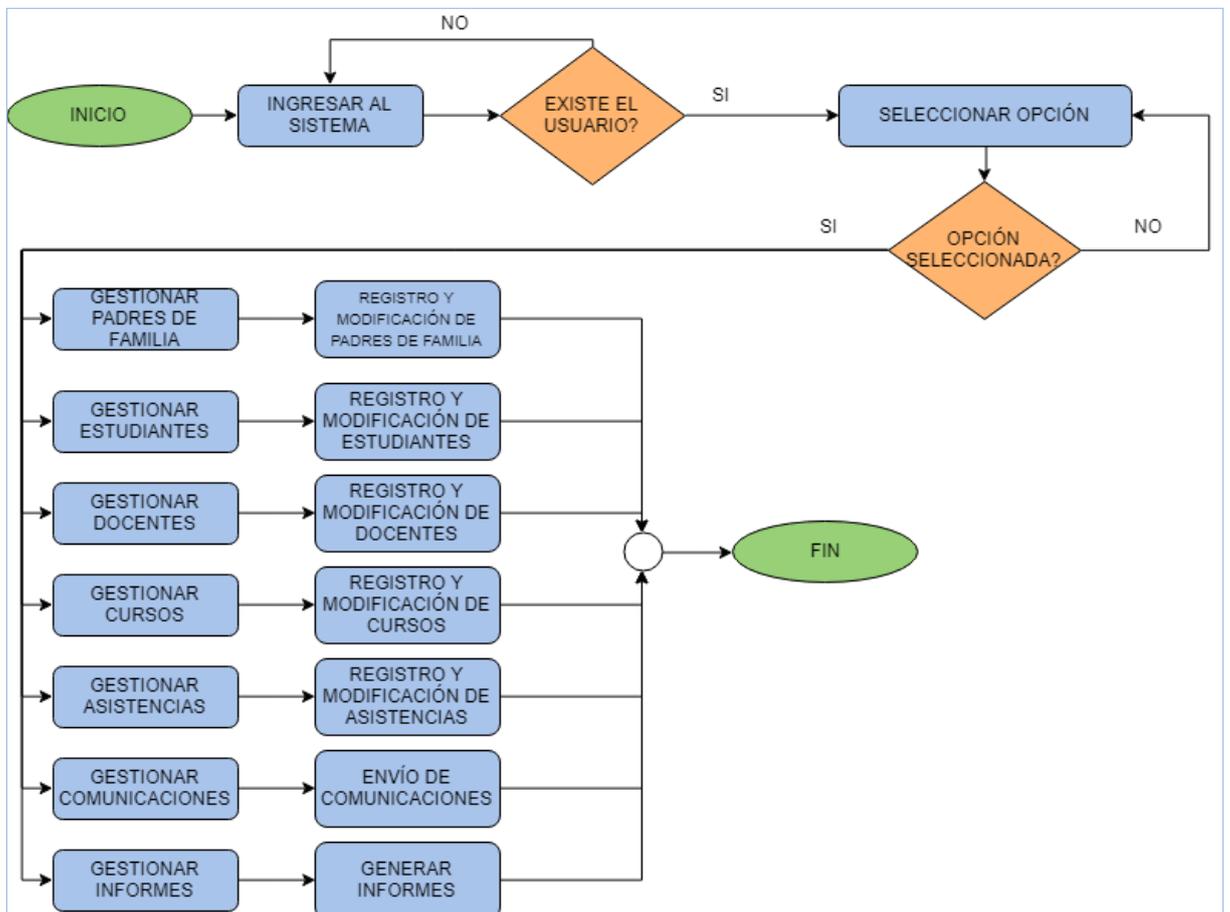


Figura 20: Diagrama de flujos de procesos del sistema SISCAE

Fuente: Elaboración propia

El diagrama anterior, indica que el usuario debe autenticarse en el sistema con sus credenciales para dar acceso a las opciones del sistema que le corresponden. Después de iniciar sesión, el sistema redireccionará a una pantalla principal de administración, en donde se presentará un menú con las opciones que gestionan: estudiantes, padres de familia, docentes, cursos, asistencias, comunicaciones e informes. El acceso a las opciones estará definida por los roles administrador, secretaria y docente.

5.10. Especificación de casos de uso

Gestionar usuario

Tabla 23: Descripción de caso de uso gestionar usuario

Identificador	RF01	
Nombre	Gestionar usuario	
Descripción	Se utiliza para el registro, modificación y eliminación de usuarios de aplicación	
Precondición	El administrador quiere ingresar nuevos usuarios en la aplicación para que tengan acceso a las opciones del sistema	
Postcondición	La información de los usuarios se registra en las base de datos del sistema, además, puede ser modificada y eliminada	
Actores	Administrador	
Secuencia Normal	Paso	Acción
	1	El administrador inicia sesión con su usuario y contraseña. En caso de que el usuario y contraseña sean erróneos continúa por la secuencia de error 1
	2	El sistema presenta un menú de administración
	3	El administrador da clic en la opción "usuarios"
	4	El sistema muestra un listado de todos los usuarios con opciones para cada uno. En caso de que no se presente la información ir a la secuencia de error 2
	5	El sistema muestra un toolbar con la opción de nuevo y un cuadro de texto buscar. En caso de que no se requiera ingresar un nuevo usuario ir a la secuencia alternativa 1
	6	El administrador da clic en el botón nuevo
	7	El sistema muestra un formulario para llenar la información del usuario
	8	El administrador llena la información en el formulario de usuario y da clic en el botón guardar. En caso de inconsistencia en la información ingresada ir a la secuencia de error 2
9	El sistema muestra un mensaje de éxito de registro de nuevo usuario y recarga la tabla para mostrar los cambios.	
Secuencia alternativa 1	Paso	Acción
	1	El administrador busca el registro y da clic en el botón de edición de usuario. En caso de que no sea una edición ir a la secuencia alternativa 2
	2	El sistema presenta un formulario con los campos llenos con la información del usuario seleccionado para editar.
	3	El administrador cambia la información del usuario y da clic en el botón guardar.
4	El sistema presenta un mensaje de éxito en la modificación y vuelve al flujo normal paso 4.	
Secuencia	1	El administrador busca el registro y da clic en el botón eliminar de usuario.

alternativa 2	2	El sistema presenta un mensaje de confirmación de eliminación
	3	El administrador da clic en el botón aceptar. En caso de que se seleccione el botón cancelar ir a la secuencia normal paso 4
	4	El sistema muestra un mensaje de éxito de eliminación de registro
Secuencia de error 1	Paso	Acción
	1	Se informa que se ha cometido un error en ingreso del usuario y contraseña
Secuencia de error 2	2	Con el usuario y contraseña rellenos se vuelve a la secuencia normal 1 paso 1
	1	Se presenta un mensaje de error de problemas con la conexión a la base de datos
Secuencia de error 3	2	Se informa que se contacte con el soporte técnico
	1	El sistema muestra un mensaje de datos erróneos ingresados y señala con color rojo los campos incorrectos
Importancia	2	El administrador corrige los campos incorrectos y se dirige a la secuencia normal paso 8
		Vital
Urgencia		No puede esperar
Observaciones		El caso de uso descrito, debe cumplirse para que existan usuarios que accedan a las opciones del sistema

Fuente: Elaboración propia

Autenticar y autorizar

Tabla 24: Descripción de caso de uso autenticar y autorizar

Identificador	RF02	
Nombre	Autenticar y autorizar	
Descripción	Se utiliza para identificar a un usuario y dar acceso a las opciones correspondientes	
Precondición	Un usuario desea ingresar a las opciones del sistema mediante sus credenciales (usuario y clave)	
Postcondición	El sistema da la autorización y muestra las opciones a las que puede acceder el usuario	
Actores	Administrador, secretaria, docente	
Secuencia Normal	Paso	Acción
	1	El sistema muestra un formulario de logeo.
	2	El usuario ingresa sus credenciales (usuario y clave) en los cuadros de texto del formulario de logeo y da clic en el botón iniciar sesión.
	3	El sistema valida los datos ingresados. En caso de problemas de validación ir a la secuencia de error 1
	4	El sistema verifica la existencia del usuario y clave. En caso de que no se realice la verificación ir a la secuencia de error 2
	5	El sistema autoriza y muestra las opciones correspondientes del usuario. En caso de no autorizar el acceso ir a la secuencia de error 3
Secuencia de error 1	6	El usuario se encuentra logeado en el sistema.
	Paso	Acción
Secuencia de error 2	1	Se informa que se ha cometido un error en ingreso del usuario y contraseña
	2	Con el cuadro de texto usuario y contraseña llenos nuevamente se vuelve a la secuencia normal paso 2
Secuencia de error 3	1	Se presenta un mensaje de error de problemas con la conexión a la base de datos
	2	Se informa que se contacte con el soporte técnico
Secuencia de error 3	1	El sistema muestra un mensaje de que el usuario y clave no existen en los registros de la base de datos y pide que los ingrese nuevamente.
	2	El usuario ingresa nuevamente su nombre de usuario y clave y se dirige a la secuencia normal paso 2

Importancia	Vital
Urgencia	No puede esperar
Observaciones	El caso de uso descrito, debe cumplirse para que los usuarios puedan acceder a las opciones del sistema asignadas

Fuente: Elaboración propia

Gestionar estudiante

Tabla 25: Descripción de caso de uso gestionar estudiante

Identificador	RF03	
Nombre	Gestionar estudiante	
Descripción	Se utiliza para el registro, modificación y eliminación de la información de estudiantes	
Precondición	El usuario quiere registrar nuevos estudiantes en la aplicación para que se pueda gestionar el registro de asistencias	
Postcondición	La información de los estudiantes se registra en las base de datos del sistema, además, puede ser modificada y eliminada	
Actores	Administrador, secretaria	
Secuencia Normal	Paso	Acción
	1	El usuario inicia sesión con su usuario y contraseña. En caso de que el usuario y contraseña sean erróneos continúa por la secuencia de error 1
	2	El sistema presenta un menú de administración
	3	El usuario da clic en la opción "estudiantes"
	4	El sistema muestra un listado de todos los estudiantes con opciones para cada uno. En caso de que no se presente la información ir a la secuencia de error 2
	5	El sistema muestra un toolbar con la opción de nuevo. En caso de que no se requiera ingresar un nuevo estudiante ir a la secuencia alternativa 1
	6	El usuario da clic en el botón nuevo
	7	El sistema muestra un formulario para llenar la información del estudiante
	8	El administrador llena la información en el formulario de estudiante y da clic en el botón guardar. En caso de inconsistencia en la información ingresada ir a la secuencia de error 2
9	El sistema muestra un mensaje de éxito de registro de nuevo estudiante y recarga la tabla para mostrar los cambios.	
Secuencia alternativa 1	Paso	Acción
	1	El usuario busca el registro y da clic en el botón de edición de estudiante. En caso de que no sea una edición ir a la secuencia alternativa 2
	2	El sistema presenta un formulario con los campos llenos con la información del estudiante seleccionado para editar.
	3	El usuario cambia la información del estudiante y da clic en el botón guardar.
4	El sistema presenta un mensaje de éxito en la modificación y vuelve al flujo normal paso 4 .	
Secuencia alternativa 2	1	El usuario busca el registro y da clic en el botón eliminar estudiante.
	2	El sistema presenta un mensaje de confirmación de eliminación
	3	El usuario da clic en el botón aceptar. En caso de que se seleccione el botón cancelar ir a la secuencia normal paso 4
	4	El sistema muestra un mensaje de éxito de eliminación de registro
Secuencia de error 1	Paso	Acción
	1	Se informa que se ha cometido un error en ingreso del usuario y contraseña
2	El usuario ingresa sus credenciales nuevamente y se vuelve a la secuencia normal 1 paso 1	
Secuencia de error 2	1	Se presenta un mensaje de error de problemas con la conexión a la base de datos
	2	Se informa que se contacte con el soporte técnico
Secuencia de error 3	1	El sistema muestra un mensaje de datos erróneos ingresados y señala con color rojo los campos incorrectos

	2	El usuario corrige los campos incorrectos y se dirige a la secuencia normal paso 8
Importancia	Vital	
Urgencia	No puede esperar	
Observaciones	El caso de uso descrito, debe cumplirse para que existan en el sistema estudiantes a los que se les registrará la asistencia.	

Fuente: Elaboración propia

Gestionar padre de familia

Tabla 26: Descripción de caso de uso gestionar padre de familia

Identificador	RF04	
Nombre	Gestionar padre de familia	
Descripción	Se utiliza para el registro, modificación y eliminación de la información de padres de familia	
Precondición	El usuario quiere registrar nuevos padres de familia en la aplicación para que se pueda gestionar el envío de comunicaciones	
Postcondición	La información de los padres de familia se registra en las base de datos del sistema, además, puede ser modificada y eliminada	
Actores	Administrador, secretaria	
Secuencia Normal	Paso	Acción
	1	El usuario inicia sesión con su usuario y contraseña. En caso de que el usuario y contraseña sean erróneos continúa por la secuencia de error 1
	2	El sistema presenta un menú de administración
	3	El usuario da clic en la opción "padre de familia"
	4	El sistema muestra un listado de todos los padres de familia con opciones para cada uno. En caso de que no se presente la información ir a la secuencia de error 2
	5	El sistema muestra un toolbar con la opción de nuevo y un cuadro de texto buscar. En caso de que no se requiera ingresar un nuevo padre de familia ir a la secuencia alternativa 1
	6	El usuario da clic en el botón nuevo
	7	El sistema muestra un formulario para llenar la información del padre de familia
	8	El usuario llena la información en el formulario de padre de familia y da clic en el botón guardar. En caso de inconsistencia en la información ingresada ir a la secuencia de error 2
9	El sistema muestra un mensaje de éxito de registro de nuevo padre de familia y recarga la tabla para mostrar los cambios.	
Secuencia alternativa 1	Paso	Acción
	1	El usuario busca al padre de familia y da clic en el botón de edición. En caso de que no sea una edición ir a la secuencia alternativa 2
	2	El sistema presenta un formulario con los campos llenos con la información del padre de familia seleccionado para editar.
	3	El usuario cambia la información del padre de familia y da clic en el botón guardar.
4	El sistema presenta un mensaje de éxito en la modificación y vuelve al flujo normal paso 4 .	
Secuencia alternativa 2	1	El usuario busca al padre de familia y da clic en el botón eliminar.
	2	El sistema presenta un mensaje de confirmación de eliminación
	3	El usuario da clic en el botón aceptar. En caso de que se seleccione el botón cancelar ir a la secuencia normal paso 4
	4	El sistema muestra un mensaje de éxito de eliminación de registro
Secuencia de error 1	Paso	Acción
	1	Se informa que se ha cometido un error en ingreso del usuario y contraseña

	2	El usuario ingresa sus credenciales nuevamente y se vuelve a la secuencia normal 1 paso 1
Secuencia de error 2	1	Se presenta un mensaje de error de problemas con la conexión a la base de datos
	2	Se informa que se contacte con el soporte técnico
Secuencia de error 3	1	El sistema muestra un mensaje de datos erróneos ingresados y señala con color rojo los campos incorrectos
	2	El usuario corrige los campos incorrectos y se dirige a la secuencia normal paso 8
Importancia	Vital	
Urgencia	Puede esperar	
Observaciones	El caso de uso descrito, debe cumplirse para que los estudiantes del sistema tengan asignado padres familias a los cuales se enviarán las comunicaciones correspondientes	

Fuente: Elaboración propia

Gestionar curso

Tabla 27: Descripción de caso de uso gestionar curso

Identificador	RF05	
Nombre	Gestionar curso	
Descripción	Se utiliza para el registro, modificación y eliminación de la información de cursos, la matriculación de estudiantes y asignación de docentes al curso.	
Precondición	El usuario quiere registrar nuevos cursos en el sistema, para que se pueda gestionar el la matriculación de estudiantes y asignación de docentes	
Postcondición	La información de los cursos se registra en las base de datos del sistema, además, puede ser modificada y eliminada. También, se matriculan los estudiantes y se asignan los docentes a los cursos.	
Actores	Administrador, secretaria	
Secuencia Normal	Paso	Acción
	1	El usuario inicia sesión con su usuario y contraseña. En caso de que el usuario y contraseña sean erróneos continúa por la secuencia de error 1
	2	El sistema presenta un menú de administración
	3	El usuario da clic en la opción "cursos"
	4	El sistema muestra un listado de todos los cursos con opciones para cada uno. En caso de que no se presente la información ir a la secuencia de error 2
	5	El sistema muestra un toolbar con la opción de nuevo. En caso de que no se requiera ingresar un curso ir a la secuencia alternativa 1
	6	El usuario da clic en el botón nuevo
	7	El sistema muestra un formulario para llenar la información del curso
	8	El usuario llena la información en el formulario de curso y da clic en el botón guardar. En caso de inconsistencia en la información ingresada ir a la secuencia de error 2
9	El sistema muestra un mensaje de éxito de registro de nuevo curso y recarga la tabla para mostrar los cambios. En caso de querer matricular estudiantes y asignar docentes al curso ir a la secuencia alternativa 3	
Secuencia alternativa 1	Paso	Acción
	1	El usuario busca el curso y da clic en el botón de edición. En caso de que no sea una edición ir a la secuencia alternativa 2
	2	El sistema presenta un formulario con los campos llenos con la información del curso seleccionado para editar.
	3	El usuario cambia la información del curso y da clic en el botón guardar.
Secuencia alternativa 2	4	El sistema presenta un mensaje de éxito en la modificación y vuelve al flujo normal paso 4 .
	1	El usuario busca el curso y da clic en el botón eliminar.
	2	El sistema presenta un mensaje de confirmación de eliminación

	3	El usuario da clic en el botón aceptar. En caso de que se seleccione el botón cancelar ir a la secuencia normal paso 4
	4	El sistema muestra un mensaje de éxito de eliminación de registro
Secuencia alternativa 3	1	El sistema muestra el listado de los cursos
	2	El usuario selecciona la opción del curso matricular estudiantes. En caso de no ser una matriculación continuar en la secuencia alternativa 4
	3	El sistema presenta un listado de estudiantes
	4	El usuario selecciona los estudiantes y los matricula en el curso
Secuencia alternativa 4	1	El sistema muestra un listado de cursos
	2	El usuario selecciona la opción asignar docente.
	3	El sistema muestra una lista de los docentes registrados
	4	El administrador asigna los docentes al curso seleccionado
Secuencia de error 1	Paso	Acción
	1	Se informa que se ha cometido un error en ingreso del usuario y contraseña
	2	El usuario ingresa sus credenciales nuevamente y se vuelve a la secuencia normal 1 paso 1
Secuencia de error 2	1	Se presenta un mensaje de error de problemas con la conexión a la base de datos
	2	Se informa que se contacte con el soporte técnico
Secuencia de error 3	1	El sistema muestra un mensaje de datos erróneos ingresados y señala con color rojo los campos incorrectos
	2	El usuario corrige los campos incorrectos y se dirige a la secuencia normal paso 8
Importancia	Vital	
Urgencia	Puede esperar	
Observaciones	El caso de uso descrito, debe cumplirse para realizar el registro de los cursos y posterior matriculación de estudiantes y asignación de docentes	

Fuente: Elaboración propia

Gestionar asistencia estudiantil

Tabla 28: Descripción de caso de uso gestionar asistencia estudiantil

Identificador	RF06	
Nombre	Gestionar asistencia estudiantil	
Descripción	Se utiliza para el registro y modificación de la asistencia de estudiantes	
Precondición	El usuario quiere registrar la asistencia a clases de los estudiantes de forma manual o masiva utilizando reconocimiento facial.	
Postcondición	La información de la asistencia a clases de los estudiantes se registra en la base de datos del sistema, además, puede ser modificada y eliminada	
Actores	Administrador, secretaria, docente	
Secuencia Normal	Paso	Acción
	1	El usuario inicia sesión con sus credenciales de acceso. En caso de que el usuario y contraseña sean erróneos continúa por la secuencia de error 1
	2	El sistema presenta un menú de administración
	3	El administrador da clic en la opción "asistencia estudiantil"
	4	El sistema muestra un listado de todos los cursos asignados y con opciones para cada uno. En caso de que no se presente la información ir a la secuencia de error 2
	5	El sistema muestra un toolbar con la opción de nuevo y un cuadro de texto buscar.
6	El usuario selecciona el curso.	

	7	El sistema presenta la información del curso y las opciones que se pueden realizar en él.
	8	El usuario selecciona la opción asistencia manual. En caso de que no se requiera realizar el registro de asistencia manual ir a la secuencia alternativa 1
	9	El sistema presenta un formulario con la fecha actual, la materia, el docente y el listado de estudiantes.
	10	El usuario marca la asistencia de los estudiantes y da clic en el botón finalizar toma de asistencia
	11	El sistema verifica la información ingresada. En caso de errores en la validación ir a la secuencia de error 2
	12	El sistema muestra un mensaje de éxito en el registro de asistencia. En caso de que no se finalice correctamente la transacción ir a la secuencia de error 3
Secuencia alternativa 1	Paso	Acción
	1	El usuario selecciona la opción asistencia automática.
	2	El sistema presenta un formulario con la fecha actual, la materia, el docente y el listado de estudiantes.
	3	El usuario da clic en el botón generar asistencia automática
	4	El sistema inicializa la cámara en una ventana de la aplicación. En caso de no inicialización de la cámara ira a la secuencia de error 4
	5	El usuario toma la foto
	6	El sistema sube la imagen al servidor de reconocimiento facial.
	7	El servidor de reconocimiento realiza la identificación y devuelve los rostros encontrados
	8	El sistema marca la asistencia de los rostros identificados
9	El sistema muestra un mensaje de éxito en el registro de asistencia automática	
Secuencia de error 1	Paso	Acción
	1	Se informa que se ha cometido un error en ingreso del usuario y contraseña
Secuencia de error 2	1	Con el usuario y contraseña rellenos se vuelve a la secuencia normal 1 paso 1
	2	El sistema muestra un mensaje de datos erróneos ingresados y señala con color rojo los campos incorrectos
Secuencia de error 3	1	El usuario corrige los campos incorrectos y se dirige a la secuencia normal paso 10
	2	Se presenta un mensaje de error de problemas con la conexión a la base de datos
Secuencia de error 4	1	Se informa que se contacte con el soporte técnico
	2	El sistema muestra un mensaje de error que no localiza ninguna cámara conectada.
Importancia	1	El usuario conecta y selecciona la cámara y va a la secuencia alternativa 1 paso 3
	2	
Urgencia	Vital	
Observaciones	No puede esperar	
Observaciones	El caso de uso descrito, debe cumplirse para el registro de la asistencia estudiantil ya sea de forma manual o automáticamente usando reconocimiento facial.	

Fuente: Elaboración propia

Gestionar comunicaciones

Tabla 29: Descripción de caso de uso gestionar comunicaciones

Identificador	RF07
Nombre	Gestionar comunicaciones
Descripción	Se utiliza para el envío de comunicaciones a los padres de familia
Precondición	El usuario quiere enviar comunicaciones a los padres de familia respecto a las asistencias a clases de los estudiantes.

Postcondición	La información de la asistencia del estudiante es enviada mediante una comunicación al correo electrónico del padre de familia.	
Actores	Administrador, secretaria, docente	
Secuencia Normal	Paso	Acción
	1	El usuario inicia sesión con su usuario y contraseña. En caso de que el usuario y contraseña sean erróneos continúa por la secuencia de error 1
	2	El sistema presenta un menú de administración
	3	El usuario da clic en la opción "cursos"
	4	El sistema muestra un listado de todos los cursos con opciones para cada uno. En caso de que no se presente la información ir a la secuencia de error 2
	5	El sistema muestra la información del curso (fechas, docentes, lista de estudiantes).
	6	El usuario selecciona una fecha y al estudiante, escoge la opción enviar comunicación a padre de familia. En caso de querer enviar comunicaciones automáticas respecto a la asistencia estudiantil ir a la secuencia alternativa 1
	7	El sistema muestra un formulario para redactar la comunicación.
	8	El usuario escribe la comunicación y da clic en el botón enviar. En caso de inconsistencia en la información ingresada ir a la secuencia de error 3
9	El sistema muestra un mensaje de éxito en el envío de una comunicación.	
Secuencia alternativa 1	Paso	Acción
	1	El sistema muestra una opción de enviar comunicaciones a los padres de familia de los estudiantes que tienen falta
	2	El usuario da clic en enviar
Secuencia de error 1	Paso	Acción
	1	Se informa que se ha cometido un error en ingreso del usuario y contraseña
	2	Con el usuario y contraseña rellenos se vuelve a la secuencia normal 1 paso 1
Secuencia de error 2	1	Se presenta un mensaje de error de problemas con la conexión a la base de datos
	2	Se informa que se contacte con el soporte técnico
Secuencia de error 3	1	El sistema muestra un mensaje de datos erróneos ingresados y señala con color rojo los campos incorrectos
	2	El usuario corrige los campos incorrectos y se dirige a la secuencia normal paso 8
Importancia	Vital	
Urgencia	No puede esperar	
Observaciones	El caso de uso descrito, debe cumplirse para el envío de comunicaciones a los padres de familia de los estudiantes que faltan a clases.	

Fuente: Elaboración propia

Gestionar informes

Tabla 30: Descripción de caso de uso gestionar informes

Identificador	RF08
Nombre	Gestionar informes
Descripción	Se utiliza para generar informes de tipo general y estadísticos de las asistencias de estudiantes
Precondición	El usuario quiere generar los informes correspondientes de cada estudiante respecto a su asistencia a clases.
Postcondición	El informe es generado correctamente
Actores	Administrador, secretaria

Secuencia Normal	Paso	Acción
	1	El usuario inicia sesión con sus credenciales. En caso de que el usuario y contraseña sean erróneos continúa por la secuencia de error 1
	2	El sistema presenta un menú de administración
	3	El usuario da clic en la opción "cursos"
	4	El sistema muestra un listado de todos los cursos con opciones para cada uno. En caso de que no se presente la información ir a la secuencia de error 2
	5	El usuario selecciona el curso
	6	El sistema muestra toda la información del curso.
	7	El usuario da clic en el botón generar informe de curso.
	8	El sistema muestra opciones de informes individuales o por curso.
	9	El usuario selecciona un tipo de informe, selecciona la fecha y da clic en el botón generar. En caso de inconsistencia en la información ingresada ir a la secuencia de error 3
10	El sistema muestra el informe generado.	
Secuencia de error 1	Paso	Acción
	1	Se informa que se ha cometido un error en ingreso del usuario y contraseña
Secuencia de error 2	2	Con el usuario y contraseña rellenos se vuelve a la secuencia normal 1 paso 1
	1	Se presenta un mensaje de error de problemas con la conexión a la base de datos
Secuencia de error 3	2	Se informa que se contacte con el soporte técnico
	1	El sistema muestra un mensaje de datos erróneos ingresados y señala con color rojo los campos incorrectos
Secuencia de error 3	2	El sistema corrige los campos incorrectos y se dirige a la secuencia normal paso 8
	Importancia	Vital
Urgencia	Puede esperar	
Observaciones	El caso de uso descrito, debe cumplirse para generar los informe parametrizables	

Fuente: *Elaboración propia*

Gestionar docente

Tabla 31: Descripción de caso de uso gestionar docente

Identificador	RF09	
Nombre	Gestionar docente	
Descripción	Se utiliza para el registro, modificación y eliminación de la información de docentes	
Precondición	El usuario quiere registrar nuevos docentes en la aplicación que gestionen las asistencias de estudiantes	
Postcondición	La información de los docentes se registra en las base de datos del sistema, además, puede ser modificada y eliminada	
Actores	Administrador, secretaria	
Secuencia Normal	Paso	Acción
	1	El usuario inicia sesión con su usuario y contraseña. En caso de que el usuario y contraseña sean erróneos continúa por la secuencia de error 1
	2	El sistema presenta un menú de administración
	3	El usuario da clic en la opción "docentes"
	4	El sistema muestra un listado de todos los docentes con opciones para cada uno. En caso de que no se presente la información ir a la secuencia de error 2
5	El sistema muestra un toolbar con la opción de nuevo. En caso de que no se requiera ingresar un nuevo docente ir a la secuencia alternativa 1	

	6	El usuario da clic en el botón nuevo
	7	El sistema muestra un formulario para llenar la información del docente
	8	El administrador llena la información en el formulario de docente y da clic en el botón guardar. En caso de inconsistencia en la información ingresada ir a la secuencia de error 2
	9	El sistema muestra un mensaje de éxito de registro de nuevo docente y recarga la tabla para mostrar los cambios.
Secuencia alternativa 1	Paso	Acción
	1	El usuario busca el registro y da clic en el botón de edición de docente. En caso de que no sea una edición ir a la secuencia alternativa 2
	2	El sistema presenta un formulario con los campos llenos con la información del docente seleccionado para editar.
	3	El usuario cambia la información del docente y da clic en el botón guardar.
	4	El sistema presenta un mensaje de éxito en la modificación y vuelve al flujo normal paso 4 .
Secuencia alternativa 2	1	El usuario busca el registro y da clic en el botón eliminar docente.
	2	El sistema presenta un mensaje de confirmación de eliminación
	3	El usuario da clic en el botón aceptar. En caso de que se seleccione el botón cancelar ir a la secuencia normal paso 4
	4	El sistema muestra un mensaje de éxito de eliminación de registro
Secuencia de error 1	Paso	Acción
	1	Se informa que se ha cometido un error en ingreso del usuario y contraseña
	2	El usuario ingresa sus credenciales nuevamente y se vuelve a la secuencia normal 1 paso 1
Secuencia de error 2	1	Se presenta un mensaje de error de problemas con la conexión a la base de datos
	2	Se informa que se contacte con el soporte técnico
Secuencia de error 3	1	El sistema muestra un mensaje de datos erróneos ingresados y señala con color rojo los campos incorrectos
	2	El usuario corrige los campos incorrectos y se dirige a la secuencia normal paso 8
Importancia	Vital	
Urgencia	No puede esperar	
Observaciones	El caso de uso descrito, debe cumplirse para que existan en el sistema docentes que gestionen la asistencia de estudiantes.	

Fuente: Elaboración propia

6. Construcción

Durante la fase de construcción, todos los componentes y características de la aplicación se desarrollan e integran como un solo producto. Además, se van realizando pruebas de validación en la codificación de los componentes. (IBM, s.f.)

En esta fase se desarrollarán todas las vistas (lógica, procesos, despliegue, implementación) de los componentes del sistema SISCAE, para ir separando e identificando los módulos prioritarios y que se deben desarrollar para obtener una versión funcional de la aplicación.

6.1. Vista lógica

6.1.1. Diagrama de secuencias

Gestionar usuario

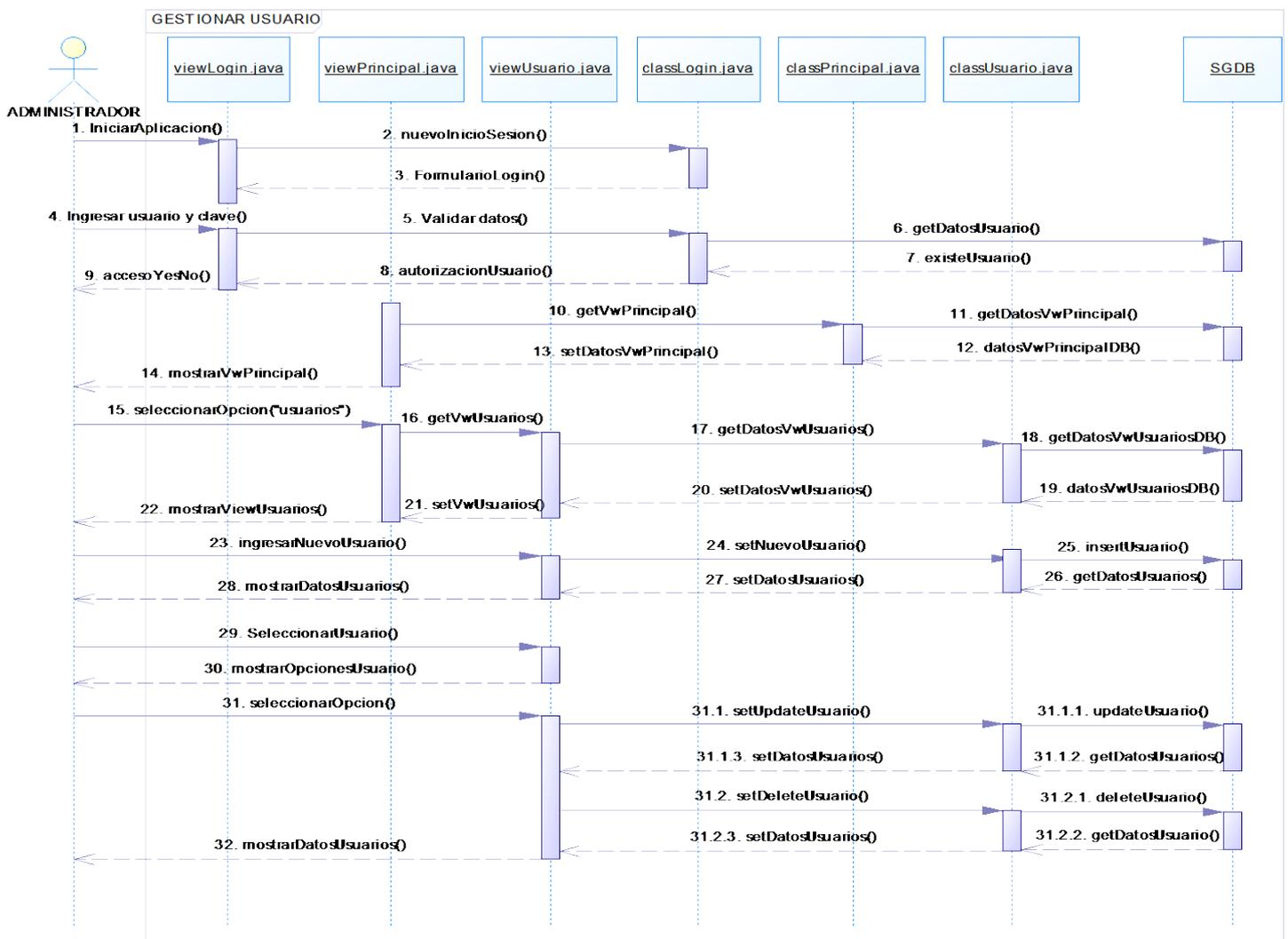


Figura 21: Diagrama de secuencia, gestionar usuario
Fuente: Elaboración propia

Autenticar y autorizar

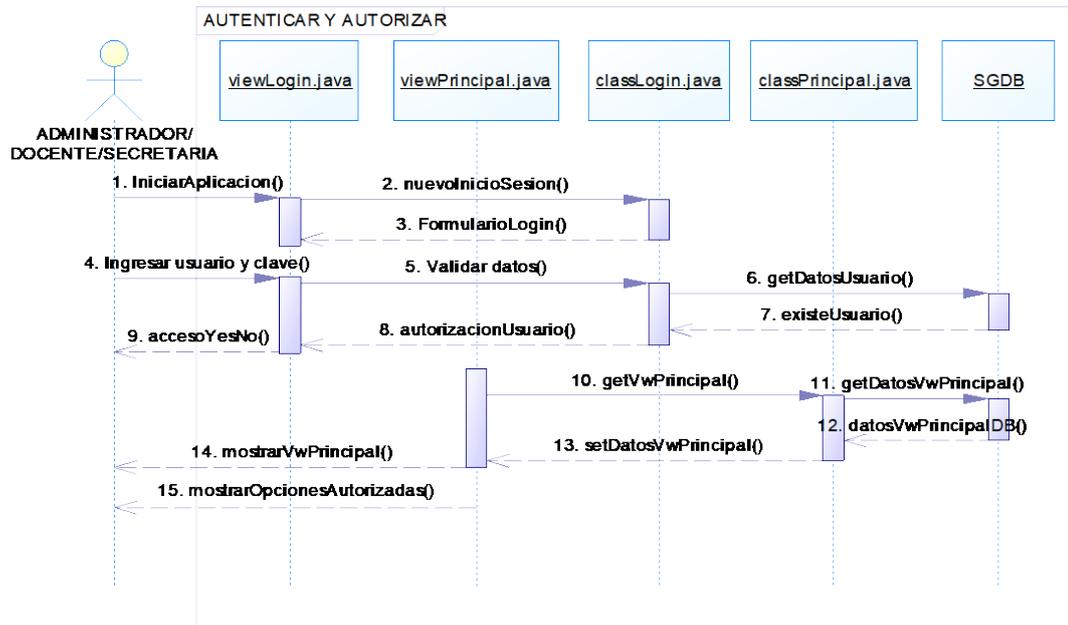


Figura 22: Diagrama de secuencia, gestionar autentcar y autorizar

Fuente: Elaboración propia

Gestionar estudiante

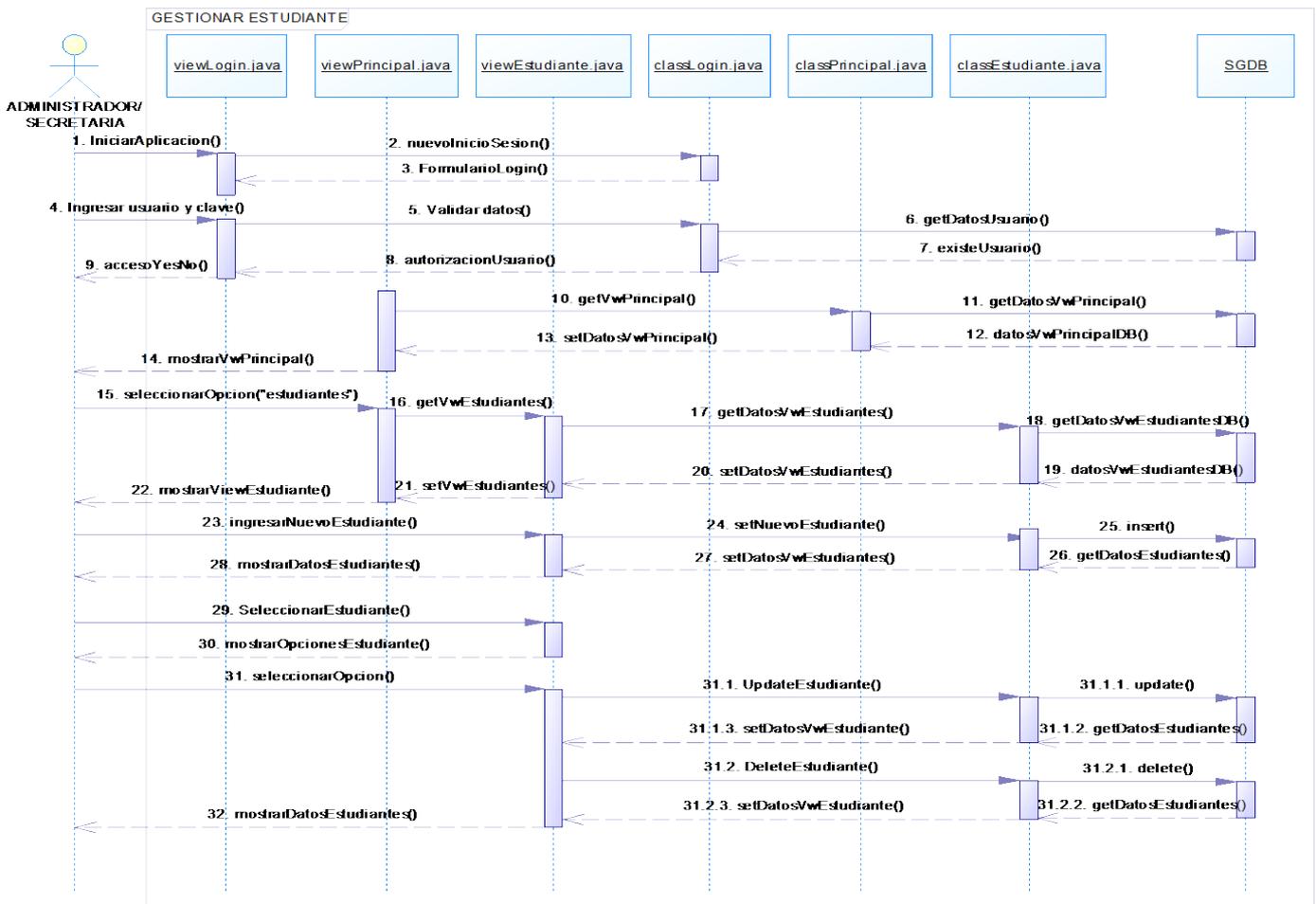


Figura 23: Diagrama de secuencia, gestionar estudiante

Fuente: Elaboración propia

Gestionar padre de familia

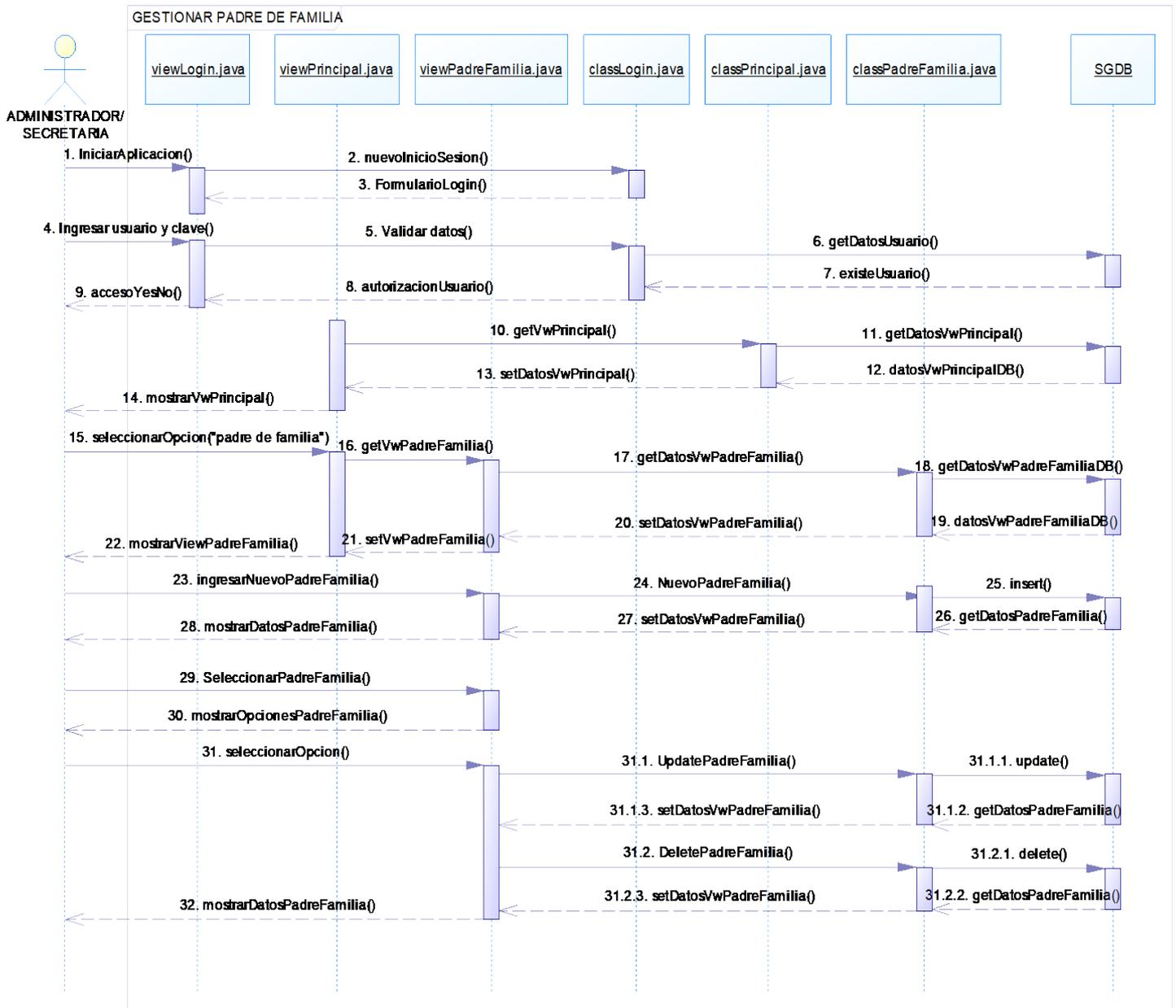


Figura 24: Diagrama de secuencia, gestionar padre de familia

Fuente: Elaboración propia

Gestionar curso

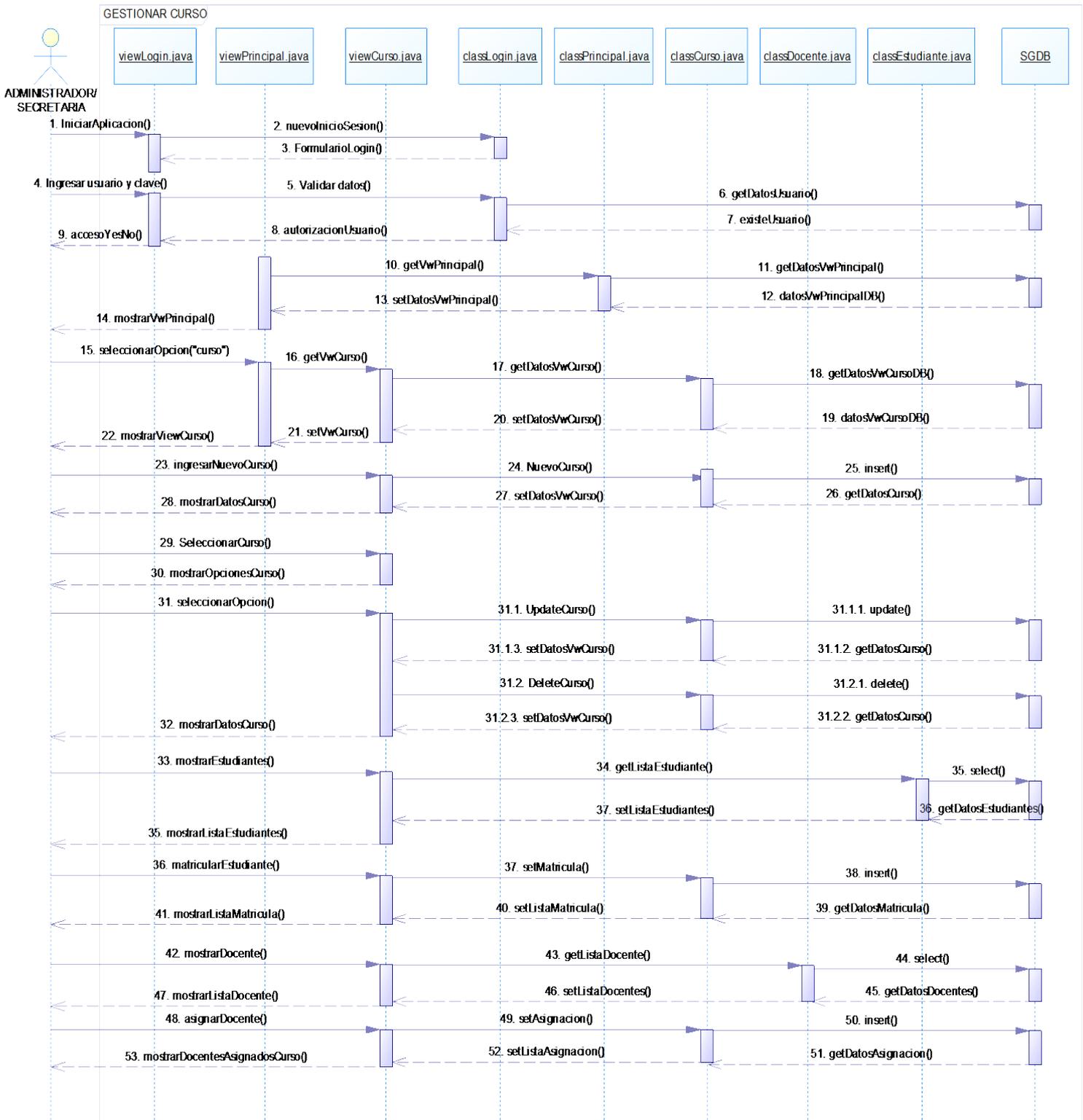


Figura 25: Diagrama de secuencia, gestionar curso
Fuente: Elaboración propia

Gestionar asistencia estudiantil

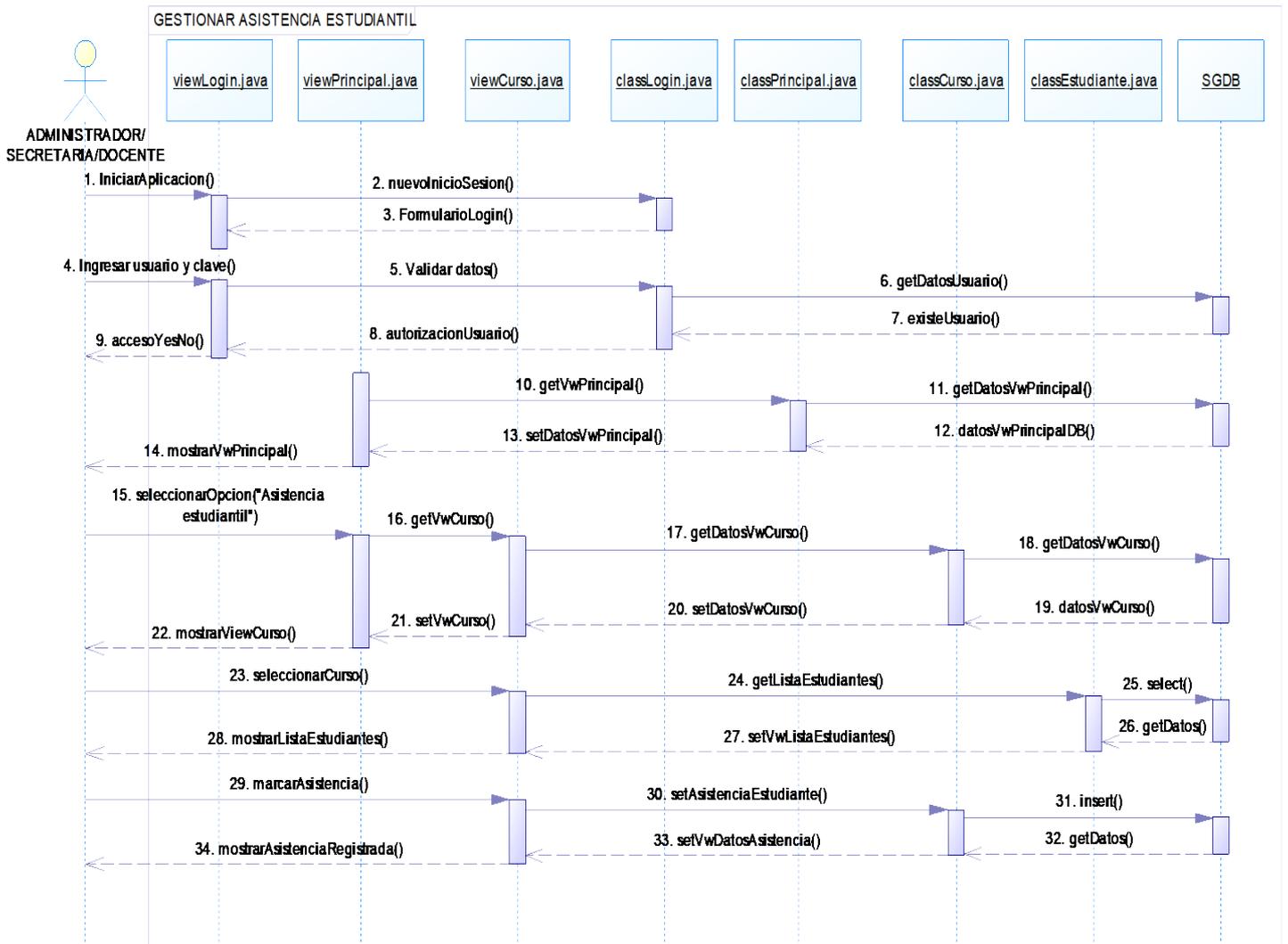


Figura 26: Diagrama de secuencia, gestionar asistencia estudiantil
Fuente: Elaboración propia

Gestionar comunicaciones



Figura 27: Diagrama de secuencia, gestionar comunicaciones
Fuente: Elaboración propia

Gestionar informes

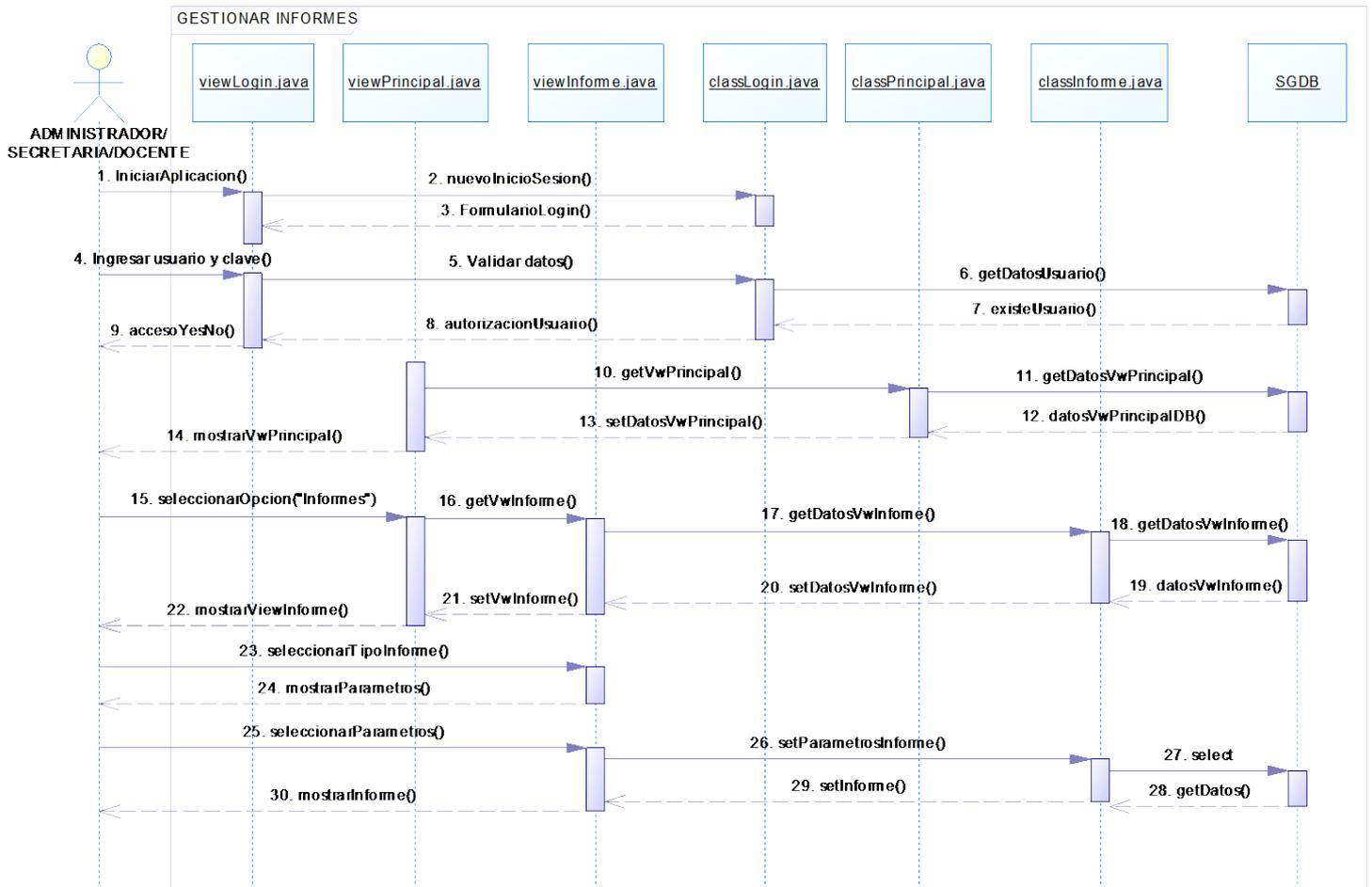


Figura 28: Diagrama de secuencia, gestionar informes
Fuente: Elaboración propia

Gestionar docente

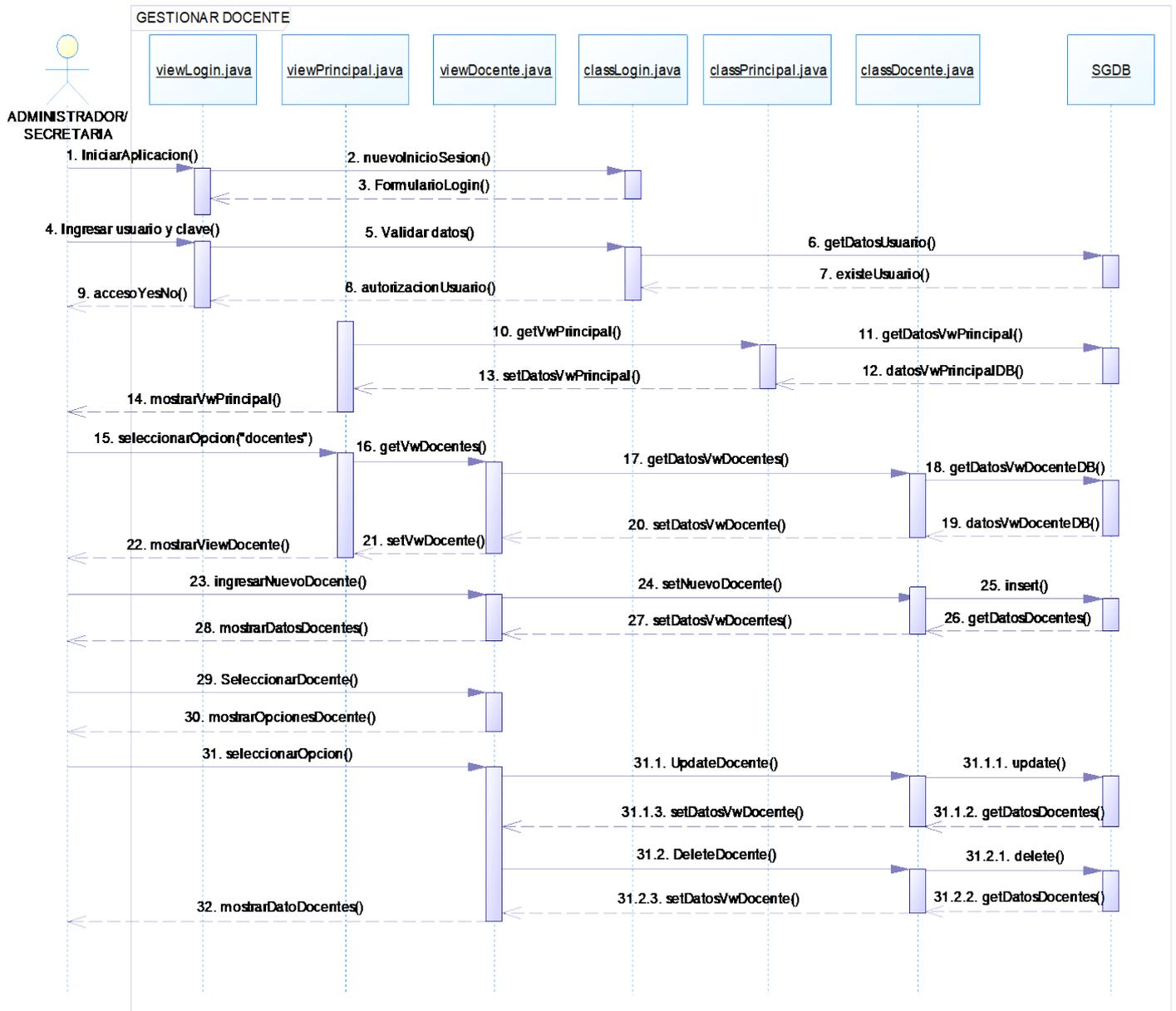


Figura 29: Diagrama de secuencia, gestionar docente
Fuente: Elaboración propia

6.1.2. Diagrama de clases

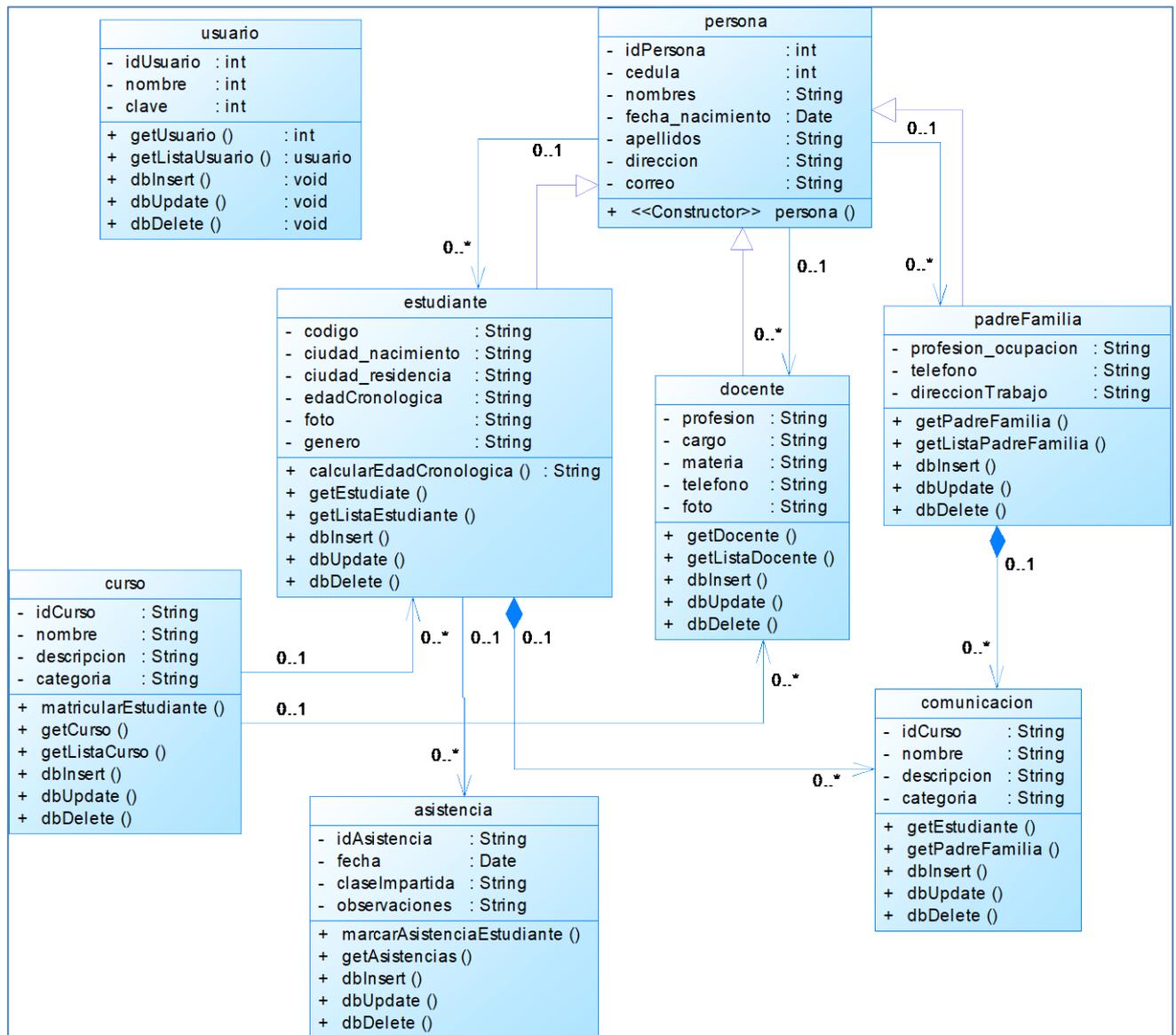


Figura 30: Diagrama de clases, SISCAE

Fuente: Elaboración propia

A continuación se describen cada una de las clases consideradas en el diagrama de la figura 30 respecto a: nombre, descripción, atributos, métodos y relaciones.

Nombre de la clase: usuario

Descripción: Esta clase describe la información de los usuarios que tendrán acceso al sistema y la forma de autenticación a través de un nombre de usuario y contraseña, cada cuenta es única y por ende se establece un idUsuario que servirá de token para el inicio de sesión.

Atributos: Esta clase presenta atributos como: idUsuario, nombre y clave; los dos primeros deben ser atributos únicos de cada usuario.

Métodos: Se implementan métodos para la recuperación de información (general, parametrizada), además, operaciones de inserción, actualización y eliminado de registros en la base de datos.

Relaciones: Esta clase no se encuentra relacionada con ninguna otra.

Nombre de la clase: persona

Descripción: Esta clase describe la información común requerida para los procesos que realizará la aplicación.

Atributos: Esta clase presenta atributos como: idPersona, cedula, nombres, apellidos, fecha de nacimiento, dirección y correo; estos atributos serán transferidos a otros objetos mediante la herencia.

Métodos: Se implementará un constructor y los setters y getters para cada atributo, en el diagrama no se reflejan estos métodos, puesto que la clase se visualizaría más extensa. Sin embargo, en el apartado estándar de codificación se explicará cómo interactuarán las clases con otras mediante sus métodos.

Relaciones: Esta clase se encuentra relacionada con las clases: estudiante, docente y padreFamilia a través de la herencia, ya que las clases mencionadas heredan los atributos de la clase persona y que son comunes para todas ellas. Además de la herencia, esta clase presenta las siguientes relaciones:

- ✓ Persona con estudiante se encuentran unidos por una relación de asociación porque las personas que se ingresen al sistema pueden ser estudiantes.
- ✓ La relación persona docente es de asociación ya que las personas que se registren en el sistema pueden ser docentes.
- ✓ La relación persona padreFamilia es de asociación ya que las personas que se registren en el sistema pueden ser padres de familia.

Nombre de la clase: estudiante

Descripción: Esta clase describe la información que se requiere para los estudiantes, a los que posteriormente se les asignará un curso y se registrará la asistencia a clases.

Atributos: Esta clase presenta atributos como: codigo, ciudad_nacimiento, ciudad_residencia, edadCronológica, foto, genero. Los otros atributos comunes como nombres, apellidos, cedula entre otros, serán heredados de la clase persona.

Métodos: Se implementará una función para el cálculo de la edad cronológica utilizando la fecha de nacimiento. Además, se incluyen métodos para la recuperación de información (general, parametrizada), además, operaciones de inserción, actualización y eliminado de registros en la base de datos.

Relaciones: Esta clase se encuentra relacionada con la clase persona y curso de la siguiente manera:

- ✓ Estudiante persona se encuentran unidos por herencia y por una relación de asociación ya que los estudiantes son personas.
- ✓ La relación curso estudiantes es de asociación porque un curso puede contener varios estudiantes a los cuales se les registrará la asistencia.

Nombre de la clase: docente

Descripción: Esta clase describe la información que se requiere para los docentes, a los que posteriormente se les asignará un curso y serán los encargados de registrar la asistencia a clases de los estudiantes.

Atributos: Esta clase presenta atributos como: profesión, cargo, materia, teléfono, foto. Los otros atributos comunes como nombres, apellidos, cedula entre otros, serán heredados de la clase persona.

Métodos: Se incluyen métodos para la recuperación de información (general, parametrizada), además, operaciones de inserción, actualización y eliminado de registros en la base de datos.

Relaciones: Esta clase se encuentra relacionada con la clase persona y curso de la siguiente manera:

- ✓ Docente persona se encuentran unidos por herencia y una relación de asociación ya que los docentes son personas.
- ✓ La relación curso docente es de asociación porque un curso puede contener varios docentes los cuales imparten sus clases y registran la asistencia de los estudiantes.

Nombre de la clase: padreFamilia

Descripción: Esta clase describe la información que se requiere para los padres de familia de los estudiantes, servirá para registrar la información necesaria para el envío de comunicaciones respecto a la asistencia de los estudiantes.

Atributos: Esta clase presenta atributos como: profesión u ocupación, teléfono y dirección de trabajo. Los otros atributos comunes como nombres, apellidos, cedula entre otros, serán heredados de la clase persona.

Métodos: Se incluyen métodos para la recuperación de información (general, parametrizada), además, operaciones de inserción, actualización y eliminado de registros en la base de datos.

Relaciones: Esta clase se encuentra relacionada con la clase persona y comunicación de la siguiente manera:

- ✓ Relación PadreFamilia persona, se encuentran unidos por herencia y una relación de asociación ya que los padres de familia son personas.
- ✓ La relación comunicación padreFamilia es de composición, porque la comunicación debe contener la información de los padres de familia a quien o quienes será enviada.

Nombre de la clase: curso

Descripción: Esta clase describe la información de los cursos a los cuales pertenecen los estudiantes y docentes.

Atributos: Esta clase presenta los siguientes atributos: idCurso, nombre, descripción y categoría.

Métodos: Se implementará una función para la matriculación de estudiantes. Además, se incluyen métodos para la recuperación de información (general, parametrizada), además, operaciones de inserción, actualización y eliminado de registros en la base de datos.

Relaciones: Esta clase se encuentra relacionada con la clase estudiante y docente de la siguiente manera:

- ✓ La relación curso estudiantes es de asociación porque un curso puede contener varios estudiantes a los cuales se les registrará la asistencia.
- ✓ La relación curso docente es de asociación porque un curso puede contener varios docentes que serán los encargados de registrar la asistencia de los estudiantes.

Nombre de la clase: asistencia

Descripción: Esta clase describe la información que se registra en la asistencia de los estudiantes de un determinado curso.

Atributos: Esta clase presenta los siguientes atributos: idAsistencia, fecha, claseImpartida y observaciones.

Métodos: Se implementará una función marcarAsistenciaEstudaintes. Además, se incluyen métodos para la recuperación de información (general, parametrizada), además, operaciones de inserción, actualización y eliminado de registros en la base de datos.

Relaciones: Esta clase se encuentra relacionada con la clase estudiante de la siguiente manera:

- ✓ La relación estudiante asistencia es de asociación porque un estudiante puede tener varias asistencias en un periodo lectivo.

Nombre de la clase: comunicación

Descripción: Esta clase describe la información que se requiere para las comunicaciones que serán enviadas a los padres de familia, respecto a la asistencia a clases de los estudiantes.

Atributos: Esta clase presenta atributos como: idCurso, nombre o asunto, descripción, y categoría.

Métodos: Se incluyen métodos para la recuperación de información (general, parametrizada) de estudiantes y padres de familia, además, operaciones de inserción, actualización y eliminado de registros en la base de datos.

Relaciones: Esta clase se encuentra relacionada con la clase padreFamilia y estudiante de la siguiente manera:

- ✓ Relación comunicación padreFamilia, se encuentran unidos por una relación de composición ya que la información de los padres es necesaria para el envío de comunicaciones.
- ✓ La relación comunicación estudiante es de composición porque una comunicación debe obligatoriamente contener la información del estudiante para ser enviada.

6.1.3. Diseño de la base de datos

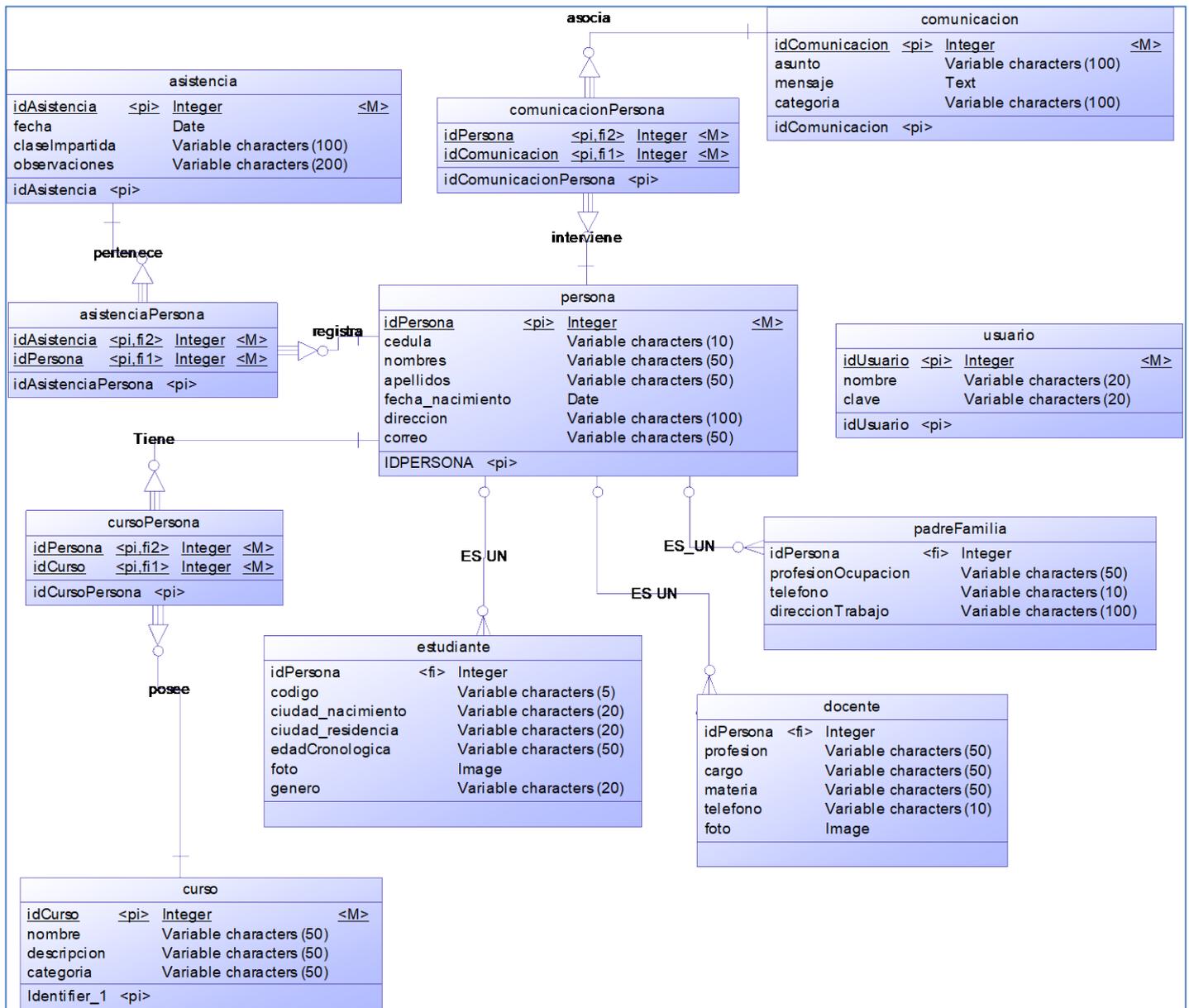


Figura 31: Diseño lógico de la base de datos del sistema
Fuente: Elaboración propia

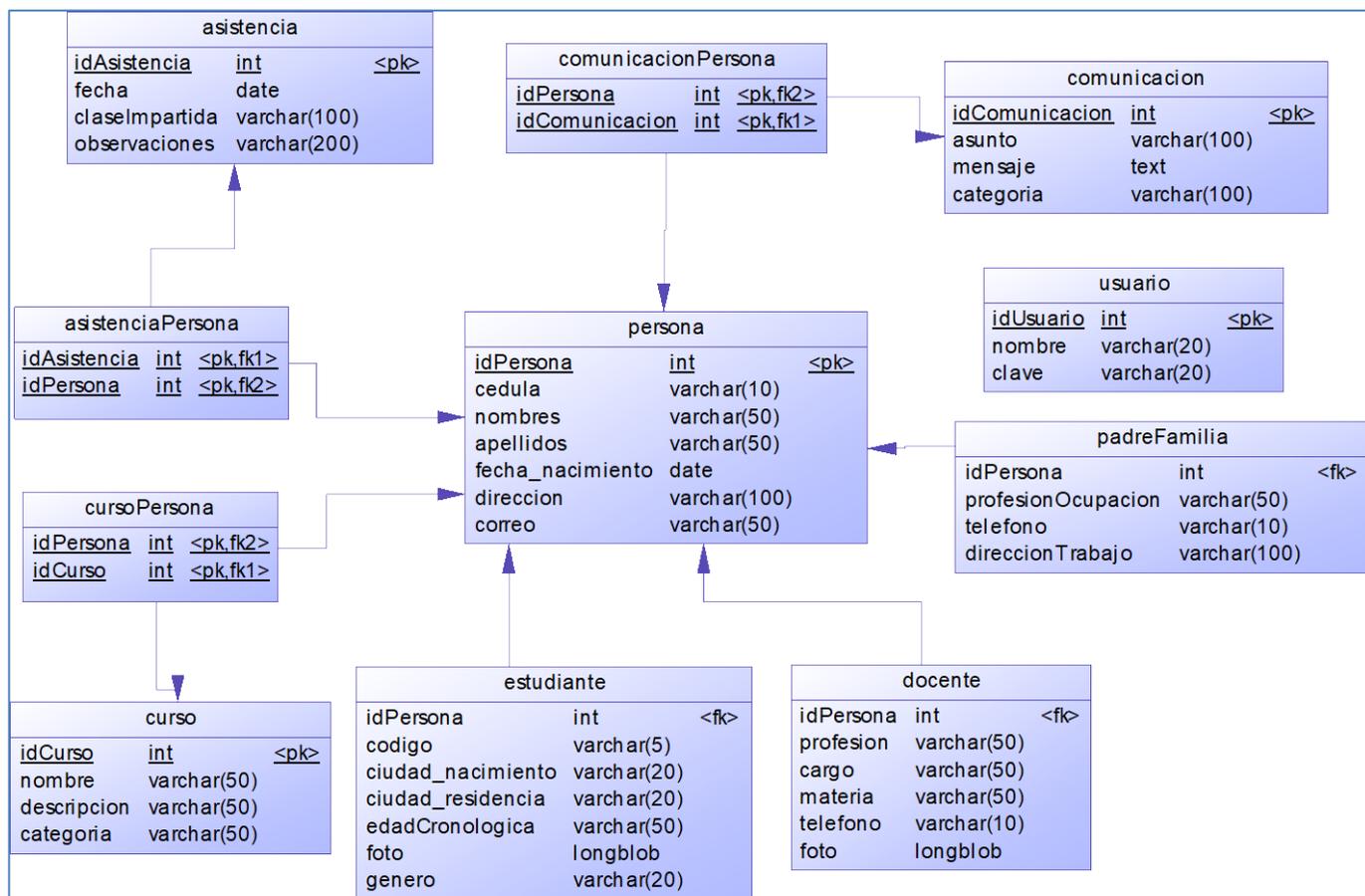


Figura 32: Diseño físico de la base de datos del sistema

Fuente: Elaboración propia

El diagrama físico, fue generado automáticamente a partir del diagrama lógico de la base de datos, utilizando la herramienta *Power Designer*, en el cual se puede observar once tablas. Sin embargo, esta estructura estará sujeta a cambios de acuerdo a los requisitos de la aplicación.

El diseño actual, se encuentra compuesto por las tablas: persona, estudiante, docente, padreFamilia, curso, asistencia, comunicación, usuario, comunicacionPersona, asistenciaPersona y cursoPersona; cuyas relaciones y descripción se enuncian a continuación.

La tabla persona se encuentra relacionada con las tablas estudiante, docente y padreFamilia, a través de una cardinalidad de uno a muchos puesto que una persona puede ser un estudiante, docente o padre de familia.

Se observa la relación curso persona, por medio de una tabla frágil, ya que varios cursos pueden contener varias personas que pueden ser estudiantes y docentes. Además, las

personas de tipo docente, deben de registrar la asistencia de las personas que son estudiantes, por lo cual, la relación persona asistencia se da por medio de una tabla frágil porque muchas personas pueden tener muchas asistencias.

Así mismo, la relación comunicación persona, se realiza con una tabla frágil por el motivo de que en una comunicación intervienen varias personas que son estudiantes, docentes y padres de familia.

Como es normal en todo desarrollo, los modelos siempre estarán sujetos a cambios ya que pueden intervenir más campos y tablas que aún no han sido considerados. Sin embargo, se ha realizado un análisis para que los requisitos cambiantes no tengan un impacto mayor en el desarrollo.

6.2. Vista de procesos

6.2.1. Situación actual del proceso

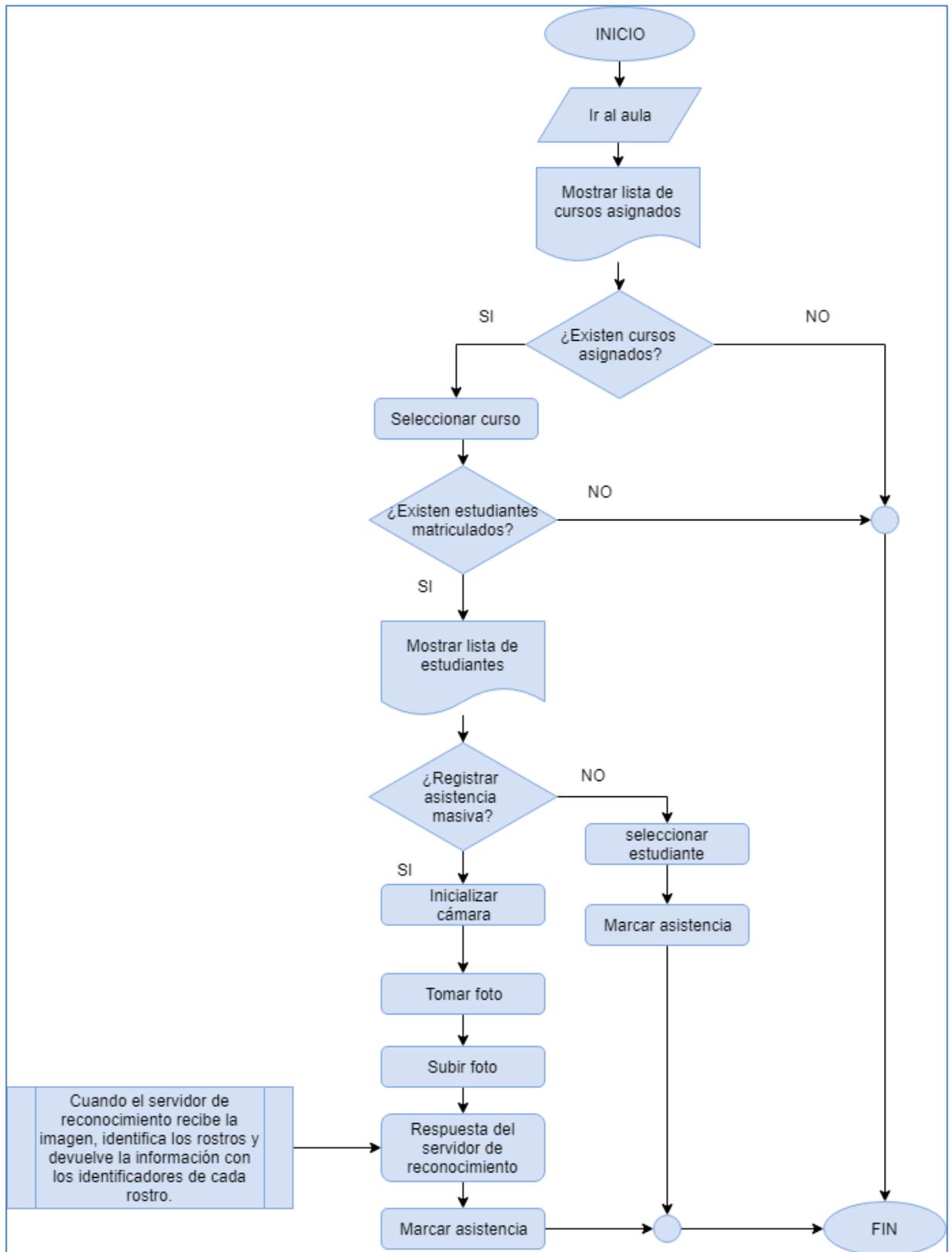


Figura 33: Proceso actual de registro de asistencias masivas o individuales

Fuente: Elaboración propia

Para registrar las asistencias de los estudiantes de un determinado curso, el sistema muestra al usuario (docente, secretaria, administrador) los cursos en los que se encuentra asignado; posteriormente el usuario selecciona el curso y elige si desea registrar la asistencia masiva o individual, en caso de seleccionar la primera el sistema inicia la cámara, toma la foto y la envía al servidor de reconocimiento en espera de los identificadores de los rostros reconocidos en la foto. Además, el usuario puede seleccionar la opción asistencia individual y marcarla de forma tradicional, la idea es que se cuente con los dos procesos.

6.2.2. Identificación de procesos

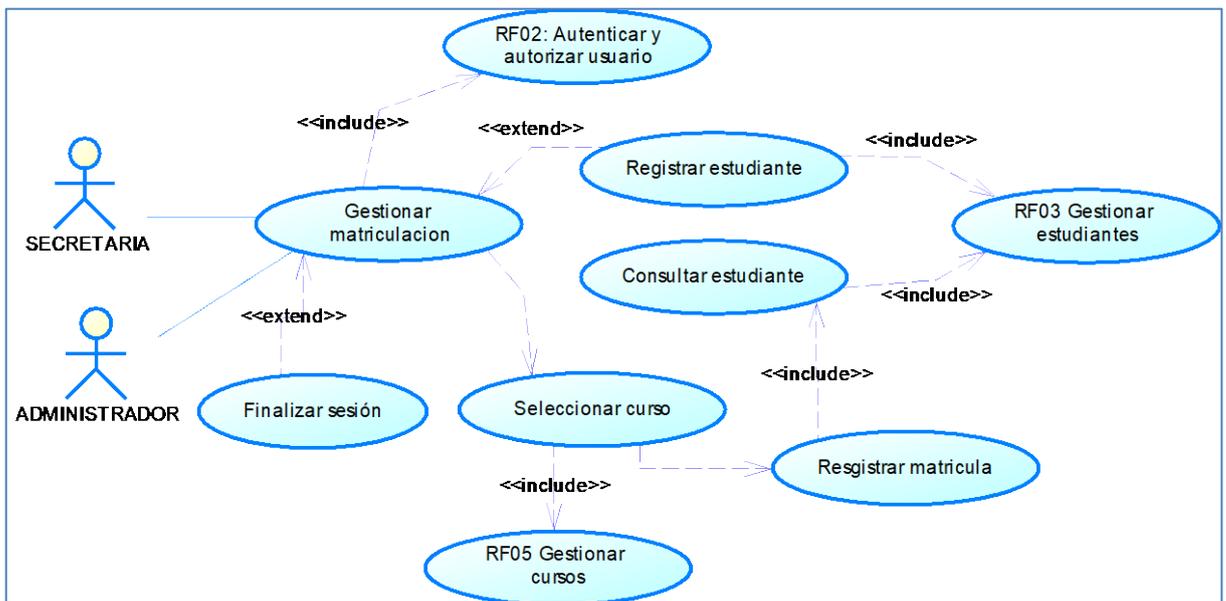


Figura 34: Gestión de proceso matriculación estudiante curso
Fuente: Elaboración propia

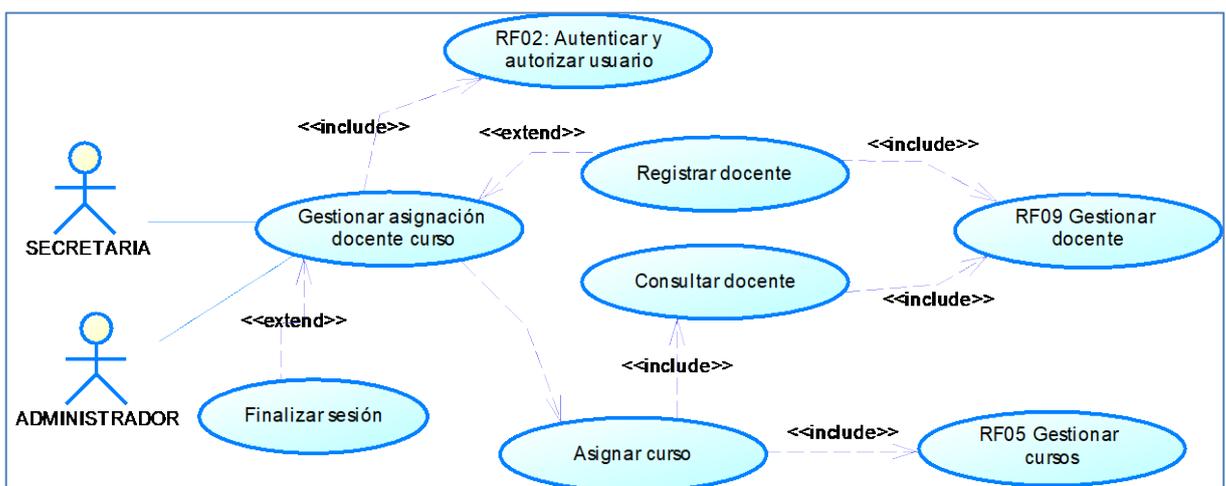


Figura 35: Gestión de proceso asignación docente curso
Fuente: Elaboración propia



Figura 36: Gestión de proceso asistencia estudiantil
Fuente: Elaboración propia



Figura 37: Gestión de proceso envío de comunicaciones
Fuente: Elaboración propia

6.3. Vista de despliegue

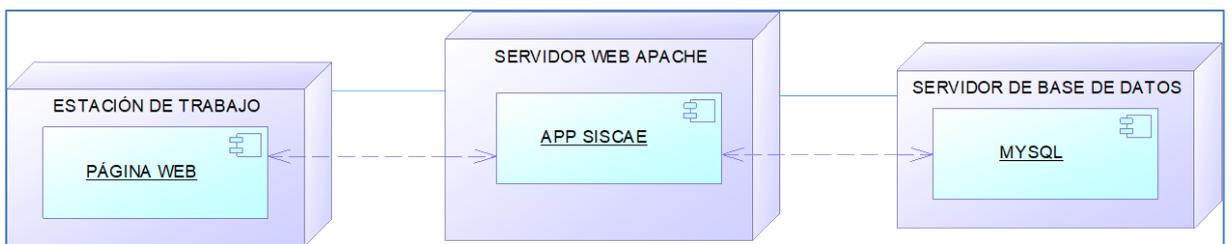


Figura 38: Diagrama de despliegue
Fuente: Elaboración propia

6.4. Vista de implementación

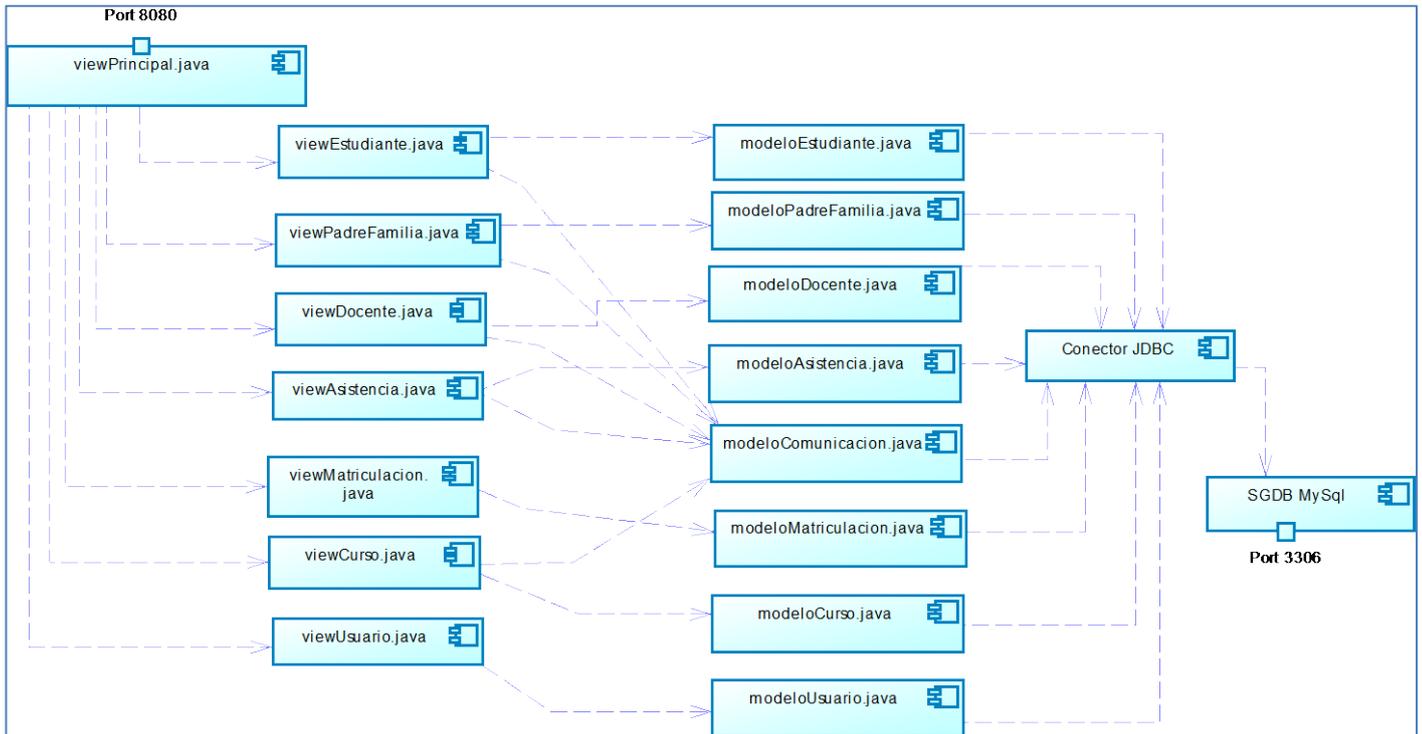


Figura 39: Diagrama de componentes del sistema
Fuente: Elaboración propia

6.5. Diccionario de datos

Ver anexo V.

6.6. Diseño de modelo en capas

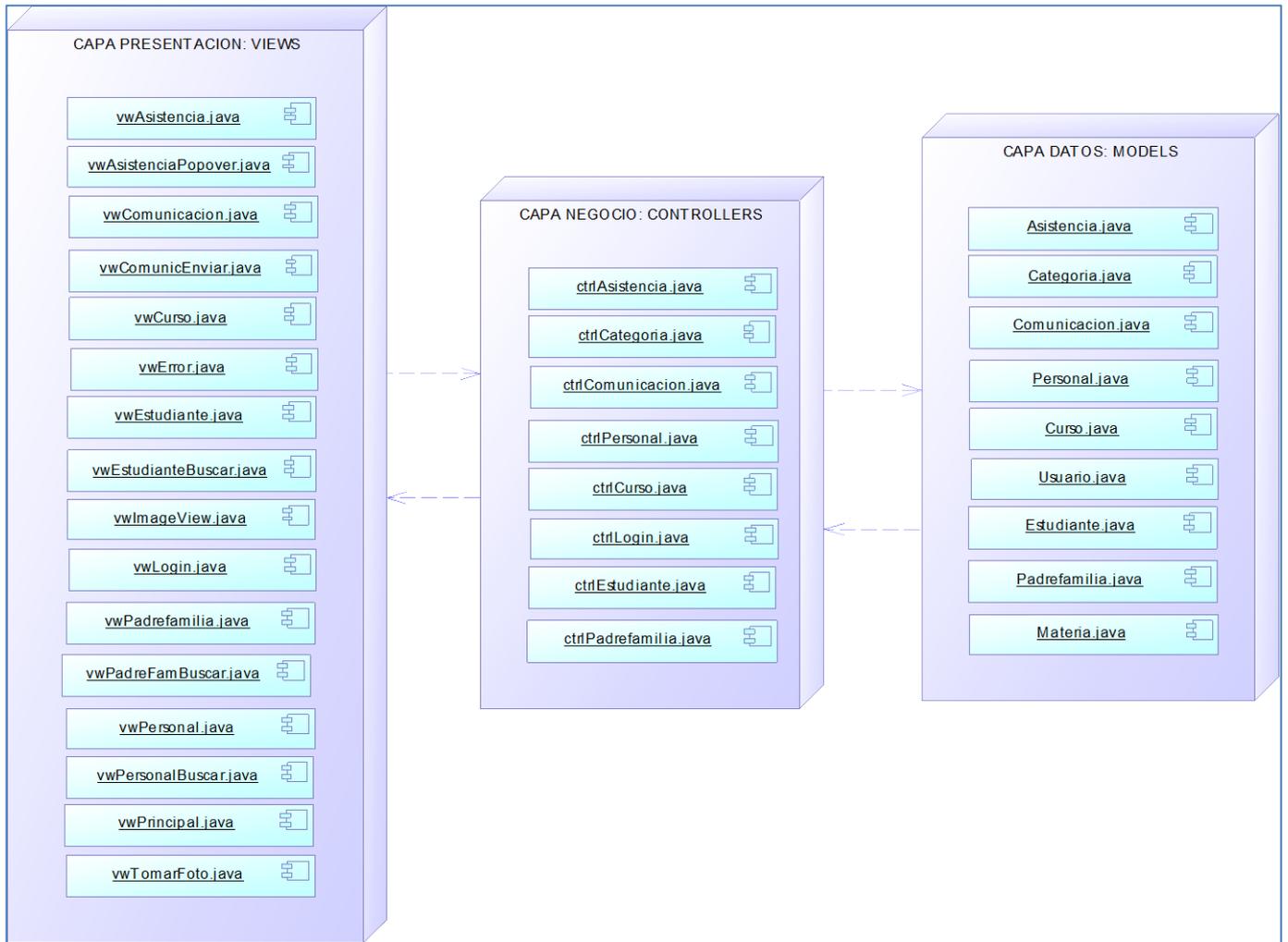


Figura 40: Diagrama de componentes del sistema
Fuente: Elaboración propia

La capa modelo, es la encargada gestionar los datos del sistema, y para conseguir este objetivo se ha considerado utilizar el framework Hibernate junto con JPA (persistencia de datos), que garantizará las transacciones realizadas a la base de datos como inserciones, actualizaciones, eliminación y recuperación de la información. Además, generará automáticamente el diseño físico de la base de datos sin la necesidad de utilizar un administrador de SGDB.

Una vez creada la capa de modelos utilizando Hibernate y las anotaciones de JPA se obtiene el diseño final de la base de datos y sus relaciones, como se presenta en la siguiente figura.

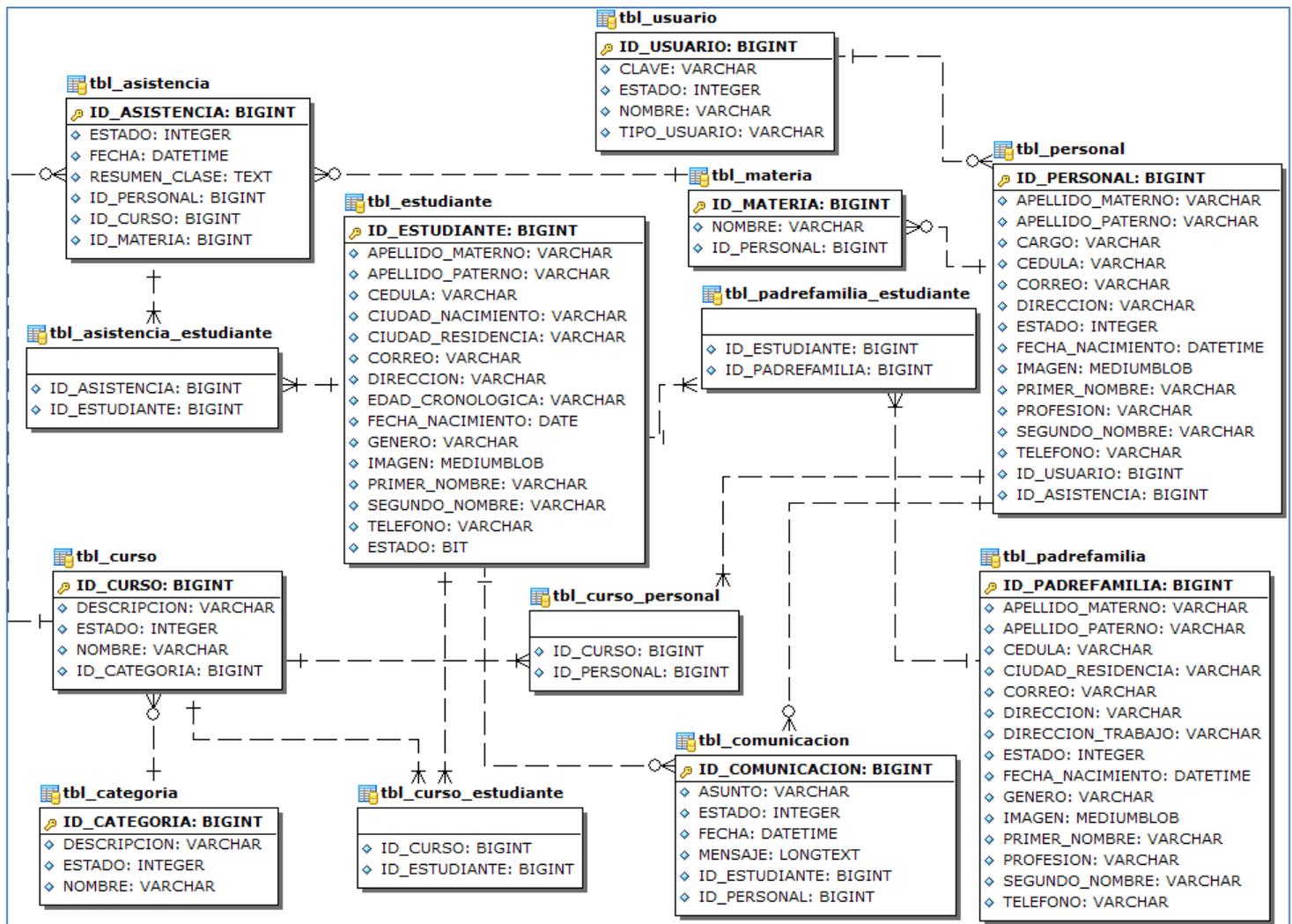


Figura 41: Diseño final de la base de datos

Fuente: Elaboración propia

Como se puede observar en la figura 41, el diseño final de la base de datos posee ciertos cambios respecto al diseño inicial, JPA hace que todo sea tratado como un objeto, por tal motivo la tabla persona se dividió en varios objetos como Estudiante, PadreFamilia, Personal (docente, secretaria), Asistencia, Curso, Materia, Categoría y sus entidades débiles que los relacionan con una cardinalidad de muchos a muchos.

6.7. Diseño de interfaces del sistema

Ver anexo III.

6.8. Estándares de codificación

6.8.1. Generales

Para la codificación de la aplicación SISCAE, se utilizará el lenguaje de programación Java. Las interfaces, serán desarrolladas utilizando el Framework Vaadin que está basado en Google Web Toolkit (GWT) y permite desarrollar aplicaciones Webs como si fueran de escritorio, es decir, que solo se necesita conocimiento de Java para la creación de los componentes que se visualizarán en el navegador web. Para el tratamiento de los datos, se optó por la utilización del Framework Hibernate y JPA, lo que garantiza la persistencia de los datos y optimiza en gran medida las transacciones de creación, recuperación, actualización y eliminación. Así mismo, solo se utilizarán modelos en Java que permitirán la persistencia hacia la base de datos, además, brinda la posibilidad de migrar de SGDB sin muchos procedimientos engorrosos, tan solo con configurar una línea del archivo persistense.xml de Hibernate, especificando el gestor de base de datos, para este TFM se utilizó MySQL.

6.8.2. Indentación

El IDE de Eclipse, proporciona una Indentación automática, por tal motivo el código fuente estará sujeto a las exigencias de este entorno de desarrollo integrado.

6.8.3. Nombrado de símbolos

- **Nombrado de clases**

Las clases en el sistema están divididas por paquetes, estos están nombrados de acuerdo a su función por ejemplo views, models, controllers.

La nomenclatura de clases views

- vwNombreDeLaClase (vwEstudiante, vwAsistencia)

La nomenclatura de clases models

- NombreClase (Estudiante, Asistencia)

La nomenclatura de clases controllers

- ctrlNombreDeLaClase (ctrlEstudiante, ctrlAsistencia)

- **Nombrado de métodos de clases**

Java, proporciona su propio estándar de nomenclatura para los métodos de las clases utilizando get y set al inicio de una operación seguido del atributo por ejemplo.

```
GetNombreEstudiante ()
```

```
GetDireccion ()
```

6.8.4. Espaciado dentro del código

Utilizar espaciados dentro de la programación cuando se usan operadores o concatenaciones una buena práctica para visión y mejor lectura del código fuente.

```
if (getNombre() != null)
String concat = "Gabriel" + "Salvatierra"
```

6.8.5. Comentarios dentro del código

Es importante comentar el código fuente, ya que puede ser utilizado por otros programadores para dar pautas de las acciones que se realizan, además, permite recordar que fue lo que se programó en caso de visualizar el código en un futuro. Los comentarios en Java se especifican con doble // para una línea o con /* */ para varias líneas de código.

7. Transición

7.1. Implementación

La aplicación se implementó en un servidor Tomcat 7, desplegado en una instancia de Windows Server 2017 en EC2 de AWS. Además, las imágenes de entrenamiento para el servidor de reconocimiento se subieron en un Bucket de S3 que también es un servicio proporcionado por AWS. Esta implementación se resume en las siguientes actividades.

Configuración de Instancia EC2

La instancia desplegada, contiene el sistema operativo Windows server 2012 R2, se optó por este sistema ya que brinda la posibilidad de utilizar el escritorio remoto para su administración.



Figura 42: Instancia EC2, Windows Server 2012 R2

Fuente: AWS EC2, Instances

Para el funcionamiento de la aplicación, es necesario instalar en el servidor, el AWS CLI, JDK 8, Tomcat 7 (xampp paquete de software libre) y MySQL Server. Una vez instalados los programas, se debe configurar el AWS CLI para que funcione con la cuenta IAM que permite administrar el acceso a los servicios y recursos de AWS, dicho servicio proporcionó la información que se muestra en el en la figura 43 y que debe ser establecida en los archivos del configuración del AWS CLI utilizando el comando en la consola `aws configure`.

```
Secret access key: fcZ6oajWe0MJ2pbLh1Lw48aDdU0RmLBupYNBmeb5
Acces key ID: AKIAJBYVDSBP3VYBQQVA
region: us-east-1
```

Figura 43: Datos de configuración para AWS CLI

Fuente: AWS IAM

Además, se deben configurar las variables de entorno para que el SDK de AWS para Java, pueda ser utilizado de forma correcta.

```
set AWS_ACCESS_KEY_ID=AKIAJBYVDSBP3VYBQQVA
set AWS_SECRET_ACCESS_KEY=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
set AWS_REGION=us-east-1
```

Figura 44: Configuración de variables de entorno

Fuente: AWS IAM – elaboración propia

Para verificar el correcto funcionamiento se comprueba que los servicios estén inicializados, utilizando el panel de control del xampp y accediendo al servidor mediante la Ip pública y el puerto 8080 que corresponde al servidor Tomcat

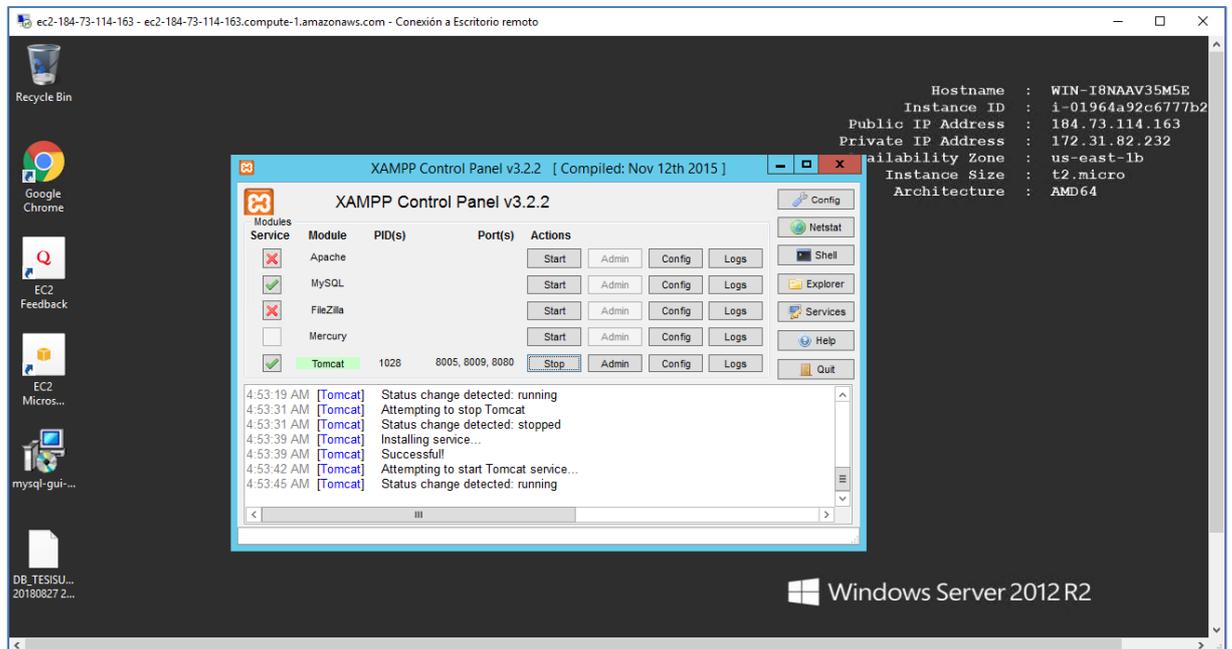


Figura 45: Verificación de ejecución de servicios en el Windows Server
Fuente: Elaboración propia

Compilación y despliegue de aplicación

Para desplegar el sistema en el servidor Tomcat, es necesario compilar el código del mismo en un archivo .war, gracias a las bondades del IDE Eclipse, es posible realizar esta acción usando el asistente de exportación de proyectos.

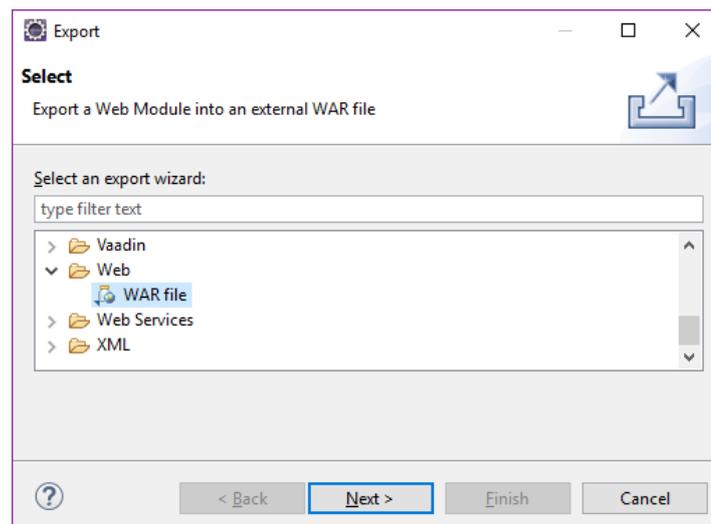


Figura 46: Compilación de aplicación a .war
Fuente: Elaboración propia

El archivo generado, deber ser copiado en la carpeta webapps del servidor Tomcat, este se encargará de desplegarlo y crear las carpetas necesarias para su funcionamiento.

Implementación de base de datos

Para este proceso, se debe tener el backup de la base de datos y utilizar el administrador de MySQL para importar los datos del script, aunque se puede realizar por consola, los administradores de tipo GUI facilitan en gran medida el trabajo.

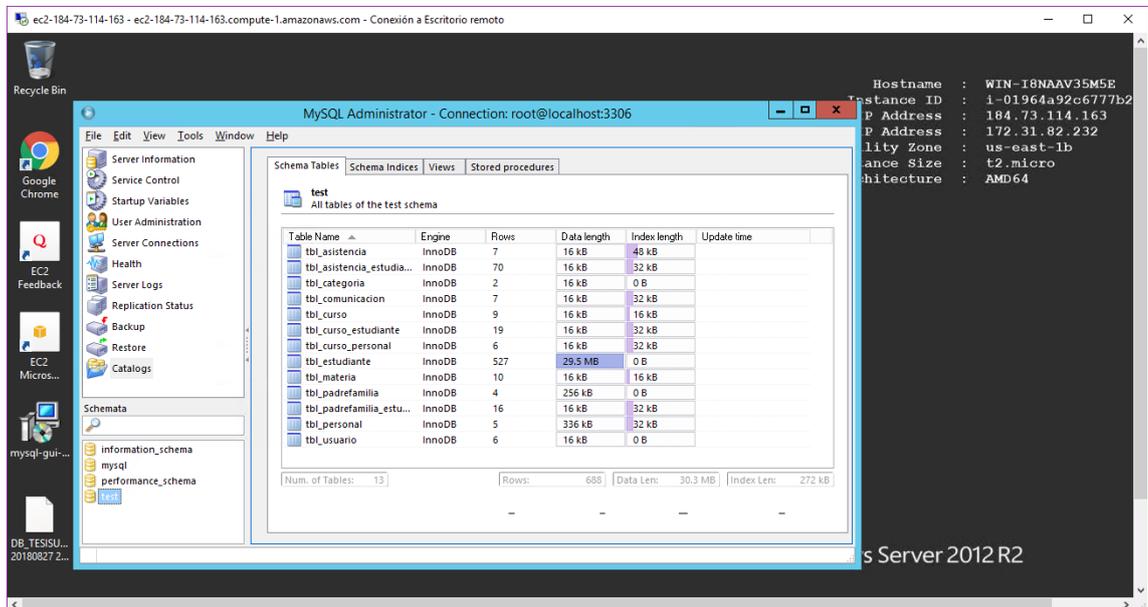


Figura 47: Administrador de MySQL

Fuente: Elaboración propia

Imágenes de entrenamiento en S3 y colección de rostros en Rekognition

Las imágenes de los rostros de personas, en este caso los estudiantes, se subieron a un Bucket de S3 creado con el nombre "tesisunir", para su posterior tratamiento en Rekognition, en este último se creó una colección de rostros denominada tesisUnirRostros para compararlos con los rostros identificados en la imagen capturada para la asistencia.

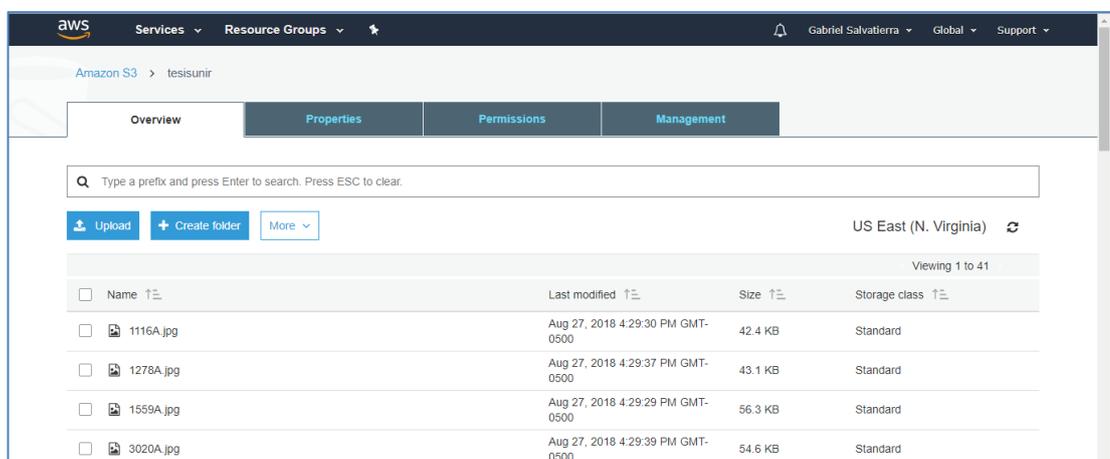


Figura 48: Bucket de imágenes con las fotografías de cada estudiante

Fuente: Elaboración propia

La imagen de cada estudiante, debe ser analizada por Rekognition para que reconozca y extraiga los rasgos faciales creando un identificador mediante un algoritmo de AWS, este identificador es el que luego se comparará con los rostros identificados en la imagen de asistencia.

Para comprender un poco más de lo que sucede en el servidor de reconocimiento, se representa en la siguiente figura este proceso.

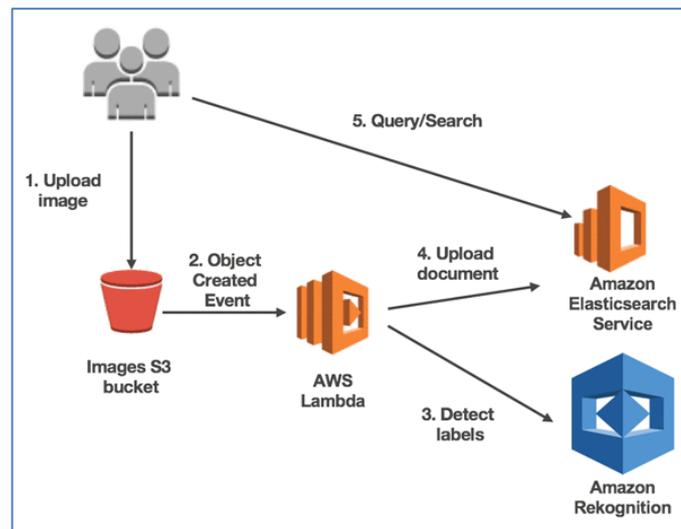


Figura 49: Funcionamiento de Rekognition

Fuente: AWS - Rekognition

Una vez concluida la implementación, se da paso a la realización de las respectivas pruebas y evaluación de los módulos correspondientes al sistema.

Categoría	Descripción
1 PERIODO 2017 - 2018	INICIO 14/04/2017
2 PERIODO 2018 - 2019	INICIO 20/04/2018

Curso	Descripción
1 SEGUNDO AÑO	35 ESTUDIANTES
2 TERCER AÑO	20 ESTUDIANTES
3 CUARTO AÑO	26 ESTUDIANTES
4 QUINTO AÑO	35 ESTUDIANTES
5 SEXTO AÑO	27 ESTUDIANTES
6 SEPTIMO AÑO	30 ESTUDIANTES
7 OCTAVO AÑO	24 ESTUDIANTES
8 NOVENO AÑO	CAPACIDAD 20 ESTUDIANTES

Imagen	Docente
	MENDOZA NAVARRETE KATHERINE JOHANNA
	SALVATIERRA TUMBACO GABRIEL GREGORIO

Imagen	Estudiante
	ALAVA PINARGOTE SILVIA DANIELA
	ERAS VELEZ LAURA MICEL
	FERNANDEZ PINARGOTE MAURICIO ANDRES
	GUERRERO LINO MICHAEL ALEJANDRO
	JIMENEZ CEVALLOS VICTOR ANDRES
	LOOR ZAMBRANO DIDIEL ARGENIS

Figura 50: Prueba del sistema, desplegado en una instancia de AWS

Fuente: Elaboración propia

7.2. Evaluación del sistema

La evaluación del sistema, se realizó tomando como referencia 40 preguntas seleccionadas y preestablecidas en el test SUMI (Software Usability Measurement Inventory), utilizado para la comprobación de la calidad de un producto software, centrado desde el punto de vista del usuario final. (SUMI, s.f.)

7.2.1. Usabilidad

El test de SUMI, es una de las formas más efectivas para medir la calidad y usabilidad de un sistema Web. Para conseguir resultados óptimos del test, el usuario debe completar las preguntas propuestas, mismas que tienen un rango de respuesta (De acuerdo, indeciso, desacuerdo). El objetivo de la realización del test, es la obtención de respuestas concretas y medibles, que permitan determinar en qué falla el software para posteriormente corregirlo.

7.2.1.1. Sujetos experimentales

Los sujetos experimentales para el test, fueron de varios tipos de acuerdo al departamento al que pertenecen y el rol que desempeñan en la Institución. Los docentes, el personal administrativo y consejería estudiantil, integrados por personas con profesión de Licenciados en Ciencias de la educación, Psicólogos, Economistas e Ingenieros en Sistemas informáticos, manipularon el sistema y realizaron el test SUMI; con lo cual se obtuvo un total de 15 personas evaluadas.

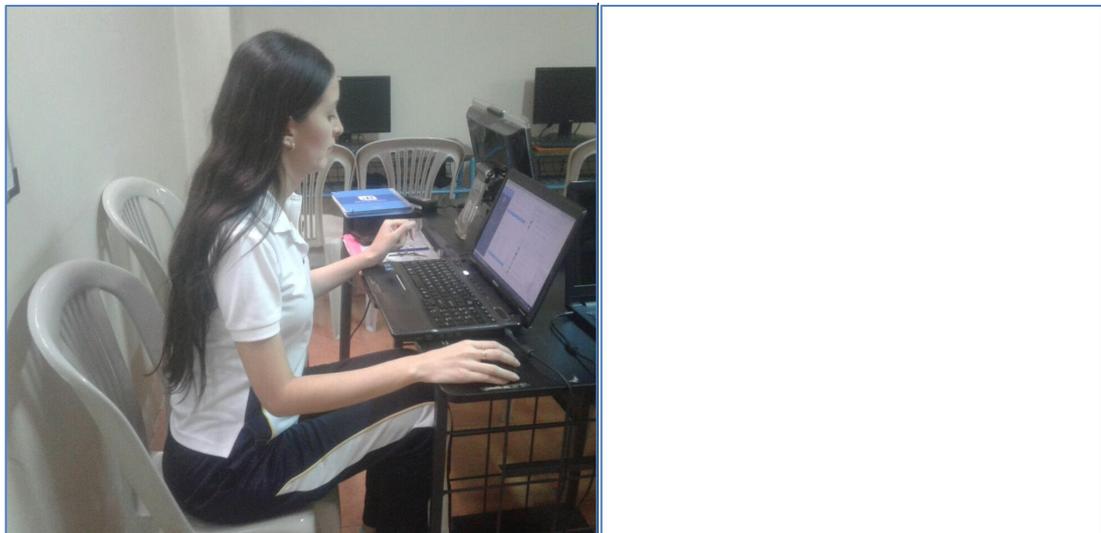


Figura 51: Manipulación del sistema y aplicación de test SUMI

Fuente: Elaboración propia

7.2.1.2. Formato de test

Las preguntas aplicadas se resumen en el anexo III.

7.2.1.3. Aplicación del test

En la **tabla 32**, se pueden visualizar los resultados que se obtuvieron en la aplicación del test; hay que recordar que el test solo es aplicable a los usuarios que manipularon el software. El total de usuarios que realizaron el test fue de 15 personas que laboran en la institución. Cada pregunta está representada con el número de respuesta en el rango establecido.

Tabla 32: Descripción de caso de uso gestionar docente

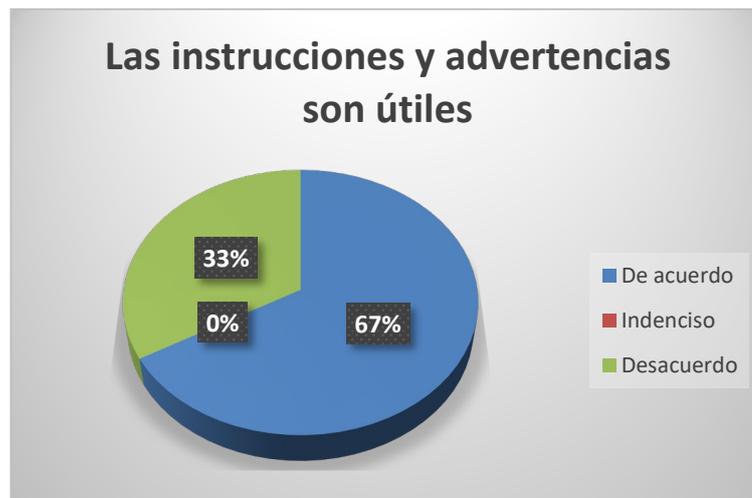
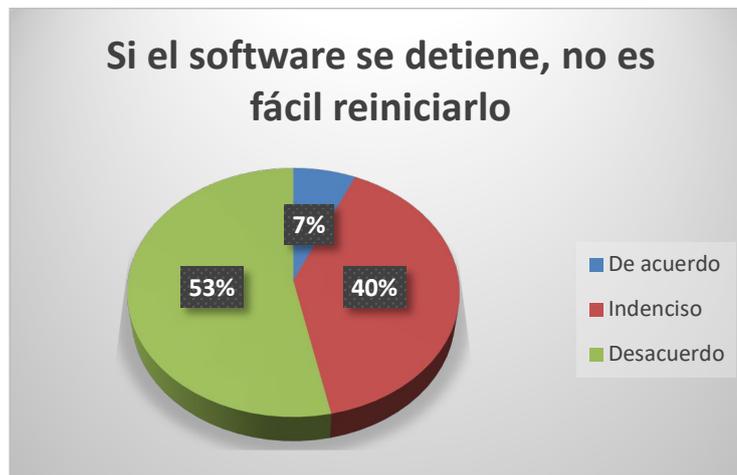
	Pregunta	De acuerdo	Indeciso	Desacuerdo
1	Este software responde muy lentamente a las entradas	0	0	15
2	Las instrucciones y advertencias son útiles	10	0	5
3	El software se detiene inesperadamente	0	3	12
4	Es muy difícil aprender a manejar el software	2	0	13
5	En algunos momentos no sé cuál es el siguiente paso en este software	0	0	15
6	Disfruto el tiempo que dedico a usar este software	11	4	0
7	La ayuda que brinda el software no es de mucha ayuda	1	2	12
8	Si el software se detiene, no es fácil reiniciarlo	1	6	8
9	Toma mucho tiempo aprender las funciones del software	0	0	15
10	A veces me pregunto si estoy usando la función correcta	3	2	10
11	Trabajar con este software es muy satisfactorio	14	1	0
12	La forma en que se presenta la información del sistema es clara y comprensible	15	0	0
13	Me siento más seguro si solo uso algunas funciones familiares	4	8	3
14	Trabajar con este software es mentalmente estimulante	10	4	1
15	Nunca hay suficiente información en la pantalla cuando se necesita	0	0	15
16	Yo siento que controlo el software cuando lo utilizo	15	0	0
17	Creo que este software es inconsistente	0	1	14
18	No me gustaría usar este software todos los días	1	3	11
19	Yo puedo entender y trabajar con la información que me brinda el software	15	0	0
20	El software es frustrante	0	0	15
21	Hay mucho que leer antes de poder usar el software	0	0	15
22	La velocidad de este software es lo suficientemente rápido	15	0	0

23	Habido momentos en el uso de este software cuando me he sentido bastante tenso o tensa	0	1	14
24	La organización de los menús parece bastante lógico	15	0	0
25	El software me permite ser económico en las pulsaciones de teclas	12	1	2
26	Aprender a usar nuevas funciones es difícil	2	1	12
27	Hay demasiados pasos requeridos para conseguir trabajar	3	2	10
28	Creo que este software a veces me ha dado un dolor de cabeza	0	0	15
29	Los mensajes de error no son adecuados	15	0	0
30	Es fácil hacer que el software haga exactamente lo que quiero	14	1	0
31	Nunca aprenderé a usar todo lo que se ofrece en este software	0	5	10
32	El software no siempre ha hecho lo que esperaba	1	3	11
33	El software se presenta de una manera muy atractiva	15	0	0
34	Es fácil moverse de una tarea a otra	15	0	0
35	El fácil olvidar cómo hacer las cosas con este software	1	1	13
36	Este software ocasionalmente se comporta de una manera que no se puede entender	1	2	12
37	Este software es realmente muy incómodo	2	3	10
38	Es fácil ver de un vistazo cuales son las opciones que ofrece el software	15	0	0
39	Obtener y guardar información es difícil en este software	0	0	15
40	Tengo que buscar asistencia la mayoría de la veces cuando uso este software	3	0	12

Fuente: SUMI

7.2.1.4. Resultado del test

Para no hacer tan engorroso este apartado, solo se escogieron cinco preguntas que permitieron determinar qué es lo que no está bien en el software y cómo se debe mejorar. Además, se realizó el gráfico estadístico de las preguntas escogidas, también, se analizó los puntos fuertes del sistema para determinar un nivel de satisfacción por parte de los usuarios.

Pregunta 2: Las instrucciones y advertencias son útiles*Figura 52: Resultado pregunta 2 SUMI**Fuente: Elaboración propia***Pregunta 8:** Si el software se detiene, no es fácil reiniciarlo*Figura 53: Resultado pregunta 8 SUMI**Fuente: Elaboración propia***Pregunta 13:** Me siento más seguro si solo uso algunas funciones familiares*Figura 54: Resultado pregunta 13 SUMI**Fuente: Elaboración propia*

Pregunta 25: El software me permite ser económico en las pulsaciones de teclas



Figura 55: Resultado pregunta 25 SUMI

Fuente: Elaboración propia

Pregunta 37: Este software es realmente muy incómodo



Figura 56: Resultado pregunta 37 SUMI

Fuente: Elaboración propia

Tomando en cuenta todas las respuestas del test SUMI se puede concluir que:

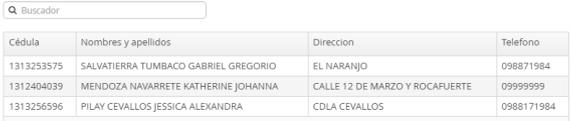
- El software responde muy bien a las entradas de datos por parte de los usuarios lo que genera satisfacción.
- La mayor parte de los usuarios considera que las instrucciones y advertencias que muestra la aplicación, son muy útiles, ayudando en el manejo del sistema.
- Se considera que el software no se detiene inesperadamente, incluso en la carga masiva de datos el framework Vaadin, renderiza de a poco el contenido mientras se visualiza, con el fin de que el usuario no perciba lentitud en la carga.
- Los usuarios se sintieron conformes con el manejo del sistema, considerándolo como un sistema sencillo e intuitivo de utilizar, ya que se familiarizaron rápidamente con los menús y botones de acción de cada vista.

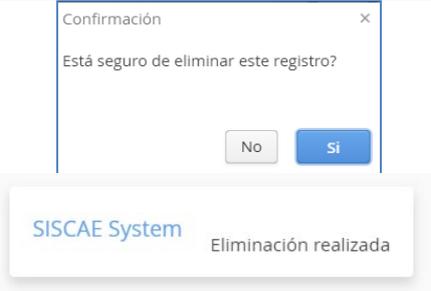
- La forma como se presenta la información, es muy clara y comprensible para los participantes.
- Todos los usuarios, quedaron satisfechos con la organización del sistema. Además, les brinda seguridad y confianza al utilizarlo, también, no necesito mucha asistencia técnica en el uso de la funcionalidades del mismo.
- Lo que el sistema debe mejorar, es en las guías, ya en determinados momentos ciertos usuarios no sabía que hacer o que proseguía.

7.2.2. Trazabilidad de requisitos

7.2.2.1. Requisitos funcionales

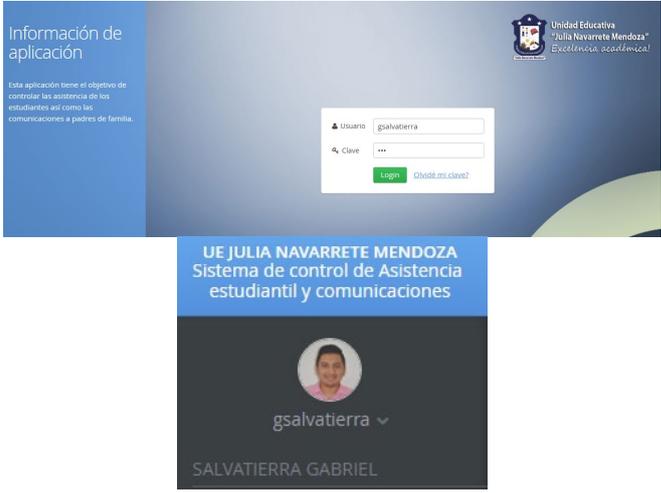
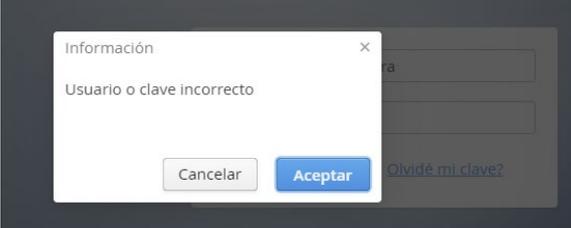
Tabla 33: Caso de prueba RF01 Gestionar usuario

Caso de prueba: RF01 Gestionar usuario			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperados del sistema	Cumple	Resultado obtenido
Ingreso			
1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Personal UE 3.- Ingresar la información en los campos obligatorios 4.- Llenar la información de usuario 5.- Dar clic en el botón guardar	Poder ingresar los usuarios que utilizaran el sistema	Si	
Consulta			
1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Personal UE 3.- Dar clic en el botón buscar	Poder consultar la información de los usuarios registrados en el sistema	Si	
Modificación			
1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Personal UE 3.- Dar clic en el botón buscar 4.- Seleccionar el usuario 5.- Dar clic en el botón guardar	Poder modificar la información de los usuarios registrados en el sistema	Si	
Eliminación			
1.- Ingresar a	Poder eliminar	Si	

<p>http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Personal UE 3.- Dar clic en el botón buscar 4.- Seleccionar el usuario 5.- Dar clic en el botón eliminar 6.- Aceptar el mensaje de confirmación</p>	<p>la información de los usuarios registrados en el sistema</p>		
--	---	--	--

Fuente: Elaboración propia

Tabla 34: Caso de prueba RF02 autenticar y autorizar usuario

Caso de prueba: RF02 Autenticar y autorizar usuario			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperado del sistema	Cumple	Resultado obtenido
Autenticar			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Escribir el usuario y clave 3.- Dar clic en el botón login</p>	<p>Ingresar al sistema mediante el usuario y clave</p>	<p>Si</p>	
No autenticar			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Escribir el usuario y clave 3.- Dar clic en el botón login</p>	<p>Impedir el ingreso a usuarios no registrados</p>	<p>Si</p>	

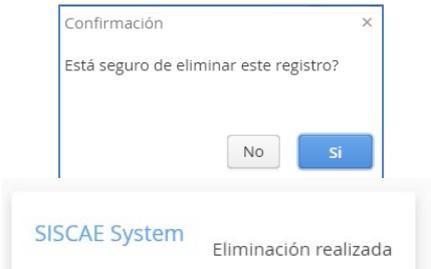
Fuente: Elaboración propia

Tabla 35: Caso de prueba RF03 Gestionar estudiante

Caso de prueba: RF03 Gestionar estudiante			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperado del sistema	Cumple	Resultado obtenido
Ingreso			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Estudiante 3.- Ingresar la información en los campos obligatorios 5.- Dar clic en el botón guardar</p>	<p>Poder ingresar los estudiantes en el sistema</p>	<p>Si</p>	
Consulta			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Estudiante 3.- Dar clic en el botón buscar</p>	<p>Poder consultar los estudiantes en el sistema</p>	<p>Si</p>	
Modificación			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Estudiante 3.- Dar clic en el botón buscar 4.- Seleccionar el estudiante 5.- Dar clic en el botón guardar</p>	<p>Poder modificar los estudiantes en el sistema</p>	<p>Si</p>	
Eliminación			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Eliminar 3.- Dar clic en el botón buscar 4.- Seleccionar el estudiante 5.- Dar clic en el botón eliminar 6.- Aceptar el mensaje de confirmación</p>	<p>Poder eliminar los estudiantes en el sistema</p>	<p>Si</p>	

Fuente: Elaboración propia

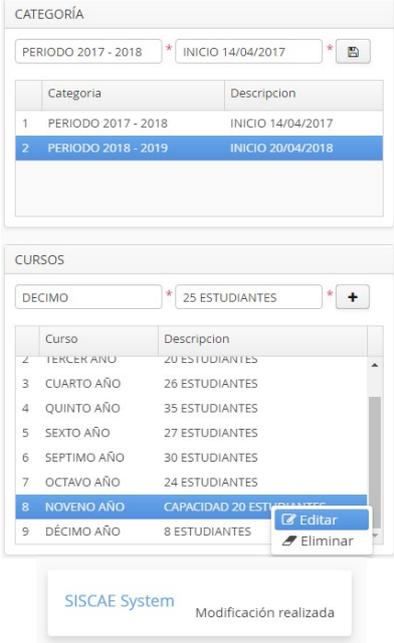
Tabla 36: Caso de prueba RF04 gestionar padre de familia

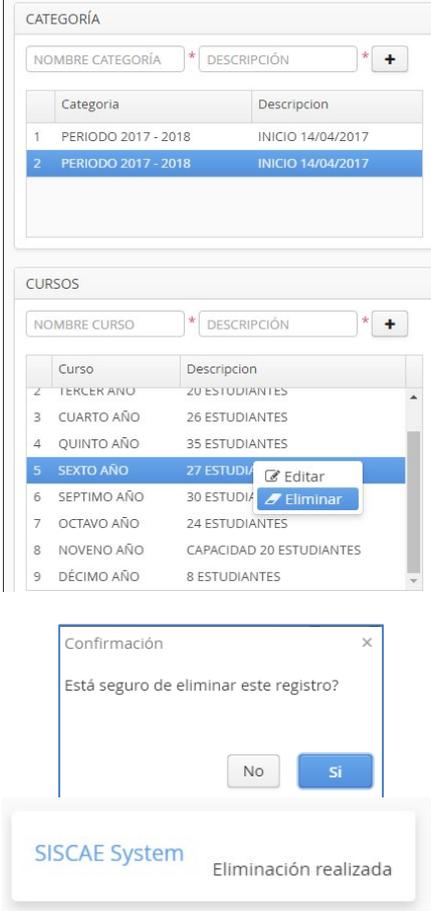
Caso de prueba: RF04 Gestionar padre de familia			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperados del sistema	Cumple	Resultado obtenido
Ingreso			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Padre familia 3.- Ingresar la información en los campos obligatorios 5.- Dar clic en el botón guardar</p>	<p>Poder ingresar a los Padres familia en el sistema</p>	<p>Si</p>	
Consulta			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Padre familia 3.- Dar clic en el botón buscar</p>	<p>Poder consultar a los Padres familia en el sistema</p>	<p>Si</p>	
Modificación			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Padre familia 3.- Dar clic en el botón buscar 4.- Seleccionar el estudiante 5.- Dar clic en el botón guardar</p>	<p>Poder modificar a los Padres familia en el sistema</p>	<p>Si</p>	
Eliminación			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Padre familia 3.- Dar clic en el botón buscar 4.- Seleccionar el estudiante 5.- Dar clic en el botón eliminar 6.- Aceptar el mensaje de confirmación</p>	<p>Poder eliminar a los Padres familia en el sistema</p>	<p>Si</p>	

Fuente: Elaboración propia

Tabla 37: Caso de prueba RF05 gestionar cursos

Caso de prueba: RF05 Gestionar cursos			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperado del sistema	Cumple	Resultado obtenido
Ingreso categoría			
1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Categorías y cursos 3.- Ingresar la información de la categoría (nombre, descripción) 5.- Dar clic en el botón (+)	Poder ingresar las categorías de cursos	Si	
Modificación categoría			
1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Categorías y cursos 3.- Clic derecho sobre la categoría a modificar y escoger editar 4.- Realizar modificación 5.- Dar clic en guardar	Poder modificar las categorías de cursos	Si	
Eliminación de categoría			
1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Categorías y cursos 3.- Clic derecho sobre la categoría a eliminar 4.- Confirmar el mensaje	Poder eliminar las categorías de cursos	Si	

Ingreso curso			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Categorías y cursos 3.- Seleccionar la categoría 5.- Ingresar la información correspondiente a nombre y descripción 6.- Dar clic en el botón (+)</p>	<p>Poder ingresar los cursos pertenecientes a una categoría</p>	<p>Si</p>	
Modificación curso			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Categorías y cursos 3.- Seleccionar la categoría 3.- Clic derecho sobre el curso a modificar y escoger editar 4.- Realizar modificación 5.- Dar clic en guardar</p>	<p>Poder modificar los cursos pertenecientes a una categoría</p>	<p>Si</p>	
Eliminación de curso			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Categorías y cursos 3.- Seleccionar la categoría 4.- Clic derecho sobre el</p>	<p>Poder eliminar los cursos pertenecientes a una categoría</p>	<p>Si</p>	

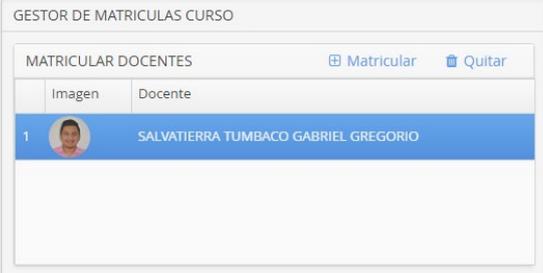
<p>curso a modificar y escoger eliminar 5.- Confirmar mensaje</p>			
---	--	--	---

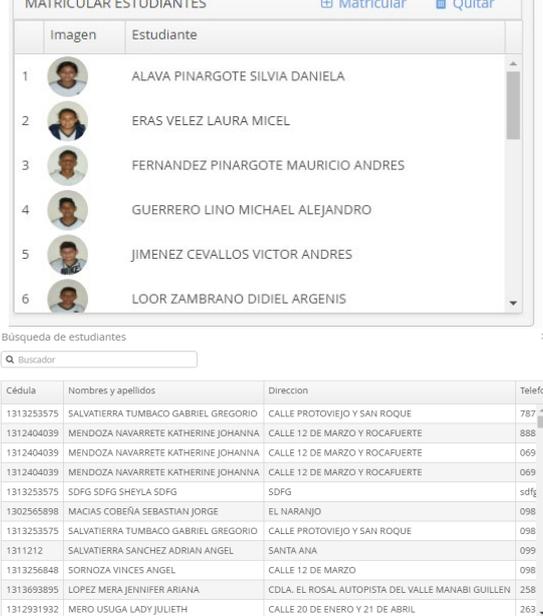
Matricular docente en el curso

<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Categorías y cursos 3.- Seleccionar la categoría 4.- Seleccionar el curso 5.- Dar clic en el botón matricular del panel docente 6.- Seleccionar al docente</p>	<p>Permitir la matriculación de docentes en un determinado curso</p>	<p>Si</p>	
--	--	-----------	--

Quitar docente de un curso

<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Categorías y cursos 3.- Seleccionar la categoría 4.- Seleccionar el curso</p>	<p>Permitir la anulación de una matrícula de docente en un determinado curso</p>	<p>Si</p>	
---	--	-----------	--

<p>5.- Seleccionar al docente 5.- Dar clic en el botón quitar del panel docente</p>			
---	--	--	--

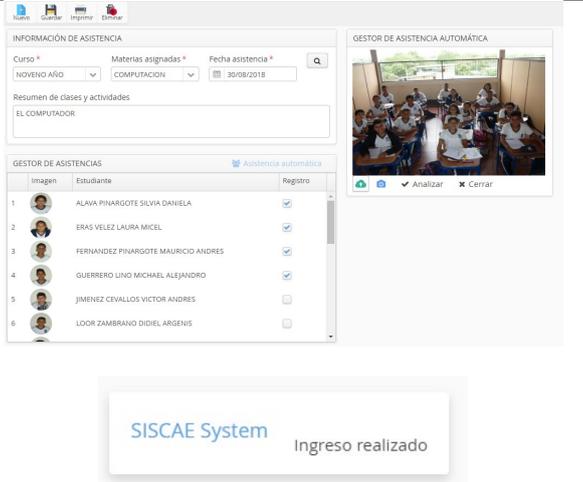
<p>Matricular estudiante en el curso</p>																																																			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Categorías y cursos 3.- Seleccionar la categoría 4.- Seleccionar el curso 5.- Dar clic en el botón matricular del panel estudiante 6.- Seleccionar el estudiante</p>	<p>Permitir la matriculación de estudiantes en un determina curso</p>	<p>Si</p>	 <table border="1" data-bbox="917 929 1460 1164"> <thead> <tr> <th>Cédula</th> <th>Nombres y apellidos</th> <th>Direccion</th> <th>Telefon</th> </tr> </thead> <tbody> <tr> <td>1313253575</td> <td>SALVATIERRA TUMBACO GABRIEL GREGORIO</td> <td>CALLE PROTOVIEJO Y SAN ROQUE</td> <td>787</td> </tr> <tr> <td>1312404039</td> <td>MENDOZA NAVARRETE KATHERINE JOHANNA</td> <td>CALLE 12 DE MARZO Y ROCAFUERTE</td> <td>888</td> </tr> <tr> <td>1312404039</td> <td>MENDOZA NAVARRETE KATHERINE JOHANNA</td> <td>CALLE 12 DE MARZO Y ROCAFUERTE</td> <td>069</td> </tr> <tr> <td>1312404039</td> <td>MENDOZA NAVARRETE KATHERINE JOHANNA</td> <td>CALLE 12 DE MARZO Y ROCAFUERTE</td> <td>069</td> </tr> <tr> <td>1313253575</td> <td>SDFG SDGF SHEYLA SDGF</td> <td>SDGF</td> <td>sdgf</td> </tr> <tr> <td>1302565898</td> <td>MACIAS COBEÑA SEBASTIAN JORGE</td> <td>EL NARANJO</td> <td>098</td> </tr> <tr> <td>1313253575</td> <td>SALVATIERRA TUMBACO GABRIEL GREGORIO</td> <td>CALLE PROTOVIEJO Y SAN ROQUE</td> <td>098</td> </tr> <tr> <td>1311212</td> <td>SALVATIERRA SANCHEZ ADRIAN ANGEL</td> <td>SANTA ANA</td> <td>099</td> </tr> <tr> <td>1313256848</td> <td>SORNOZA VINCES ANGEL</td> <td>CALLE 12 DE MARZO</td> <td>096</td> </tr> <tr> <td>1313693895</td> <td>LOPEZ MERA JENNIFER ARIANA</td> <td>CDLA. EL ROSAL AUTOPISTA DEL VALLE MANABI GUILLEN</td> <td>258</td> </tr> <tr> <td>1312931932</td> <td>MERO USUGA LADY JULIETH</td> <td>CALLE 20 DE ENERO Y 21 DE ABRIL</td> <td>263</td> </tr> </tbody> </table>	Cédula	Nombres y apellidos	Direccion	Telefon	1313253575	SALVATIERRA TUMBACO GABRIEL GREGORIO	CALLE PROTOVIEJO Y SAN ROQUE	787	1312404039	MENDOZA NAVARRETE KATHERINE JOHANNA	CALLE 12 DE MARZO Y ROCAFUERTE	888	1312404039	MENDOZA NAVARRETE KATHERINE JOHANNA	CALLE 12 DE MARZO Y ROCAFUERTE	069	1312404039	MENDOZA NAVARRETE KATHERINE JOHANNA	CALLE 12 DE MARZO Y ROCAFUERTE	069	1313253575	SDFG SDGF SHEYLA SDGF	SDGF	sdgf	1302565898	MACIAS COBEÑA SEBASTIAN JORGE	EL NARANJO	098	1313253575	SALVATIERRA TUMBACO GABRIEL GREGORIO	CALLE PROTOVIEJO Y SAN ROQUE	098	1311212	SALVATIERRA SANCHEZ ADRIAN ANGEL	SANTA ANA	099	1313256848	SORNOZA VINCES ANGEL	CALLE 12 DE MARZO	096	1313693895	LOPEZ MERA JENNIFER ARIANA	CDLA. EL ROSAL AUTOPISTA DEL VALLE MANABI GUILLEN	258	1312931932	MERO USUGA LADY JULIETH	CALLE 20 DE ENERO Y 21 DE ABRIL	263
Cédula	Nombres y apellidos	Direccion	Telefon																																																
1313253575	SALVATIERRA TUMBACO GABRIEL GREGORIO	CALLE PROTOVIEJO Y SAN ROQUE	787																																																
1312404039	MENDOZA NAVARRETE KATHERINE JOHANNA	CALLE 12 DE MARZO Y ROCAFUERTE	888																																																
1312404039	MENDOZA NAVARRETE KATHERINE JOHANNA	CALLE 12 DE MARZO Y ROCAFUERTE	069																																																
1312404039	MENDOZA NAVARRETE KATHERINE JOHANNA	CALLE 12 DE MARZO Y ROCAFUERTE	069																																																
1313253575	SDFG SDGF SHEYLA SDGF	SDGF	sdgf																																																
1302565898	MACIAS COBEÑA SEBASTIAN JORGE	EL NARANJO	098																																																
1313253575	SALVATIERRA TUMBACO GABRIEL GREGORIO	CALLE PROTOVIEJO Y SAN ROQUE	098																																																
1311212	SALVATIERRA SANCHEZ ADRIAN ANGEL	SANTA ANA	099																																																
1313256848	SORNOZA VINCES ANGEL	CALLE 12 DE MARZO	096																																																
1313693895	LOPEZ MERA JENNIFER ARIANA	CDLA. EL ROSAL AUTOPISTA DEL VALLE MANABI GUILLEN	258																																																
1312931932	MERO USUGA LADY JULIETH	CALLE 20 DE ENERO Y 21 DE ABRIL	263																																																

<p>Quitar estudiante de un curso</p>			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Categorías y cursos 3.- Seleccionar la categoría 4.- Seleccionar el curso 5.- Seleccionar al estudiante 5.- Dar clic en el botón quitar del panel estudiante</p>	<p>Permitir la anulación de una matrícula de estudiante en un determinado curso</p>	<p>Si</p>	

Fuente: Elaboración propia

Tabla 38: Caso de prueba RF06 gestionar asistencia estudiantil

Caso de prueba: RF06 Gestionar asistencia estudiantil			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperado del sistema	Cumple	Resultado obtenido
Registro de asistencia estudiantil			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Asistencia estudiantil 3.- Seleccionar el curso 4.- Seleccionar la materia 5.- Seleccionar la fecha 6.- Ingresar el resumen de las clases y actividades 7.- Checkear la asistencia a cada uno de los estudiantes 8.- Dar clic en el botón guardar.</p>	<p>Registrar las asistencias de los estudiantes</p>	<p>Si</p>	
Modificación de asistencia estudiantil			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Asistencia estudiantil 3.- Seleccionar la asistencia a modificar en el popover 4.- Ingresar el resumen de las clases y actividades 5.- Checkear la asistencia a cada uno de los estudiantes 6.- Dar clic en el botón guardar.</p>	<p>Poder modificar las asistencias registradas de los estudiantes</p>	<p>Si</p>	
Registro de asistencia estudiantil por reconocimiento facial			

<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Asistencia estudiantil 3.- Seleccionar el curso 4.- Seleccionar la materia 5.- Seleccionar la fecha 6.- Ingresar el resumen de las clases y actividades 7.- Dar clic en el botón asistencia automática 8.- Tomar o subir la foto 9.- Dar clic en el botón analizar 10.- Dar clic en el botón guardar.</p>	<p>Registrar las asistencias de los estudiantes utilizando el reconocimiento facial.</p>	<p>Si</p>	
--	--	-----------	--

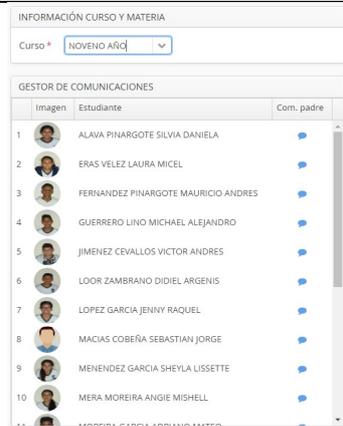
Fuente: Elaboración propia

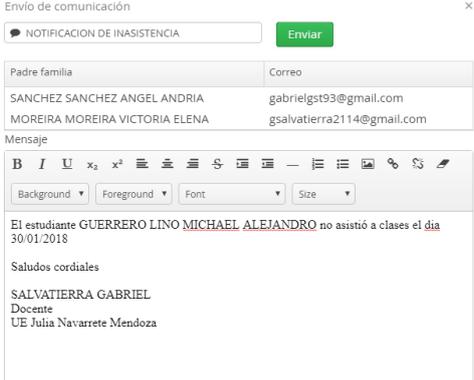
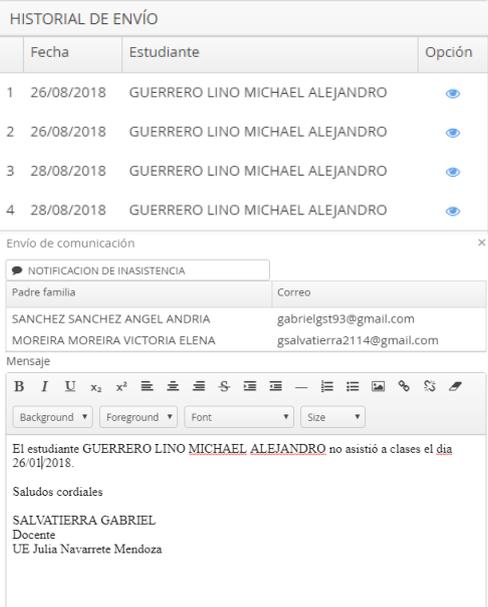
Tabla 39: Caso de prueba RF07 gestionar reportes

Caso de prueba: RF07 Gestionar reportes			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperados del sistema	Cumple	Resultado obtenido
Registro de asistencia estudiantil			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú reportes 3.- Seleccionar las fechas desde hasta 4.- Seleccionar el tipo de reporte 5.- Dar clic en generar</p>	<p>Generar informes de asistencias de estudiantes</p>	<p>No, por cuestión de tiempo aún no se ha implementado pero queda para trabajo futuro</p>	

Fuente: Elaboración propia

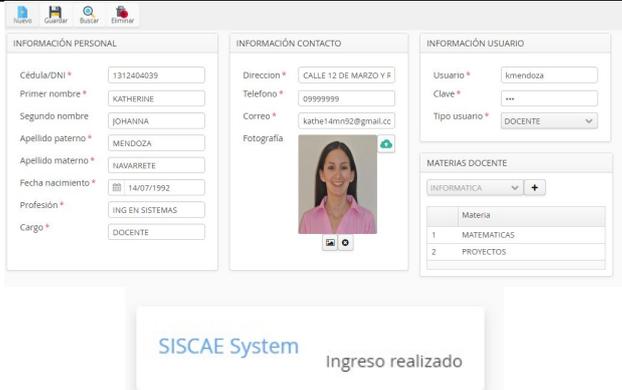
Tabla 40: Caso de prueba RF08 gestionar comunicaciones

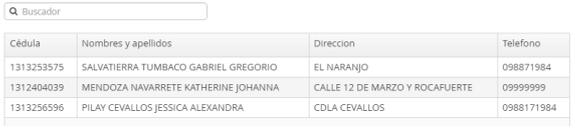
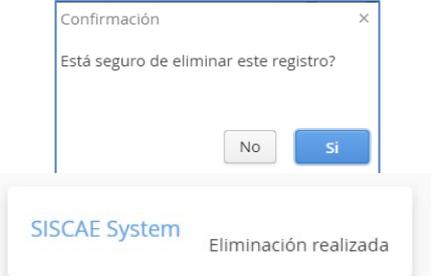
Caso de prueba: RF08 Gestionar comunicaciones			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperados del sistema	Cumple	Resultado obtenido
Registro y envío de comunicaciones			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú comunicaciones 3.- Seleccionar el curso 4.- Dar clic en el botón comunicación del estudiante. 5.- Redactar comunicación 6.- Presionar el botón enviar</p>	<p>Enviar comunicaciones al correo electrónico de los padres de familia</p>	<p>Si</p>	

			
Consulta de comunicaciones enviadas			
<p>1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú comunicaciones 4.- Dar clic en el botón vista de comunicación del estudiante del panel historial de envío.</p>	<p>Enviar comunicaciones al correo electrónico de los padres de familia</p>	<p>Si</p>	

Fuente: Elaboración propia

Tabla 41: Caso de prueba RF09 Gestionar docente

Caso de prueba: RF09 Gestionar docente			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperado del sistema	Cumple	Resultado obtenido
<p>Ingreso 1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Personal UE 3.- Ingresar la información en los campos obligatorios 4.- Llenar la información de usuario 5.- Seleccionar el tipo de usuario 6.- Agregar materias 7.- Dar clic en el botón guardar</p>	<p>Poder ingresar los docentes que utilizaran el sistema para controlar las asistencias de estudiantes y comunicaciones a padres de familia</p>	<p>Si</p>	

Consulta			
1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Personal UE 3.- Dar clic en el botón buscar	Poder consultar la información de los docentes registrados en el sistema	Si	
Modificación			
1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Personal UE 3.- Dar clic en el botón buscar 4.- Seleccionar el docente 5.- Dar clic en el botón guardar	Poder modificar la información de los docentes registrados en el sistema	Si	
Eliminación			
1.- Ingresar a http://siscae.uejuname.edu.ec 2.- Seleccionar el menú Personal UE 3.- Dar clic en el botón buscar 4.- Seleccionar el docente 5.- Dar clic en el botón eliminar 6.- Aceptar el mensaje de confirmación	Poder eliminar la información de los docentes registrados en el sistema	Si	

Fuente: Elaboración propia

7.2.2.2. Requisitos no funcionales

Tabla 42: Caso de prueba RNF01 espacio y rendimiento

Caso de prueba: RNF01 espacio y rendimiento			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperados del sistema	Cumple	Resultado obtenido
1.- Verificar tiempo de respuesta en producción 2.- Comprobar el crecimiento de la base de datos con la información de todos los estudiantes de la Unidad Educativa	Tiempo que el usuario espera por solicitud	Si	El tiempo estará condicionado por el Internet de la Institución, sin embargo, se obtuvo un promedio de 2 segundos por cada solicitud, a excepción de la subida de imágenes en la cual el tiempo fue de 20 a 30 segundos. La base de datos incrementó en un 20% ya que las imágenes de los estudiantes están almacenadas en dicha base.

Fuente: Elaboración propia

Tabla 43: Caso de prueba RNF02 usabilidad

Caso de prueba: RNF02 usabilidad			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperado del sistema	Cumple	Resultado obtenido
1.- Verificar si el sistema es fácil de usar e intuitivo	Usuarios manejen el sistema sin mucha ayuda técnica	Si	Según los resultados del test SUMI, el sistema es intuitivo y fácil de usar, por un usuario con conocimientos básicos en informática.

*Fuente: Elaboración propia***Tabla 44:** Caso de prueba RNF03 seguridad

Caso de prueba: RNF03 seguridad			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperado del sistema	Cumple	Resultado obtenido
1.- Verificar que la autenticación y autorización se realizan mediante usuario y clave	No permitir el ingreso a usuario que no estén registrados	Si	El cumplimiento de este requisito, se puede verificar en el caso de prueba RF02
1.- Implementación de certificado de seguridad	Verificar que la comunicación con el servidor es segura	No	No se encuentra implementado ningún certificado de seguridad.
1.- Encriptación de claves de usuario	Verificar en la base de datos que las claves se guardan cifradas	No	Por implementar

*Fuente: Elaboración propia***Tabla 45:** Caso de prueba RNF04 portabilidad

Caso de prueba: RNF04 portabilidad			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperado del sistema	Cumple	Resultado obtenido
1.- Verificar que la aplicación web funciona en la mayoría de navegadores.	Que todos los componentes de la app funcionen correctamente en los navegadores más populares	Si	Prueba exitosa en navegadores como: Google Chrome, Mozilla Firefox, Opera, Safari e Internet Explorer

Fuente: Elaboración propia

Tabla 46: Caso de prueba RNF05 operacional

Caso de prueba: RNF05 operacional			
PROCEDIMIENTO DE PRUEBA			
Pasos	Resultados esperados del sistema	Cumple	Resultado obtenido
1.- La aplicación se despliega correctamente en el servidor Tomcat	Despliegue de la app en un servidor Tomcat instalado en Windows Server	Si	El servidor desplegó el archivo compilado (.war) de la aplicación y creó las carpetas necesarias para el funcionamiento.

Fuente: Elaboración propia

8. Conclusiones y trabajo futuro

8.1. Conclusiones

Terminado el trabajo de fin de master se concluye que:

- El análisis y levantamiento de requisitos, es una de las fases con mayor importancia dentro del proceso de desarrollo de software, ya que es aquí donde se establecen las bases para el modelado y correcta construcción del sistema.
- La utilización de frameworks de diseño tipo RIA como Vaadin, permiten crear interfaces de usuario de una forma más rápida, ya que no se necesitan conocimientos profundos de HTML o CSS, porque todo se realiza con programación en Java, como si fuera una aplicación de escritorio; esto optimizó en gran medida el tiempo utilizado en el desarrollo, con lo cual se pudo obtener un sistema en producción y de buena calidad.
- Cuando se trabaja con Java, utilizar frameworks ORM como Hibernate y de persistencia como JPA, garantizan la correcta gestión de las transacciones CRUD y optimizan las conexiones hacia el servidor. Además, permite la creación automática de la base de datos y facilitan la migración a cualquier gestor que admita el framework. Así mismo, generó un ahorro de tiempo en el desarrollo de la aplicación de este TFM, ya que solo se crearon los modelos en Java y el ORM junto a JPA se encargaron de la administración de las sentencias SQL.
- La combinación de herramientas de cloud computing como EC2, S3 y Rekognition de AWS, permitieron gestionar el despliegue del sistema garantizando el acceso a la información e incrementando su funcionalidad, a través de la implementación del reconocimiento facial. Además, las opciones del servidor de reconocimiento, hacen que el software sea escalable y pueda incorporar en un futuro nuevas opciones respecto al análisis de imágenes y videos.
- Usar una metodología de desarrollo como RUP, beneficia a todos involucrados en el desarrollo del software, porque permite tener los procesos bien definidos, gestiona la obtención de requisitos y los cambios de estos; además, es adaptable a la necesidad de cada empresa.

- La cámara que captura la foto de los estudiantes en el curso, debe estar ubicada en un lugar estratégico con un ángulo y posición precisa, para que tenga la capacidad de capturar todo el lugar del salón de clases, evitando los puntos ciegos.
- Los algoritmos de reconocimiento facial de Amazon Web Services, son tan eficaces que solo se necesitó una foto de cada estudiante para el entrenamiento del servidor de reconocimiento.

8.2. Líneas de trabajo futuro

Dadas las conclusiones alcanzadas en la terminación de este trabajo de fin de master, se establecen las siguientes líneas de trabajo futuro, pudiendo ser estas a corto mediana o largo plazo.

Corto plazo:

- Añadir reportes estadísticos y consolidados mensuales, que ayuden en la toma de decisiones en los procesos educativos.
- Revisar el diseño y presentación de las interfaces de usuario, para evitar la sensación de rutina y monotonía. Además, implementar el diseño responsive o adaptativo a diferentes pantallas de dispositivos.
- Incorporar textos de ayudas, que permitan guiar al usuario en las diferentes funcionalidades del sistema.

Mediano plazo:

- Trabajar con cursos más numerosos, para analizar la eficiencia del servidor de reconocimiento.
- Integrar perfiles de usuario como estudiante y padre de familia, para dar acceso a nuevos módulos como inscripciones y consultas de notas.
- Implementación de una cámara HD para la captura de la foto.

Largo plazo:

- Diseño e implementación de una aplicación móvil, que permita gestionar notificaciones en tiempo real y consulta de datos académicos. Además, se puede integrar al sistema de reconocimiento, para utilizar los recursos de este dispositivo como la cámara fotográfica, y así capturar la foto que posteriormente será analizada por el servidor de reconocimiento.
- Incorporar más cámaras y que la aplicación tenga opción de seleccionarlas para que el personal encargado del departamento de consejería estudiantil pueda visualizar todos los cursos sin la necesidad de ir uno a uno.

- Implementar el análisis de video, utilizando el servicio Kinesis de aws, que permite el procesamiento de datos en tiempo real.

Bibliografía

Amazon. (2018). *Amazon Rekognition*. Obtenido de

https://aws.amazon.com/es/rekognition/?nc2=h_a1

Amazon. (2018). *Amazon S3*. Obtenido de https://aws.amazon.com/es/s3/?nc2=h_m1

Amazon. (2018). *Cloud Computing con Amazon Web Services*. Obtenido de

<https://aws.amazon.com/es/what-is-aws/>

Bertino Salazar, C., & Orozco Alzate, S. (s.f.). *Universidad Tecnológica de Pereira*. Obtenido de

<http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/6946/0062483995S161.pdf?sequence=1>

EFE, A. (25 de Octubre de 2017). *CHINA TECNOLOGÍA*. Obtenido de

<https://www.efe.com/efe/america/tecnologia/una-universidad-china-utilizara-el-reconocimiento-facial-para-controlar-la-asistencia/20000036-3418428>

Espinoza Olguin, D. E., & Jorquera Guillen, P. I. (2015). *Pontificia Universidad Católica de*

Valparaíso. Obtenido de http://opac.pucv.cl/pucv_txt/txt-1000/UCD1453_01.pdf

Gonzales Marin, L. K. (2016). *Universidad Mayor de San Andrés*. Obtenido de

<http://repositorio.umsa.bo/bitstream/handle/123456789/7662/T.3104.pdf?sequence=1&isAllowed=y>

Hidalgo Veliz, R. N. (Diciembre de 2017). *Universidad de Guayaquil*. Obtenido de

<http://repositorio.ug.edu.ec/bitstream/redug/24298/1/B-CISC-PTG.1413.Hidalgo%20Veliz%20Raul%20Nassib.pdf>

Hidalgo Véliz, R. N. (2017). *Universidad de Guayaquil*. Obtenido de

<http://repositorio.ug.edu.ec/bitstream/redug/24298/1/B-CISC-PTG.1413.Hidalgo%20Veliz%20Raul%20Nassib.pdf>

IBM. (s.f.). *IBM Developerworks*. Obtenido de Rational Unified Process:

https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf

Jacobson, I., Booch, G., & Rumbaugh, J. (2000). *El proceso unificado de desarrollo de software*. Madrid: Pearson Education S.A.

Mysql. (2018). *Dev Mysql*. Obtenido de <https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>

Pressman, R. (2010). *Ingeniería de software, un enfoque práctico*. México D.F.: McGraw-Hill Companies.

Rational, t. s. (23 de Julio de 2005). *Rational Unified Process*. Obtenido de Best Practices for Software Development Teams:
https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf

SAP. (s.f.). Obtenido de <https://www.sap.com/products/powerdesigner-data-modeling-tools.html>

Sommerville, I. (2011). *Ingeniería de software*. Mexico: Pearson Educación.

SUMI. (s.f.). *Software Usability Measurement Inventory*. Obtenido de <http://sumi.uxp.ie/en/index.php>

Vaadin. (2018). *VAADIN FRAMEWORK*. Obtenido de <https://vaadin.com/framework>

Anexos

Anexo I. Artículo

Máster Universitario en Ingeniería de Software y Sistemas Informáticos

Desarrollo de un sistema de control de asistencia estudiantil mediante reconocimiento facial

GABRIEL GREGORIO SALVATIERRA TUMBACO
gabrielgregorio.salvatierra131@comunidadunir.net

Universidad Internacional de la Rioja – Escuela de ingeniería
Avda. Gran Vía Rey Juan Carlos I núm. 41, 28002
Logroño (La Rioja) España – info@unir.net

Abstract - The control of assistance, is a process that every institution, company or organization must perform daily to monitor the proper work of the staff. However, this becomes a problem when the number of people to be controlled is large and there are no mechanisms available to help speed up the process of taking care. The present work of end of master, shows the development of a system of control of assistances using the facial recognition, through the integration of services Cloud of Amazon Web Services. The main objective is to control the attendance record of the students to de High School "Julia Navarrete Mendoza", in a massive way, that is to say, that by means of photographs taken to the students of the course, the faces are recognized and identify to the people presents. To control the software development process, the RUP (Unified Rational Process) methodology was used, taking into account each of the activities and disciplines that guarantee the creation of a quality system. In the end a functional product was obtained, which allows the registration of the information of those involved in student assistance and communications to parents, successfully achieving the objective of this work, which is the use of Cloud services for mass facial recognition.

Keywords: Facial recognition, massive assistance, usability test, Rational Unified Process.

I. INTRODUCCIÓN

El control de asistencias, es un proceso que toda institución, Empresa u organización debe realizar diariamente para vigilar la correcta labor del personal. Sin embargo, este se convierte en problema cuando el número de personas a controlar es grande y no se dispone de mecanismos que ayuden a agilizar el proceso de toma de asistencias.

El control de asistencia es tan importante como necesario en cualquier ámbito; por ejemplo, en eventos masivos donde se escapa de las manos este control, se pueden implementar herramientas tecnológicas que ayuden a realizar este fin; y es ahí donde aparecen los sistemas biométricos (huellas, reconocimiento facial, reconocimiento de voz, entre otros).

En el campo educativo, la asistencia estudiantil es fundamental para controlar el proceso de estudio de un alumno. Por tal motivo, se ha desarrollado un sistema experimental implementado en la Unidad Educativa "Julia Navarrete Mendoza" que permita llevar fácilmente el control de las asistencias de estudiantes en la Institución, utilizando tecnologías de Cloud para

el reconocimiento facial, almacenamiento y consulta de información.

El objetivo principal, de la investigación es poder controlar las asistencias en forma masiva, es decir, que mediante fotografías tomadas a los estudiantes del curso, se reconozcan los rostros e identifique a las personas que se encuentran presentes.

Para controlar el proceso de desarrollo del software, se utilizó la metodología RUP (Proceso Racional Unificado), teniendo en cuenta cada una de las actividades y las disciplinas que garantiza la creación de un sistema de calidad. Además, la metodología está dirigida por casos de uso, centrado en la arquitectura de componentes y adaptable a la necesidad del proyecto y requisitos cambiantes de usuario. [1]

Se determinó, que el lenguaje de programación a utilizar sea Java con el framework Vaadin para las interfaces de usuario e Hibernate con JPA para las transacciones CRUD de la base de datos que lo gestionará MySQL.

Los requisitos del software, se validaron con un plan de pruebas por cada requisito funcional y no funcional; además, para garantizar la usabilidad, integridad y disponibilidad del sistema, se utilizó el test SUMI centrado desde la perspectiva del usuario final.

Al final se obtuvo un producto funcional, que permite el registro de la información de los involucrados en las asistencias estudiantiles y comunicaciones a padres de familia, logrando con éxito el objetivo de este trabajo, que es la utilización de servicios Cloud para el reconocimiento facial masivo.

II. TRABAJO RELACIONADO

Existen en la actualidad sistemas que permiten realizar el reconocimiento facial para la toma de asistencia estudiantil. Los casos analizados a continuación utilizan diferentes formas para cumplir con dicho objetivo.

Estudiantes de la Universidad Veracruzana de México, propusieron mediante la ayuda de Matlab un registro de asistencia para estudiante utilizando reconocimiento facial, donde según esta investigación, los aspectos importantes que se deben considerar son: 1) el equipo se debe encontrar a una altura de 1.50 m.; 2) el estudiante se debe colocar atrás de una línea marcada en el suelo frente a la cámara, aproximadamente a unos 60 cm. de distancia; 3) los estudiantes deben ir pasando conforme van llegando, con la finalidad que el registro de asistencia se diera de forma natural y; 4) cada estudiante debe ver detenidamente a la cámara durante tres segundos.

Un proyecto similar fue realizado en la Pontificia Universidad Católica de Valparaíso Chile, donde se desarrolló un sistema de

reconocimiento facial mediante el uso de las librerías OpenCV/EmguCV para optimizar su eficiencia y rendimiento. Dispone a su vez de una base de datos local, la cual permite almacenar las fotografías tomadas a los usuarios para su posterior uso al momento de reconocer a una persona mediante la utilización de diferentes algoritmos y poder identificar así cual es el más eficiente. [2]

La Universidad de China en Pekín, cuenta con un sistema de reconocimiento facial que permite controlar la asistencia de estudiantes. El funcionamiento de este software consiste, en que antes de iniciar la clase, el docente activa el sistema instalado en una tableta y toma fotos de los estudiantes que pasan frente de la cámara, “uno por uno” para después comparar las fotos con las almacenadas en una base de datos. [3] Dicho sistema se basa en la tecnología de reconocimiento facial de la plataforma de inteligencia artificial del gigante chino de Internet, Baidu.

En el Ecuador, Estudiantes de la Universidad de Guayaquil realizaron una propuesta de un prototipo de sistema para el control de asistencia estudiantil de un paralelo mediante procesamiento digital de imágenes. Utilizaron el lenguaje de programación Python, la librería OpenCV, y para poder alimentar la base de datos realizaron 20 capturas por cada estudiante, para de esta forma obtener expresiones diferentes, considerando dos ambientes (mucha y poca iluminación) para poder comprobar la efectividad del algoritmo utilizado.

Según Hidalgo Véliz, (2017) menciona que se recomienda un ambiente controlado, una correcta orientación de la cámara y la posición de la cara de los estudiantes en el salón, para lograr mayor precisión en el reconocimiento facial. [4]

III. ANÁLISIS Y DESARROLLO DEL SISTEMA

a) Perspectiva del producto

El sistema de control de asistencias estudiantiles Siscac como se lo ha denominado, será desarrollado para trabajar en un entorno Web, y desplegado en una instancia de EC2 de Amazon Web Services, accesible desde cualquier parte mediante el Internet.

b) Características de los usuarios del sistema

Tipo de usuario	Administrador
Formación	Conocimientos intermedios en informática, manejo de computadora e Internet.
Actividades	Autenticarse en el sistemas, controlar y manejar del sistema, administración de usuarios, gestión de cursos y paralelos, periodos lectivos. Control total de SISCAC

Tabla 1: Característica de usuario administrador

Tipo de usuario	Secretaria
Formación	Administración de procesos educativos
Actividades	Autenticarse en el sistema, generación de reportes de asistencias, registro de estudiantes y padres de familia, envío de notificaciones a correos, registro de nuevos docentes y gestión de cursos.

Tabla 2: Característica de usuario secretaria

Tipo de usuario	Docente
Formación	Conocimiento básico en informática, manejo de Internet y aplicaciones.
Actividades	Autenticarse en el sistema, realizar el registro de asistencias estudiantiles, sea manual o automáticamente (Reconocimiento facial).

Tabla 3: Característica de usuario docente

c) Identificación de requisitos

Cada uno de los requisitos, se analizaron de acuerdo a la necesidad del usuario y recursos disponibles para el desarrollo de la aplicación, los requisitos funcionales forman parte de los procesos que realiza el sistema, mientras que los no funcionales hacen referencia al entorno donde funcionará el sistema. Estos requisitos se resumen en la tabla 4.

Tipo de requisito	Usuario	Requisito
Funcionales	Administrador	Gestionar usuario, autenticar y autorizar, gestionar estudiante, gestionar padre de familia, gestionar cursos, gestionar asistencia estudiantil, gestionar informes, gestionar comunicaciones, gestionar personal UE
	Secretaria, docente	Gestionar usuario, autenticar y autorizar, gestionar estudiante, gestionar padre de familia, gestionar asistencia estudiantil, gestionar informes, gestionar comunicaciones
No funcionales	Administrador	Autenticarse en el sistemas, controlar y manejar del sistema, administración de usuarios, gestión de cursos y paralelos, periodos lectivos. Control total de SISCAC

Tabla 4: Resumen de requisitos funcionales y no funcionales

d) Interfaces

Las interfaces, son los conectores entre componentes para que puedan tener comunicación e integración entre ellos, a continuación se describen en la tabla 5, cada una de las interfaces relacionadas con el tipo y la descripción de cada una de ellas.

Tipo de interfaz	Descripción
Interfaz de usuario	Conjunto de ventanas, imágenes, botones, enlaces, entre otros. Cada interfaz se creó de acuerdo a las necesidades de las personas involucradas en el desarrollo.
Interfaz de hardware cliente	Procesador de segunda generación, memoria ram superior a 512 Mb, adaptador de Ethernet, teclado, mouse y monitor.
Interfaz de hardware servidor	80 Gb de disco duro, memoria ram de 1 Gb, procesador de cuarta generación. La virtualización es aceptable y es como se implementó este software utilizando servicios Cloud de AWS
Interfaz de software cliente	Sistema operativo Windows XP o superior con un navegador instalado, recomendado Chrome y Mozilla
Interfaz software del servidor	Servidor Tomcat, máquina virtual de Java y MySQL

Tabla 5: Interfaces para el funcionamiento del sistema

e) Arquitectura implementada

Java posee un estándar para la arquitectura de aplicaciones web, por tal motivo y tomando en cuenta este estándar la arquitectura implementada es de cuatro capas. Además, Sommerville (2011) señala que, las nociones de separación e independencia son fundamentales para el diseño arquitectónico porque permiten localizar cambios. [5]

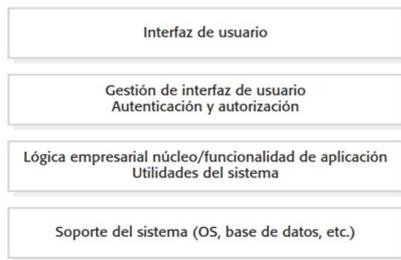


Figura 1: Arquitectura genérica en 4 capas
Fuente: Ingeniería de software, Sommerville 2011

f) Diagrama arquitectónico del sistema

Para conocer el funcionamiento de forma general de la aplicación, se presenta un diseño arquitectónico de los procesos que realiza el sistema.

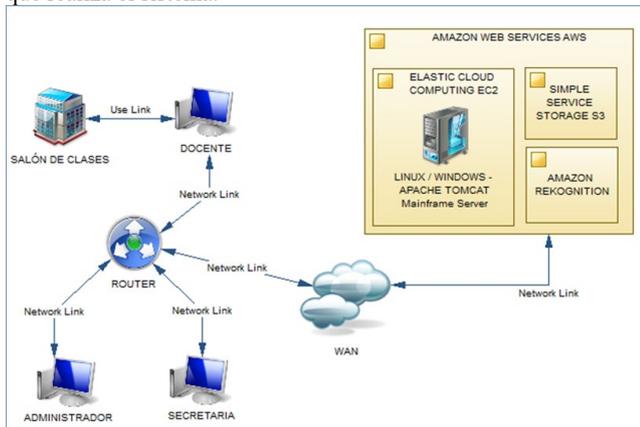


Figura 2: Arquitectura genérica en 4 capas
Fuente: Elaboración propia.

La figura 2, muestra cómo está constituido el software, y cómo interactúan los componentes del mismo. El proceso de toma de asistencias empieza capturando la foto del salón de clases, que es enviada a un servidor Tomcat instalado en una instancia de Windows Server desplegada en EC2, que se encargará en enviar la foto al almacenamiento S3 para que posteriormente el servidor de reconocimiento llamado Rekognition analice los rostros que tienen en la imagen y los compare con los rostros almacenados; todos estos servicios pertenecen a Amazon Web Services.

g) Vista general del proceso asistencia

Para registrar las asistencias de los estudiantes de un determinado curso, el sistema muestra al usuario (docente, secretaria, administrador) los cursos en los que se encuentra asignado; posteriormente el usuario selecciona el curso y elige si desea registrar la asistencia masiva o individual, en caso de seleccionar la primera el sistema inicia la cámara o la subida de archivos; obtenida la foto se envía al servidor de reconocimiento en espera de los identificadores de los rostros reconocidos en la imagen. Además, el usuario puede seleccionar la opción asistencia individual y marcarla de forma tradicional, la idea fue implementar los dos procesos tradicional y automático la figura 3 presenta este proceso.

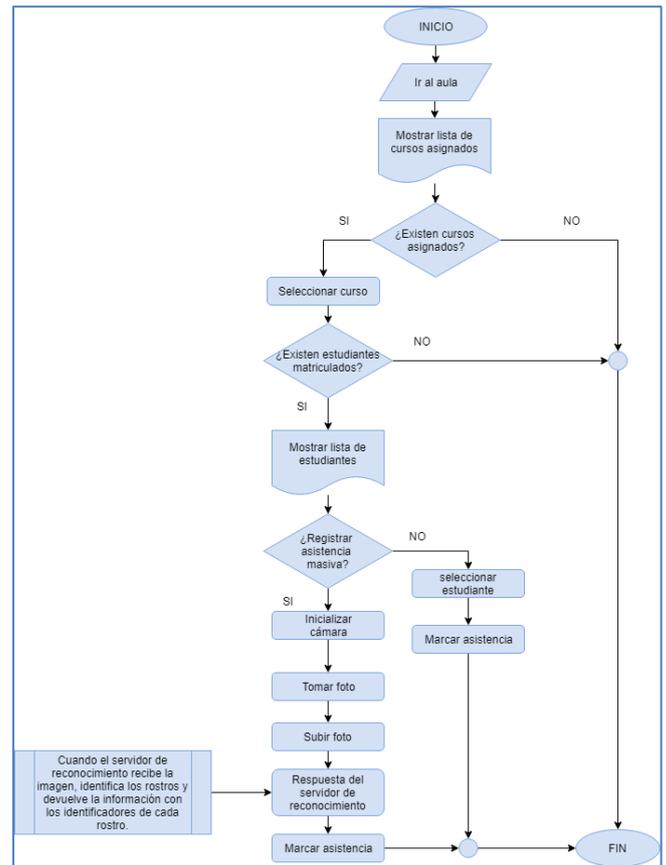


Figura 3: Proceso actual de registro de asistencias masivas o individuales
Fuente: Elaboración propia

h) Vista de despliegue



Figura 4: Diagrama de despliegue
Fuente: Elaboración propia

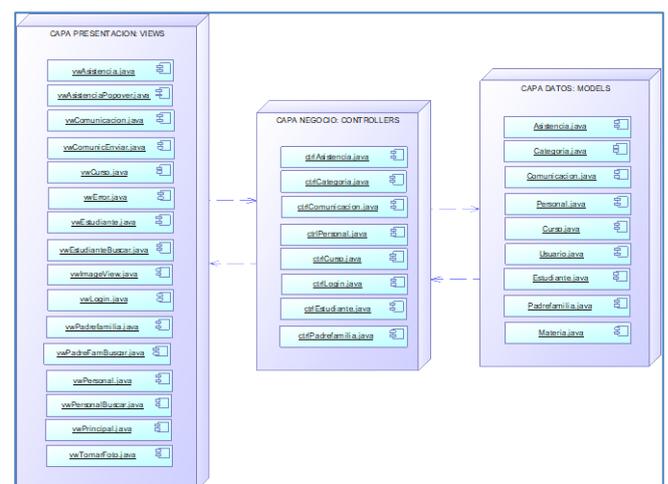


Figura 5: Diagrama de componentes del sistema
Fuente: Elaboración propia

La figura 5, correspondiente al diagrama de componentes, indica cómo está compuesta internamente la aplicación. La capa modelo, es la encargada gestionar los datos del sistema; y para conseguir este objetivo se ha considerado utilizar el framework

Hibernate junto con JPA (persistencia de datos), que garantizará las transacciones realizadas a la base de datos como inserciones, actualizaciones, eliminación y recuperación de la información. Además, generará automáticamente el diseño físico de la base de datos sin la necesidad de utilizar un administrador de SGDB.

Una vez creada la capa de modelos, utilizando Hibernate y las anotaciones de JPA se obtiene el diseño final de la base de datos y sus relaciones, como se presenta en la figura 6.

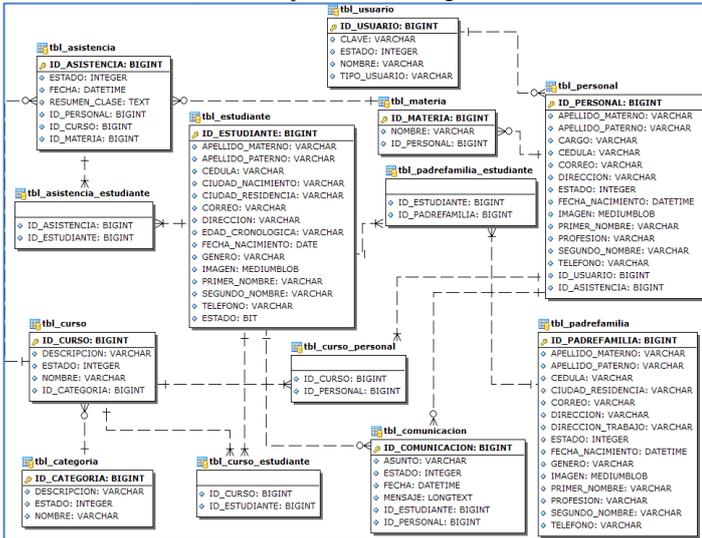


Figura 6: Diagrama final de base de datos
Fuente: Elaboración propia

IV. IMPLEMENTACIÓN

La aplicación se implementó en un servidor Tomcat 7, desplegado en una instancia de Windows Server 2017 en EC2 de AWS. Además, las imágenes de entrenamiento para el servidor de reconocimiento se subieron en un Bucket de S3 que también es un servicio proporcionado por AWS. Esta implementación se resume en las siguientes actividades.

a) Configuración de instancia EC2

La instancia desplegada, contiene el sistema operativo Windows server 2012 R2, se optó por este sistema ya que brinda la posibilidad de utilizar el escritorio remoto para su administración.



Figura 7: Instancia EC2, Windows Server 2012 R2
Fuente: AWS EC2, Instances

Para el funcionamiento de la aplicación, es necesario instalar en el servidor, el AWS CLI, JDK 8, Tomcat 7 (xampp paquete de software libre) y MySQL Server. Una vez instalados los programas, se debe configurar el AWS CLI para que funcione con la cuenta IAM que permite administrar el acceso a los servicios y recursos de AWS, dicho servicio proporcionó la información que se muestra en el en la figura 8 y que debe ser establecida en los archivos del configuración del AWS CLI utilizando el comando en la consola aws configure.

```
Secret access key: fcZ6oajWe0MJ2pbLh1Lw48aDdU0RmLBupYnBmeb5
Acces key ID: AKIAJBYVDSBP3VYBQQVA
region: us-east-1
```

Figura 8: Datos de configuración para AWS CLI
Fuente: AWS IAM

Además, se deben configurar las variables de entorno para que el SDK de AWS para Java, pueda ser utilizado de forma correcta.

```
set AWS_ACCESS_KEY_ID=AKIAJBYVDSBP3VYBQQVA
set AWS_SECRET_ACCESS_KEY=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
set AWS_REGION=us-east-1
```

Figura 9: Configuración de variables de entorno
Fuente: AWS IAM – elaboración propia

b) Compilación y despliegue de la aplicación

Para desplegar el sistema en el servidor Tomcat, es necesario compilar el código del mismo en un archivo .war, gracias a las bondades del IDE Eclipse, es posible realizar esta acción usando el asistente de exportación de proyectos.

El archivo generado por el asistente, deber ser copiado en la carpeta webapps del servidor Tomcat, este se encargará de desplegarlo y crear las carpetas necesarias para su funcionamiento.

c) Implementación de base de datos

Para este proceso, se debe tener el backup de la base de datos y utilizar el administrador de MySQL para importar los datos del script, aunque se puede realizar por consola, los administradores de tipo GUI facilitan en gran medida el trabajo.

d) Imágenes de entrenamiento en S3 y colección de rostros en Rekognition

Las imágenes de los rostros de personas, en este caso los estudiantes, se subieron a un Bucket de S3 creado con el nombre "tesisunir", para su posterior tratamiento en Rekognition, en este último se creó una colección de rostros denominada tesisUnirRostros para compararlos con los rostros identificados en la imagen capturada para la asistencia.

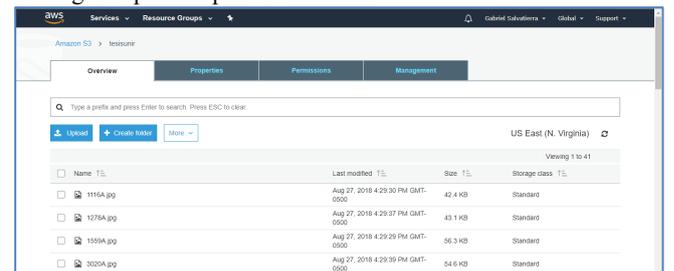


Figura 10: Bucket de imágenes con las fotografías de cada estudiante
Fuente: Elaboración propia

La imagen de cada estudiante, debe ser analizada por Rekognition para que reconozca y extraiga los rasgos faciales creando un identificador mediante un algoritmo de AWS, este identificador es el que luego se comparará con los rostros identificados en la imagen de asistencia.

Para comprender un poco más de lo que sucede en el servidor de reconocimiento, se representa en la siguiente figura este proceso.

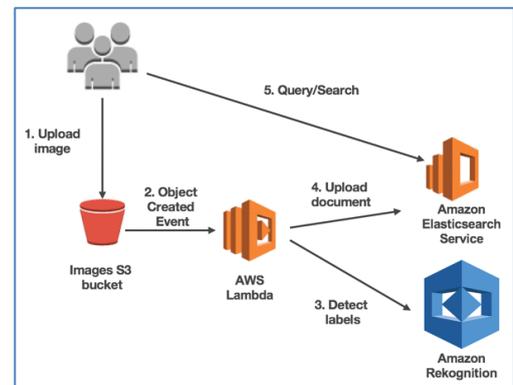


Figura 11: Funcionamiento de Rekognition
Fuente: AWS - Rekognition

Una vez concluida la implementación, se da paso a la realización de las respectivas pruebas y evaluación de los módulos correspondientes al sistema.

V. EVALUACIÓN DEL SISTEMA

La evaluación del sistema, se realizó tomando como referencia 40 preguntas seleccionadas y preestablecidas en el test SUMI(Software Usability Measurement Inventory), utilizado para la comprobación de la calidad de un producto software, centrado desde el punto de vista del usuario final. [6]

El test de SUMI, es una de las formas más efectivas para medir la calidad y usabilidad de un sistema Web. Para conseguir resultados óptimos del test, el usuario debe completar las preguntas propuestas, mismas que tienen un rango de respuesta (De acuerdo, indeciso, desacuerdo). El objetivo de la realización del test, es la obtención de respuestas concretas y medibles, que permitan determinar en qué falla el software para posteriormente corregirlo.

a) Sujetos experimentales y resultados de test.

Los sujetos experimentales para el test, fueron de varios tipos de acuerdo al departamento al que pertenecen y el rol que desempeñan en la Institución. Los docentes, el personal administrativo y consejería estudiantil, integrados por personas con profesión de Licenciados en Ciencias de la educación, Psicólogos, Economistas e Ingenieros en Sistemas informáticos, manipularon el sistema y realizaron el test SUMI; con lo cual se obtuvo un total de 15 personas evaluadas.

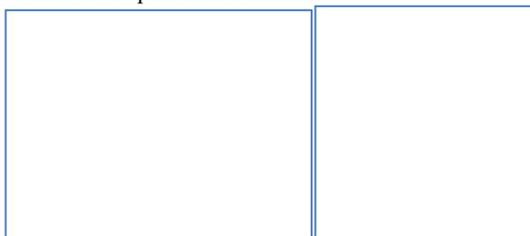


Figura 12: Manipulación del sistema y aplicación de test SUMI
Fuente: Elaboración propia

Tomando en cuenta todas las respuestas del test SUMI se puede concluir que:

- El software responde muy bien a las entradas de datos por parte de los usuarios lo que genera satisfacción.
- La mayor parte de los usuarios considera que las instrucciones y advertencias que muestra la aplicación, son muy útiles, ayudando en el manejo del sistema
- Se considera que el software no se detiene inesperadamente, incluso en la carga masiva de datos el framework Vaadin, renderiza de a poco el contenido mientras se visualiza, con el fin de que el usuario no perciba lentitud en la carga.
- Los usuarios se sintieron conformes con el manejo del sistema, considerándolo como un sistema sencillo e intuitivo de utilizar, ya que se familiarizaron rápidamente con los menús y botones de acción de cada vista.
- La forma como se presenta la información, es muy clara y comprensible para los participantes.
- Todos los usuarios, quedaron satisfechos con la organización del sistema. Además, les brinda seguridad y confianza al utilizarlo, también, no necesito mucha asistencia técnica en el uso de la funcionalidades del mismo.

- Lo que el sistema debe mejorar, es en las guías, ya en determinados momentos ciertos usuarios no sabía que hacer o que proseguía.

b) Evaluación de objetivo principal, reconocimiento facial

Cada uno de los módulos del sistema fue evaluado mediante la matriz de trazabilidad, que permitió comprobar el correcto funcionamiento de los componentes del sistema. Además, el cumplimiento del registro de asistencia por reconocimiento facial se alcanzó con éxito y se demuestra en la siguiente figura.

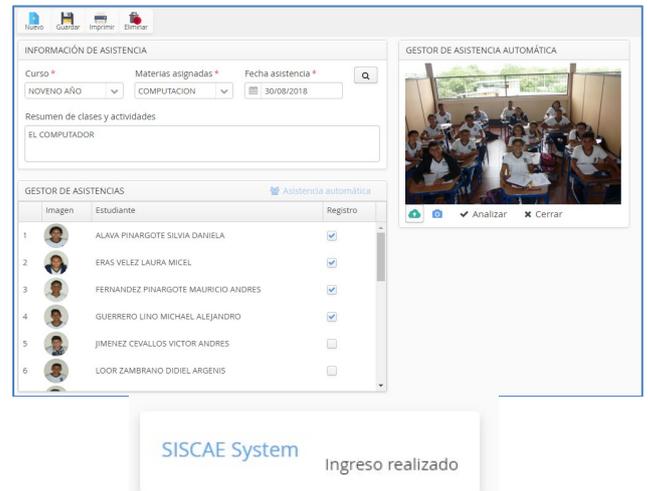


Figura 13: Registro de asistencia masiva por reconocimiento facial
Fuente: Elaboración propia

VI. CONCLUSIONES

El análisis y levantamiento de requisitos, es una de las fases con mayor importancia dentro del proceso de desarrollo de software, ya que es aquí donde se establecen las bases para el modelado y correcta construcción del sistema.

La utilización de frameworks de diseño tipo RIA como Vaadin, permiten crear interfaces de usuario de una forma más rápida, ya que no se necesitan conocimientos profundos de HTML o CSS, porque todo se realiza con programación en Java, como si fuera una aplicación de escritorio; esto optimizó en gran medida el tiempo utilizado en el desarrollo, con lo cual se pudo obtener un sistema en producción y de buena calidad.

Cuando se trabaja con Java, utilizar frameworks CRM como Hibernate y de persistencia como JPA, garantizan la correcta gestión de las transacciones CRUD y optimizan las conexiones hacia el servidor. Además, permite la creación automática de la base de datos y facilitan la migración a cualquier gestor que admita el framework. Así mismo, generó un ahorro de tiempo en el desarrollo de la aplicación de este TFM, ya que solo se crearon los modelos en Java y el CRM junto a JPA se encargaron de la administración de las sentencias SQL.

La combinación de herramientas de cloud computing como EC2, S3 y Rekognition de AWS, permitieron gestionar el despliegue del sistema garantizando el acceso a la información e incrementando su funcionalidad, a través de la implementación del reconocimiento facial. Además, las opciones del servidor de reconocimiento, hacen que el software sea escalable y pueda incorporar en un futuro nuevas opciones respecto al análisis de imágenes y videos.

Usar una metodología de desarrollo como RUP, beneficia a todos involucrados en el desarrollo del software, porque permite

tener los procesos bien definidos, gestiona la obtención de requisitos y los cambios de estos; además, es adaptable a la necesidad de cada empresa.

REFERENCIAS

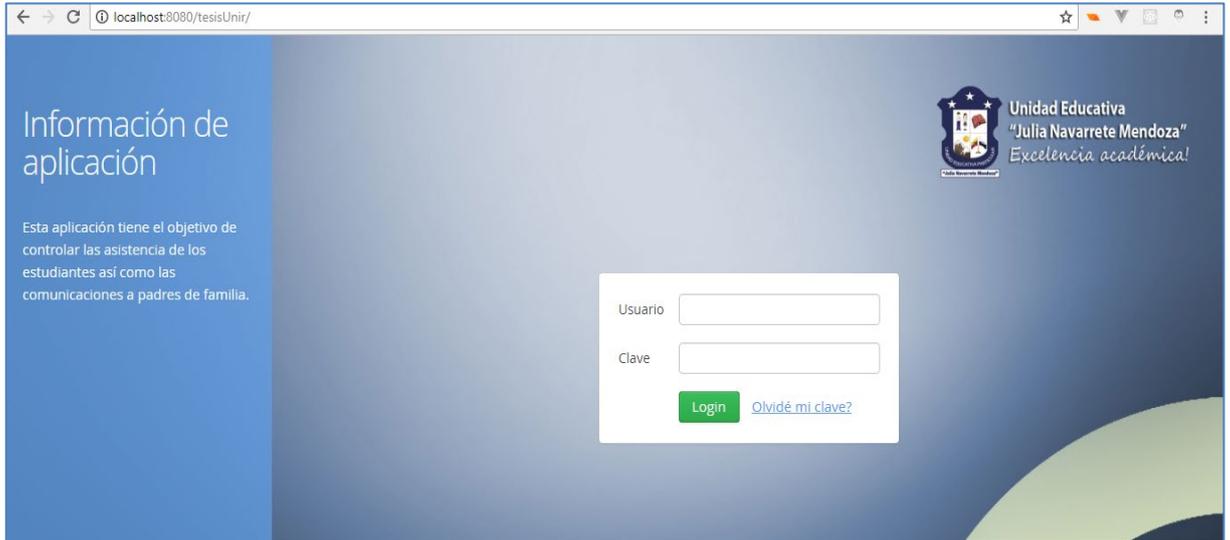
- [1] I. Jacobson, G. Booch and J. Rumbaugh, El proceso unificado de desarrollo de software. Madrid: Pearson Education S.A., 2000.
- [2] D. Espinoza Olguin and J. Peter Ignacio, "Pontificia Universidad Católica de Valparaíso", Pontificia Universidad Católica de Valparaíso, 2015. [Online]. Available: http://opac.pucv.cl/pucv_txt/txt-1000/UCD1453_01.pdf [Accessed: 07- Jul - 2018].
- [3]"Una universidad china utilizará el reconocimiento facial para controlar la asistencia", CHINA TECNOLOGÍA, 2018. [Online]. Available: <https://www.efe.com/efe/america/tecnologia/una-universidad-china-utilizara-el-reconocimiento-facial-para-controlar-la-asistencia/20000036-3418428>. [Accessed: 01- Sep- 2018].
- [4]R. Hidalgo Véliz, "Repositorio Universidad de Guayaquil", Universidad de Guayaquil, 2017. [Online]. Available: <http://repositorio.ug.edu.ec/bitstream/redug/24298/1/B-CISC-PTG.1413.Hidalgo%20Veliz%20Raul%20Nassib.pdf>. [Accessed: 01- Sep- 2018].
- [5] I. Sommerville, Ingeniería de software, 7th ed. Mexico: Pearson Educación, 2011.
- [6] Software Usability Measurement Inventory. [Online]. Available: <http://sumi.uxp.ie/en/index.php>. [Accessed: 01- Sep- 2018].



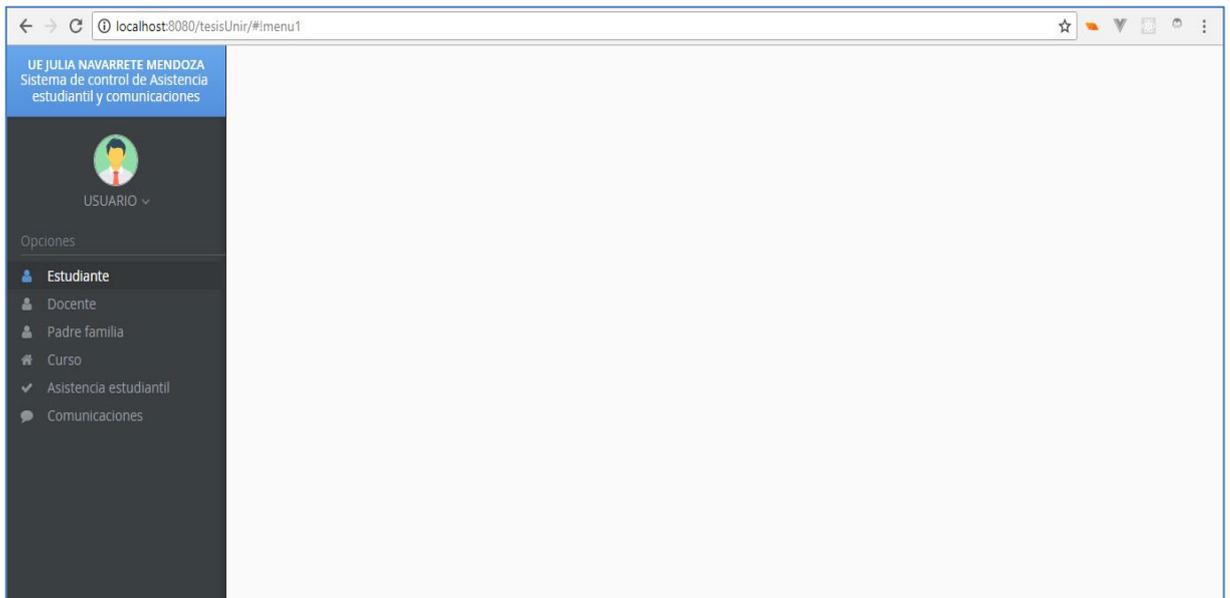
Gabriel Salvatierra es Ingeniero el Sistemas Informáticos, graduado en la Universidad Técnica de Manabí – Ecuador. En la actualidad es Analista de Tecnologías en la UE “Julia Navarrete Mendoza”. Sus campos de investigación son: Seguridad informática, cloud computing y los procesos de desarrollo de software.

Anexo II. Interfaces del sistema

Vista de login



Vista principal

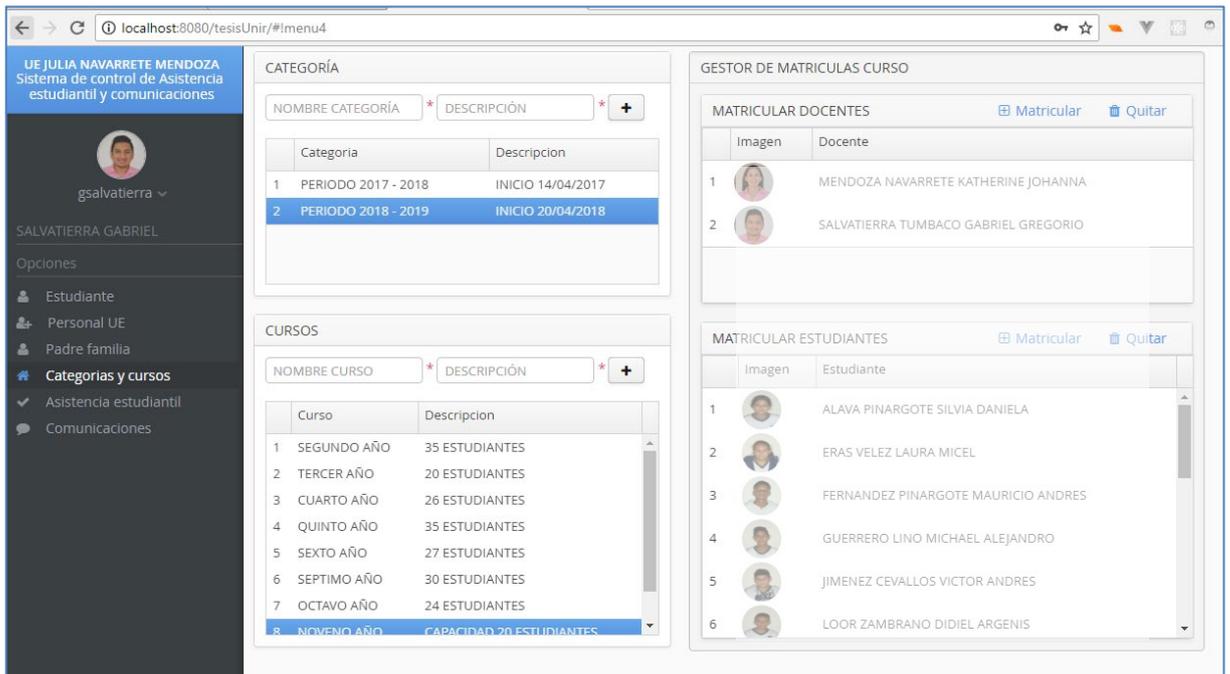


Vista Estudiante

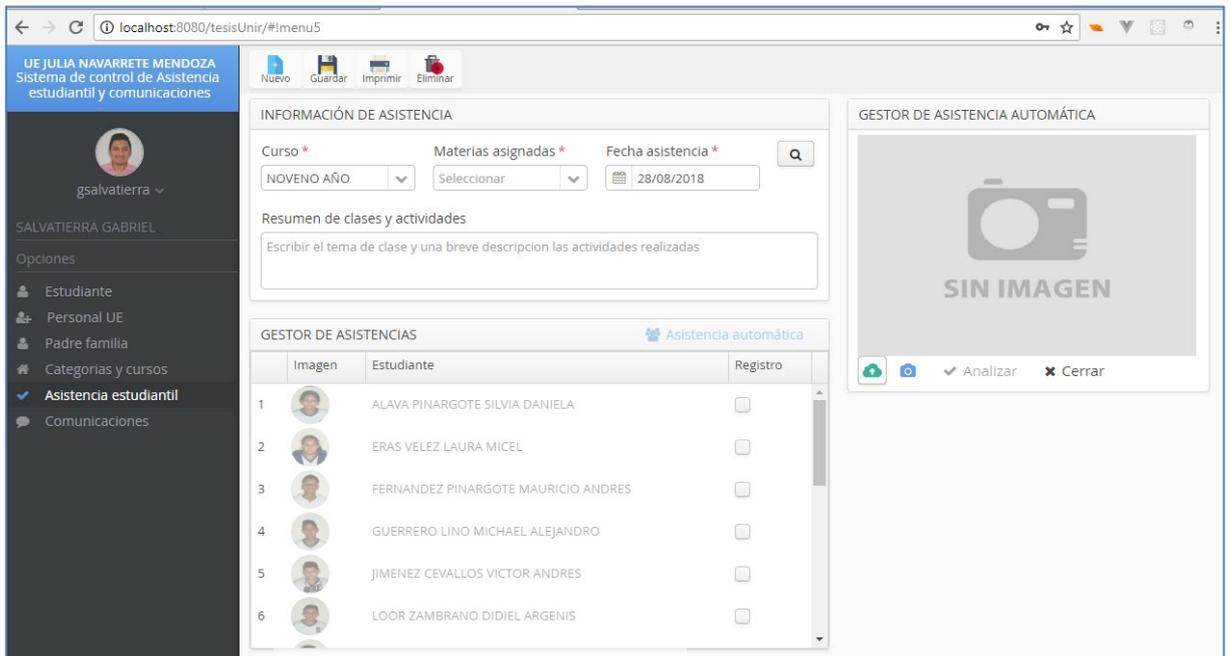
Vista Personal UE

Vista padre familia

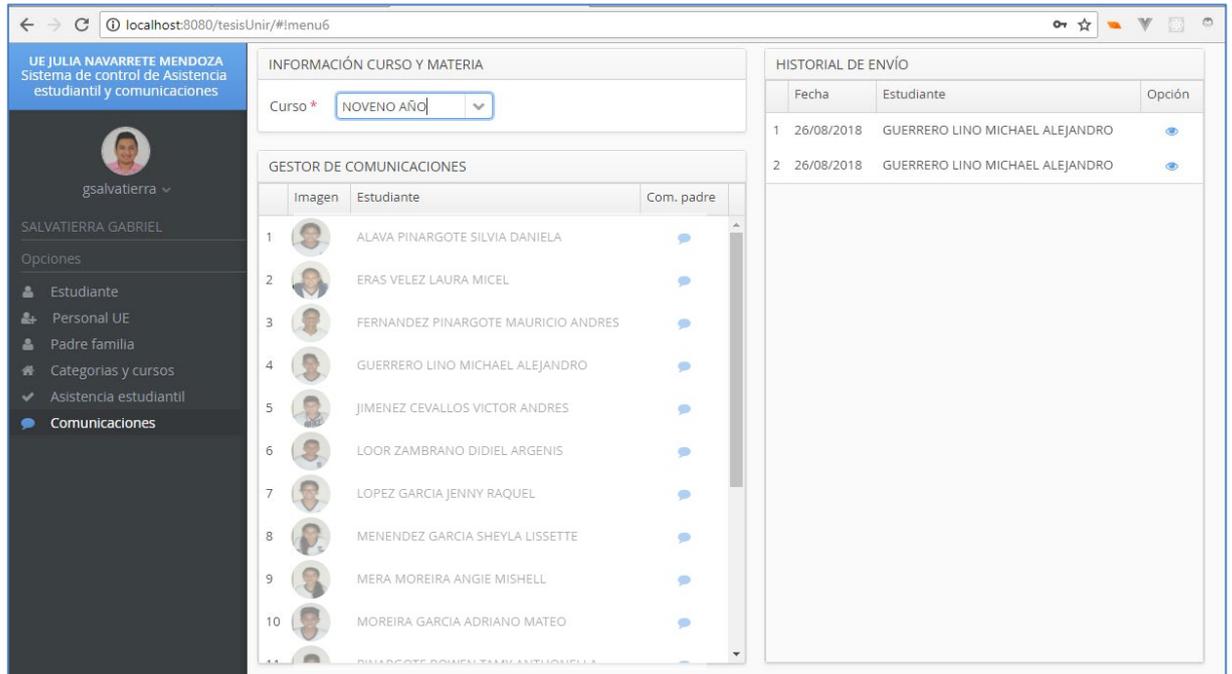
Vista categorías y cursos



Vista asistencia estudiantil



Vista comunicaciones



Anexo III. Ficha de aplicación de test SUMI

	Pregunta	De acuerdo	Indeciso	Desacuerdo
1	Este software responde muy lentamente a las entradas			
2	Las instrucciones y advertencias son útiles			
3	El software se detiene inesperadamente			
4	Es muy difícil aprender a manejar el software			
5	En algunos momentos no sé cuál es el siguiente paso en este software			
6	Disfruto el tiempo que dedico a usar este software			
7	La ayuda que brinda el software no es de mucha ayuda			
8	Si el software se detiene, no es fácil reiniciarlo			
9	Toma mucho tiempo aprender las funciones del software			
10	A veces me pregunto si estoy usando la función correcta			
11	Trabajar con este software es muy satisfactorio			
12	La forma en que se presenta la información del sistema es clara y comprensible			
13	Me siento más seguro si solo uso algunas funciones familiares			
14	Trabajar con este software es mentalmente estimulante			
15	Nunca hay suficiente información en la pantalla cuando se necesita			
16	Yo siento que controlo el software cuando lo utilizo			

17	Creo que este software es inconsistente			
18	No me gustaría usar este software todos los días			
19	Yo puedo entender y trabajar con la información que me brinda el software			
20	El software es frustrante			
21	Hay mucho que leer antes de poder usar el software			
22	La velocidad de este software es lo suficientemente rápido			
23	Habido momentos en el uso de este software cuando me he sentido bastante tenso o tensa			
24	La organización de los menús parece bastante lógico			
25	El software me permite ser económico en las pulsaciones de teclas			
26	Aprender a usar nuevas funciones es difícil			
27	Hay demasiados pasos requeridos para conseguir trabajar			
28	Creo que este software a veces me ha dado un dolor de cabeza			
29	Los mensajes de error no son adecuados			
30	Es fácil hacer que el software haga exactamente lo que quiero			
31	Nunca aprenderé a usar todo lo que se ofrece en este software			
32	El software no siempre ha hecho lo que esperaba			
33	El software se presenta de una manera muy atractiva			
34	Es fácil moverse de una tarea a otra			
35	El fácil olvidar cómo hacer las cosas con este software			
36	Este software ocasionalmente se comporta de una manera que no se puede entender			
37	Este software es realmente muy incómodo			
38	Es fácil ver de un vistazo cuales son las opciones que ofrece el software			
39	Obtener y guardar información es difícil en este software			
40	Tengo que buscar asistencia la mayoría de la veces cuando uso este software			

Anexo IV. Evidencias

FOTO N°1



Descripción: Aplicación del test SUMI

FOTO N°2

Descripción: Captura de fotos para el servidor de reconocimiento

FOTO N°3

Descripción: Pruebas del sistema

FOTO N°4

Descripción: Instalación de cámara IP

Anexo V. Diccionario de datos

<p style="text-align: center;">Sybase PowerDesigner Physical Data Model Report</p>

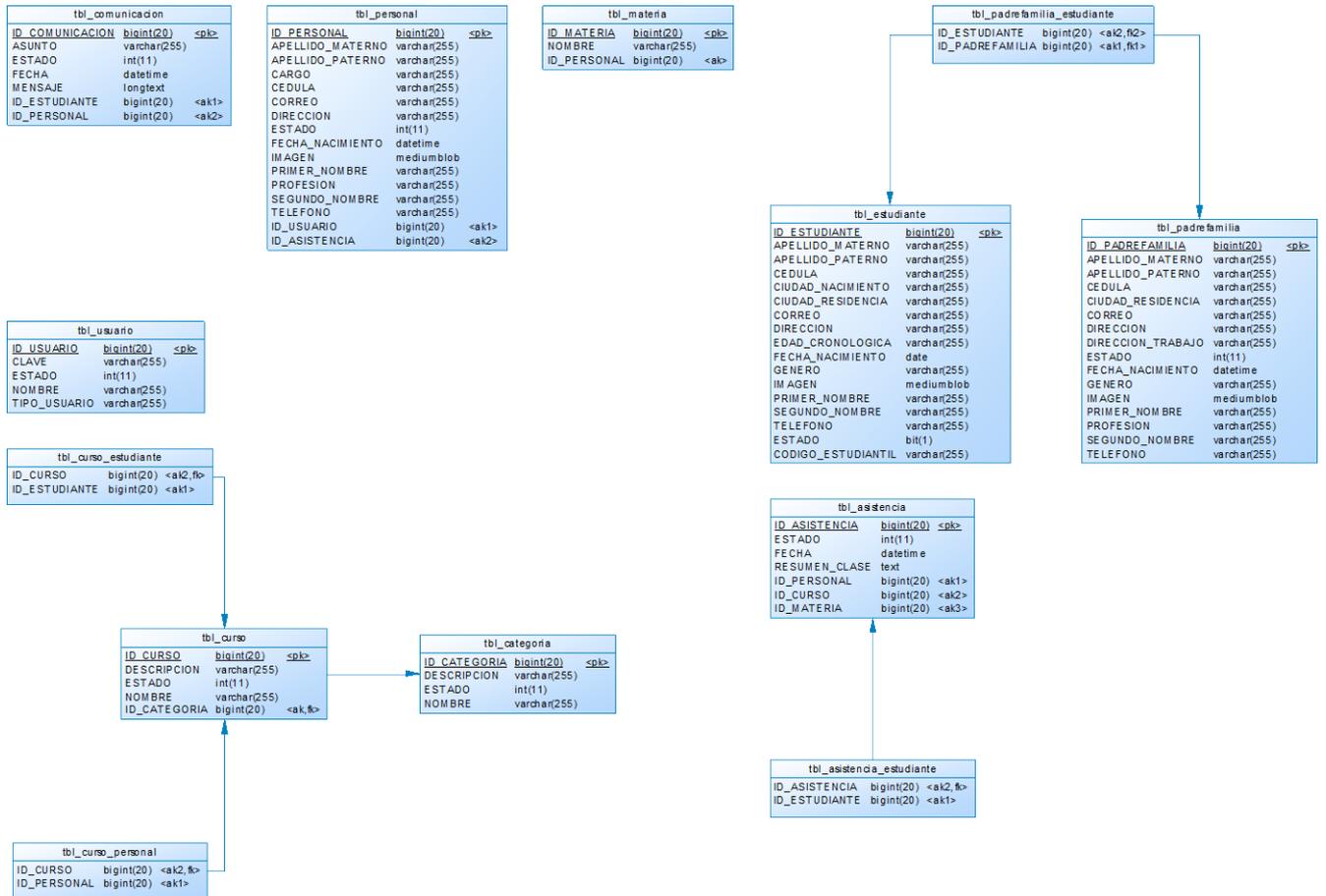
Model:	DB FINAL
Report:	Report
Author:	Gabriel S
Version:	1.0
Date:	30/8/2018

Short model description

List of diagrams

Name	Code
PhysicalDiagram_1	PHYSICALDIAGRAM_1

Diagram PhysicalDiagram_1



List of tables in diagram

Name	Code
tbl_asistencia	tbl_asistencia
tbl_asistencia_estudiante	tbl_asistencia_estudiante
tbl_categoria	tbl_categoria
tbl_comunicacion	tbl_comunicacion
tbl_curso	tbl_curso
tbl_curso_estudiante	tbl_curso_estudiante
tbl_curso_personal	tbl_curso_personal
tbl_estudiante	tbl_estudiante
tbl_materia	tbl_materia
tbl_padrefamilia	tbl_padrefamilia

tbl_padrefamilia_estudiante	tbl_padrefamilia_estudiante
tbl_personal	tbl_personal
tbl_usuario	tbl_usuario

List of references in diagram

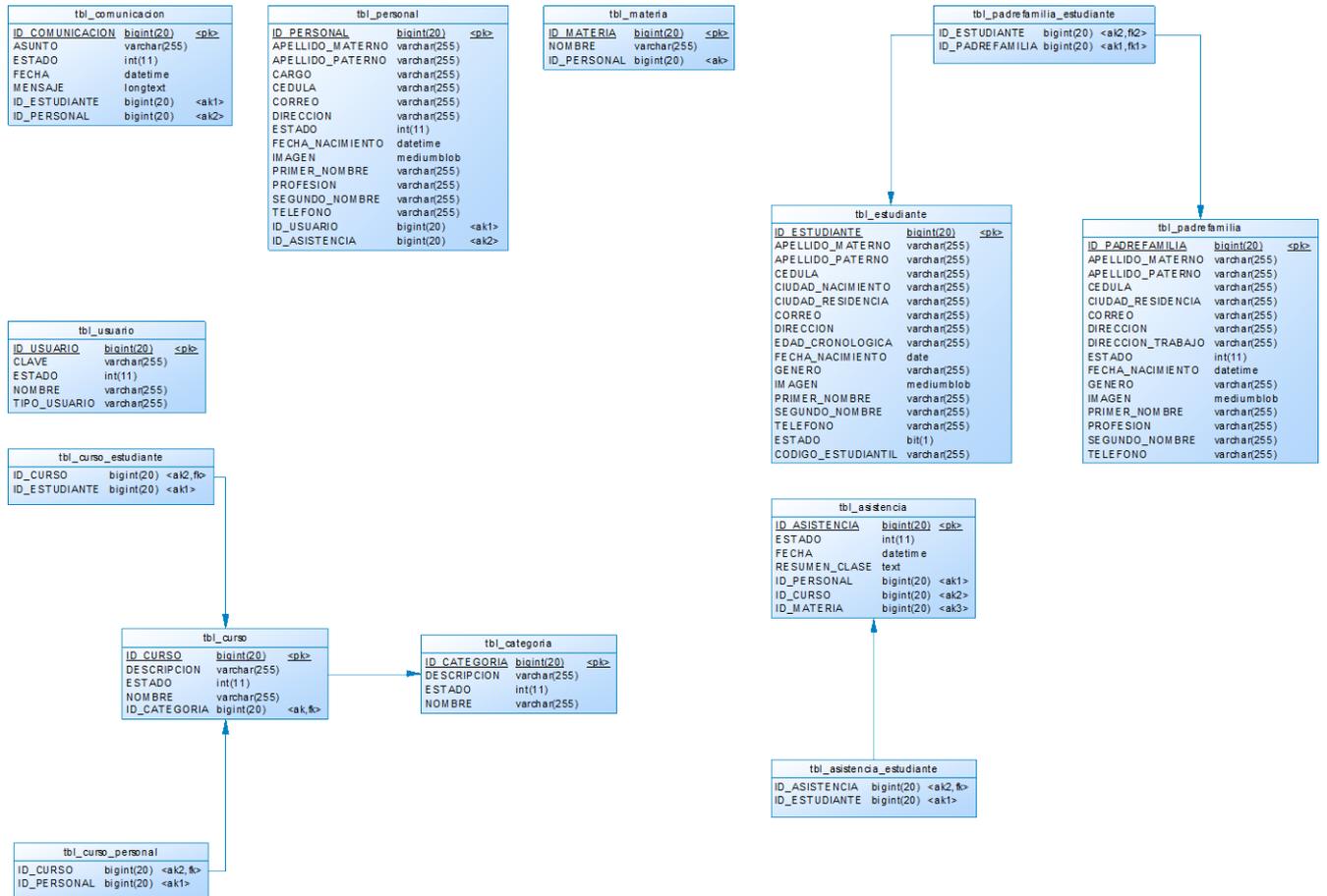
Name	Code	Parent Table	Child Table	Foreign Key Columns	Parent Role	Child Role
FK11xawx1d23h3n7wbms2e7jv2	FK11xawx1d23h3n7wbms2e7jv2	tbl_curso	tbl_curso_personal	ID_CURSO		
FK99c8rqdm85d5j0a7qt8a7miwd	FK99c8rqdm85d5j0a7qt8a7miwd	tbl_asistencia	tbl_asistencia_estudiante	ID_ASISTENCIA		
FKb18o39syi39j2g6xy7m7d1gf	FKb18o39syi39j2g6xy7m7d1gf	tbl_padrefamilia	tbl_padrefamilia_estudiante	ID_PADREFAMILIA		
FKcwaavcilwieiajnt0hjc6rce	FKcwaavcilwieiajnt0hjc6rce	tbl_categoria	tbl_curso	ID_CATEGORIA		
FKg2y2gax6xjkhkikm079i9p01pu	FKg2y2gax6xjkhkikm079i9p01pu	tbl_curso	tbl_curso_estudiante	ID_CURSO		
FKsc337y7qfvknuw4v9uuvepxg0	FKsc337y7qfvknuw4v9uuvepxg0	tbl_estudiante	tbl_padrefamilia_estudiante	ID_ESTUDIANTE		

Full model description

List of diagrams

Name	Code
PhysicalDiagram_1	PHYSICALDIAGRAM_1

Diagram PhysicalDiagram_1



List of tables in diagram

Name	Code
tbl_asistencia	tbl_asistencia
tbl_asistencia_estudiante	tbl_asistencia_estudiante
tbl_categoria	tbl_categoria
tbl_comunicacion	tbl_comunicacion
tbl_curso	tbl_curso
tbl_curso_estudiante	tbl_curso_estudiante
tbl_curso_personal	tbl_curso_personal
tbl_estudiante	tbl_estudiante
tbl_materia	tbl_materia
tbl_padrefamilia	tbl_padrefamilia
tbl_padrefamilia_estudiante	tbl_padrefamilia_estudiante
tbl_personal	tbl_personal

tbl_usuario

tbl_usuario

Table tbl_asistencia**Card of table tbl_asistencia**

Name	tbl_asistencia
Code	tbl_asistencia

Server validation rule of table tbl_asistencia

%RULES%

Options of table tbl_asistencia

ENGINE=InnoDB AUTO_INCREMENT=15 DEFAULT CHARSET=utf8

Check constraint name of table tbl_asistencia

TK_tbl_asistencia

List of incoming references of the table tbl_asistencia

Name	Code	Child Table	Foreign Key Columns	Parent Role	Child Role
FK99c8rqd m85d5j0a7q t8a7miwd	FK99c8rqdm 85d5j0a7qt8a 7miwd	tbl_asistencia _estudiante	ID_ASISTEN CIA		

List of columns of the table tbl_asistencia

Name	Code
ID_ASISTENCIA	ID_ASISTENCIA
ESTADO	ESTADO
FECHA	FECHA
RESUMEN_CLASE	RESUMEN_CLASE
ID_PERSONAL	ID_PERSONAL
ID_CURSO	ID_CURSO
ID_MATERIA	ID_MATERIA

List of keys of the table tbl_asistencia

Name	Code	Primary
Key_1	Key_1	X
FKbj6isytdkq3hghpsn75vj8rc9	FKbj6isytdkq3hghpsn75vj8rc9	
FKh2e91do2iev3atxopto7hey7w	FKh2e91do2iev3atxopto7hey7w	
FKonu4i2jl9qhe9wq0uikqptg5h	FKonu4i2jl9qhe9wq0uikqptg5h	

Table tbl_asistencia_estudiante**Card of table tbl_asistencia_estudiante**

Name	tbl_asistencia_estudiante
Code	tbl_asistencia_estudiante

Server validation rule of table tbl_asistencia_estudiante

%RULES%

Options of table tbl_asistencia_estudiante

ENGINE=InnoDB DEFAULT CHARSET=utf8

Check constraint name of table tbl_asistencia_estudiante

TK_tbl_asistencia_estudiante

List of outgoing references of the table tbl_asistencia_estudiante

Name	Code	Parent Table	Foreign Key Columns	Parent Role	Child Role
FK99c8rqdm85d5j0a7qt8a7miwd	FK99c8rqdm85d5j0a7qt8a7miwd	tbl_asistencia	ID_ASISTENCIA		

List of columns of the table tbl_asistencia_estudiante

Name	Code
ID_ASISTENCIA	ID_ASISTENCIA
ID_ESTUDIANTE	ID_ESTUDIANTE

List of keys of the table tbl_asistencia_estudiante

Name	Code	Primary
FK55oicnkx88n2ikfmb4b4nx20	FK55oicnkx88n2ikfmb4b4nx20	
FK99c8rqdm85d5j0a7qt8a7miwd	FK99c8rqdm85d5j0a7qt8a7miwd	

Table tbl_categoria**Card of table tbl_categoria**

Name	tbl_categoria
Code	tbl_categoria

Server validation rule of table tbl_categoria

%RULES%

Options of table tbl_categoria

ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8

Check constraint name of table tbl_categoria

TK_tbl_categoria

List of incoming references of the table *tbl_categoria*

Name	Code	Child Table	Foreign Key Columns	Parent Role	Child Role
FKcwaavcilwieyiajnt0hjc6rce	FKcwaavcilwieyiajnt0hjc6rce	tbl_curso	ID_CATEGORIA		

List of columns of the table *tbl_categoria*

Name	Code
ID_CATEGORIA	ID_CATEGORIA
DESCRIPCION	DESCRIPCION
ESTADO	ESTADO
NOMBRE	NOMBRE

List of keys of the table *tbl_categoria*

Name	Code	Primary
Key_1	Key_1	X

Table *tbl_comunicacion***Card of table *tbl_comunicacion***

Name	tbl_comunicacion
Code	tbl_comunicacion

Server validation rule of table *tbl_comunicacion*

%RULES%

Options of table *tbl_comunicacion*

ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8

Check constraint name of table *tbl_comunicacion*

TK_tbl_comunicacion

List of columns of the table *tbl_comunicacion*

Name	Code
ID_COMUNICACION	ID_COMUNICACION
ASUNTO	ASUNTO
ESTADO	ESTADO
FECHA	FECHA
MENSAJE	MENSAJE
ID_ESTUDIANTE	ID_ESTUDIANTE
ID_PERSONAL	ID_PERSONAL

List of keys of the table *tbl_comunicacion*

Name	Code	Primary
Key_1	Key_1	X
FKhvv5mmnl5pjbct1em5i5vy80y	FKhvv5mmnl5pjbct1em5i5vy80y	
FKlfw0b5bbajg0oju1qejfixvwg	FKlfw0b5bbajg0oju1qejfixvwg	

Table *tbl_curso*

Card of table *tbl_curso*

Name	tbl_curso
Code	tbl_curso

Server validation rule of table *tbl_curso*

%RULES%

Options of table *tbl_curso*

ENGINE=InnoDB AUTO_INCREMENT=10 DEFAULT CHARSET=utf8

Check constraint name of table *tbl_curso*

TK_tbl_curso

List of incoming references of the table *tbl_curso*

Name	Code	Child Table	Foreign Key Columns	Parent Role	Child Role
FK1lxawxld23h3n7wbms2e7jjv2	FK1lxawxld23h3n7wbms2e7jjv2	tbl_curso_personal	ID_CURSO		
FKgjy2gax6xjhkikm079i9p01pu	FKgjy2gax6xjhkikm079i9p01pu	tbl_curso_estudiante	ID_CURSO		

List of outgoing references of the table *tbl_curso*

Name	Code	Parent Table	Foreign Key Columns	Parent Role	Child Role
FKcwaavcilwieyiajnt0hjc6rce	FKcwaavcilwieyiajnt0hjc6rce	tbl_categoria	ID_CATEGORIA		

List of columns of the table *tbl_curso*

Name	Code
ID_CURSO	ID_CURSO
DESCRIPCION	DESCRIPCION
ESTADO	ESTADO

NOMBRE ID_CATEGORIA	NOMBRE ID_CATEGORIA
------------------------	------------------------

List of keys of the table tbl_curso

Name	Code	Primary
Key_1 FKcwaavcilwieyiajnt0hjc6rce	Key_1 FKcwaavcilwieyiajnt0hjc6rce	X

Table tbl_curso_estudiante**Card of table tbl_curso_estudiante**

Name	tbl_curso_estudiante
Code	tbl_curso_estudiante

Server validation rule of table tbl_curso_estudiante

%RULES%

Options of table tbl_curso_estudiante

ENGINE=InnoDB DEFAULT CHARSET=utf8

Check constraint name of table tbl_curso_estudiante

TK_tbl_curso_estudiante

List of outgoing references of the table tbl_curso_estudiante

Name	Code	Parent Table	Foreign Key Columns	Parent Role	Child Role
FKggy2gax6xjhkikm079i9p01pu	FKggy2gax6xjhkikm079i9p01pu	tbl_curso	ID_CURSO		

List of columns of the table tbl_curso_estudiante

Name	Code
ID_CURSO ID_ESTUDIANTE	ID_CURSO ID_ESTUDIANTE

List of keys of the table tbl_curso_estudiante

Name	Code	Primary
FK8b1qq60c3dmfwyptn1nab4sm1 FKggy2gax6xjhkikm079i9p01pu	FK8b1qq60c3dmfwyptn1nab4sm1 FKggy2gax6xjhkikm079i9p01pu	

Table tbl_curso_personal

Card of table *tbl_curso_personal*

Name	tbl_curso_personal
Code	tbl_curso_personal

Server validation rule of table *tbl_curso_personal*

%RULES%

Options of table *tbl_curso_personal*

ENGINE=InnoDB DEFAULT CHARSET=utf8

Check constraint name of table *tbl_curso_personal*

TK_tbl_curso_personal

List of outgoing references of the table *tbl_curso_personal*

Name	Code	Parent Table	Foreign Key Columns	Parent Role	Child Role
FK11xawxld23h3n7wbms2e7jjv2	FK11xawxld23h3n7wbms2e7jjv2	tbl_curso	ID_CURSO		

List of columns of the table *tbl_curso_personal*

Name	Code
ID_CURSO	ID_CURSO
ID_PERSONAL	ID_PERSONAL

List of keys of the table *tbl_curso_personal*

Name	Code	Primary
FKb0d9pd7urh8wdgk4lfxramypo	FKb0d9pd7urh8wdgk4lfxramypo	
FK11xawxld23h3n7wbms2e7jjv2	FK11xawxld23h3n7wbms2e7jjv2	

Table *tbl_estudiante***Card of table *tbl_estudiante***

Name	tbl_estudiante
Code	tbl_estudiante

Server validation rule of table *tbl_estudiante*

%RULES%

Options of table *tbl_estudiante*

ENGINE=InnoDB AUTO_INCREMENT=543 DEFAULT CHARSET=utf8

Check constraint name of table *tbl_estudiante*

TK_tbl_estudiante

List of incoming references of the table *tbl_estudiante*

Name	Code	Child Table	Foreign Key Columns	Parent Role	Child Role
FKsc337y7qfvknuw4v9uuvepxg0	FKsc337y7qfvknuw4v9uuvepxg0	tbl_padrefamilia_estudiante	ID_ESTUDIANTE		

List of columns of the table *tbl_estudiante*

Name	Code
ID_ESTUDIANTE	ID_ESTUDIANTE
APELLIDO_MATERNO	APELLIDO_MATERNO
APELLIDO_PATERNO	APELLIDO_PATERNO
CEDULA	CEDULA
CIUDAD_NACIMIENTO	CIUDAD_NACIMIENTO
CIUDAD_RESIDENCIA	CIUDAD_RESIDENCIA
CORREO	CORREO
DIRECCION	DIRECCION
EDAD_CRONOLOGICA	EDAD_CRONOLOGICA
FECHA_NACIMIENTO	FECHA_NACIMIENTO
GENERO	GENERO
IMAGEN	IMAGEN
PRIMER_NOMBRE	PRIMER_NOMBRE
SEGUNDO_NOMBRE	SEGUNDO_NOMBRE
TELEFONO	TELEFONO
ESTADO	ESTADO
CODIGO_ESTUDIANTIL	CODIGO_ESTUDIANTIL

List of keys of the table *tbl_estudiante*

Name	Code	Primary
Key_1	Key_1	X

Table *tbl_materia*

Card of table *tbl_materia*

Name	tbl_materia
Code	tbl_materia

Server validation rule of table *tbl_materia*

%RULES%

Options of table *tbl_materia*

ENGINE=InnoDB AUTO_INCREMENT=11 DEFAULT CHARSET=utf8

Check constraint name of table *tbl_materia*

TK_tbl_materia

List of columns of the table *tbl_materia*

Name	Code
ID_MATERIA	ID_MATERIA
NOMBRE	NOMBRE
ID_PERSONAL	ID_PERSONAL

List of keys of the table *tbl_materia*

Name	Code	Primary
Key_1	Key_1	X
FKr67ve7u8xge7y8aa4vodsmxwy	FKr67ve7u8xge7y8aa4vodsmxwy	

Table *tbl_padrefamilia***Card of table *tbl_padrefamilia***

Name	tbl_padrefamilia
Code	tbl_padrefamilia

Server validation rule of table *tbl_padrefamilia*

%RULES%

Options of table *tbl_padrefamilia*

ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8

Check constraint name of table *tbl_padrefamilia*

TK_tbl_padrefamilia

List of incoming references of the table *tbl_padrefamilia*

Name	Code	Child Table	Foreign Key Columns	Parent Role	Child Role
FKb18o39syi39j2g6xyy7m7d1gf	FKb18o39syi39j2g6xyy7m7d1gf	tbl_padrefamilia_estudiante	ID_PADREFAMILIA		

List of columns of the table *tbl_padrefamilia*

Name	Code
ID_PADREFAMILIA	ID_PADREFAMILIA
APELLIDO_MATERNO	APELLIDO_MATERNO
APELLIDO_PATERNO	APELLIDO_PATERNO
CEDULA	CEDULA

CIUDAD_RESIDENCIA	CIUDAD_RESIDENCIA
CORREO	CORREO
DIRECCION	DIRECCION
DIRECCION_TRABAJO	DIRECCION_TRABAJO
ESTADO	ESTADO
FECHA_NACIMIENTO	FECHA_NACIMIENTO
GENERO	GENERO
IMAGEN	IMAGEN
PRIMER_NOMBRE	PRIMER_NOMBRE
PROFESION	PROFESION
SEGUNDO_NOMBRE	SEGUNDO_NOMBRE
TELEFONO	TELEFONO

List of keys of the table *tbl_padrefamilia*

Name	Code	Primary
Key_1	Key_1	X

Table *tbl_padrefamilia_estudiante***Card of table *tbl_padrefamilia_estudiante***

Name	tbl_padrefamilia_estudiante
Code	tbl_padrefamilia_estudiante

Server validation rule of table *tbl_padrefamilia_estudiante*

%RULES%

Options of table *tbl_padrefamilia_estudiante*

ENGINE=InnoDB DEFAULT CHARSET=utf8

Check constraint name of table *tbl_padrefamilia_estudiante*

TK_tbl_padrefamilia_estudiante

List of outgoing references of the table *tbl_padrefamilia_estudiante*

Name	Code	Parent Table	Foreign Key Columns	Parent Role	Child Role
FKbl8o39syi39j2g6xyy7m7d1gf	FKbl8o39syi39j2g6xyy7m7d1gf	tbl_padrefamilia	ID_PADREFAMILIA		
FKsc337y7qfvknuw4v9uuvepxg0	FKsc337y7qfvknuw4v9uuvepxg0	tbl_estudiante	ID_ESTUDIANTE		

List of columns of the table *tbl_padrefamilia_estudiante*

Name	Code
------	------

ID_ESTUDIANTE	ID_ESTUDIANTE
ID_PADREFAMILIA	ID_PADREFAMILIA

List of keys of the table *tbl_padrefamilia_estudiante*

Name	Code	Primary
FKbl8o39syi39j2g6xyy7m7d1gf	FKbl8o39syi39j2g6xyy7m7d1gf	
FKsc337y7qfvknuw4v9uuvepxg0	FKsc337y7qfvknuw4v9uuvepxg0	

Table *tbl_personal*

Card of table *tbl_personal*

Name	tbl_personal
Code	tbl_personal

Server validation rule of table *tbl_personal*

%RULES%

Options of table *tbl_personal*

ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8

Check constraint name of table *tbl_personal*

TK_tbl_personal

List of columns of the table *tbl_personal*

Name	Code
ID_PERSONAL	ID_PERSONAL
APELLIDO_MATERNO	APELLIDO_MATERNO
APELLIDO_PATERNO	APELLIDO_PATERNO
CARGO	CARGO
CEDULA	CEDULA
CORREO	CORREO
DIRECCION	DIRECCION
ESTADO	ESTADO
FECHA_NACIMIENTO	FECHA_NACIMIENTO
IMAGEN	IMAGEN
PRIMER_NOMBRE	PRIMER_NOMBRE
PROFESION	PROFESION
SEGUNDO_NOMBRE	SEGUNDO_NOMBRE
TELEFONO	TELEFONO
ID_USUARIO	ID_USUARIO
ID_ASISTENCIA	ID_ASISTENCIA

List of keys of the table *tbl_personal*

Name	Code	Primary
------	------	---------

Key_1 FK87sxjybgr1hej8ht5mr2gie3n FK3iue9jfrvy0m4y7puc4407n0d	Key_1 FK87sxjybgr1hej8ht5mr2gie3n FK3iue9jfrvy0m4y7puc4407n0d	X
---	---	---

Table tbl_usuario**Card of table tbl_usuario**

Name	tbl_usuario
Code	tbl_usuario

Server validation rule of table tbl_usuario

%RULES%

Options of table tbl_usuario

ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8

Check constraint name of table tbl_usuario

TK_tbl_usuario

List of columns of the table tbl_usuario

Name	Code
ID_USUARIO	ID_USUARIO
CLAVE	CLAVE
ESTADO	ESTADO
NOMBRE	NOMBRE
TIPO_USUARIO	TIPO_USUARIO

List of keys of the table tbl_usuario

Name	Code	Primary
Key_1	Key_1	X

List of references in diagram

Name	Code	Parent Table	Child Table	Foreign Key Columns	Parent Role	Child Role
FK1lxawxld23h3n7wbms2e7jjv2	FK1lxawxld23h3n7wbms2e7jjv2	tbl_curso	tbl_curso_personal	ID_CURSO		
FK99c8rqdm85d5j0a7qt8a7miwd	FK99c8rqdm85d5j0a7qt8a7miwd	tbl_asistencia	tbl_asistencia_estudiante	ID_ASISTENCIA		
FKbl8o39syi39j2g6xyy7m7d1gf	FKbl8o39syi39j2g6xyy7m7d1gf	tbl_padrefamilia	tbl_padrefamilia_estudiante	ID_PADREFAMILIA		
FKcwaavcilwiewiajnt0hjc6rce	FKcwaavcilwiewiajnt0hjc6rce	tbl_categoria	tbl_curso	ID_CATEGORIA		

FKggy2gax6xjhkikm079i9p01pu	FKggy2gax6xjhkikm079i9p01pu	tbl_curso	tbl_curso_estudiante	ID_CURSO		
FKsc337y7qfvknuw4v9uuv	FKsc337y7qfvknuw4v9uuv	tbl_estudiante	tbl_padrefamilia_estudiante	ID_ESTUDIANTE		

Reference FK1lxawxld23h3n7wbms2e7jjv2

Card of reference FK1lxawxld23h3n7wbms2e7jjv2

Name	FK1lxawxld23h3n7wbms2e7jjv2
Code	FK1lxawxld23h3n7wbms2e7jjv2
Child Table	tbl_curso_personal
Parent Table	tbl_curso
Foreign Key Columns	ID_CURSO
Parent Role	
Child Role	

List of reference joins of the reference FK1lxawxld23h3n7wbms2e7jjv2

Display Name	Parent Table Column	Child Table Column
ID_CURSO = ID_CURSO	ID_CURSO	ID_CURSO

Reference FK99c8rqdm85d5j0a7qt8a7miwd

Card of reference FK99c8rqdm85d5j0a7qt8a7miwd

Name	FK99c8rqdm85d5j0a7qt8a7miwd
Code	FK99c8rqdm85d5j0a7qt8a7miwd
Child Table	tbl_asistencia_estudiante
Parent Table	tbl_asistencia
Foreign Key Columns	ID_ASISTENCIA
Parent Role	
Child Role	

List of reference joins of the reference FK99c8rqdm85d5j0a7qt8a7miwd

Display Name	Parent Table Column	Child Table Column
ID_ASISTENCIA = ID_ASISTENCIA	ID_ASISTENCIA	ID_ASISTENCIA

Reference FKbl8o39syi39j2g6xyy7m7d1gf

Card of reference FKbl8o39syi39j2g6xyy7m7d1gf

Name	FKbl8o39syi39j2g6xyy7m7d1gf
Code	FKbl8o39syi39j2g6xyy7m7d1gf
Child Table	tbl_padrefamilia_estudiante

Parent Table	tbl_padrefamilia
Foreign Key Columns	ID_PADREFAMILIA
Parent Role	
Child Role	

List of reference joins of the reference FKbl8o39syi39j2g6xyy7m7d1gf

Display Name	Parent Table Column	Child Table Column
ID_PADREFAMILIA = ID_PADREFAMILIA	ID_PADREFAMILIA	ID_PADREFAMILIA

Reference FKcwaavcilwieyiajnt0hjc6rce

Card of reference FKcwaavcilwieyiajnt0hjc6rce

Name	FKcwaavcilwieyiajnt0hjc6rce
Code	FKcwaavcilwieyiajnt0hjc6rce
Child Table	tbl_curso
Parent Table	tbl_categoria
Foreign Key Columns	ID_CATEGORIA
Parent Role	
Child Role	

List of reference joins of the reference FKcwaavcilwieyiajnt0hjc6rce

Display Name	Parent Table Column	Child Table Column
ID_CATEGORIA = ID_CATEGORIA	ID_CATEGORIA	ID_CATEGORIA

Reference FKggy2gax6xjhkikm079i9p01pu

Card of reference FKggy2gax6xjhkikm079i9p01pu

Name	FKggy2gax6xjhkikm079i9p01pu
Code	FKggy2gax6xjhkikm079i9p01pu
Child Table	tbl_curso_estudiante
Parent Table	tbl_curso
Foreign Key Columns	ID_CURSO
Parent Role	
Child Role	

List of reference joins of the reference FKggy2gax6xjhkikm079i9p01pu

Display Name	Parent Table Column	Child Table Column
ID_CURSO = ID_CURSO	ID_CURSO	ID_CURSO

Reference FKsc337y7qfvknuw4v9uuvepxg0

Card of reference FKsc337y7qfvknuw4v9uuvepxg0

Name	FKsc337y7qfvknuw4v9uuvepxg0
Code	FKsc337y7qfvknuw4v9uuvepxg0
Child Table	tbl_padrefamilia_estudiante
Parent Table	tbl_estudiante
Foreign Key Columns	ID_ESTUDIANTE
Parent Role	
Child Role	

List of reference joins of the reference FKsc337y7qfvknuw4v9uuvepxg0

Display Name	Parent Table Column	Child Table Column
ID_ESTUDIANTE = ID_ESTUDIANTE	ID_ESTUDIANTE	ID_ESTUDIANTE

